7.5

Protezione di IBM WebSphere MQ

Nota

Prima di utilizzare queste informazioni e il prodotto che supportano, leggere le informazioni in “Informazioni particolari” a pagina 327.

Questa edizione si applica alla versione 7 release 5 di IBM® WebSphere MQ e a tutte le release e modifiche successive, se non diversamente indicato nelle nuove edizioni.

Quando si inviano informazioni a IBM, si concede a IBM un diritto non esclusivo di utilizzare o distribuire le informazioni in qualsiasi modo ritenga appropriato senza incorrere in alcun obbligo verso l'utente.

Indice

Sicurezza..... 5
Panoramica della sicurezza..... 5
Concetti e meccanismi..... 5
IBM WebSphere MQ..... 20
Pianificazione dei requisiti di sicurezza..... 46
Identificazione e autenticazione della pianificazione..... 47
Autorizzazione di pianificazione..... 49
Pianificazione della riservatezza..... 60
Pianificazione dell'integrità dei dati..... 68
Controllo pianificazione..... 68
Pianificazione della sicurezza per topologia..... 69
Firewall e pass-thru Internet..... 80
Configurazione della sicurezza..... 81
Impostazione della sicurezza su sistemi UNIX e Linuxe Windows..... 81
Impostazione della sicurezza su HP NSS..... 107
Impostazione della protezione del client IBM WebSphere MQ MQI..... 108
Impostazione delle comunicazioni per SSL o TLS su sistemi UNIX, Linux, and Windows..... 110
Utilizzo di SSL o TLS..... 111
Identificazione e autenticazione degli utenti..... 144
Utenti privilegiati..... 147
Identificazione e autenticazione degli utenti utilizzando la struttura MQCSP..... 147
Implementazione dell'identificazione e dell'autenticazione nelle uscite di sicurezza..... 147
Mappature di identità nelle uscite del messaggio..... 148
Associazione di identità nell'uscita API e nell'uscita incrociata API..... 149
Utilizzo dei certificati revocati..... 150
Autorizzazione dell'accesso agli oggetti..... 159
Controllo dell'accesso agli oggetti utilizzando OAM su sistemi UNIX, Linux e Windows..... 159
Concessione dell'accesso richiesto alle risorse..... 167
Autorizzazione per gestire IBM WebSphere MQ su sistemi UNIX, Linuxe Windows..... 197
Autorizzazione per gestire gli oggetti IBM WebSphere MQ..... 199
Implementazione del controllo accessi nelle uscite di sicurezza..... 204
Implementazione del controllo accessi nelle uscite dei messaggi..... 205
Implementazione del controllo accessi nell'uscita API e nell'uscita incrociata API..... 206
Riservatezza dei messaggi..... 206
Connessione di due gestori code mediante SSL o TLS..... 206
Connessione sicura di un client a un gestore code..... 212
Specifica di CipherSpecs..... 218
Reimpostazione delle chiavi segrete SSL..... 224
Implementazione della riservatezza nei programmi di uscita utente..... 225
Integrità dei dati dei messaggi..... 227
Connessione di due gestori code mediante SSL o TLS..... 227
Connessione sicura di un client a un gestore code..... 235
Specifica di CipherSpecs..... 241
Audit..... 245
Proteggere i cluster..... 245
Arresto dei messaggi di invio dei gestori code non autorizzati..... 245
Arresto dei gestori code non autorizzati che immettono messaggi nelle code..... 245
Autorizzazione all'inserimento di messaggi nelle code del cluster remoto..... 246
Impedire ai gestori code di unirsi a un cluster..... 247
Forzare i gestori code indesiderati a lasciare un cluster..... 248
Come impedire ai gestori code di ricevere messaggi..... 248
SSL e cluster..... 249

Sicurezza di pubblicazione/sottoscrizione....251
Impostazione della sicurezza di pubblicazione / sottoscrizione di esempio.... 259
Sicurezza sottoscrizione.... 269
IBM WebSphere MQ Advanced Message Security....271
Panoramica di IBM WebSphere MQ Advanced Message Security....271
Installazione di IBM WebSphere MQ Advanced Message Security....295
Utilizzo di keystore e certificati....296
Gestione delle politiche di protezione IBM WebSphere MQ Advanced Message Security....308
Problemi e soluzioni....324

Informazioni particolari....327
Informazioni sull'interfaccia di programmazione....328
Marchi.... 328

Sicurezza

La sicurezza è una questione importante da considerare sia per gli sviluppatori di applicazioni IBM WebSphere MQ che per gli amministratori di sistema responsabili della configurazione di autorizzazioni IBM WebSphere MQ.

Panoramica della sicurezza

Questa raccolta di argomenti introduce i concetti di sicurezza IBM WebSphere MQ .

I concetti e i meccanismi di sicurezza, in quanto si applicano a qualsiasi sistema di computer, vengono presentati per primi, seguiti da una discussione di tali meccanismi di sicurezza in quanto sono implementati in IBM WebSphere MQ.

Concetti e meccanismi di sicurezza

Questa raccolta di argomenti descrive gli aspetti della sicurezza da considerare nell'installazione di IBM WebSphere MQ .

Gli aspetti comunemente accettati della sicurezza sono i seguenti:

- “Identificazione e autenticazione” a pagina 5
- “Authorization” a pagina 6
- “Audit” a pagina 6
- “Riservatezza” a pagina 7
- “Integrità dati” a pagina 7

I *meccanismi di protezione* sono strumenti tecnici e tecniche utilizzati per implementare servizi di sicurezza. Un meccanismo potrebbe funzionare da solo, o con altri, per fornire un particolare servizio. Esempi di meccanismi di sicurezza comuni sono i seguenti:

- “Crittografia” a pagina 7
- “Digest di messaggi e firme digitali” a pagina 9
- “Certificati digitali” a pagina 9
- “PKI (Public Key Infrastructure)” a pagina 14

Quando si pianifica un'implementazione IBM WebSphere MQ , considerare quali meccanismi di sicurezza sono necessari per implementare quegli aspetti della sicurezza che sono importanti per l'utente. Per informazioni su cosa considerare dopo aver letto questi argomenti, consultare “Pianificazione dei requisiti di sicurezza” a pagina 46.

Concetti correlati

“Connessione di due gestori code mediante SSL o TLS” a pagina 206
Le comunicazioni sicure che utilizzano i protocolli di sicurezza crittografica SSL o TLS richiedono l'impostazione dei canali di comunicazione e la gestione dei certificati digitali che verranno utilizzati per l'autenticazione.

“Utilizzo di SSL o TLS” a pagina 111
Questi argomenti forniscono istruzioni per l'esecuzione di singole attività relative all'utilizzo di SSL o TLS con IBM WebSphere MQ.

Identificazione e autenticazione

Identificazione è la capacità di identificare in maniera univoca un utente di un sistema o di un'applicazione in esecuzione nel sistema. L' *autenticazione* è la capacità di dimostrare che un utente o un'applicazione è realmente la persona o ciò che tale applicazione dichiara di essere.

Ad esempio, considerare un utente che accede ad un sistema immettendo un ID utente e una password. Il sistema utilizza l'ID utente per identificare l'utente. Il sistema autentica l'utente al momento dell'accesso controllando che la password fornita sia corretta.

Non rifiuto

Il servizio *non - repudiation* può essere visualizzato come un'estensione del servizio di identificazione e autenticazione. In generale, il non rifiuto si applica quando i dati vengono trasmessi elettronicamente; ad esempio, un ordine a un broker di azioni per acquistare o vendere azioni o un ordine a una banca per trasferire fondi da un conto a un altro.

L'obiettivo generale del servizio di non rifiuto è quello di essere in grado di dimostrare che un particolare messaggio è associato a un particolare individuo.

Il servizio non di rifiuto può contenere più di un componente, dove ogni componente fornisce una funzione diversa. Se il mittente di un messaggio nega mai l'invio, il servizio di non rifiuto con *prova di origine* può fornire al ricevente una prova innegabile che il messaggio è stato inviato da quella particolare persona. Se il destinatario di un messaggio nega mai di riceverlo, il servizio di non rifiuto con *prova di consegna* può fornire al mittente una prova innegabile che il messaggio è stato ricevuto da quel particolare individuo.

In pratica, la prova con quasi il 100% di certezza, o prova innegabile, è un obiettivo difficile. Nel mondo reale, nulla è completamente sicuro. La gestione della sicurezza è più interessata alla gestione del rischio a un livello accettabile per il business. In un tale contesto, un'aspettativa più realistica del servizio di non ripudio è quella di essere in grado di fornire prove che siano ammissibili e che supportino il vostro caso in tribunale.

Il non rifiuto è un servizio di sicurezza rilevante in un ambiente IBM WebSphere MQ perché IBM WebSphere MQ è un mezzo per trasmettere i dati elettronicamente. Ad esempio, si potrebbe richiedere la prova contemporanea che un particolare messaggio è stato inviato o ricevuto da una domanda associata a un particolare individuo.

IBM WebSphere MQ con IBM WebSphere MQ Advanced Message Security non fornisce un servizio di non rifiuto come parte della sua funzione di base. Tuttavia, questa documentazione del prodotto contiene suggerimenti su come fornire il proprio servizio non di rifiuto all'interno di un ambiente WebSphere MQ scrivendo i propri programmi di uscita.

Concetti correlati

“Identificazione e autenticazione in IBM WebSphere MQ” a pagina 20
In IBM WebSphere MQ, è possibile implementare l'identificazione e l'autenticazione utilizzando le informazioni di contesto del messaggio e l'autenticazione reciproca.

Authorization

Autorizzazione protegge le risorse critiche in un sistema limitando l'accesso solo agli utenti autorizzati e alle rispettive applicazioni. Impedisce l'uso non autorizzato di una risorsa o l'uso non autorizzato di una risorsa.

Concetti correlati

“Autorizzazione in IBM WebSphere MQ” a pagina 21
È possibile utilizzare l'autorizzazione per limitare ciò che particolari individui o applicazioni possono fare nell'ambiente IBM WebSphere MQ .

Audit

Il *Controllo* è il processo di registrazione e controllo degli eventi per rilevare se si è verificata un'attività imprevista o non autorizzata o se è stato effettuato un tentativo di eseguire tale attività.

Per ulteriori informazioni su come impostare l'autorizzazione, consultare “Autorizzazione di pianificazione” a pagina 49 e gli argomenti secondari associati.

Concetti correlati

“Controllo in IBM WebSphere MQ” a pagina 21

IBM WebSphere MQ può emettere messaggi di evento per registrare che si è svolta un'attività insolita.

Riservatezza

Il servizio *riservatezza* protegge le informazioni sensibili dalla divulgazione non autorizzata.

Quando i dati sensibili vengono memorizzati localmente, i meccanismi di controllo degli accessi potrebbero essere sufficienti per proteggerli supponendo che i dati non possano essere letti se non è possibile accedervi. Se è richiesto un livello di sicurezza maggiore, i dati possono essere codificati.

Crittografare i dati sensibili quando vengono trasmessi su una rete di comunicazione, specialmente su una rete non sicura come Internet. In un ambiente di rete, i meccanismi di controllo degli accessi non sono efficaci contro i tentativi di intercettare i dati, come le intercettazioni.

Integrità dati

Il servizio *integrità dati* rileva se è stata effettuata una modifica non autorizzata dei dati.

Ci sono due modi in cui i dati potrebbero essere modificati: accidentalmente, attraverso errori hardware e di trasmissione o a causa di un attacco deliberato. Molti prodotti hardware e protocolli di trasmissione hanno meccanismi per rilevare e correggere errori hardware e di trasmissione. Lo scopo del servizio di integrità dei dati è rilevare un attacco deliberato.

Il servizio di integrità dei dati mira solo a rilevare se i dati sono stati modificati. Non mira a ripristinare i dati allo stato originale se sono stati modificati.

I meccanismi di controllo degli accessi possono contribuire all'integrità dei dati nella misura in cui i dati non possono essere modificati se l'accesso viene negato. Ma, come per la riservatezza, i meccanismi di controllo degli accessi non sono efficaci in un ambiente di rete.

Concetti crittografici

Questa raccolta di argomenti descrive i concetti di crittografia applicabili a WebSphere MQ.

Il termine *entità* viene utilizzato per fare riferimento a un gestore code, a un client MQI WebSphere MQ , a un singolo utente o a qualsiasi altro sistema in grado di scambiare messaggi.

Concetti correlati

“Crittografia in IBM WebSphere MQ” a pagina 22
IBM WebSphere MQ fornisce la crittografia utilizzando i protocolli SSL (Secure sockets layer) e TLS (Transport Security Layer).

Crittografia

La crittografia è il processo di conversione tra un testo leggibile, denominato *testo semplice*, e un formato illeggibile, denominato *testo crittografico*.

Ciò si verifica come segue:

1. Il mittente converte il messaggio in testo semplice in testo cifrato. Questa parte del processo è denominata *crittografia* (a volte *crittografia*).
2. Il testo cifrato viene trasmesso al destinatario.
3. Il destinatario converte nuovamente il messaggio in testo semplice. Questa parte del processo è denominata *decodifica* (a volte *decifrazione*).

Consultare il Glossario per una definizione di codifica.

La conversione implica una sequenza di operazioni matematiche che modificano l'aspetto del messaggio durante la trasmissione ma non influiscono sul contenuto. Le tecniche crittografiche possono garantire la riservatezza e proteggere i messaggi dalla visualizzazione non autorizzata (intercettazione), poiché un messaggio crittografato non è comprensibile. Le firme digitali, che forniscono una garanzia di integrità dei messaggi, utilizzano tecniche di codifica. Per ulteriori informazioni, fare riferimento a “Firme digitali in SSL e TLS” a pagina 19.

Le tecniche crittografiche implicano un algoritmo generale, reso specifico dall'uso delle chiavi. Esistono due classi di algoritmo:

- Quelli che richiedono a entrambe le parti di utilizzare la stessa chiave segreta. Gli algoritmi che utilizzano una chiave condivisa sono noti come algoritmi *simmetrici* . Figura 1 a pagina 8 illustra la codifica della chiave simmetrica.
- Quelli che utilizzano una chiave per la crittografia e una diversa per la decrittografia. Uno di questi deve essere tenuto segreto, ma l'altro può essere pubblico. Gli algoritmi che utilizzano coppie di chiave pubblica e privata sono noti come algoritmi *asimmetrici* . Figura 2 a pagina 8 illustra la codifica della chiave asimmetrica, nota anche come *codifica della chiave pubblica*.

Gli algoritmi di codifica e decodifica utilizzati possono essere pubblici, ma la chiave segreta condivisa e la chiave privata devono essere mantenute segrete.

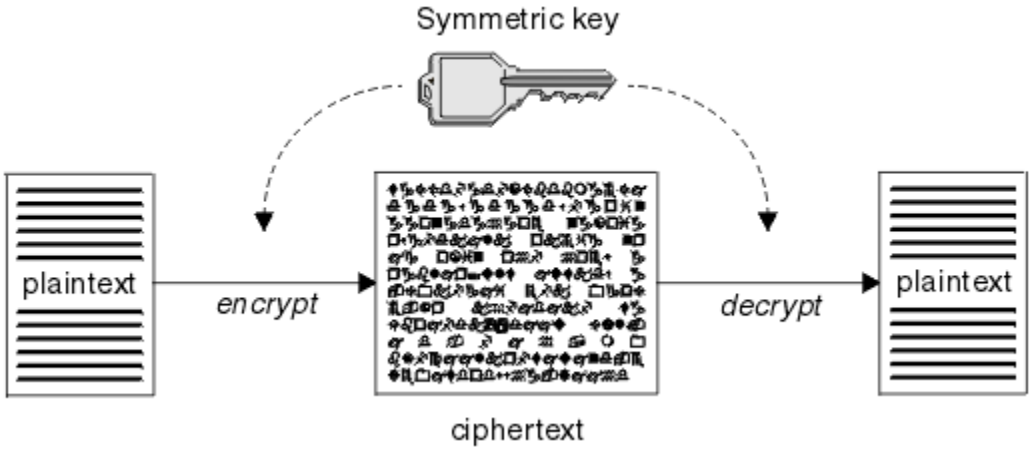


Figura 1. crittografia di chiavi simmetrica

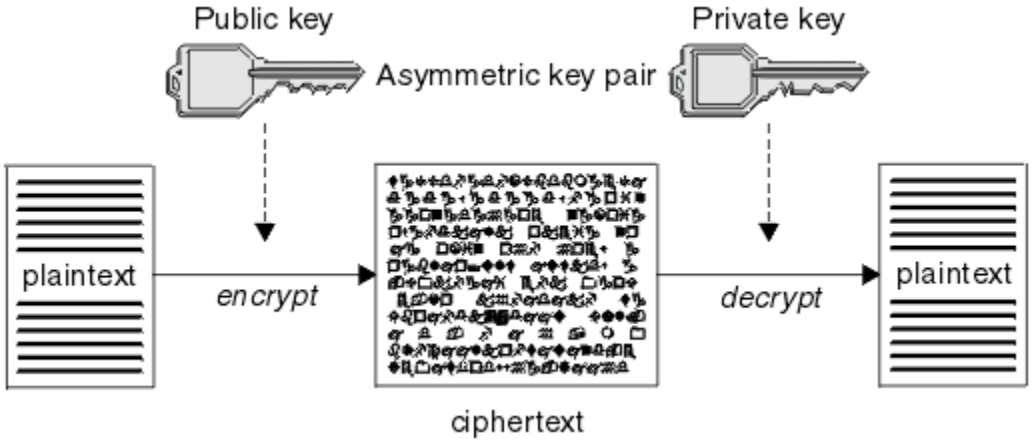


Figura 2. crittografia di chiavi asimmetrica

Figura 2 a pagina 8 mostra il testo semplice codificato con la chiave pubblica del ricevitore e decodificato con la chiave privata del destinatario. Solo il destinatario previsto conserva la chiave privata per decodificare il testo cifrato. Si noti che il mittente può anche crittografare i messaggi con una chiave privata, che consente a chiunque detenga la chiave pubblica del mittente di decrittografare il messaggio, con la certezza che il messaggio deve provenire dal mittente.

Con gli algoritmi asimmetrici, i messaggi vengono codificati con la chiave pubblica o privata, ma possono essere decodificati solo con l'altra chiave. Solo la chiave privata è segreta, la chiave pubblica può essere conosciuta da tutti. Con gli algoritmi simmetrici, la chiave condivisa deve essere nota solo alle due parti. Questo è denominato *problema di distribuzione della chiave* . Gli algoritmi asimmetrici sono più lenti ma hanno il vantaggio di non avere problemi di distribuzione delle chiavi.

Altra terminologia associata alla crittografia è:

Complessità
Il livello di crittografia è determinato dalla dimensione della chiave. Gli algoritmi asimmetrici richiedono chiavi di grandi dimensioni, ad esempio:

1024 bit	Chiave asimmetrica a bassa resistenza
2048 bit	Chiave asimmetrica di media intensità
4096 bit	Chiave asimmetrica ad alta resistenza

Le chiavi simmetriche sono più piccole: le chiavi a 256 bit forniscono una crittografia complessa.

Algoritmo di cifratura a blocchi
Questi algoritmi codificano i dati in base ai blocchi. Ad esempio, l'algoritmo RC2 di RSA Data Security Inc. utilizza blocchi di 8 byte. Gli algoritmi di blocco sono generalmente più lenti degli algoritmi di flusso.

Algoritmo di cifratura del flusso
Questi algoritmi operano su ogni byte di dati. Gli algoritmi di flusso sono generalmente più veloci degli algoritmi di blocco.

Digest di messaggi e firme digitali

Un digest del messaggio è una rappresentazione numerica a dimensione fissa del contenuto di un messaggio, calcolata da una funzione hash. Un digest del messaggio può essere codificato, formando una firma digitale.

La dimensione dei messaggi è intrinsecamente variabile. Un digest del messaggio è una rappresentazione numerica a dimensione fissa del contenuto di un messaggio. Un digest del messaggio viene calcolato da una funzione hash, che è una trasformazione che soddisfa due criteri:

- La funzione hash deve essere unidirezionale. Non deve essere possibile invertire la funzione per trovare il messaggio corrispondente ad un particolare digest del messaggio, se non verificando tutti i possibili messaggi.
- Deve essere computazionalmente non fattibile per trovare due messaggi che hash allo stesso digest.

Il digest del messaggio viene inviato con il messaggio stesso. Il destinatario può generare un digest per il messaggio e confrontarlo con il digest del mittente. L'integrità del messaggio viene verificata quando i due digest del messaggio sono uguali. Qualsiasi manomissione del messaggio durante la trasmissione risulta quasi certamente in un diverso digest del messaggio.

Un digest del messaggio creato utilizzando una chiave simmetrica segreta è noto come MAC (Message Authentication Code), perché può fornire la garanzia che il messaggio non è stato modificato.

Il mittente può anche generare un digest del messaggio e quindi codificare il digest utilizzando la chiave privata di una coppia di chiavi asimmetriche, formando una firma digitale. La firma deve quindi essere decodificata dal destinatario, prima di confrontarla con un digest generato localmente.

Concetti correlati

“Firme digitali in SSL e TLS” a pagina 19
Una firma digitale è formata dalla codifica di una rappresentazione di un messaggio. La crittografia utilizza la chiave privata del firmatario e, per efficienza, di solito opera su un digest del messaggio piuttosto che sul messaggio stesso.

Certificati digitali

I certificati digitali proteggono dall'impersonificazione, certificando che una chiave pubblica appartiene a un'entità specificata. Sono emessi da una CA (Certificate Authority).

I certificati digitali forniscono una protezione contro l'impersonificazione, poiché un certificato digitale collega una chiave pubblica al suo proprietario, se tale proprietario è un individuo, un gestore code o un'altra entità. I certificati digitali sono anche noti come certificati di chiave pubblica, poiché forniscono garanzie sulla proprietà di una chiave pubblica quando si utilizza uno schema di chiave asimmetrica. Un

certificato digitale contiene la chiave pubblica per un'entità ed è un'istruzione che indica che la chiave pubblica appartiene a tale entità:

- Quando il certificato è per una singola entità, il certificato viene denominato *certificato personale* o *certificato utente*.
- Quando il certificato è per una CA (Certificate Authority), il certificato viene denominato *certificato CA* o *certificato firmatario*.

Se le chiavi pubbliche vengono inviate direttamente dal loro proprietario a un'altra entità, c'è il rischio che il messaggio possa essere intercettato e la chiave pubblica sostituita da un'altra. Questo è noto come *man in the middle attack*. La soluzione a questo problema è scambiare le chiavi pubbliche tramite una terza parte attendibile, dandoti una forte garanzia che la chiave pubblica appartiene realmente all'entità con cui stai comunicando. Invece di inviare la tua chiave pubblica direttamente, chiedi alla terza parte attendibile di incorporarla in un certificato digitale. La terza parte attendibile che emette i certificati digitali è denominata CA (Certificate Authority), come descritto in "Autorità di certificazione (CA)" a pagina 11.

Contenuto di un certificato digitale
I certificati digitali contengono informazioni specifiche, determinate dallo standard X.509 .

I certificati digitali utilizzati da WebSphere MQ sono conformi allo standard X.509 , che specifica le informazioni richieste e il formato per inviarle. X.509 è la parte del framework di autenticazione della serie di standard X.500 .

I certificati digitali contengono almeno le seguenti informazioni sull'entità certificata:

- La chiave pubblica del proprietario
- Il DN (Distinguished Name) del proprietario
- Il DN della CA che ha emesso il certificato
- La data a partire dalla quale il certificato è valido
- La data di scadenza del certificato
- Il numero di versione del formato dati del certificato come definito in X.509. La versione corrente dello standard X.509 è la versione 3 e la maggior parte dei certificati è conforme a tale versione.
- Un numero di serie. Questo è un identificativo univoco assegnato dalla CA che ha emesso il certificato. Il numero di serie è univoco all'interno della CA che ha emesso il certificato: due certificati firmati dallo stesso certificato CA non hanno lo stesso numero di serie.

Un certificato X.509 Versione 2 contiene anche un Identificativo emittente e un Identificativo oggetto e un certificato X.509 Versione 3 può contenere un numero di estensioni. Alcune estensioni certificato, come l'estensione Basic Constraint, sono *standard*, ma altre sono specifiche dell'implementazione. Un'estensione può essere *critica*, nel qual caso un sistema deve essere in grado di riconoscere il campo; se non riconosce il campo, deve rifiutare il certificato. Se un'estensione non è critica, il sistema può ignorarla se non la riconosce.

La firma digitale in un certificato personale viene generata utilizzando la chiave privata della CA che ha firmato tale certificato. Chiunque abbia bisogno di verificare il certificato personale può utilizzare la chiave pubblica della CA per farlo. Il certificato della CA contiene la chiave pubblica.

I certificati digitali non contengono la chiave privata. È necessario mantenere la chiave privata segreta.

Requisiti per i certificati personali
WebSphere MQ supporta certificati digitali conformi allo standard X.509 . Richiede l'opzione di autenticazione client.

Poiché IBM WebSphere MQ è un sistema peer to peer, viene visualizzato come autenticazione client nella terminologia SSL. Pertanto, qualsiasi certificato personale utilizzato per l'autenticazione SSL deve consentire un utilizzo chiave dell'autenticazione client. Non tutti i certificati server hanno questa opzione abilitata, quindi il provider di certificati potrebbe dover abilitare l'autenticazione client sulla CA root per il certificato protetto.

Oltre agli standard che specificano il formato dei dati per un certificato digitale, esistono anche standard per determinare se un certificato è valido. Questi standard sono stati aggiornati nel corso del tempo per prevenire alcuni tipi di violazioni della sicurezza. Ad esempio, i precedenti certificati X.509 versione 1 e 2 non indicavano se il certificato poteva essere legittimamente utilizzato per firmare altri certificati. È stato pertanto possibile per un utente malintenzionato ottenere un certificato personale da una fonte legittima e creare nuovi certificati progettati per impersonare altri utenti.

Quando si utilizzano i certificati X.509 versione 3, le estensioni certificato BasicConstraints e KeyUsage vengono utilizzate per specificare quali certificati possono firmare legittimamente altri certificati. Lo standard IETF RFC 5280 specifica una serie di regole di convalida del certificato che il software dell'applicazione conforme deve implementare per prevenire attacchi di impersonificazione. Una serie di regole del certificato è nota come politica di convalida del certificato.

Per ulteriori informazioni sulle politiche di convalida dei certificati in IBM WebSphere MQ, consultare “Politiche di convalida dei certificati in IBM WebSphere MQ” a pagina 33.

Autorità di certificazione (CA)
Una CA (Certificate Authority) è una terza parte attendibile che emette certificati digitali per fornire una garanzia che la chiave pubblica di un'entità appartiene realmente a tale entità.

I ruoli di una CA sono:

- Al ricevimento di una richiesta di certificato digitale, per verificare l'identità del richiedente prima di costruire, firmare e restituire il certificato personale
- Per fornire la chiave pubblica della CA nel relativo certificato CA
- Per pubblicare elenchi di certificati che non sono più attendibili in un CRL (Certificate Revocation List). Per ulteriori informazioni, consultare “Utilizzo dei certificati revocati” a pagina 150
- Per fornire l'accesso allo stato di revoca del certificato gestendo un server responder OCSP

Nomi distinti
Il DN (distinguished name) identifica in modo univoco un'entità in un certificato X.509 .

I seguenti tipi di attributo si trovano comunemente nel DN:

SERIALNUMBER	Numero di serie del certificato
MAIL	Indirizzo e-mail
E	Indirizzo e-mail (obsoleto, preferenza:n MAIL)
UID o USERID	Identificativo utente
CN	Nome comune (Common Name)
T	Titolo
OU	Nome unità organizzativa
CC	Componente dominio
O	Nome organizzazione
STREET	Via / Prima riga dell'indirizzo
L	Nome località
ST (o SP o S)	Nome stato o provincia
PC	Codice postale
C	Paese (Country)
UNSTRUCTUREDNAME	Nome host
UNSTRUCTUREDADDRESS	Indirizzo IP
DNQ	Identificativo DN (Distinguished Name)

Lo standard X.509 definisce altri attributi che in genere non fanno parte del DN ma possono fornire estensioni facoltative al certificato digitale.

Lo standard X.509 fornisce un DN da specificare in un formato stringa. Ad esempio:

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

Il CN (Common Name) può descrivere un singolo utente o qualsiasi altra entità, ad esempio un server Web.

Il DN può contenere più attributi OU e DC. È consentita una sola istanza di ognuno degli altri attributi. L'ordine delle voci OU è significativo: l'ordine specifica una gerarchia di nomi di unità organizzative, con l'unità di livello più alto per prima. Anche l'ordine delle voci DC è significativo.

IBM WebSphere MQ tollera alcuni DN non corretti. Per ulteriori informazioni, consultare Regole diWebSphere MQ per i valori SSLPEER.

Concetti correlati

“Contenuto di un certificato digitale” a pagina 10
I certificati digitali contengono informazioni specifiche, determinate dallo standard X.509 .

Ottenimento di certificati personali da un'autorità di certificazione
È possibile ottenere un certificato da una CA (Certificate Authority) esterna attendibile.

Si ottiene un certificato digitale inviando informazioni a una CA, sotto forma di una richiesta di certificato. Lo standard X.509 definisce un formato per queste informazioni, ma alcune CA hanno il proprio formato. Le richieste di certificati vengono generalmente generate dallo strumento di gestione dei certificati che il sistema utilizza, ad esempio lo strumento iKeyman sui sistemi UNIX, Linux®, e Windows e RACF su z/OS. Le informazioni contengono il DN (Distinguished Name) e la chiave pubblica. Quando il tuo strumento di gestione dei certificati genera la tua richiesta di certificato, genera anche la tua chiave privata, che devi mantenere sicura. Non distribuire mai la chiave privata.

Quando la CA riceve la tua richiesta, l'autorità verifica la tua identità prima di creare il certificato e restituirlo all'utente come certificato personale.

Figura 3 a pagina 12 illustra il processo per ottenere un certificato digitale da una CA.

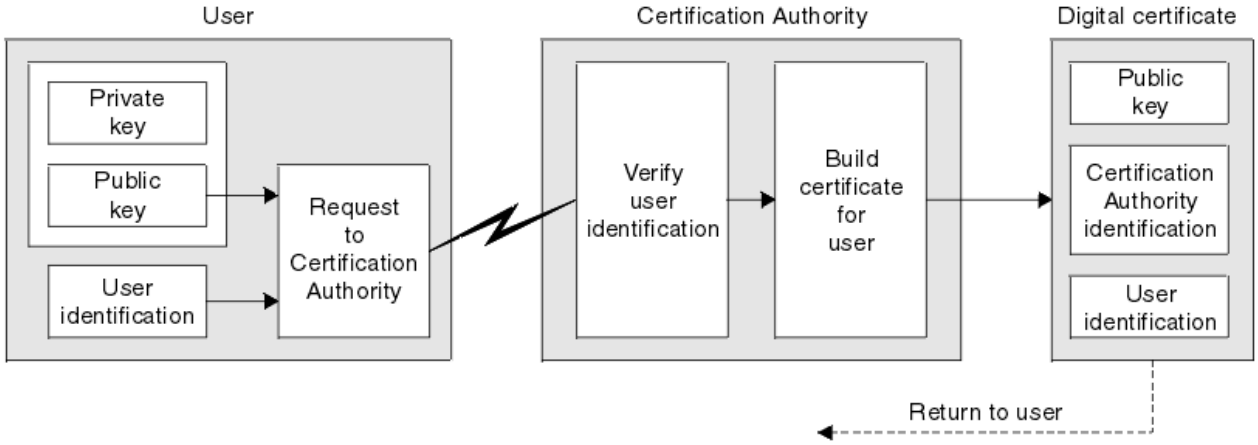


Figura 3. Ottenimento di un certificato digitale

Nel diagramma:

- L'"identificazione utente" include il tuo DN (Distinguished Name) del Soggetto.
- L'"identificazione della CA" include il DN (Distinguished Name) della CA che sta emettendo il certificato.
-

I certificati digitali contengono campi aggiuntivi diversi da quelli mostrati nel diagramma. Per ulteriori informazioni sugli altri campi in un certificato digitale, consultare “Contenuto di un certificato digitale” a pagina 10.

Funzionamento delle catene di certificati
Quando si riceve il certificato per un'altra entità, potrebbe essere necessario utilizzare una *catena di certificati* per ottenere il certificato *CA root* .

La catena di certificati, nota anche come *percorso di certificazione* , è un elenco di certificati utilizzati per autenticare un'entità. La catena, o percorso, inizia con il certificato di tale entità e ogni certificato nella catena è firmato dall'entità identificata dal certificato successivo nella catena. La catena termina con un certificato CA root. Il certificato CA root è sempre firmato dalla CA (certificate authority) stessa. Le forme di tutti i certificati nella catena devono essere verificate fino al raggiungimento del certificato root della CA.

Figura 4 a pagina 13 illustra un percorso di certificazione dal proprietario del certificato alla CA root, dove inizia la catena di attendibilità.

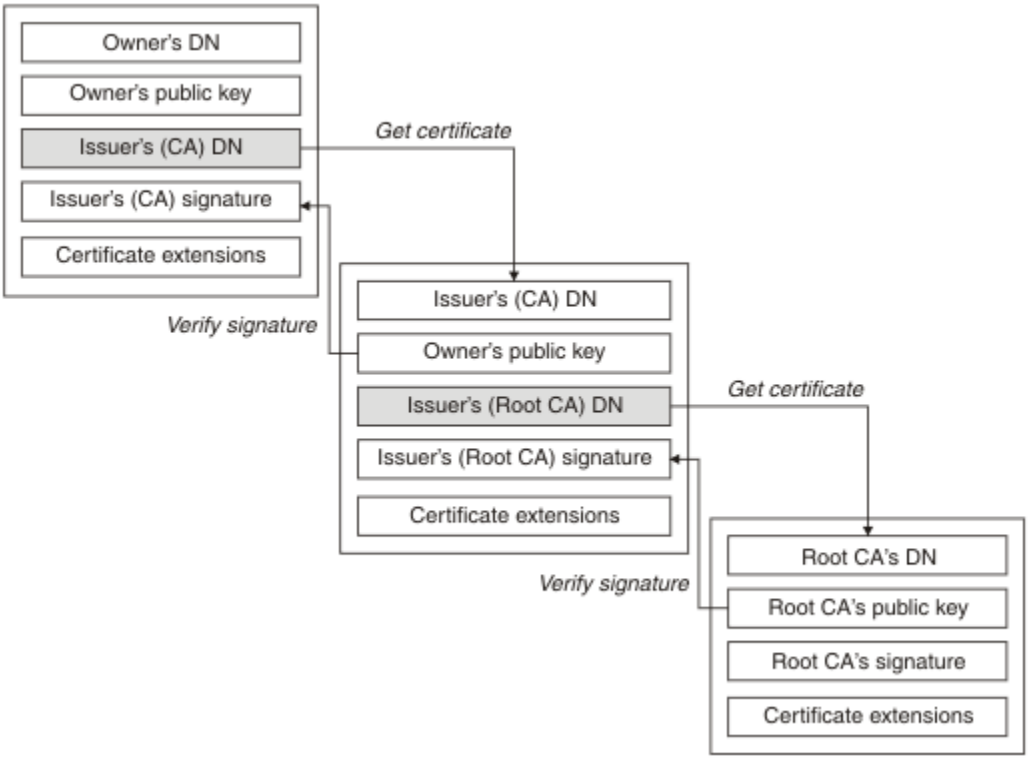


Figura 4. Catena di fiducia

Ogni certificato può contenere una o più estensioni. Un certificato appartenente a una CA generalmente contiene un'estensione BasicConstraints con l'indicatore isCA impostato per indicare che è consentito firmare altri certificati.

Quando i certificati non sono più validi
I certificati digitali possono scadere o essere revocati.

I certificati digitali sono emessi per un periodo fisso e non sono validi dopo la data di scadenza.

Consultare il Glossario per una definizione della scadenza del certificato.

I certificati possono essere revocati per vari motivi, tra cui:

- Il proprietario è stato spostato in un'organizzazione diversa.
- La chiave privata non è più segreta.

WebSphere MQ può verificare se un certificato è revocato inviando una richiesta a un responder OCSP (Online Certificate Status Protocol) (solo su sistemi UNIX, Linux e Windows). In alternativa, possono accedere a un CRL su un server LDAP. La revoca OCSP e le informazioni CRL sono pubblicate da una CA (Certificate Authority). Per ulteriori informazioni, vedere “Utilizzo dei certificati revocati” a pagina 150.

PKI (Public Key Infrastructure)

Una PKI (Public Key Infrastructure) è un sistema di strutture, politiche e servizi che supportano l'uso della crittografia a chiave pubblica per autenticare le parti coinvolte in una transazione.

Non esiste un singolo standard che definisce i componenti di una Public Key Infrastructure, ma un PKI in genere comprende le autorità di certificazione (CA) e le autorità di registrazione (RA). Le CA forniscono i seguenti servizi:

- Emissione di certificati digitali
- Convalida dei certificati digitali
- Revoca di certificati digitali
- Distribuzione di chiavi pubbliche

Gli standard X.509 forniscono la base per l'infrastruttura della chiave pubblica standard del settore.

Fare riferimento a “Certificati digitali” a pagina 9 per ulteriori informazioni sui certificati digitali e sulle CA (Certificate Authority). RAs verifica che le informazioni fornite quando i certificati digitali sono richiesti. Se la RA verifica tali informazioni, la CA può emettere un certificato digitale per il richiedente.

Un PKI può anche fornire strumenti per la gestione di certificati digitali e chiavi pubbliche. Una PKI è talvolta descritta come una *gerarchia di attendibilità* per la gestione dei certificati digitali, ma la maggior parte delle definizioni include servizi aggiuntivi. Alcune definizioni includono i servizi di crittografia e firma digitale, ma questi servizi non sono essenziali per il funzionamento di un PKI.

Protocolli di sicurezza crittografici: SSL e TLS

I protocolli crittografici forniscono connessioni sicure, consentendo a due parti di comunicare con privacy e integrità dei dati. Il protocollo TLS (Transport Layer Security) si è evoluto da quello SSL (Secure Sockets Layer). IBM WebSphere MQ supporta sia SSL che TLS.

L'obiettivo principale di entrambi i protocolli è quello di fornire la riservatezza, (a volte indicata come *privacy*), l'integrità dei dati, l'identificazione e l'autenticazione utilizzando i certificati digitali.

Sebbene i due protocolli siano simili, le differenze sono sufficientemente significative che SSL 3.0 e le varie versioni di TLS non interagiscono.

Concetti correlati

“Protocolli di sicurezza in IBM WebSphere MQ” a pagina 22
IBM WebSphere MQ supporta sia i protocolli TLS (Transport Layer Security) che SSL (Secure Sockets Layer) per fornire la sicurezza a livello di link per i canali di messaggi e MQI.

Concetti SSL (Secure Sockets Layer) e TLS (Transport Layer Security)

I protocolli SSL e TLS consentono a due parti di identificarsi e autenticarsi reciprocamente e comunicare con riservatezza e integrità dei dati. Il protocollo TLS si è evoluto dal protocollo Netscape SSL 3.0 ma TLS e SSL non interagiscono.

I protocolli SSL e TLS forniscono la sicurezza delle comunicazioni su Internet e consentono alle applicazioni client / server di comunicare in modo confidenziale e affidabile. I protocolli hanno due livelli: un protocollo di record e un protocollo di handshake, e questi sono sovrapposti a un protocollo di trasporto come TCP/IP. Entrambi usano tecniche di crittografia asimmetrica e simmetrica.

Una connessione SSL o TLS viene avviata da un'applicazione, che diventa il client SSL o TLS. L'applicazione che riceve la connessione diventa il server SSL o TLS. Ogni nuova sessione inizia con un handshake, come definito dai protocolli SSL o TLS.

Un elenco completo di CipherSpecs supportati da IBM WebSphere MQ è disponibile all'indirizzo “Specifica di CipherSpecs” a pagina 218

Per ulteriori informazioni sul protocollo SSL, consultare le informazioni fornite in https://www.mozilla.org/projects/security/pki/nss/ssl/draft302.txt. Per ulteriori informazioni sul protocollo TLS, consultare le informazioni fornite dal gruppo di lavoro TLS sul sito Web della task force Internet Engineering all'indirizzo https://www.ietf.org

Una panoramica dell'handshake SSL o TLS

L'handshake SSL o TLS abilita il client e il server SSL o TLS a stabilire le chiavi segrete con cui comunicano.

Questa sezione fornisce un riepilogo dei passi che consentono al client e al server SSL o TLS di comunicare tra loro.

- Concordare la versione del protocollo da utilizzare.
- Selezionare algoritmi crittografici.
- Autenticarsi reciprocamente scambiando e convalidando certificati digitali.
- Utilizzare tecniche di codifica asimmetrica per generare una chiave segreta condivisa, che eviti il problema di distribuzione della chiave. SSL o TLS utilizza quindi la chiave condivisa per la codifica simmetrica dei messaggi, che è più veloce della codifica asimmetrica.

Per ulteriori informazioni sugli algoritmi crittografici e sui certificati digitali, fare riferimento alle informazioni correlate.

Nella panoramica, i passaggi coinvolti nell'handshake SSL sono i seguenti:

1. Il client SSL o TLS invia un messaggio "client hello" che elenca le informazioni crittografiche come la versione SSL o TLS e, nell'ordine di preferenza del client, le CipherSuites supportate dal client. Il messaggio contiene anche una stringa di byte casuale utilizzata nei calcoli successivi. Il protocollo consente al "client hello" di includere metodi di compressione dati supportati dal client.
2. Il server SSL o TLS risponde con un messaggio "server hello" che contiene la CipherSuite scelta dal server dall'elenco fornito dal client, dall'ID sessione e da un'altra stringa di byte casuale. Il server invia anche il certificato digitale. Se il server richiede un certificato digitale per l'autenticazione client, il server invia una "richiesta di certificato client" che include un elenco dei tipi di certificati supportati e i DN (Distinguished Name) delle CA (Certification Authority) accettabili.
3. Il client SSL o TLS verifica il certificato digitale del server. Per ulteriori informazioni, vedere “In che modo SSL e TLS forniscono identificazione, autenticazione, riservatezza e integrità” a pagina 16.
4. Il client SSL o TLS invia la stringa di byte casuale che consente sia al client che al server di calcolare la chiave segreta da utilizzare per codificare i dati del messaggio successivi. La stringa di byte casuale stessa viene codificata con la chiave pubblica del server.
5. Se il server SSL o TLS ha inviato una "richiesta di certificato del client", il client invia una stringa di byte casuale codificata con la chiave privata del client, insieme con il certificato digitale del client o un "avviso di nessun certificato digitale". Questo avviso è solo un'avvertenza, ma con alcune implementazioni l'handshake ha esito negativo se l'autenticazione client è obbligatoria.
6. Il server SSL o TLS verifica il certificato del client. Per ulteriori informazioni, vedere “In che modo SSL e TLS forniscono identificazione, autenticazione, riservatezza e integrità” a pagina 16.
7. Il client SSL o TLS invia al server un messaggio "terminato" , codificato con la chiave segreta, che indica che la parte client dell'handshake è completa.
8. Il server SSL o TLS invia al client un messaggio "terminato" , codificato con la chiave segreta, che indica che la parte server dell'handshake è completa.
9. Per la durata della sessione SSL o TLS, il server e il client possono ora scambiare i messaggi codificati simmetricamente con la chiave segreta condivisa.

Figura 5 a pagina 16 illustra l'handshake SSL o TLS.

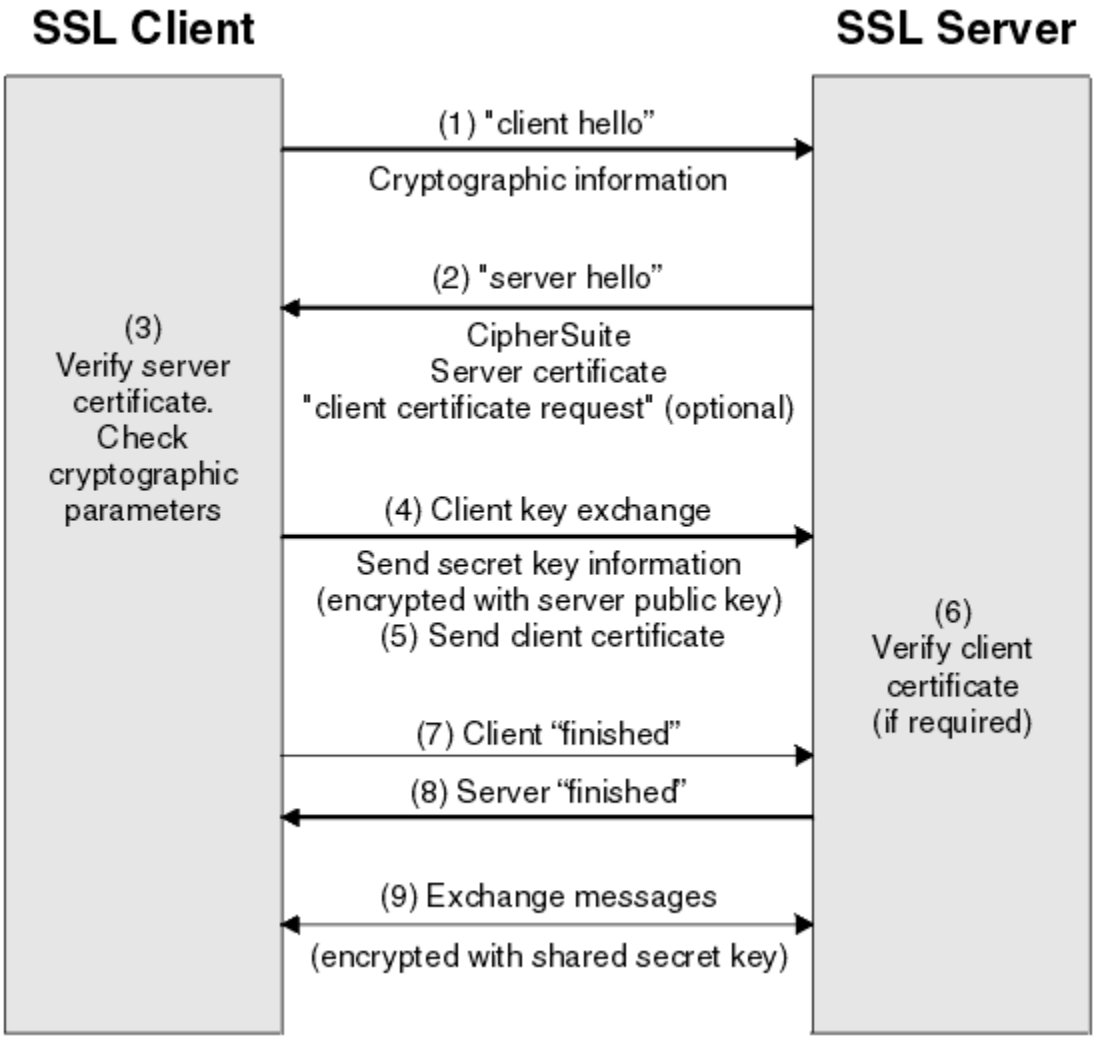


Figura 5. Panoramica dell'handshake SSL o TLS

In che modo SSL e TLS forniscono identificazione, autenticazione, riservatezza e integrità

Durante l'autenticazione del client e del server, è necessario che i dati siano codificati con una delle chiavi in una coppia di chiavi asimmetriche e decodificati con l'altra chiave della coppia. Un digest del messaggio viene utilizzato per fornire l'integrità.

Per una panoramica dei passi coinvolti nell'handshake TLS, consultare "Una panoramica dell'handshake SSL o TLS" a pagina 15.

Come SSL e TLS forniscono l'autenticazione

Per l'autenticazione del server, il client utilizza la chiave pubblica del server per codificare i dati utilizzati per calcolare la chiave segreta. Il server può generare la chiave segreta solo se può decodificare tali dati con la chiave privata corretta.

Per l'autenticazione client, il server utilizza la chiave pubblica nel certificato client per decodificare i dati che il client invia durante il passo "5" a pagina 15 dell'handshake. Lo scambio di messaggi terminati crittografati con la chiave segreta (passi "7" a pagina 15 e "8" a pagina 15 nella panoramica) conferma che l'autenticazione è completa.

Se una delle fasi di autenticazione ha esito negativo, l'handshake ha esito negativo e la sessione viene terminata.

Lo scambio di certificati digitali durante l'handshake SSLor TLS fa parte del processo di autenticazione. Per ulteriori informazioni sul modo in cui i certificati forniscono protezione contro l'impersonificazione, fare riferimento alle informazioni correlate. I certificati richiesti sono i seguenti, dove CA X emette il certificato per il client SSL o TLS e CA Y emette il certificato per il server SSL o TLS:

Solo per l'autenticazione del server, il server SSL o TLS ha bisogno di:

- Il certificato personale emesso al server dalla CA Y
- La chiave privata del server

e il client SSL o TLS ha bisogno di:

- Il certificato CA per CA Y

Se il server SSL o TLS richiede l'autenticazione client, il server verifica l'identità del client verificando il certificato digitale del client con la chiave pubblica per la CA che ha emesso il certificato personale al client, in questo caso la CA X . Per l'autenticazione server e client, il server ha bisogno di:

- Il certificato personale emesso al server dalla CA Y
- La chiave privata del server
- Il certificato CA per CA X

e il cliente ha bisogno di:

- Il certificato personale emesso al client dalla CA X
- La chiave privata del cliente
- Il certificato CA per CA Y

Sia il client che il server SSL o TLS potrebbero aver bisogno di altri certificati CA per formare una catena di certificati per il certificato CA root. Per ulteriori informazioni sulle catene di certificati, fare riferimento alle relative informazioni.

Cosa accade durante la verifica del certificato

Come indicato nei passi “3” a pagina 15 e “6” a pagina 15 della panoramica, il client SSL o TLS verifica il certificato del server e il server SSL o TLS verifica il certificato del client. Questa verifica presenta quattro aspetti:

1. La firma digitale viene controllata (consultare “Firme digitali in SSL e TLS” a pagina 19).
2. La catena di certificati è controllata; è necessario disporre di certificati CA intermedi (consultare “Funzionamento delle catene di certificati” a pagina 13).
3. Vengono verificate le date di scadenza e di attivazione e il periodo di validità.
4. Viene verificato lo stato di revoca del certificato (consultare “Utilizzo dei certificati revocati” a pagina 150).

Reimpostazione chiave segreta

Durante un handshake SSL o TLS viene generata una *chiave segreta* per codificare i dati tra il client SSL o TLS e il server. La chiave segreta viene utilizzata in una formula matematica applicata ai dati per trasformare il testo non codificato in testo non leggibile e il testo codificato in testo non codificato.

La chiave segreta viene generata dal testo casuale inviato come parte dell'handshake e viene utilizzata per codificare il testo non crittografato in testo crittografato. La chiave segreta viene utilizzata anche nell'algoritmo MAC (Message Authentication Code), che viene utilizzato per stabilire se un messaggio è stato modificato. Per ulteriori informazioni, fare riferimento a “Digest di messaggi e firme digitali” a pagina 9.

Se la chiave segreta viene rilevata, il testo semplice di un messaggio potrebbe essere decifrato dal testo cifrato o il digest del messaggio potrebbe essere calcolato, consentendo la modifica dei messaggi senza rilevamento. Anche per un algoritmo complesso, il testo in chiaro può alla fine essere scoperto applicando ogni possibile trasformazione matematica al testo cifrato. Per ridurre al minimo la quantità

di dati che possono essere decifrati o modificati se la chiave segreta viene interrotta, è possibile rinegoziare periodicamente la chiave segreta. Quando la chiave segreta è stata rinegoziata, la chiave segreta precedente non può più essere utilizzata per decodificare i dati codificati con la nuova chiave segreta.

Come SSL e TLS forniscono la riservatezza

SSL e TLS utilizzano una combinazione di codifica simmetrica e asimmetrica per garantire la privacy dei messaggi. Durante l'handshake SSL o TLS, il client e il server SSL o TLS concordano un algoritmo di codifica e una chiave segreta condivisa da utilizzare per una sola sessione. Tutti i messaggi trasmessi tra il client SSL o TLS e il server vengono crittografati utilizzando tale algoritmo e chiave, garantendo che il messaggio rimanga privato anche se viene intercettato. SSL supporta una vasta gamma di algoritmi crittografici. Poiché SSL e TLS utilizzano la crittografia asimmetrica durante il trasporto della chiave segreta condivisa, non vi è alcun problema di distribuzione della chiave. Per ulteriori informazioni sulle tecniche di codifica, fare riferimento a “Crittografia” a pagina 7.

Come SSL e TLS forniscono l'integrità

SSL e TLS forniscono l'integrità dei dati calcolando un digest del messaggio. Per ulteriori informazioni, fare riferimento a “Integrità dei dati dei messaggi” a pagina 227.

L'utilizzo di SSL o TLS garantisce l'integrità dei dati, purché CipherSpec nella tua definizione di canale utilizzi un algoritmo hash come descritto nella tabella in “Specifica di CipherSpecs” a pagina 218.

In particolare, se l'integrità dei dati è un problema, è necessario evitare di scegliere un CipherSpec il cui algoritmo hash è elencato come "Nessuno". Anche l'uso di MD5 è fortemente sconsigliato in quanto ora è molto vecchio e non più sicuro per la maggior parte degli scopi pratici.

CipherSpecs e CipherSuites

I protocolli di sicurezza crittografici devono concordare gli algoritmi utilizzati da una connessione sicura. CipherSpecs e CipherSuites definiscono combinazioni specifiche di algoritmi.

Un CipherSpec identifica una combinazione di algoritmo di codifica e algoritmo MAC (Message Authentication Code). Entrambe le estremità di una connessione TLS o SSL devono essere d'accordo sulla stessa CipherSpec per poter comunicare.

Importante: Quando si gestiscono i canali IBM WebSphere MQ , si utilizza una CipherSpec. Quando si utilizzano canali Java, JMS o MQTT, si specifica una CipherSuite.

Per ulteriori informazioni su CipherSpecs, consultare “Specifica di CipherSpecs” a pagina 218.

Una CipherSuite è una suite di algoritmi crittografici utilizzati da una connessione SSL o TLS. Una suite comprende tre algoritmi distinti:

- L'algoritmo di autenticazione e scambio di chiavi, utilizzato durante l'handshake
- L'algoritmo di codifica, utilizzato per codificare i dati
- L'algoritmo MAC (Message Authentication Code), utilizzato per generare il digest del messaggio

Esistono diverse opzioni per ciascun componente della suite, ma solo alcune combinazioni sono valide quando specificate per una connessione TLS o SSL. Il nome di una CipherSuite valida definisce la combinazione di algoritmi utilizzati. Ad esempio, CipherSuite SSL_RSA_WITH_RC4_128_MD5 specifica:

- Lo scambio di chiavi RSA e l'algoritmo di autenticazione
- L'algoritmo di codifica RC4 , utilizzando una chiave a 128 bit
- L'algoritmo MAC MD5

Diversi algoritmi sono disponibili per lo scambio di chiavi e l'autenticazione, ma l'algoritmo RSA è attualmente il più ampiamente utilizzato. Esiste una maggiore varietà negli algoritmi di crittografia e negli algoritmi MAC utilizzati.

Firme digitali in SSL e TLS

Una firma digitale è formata dalla codifica di una rappresentazione di un messaggio. La crittografia utilizza la chiave privata del firmatario e, per efficienza, di solito opera su un digest del messaggio piuttosto che sul messaggio stesso.

Le firme digitali variano con i dati che vengono firmati, a differenza delle firme scritte a mano, che non dipendono dal contenuto del documento che viene firmato. Se due messaggi diversi sono firmati digitalmente dalla stessa entità, le due firme differiscono, ma entrambe le firme possono essere verificate con la stessa chiave pubblica, ovvero la chiave pubblica dell'entità che ha firmato i messaggi.

Le fasi del processo di firma digitale sono le seguenti:

1. Il mittente elabora un digest del messaggio e quindi lo codifica utilizzando la chiave privata del mittente, formando la firma digitale.
2. Il mittente trasmette la firma digitale con il messaggio.
3. Il ricevente decodifica la firma digitale utilizzando la chiave pubblica del mittente, rigenerando il digest del messaggio del mittente.
4. Il destinatario calcola un digest del messaggio dai dati del messaggio ricevuti e verifica che i due digest siano uguali.

Figura 6 a pagina 19 illustra questo processo.

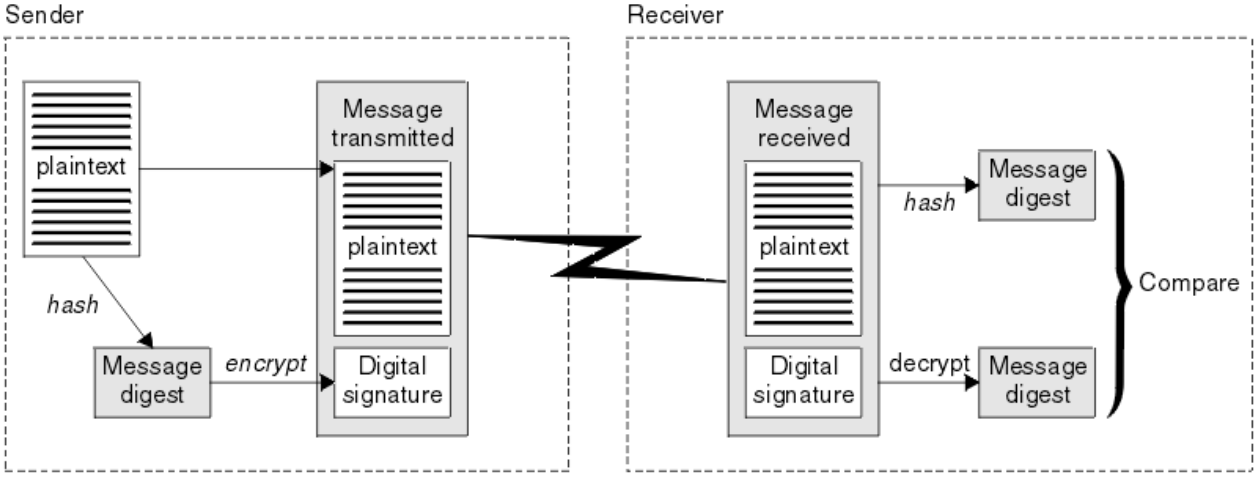


Figura 6. Il processo di firma digitale

Se la firma digitale è verificata, il destinatario sa che:

- Il messaggio non è stato modificato durante la trasmissione.
- Il messaggio è stato inviato dall'entità che dichiara di aver inviato.

Le firme digitali fanno parte dei servizi di integrità e autenticazione. Le firme digitali forniscono anche la prova dell'origine. Solo il mittente conosce la chiave privata, il che dimostra che il mittente è l'autore del messaggio.

Nota: È anche possibile codificare il messaggio stesso, che protegge la riservatezza delle informazioni nel messaggio.

FIS (Federal Information Processing Standards)

Il governo degli Stati Uniti fornisce consulenza tecnica sui sistemi IT e sulla sicurezza, inclusa la crittografia dei dati. Il National Institute for Standards and Technology (NIST) è un importante organismo che si occupa dei sistemi informatici e della sicurezza. NIST produce raccomandazioni e standard, inclusi FIPS (Federal Information Processing Standards).

Uno di questi standard è FIPS 140-2, che richiede l'utilizzo di forti algoritmi crittografici. FIPS 140-2 specifica anche i requisiti per gli algoritmi di hash da utilizzare per proteggere i pacchetti dalle modifiche in transito.

IBM WebSphere MQ fornisce il supporto FIPS 140-2 quando è stato configurato per farlo.

Nel tempo, gli analisti sviluppano attacchi contro gli algoritmi di crittografia e hashing esistenti. Nuovi algoritmi sono adottati per resistere a questi attacchi. FIPS 140-2 viene aggiornato periodicamente per tenere conto di tali cambiamenti.

Crittografia della National Security Agency (NSA) Suite B

Il governo degli Stati Uniti d'America fornisce consulenza tecnica sui sistemi IT e sulla sicurezza, compresa la crittografia dei dati. La US National Security Agency (NSA) raccomanda una serie di algoritmi di crittografia interoperabili nel suo standard Suite B.

Lo standard Suite B specifica una modalità operativa in cui viene utilizzato solo un insieme specifico di algoritmi crittografici sicuri. Lo standard Suite B specifica:

- L'algoritmo di crittografia (AES)
- L'algoritmo di scambio chiave (Elliptic Curve Diffie-Hellman, noto anche come ECDH)
- L'algoritmo di firma digitale (Elliptic Curve Digital Signature Algorithm, noto anche come ECDSA)
- Gli algoritmi di hash (SHA-256 o SHA-384)

Inoltre, lo standard IETF RFC 6460 specifica i profili conformi a Suite B che definiscono la configurazione e il comportamento dettagliati dell'applicazione necessari per conformarsi allo standard Suite B. Definisce due profili:

1. Un profilo conforme a Suite B da utilizzare con TLS versione 1.2. Quando configurato per l'operazione conforme a Suite B, verrà utilizzata solo la serie limitata di algoritmi di codifica sopra elencati.
2. Un profilo di transizione da utilizzare con TLS versione 1.0 o TLS versione 1.1. Questo profilo consente l'interoperabilità con server non conformi a Suite B. Quando è configurato per l'operazione di transizione Suite B, è possibile utilizzare ulteriori algoritmi di crittografia e hashing.

Lo standard Suite B è concettualmente simile a FIPS 140-2, perché limita la serie di algoritmi crittografici abilitati al fine di fornire un livello di sicurezza garantito.

Su sistemi Windows, UNIX e Linux , WebSphere MQ, può essere configurato per essere conforme al profilo TLS 1.2 conforme a Suite B, ma non supporta il profilo di transizione Suite B. Per ulteriori informazioni, fare riferimento a “NSA Suite B Crittografia in IBM WebSphere MQ” a pagina 30.

Informazioni correlate

“FIS (Federal Information Processing Standards)” a pagina 19
Il governo degli Stati Uniti fornisce consulenza tecnica sui sistemi IT e sulla sicurezza, inclusa la crittografia dei dati. Il National Institute for Standards and Technology (NIST) è un importante organismo che si occupa dei sistemi informatici e della sicurezza. NIST produce raccomandazioni e standard, inclusi FIPS (Federal Information Processing Standards).

Meccanismi di sicurezza IBM WebSphere MQ

Questa raccolta di argomenti spiega come implementare i diversi concetti di sicurezza in IBM WebSphere MQ.

IBM WebSphere MQ fornisce meccanismi per implementare tutti i concetti di sicurezza introdotti in “Concetti e meccanismi di sicurezza” a pagina 5. Questi sono discussi in modo più dettagliato nelle seguenti sezioni.

Identificazione e autenticazione in IBM WebSphere MQ

In IBM WebSphere MQ, è possibile implementare l'identificazione e l'autenticazione utilizzando le informazioni di contesto del messaggio e l'autenticazione reciproca.

Di seguito sono riportati alcuni esempi di identificazione e autenticazione in un ambiente IBM WebSphere MQ :

- Ogni messaggio può contenere informazioni sul *contesto del messaggio* . Queste informazioni vengono conservate nel descrittore del messaggio. Può essere generato dal gestore code quando un messaggio

viene inserito su una coda da un'applicazione. In alternativa, l'applicazione può fornire le informazioni se l'ID utente associato all'applicazione è autorizzato a farlo.

Le informazioni di contesto in un messaggio consentono all'applicazione ricevente di individuare il mittente del messaggio. Contiene, ad esempio, il nome dell'applicazione che inserisce il messaggio e l'ID utente associato all'applicazione.

- Quando un canale di messaggi viene avviato, è possibile che l'agente MCA (message channel agent) a ciascuna estremità del canale esegua l'autenticazione del relativo partner. Questa tecnica è nota come *autenticazione reciproca*. Per l'MCA di invio, fornisce la garanzia che il partner a cui sta per inviare i messaggi è autentico. Per l'MCA ricevente, esiste una garanzia simile che sta per ricevere messaggi da un partner autentico.

Concetti correlati

“Identificazione e autenticazione” a pagina 5

Identificazione è la capacità di identificare in maniera univoca un utente di un sistema o di un'applicazione in esecuzione nel sistema. L' *autenticazione* è la capacità di dimostrare che un utente o un'applicazione è realmente la persona o ciò che tale applicazione dichiara di essere.

Autorizzazione in IBM WebSphere MQ

È possibile utilizzare l'autorizzazione per limitare ciò che particolari individui o applicazioni possono fare nell'ambiente IBM WebSphere MQ .

Di seguito sono riportati alcuni esempi di autorizzazione in ambiente IBM WebSphere MQ :

- Consentire solo a un amministratore autorizzato di immettere comandi per gestire le risorse IBM WebSphere MQ .
- Consentire a un'applicazione di connettersi a un gestore code solo se l'ID utente associato all'applicazione è autorizzato a farlo.
- Consentire a un'applicazione di aprire solo le code necessarie per la sua funzione.
- Consentire a un'applicazione di sottoscrivere solo gli argomenti necessari per la sua funzione.
- Consentire a un'applicazione di eseguire solo le operazioni su una coda necessarie per la sua funzione. Ad esempio, un'applicazione potrebbe dover solo sfogliare i messaggi su una particolare coda e non inserire o richiamare messaggi.

Per ulteriori informazioni su come impostare l'autorizzazione, consultare “Autorizzazione di pianificazione” a pagina 49 e gli argomenti secondari associati.

Concetti correlati

“Authorization” a pagina 6

Autorizzazione protegge le risorse critiche in un sistema limitando l'accesso solo agli utenti autorizzati e alle rispettive applicazioni. Impedisce l'uso non autorizzato di una risorsa o l'uso non autorizzato di una risorsa.

Controllo in IBM WebSphere MQ

IBM WebSphere MQ può emettere messaggi di evento per registrare che si è svolta un'attività insolita.

Di seguito sono riportati alcuni esempi di controllo in un ambiente IBM WebSphere MQ :

- Un'applicazione tenta di aprire una coda che non è autorizzata ad aprire. Viene emesso un messaggio di evento di strumentazione. Esaminando il messaggio di evento, si scopre che questo tentativo si è verificato e si può decidere quale azione è necessaria.
- Un'applicazione tenta di aprire un canale, ma il tentativo non riesce perché SSL non consente la connessione. Viene emesso un messaggio di evento di strumentazione. Esaminando il messaggio di evento, si scopre che questo tentativo si è verificato e si può decidere quale azione è necessaria.

Concetti correlati

“Audit” a pagina 6

Il *Controllo* è il processo di registrazione e controllo degli eventi per rilevare se si è verificata un'attività imprevista o non autorizzata o se è stato effettuato un tentativo di eseguire tale attività.

Riservatezza in IBM WebSphere MQ

È possibile implementare la riservatezza in IBM WebSphere MQ codificando i messaggi.

Di seguito sono riportati alcuni esempi di come è possibile garantire la riservatezza in un ambiente IBM WebSphere MQ :

- Dopo che un MCA mittente riceve un messaggio da una coda di trasmissione, IBM WebSphere MQ utilizza SSL o TLS per codificare il messaggio prima che venga inviato sulla rete all'MCA ricevente. All'altra estremità del canale, il messaggio viene decodificato prima che l'MCA ricevente lo inmetta nella sua coda di destinazione.
- Mentre i messaggi vengono memorizzati in una coda locale, i meccanismi di controllo accessi forniti da IBM WebSphere MQ potrebbero essere considerati sufficienti per proteggere il loro contenuto da una divulgazione non autorizzata. Tuttavia, per un livello di sicurezza maggiore, è possibile utilizzare IBM WebSphere MQ Advanced Message Security per codificare i messaggi memorizzati nelle code.

Concetti correlati

“Riservatezza” a pagina 7
Il servizio *riservatezza* protegge le informazioni sensibili dalla divulgazione non autorizzata.

Integrità dei dati in IBM WebSphere MQ

È possibile utilizzare un servizio di integrità dati per rilevare se un messaggio è stato modificato.

Di seguito sono riportati alcuni esempi di come è possibile garantire l'integrità dei dati in un ambiente IBM WebSphere MQ :

- È possibile utilizzare SSL o TLS per rilevare se il contenuto di un messaggio è stato deliberatamente modificato mentre veniva trasmesso su una rete. In SSL e TLS, l'algoritmo digest del messaggio fornisce il rilevamento dei messaggi modificati in transito. Tutti i IBM WebSphere MQ CipherSpecs forniscono un algoritmo digest del messaggio, tranne TLS_RSA_WITH_NULL_NULL che non fornisce l'integrità dei dati del messaggio.
- Mentre i messaggi sono memorizzati in una coda locale, i meccanismi di controllo degli accessi forniti da IBM WebSphere MQ potrebbero essere considerati sufficienti per impedire la modifica deliberata del contenuto dei messaggi. Tuttavia, per un livello di sicurezza maggiore, è possibile utilizzare IBM WebSphere MQ Advanced Message Security per rilevare se il contenuto di un messaggio è stato deliberatamente modificato tra l'ora in cui il messaggio è stato inserito nella coda e l'ora in cui è stato richiamato dalla coda.

Concetti correlati

“Integrità dati” a pagina 7
Il servizio *integrità dati* rileva se è stata effettuata una modifica non autorizzata dei dati.

Crittografia in IBM WebSphere MQ

IBM WebSphere MQ fornisce la crittografia utilizzando i protocolli SSL (Secure sockets layer) e TLS (Transport Security Layer).

Per ulteriori informazioni, fare riferimento a “Protocolli di sicurezza in IBM WebSphere MQ” a pagina 22.

Concetti correlati

“Concetti crittografici” a pagina 7
Questa raccolta di argomenti descrive i concetti di crittografia applicabili a WebSphere MQ.

Protocolli di sicurezza in IBM WebSphere MQ

IBM WebSphere MQ supporta sia i protocolli TLS (Transport Layer Security) che SSL (Secure Sockets Layer) per fornire la sicurezza a livello di link per i canali di messaggi e MQI.

I canali dei messaggi e i canali MQI possono utilizzare il protocollo SSL o TLS per fornire la sicurezza a livello di link.Un MCA del chiamante è un client SSL o TLS e un MCA del risponditore è un server SSL o TLS. WebSphere MQ supporta la Versione 3.0 del protocollo SSL e la Versione 1.0 e la Versione 1.2 del protocollo TLS (Transport Layer Security). Specificare gli algoritmi crittografici utilizzati da SSL o dal protocollo fornendo una CipherSpec come parte della definizione del canale.

Ad ogni estremità di un canale di messaggi e all'estremità server di un canale MQI, l'MCA agisce per conto del gestore code a cui è connesso. Durante l'handshake SSL o TLS, l'MCA invia il certificato digitale del gestore code al proprio MCA partner all'altra estremità del canale. Il codice WebSphere MQ all'estremità client di un canale MQI agisce per conto dell'utente dell'applicazione client WebSphere MQ . Durante l'handshake SSL o TLS, il codice WebSphere MQ invia il certificato digitale dell'utente all'MCA all'estremità server del canale MQI.

I gestori code e gli utenti del client WebSphere MQ non devono avere certificati digitali personali associati quando agiscono come client SSL o TLS, a meno che SSLCAUTH (REQUIRED) non sia specificato sul lato server del canale.

I certificati digitali sono memorizzati in un *repository chiavi* . L'attributo del gestore code *SSLKeyRepository* specifica l'ubicazione del repository delle chiavi che contiene il certificato digitale del gestore code . On a WebSphere MQ client system, the MQSSLKEYR environment variable specifies the location of the key repository that holds the user's digital certificate. In alternativa, un'applicazione client WebSphere MQ può specificare la sua posizione nel campo *KeyRepository* della struttura delle opzioni di configurazione SSL e TLS, MQSCO, su una chiamata MQCONNX. Consultare gli argomenti correlati per ulteriori informazioni sui repository chiave e su come specificare dove si trovano.

Concetti correlati

“Protocolli di sicurezza crittografici: SSL e TLS” a pagina 14
I protocolli crittografici forniscono connessioni sicure, consentendo a due parti di comunicare con privacy e integrità dei dati. Il protocollo TLS (Transport Layer Security) si è evoluto da quello SSL (Secure Sockets Layer). IBM WebSphere MQ supporta sia SSL che TLS.

Supporto IBM WebSphere MQ per SSL e TLS

IBM WebSphere MQ supporta sia il protocollo SSL (Secure Sockets Layer) che il protocollo TLS (Transport Layer Security).

Per ulteriori informazioni sui protocolli SSL e TLS, fare riferimento alle informazioni correlate.

IBM WebSphere MQ fornisce il seguente supporto per SSL Versione 3.0 e TLS 1.0 e TLS 1.2:

Client Java e JMS
Questi client utilizzano la JVM per fornire il supporto SSL e TLS.

Sistemi UNIX, Linux, and Windowse HP Integrity NonStop Server
Per sistemi UNIX, Linux, and Windowse HP Integrity NonStop Server , il supporto SSL e TLS è installato con IBM WebSphere MQ.

Per informazioni su eventuali prerequisiti per il supporto IBM WebSphere MQ SSL e TLS, consultare Requisiti di sistema per IBM WebSphere MQ.

Il repository delle chiavi SSL o TLS
Una connessione SSL o TLS reciprocamente autenticata richiede un repository di chiavi (che può essere riconosciuto da nomi differenti su piattaforme differenti) ad ogni estremità della connessione. Il repository di chiavi include certificati digitali e chiavi private.

Queste informazioni utilizzano il termine generico *repository chiavi* per descrivere l'archivio per certificati digitali e le relative chiavi private associate. I nomi di archivio specifici utilizzati sulle piattaforme e gli ambienti che supportano SSL e TLS sono:

Java e JMS	keystore e truststore

Sistemi Windows , UNIX and Linux

file database delle chiavi

Per ulteriori informazioni, fare riferimento a “Certificati digitali” a pagina 9 e “Concetti SSL (Secure Sockets Layer) e TLS (Transport Layer Security)” a pagina 14.

Una connessione SSL o TLS autenticata reciprocamente richiede un repository di chiavi ad ogni estremità della connessione. Il repository delle chiavi può contenere:

- Un numero di certificati CA da varie autorità di certificazione che consentono al gestore code o al client di verificare i certificati che riceve dal partner all'estremità remota della connessione. I singoli certificati potrebbero trovarsi in una catena di certificati.
- Uno o più certificati personali ricevuti da un'autorità di certificazione. Si associa un certificato personale separato a ciascun gestore code o client WebSphere MQ MQI. I certificati personali sono essenziali su un client SSL o TLS se è richiesta l'autenticazione reciproca. Se non è richiesta l'autenticazione reciproca, i certificati personali non sono necessari sul client. Il repository delle chiavi potrebbe contenere anche la chiave privata corrispondente a ciascun certificato personale.
- Richieste di certificati che sono in attesa di essere firmate da un certificato CA attendibile.

Per ulteriori informazioni sulla protezione del repository delle chiavi, consultare “Protezione dei repository delle chiavi IBM WebSphere MQ” a pagina 24.

L'ubicazione del repository delle chiavi dipende dalla piattaforma che si sta utilizzando:

Windows, sistemi UNIX and Linux

Su sistemi Windows, UNIX and Linux il repository delle chiavi è un file di database delle chiavi. Il nome del file database di chiavi deve avere un'estensione file `.kdb`. Ad esempio, su UNIX and Linux, il file database delle chiavi predefinito per il gestore code `QM1` è `/var/mqm/qmgrs/QM1/ssl/key.kdb` . Se IBM WebSphere MQ è installato nell'ubicazione predefinita, il percorso equivalente in Windows è `C:\Program Files\IBM\WebSphere MQ\Qmgrs\QM1\ssl\key.kdb`.

Su sistemi Windows , UNIX and Linux , ogni file di database delle chiavi ha un file stash delle password associato. Questo file contiene password codificate che consentono ai programmi di accedere al database delle chiavi. Il file stash delle password deve trovarsi nella stessa directory e avere lo stesso file system del database delle chiavi e deve terminare con il suffisso `.sth` , ad esempio `/var/mqm/qmgrs/QM1/ssl/key.sth`

Nota: Sui sistemi Windows, UNIX and Linux , le schede hardware di crittografia PKCS #11 possono contenere i certificati e le chiavi che sono contenuti in un file di database delle chiavi. Quando i certificati e le chiavi vengono conservati su schede PKCS #11 , WebSphere MQ richiede ancora l'accesso sia a un file di database di chiavi che a un file stash di password.

Su sistemi Windows e UNIX , il database delle chiavi contiene anche la chiave privata per il certificato personale associato al gestore code o al client WebSphere MQ MQI.

Protezione dei repository delle chiavi IBM WebSphere MQ

Il repository delle chiavi per IBM WebSphere MQ è un file. Assicurarsi che solo l'utente desiderato possa accedere al file del repository delle chiavi. Ciò impedisce ad un intruso o ad un altro utente non autorizzato di copiare il file del repository delle chiavi su un altro sistema e quindi di impostare un ID utente identico su tale sistema per impersonare l'utente previsto.

Le autorizzazioni sui file dipendono dall'umask dell'utente e da quale strumento viene utilizzato. Su Windows, IBM WebSphere MQ account richiedono autorizzazione `BypassTraverseChecking` , il che significa che le autorizzazioni delle cartelle nel percorso file non hanno alcun effetto.

Controllare le autorizzazioni file dei file del repository delle chiavi e assicurarsi che i file e la cartella di contenimento non siano leggibili, preferibilmente non leggibili dal gruppo.

Rendere il keystore di sola lettura è una buona pratica, su qualsiasi sistema si utilizzi, con solo l'amministratore autorizzato ad abilitare le operazioni di scrittura per eseguire la manutenzione.

In pratica, è necessario proteggere tutti i keystore, indipendentemente dall'ubicazione e se sono protetti da password o meno; proteggere i repository delle chiavi.

Aggiornamento del repository delle chiavi del gestore code
Quando si modifica il contenuto di un repository delle chiavi, il gestore code non seleziona immediatamente il nuovo contenuto. Affinché un gestore code utilizzi il nuovo contenuto del repository delle chiavi, è necessario immettere il comando REFRESH SECURITY TYPE (SSL).

Questo processo è intenzionale e impedisce la situazione in cui più canali in esecuzione potrebbero utilizzare versioni differenti di un repository di chiavi. Come controllo di sicurezza, solo una versione di un repository delle chiavi può essere caricata dal gestore code in qualsiasi momento.

Per ulteriori informazioni sul comando REFRESH SECURITY TYPE (SSL), consultare REFRESH SECURITY.

È anche possibile aggiornare un repository delle chiavi utilizzando i comandi PCF o WebSphere MQ Explorer. Per ulteriori informazioni, consultare il Comando MQCMD_REFRESH_SECURITY e l'argomento *Aggiornamento della sicurezza SSL o TLS* nella sezione WebSphere MQ Explorer di questa documentazione del prodotto.

Concetti correlati
“Aggiornamento di una vista del client del contenuto del repository chiavi SSL e delle impostazioni SSL” a pagina 25
Per aggiornare l'applicazione client con il contenuto aggiornato del repository chiavi, è necessario arrestare e riavviare l'applicazione client.

Aggiornamento di una vista del client del contenuto del repository chiavi SSL e delle impostazioni SSL
Per aggiornare l'applicazione client con il contenuto aggiornato del repository chiavi, è necessario arrestare e riavviare l'applicazione client.

Non è possibile aggiornare la sicurezza su un client WebSphere MQ ; non esiste un equivalente del comando REFRESH SECURITY TYPE (SSL) per i client (consultare REFRESH SECURITY) per ulteriori informazioni.

È necessario arrestare e riavviare l'applicazione, ogni volta che si modifica il certificato di protezione, per aggiornare l'applicazione client con il contenuto aggiornato del repository chiavi.

Se il riavvio del canale aggiorna le configurazioni e se l'applicazione dispone di una logica di riconnessione, è possibile aggiornare la sicurezza sul client immettendo il comando STOP CHL STATUS (INACTIVE).

Concetti correlati
“Aggiornamento del repository delle chiavi del gestore code” a pagina 25
Quando si modifica il contenuto di un repository delle chiavi, il gestore code non seleziona immediatamente il nuovo contenuto. Affinché un gestore code utilizzi il nuovo contenuto del repository delle chiavi, è necessario immettere il comando REFRESH SECURITY TYPE (SSL).

FIPS (Federal Information Processing Standards)
Questo argomento introduce il FIPS (Federal Information Processing Standards) Cryptomodule Validation Program dell'US National Institute of Standards and Technology e le funzioni crittografiche che possono essere utilizzate sui canali SSL o TLS, per i sistemi Windows, UNIX and Linuxe z/OS .

La conformità a FIPS 140-2 di una connessione IBM WebSphere MQ SSL o TLS su sistemi UNIX, Linuxe Windows si trova qui “FIPS (Federal Information Processing Standards) per UNIX, Linuxe Windows” a pagina 26.

Se l'hardware di crittografia è presente, i moduli di codifica utilizzati da IBM WebSphere MQ possono essere configurati in modo da essere quelli forniti dal produttore dell'hardware. In questo caso, la configurazione è conforme a FIPS solo se tali moduli crittografici sono certificati FIPS.

Nel tempo, i Federal Information Processing Standards vengono aggiornati per riflettere nuovi attacchi contro protocolli e algoritmi di crittografia. Ad esempio, alcuni CipherSpecs potrebbero non essere più

certificati FIPS. Quando si verificano tali modifiche, anche IBM WebSphere MQ viene aggiornato per implementare lo standard più recente. Di conseguenza, si potrebbero notare dei cambiamenti nelle modalità di funzionamento dopo l'applicazione della manutenzione.

Concetti correlati

“Specifica che solo i CipherSpecs certificati FIPS vengono utilizzati al runtime sul client MQI” a pagina 108
Creare i repository delle chiavi utilizzando il software conforme a FIPS, quindi specificare che il canale deve utilizzare CipherSpecscertificati FIPS.

“Utilizzo di iKeyman, iKeycmd, runmqakm e runmqckm” a pagina 114
Sui sistemi UNIX, Linux e Windows , gestire le chiavi e i certificati digitali con la GUI iKeyman o dalla riga comandi utilizzando iKeycmd o runmqakm.

Attività correlate

Abilitazione di SSL nelle classi WebSphere MQ per Java

Utilizzo di SSL (Secure Sockets Layer) con classi WebSphere MQ per JMS

Riferimenti correlati

Proprietà SSL degli oggetti JMS

Informazioni correlate

“FIS (Federal Information Processing Standards)” a pagina 19
Il governo degli Stati Uniti fornisce consulenza tecnica sui sistemi IT e sulla sicurezza, inclusa la crittografia dei dati. Il National Institute for Standards and Technology (NIST) è un importante organismo che si occupa dei sistemi informatici e della sicurezza. NIST produce raccomandazioni e standard, inclusi FIPS (Federal Information Processing Standards).

FIPS (Federal Information Processing Standards) per UNIX, Linuxe Windows
Quando la codifica è richiesta su un canale SSL o TLS su sistemi Windows, UNIX and Linux , WebSphere MQ utilizza un package di crittografia denominato IBM Crypto for C (ICC). Sulle piattaforme Windows, UNIX and Linux , il software ICC ha passato il programma di convalida crittografico FIPS (Federal Information Processing Standards) dello US National Institute of Standards and Technology, al livello 140-2.

La conformità FIPS 140-2 di una connessione WebSphere MQ SSL o TLS su sistemi Windows, UNIX and Linux è la seguente:

- Per tutti i canali di messaggi IBM WebSphere MQ (ad eccezione dei tipi di canale CLNTCONN), la connessione è conforme a FIPS se sono soddisfatte le seguenti condizioni:
  - La versione ICC GSKit installata è stata certificata conforme a FIPS 140-2 sulla versione del sistema operativo installato e sull'architettura hardware.
  - L'attributo SSLFIPS del gestore code è stato impostato su YES.
  - Tutti i repository di chiavi sono stati creati e manipolati utilizzando solo software compatibile con FIPS, come **runmqakm** con l'opzione `-fips` .
- Per tutte le applicazioni client IBM WebSphere MQ MQI, la connessione utilizza GSKit ed è conforme a FIPS se vengono soddisfatte le seguenti condizioni:
  - La versione ICC GSKit installata è stata certificata conforme a FIPS 140-2 sulla versione del sistema operativo installato e sull'architettura hardware.
  - È stato specificato di utilizzare solo la crittografia certificata FIPS, come descritto nell'argomento correlato per il client MQI.
  - Tutti i repository di chiavi sono stati creati e manipolati utilizzando solo software compatibile con FIPS, come **runmqakm** con l'opzione `-fips` .
- Per le classi IBM WebSphere MQ per le applicazioni Java che utilizzano la modalità client, la connessione utilizza le implementazioni SSL e TLS di JRE ed è conforme a FIPS se vengono soddisfatte le seguenti condizioni:
  - Java Runtime Environment utilizzato per eseguire l'applicazione è conforme a FIPS sulla versione del sistema operativo installato e sull'architettura hardware.

  - È stato specificato che deve essere utilizzata solo la crittografia certificata FIPS, come descritto nell'argomento correlato per il client Java.
  - Tutti i repository di chiavi sono stati creati e manipolati utilizzando solo software compatibile con FIPS, come **runmqakm** con l'opzione `-fips` .
- Per le classi IBM WebSphere MQ per le applicazioni JMS che utilizzano la modalità client, la connessione utilizza le implementazioni SSL e TLS di JRE ed è conforme a FIPS se vengono soddisfatte le seguenti condizioni:
  - Java Runtime Environment utilizzato per eseguire l'applicazione è conforme a FIPS sulla versione del sistema operativo installato e sull'architettura hardware.
  - È stato specificato di utilizzare solo la crittografia certificata FIPS, come descritto nell'argomento correlato per il client JMS.
  - Tutti i repository di chiavi sono stati creati e manipolati utilizzando solo software compatibile con FIPS, come **runmqakm** con l'opzione `-fips` .
- Per applicazioni client .NET non gestite, la connessione utilizza GSKit ed è conforme a FIPS se vengono soddisfatte le condizioni riportate di seguito:
  - La versione ICC GSKit installata è stata certificata conforme a FIPS 140-2 sulla versione del sistema operativo installato e sull'architettura hardware.
  - È stata specificata l'utilizzo di solo crittografia certificata FIPS, come descritto nell'argomento correlato per il client .NET.
  - Tutti i repository di chiavi sono stati creati e manipolati utilizzando solo software compatibile con FIPS, come **runmqakm** con l'opzione `-fips` .
- Per applicazioni client XMS .NET non gestite, la connessione utilizza GSKit ed è conforme a FIPS se vengono soddisfatte le condizioni riportate di seguito:
  - La versione ICC GSKit installata è stata certificata conforme a FIPS 140-2 sulla versione del sistema operativo installato e sull'architettura hardware.
  - È stato specificato che deve essere utilizzata solo la crittografia certificata FIPS, come descritto nella documentazione XMS .NET.
  - Tutti i repository di chiavi sono stati creati e manipolati utilizzando solo software compatibile con FIPS, come **runmqakm** con l'opzione `-fips` .

Tutte le piattaforme AIX, Linux, HP-UX, Solaris, Windowse z/OS supportate sono certificate FIPS 140-2, ad eccezione di quanto indicato nel file readme incluso con ogni fix pack o pacchetto di aggiornamento.

Per connessioni SSL e TLS che utilizzano GSKit, il componente certificato FIPS 140-2 è denominato *ICC*. È la versione di questo componente che determina la conformità GSKit FIPS su una determinata piattaforma. Per determinare la versione ICC attualmente installata, eseguire il comando **dspmqver -p 64 -v** .

Di seguito è riportato un estratto di esempio dell'output **dspmqver -p 64 -v** relativo a ICC:

```
  icc
 ============
@ (#)CompanyName: IBM Corporation
@(#)LegalTrademarks:  IBM
@ (#)FileDescription: IBM Crypto for C - language
@(#)FileVersion:      8.0.0.0
@ (#)LegalCopyright: Materiali su licenza - Proprietà di IBM
ICC @ (#)
@ (#) (C) Copyright IBM Corp. 2002, 2024.
@ (#) Tutti i diritti riservati. Utenti del Governo degli Stati Uniti
@ (#) Diritti limitati - Utilizzo, duplicazione o divulgazione
@ (#) è limitato dal GSA ADP Schedule Contract con 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 dichiarazione di certificazione NIST per GSKit ICC 8 (incluso in GSKit 8) è disponibile al seguente indirizzo: Cryptographic Module Validation Program.

Se l'hardware di crittografia è presente, i moduli di codifica utilizzati da IBM WebSphere MQ possono essere configurati in modo da essere quelli forniti dal produttore dell'hardware. In questo caso, la configurazione è conforme a FIPS solo se tali moduli crittografici sono certificati FIPS.

Nota: I client SSL e TLS Solaris x86 a 32 bit configurati per il funzionamento conforme a FIPS 140-2 non vengono eseguiti correttamente sui sistemi Intel. Questo errore si verifica perché il file di libreria a 32 bit GSKit-Crypto Solaris x86 conforme a FIPS 140-2 non viene caricato sul chipset Intel. Sui sistemi interessati, viene riportato l'errore AMQ9655 nel log degli errori del client. Per risolvere questo problema, disabilitare la conformità a FIPS 140-2 o ricompilare l'applicazione client a 64 bit, poiché il codice a 64 bit non viene interessato.

Triplice restrizioni DES applicate quando si opera in conformità con FIPS 140-2

Quando WebSphere MQ è configurato per funzionare in conformità con FIPS 140-2, vengono applicate ulteriori limitazioni in relazione a Triple DES (3DES) CipherSpecs. Queste limitazioni consentono la conformità con il suggerimento US NIST SP800-67 .

1. Tutte le parti della chiave Triple DES devono essere univoche.
2. Nessuna parte della chiave Triple DES può essere una chiave Weak, Semi - Weak o Possibilmente - Weak secondo le definizioni in NIST SP800-67.
3. Non è possibile trasmettere più di 32 GB di dati sulla connessione prima che si verifichi una reimpostazione della chiave segreta. Per impostazione predefinita, WebSphere MQ non reimposta la chiave della sessione segreta, pertanto questa reimpostazione deve essere configurata. L'errore nell'abilitare la reimpostazione della chiave segreta quando si utilizza una conformità Triple DES CipherSpec e FIPS 140-2 determina la chiusura della connessione con errore AMQ9288 dopo il superamento del numero massimo di byte. Per informazioni su come configurare la reimpostazione della chiave segreta, consultare “Reimpostazione delle chiavi segrete SSL e TLS” a pagina 224.

WebSphere MQ genera chiavi di sessione Triple DES già conformi alle regole 1 e 2. Tuttavia, per soddisfare la terza limitazione, è necessario abilitare la reimpostazione della chiave segreta quando si utilizza Triple DES CipherSpecs in una configurazione FIPS 140-2. In alternativa, è possibile evitare di utilizzare Triple DES.

Concetti correlati

“Specifica che solo i CipherSpecs certificati FIPS vengono utilizzati al runtime sul client MQI” a pagina 108
Creare i repository delle chiavi utilizzando il software conforme a FIPS, quindi specificare che il canale deve utilizzare CipherSpecscertificati FIPS.

“Utilizzo di iKeyman, iKeycmd, runmqakm e runmqckm” a pagina 114
Sui sistemi UNIX, Linux e Windows , gestire le chiavi e i certificati digitali con la GUI iKeyman o dalla riga comandi utilizzando iKeycmd o runmqakm.

Attività correlate

Abilitazione di SSL nelle classi WebSphere MQ per Java

Utilizzo di SSL (Secure Sockets Layer) con classi WebSphere MQ per JMS

Riferimenti correlati

Proprietà SSL degli oggetti JMS

Informazioni correlate

“FIS (Federal Information Processing Standards)” a pagina 19
Il governo degli Stati Uniti fornisce consulenza tecnica sui sistemi IT e sulla sicurezza, inclusa la crittografia dei dati. Il National Institute for Standards and Technology (NIST) è un importante organismo che si occupa dei sistemi informatici e della sicurezza. NIST produce raccomandazioni e standard, inclusi FIPS (Federal Information Processing Standards).

SSL e TLS sul client IBM WebSphere MQ MQI
IBM WebSphere MQ supporta SSL e TLS sui client. È possibile personalizzare l'utilizzo di SSL o TLS in vari modi.

IBM WebSphere MQ fornisce il supporto SSL e TLS per client IBM WebSphere MQ MQI su sistemi Windows, UNIX and Linux . Se si utilizzano le classi IBM WebSphere MQ per Java, consultare Utilizzo delle classi WebSphere MQ per Java e se si utilizzano le classi IBM WebSphere MQ per JMS, consultare Utilizzo delle classi WebSphere MQ per JMS. Il resto di questa sezione non si applica agli ambienti Java o JMS.

È possibile specificare il repository delle chiavi per un client IBM WebSphere MQ MQI con il valore MQSSLKEYR nel file di configurazione del client IBM WebSphere MQ o quando l'applicazione effettua una chiamata MQCONNX. Sono disponibili tre opzioni per specificare che un canale utilizza SSL:

- Utilizzo di una tabella di definizione di canale
- Utilizzo della struttura delle opzioni di configurazione SSL, MQSCO, su una chiamata MQCONNX
- Utilizzo di Active Directory (su sistemi Windows)

Non è possibile utilizzare la variabile di ambiente MQSERVER per specificare che un canale utilizza SSL.

È possibile continuare ad eseguire le proprie applicazioni client IBM WebSphere MQ MQI esistenti senza SSL, purché SSL non sia specificato sull'altra estremità del canale.

Se su una macchina client vengono apportate modifiche al contenuto del repository delle chiavi SSL, all'ubicazione del repository delle chiavi SSL, alle informazioni di autenticazione o ai parametri hardware crittografici, è necessario terminare tutte le connessioni SSL per riflettere tali modifiche nei canali di connessione client che l'applicazione sta utilizzando per connettersi al gestore code. Una volta terminate tutte le connessioni, riavviare i canali SSL. Vengono utilizzate tutte le nuove impostazioni SSL. Queste impostazioni sono analoghe a quelle aggiornate dal comando REFRESH SECURITY TYPE (SSL) sui sistemi del gestore code.

Quando il client IBM WebSphere MQ MQI viene eseguito su un sistema Windows, UNIX and Linux con hardware crittografico, configurare tale hardware con la variabile di ambiente MQSSLCRYP. Questa variabile equivale al parametro SSLCRYP nel comando ALTER QMGR MQSC. Fare riferimento a ALTER QMGR per una descrizione del parametro SSLCRYP nel comando ALTER QMGR MQSC. Se si utilizza la versione GSK_PCS11 del parametro SSLCRYP, l'etichetta del token PKCS #11 deve essere specificata interamente in minuscolo.

La reimpostazione della chiave segreta SSL e FIPS sono supportate su client IBM WebSphere MQ MQI. Per ulteriori informazioni, consultare “Reimpostazione delle chiavi segrete SSL e TLS” a pagina 224 e “FIPS (Federal Information Processing Standards) per UNIX, Linuxe Windows” a pagina 26.

Consultare “Impostazione della sicurezza client IBM WebSphere MQ MQI” a pagina 108 per ulteriori informazioni sul supporto SSL per i client IBM WebSphere MQ MQI.

Attività correlate

Configurazione di un client utilizzando un file di configurazione

Specifica che un canale MQI utilizza SSL

Per un canale MQI per utilizzare SSL, il valore dell'attributo *SSLCipherSpec* del canale di connessione client deve essere il nome di un CipherSpec supportato da IBM WebSphere MQ sulla piattaforma client.

È possibile definire un canale di collegamento client con un valore per questo attributo nei seguenti modi. Sono elencati in ordine decrescente di precedenza.

1. Quando un'uscita PreConnect fornisce una struttura di definizione del canale da utilizzare.

   Un'uscita PreConnect può fornire il nome di un CipherSpec nel campo *SSLCipherSpec* di MQCD (channel definition structure). Questa struttura viene restituita nel campo **ppMQCDArrayPtr** della struttura del parametro di uscita MQNXP utilizzata dall'exit PreConnect .
2. Quando un'applicazione client MQI WebSphere MQ emette una chiamata MQCONNX.

   L'applicazione può specificare il nome di CipherSpec nel campo *SSLCipherSpec* di una struttura di definizione del canale, MQCD.Questa struttura è indicata dalla struttura delle opzioni di connessione, MQCNO, che è un parametro sulla chiamata MQCONNX.
3. Utilizzo di una CCDT (client channel definition table).

Una o più voci in una tabella di definizione del canale client possono specificare il nome di un CipherSpec.Ad esempio, se si crea una voce utilizzando il comando DEFINE CHANNEL MQSC, è possibile utilizzare il parametro SSLCIPH sul comando per specificare il nome di una CipherSpec.

4. Utilizzo di Active Directory in Windows .

   Sui sistemi Windows , è possibile utilizzare il comando di controllo **setmqscp** per pubblicare le definizioni di canale di connessione client in Active Directory. Una o più di queste definizioni possono specificare il nome di una CipherSpec.

Ad esempio, se un'applicazione client fornisce una definizione di canale di connessione client in una struttura MQCD su una chiamata MQCONNX, questa definizione viene utilizzata in preferenza a tutte le voci in una tabella di definizione di canale client a cui può accedere il client WebSphere MQ .

Non è possibile utilizzare la variabile di ambiente MQSERVER per fornire la definizione del canale all'estremità client di un canale MQI che utilizza SSL.

Per controllare se è stato eseguito il flusso di un certificato client, visualizzare lo stato del canale all'estremità server di un canale per la presenza di un valore del parametro del nome peer.

Concetti correlati

“Specifica di un CipherSpec per un client IBM WebSphere MQ MQI” a pagina 224
Sono disponibili tre opzioni per specificare un CipherSpec per un client MQI IBM WebSphere MQ .

CipherSpecs e CipherSuites in IBM WebSphere MQ

IBM WebSphere MQ supporta sia SSL che TLS CipherSpecse gli algoritmi RSA e Diffie - Hellman.

WebSphere MQ supporta SSL V3 e TLS V1.0 e V1.2 CipherSpecs.

WebSphere MQ supporta lo scambio di chiavi RSA e Diffie - Hellman e gli algoritmi di autenticazione. La dimensione della chiave utilizzata durante l'handshake SSL può dipendere dal certificato digitale utilizzato, ma alcuni CipherSpecs includono una specifica della dimensione della chiave dell'handshake. Una dimensione maggiore della chiave dell'handshake comporta un'autenticazione più avanzata. Con dimensioni della chiave minori, l'handshake risulta più veloce.

Concetti correlati

“CipherSpecs e CipherSuites” a pagina 18
I protocolli di sicurezza crittografici devono concordare gli algoritmi utilizzati da una connessione sicura. CipherSpecs e CipherSuites definiscono combinazioni specifiche di algoritmi.

NSA Suite B Crittografia in IBM WebSphere MQ

Questo argomento fornisce informazioni su come configurare IBM WebSphere MQ su sistemi Windows, Linuxe UNIX per la conformità al profilo TLS 1.2 conforme a Suite B.

Nel corso del tempo, la NSA Cryptography Suite B Standard è stata aggiornata per riflettere nuovi attacchi contro algoritmi e protocolli di crittografia. Ad esempio, alcuni CipherSpecs potrebbero non essere più certificati Suite B. Quando si verificano tali modifiche, anche IBM WebSphere MQ viene aggiornato per implementare lo standard più recente. Di conseguenza, si potrebbero notare dei cambiamenti nelle modalità di funzionamento dopo l'applicazione della manutenzione. Il file readme di IBM WebSphere MQ Version 7.5 elenca la versione di Suite B applicata da ciascun livello di manutenzione del prodotto. Se si configura IBM WebSphere MQ per applicare la conformità Suite B, consultare sempre il file readme quando si pianifica di applicare la manutenzione (consultare IBM MQ, WebSphere MQe i README del prodotto MQSeries).

Su sistemi Windows, UNIXe Linux , IBM WebSphere MQ può essere configurato per essere conforme al profilo TLS 1.2 conforme a Suite B ai livelli di sicurezza riportati nella Tabella 1.

Tabella 1. Livelli di sicurezza della suite B con CipherSpecs e algoritmi di firma digitale consentiti		
Livello di sicurezza	**CipherSpecs consentiti**	**Algoritmi di firma digitale consentiti**
128 bit	ECDHE_ECDSA_AES_128_GCM_SHA256 ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA con SHA-256 ECDSA con SHA-384
192 bit	ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA con SHA-384
Entrambi[1]	ECDHE_ECDSA_AES_128_GCM_SHA256 ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA con SHA-256 ECDSA con SHA-384

1. È possibile configurare contemporaneamente i livelli di sicurezza a 128 bit e a 192 bit. Poiché la configurazione della Suite B determina gli algoritmi crittografici minimi accettabili, la configurazione di entrambi i livelli di protezione è equivalente alla configurazione solo del livello di protezione a 128 bit. Gli algoritmi crittografici del livello di sicurezza a 192 bit sono più potenti del minimo richiesto per il livello di sicurezza a 128 bit, quindi sono consentiti per il livello di sicurezza a 128 bit anche se il livello di sicurezza a 192 bit non è abilitato.

Nota: Le convenzioni di denominazione utilizzate per Livello di sicurezza non rappresentano necessariamente la dimensione della curva ellittica o la dimensione chiave dell'algoritmo di codifica AES.

CipherSpec conformazione a Suite B

Anche se il comportamento predefinito di IBM WebSphere MQ non è conforme allo standard Suite B, IBM WebSphere MQ può essere configurato per essere conforme a uno o a entrambi i livelli di sicurezza sui sistemi Windows, UNIX e Linux . In seguito alla corretta configurazione di IBM WebSphere MQ per utilizzare la suite B, qualsiasi tentativo di avviare un canale in uscita utilizzando un CipherSpec non conforme alla suite B provoca l'errore AMQ9282. Questa attività determina anche la restituzione da parte del client MQI del codice motivo MQRC_CIPHER_SPEC_NOT_SUITE_B. Allo stesso modo, il tentativo di avviare un canale in ingresso utilizzando un CipherSpec non conforme alla configurazione Suite B provoca l'errore AMQ9616.

Per ulteriori informazioni su WebSphere MQ CipherSpecs, consultare “Specifica di CipherSpecs” a pagina 218

Suite B e certificati digitali

Suite B limita gli algoritmi di firma digitale che possono essere utilizzati per firmare i certificati digitali. Suite B inoltre limita il tipo di chiave pubblica che i certificati possono contenere. Pertanto, WebSphere MQ deve essere configurato per utilizzare i certificati il cui algoritmo di firma digitale e il tipo di chiave pubblica sono consentiti dal livello di sicurezza Suite B configurato del partner remoto. I certificati digitali che non sono conformi ai requisiti del livello di sicurezza vengono rifiutati e la connessione non riesce con errore AMQ9633 o AMQ9285.

Per il livello di sicurezza Suite B a 128 bit, la chiave pubblica dell'oggetto certificato è richiesta per utilizzare la curva ellittica NIST P-256 o la curva ellittica NIST P-384 e per essere firmata con la curva ellittica NIST P-256 o la curva ellittica NIST P-384 . A livello di sicurezza Suite B a 192 bit, la chiave pubblica del soggetto del certificato è richiesta per utilizzare la curva ellittica NIST P-384 e per essere firmata con la curva ellittica NIST P-384 .

Per ottenere un certificato adatto per l'operazione conforme a Suite B, utilizzare il comando **runmqakm** e specificare il parametro **-sig_alg** per richiedere un algoritmo di firma digitale adatto. I valori di parametro EC_ecdsa_with_SHA256 e EC_ecdsa_with_SHA384 **-sig_alg** corrispondono alle chiavi della curva ellittica firmate dagli algoritmi di firma digitale Suite B.

Per ulteriori informazioni relative al comando **runmqakm** , consultare opzioni runmqckm e runmqakm.

Nota: Gli strumenti **iKeycmd** e **iKeyman** non supportano la creazione di certificati digitali per l'operazione conforme a Suite B.

Creazione e richiesta di certificati digitali

Per creare un certificato digitale autofirmato per il test Suite B, consultare “Creazione di un certificato personale autofirmato su sistemi UNIX, Linux, and Windows” a pagina 122

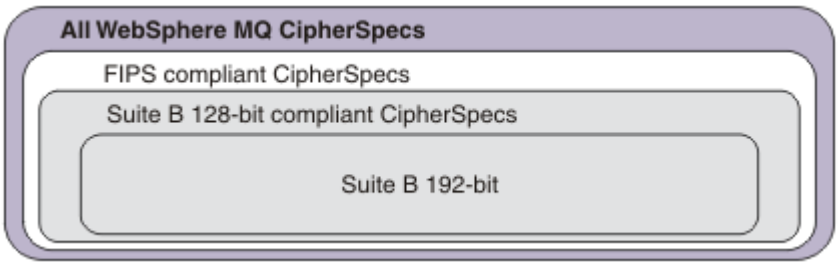
Per richiedere un certificato digitale firmato CA per l'utilizzo di produzione Suite B, consultare “Richiesta di certificato personale su sistemi UNIX, Linux, and Windows” a pagina 124.

Nota: L'autorità di certificazione utilizzata deve generare certificati digitali che soddisfino i requisiti descritti in IETF RFC 6460.

FIPS 140-2 e Suite B

Lo standard Suite B è concettualmente simile a FIPS 140-2, in quanto limita la serie di algoritmi crittografici abilitati al fine di fornire un livello di sicurezza garantito. La suite B CipherSpecs attualmente supportata può essere utilizzata quando IBM WebSphere MQ è configurata per operazioni compatibili con FIPS 140-2. È quindi possibile configurare contemporaneamente WebSphere MQ per la conformità FIPS e Suite B, nel qual caso si applicano entrambe le serie di limitazioni.

Il seguente diagramma illustra la relazione tra questi sottoinsiemi:



Configurazione di WebSphere MQ per operazioni conformi a Suite B

Per informazioni su come configurare IBM WebSphere MQ in Windows, UNIX e Linux per operazioni conformi a Suite B, consultare “Configurazione di IBM WebSphere MQ per Suite B” a pagina 32.

IBM WebSphere MQ non supporta operazioni conformi a Suite B sulle piattaforme IBM i e z/OS . Anche i client WebSphere MQ Java e JMS non supportano le operazioni compatibili con Suite B.

Concetti correlati

“Specifica che solo i CipherSpecs certificati FIPS vengono utilizzati al runtime sul client MQI” a pagina 108
Creare i repository delle chiavi utilizzando il software conforme a FIPS, quindi specificare che il canale deve utilizzare CipherSpecscertificati FIPS.

Configurazione di IBM WebSphere MQ per Suite B
IBM WebSphere MQ può essere configurato per operare in conformità con lo standard NSA Suite B su sistemi UNIX, Linux, and Windows .

La suite B limita la serie di algoritmi di crittografia abilitati per fornire un livello di sicurezza garantito. IBM WebSphere MQ può essere configurato per operare in conformità con la Suite B per fornire un livello di sicurezza migliorato. Per ulteriori informazioni sulla Suite B, consultare “Crittografia della National Security Agency (NSA) Suite B” a pagina 20. Per ulteriori informazioni sulla configurazione della Suite B e il suo effetto sui canali SSL e TLS, vedi “NSA Suite B Crittografia in IBM WebSphere MQ” a pagina 30.

Gestore code

Per un gestore code, utilizzare il comando **ALTER QMGR** con il parametro **SUITEB** per impostare i valori appropriati per il livello di sicurezza richiesto. Per ulteriori informazioni, consultare ALTER QMGR.

È anche possibile utilizzare il comando PCF **MQCMD_CHANGE_Q_MGR** con il parametro **MQIA_SUITE_B_STRENGTH** per configurare il gestore code per l'operazione conforme a Suite B

Client MQI

Per impostazione predefinita, i client MQI non applicano la conformità Suite B. È possibile attivare il client MQI per la conformità Suite B eseguendo una delle seguenti opzioni:

1. Impostando il campo **EncryptionPolicySuiteB** nella struttura MQSCO su una chiamata MQCONNX su uno o più dei seguenti valori:
   - MQ_SUITE_B_NONE
   - MQ_SUITE_B_128_BIT
   - MQ_SUITE_B_192_BIT

   L'utilizzo di MQ_SUITE_B_NONE con qualsiasi altro valore non è valido.
2. Impostando la variabile di ambiente MQSUITEB su uno o più dei seguenti valori:
   - NESSUNO
   - 128_BIT
   - 192_BIT

   È possibile specificare più valori utilizzando un elenco separato da virgole. L'utilizzo del valore NONE con qualsiasi altro valore non è valido.
3. Impostando l'attributo **EncryptionPolicySuiteB** nella stanza SSL del file di configurazione del client MQI su uno o più dei seguenti valori:
   - NESSUNO
   - 128_BIT
   - 192_BIT

   È possibile specificare più valori utilizzando un elenco separato da virgole. L'utilizzo di NONE con qualsiasi altro valore non è valido.

Nota: Le impostazioni del client MQI sono elencate in ordine di priorità. La struttura MSCO sulla chiamata MQCONNX sovrascrive l'impostazione sulla variabile d'ambiente MQSUITEB, che sovrascrive l'attributo nella stanza SSL.

Per dettagli completi sulla struttura MQSCO, consultare MQSCO - Opzioni di configurazione SSL.

Per ulteriori informazioni sull'utilizzo della suite B nel file di configurazione client, consultare Stanza SSL del file di configurazione client .

Per ulteriori informazioni sull'utilizzo della variabile di ambiente MQSUITEB, consultare Variabili di ambiente.

.NET

Per client .NET non gestiti, la proprietà **MQC.ENCRYPTION_POLICY_SUITE_B** indica il tipo di sicurezza Suite B richiesto.

Per informazioni sull'utilizzo della suite B nelle classi IBM WebSphere MQ per .NET, consultare MQEnvironment .NET class.

Politiche di convalida dei certificati in IBM WebSphere MQ
La politica di convalida del certificato determina la conformità della convalida della catena di certificati agli standard di sicurezza del settore.

La politica di convalida del certificato dipende dalla piattaforma e dall'ambiente come segue:

- Per le applicazioni Java e JMS su tutte le piattaforme, la politica di convalida del certificato dipende dal componente JSSE dell'ambiente di runtime Java. Per ulteriori informazioni sulla politica di convalida del certificato, consultare la documentazione per il proprio JRE.
- Per sistemi UNIX, Linux, and Windows , la politica di convalida del certificato viene fornita da GSKit e può essere configurata. Sono supportate due diverse politiche di convalida dei certificati:
  - Una politica di convalida dei certificati legacy, utilizzata per la massima compatibilità e interoperabilità con i vecchi certificati digitali che non sono conformi agli standard di convalida dei certificati IETF correnti. Questa politica è nota come politica di base.
  - Una politica di convalida dei certificati rigorosa e conforme agli standard che applica lo standard RFC 5280. Questa politica è nota come politica standard.

Per informazioni su come configurare la politica di convalida dei certificati su sistemi UNIX, Linux, and Windows , consultare “Configurazione delle politiche di convalida dei certificati in IBM WebSphere MQ” a pagina 34. Per ulteriori informazioni sulle differenze tra le politiche di convalida dei certificati di base e standard, consultare Convalida dei certificati e progettazione delle politiche di trust sui sistemi UNIX, Linux e Windows .

Configurazione delle politiche di convalida dei certificati in IBM WebSphere MQ
È possibile specificare quale politica di convalida del certificato SSL/TLS viene utilizzata per convalidare i certificati digitali ricevuti dai sistemi partner remoti in quattro modi.

Sul gestore code, la politica di convalida del certificato può essere impostata nei seguenti modi:

- Utilizzo dell'attributo del gestore code *CERTVPOL*. Per ulteriori informazioni sull'impostazione di questo attributo, consultare ALTER QMGR .

Sul client, ci sono diversi metodi che possono essere utilizzati per impostare la politica di convalida del certificato. Se si utilizza più di un metodo per impostare la politica, il client utilizza le impostazioni nel seguente ordine di priorità:

1. Utilizzo del campo *CertificateValPolicy* nella struttura MQSCO client. Per ulteriori informazioni sull'utilizzo di questo campo, consultare MQSCO - Opzioni di configurazione SSL.
2. Utilizzo della variabile di ambiente client, *MQCERTVPOL*. Per ulteriori informazioni sull'utilizzo di questa variabile, consultare MQCERTVPOL .
3. Utilizzando l'impostazione del parametro di regolazione della stanza SSL client, *CertificateValPolicy*. Per ulteriori informazioni sull'utilizzo di questa impostazione, consultare Stanza SSL del file di configurazione client .

Per ulteriori informazioni sulle politiche di convalida dei certificati, consultare “Politiche di convalida dei certificati in IBM WebSphere MQ” a pagina 33.

Certificati digitali e compatibilità CipherSpec in IBM WebSphere MQ
Questo argomento fornisce informazioni su come scegliere i CipherSpecs e i certificati digitali appropriati per la politica di sicurezza, evidenziando la relazione tra CipherSpecs e i certificati digitali in IBM WebSphere MQ.

Nei rilasci precedenti di IBM WebSphere MQ, tutti i CipherSpecs SSL e TLS supportati utilizzavano l'algoritmo RSA per le firme digitali e l'accordo chiave. Tutti i tipi di certificati digitali supportati erano compatibili con tutti i CipherSpecssupportati, quindi è stato possibile modificare la CipherSpec per qualsiasi canale senza dover modificare i certificati digitali.

In IBM WebSphere MQ v7.5 , è possibile utilizzare solo un sottoinsieme dei CipherSpecs supportati con tutti i tipi di certificato digitale supportati. È quindi necessario scegliere un CipherSpec appropriato per il certificato digitale. Allo stesso modo, se la politica di sicurezza della propria organizzazione richiede l'utilizzo di un particolare CipherSpec , è necessario ottenere un certificato digitale appropriato per tale CipherSpec.

Algoritmo di firma digitale MD5 e TLS 1.2

I certificati digitali firmati utilizzando l'algoritmo MD5 vengono rifiutati quando si utilizza il protocollo TLS 1.2 . Ciò è dovuto al fatto che l'algoritmo MD5 è ora considerato debole da molti analisti crittografici e il suo utilizzo è generalmente sconsigliato. Se si desidera utilizzare CipherSpecs più recenti basati sul protocollo TLS 1.2 , accertarsi che i certificati digitali non utilizzeranno l'algoritmo MD5 nelle relative firme digitali. I CipherSpecs meno recenti che utilizzano i protocolli SSL 3.0 e TLS 1.0 non sono soggetti a questa restrizione e possono continuare a utilizzare certificati con firme digitali MD5 .

Per visualizzare l'algoritmo di firma digitale per un determinato certificato, è possibile utilizzare il comando **runmqakm** :

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

dove `cert_label` è l'etichetta del certificato dell'algoritmo di firma digitale che è necessario visualizzare.

Nota: Sebbene lo strumento **iKeycmd (runmqckm)** e la GUI **iKeyman (strmqikm)** possano essere utilizzati per visualizzare una selezione di algoritmi di firma digitale, lo strumento **runmqakm** fornisce una gamma più ampia.

L'esecuzione del comando **runmqakm** produrrà l'output che visualizza l'utilizzo dell'algoritmo di firma specificato:

```
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 riga `Signature Algorithm` indica che viene utilizzato l'algoritmo `MD5WithRSASignature` . Questo algoritmo è basato su MD5 e, pertanto, questo certificato digitale non può essere utilizzato con TLS 1.2 CipherSpecs.

Interoperabilità di Elliptic Curve e RSA CipherSpecs

Non tutti i CipherSpecs possono essere utilizzati con tutti i certificati digitali. Esistono tre tipi di CipherSpec, indicati dal prefisso del nome CipherSpec . Ogni tipo di CipherSpec impone diverse restrizioni

sul tipo di certificato digitale che può essere utilizzato. Queste limitazioni si applicano a tutte le connessioni SSL e TLS di WebSphere MQ , ma sono particolarmente rilevanti per gli utenti della crittografia Elliptic Curve.

Le relazioni tra CipherSpecs e certificati digitali sono riepilogate nella seguente tabella:

Tabella 2. Relazioni tra CipherSpecs e certificati digitali

Tipo	Prefisso nome CipherSpec	Descrizione	Tipo di chiave pubblica richiesto	Algoritmo di crittografia della firma digitale	Metodo di creazione della chiave segreta
1	ECDHE_ECDSA_	CipherSpecs che utilizzano le chiavi pubbliche Elliptic Curve, le chiavi segrete Elliptic Curve e gli algoritmi di firma digitale Elliptic Curve.	Curva ellittica	ECDSA	ECDHE
2	ECDHE_RSA_	CipherSpecs che utilizzano le chiavi pubbliche RSA, le chiavi segrete Elliptic Curve e gli algoritmi di firma digitale Elliptic Curve.	RSA	RSA	ECDHE
3	(Tutti gli altri)	CipherSpecs che utilizzano le chiavi pubbliche RSA e gli algoritmi di firma digitale RSA.	RSA	RSA	RSA

Nota: I CipherSpecs di tipo 1 e 2 sono supportati solo da gestori code WebSphere MQ e client MQ sulle piattaforme UNIX, Linux, and Windows .

La colonna del tipo di chiave pubblica richiesta mostra il tipo di chiave pubblica che il certificato personale deve avere quando si utilizza ciascun tipo di CipherSpec. Il certificato personale è il certificato di entità finale che identifica il gestore code o il client per il partner remoto.

L'algoritmo di codifica della firma digitale fa riferimento all'algoritmo di codifica utilizzato per convalidare il peer. L'algoritmo di crittografia viene utilizzato insieme a un algoritmo hash come MD5, SHA-1 o SHA-256 per calcolare la firma digitale. Esistono vari algoritmi di firma digitale che possono essere utilizzati, ad esempio "RSA con MD5" o "ECDSA con SHA-256". Nella tabella, ECDSA si riferisce alla serie di algoritmi di firma digitale che utilizzano ECDSA; RSA si riferisce alla serie di algoritmi di firma digitale che utilizzano RSA. È possibile utilizzare qualsiasi algoritmo di firma digitale supportato nell'insieme, purché sia basato sull'algoritmo di codifica indicato.

I CipherSpecs di tipo 1 richiedono che il certificato personale disponga di una chiave pubblica Elliptic Curve. Quando si utilizzano questi CipherSpecs , viene utilizzato l'accordo di chiave effimera Elliptic Curve Diffie Hellman per stabilire la chiave segreta per la connessione.

I CipherSpecs di tipo 2 richiedono che il certificato personale abbia una chiave pubblica RSA. Quando si utilizzano questi CipherSpecs , viene utilizzato l'accordo di chiave effimera Elliptic Curve Diffie Hellman per stabilire la chiave segreta per la connessione.

Il tipo 3 CipherSpecs richiede che il certificato personale disponga di una chiave pubblica RSA. Quando vengono utilizzati questi CipherSpecs , viene utilizzato lo scambio di chiavi RSA per stabilire la chiave segreta per la connessione.

Questo elenco di limitazioni non è esaustivo: a seconda della configurazione, potrebbero essere presenti ulteriori limitazioni che possono influire ulteriormente sulla capacità di interagire. Ad esempio, se WebSphere MQ è configurato per essere conforme agli standard FIPS 140-2 o NSA Suite B, ciò limiterà anche l'intervallo di configurazioni consentite. Fare riferimento alla seguente sezione per ulteriori informazioni.

Un gestore code WebSphere MQ può utilizzare solo un certificato personale per identificarsi. Ciò significa che tutti i canali sul gestore code utilizzeranno lo stesso certificato digitale e quindi ogni gestore code può utilizzare solo un tipo di CipherSpec alla volta. Allo stesso modo, un'applicazione client WebSphere MQ può utilizzare solo un singolo certificato personale per identificarsi. Questo significa che tutte le connessioni SSL e TLS all'interno di un singolo processo dell'applicazione utilizzeranno lo stesso certificato digitale e quindi ogni processo dell'applicazione client può utilizzare solo un tipo di CipherSpec alla volta.

I tre tipi di CipherSpec non interagiscono direttamente: questa è una limitazione degli standard SSL e TLS correnti. Ad esempio, si supponga di aver scelto di utilizzare ECDHE_ECDSA_AES_128_CBC_SHA256 CipherSpec per un canale ricevente denominato TO.QM1 su un gestore code denominato QM1. ECDHE_ECDSA_AES_128_CBC_SHA256 è una CipherSpecdi Tipo 1, quindi QM1 deve disporre di un certificato personale con una chiave Elliptic Curve e una firma digitale basata su ECDSA. Tutti i client e gli altri gestori code che comunicano direttamente con QM1 devono quindi avere certificati digitali che soddisfino i requisiti CipherSpec di tipo 1. Altri canali che si collegano al gestore code QM1 possono utilizzare altri CipherSpecs (ad esempio ECDHE_ECDSA_3DES_EDE_CBC_SHA256), ma possono utilizzare solo CipherSpecs di tipo 1 per comunicare con QM1.

Quando si pianificano le reti WebSphere MQ , considerare attentamente quali canali richiedono SSL o TLS e verificare che tutti i client e i gestori code che devono interoperare utilizzino lo stesso tipo di CipherSpecs e i certificati digitali appropriati. Gli standard IETF RFC 4492, RFC 5246 e RFC 6460 descrivono l'utilizzo dettagliato di Elliptic Curve CipherSpecs in TLS 1.2.

Per visualizzare l'algoritmo di firma digitale e il tipo di chiave pubblica per un certificato digitale, è possibile utilizzare il comando **runmqakm** :

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

dove `cert_label` è l'etichetta del certificato di cui è necessario visualizzare l'algoritmo di firma digitale.

L'esecuzione del comando **runmqakm** produrrà l'output che visualizza il tipo di chiave pubblica:

```
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
```

La riga Tipo di chiave pubblica in questo caso mostra che il certificato ha una chiave pubblica della curva ellittica. La riga Algoritmo di firma in questo caso mostra che l'algoritmo EC_ecdsa_with_SHA384 è utilizzato: si basa sull'algoritmo ECDSA. Questo certificato è quindi adatto solo per l'utilizzo con CipherSpecsdi Tipo 1.

È inoltre possibile utilizzare lo strumento **iKeycmd (runmqckm)** con gli stessi parametri. Inoltre, la GUI **iKeyman (strmqikm)** può essere utilizzata per visualizzare gli algoritmi di firma digitale se si apre il repository delle chiavi e si fa doppio clic sull'etichetta del certificato. Tuttavia, si consiglia di utilizzare lo strumento **runmqakm** per visualizzare i certificati digitali perché supporta una gamma più ampia di algoritmi.

Curva ellittica CipherSpecs e NSA Suite B

Quando WebSphere MQ viene configurato per essere conforme al profilo TLS 1.2 conforme a Suite B, gli CipherSpecs e gli algoritmi di firma digitale consentiti sono limitati come descritto in “NSA Suite B Crittografia in IBM WebSphere MQ” a pagina 30. Inoltre, la gamma di chiavi Elliptic Curve accettabili è ridotta in base al livello di sicurezza configurato.

Al livello di sicurezza Suite B a 128 bit, la chiave pubblica del soggetto del certificato è richiesta per utilizzare la curva ellittica NIST P-256 o NIST P-384 e per essere firmata con la curva ellittica NIST P-256 o la curva ellittica NIST P-384 . Il comando **runmqakm** può essere utilizzato per richiedere certificati digitali per questo livello di sicurezza utilizzando un parametro -sig_alg di EC_ecdsa_with_SHA256o EC_ecdsa_with_SHA384.

Al livello di sicurezza B della suite a 192 bit, la chiave pubblica del soggetto del certificato è richiesta per utilizzare la curva ellittica NIST P-384 e per essere firmata con la curva ellittica NIST P-384 . Il comando **runmqakm** può essere utilizzato per richiedere certificati digitali per questo livello di sicurezza utilizzando un parametro -sig_alg di EC_ecdsa_with_SHA384.

Le curve ellittiche NIST supportate sono le seguenti:

Tabella 3. Curve ellittiche NIST supportate

Nome curva NIST FIPS 186 - 3	Nome curva RFC 4492	Dimensione chiave curva ellittica (bit)
P-256	secp256r1	256
P-384	secp384r1	384
P-521	secp521r1	521

Nota: La curva ellittica NIST P-521 non può essere utilizzata per operazioni conformi alla Suite B.

Concetti correlati

“Specifica di CipherSpecs” a pagina 218
Specificare una CipherSpec utilizzando il parametro **SSLCIPH** nel comando MQSC **DEFINE CHANNEL** o nel comando MQSC **ALTER CHANNEL** .

“Specifica che solo i CipherSpecs certificati FIPS vengono utilizzati al runtime sul client MQI” a pagina 108

Creare i repository delle chiavi utilizzando il software conforme a FIPS, quindi specificare che il canale deve utilizzare CipherSpecscertificati FIPS.

"NSA Suite B Crittografia in IBM WebSphere MQ" a pagina 30
Questo argomento fornisce informazioni su come configurare IBM WebSphere MQ su sistemi Windows, Linuxe UNIX per la conformità al profilo TLS 1.2 conforme a Suite B.

"Crittografia della National Security Agency (NSA) Suite B" a pagina 20
Il governo degli Stati Uniti d'America fornisce consulenza tecnica sui sistemi IT e sulla sicurezza, compresa la crittografia dei dati. La US National Security Agency (NSA) raccomanda una serie di algoritmi di crittografia interoperabili nel suo standard Suite B.

Valori CipherSpec supportati in IBM WebSphere MQ
La serie di CipherSpecs predefiniti consente solo i seguenti valori:

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

Abilitazione di CipherSpecs obsoleti

Per impostazione predefinita, non è consentito specificare una CipherSpec obsoleta su una definizione di canale. Se si tenta di specificare una CipherSpecobsoleta, si riceve il messaggio AMQ9788 nel log degli errori per il gestore code.

È possibile riattivare i CipherSpecs obsoleti modificando il file `qm.ini` . All'interno della stanza SSL del file `qm.ini` , aggiungere la riga seguente:

```
SSL:
AllowWeakCipherSpec=Yes
```

È anche possibile riattivare uno o più dei CipherSpecs obsoleti al runtime sul server impostando la variabile di ambiente *AMQ_SSL_WEAK_CIPHER_ENABLE* su qualsiasi valore. Questa variabile di ambiente abilita CipherSpecs indipendentemente dal valore specificato nel file `qm.ini` .

Record di autenticazione di canale

Per esercitare un controllo più preciso sull'accesso concesso ai sistemi di connessione a livello di canale, è possibile utilizzare i record di autenticazione di canale.

Si potrebbe rilevare che i client tentano di connettersi al proprio gestore code utilizzando ID utente vuoti o un ID utente di alto livello che consentirebbe al client di eseguire azioni indesiderate. È possibile bloccare

l'accesso a questi client utilizzando record di autenticazione di canale. In alternativa, un client potrebbe dichiarare un ID utente valido nella piattaforma client, ma è sconosciuto oppure di un formato non valido nella piattaforma server. È possibile utilizzare un record di autenticazione di canale per associare l'ID utente dichiarato a un ID utente valido.

Si potrebbe rilevare che un'applicazione client che si collega al proprio gestore code e che si comporta in modo errato in qualche modo. Per proteggere il server dai problemi causati da questa applicazione, è necessario bloccarlo temporaneamente utilizzando l'indirizzo IP in cui si trova l'applicazione client fino a quando vengono aggiornate le regole del firewall oppure viene corretta l'applicazione client. È possibile utilizzare un record di autenticazione di canale per bloccare l'indirizzo IP da cui si connette l'applicazione client.

Se è stato configurato uno strumento di amministrazione come IBM WebSphere MQ Explorer e un canale per quello specifico utilizzo, è possibile fare in modo che soltanto computer client specifici possano utilizzarlo. È possibile utilizzare un record di autenticazione di canale per consentire al canale di essere utilizzato soltanto da determinati indirizzi IP.

Se si sta iniziando con alcune applicazioni di esempio in esecuzione come client, consultare Preparazione ed esecuzione dei programmi di esempio per un esempio di impostazione sicura del gestore code utilizzando i record di autenticazione di canale.

Per richiamare i record di autenticazione di canale per controllare i canali in entrata, utilizzare il comando MQSC **ALTER QMGR CHLAUTH(ENABLED)**.

Le regole **CHLAUTH** vengono applicate per un MCA del canale creato in risposta a una nuova connessione in entrata. Per un MCA del canale creato in risposta all'avvio locale del canale, non vengono applicate regole **CHLAUTH**.

Tabella 4. Dove le regole CHLAUTH vengono applicate per diverse coppie di canali

Tipo di canale	**MCA dove vengono applicate le regole CHLAUTH**
SDR-RCVR	RCVR
RQSTR-SVR (avviato a SVR)	RQSTR
RQSTR-SVR (avviato a RQSTR)	SVR
RQSTR-SDR (avviato a SDR)	RQSTR
RQSTR-SDR (avviato a RQSTR)	SDR per la connessione iniziale. RQSTR per la connessione di callback.

I record di autenticazione di canale possono essere creati per eseguire le seguenti funzioni:

- Bloccare le connessioni da indirizzi IP specifici.
- Bloccare le connessioni da ID utente specifici.
- Impostare un valore MCAUSER da utilizzare per qualsiasi canale che si connette da un indirizzo IP specifico.
- Impostare un valore MCAUSER da utilizzare per qualsiasi canale che dichiara un ID utente specifico.
- Impostare un valore MCAUSER da utilizzare per qualsiasi canale che ha un DN (Distinguished Name) SSL o TLS specifico.
- Impostare un valore MCAUSER da utilizzare per qualsiasi canale che si connette da un gestore code specifico.
- Bloccare le connessioni che dichiarano di provenire da un determinato gestore code, salvo il caso in cui la connessione proviene da un indirizzo IP specifico.
- Bloccare le connessioni che presentano un determinato certificato SSL o TLS, salvo il caso in cui la connessione proviene da un indirizzo IP specifico.

Tali modalità di utilizzo vengono descritte in modo più dettagliato nelle sezioni che seguono.

Creare, modificare o rimuovere i record di autenticazione di canale utilizzando il comando MQSC **SET CHLAUTH** o il comando PCF **Set Channel Authentication Record**.

Nota: Un numero elevato di record di autenticazione di canale può avere un impatto negativo sulle prestazioni di un gestore code.

Blocco degli indirizzi IP

Di solito, spetta al firewall prevenire l'accesso da parte di determinati indirizzi IP. Tuttavia, a volte possono verificarsi tentativi di connessione da un indirizzo IP che non dovrebbe avere accesso al sistema WebSphere MQ ed è necessario bloccare temporaneamente l'indirizzo prima dell'aggiornamento del firewall. Questi tentativi di connessione potrebbero non provenire nemmeno dai canali WebSphere MQ , ma da altre applicazioni socket non configurate correttamente per il listener WebSphere MQ . Bloccare gli indirizzi IP impostando un record di autenticazione di canale di tipo BLOCKADDR. È possibile specificare uno o più indirizzi singoli, intervalli di indirizzi o modelli compresi i caratteri jolly.

Ogni volta che una connessione in entrata viene rifiutata a causa di questo blocco dell'indirizzo IP, viene emesso un messaggio di evento MQRC_CHANNEL_BLOCKED con il qualificatore motivo MQRQ_CHANNEL_BLOCKED_ADDRESS, a condizione che gli eventi del canale siano abilitati e che il gestore code sia in esecuzione. Inoltre, la connessione è tenuta aperta per 30 secondi prima di restituire l'errore, per assicurare che il listener non venga sovraccaricato con tentativi ripetuti di connessione bloccati.

Per bloccare gli indirizzi IP solo su specifici canali o per evitare il ritardo prima che l'errore venga riportato, impostare un record di autenticazione di canale di tipo ADDRESSMAP con il parametro USERSRC(NOACCESS).

Ogni volta che una connessione in entrata viene rifiutata per questo motivo, viene emesso un messaggio di evento MQRC_CHANNEL_BLOCKED con il qualificatore motivo MQRQ_CHANNEL_BLOCKED_NOACCESS, a condizione che gli eventi del canale siano abilitati e che il gestore code sia in esecuzione.

Per un esempio, consultare “Blocco di specifici indirizzi IP” a pagina 182.

Blocco degli ID utente

Per evitare che determinati ID utente si connettano a un canale client, impostare un record di autenticazione di canale di tipo BLOCKUSER. Questo tipo di record di autenticazione di canale si applica soltanto ai canali client, non ai canali di messaggi. È possibile specificare uno o più ID utente singoli da bloccare, ma non è possibile utilizzare caratteri jolly.

Ogni volta che una connessione in entrata viene rifiutata per questo motivo, viene emesso un messaggio di evento MQRC_CHANNEL_BLOCKED con identificativo motivo MQRQ_CHANNEL_BLOCKED_USERID, a condizione che gli eventi del canale siano abilitati.

Per un esempio, consultare “Blocco di ID utente specifici” a pagina 184.

È anche possibile bloccare qualsiasi accesso per gli ID utente specificati in determinati canali impostando un record di autenticazione di canale di tipo USERMAP con parametro USERSRC(NOACCESS).

Ogni volta che una connessione in entrata viene rifiutata per questo motivo, viene emesso un messaggio di evento MQRC_CHANNEL_BLOCKED con il qualificatore motivo MQRQ_CHANNEL_BLOCKED_NOACCESS, a condizione che gli eventi del canale siano abilitati e che il gestore code sia in esecuzione.

Per un esempio, consultare “Blocco dell'accesso per un ID utente dichiarato dal client” a pagina 187.

Blocco dei nomi dei gestori code

Per specificare che qualsiasi canale che si connette da un gestore code specificato non deve avere alcun accesso, impostare un record di autenticazione di canale di tipo QMGRMAP con il parametro USERSRC(NOACCESS). È possibile specificare un unico nome o modello del gestore code, compresi i caratteri jolly. Non esiste alcun equivalente della funzione BLOCKUSER per bloccare l'accesso dai gestori code.

Ogni volta che una connessione in entrata viene rifiutata per questo motivo, viene emesso un messaggio di evento MQRC_CHANNEL_BLOCKED con il qualificatore motivo MQRQ_CHANNEL_BLOCKED_NOACCESS, a condizione che gli eventi del canale siano abilitati e che il gestore code sia in esecuzione.

Per un esempio, consultare "Blocco dell'accesso da un gestore code remoto" a pagina 186.

Blocco dei DN SSL o TLS

Per specificare che qualsiasi utente che presenta un certificato personale SSL o TLS contenente un DN specificato non deve avere alcun accesso, impostare un record di autenticazione di canale di tipo SSLPEERMAP con il parametro USERSRC(NOACCESS). È possibile specificare un unico DN (distinguished name) o modello, compresi i caratteri jolly. Non esiste alcun equivalente della funzione BLOCKUSER per bloccare l'accesso per i DN.

Ogni volta che una connessione in entrata viene rifiutata per questo motivo, viene emesso un messaggio di evento MQRC_CHANNEL_BLOCKED con il qualificatore motivo MQRQ_CHANNEL_BLOCKED_NOACCESS, a condizione che gli eventi del canale siano abilitati e che il gestore code sia in esecuzione.

Per un esempio, consultare "Blocco dell'accesso per un DN (Distinguished Name) SSL" a pagina 187.

Associazione degli indirizzi IP agli ID utente da utilizzare

Per specificare che qualsiasi canale che si connette da un indirizzo IP specificato deve utilizzare un MCAUSER specificato, impostare un record di autenticazione di canale di tipo ADDRESSMAP. È possibile specificare un unico indirizzo, un intervallo di indirizzi o un modello compresi i caratteri jolly.

Se si utilizza un servizio di inoltro di porte, un'interruzione di sessione DMZ o qualsiasi altra configurazione che modifica l'indirizzo IP presentato al gestore code, l'associazione degli indirizzi IP non è necessariamente adeguata per l'uso.

Per un esempio, consultare "Associazione di un indirizzo IP a un ID utente MCAUSER" a pagina 187.

Associazione dei nomi dei gestori code agli ID utente da utilizzare

Per specificare che qualsiasi canale che si connette da un gestore code specificato deve utilizzare un MCAUSER specifico, impostare un record di autenticazione di canale di tipo QMGRMAP. È possibile specificare un unico nome o modello del gestore code, compresi i caratteri jolly.

Per un esempio, consultare "Associazione di un gestore code remoto a un ID utente MCAUSER" a pagina 184.

Associazione degli ID utente dichiarati da un client agli ID utente da utilizzare

Per specificare che se un determinato ID utente viene utilizzato da una connessione da un client WebSphere MQ MQI, è necessario utilizzare un MCAUSER specificato, diverso, impostare un record di autenticazione di canale di tipo USERMAP. L'associazione degli ID utente non utilizza caratteri jolly.

Per un esempio, consultare "Associazione di un ID utente asserito dal client ad un ID utente MCAUSER" a pagina 185.

Associazione dei DN SSL o TLS agli ID utente da utilizzare

Per specificare che qualsiasi utente che presenta un certificato personale SSL/TLS contenente un DN specificato deve utilizzare un MCAUSER specifico, impostare un record di autenticazione di canale di tipo SSLPEERMAP. È possibile specificare un unico DN (distinguished name) o modello, compresi i caratteri jolly.

Per un esempio, consultare "Associazione di un DN (Distinguished Name) SSL o TLS a un ID utente MCAUSER" a pagina 186.

Associazione di gestori code, client o DN SSL o TLS in base all'indirizzo IP

È possibile che, a volte, un terzo interferisca con un nome del gestore code. Un certificato SSL o TLS o un file del database delle chiavi potrebbe anche essere sottratto e riutilizzato. Per proteggersi da queste minacce, è possibile specificare che una connessione da un determinato gestore code o client oppure che per l'uso di un determinato DN è necessario connettersi da un indirizzo IP specificato. Impostare un record di autenticazione di canale di tipo USERMAP, QMGRMAP o SSLPEERMAP e specificare l'indirizzo IP consentito o il pattern degli indirizzi IP utilizzando il parametro ADDRESS.

Per un esempio, consultare "Associazione di un gestore code remoto a un ID utente MCAUSER" a pagina 184.

Interazione tra i record di autenticazione di canale

È possibile che un canale che tenta di connettersi corrisponda a più record di autenticazione di canale e che abbia effetti contraddittori. Ad esempio, un canale potrebbe dichiarare un ID utente bloccato da un record di autenticazione di canale BLOCKUSER, ma con un certificato SSL o TLS che corrisponde a un record SSLPEERMAP che imposta un ID utente diverso. Inoltre, se i record di autenticazione di canale utilizzano caratteri jolly, un unico indirizzo IP, un nome del gestore code o DN SSL o TLS potrebbero corrispondere a diversi modelli. Ad esempio, l'indirizzo IP 192.0.2.6 corrisponde ai modelli 192.0.2.0-24, 192.0.2. *, e 192.0.*.6. L'azione intrapresa viene determinata come segue.

- Il record di autenticazione di canale utilizzato viene selezionato nel seguente modo:
  - Un record di autenticazione di canale che corrisponde esplicitamente al nome del canale assume la priorità su un record di autenticazione di canale che corrisponde al nome del canale utilizzando un carattere jolly.
  - Un record di autenticazione di canale che utilizza un DN SSL o TLS assume la priorità su un record utilizzando un ID utente, un nome del gestore code o un indirizzo IP.
  - Un record di autenticazione di canale che utilizza un ID utente o un nome del gestore code assume la priorità su un record utilizzando un indirizzo IP.
- Se viene trovato un record di autenticazione di canale corrispondente che specifica un MCAUSER, questo MCAUSER viene assegnato al canale.
- Se viene trovato un record di autenticazione di canale corrispondente che specifica che il canale non ha alcun accesso, al canale viene assegnato un valore MCAUSER di *NOACCESS. Questo valore può essere modificato in un secondo momento da un programma di uscita di sicurezza.
- Se non viene trovato alcun record di autenticazione di canale corrispondente oppure viene trovato un record di autenticazione di canale corrispondente che specifica che deve essere utilizzato l'ID del canale, il campo MCAUSER viene ispezionato.
  - Se il campo MCAUSER è vuoto, l'ID utente client viene assegnato al canale.
  - Se il campo MCAUSER non è vuoto, viene assegnato al canale.
- Viene eseguito qualsiasi programma di uscita di sicurezza. Questo programma delle uscite consente di impostare l'ID utente del canale oppure stabilire il blocco degli accessi.
- Se la connessione è bloccata oppure se MCAUSER è impostato su *NOACCESS, il canale termina.
- Se la connessione non è bloccata, per qualsiasi canale ad eccezione di un canale client, l'ID utente del canale stabilito nei passaggi precedenti viene confrontato con l'elenco degli utenti bloccati.
  - Se l'ID utente è presente nell'elenco degli utenti bloccati, il canale termina.
  - Se l'ID utente non è presente nell'elenco degli utenti bloccati, il canale viene eseguito.

Se una serie di record di autenticazione di canale corrisponde a un nome canale, indirizzo IP, nome del gestore code o DN SSL o TLS, viene utilizzata la corrispondenza più specifica. La corrispondenza considerata più specifica viene stabilita nel seguente modo.

- Per un nome del canale:
  - La corrispondenza più specifica è un nome senza caratteri jolly, ad esempio A.B.C.
  - La corrispondenza più generica è un singolo asterisco (*), che corrisponde a tutti i nomi dei canali.

  - Un modello con un asterisco nella posizione più a sinistra è più generico di un modello con un valore definito nella posizione più a sinistra. Quindi, *.B.C è più generico di A.*.
  - Un modello con un asterisco nella seconda posizione è più generico di un modello con un valore definito nella seconda posizione; lo stesso vale per tutte le posizioni successive. Quindi, A.*.C è più generico di A.B.*
  - Dove due o più modelli contengono un asterisco nella stessa posizione, quello con la quantità minore di nodi dopo l'asterisco è più generico. Così A. * è più generico di A.*.C
- Per un indirizzo IP:
  - La corrispondenza più specifica è un nome senza caratteri jolly, ad esempio 192.0.2.6.
  - La corrispondenza più generica è un singolo asterisco (*), che corrisponde a tutti i nomi dei canali.
  - Un modello con un asterisco nella posizione più a sinistra è più generico di un modello con un valore definito nella posizione più a sinistra. Quindi, *.0.2.6 è più generico di 192.*.
  - Un modello con un asterisco nella seconda posizione è più generico di un modello con un valore definito nella seconda posizione; lo stesso vale per tutte le posizioni successive. Quindi, 192.*.2.6 è più generico di 192.0.*.
  - Dove due o più modelli contengono un asterisco nella stessa posizione, quello con la quantità minore di nodi dopo l'asterisco è più generico. Quindi 192 .* è più generico di 192.*.2.*.
  - Un intervallo indicato con un trattino (-) è più specifico di un asterisco. Quindi, 192.0.2.0-24 è più specifico di 192.0.2.*.
  - Un intervallo che è un sottoinsieme di un altro è più specifico dell'intervallo più grande. Quindi, 192.0.2.5-15 è più specifico di 192.0.2.0-24.
  - Gli intervalli sovrapposti non sono consentiti. Ad esempio, non è possibile avere record di autenticazione di canale sia per 192.0.2.0-15 che per 192.0.2.10-20.
  - Un modello non può avere un numero di parti inferiore a quello obbligatorio, fatto salvo il caso in cui il modello termina con un unico asterisco finale. Ad esempio 192.0.2 non è valido, ma 192.0.2.* è valido.
  - Un asterisco finale deve essere separato dal resto dell'indirizzo dall'appropriato separatore di parti (un punto (.) per IPv4, due punti (:) per IPv6). Ad esempio, 192.0* non è valido in quanto l'asterisco non si trova in una parte.
  - Un modello può contenere asterischi aggiunti a condizione che nessun asterisco sia adiacente all'asterisco finale. Ad esempio, 192.*.2.* è valido, ma 192,0.*.* non è valida.
  - Un modello di indirizzo IPv6 non può contenere un segno di due punti doppio e un asterisco finale, in quanto l'indirizzo risultante sarebbe ambiguo. Ad esempio, 2001::* potrebbe espandersi a 2001:0000:*, 2001:0000:0000:* e così via
- Per un nome del gestore code:
  - La corrispondenza più specifica è un nome senza caratteri jolly, ad esempio 192.0.2.6.
  - La corrispondenza più generica è un singolo asterisco (*), che corrisponde a tutti i nomi dei canali.
  - Un modello con un asterisco nella posizione più a sinistra è più generico di un modello con un valore definito nella posizione più a sinistra. Quindi, *QUEUEMANAGER è più generico di QUEUEMANAGER*.
  - Un modello con un asterisco nella seconda posizione è più generico di un modello con un valore definito nella seconda posizione; lo stesso vale per tutte le posizioni successive. Quindi, Q*MANAGER è più generico di QUEUE*.
  - Dove due o più modelli contengono un asterisco nella stessa posizione, quello con il numero inferiore di caratteri dopo l'asterisco è più generico. Quindi Q* è più generico di Q*MGR.
- Per un DN (Distinguished Name) SSL o TLS, l'ordine di precedenza delle sottostringhe è il seguente:

Tabella 5. Ordine di precedenza delle sottostringhe

Ordina	**Sottostringa DN**	**Nome**
1	SERIALNUMBER=	Numero di serie del certificato

Tabella 5. Ordine di precedenza delle sottostringhe (Continua)		
Ordina	**Sottostringa DN**	**Nome**
2	MAIL=	Indirizzo e-mail
3	E=	Indirizzo e-mail (obsoleto, preferenza:n MAIL)
4	UID=, USERID=	Identificativo utente
5	CN=	CN (Common Name)
6	T =	Titolo
7	OU=	Unità organizzativa
8	DC=	Componente dominio
9	O=	Organizzazione
10	STREET=	Via / Prima riga dell'indirizzo
11	L=	Località
12	ST=, SP=, S=	Stato o provincia
13	PC=	Codice postale
14	C=	Paese (Country)
15	UNSTRUCTUREDNAME=	Nome host
16	UNSTRUCTUREDADDRESS=	Indirizzo IP
17	DNQ=	Identificativo DN (Distinguished Name)

Quindi, se un certificato SSL o TLS viene presentato con un DN contenente le sottostringhe O=IBM e C=UK, WebSphere MQ utilizza un record di autenticazione di canale per O=IBM, preferendolo a C=UK, se sono presenti entrambi.

Un DN può contenere più OU, le quali devono essere specificate nell'ordine gerarchico con le unità organizzative grandi specificate per prime. Se due DN sono uguali sotto tutti gli aspetti, tranne che per i valori OU, il DN più specifico viene determinato nel seguente modo:

1. Se hanno numeri diversi di attributi OU, il DN con più valori OU è più specifico. Questo perché il DN con un numero maggiore di unità organizzative completa il DN in modo più dettagliato e fornisce un numero maggiore di criteri di corrispondenza. Anche se l'OU di massimo livello è un carattere jolly (OU=*), il DN con più OU viene ancora considerato come il più specifico.
2. Se hanno lo stesso numero di attributi OU, le coppie corrispondenti di valori OU vengono confrontate nella sequenza da sinistra a destra, dove l'OU più a sinistra è il livello più alto (meno specifico), in base alle seguenti regole.
   a. Un OU che non contiene alcun carattere jolly è il più specifico in quanto può corrispondere esattamente solo a uno stringa.
   b. Un OU con un unico carattere jolly all'inizio o alla fine (ad esempio, OU=ABC* oppure OU=*ABC) è quello successivo più specifico.
   c. Un OU con due caratteri jolly, (ad esempio, OU=*ABC*), è quello successivo più specifico.
   d. Un OU che contiene solo un asterisco (OU=*) è il meno specifico.
3. Se il confronto delle stringhe viene legato tra due valori di attributi della stessa specificità, la stringa dell'attributo più lunga è la più specifica.

4. Se il confronto delle stringhe viene legato tra due valori di attributi della stessa specificità e lunghezza, il risultato viene determinato dal confronto tra due stringhe non sensibili al maiuscolo/minuscolo della parte del DN che non contiene caratteri jolly.

Se due DN sono uguali sotto tutti gli aspetti tranne i loro valori DC, si applicano le stesse regole di corrispondenza degli OU, tranne per il fatto che, nei valori DC, il DC più a sinistra è quello di livello più basso (più specifico) e l'ordine di confronto varia di conseguenza.

Visualizzazione dei record di autenticazione di canale

Per visualizzare i record di autenticazione di canale, utilizzare il comando MQSC **DISPLAY CHLAUTH** o il comando PCF **Inquire Channel Authentication Records**. È possibile scegliere di restituire tutti i record che corrispondono al nome del canale fornito oppure è possibile scegliere una corrispondenza esplicita. La corrispondenza esplicita indica quale record di autenticazione di canale utilizzare nel caso in cui un canale tentasse di effettuare una connessione da un indirizzo IP specifico, da un gestore code specifico oppure tramite ID utente specifico e, facoltativamente, la presentazione di un certificato personale SSL/TLS contenente un DN specificato.

Concetti correlati

“Sicurezza per la messaggistica remota” a pagina 56
Questa sezione tratta gli aspetti della messaggistica remota della sicurezza.

Sicurezza dei messaggi in IBM WebSphere MQ

La sicurezza dei messaggi nell'infrastruttura IBM WebSphere MQ viene fornita da un componente con licenza separata IBM WebSphere MQ Advanced Message Security.

IBM WebSphere MQ Advanced Message Security (AMS) espande i servizi di sicurezza IBM WebSphere MQ per fornire la firma e la cifratura dei dati a livello di messaggio. I servizi espansi garantiscono che i dati del messaggio non siano stati modificati tra il momento in cui sono stati originariamente collocati su una coda e il momento in cui sono stati richiamati. Inoltre, AMS verifica che un mittente dei dati del messaggio sia autorizzato a inserire i messaggi firmati su una coda di destinazione.

Concetti correlati

“IBM WebSphere MQ Advanced Message Security” a pagina 271
IBM WebSphere MQ Advanced Message Security (AMS) è un componente con licenza separata di IBM WebSphere MQ Advanced Message Security che fornisce un livello elevato di protezione per i dati sensibili che passano attraverso la rete IBM WebSphere MQ Advanced Message Security , senza influire sulle applicazioni finali.

Pianificazione dei requisiti di sicurezza

Questa raccolta di argomenti spiega cosa è necessario considerare quando si pianifica la sicurezza in un ambiente IBM WebSphere MQ .

È possibile utilizzare IBM WebSphere MQ per un'ampia gamma di applicazioni su una gamma di piattaforme. I requisiti di sicurezza sono probabilmente diversi per ogni applicazione. Per alcuni, la sicurezza sarà una considerazione critica.

WebSphere MQ fornisce una gamma di servizi di sicurezza a livello di link, incluso il supporto per SSL (Secure Sockets Layer) e TLS (Transport Layer Security).

È necessario considerare alcuni aspetti della sicurezza quando si implementa WebSphere. Sui sistemi UNIX, Linux e Windows , se si ignorano questi aspetti e non si fa nulla, non è possibile utilizzare WebSphere MQ.

Le considerazioni sulla sicurezza sono descritte di seguito.

Autorizzazione per amministrare WebSphere MQ

WebSphere MQ hanno bisogno dell'autorità per:

- Immettere i comandi per amministrare WebSphere MQ

- Utilizzare IBM WebSphere MQ Explorer

Per ulteriori informazioni, vedere:

- “Autorizzazione per la gestione di IBM WebSphere MQ su sistemi UNIX, Linux, and Windows” a pagina 197

Autorizzazione per gestire gli oggetti WebSphere MQ

Le applicazioni possono accedere ai seguenti oggetti WebSphere MQ emettendo chiamate MQI:

- Gestori code
- Code
- Processo padre
- Elenchi nomi
- Argomenti

Le applicazioni possono anche utilizzare i comandi PCF (Programmable Command Format) per accedere a questi oggetti WebSphere MQ e per accedere ai canali e agli oggetti delle informazioni di autenticazione. Questi oggetti possono essere protetti da WebSphere MQ in modo che gli ID utente associati alle applicazioni abbiano l'autorità per accedervi.

Per ulteriori informazioni, vedere “Autorizzazione per le applicazioni ad utilizzare IBM WebSphere MQ” a pagina 50.

Sicurezza canale

Gli ID utente associati agli MCA (message channel agent) hanno bisogno dell'autorità per accedere a varie risorse WebSphere MQ . Ad esempio, un MCA deve essere in grado di connettersi a un gestore code. Se si tratta di un MCA di invio, deve essere in grado di aprire la coda di trasmissione per il canale. Se è un MCA ricevente, deve essere in grado di aprire le code di destinazione. Gli ID utente associati alle applicazioni che devono gestire canali, iniziatori di canali e listener necessitano dell'autorizzazione per utilizzare i comandi PCF pertinenti. Tuttavia, la maggior parte delle applicazioni non ha bisogno di tale accesso.

Per ulteriori informazioni, vedere “Autorizzazione canale” a pagina 69.

Ulteriori considerazioni

È necessario considerare i seguenti aspetti della sicurezza solo se si utilizzano determinate funzioni WebSphere MQ o le estensioni del prodotto di base:

- “Sicurezza per i cluster del gestore code” a pagina 78
- “Sicurezza per la pubblicazione / sottoscrizione IBM WebSphere MQ” a pagina 79
- “Sicurezza per IBM WebSphere MQ Internet pass - thru” a pagina 80

Identificazione e autenticazione della pianificazione

Decidere quali ID utente utilizzare e come e a quali livelli si desidera applicare i controlli di autenticazione.

È necessario decidere come identificare gli utenti delle applicazioni IBM WebSphere MQ , tenendo presente che i diversi sistemi operativi supportano ID utente di lunghezza diversa. È possibile utilizzare i record di autenticazione di canale per eseguire la mappatura da un ID utente ad un altro o per specificare un ID utente in base ad alcuni attributi della connessione. I canali IBM WebSphere MQ che utilizzano SSL o TLS utilizzano i certificati digitali come meccanismo di identificazione e autenticazione. Ogni certificato digitale ha un DN (distinguished name) del soggetto che può essere mappato su specifiche identità utilizzando i record di autenticazione di canale. Inoltre, i certificati CA nel repository delle chiavi determinano quali certificati digitali possono essere utilizzati per l'autenticazione in IBM WebSphere MQ. Per ulteriori informazioni, consultare:

- “Associazione di un gestore code remoto a un ID utente MCAUSER” a pagina 184

- “Associazione di un ID utente asserito dal client ad un ID utente MCAUSER” a pagina 185
- “Associazione di un DN (Distinguished Name) SSL o TLS a un ID utente MCAUSER” a pagina 186
- “Associazione di un indirizzo IP a un ID utente MCAUSER” a pagina 187

Pianificazione dell'autenticazione per un'applicazione client

È possibile applicare i controlli di autenticazione a quattro livelli: a livello di comunicazioni, nelle uscite di sicurezza, con i record di autenticazione di canale e in termini di identificazione passata a un'uscita di sicurezza.

Ci sono quattro livelli di sicurezza da considerare. Il diagramma mostra un client IBM WebSphere MQ MQI connesso a un server. La sicurezza viene applicata a quattro livelli, come descritto nel seguente testo. MCA è un agente del canale dei messaggi.

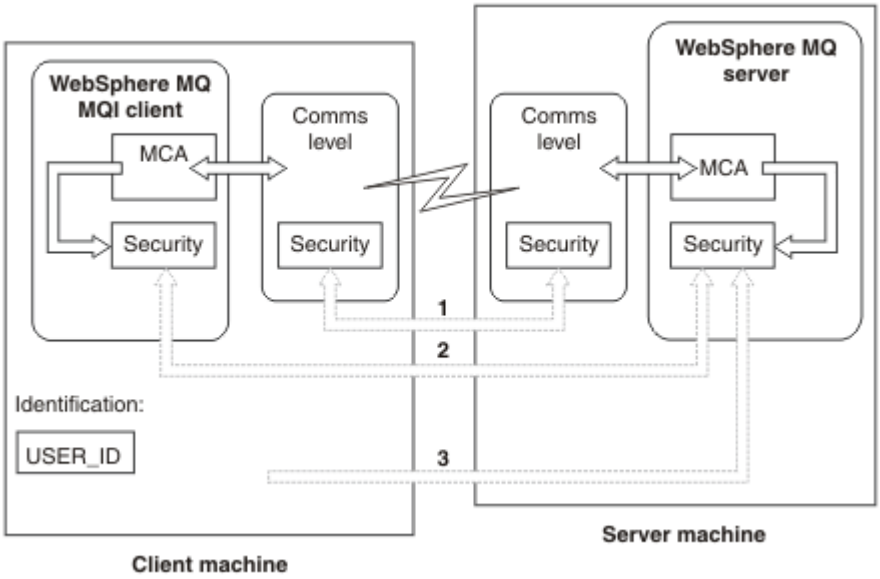


Figura 7. Sicurezza in una connessione client/server

1. Livello di comunicazione

   Vedere la freccia 1. Per implementare la sicurezza a livello delle comunicazioni, utilizzare SSL o TLS. Per ulteriori informazioni, consultare “Protocolli di sicurezza crittografici: SSL e TLS” a pagina 14

2. Record di autenticazione di canale

   Vedere frecce 2 & 3. L'autenticazione può essere controllata utilizzando l'indirizzo IP o i nomi distinti SSL/TLS al livello di sicurezza. Un ID utente può anche essere bloccato o un ID utente asserito può essere associato a un ID utente valido. Una descrizione completa è fornita in “Record di autenticazione di canale” a pagina 39.

3. Uscite di sicurezza del canale

   Vedere la freccia 2. Le uscite di sicurezza del canale per la comunicazione tra client e server possono funzionare allo stesso modo della comunicazione tra server. È possibile scrivere una coppia di uscite indipendenti dal protocollo per fornire l'autenticazione reciproca sia del client che del server. Una descrizione completa viene fornita in Programmi di uscita di sicurezza del canale.

4. Identificazione passata a un'uscita di sicurezza del canale

   Vedere la freccia 3. Nella comunicazione client - server, le uscite di sicurezza del canale non devono operare come una coppia. L'uscita sul client IBM WebSphere MQ può essere omessa. In questo caso, l'ID utente viene inserito nel descrittore del canale (MQCD) e l'uscita di sicurezza lato server può modificarlo, se necessario.

   I client Windows inviano inoltre ulteriori informazioni per assistere l'identificazione.

- L'ID utente passato al server è l'ID utente attualmente collegato sul client.
- L'ID di sicurezza dell'utente attualmente collegato.

Per assistere l'identificazione sul client di IBM WebSphere MQ per HP Integrity NonStop Server, il client passa l'alias di protezione OSS con cui è in esecuzione l'applicazione client. Questo ID è generalmente del formato <PRIMARYGROUP>.<ALIAS>. Se necessario, è possibile associare questo ID utente a un altro ID utente sul gestore code utilizzando i record di autenticazione di canale o un'uscita di sicurezza. Per ulteriori informazioni sulle uscite dei messaggi, consultare “Mappature di identità nelle uscite del messaggio” a pagina 148. Per ulteriori informazioni sulla definizione dei record di autenticazione di canale, consultare “Associazione di un ID utente asserito dal client ad un ID utente MCAUSER” a pagina 185.

I valori dell'ID utente e, se disponibili, dell'ID di sicurezza, possono essere utilizzati dall'uscita di sicurezza del server per stabilire l'identità del client MQI IBM WebSphere MQ .

ID utente

Se il client IBM WebSphere MQ MQI si trova su Windows e il server IBM WebSphere MQ si trova anche su Windows e ha accesso al dominio su cui è definito l'ID utente client, IBM WebSphere MQ supporta ID utente con una lunghezza massima di 20 caratteri. Su piattaforme e configurazioni UNIX and Linux , la lunghezza massima è 12 caratteri.

Un server WebSphere MQ per il Finestre non supporta la connessione di un client Finestre se il client è in esecuzione con un ID utente che contiene il carattere @, ad esempio `abc@d`. Il codice di ritorno alla chiamata `MQCONN` sul client è MQRC_NOT_AUTHORIZED.

Tuttavia, è possibile specificare l'ID utente utilizzando due caratteri @, ad esempio `abc@@d`. L'utilizzo del formato `id@domain` è la pratica preferita, per garantire che l'ID utente sia risolto nel dominio corretto in modo congruente; quindi `abc@@d@domain`.

Si noti che `UNKNOWN` è un ID utente riservato e l'ID utente `NOBODY` ha anche significati speciali per WebSphere MQ. La creazione di ID utente nel sistema operativo denominato `UNKNOWN` o `NOBODY` potrebbe avere risultati non previsti.

Anche se gli ID utente vengono utilizzati per l'autenticazione, i gruppi vengono utilizzati per l'autorizzazione, ad eccezione di Windows.

Se si creano account di servizio, senza prestare attenzione ai gruppi e si autorizzano in modo diverso tutti gli ID utente, ogni utente può accedere alle informazioni di ogni altro utente.

Autorizzazione di pianificazione

Pianificare gli utenti che disporranno dell'autorizzazione di gestione e pianificare come autorizzare gli utenti delle applicazioni ad utilizzare in modo appropriato gli oggetti IBM WebSphere MQ , inclusi quelli che si collegano da un client IBM WebSphere MQ MQI.

Ai singoli utenti o alle applicazioni deve essere concesso l'accesso per utilizzare IBM WebSphere MQ. L'accesso di cui hanno bisogno dipende dai ruoli che svolgono e dalle attività che devono svolgere. L'autorizzazione in IBM WebSphere MQ può essere suddivisa in due categorie principali:

- Autorizzazione ad eseguire operazioni amministrative
- Autorizzazione per le applicazioni ad utilizzare IBM WebSphere MQ

Entrambe le classi di operazioni sono controllate dallo stesso componente e ad un individuo può essere concessa l'autorità di eseguire entrambe le categorie di operazioni.

I seguenti argomenti forniscono ulteriori informazioni su specifiche aree di autorizzazione che è necessario considerare:

Autorizzazione per amministrare IBM WebSphere MQ

Gli amministratori IBM WebSphere MQ hanno bisogno dell'autorità per eseguire varie funzioni. Questa autorizzazione viene ottenuta in modi diversi su piattaforme diverse.

Gli amministratori IBM WebSphere MQ hanno bisogno dell'autorità per:

- Immettere i comandi per amministrare IBM WebSphere MQ
- UtilizzareIBM WebSphere MQ Explorer

Per ulteriori informazioni, consultare l'argomento appropriato per il proprio sistema operativo.

Autorizzazione per la gestione di IBM WebSphere MQ su sistemi UNIX e Windows

Un amministratore IBM WebSphere MQ è un membro del gruppo mqm. Questo gruppo ha accesso a tutte le risorse IBM WebSphere MQ e può immettere comandi di controllo IBM WebSphere MQ . Un amministratore può concedere autorizzazioni specifiche ad altri utenti.

Per essere un amministratore di IBM WebSphere MQ su sistemi UNIX e Windows , un utente deve essere membro del *gruppo mq*. Questo gruppo viene creato automaticamente quando si installa WebSphere MQ. Per consentire agli utenti di immettere comandi di controllo, è necessario aggiungerli al gruppo mqm. Ciò include l'utente root su sistemi UNIX .

Agli utenti che non sono membri del gruppo mqm possono essere concessi privilegi di gestione, ma non sono in grado di immettere comandi di controllo IBM WebSphere MQ e sono autorizzati ad eseguire solo i comandi per i quali è stato concesso l'accesso.

Inoltre, sui sistemi Windows , gli account SYSTEM e Administrator hanno accesso completo alle risorse IBM WebSphere MQ .

Tutti i membri del gruppo mqm hanno accesso a tutte le risorse WebSphere MQ sul sistema, inclusa la possibilità di gestire qualsiasi gestore code in esecuzione sul sistema. Questo accesso può essere revocato solo rimuovendo un utente dal gruppo mqm. Su sistemi Windows , i membri del gruppo Administrators hanno anche accesso a tutte le risorse WebSphere MQ .

Gli amministratori possono utilizzare il comando di controllo **runmqsc** per immettere i comandi WebSphere MQ Script (MQSC). Quando **runmqsc** viene utilizzato in modalità indiretta per inviare comandi MQSC a un gestore code remoto, ogni comando MQSC viene incapsulato all'interno di un comando PCF Escape. Gli amministratori devono disporre delle autorizzazioni richieste per i comandi MQSC che devono essere elaborati dal gestore code remoto.

WebSphere MQ Explorer emette comandi PCF per eseguire attività di amministrazione. Gli amministratori non richiedono ulteriori autorizzazioni per utilizzare WebSphere MQ Explorer per gestire un gestore code sul sistema locale. Quando WebSphere MQ Explorer viene utilizzato per gestire un gestore code su un altro sistema, gli amministratori devono disporre delle autorizzazioni richieste affinché i comandi PCF vengano elaborati dal gestore code remoto.

Per ulteriori informazioni sui controlli delle autorizzazioni eseguiti quando vengono elaborati i comandi PCF e MQSC, consultare i seguenti argomenti:

- Per i comandi che operano su gestori code, code, canali, processi, elenchi nomi e oggetti delle informazioni di autenticazione, consultare “Autorizzazione per le applicazioni ad utilizzare IBM WebSphere MQ” a pagina 50.
- Per i comandi che operano su canali, iniziatori di canali, listener e cluster, consultare Sicurezza canale.

Per ulteriori informazioni relative all'autorizzazione necessaria per amministrare WebSphere MQ su sistemi UNIX e Windows , consultare le informazioni correlate.

Autorizzazione per le applicazioni ad utilizzare IBM WebSphere MQ

Quando le applicazioni accedono agli oggetti, gli ID utente associati alle applicazioni richiedono l'autorità appropriata.

Le applicazioni possono accedere ai seguenti oggetti IBM WebSphere MQ emettendo chiamate MQI:

- Gestori code
- Code
- Processo padre
- Elenchi nomi

- Argomenti

Le applicazioni possono anche utilizzare comandi PCF per gestire oggetti IBM WebSphere MQ . Quando il comando PCF viene elaborato, utilizza il contesto di autorizzazione dell'ID utente che inserisce il messaggio PCF.

Le applicazioni, in questo contesto, includono quelle scritte da utenti e fornitori.

Le applicazioni che utilizzano le classi IBM WebSphere MQ per Java, le classi IBM WebSphere MQ per JMS, le classi IBM WebSphere MQ per .NET o i Message Service Clients for C/C++ and .NET utilizzano MQI indirettamente.

Gli MCA emettono anche chiamate MQI e gli ID utente associati agli MCA hanno bisogno dell'autorità per accedere a questi oggetti WebSphere MQ . Per ulteriori informazioni su questi ID utente e sulle autorizzazioni richieste, consultare “Autorizzazione canale” a pagina 69.

Quando vengono eseguiti i controlli di autorizzazione

I controlli delle autorizzazioni vengono eseguiti quando un'applicazione tenta di accedere a un gestore code, a una coda, a un processo o a un elenco nomi.

I controlli vengono eseguiti nelle seguenti circostanze:

Quando un'applicazione si connette a un gestore code utilizzando una chiamata `MQCONN` o `MQCONNX`

Il gestore code richiede al sistema operativo l'ID utente associato all'applicazione. Il gestore code verifica quindi che l'ID utente sia autorizzato a connettersi ad esso e conserva l'ID utente per i futuri controlli.

Gli utenti non devono accedere a IBM WebSphere MQ. IBM WebSphere MQ presuppone che gli utenti siano collegati al sistema operativo sottostante e che siano stati autenticati da esso.

Quando un'applicazione apre un oggetto IBM WebSphere MQ utilizzando una chiamata `MQOPEN` o `MQPUT1`

Tutti i controlli di autorizzazione vengono eseguiti quando un oggetto viene aperto, non quando vi si accede in un secondo momento. Ad esempio, i controlli di autorizzazione vengono eseguiti quando un'applicazione apre una coda. Non vengono eseguiti quando l'applicazione inserisce i messaggi nella coda o riceve i messaggi dalla coda.

Quando un'applicazione apre un oggetto, specifica i tipi di operazione da eseguire sull'oggetto. Ad esempio, un'applicazione potrebbe aprire una coda per sfogliare i messaggi su di essa, ottenere i messaggi da essa, ma non per inserire i messaggi su di essa. Per ciascun tipo di operazione, il gestore code controlla che l'ID utente associato all'applicazione disponga delle autorizzazioni per eseguire tale operazione.

Quando un'applicazione apre una coda, i controlli di autorizzazione vengono eseguiti sull'oggetto denominato nel campo `ObjectName` del descrittore dell'oggetto. Il campo `ObjectName` viene utilizzato nelle chiamate `MQOPEN` o `MQPUT1` . Se l'oggetto è una coda alias o una definizione di coda remota, i controlli di autorizzazione vengono eseguiti sull'oggetto stesso. Non vengono eseguiti sulla coda in cui si risolve la coda alias o la definizione della coda remota. Ciò significa che l'utente non ha bisogno dell'autorizzazione per accedervi. Limitare l'autorizzazione a creare code per utenti privilegiati. In caso contrario, gli utenti potrebbero ignorare il normale controllo degli accessi semplicemente creando un alias.

Un'applicazione può fare riferimento esplicitamente a una coda remota. Imposta i campi `ObjectName` e `ObjectQMgrName` nel descrizione oggetto sui nomi della coda remota e del gestore code remoto. I controlli delle autorizzazioni vengono eseguiti sulla coda di trasmissione con lo stesso nome del gestore code remoto. Su UNIX, Linux, and Windows, viene eseguito un controllo sul profilo RQMNAME che corrisponde al nome del gestore code remoto, se si sta utilizzando il cluster. Un'applicazione può fare riferimento esplicitamente a una coda cluster impostando il campo `ObjectName` nel descrittore oggetto sul nome della coda cluster. I controlli dell'autorità vengono effettuati sulla coda di trasmissione del cluster, `SYSTEM.CLUSTER.TRANSMIT.QUEUE`.

L'autorizzazione ad una coda dinamica si basa sulla coda modello da cui deriva, ma non è necessariamente la stessa; consultare la nota 1.

L'ID utente che il gestore code utilizza per i controlli di autorizzazione viene ottenuto dal sistema operativo. L'ID utente viene ottenuto quando l'applicazione si connette al gestore code. Un'applicazione adeguatamente autorizzata può emettere una chiamata MQOPEN specificando un ID utente alternativo; le verifiche del controllo accessi vengono quindi effettuate sull'ID utente alternativo. L'uso di un ID utente alternativo non modifica l'ID utente associato all'applicazione, ma solo quello utilizzato per le verifiche del controllo accessi.

Quando un'applicazione effettua la sottoscrizione a un argomento utilizzando una chiamata MQSUB
Quando un'applicazione sottoscrive un argomento, specifica il tipo di operazioni che deve eseguire. Si tratta di creare una sottoscrizione, modificare una sottoscrizione esistente o riprendere una sottoscrizione esistente senza modificarla. Per ogni tipo di operazione, il gestore code verifica che l'ID utente associato all'applicazione disponga dell'autorizzazione per eseguire l'operazione.

Quando un'applicazione effettua la sottoscrizione a un argomento, i controlli di autorizzazione vengono effettuati sugli oggetti argomento trovati nella struttura ad albero dell'argomento. Gli oggetti argomento si trovano nel punto, o al di sopra, nella struttura ad albero degli argomenti in cui l'applicazione ha effettuato la sottoscrizione. I controlli di autorizzazione potrebbero comportare controlli su più di un oggetto argomento. L'ID utente che il gestore code utilizza per i controlli di autorizzazione viene ottenuto dal sistema operativo. L'ID utente viene ottenuto quando l'applicazione si connette al gestore code.

Il gestore code esegue controlli di autorizzazione sulle code del sottoscrittore ma non sulle code gestite.

Quando un'applicazione elimina una coda dinamica permanente utilizzando una chiamata MQCLOSE
L'handle dell'oggetto specificato nella chiamata MQCLOSE non è necessariamente lo stesso restituito dalla chiamata MQOPEN che ha creato la coda dinamica permanente. Se è differente, il gestore code controlla l'ID utente associato all'applicazione che ha emesso la chiamata MQCLOSE . Verifica che l'ID utente sia autorizzato ad eliminare la coda.

Quando un'applicazione che chiude una sottoscrizione per rimuoverla non l'ha creata, è necessaria l'autorizzazione appropriata per rimuoverla.

Quando un comando PCF che opera su un oggetto WebSphere MQ viene elaborato dal server dei comandi
Questa regola include il caso in cui un comando PCF opera su un oggetto delle informazioni di autenticazione.

L'ID utente utilizzato per i controlli di autorizzazione è quello trovato nel campo UserIdentifier nel descrittore del messaggio del comando PCF. Questo ID utente deve disporre delle autorizzazioni richieste sul gestore code su cui viene elaborato il comando. Il comando MQSC equivalente incapsulato all'interno di un comando Escape PCF viene trattato nello stesso modo. Per ulteriori informazioni sul campo UserIdentifier e su come è impostato, consultare “Contesto messaggio” a pagina 52.

Autorizzazione utente alternativo

Quando un'applicazione apre un oggetto o sottoscrive un argomento, può fornire un ID utente sulla chiamata MQOPEN, MQPUT1o MQSUB. Può richiedere al gestore code di utilizzare questo ID utente per i controlli di autorizzazione invece di quello associato all'applicazione.

L'applicazione riesce ad aprire l'oggetto solo se sono soddisfatte entrambe le condizioni seguenti:

- L'ID utente associato all'applicazione ha l'autorità di fornire un diverso ID utente per i controlli dell'autorizzazione. Si dice che l'applicazione abbia l' *autorizzazione utente alternativa*.
- L'ID utente fornito dall'applicazione dispone dell'autorità per aprire l'oggetto per i tipi di operazione richiesti o per sottoscrivere l'argomento.

Contesto messaggio

Le informazioni sul *Contesto del messaggio* consentono all'applicazione che richiama un messaggio di individuare il mittente del messaggio. Le informazioni sono contenute nei campi nel descrittore del messaggio e i campi sono divisi in tre parti logiche

Queste parti sono le seguenti:

contesto di identità
Questi campi contengono informazioni sull'utente dell'applicazione che inserisce il messaggio nella coda.

contesto di origine
Questi campi contengono le informazioni sull'applicazione stessa e quando il messaggio è stato inserito nella coda.

contesto utente
Questi campi contengono proprietà del messaggio che le applicazioni possono utilizzare per selezionare i messaggi che il gestore code deve consegnare.

Quando un'applicazione inserisce un messaggio in una coda, può chiedere al gestore code di creare le informazioni di contesto nel messaggio. Questa è l'azione predefinita. In alternativa, può specificare che i campi di contesto non devono contenere informazioni. L'ID utente associato ad un'applicazione non richiede alcuna autorizzazione speciale per eseguire una di queste operazioni.

Un'applicazione può impostare i campi di contesto di identità in un messaggio, consentendo al gestore code di generare il contesto di origine oppure può impostare tutti i campi di contesto. Un'applicazione può anche passare i campi di contesto di identità da un messaggio che ha richiamato a un messaggio che sta inserendo in una coda oppure può passare tutti i campi di contesto. Tuttavia, l'ID utente associato a un'applicazione richiede l'autorizzazione per impostare o trasmettere le informazioni di contesto. Un'applicazione specifica che intende impostare o trasmettere le informazioni di contesto quando apre la coda su cui sta per inserire i messaggi e la relativa autorizzazione viene controllata in questo momento.

Di seguito viene riportata una breve descrizione di ciascuno dei campi di contesto:

contesto di identità

UserIdentifier
L'ID utente associato all'applicazione che ha inserito il messaggio. Se il gestore code imposta questo campo, esso viene impostato sull'ID utente ottenuto dal sistema operativo quando l'applicazione si connette al gestore code.

AccountingToken
Informazioni che possono essere utilizzate per addebitare il lavoro eseguito come risultato del messaggio.

ApplIdentityData
Se l'ID utente associato a un'applicazione dispone dell'autorità per impostare i campi del contesto di identità o per impostare tutti i campi del contesto, l'applicazione può impostare questo campo su qualsiasi valore correlato all'identità. Se il gestore code imposta questo campo, viene impostato su vuoto.

Contesto di origine

PutApplType
Il tipo di applicazione che ha inserito il messaggio; ad esempio, una transazione CICS .

PutApplName
Il nome dell'applicazione che ha inserito il messaggio.

PutDate
La data in cui è stato inserito il messaggio.

PutTime
L'ora in cui è stato inserito il messaggio.

ApplOriginData
Se l'ID utente associato a un'applicazione dispone dell'autorizzazione per impostare tutti i campi di contesto, l'applicazione può impostare questo campo su qualsiasi valore correlato all'origine. Se il gestore code imposta questo campo, viene impostato su vuoto.

Contesto utente

I seguenti valori sono supportati per **MQINQMP** o **MQSETMP**:

Contesto_UTENTE MQPD_

La proprietà è associata al contesto utente.

Non è richiesta alcuna autorizzazione speciale per poter impostare una proprietà associata al contesto utente utilizzando la chiamata MQSETMP.

Su un V7.0 o su un gestore code successivo, una proprietà associata al contesto utente viene salvata come descritto per MQOO_SAVE_ALL_CONTEXT. Un MQPUT con MQOO_PASS_ALL_CONTEXT specificato fa sì che la proprietà venga copiata dal contesto salvato nel nuovo messaggio.

MQPD_NO_CONTEXT

La proprietà non è associata a un contesto di messaggio.

Un valore non riconosciuto è stato rifiutato con MQRC_PD_ERROR. Il valore iniziale di questo campo è **MQPD_NO_CONTEXT**.

Per una descrizione dettagliata di ciascun campo di contesto, vedere MQMD - Descrittore messaggio. Per ulteriori informazioni su come utilizzare il contesto del messaggio, consultare Contesto del messaggio.

Autorizzazione per gestire gli oggetti IBM WebSphere MQ su sistemi UNIX, Linux e Windows

Il componente del servizio di autorizzazione fornito con IBM WebSphere MQ è denominato *OAM (object authority manager)*. Fornisce il controllo degli accessi tramite l'autenticazione e i controlli di autorizzazione.

1. Autenticazione.

   Il controllo di autenticazione eseguito da OAM fornito con IBM WebSphere MQ è di base e viene eseguito solo in circostanze specifiche. Non ha lo scopo di soddisfare i severi requisiti previsti in un ambiente altamente sicuro.

   OAM esegue il controllo di autenticazione quando un'applicazione si connette a un gestore code e le seguenti condizioni sono vere.

   Se una struttura MQCSP è stata fornita dall'applicazione di connessione e l'attributo *AuthenticationType* nella struttura MQCSP viene fornito con il valore MQCSP_AUTH_USER_ID_AND_PWD, il controllo viene eseguito da OAM nella funzione MQZID_AUTHENTICATE_USER. Di seguito viene riportato il controllo: l'ID utente nella struttura MQCSP viene confrontato con l'ID utente in *IdentityContext* (MQZIC), per determinare se corrispondono. Se non corrispondono, il controllo ha esito negativo.

   Questo controllo di base non deve essere un'autenticazione completa dell'utente. Ad esempio, non viene eseguita alcuna verifica dell'autenticità dell'utente controllando la password fornita nella struttura MQCSP. Inoltre, se l'applicazione omette una struttura MQCSP, non viene eseguito alcun controllo.

   Se sono richiesti servizi di autenticazione più completi nel gestore code tramite il componente del servizio di autorizzazione, l'OAM fornito con IBM WebSphere MQ non lo offre. È necessario scrivere un nuovo componente del servizio di autorizzazione oppure ottenerne uno da un fornitore.

2. Autorizzazione.

   I controlli di autorizzazione sono completi e mirano a soddisfare la maggior parte dei requisiti normali.

   I controlli di autorizzazione vengono eseguiti quando un'applicazione emette una chiamata MQI per accedere a un gestore code, coda, processo, argomento o elenco nomi. Vengono eseguite anche in altri momenti, ad esempio, quando un comando viene eseguito dal Server dei comandi.

Su sistemi UNIX, Linux e Windows , il *servizio di autorizzazione* fornisce il controllo dell'accesso quando un'applicazione emette una chiamata MQI per accedere a un oggetto IBM WebSphere MQ che è un gestore code, una coda, un processo, un argomento o un elenco nomi. Ciò include i controlli per l'autorizzazione utente alternativa e l'autorizzazione per impostare o trasmettere le informazioni di contesto.

Su Windows, OAM fornisce ai membri del gruppo Amministratori l'autorizzazione ad accedere a tutti gli oggetti IBM WebSphere MQ , anche quando UAC è abilitato.

Inoltre, sui sistemi Windows , l'account SYSTEM ha accesso completo alle risorse IBM WebSphere MQ .

Il servizio di autorizzazione fornisce anche i controlli di autorizzazione quando un comando PCF opera su uno di tali oggetti IBM WebSphere MQ o su un oggetto delle informazioni di autenticazione. Il comando MQSC equivalente incapsulato all'interno di un comando Escape PCF viene trattato nello stesso modo.

Il servizio di autorizzazione è un *servizio installabile*, il che significa che è implementato da uno o più *componenti del servizio installabili*. Ogni componente viene richiamato utilizzando un'interfaccia documentata. Ciò consente agli utenti e ai fornitori di fornire componenti per aumentare o sostituire quelli forniti dai prodotti IBM WebSphere MQ .

Il componente del servizio di autorizzazione fornito con IBM WebSphere MQ è denominato *OAM (object authority manager)*. OAM viene abilitato automaticamente per ogni gestore code creato.

OAM gestisce un ACL (access control list) per ogni oggetto IBM WebSphere MQ per cui controlla l'accesso. Su sistemi UNIX and Linux , solo gli ID gruppo possono essere visualizzati in un ACL. Ciò significa che tutti i membri di un gruppo hanno le stesse autorizzazioni. Su sistemi Windows , sia gli ID utente che gli ID gruppo possono essere visualizzati in un ACL. Ciò significa che le autorizzazioni possono essere concesse a singoli utenti e gruppi.

Una limitazione di 12 caratteri si applica sia al gruppo che all'ID utente. Le piattaforme UNIX generalmente limitano la lunghezza di un ID utente a 12 caratteri. AIX e Linux hanno aumentato questo limite, ma IBM WebSphere MQ continua a osservare una limitazione di 12 caratteri su tutte le piattaforme UNIX . Se si utilizza un ID utente superiore a 12 caratteri, IBM WebSphere MQ lo sostituisce con il valore "UNKNOWN". Non definire un ID utente con valore "SCONOSCIUTO".

OAM può autenticare un utente e modificare i campi del contesto di identità appropriati. È possibile abilitarlo specificando una struttura di parametri di sicurezza della connessione (MQCSP) su una chiamata MQCONNX. La struttura viene trasmessa alla funzione OAM Authenticate User (MQZ_AUTHENTICATE_USER), che imposta i campi del contesto di identità appropriati. Se una connessione MQCONNX da un client IBM WebSphere MQ , le informazioni in MQCSP vengono trasmesse al gestore code a cui il client si connette tramite il canale di connessione client e server. Se le uscite di sicurezza sono definite su tale canale, MQCSP viene passato in ciascuna uscita di sicurezza e può essere modificato dall'uscita. Le uscite di sicurezza possono anche creare MQCSP. Per ulteriori dettagli sull'utilizzo delle uscite di sicurezza in questo contesto, consultare Programmi di uscita di sicurezza del canale.

Su sistemi UNIX, Linux e Windows , il comando di controllo **setmqaut** concede e revoca le autorizzazioni e viene utilizzato per gestire gli ACL. Ad esempio, il comando:

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

consente ai membri del gruppo VOYAGER di esaminare i messaggi sulla coda MOON.EUROPA di proprietà del gestore code JUPITER. Consente ai membri di richiamare anche i messaggi dalla coda. Per revocare tali autorizzazioni successivamente, immettere il seguente comando:

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

Il comando:

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

consente ai membri del gruppo VOYAGER di inserire messaggi in qualsiasi coda con un nome che inizia con i caratteri `MOON.` . MOON.* è il nome di un profilo generico.Un *profilo generico* consente di concedere le autorizzazioni per una serie di oggetti utilizzando un singolo comando **setmqaut** .

Il comando di controllo **dspmqaut** è disponibile per visualizzare le autorizzazioni correnti di un utente o di un gruppo per un oggetto specificato.Il comando di controllo **dmpmqaut** è disponibile anche per visualizzare le autorizzazioni correnti associate ai profili generici.

Se non si desidera alcun controllo dell'autorizzazione, ad esempio, in un ambiente di test, è possibile disabilitare OAM.

Utilizzo di PCF per accedere ai comandi OAM
Su UNIX, Linux e Finestre , è possibile utilizzare i comandi PCF per accedere ai comandi di gestione OAM.

I comandi PCF e i relativi comandi OAM equivalenti sono i seguenti:

Tabella 6. Comandi PCF e relativi comandi OAM equivalenti

Comando PCF	**comando OAM**
Interrogazione record autorizzazione	dmpmqaut
Interroga autorità entità	Dspmqaut
Imposta record di autorizzazione	setmqaut
Eliminare il record di autorizzazione	setmqaut con l'opzione -remove

I comandi **setmqaut** e **dmpmqaut** sono limitati ai membri del gruppo mqm. I comandi PCF equivalenti possono essere eseguiti dagli utenti in qualsiasi gruppo a cui sono state concesse le autorizzazioni dsp e chg sul gestore code.

Per ulteriori informazioni sull'utilizzo di questi comandi, consultare Introduzione a Programmable Command Formats .

Sicurezza per la messaggistica remota

Questa sezione tratta gli aspetti della messaggistica remota della sicurezza.

È necessario fornire agli utenti l'autorità per utilizzare le funzioni IBM WebSphere MQ . Questo è organizzato in base alle azioni da intraprendere rispetto agli oggetti e alle definizioni. Ad esempio:

- I gestori code possono essere avviati e arrestati dagli utenti autorizzati
- Le applicazioni devono connettersi al gestore code e disporre dell'autorità per utilizzare le code
- I canali di messaggi devono essere creati e controllati da utenti autorizzati
- Gli oggetti vengono conservati nelle librerie e l'accesso a tali librerie può essere limitato

L'agente del canale dei messaggi in un sito remoto deve controllare che il messaggio che si sta consegnando sia stato originato da un utente con l'autorizzazione a farlo in questo sito remoto. Inoltre, poiché gli MCA possono essere avviati in remoto, potrebbe essere necessario verificare che i processi remoti che tentano di avviare gli MCA siano autorizzati a farlo. Ci sono quattro modi possibili per affrontare questo:

1. Utilizzare in modo appropriato l'attributo PutAuthority della propria definizione di canale RCVR, RQSTR o CLUSRCVR per controllare quale utente viene utilizzato per i controlli di autorizzazione nel momento in cui i messaggi in entrata vengono inseriti nelle code. Consultare la descrizione del comando DEFINE CHANNEL nel manuale MQSC Command Reference.
2. Implementare i record di autenticazione di canale per rifiutare i tentativi di connessione non desiderati o per impostare un valore MCAUSER in base a quanto segue: l'indirizzo IP remoto, l'ID utente remoto, il DN (Subject Distinguished Name) SSL o TLS fornito o il nome del gestore code remoto.
3. Implementa il controllo di protezione *user exit* per garantire che il corrispondente canale di messaggi sia autorizzato. La sicurezza dell'installazione che ospita il canale corrispondente garantisce che tutti gli utenti siano correttamente autorizzati, in modo che non sia necessario controllare i singoli messaggi.
4. Implementare l'elaborazione dei messaggi *user exit* per garantire che i singoli messaggi vengano controllati per l'autorizzazione.

Sicurezza degli oggetti sui sistemi UNIX and Linux

Gli utenti di gestione devono far parte del gruppo mqm sul sistema (incluso root) se questo ID utilizzerà i comandi di gestione IBM WebSphere MQ .

Si consiglia di eseguire sempre amqcrsta come ID utente "mqm".

ID utente su sistemi UNIX and Linux

Il gestore code converte tutti gli identificativi utente in maiuscolo o in minuscolo. Il gestore code inserisce quindi gli identificatori utente nella parte di contesto di un messaggio o ne controlla l'autorizzazione. Le autorizzazioni sono pertanto basate solo su identificativi in minuscolo.

Sicurezza degli oggetti sui sistemi Windows

Gli utenti di gestione devono far parte sia del gruppo mqm che del gruppo degli amministratori sui sistemi Windows se questo ID utilizzerà i comandi di amministrazione IBM WebSphere MQ .

ID utente su sistemi Windows

Sui sistemi Windows , *se non è installata alcuna uscita del messaggio*, il gestore code converte gli identificativi utente maiuscoli o maiuscoli e minuscoli. Il gestore code inserisce quindi gli identificatori utente nella parte di contesto di un messaggio o ne controlla l'autorizzazione. Le autorizzazioni sono pertanto basate solo su identificativi in minuscolo.

ID utente nei sistemi

Piattaforme diverse da Windows , i sistemi UNIX and Linux utilizzano caratteri maiuscoli per gli ID utente nei messaggi.

Per consentire ai sistemi Windows, UNIX and Linux di utilizzare ID utente in lettere minuscole nei messaggi, l'MCA (message channel agent) effettua le seguenti conversioni su queste piattaforme:

Alla fine dell'invio
I caratteri alfabetici in tutti gli ID utente vengono convertiti in caratteri maiuscoli, se non è installata alcuna uscita messaggio.

All'estremità ricevente
I caratteri alfabetici in tutti gli ID utente vengono convertiti in caratteri minuscoli, se non è installata alcuna uscita messaggio.

Le conversioni automatiche non vengono eseguite se si fornisce un'uscita del messaggio su sistemi UNIX, Linux e Windows per qualsiasi altro motivo.

Utilizzo di un servizio di autorizzazione personalizzato

IBM WebSphere MQ fornisce un servizio di autorizzazione installabile. È possibile scegliere di installare un servizio alternativo.

Il componente del servizio di autorizzazione fornito con IBM WebSphere MQ è denominato OAM (Object Authority Manager). Se l'OAM non fornisce le funzioni di autorizzazione necessarie, è possibile scrivere il proprio componente del servizio di autorizzazione. Le funzioni di servizio installabili che devono essere implementate da un componente del servizio di autorizzazione sono descritte in Informazioni di riferimento per l'interfaccia dei servizi installabili.

Controllo accessi per client

Il controllo accessi è basato sugli ID utente. Ci possono essere molti ID utente da gestire e gli ID utente possono essere in formati differenti. È possibile impostare la proprietà MCAUSER del canale di connessione server su un valore ID utente speciale per l'utilizzo da parte dei client.

Il controllo accessi in IBM WebSphere MQ è basato sugli ID utente. L'ID utente del processo che effettua le chiamate MQI viene normalmente utilizzato. Per i client MQI MQ , la connessione server MCA effettua chiamate MQI per conto dei client MQI MQ . È possibile selezionare un ID utente alternativo per l'MCA di connessione server da utilizzare per effettuare chiamate MQI. L'ID utente alternativo può essere

associato alla stazione di lavoro del client o a qualsiasi cosa si scelga di organizzare e controllare l'accesso dei client. L'ID utente deve disporre delle autorizzazioni necessarie ad esso assegnate sul server per emettere chiamate MQI. La scelta di un ID utente alternativo è preferibile a consentire ai client di effettuare chiamate MQI con l'autorizzazione dell'MCA di connessione server.

Tabella 7. L'ID utente utilizzato da un canale di connessione server

ID utente	Quando utilizzato
L'ID utente impostato da un'uscita di sicurezza	Utilizzato a meno che non sia bloccato da una regola **CHLAUTH TYPE(BLOCKUSER)** . Per ulteriori informazioni, vedere la seguente sezione, “Impostazione dell'ID utente in un'uscita di sicurezza” a pagina 58 .
L'ID utente impostato da una regola CHLAUTH	Utilizzato a meno che non sovrascritto da un'uscita di sicurezza. Per ulteriori informazioni, consultare Record di autenticazione di canale .
L'ID definito nell'attributo **MCAUSER** nella definizione del canale SVRCONN	Utilizzato a meno che non sovrascritto da un'uscita di sicurezza o da una regola CHLAUTH.
L'ID utente che viene fornito dalla macchina client	Utilizzato quando nessun ID utilizzato è impostato da altri mezzi.
L'ID utente che ha avviato il canale di connessione server	Utilizzato quando nessun ID utente è impostato da qualsiasi altro mezzo e nessun ID utente client è in flusso. Per ulteriori informazioni, vedere la seguente sezione, “L'ID utente che esegue il programma del canale” a pagina 59 .

Poiché la connessione server MCA effettua chiamate MQI per conto di utenti remoti, è importante considerare le implicazioni di sicurezza della connessione server MCA che emette chiamate MQI per conto di client remoti e come amministrare l'accesso di un numero potenzialmente elevato di utenti.

- Un approccio è che l'MCA di connessione server emetti chiamate MQI con la propria autorizzazione. Ma attenzione, è normalmente indesiderabile per il server - connessione MCA, con le sue potenti capacità di accesso, di emettere chiamate MQI per conto di utenti client.
- Un altro approccio consiste nell'utilizzare l'ID utente che proviene dal client. L'MCA di connessione server può emettere chiamate MQI utilizzando le funzioni di accesso dell'ID utente client. Questo approccio presenta una serie di domande da considerare:
  1. Esistono diversi formati per l'ID utente su diverse piattaforme. Ciò a volte causa problemi se il formato dell'ID utente sul client differisce dai formati accettabili sul server.
  2. Esistono potenzialmente molti client, con ID utente diversi e in fase di modifica. Gli ID devono essere definiti e gestiti sul server.
  3. L'ID utente è attendibile? Qualsiasi ID utente può essere fornito da un client, non necessariamente l'ID dell'utente collegato. Ad esempio, il client potrebbe far fluire un ID con autorizzazione `mqm` completa che è stata intenzionalmente definita sul server solo per ragioni di sicurezza.
- L'approccio preferito consiste nel definire i token di identificazione del client sul server e quindi limitare le funzioni delle applicazioni connesse al client. Questa operazione viene di solito eseguita impostando la proprietà del canale di connessione server MCAUSER su un valore ID utente speciale che deve essere utilizzato dai client e definendo pochi ID per l'utilizzo da parte dei client con un diverso livello di autorizzazione sul server.

Impostazione dell'ID utente in un'uscita di sicurezza

Per i client IBM WebSphere MQ MQI, il processo che emette le chiamate MQI è l'MCA di connessione server. L'ID utente utilizzato dall'MCA di connessione server è contenuto nei campi `MCAUserIdentifier` o `LongMCAUserIdentifier` di MQCD. Il contenuto di questi campi è impostato da:

- Qualsiasi valore impostato dalle uscite di sicurezza
- L'ID utente dal client
- MCAUSER (nella definizione del canale di connessione server)

L'uscita di sicurezza può sovrascrivere i valori che sono visibili ad essa, quando viene richiamata.

- Se l'attributo MCAUSER del canale di connessione server è impostato su non vuoto, viene utilizzato il valore MCAUSER.
- Se l'attributo MCAUSER del canale di connessione server è vuoto, viene utilizzato l'ID utente ricevuto dal client.
- Se l'attributo MCAUSER del canale di connessione server è vuoto e non si riceve alcun ID utente dal client, viene utilizzato l'ID utente che ha avviato il canale di connessione server.

Verificare che il campo MCAUSER sia limitato a 12 caratteri sulle piattaforme Windows poiché i caratteri supplementari verranno troncati, il che potrebbe causare errori di autorizzazione.

Il client IBM WebSphere MQ non esegue il flusso dell'ID utente asserito al server quando è in uso un'uscita di sicurezza lato client.

L'ID utente che esegue il programma del canale

Quando i campi ID utente derivano dall'ID utente che ha avviato il canale di connessione server, viene utilizzato il valore seguente:

- Per z/OS, l'ID utente assegnato all'attività avviata dell'iniziatore di canali dalla tabella delle procedure avviate z/OS .
- Per TCP/IP (nonz/OS), l'ID utente dalla voce `inetd.conf` o l'ID utente che ha avviato il listener.
- Per SNA (nonz/OS), l'ID utente dalla voce SNA Server o (se non esiste) la richiesta di collegamento in ingresso o l'ID utente che ha avviato il listener.
- Per NetBIOS o SPX, l'ID utente che ha avviato il listener.

Se esistono definizioni di canale di connessione server che hanno l'attributo MCAUSER impostato su vuoto, i client possono utilizzare questa definizione di canale per connettersi al gestore code con autorizzazione di accesso determinata dall'ID utente fornito dal client. Ciò potrebbe rappresentare un rischio per la sicurezza se il sistema su cui è in esecuzione il gestore code consente connessioni di rete non autorizzate. Il canale di connessione server predefinito IBM WebSphere MQ (SYSTEM.DEF.SVRCONN) ha l'attributo MCAUSER impostato su vuoto. Per impedire l'accesso non autorizzato, aggiornare l'attributo MCAUSER della definizione predefinita con un ID utente che non abbia accesso agli oggetti IBM WebSphere MQ MQ .

Caso di ID utente

Quando si definisce un canale con `runmqsc`, l'attributo MCAUSER viene modificato in maiuscolo a meno che l'ID utente non sia contenuto tra virgolette singole.

Per i server su sistemi UNIX, Linux e Windows , il contenuto del campo `MCAUserIdentifier` ricevuto dal client viene modificato in minuscolo.

Per i server su IBM i, il contenuto del campo `LongMCAUserIdentifier` ricevuto dal client viene modificato in maiuscolo.

Per i server su sistemi UNIX and Linux , il contenuto del campo `LongMCAUserIdentifier` ricevuto dal client viene modificato in minuscolo.

Per impostazione predefinita, l'ID utente che viene passato quando viene utilizzata un'applicazione di binding MQ JMS è l'ID utente per la JVM su cui è in esecuzione l'applicazione.

È anche possibile passare un ID utente tramite il metodo `createQueueConnection` .

Pianificazione della riservatezza

Pianificare come mantenere riservati i dati.

È possibile implementare la riservatezza a livello di applicazione o a livello di collegamento. È possibile scegliere di utilizzare SSL o TLS, nel qual caso è necessario pianificare l'utilizzo dei certificati digitali. È anche possibile utilizzare i programmi di uscita del canale se le funzioni standard non soddisfano i requisiti.

Concetti correlati

"Confronto tra sicurezza a livello di collegamento e sicurezza a livello di applicazione" a pagina 60
Questo argomento contiene informazioni su vari aspetti della sicurezza a livello di collegamento e a livello di applicazione e confronta i due livelli di sicurezza.

"Programmi di uscita canale" a pagina 65
I *programmi di uscita del canale* sono programmi richiamati in posizioni definite nella sequenza di elaborazione di un MCA. Gli utenti e i fornitori possono scrivere i propri programmi di uscita del canale. Alcuni sono forniti da IBM.

"Protezione dei canali con SSL" a pagina 72
Il supporto SSL in IBM WebSphere MQ utilizza l'oggetto delle informazioni di autenticazione del gestore code e vari comandi MQSC. È inoltre necessario considerare il proprio utilizzo di certificati digitali.

Confronto tra sicurezza a livello di collegamento e sicurezza a livello di applicazione

Questo argomento contiene informazioni su vari aspetti della sicurezza a livello di collegamento e a livello di applicazione e confronta i due livelli di sicurezza.

Il livello di collegamento e la sicurezza a livello di applicazione sono illustrati in Figura 8 a pagina 60.

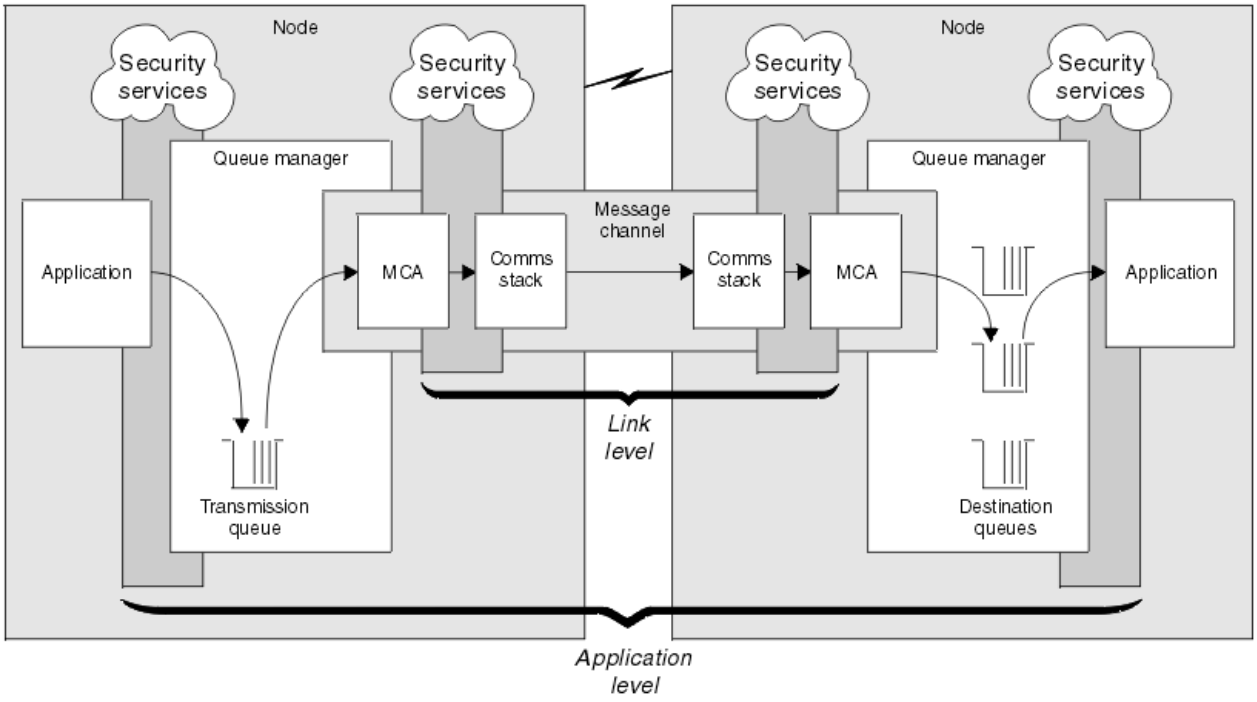


Figura 8. Sicurezza a livello di collegamento e sicurezza a livello di applicazione

Protezione dei messaggi nelle code

La sicurezza a livello di link può proteggere i messaggi mentre vengono trasferiti da un gestore code a un altro. È particolarmente importante quando i messaggi vengono trasmessi su una rete non sicura. Non può, tuttavia, proteggere i messaggi mentre sono memorizzati nelle code in un gestore code di origine, in un gestore code di destinazione o in un gestore code intermedio.

La sicurezza a livello di applicazione, per confronto, può proteggere i messaggi mentre sono memorizzati in code e si applica anche quando non viene utilizzata l'accodamento distribuito. Questa è la differenza principale tra la sicurezza a livello di link e la sicurezza a livello di applicazione ed è illustrata in Figura 8 a pagina 60.

Gestori code non in esecuzione in ambienti controllati e attendibili

Se un gestore code è in esecuzione in un ambiente controllato e sicuro, i meccanismi di controllo accessi forniti da WebSphere MQ potrebbero essere considerati sufficienti per proteggere i messaggi memorizzati nelle relative code. Ciò è particolarmente vero se è coinvolta solo l'accodamento locale e i messaggi non lasciano mai il gestore code. In questo caso, la sicurezza a livello di applicazione potrebbe essere considerata non necessaria.

La sicurezza a livello di applicazione potrebbe anche essere considerata non necessaria se i messaggi vengono trasferiti a un altro gestore code che è in esecuzione anche in un ambiente controllato e attendibile o se vengono ricevuti da tale gestore code. La necessità di sicurezza a livello di applicazione diventa maggiore quando i messaggi vengono trasferiti a, o ricevuti da, un gestore code che non è in esecuzione in un ambiente controllato e attendibile.

Differenze di costo

La sicurezza a livello di applicazione potrebbe costare più della sicurezza a livello di link in termini di gestione e prestazioni.

È probabile che il costo di gestione sia maggiore perché vi sono potenzialmente più vincoli da configurare e gestire. Ad esempio, potrebbe essere necessario assicurarsi che un particolare utente invii solo determinati tipi di messaggi e invii messaggi solo a determinate destinazioni. Al contrario, potrebbe essere necessario assicurarsi che un particolare utente riceva solo determinati tipi di messaggi e riceva messaggi solo da determinate origini. Invece di gestire i servizi di sicurezza a livello di link su un singolo canale di messaggi, potrebbe essere necessario configurare e gestire le regole per ogni coppia di utenti che si scambiano messaggi su tale canale.

Le prestazioni potrebbero essere influenzate se i servizi di sicurezza vengono richiamati ogni volta che un'applicazione inserisce o riceve un messaggio.

Le organizzazioni tendono a considerare prima la sicurezza del livello di collegamento perché potrebbe essere più semplice da implementare. Considerano la sicurezza a livello di applicazione se rilevano che la sicurezza a livello di collegamento non soddisfa tutti i requisiti.

Disponibilità dei componenti

Generalmente, in un ambiente distribuito, un servizio di sicurezza richiede un componente su almeno due sistemi. Ad esempio, un messaggio potrebbe essere codificato su un sistema e decodificato su un altro. Ciò si applica sia alla sicurezza a livello di link che a livello di applicazione.

In un ambiente eterogeneo, con diverse piattaforme in uso, ognuna con diversi livelli di funzione di sicurezza, i componenti richiesti di un servizio di sicurezza potrebbero non essere disponibili per ogni piattaforma su cui sono necessari e in un formato facile da utilizzare. Questo è probabilmente più un problema per la sicurezza a livello di applicazione che per la sicurezza a livello di collegamento, in particolare se si intende fornire la propria sicurezza a livello di applicazione acquistando componenti da varie fonti.

Messaggi in una coda di messaggi non recapitabili

Se un messaggio è protetto dalla sicurezza a livello di applicazione, potrebbe verificarsi un problema se, per qualsiasi motivo, il messaggio non raggiunge la destinazione e viene inserito in una coda di messaggi non recapitabili. Se non è possibile determinare in che modo elaborare il messaggio dalle informazioni nel descrittore del messaggio e nell'intestazione dei messaggi non recapitabili, potrebbe essere necessario esaminare il contenuto dei dati dell'applicazione. Non è possibile eseguire questa operazione se i dati dell'applicazione sono codificati e solo il destinatario desiderato può decodificarli.

Cosa non può fare la sicurezza del livello di applicazione

La sicurezza a livello di applicazione non è una soluzione completa. Anche se si implementa la sicurezza a livello di applicazione, potrebbero essere ancora necessari alcuni servizi di sicurezza a livello di collegamento. Ad esempio:

- Quando un canale viene avviato, l'autenticazione reciproca dei due MCA potrebbe essere ancora un requisito. Ciò può essere eseguito solo da un servizio di sicurezza a livello di collegamento.
- La sicurezza del livello di applicazione non può proteggere l'intestazione della coda di trasmissione, MQXQH, che include il descrittore del messaggio incorporato. Inoltre, non può proteggere i dati nei flussi del protocollo del canale WebSphere MQ diversi dai dati del messaggio. Solo la sicurezza a livello di collegamento può fornire questa protezione.
- Se i servizi di sicurezza a livello di applicazione vengono richiamati all'estremità server di un canale MQI, i servizi non possono proteggere i parametri delle chiamate MQI inviate sul canale. In particolare, i dati dell'applicazione in una chiamata MQPUT, MQPUT1o MQGET non sono protetti. Solo la sicurezza a livello di collegamento può fornire la protezione in questo caso.

sicurezza a livello di collegamento

La *sicurezza del livello di collegamento* fa riferimento ai servizi di sicurezza richiamati, direttamente o indirettamente, da un MCA, dal sottosistema di comunicazione o da una combinazione dei due che lavorano insieme.

La sicurezza del livello di collegamento è illustrata in Figura 8 a pagina 60.

Di seguito sono riportati alcuni esempi di servizi di protezione a livello di link:

- L'MCA a ciascuna estremità di un canale di messaggi può autenticare il proprio partner. Questa operazione viene eseguita quando il canale viene avviato ed è stata stabilita una connessione di comunicazione, ma prima che i messaggi inizino a fluire. Se l'autenticazione non riesce ad entrambe le estremità, il canale viene chiuso e non viene trasferito alcun messaggio. Questo è un esempio di un servizio di identificazione e autenticazione.
- Un messaggio può essere codificato all'estremità di invio di un canale e decodificato all'estremità di ricezione. Questo è un esempio di servizio di riservatezza.
- Un messaggio può essere controllato all'estremità di ricezione di un canale per determinare se il suo contenuto è stato deliberatamente modificato mentre veniva trasmesso sulla rete. Questo è un esempio di un servizio di integrità dati.

Sicurezza a livello di collegamento fornita da IBM WebSphere MQ

Il mezzo principale per fornire la riservatezza e l'integrità dei dati in IBM WebSphere MQ è l'utilizzo di SSL o TLS. Per ulteriori informazioni sull'utilizzo di SSL e TLS in IBM WebSphere MQ, consultare “Supporto IBM WebSphere MQ per SSL e TLS” a pagina 23. Per l'autenticazione, IBM WebSphere MQ fornisce la funzione per utilizzare i record di autenticazione di canale. I record di autenticazione di canale offrono un controllo preciso sull'accesso concesso ai sistemi di collegamento, a livello di singoli canali o gruppi di canali. Per ulteriori informazioni, consultare “Record di autenticazione di canale” a pagina 39.

Fornire la propria sicurezza a livello di link

Questa raccolta di argomenti descrive come è possibile fornire i propri servizi di sicurezza a livello di link. La scrittura dei propri programmi di uscita del canale è il modo principale per fornire i propri servizi di sicurezza a livello di link.

I programmi di uscita del canale sono introdotti in “Programmi di uscita canale” a pagina 65. Lo stesso argomento descrive anche il programma di uscita canale fornito con IBM WebSphere MQ per Windows (il programma di uscita canale SSPI). Questo programma di uscita del canale viene fornito in formato origine in modo da poter modificare il codice sorgente in base ai propri requisiti. Se questo programma di uscita del canale, o i programmi di uscita del canale disponibili da altri fornitori, non soddisfano i requisiti, è possibile progettare e scrivere il proprio. Questo argomento suggerisce i modi in cui i programmi di uscita del canale possono fornire servizi di sicurezza. Per informazioni su come scrivere un programma di uscita canale, consultare Scrittura di programmi di uscita canale.

Sicurezza del livello di collegamento utilizzando un'uscita di sicurezza
Le uscite di sicurezza normalmente funzionano a coppie; una a ciascuna estremità di un canale. Vengono richiamati immediatamente dopo che la negoziazione dati iniziale è stata completata all'avvio del canale.

Le uscite di sicurezza possono essere utilizzate per fornire identificazione e autenticazione, controllo degli accessi e riservatezza.

Sicurezza del livello di collegamento utilizzando un'uscita messaggio
Un'uscita messaggio può essere utilizzata solo su un canale di messaggi, non su un canale MQI.
Ha accesso sia all'intestazione della coda di trasmissione, MQXQH, che comprende il descrittore del messaggio incorporato, sia ai dati dell'applicazione in un messaggio. Può modificare il contenuto del messaggio e modificarne la lunghezza.

Un'uscita messaggio può essere utilizzata per qualsiasi scopo che richieda l'accesso all'intero messaggio piuttosto che a una parte di esso.

Le uscite dei messaggi possono essere utilizzate per fornire identificazione e autenticazione, controllo degli accessi, riservatezza, integrità dei dati e non rifiuto e per motivi diversi dalla sicurezza.

Sicurezza a livello di collegamento mediante uscite di invio e ricezione
Le uscite di invio e ricezione possono essere utilizzate su entrambi i canali MQI e messaggi. Vengono richiamati per tutti i tipi di dati che fluiscono su un canale e per i flussi in entrambe le direzioni.

Le uscite di invio e ricezione hanno accesso a ciascun segmento di trasmissione. Essi possono modificarne il contenuto e la lunghezza.

Su un canale di messaggi, se un MCA deve suddividere un messaggio e inviarlo in più di un segmento di trasmissione, viene richiamata un'uscita di invio per ciascun segmento di trasmissione contenente una porzione del messaggio e, all'estremità di ricezione, viene richiamata un'uscita di ricezione per ciascun segmento di trasmissione. Lo stesso si verifica su un canale MQI se i parametri di input o output di una chiamata MQI sono troppo grandi per essere inviati in un singolo segmento di trasmissione.

Su un canale MQI, il byte 10 di un segmento di trasmissione identifica la chiamata MQI e indica se il segmento di trasmissione contiene i parametri di input o output della chiamata. Le uscite di invio e ricezione possono esaminare questo byte per stabilire se la chiamata MQI contiene dati dell'applicazione che potrebbero dover essere protetti.

Quando un'uscita di invio viene richiamata per la prima volta, per acquisire e inizializzare tutte le risorse di cui ha bisogno, può richiedere all'MCA di riservare una quantità specificata di spazio nel buffer che contiene un segmento di trasmissione. Quando viene chiamato successivamente per elaborare un segmento di trasmissione, può utilizzare questo spazio per aggiungere una chiave codificata o una firma digitale, ad esempio. L'uscita di ricezione corrispondente all'altra estremità del canale può rimuovere i dati aggiunti dall'uscita di trasmissione e utilizzarli per elaborare il segmento di trasmissione.

Le uscite di invio e ricezione sono più adatte per scopi in cui non hanno bisogno di comprendere la struttura dei dati che stanno gestendo e possono quindi trattare ogni segmento di trasmissione come un oggetto binario.

Le uscite di invio e ricezione possono essere utilizzate per fornire riservatezza e integrità dei dati e per usi diversi dalla sicurezza.

Attività correlate
Identificazione della chiamata API in un programma di uscita di invio o ricezione

sicurezza a livello di applicazioni
La *sicurezza a livello di applicazione* fa riferimento ai servizi di sicurezza richiamati nell'interfaccia tra un'applicazione e il gestore code a cui è connessa.

Questi servizi vengono richiamati quando l'applicazione emette chiamate MQI al gestore code. I servizi possono essere richiamati, direttamente o indirettamente, dall'applicazione, dal gestore code, da un altro prodotto che supporta WebSphere MQo da una combinazione di questi. La sicurezza a livello di applicazione viene illustrata in Figura 8 a pagina 60.

La sicurezza a livello di applicazione è nota anche come *sicurezza end - to - end* o *sicurezza a livello di messaggio*.

Di seguito sono riportati alcuni esempi di servizi di sicurezza a livello di applicazione:

- Quando un'applicazione inserisce un messaggio in una coda, il descrittore del messaggio contiene un ID utente associato con l'applicazione. Tuttavia, non sono presenti dati, come una password codificata, che possono essere utilizzati per autenticare l'ID utente. Un servizio di sicurezza può aggiungere questi dati. Quando il messaggio viene richiamato dall'applicazione ricevente, un altro componente del servizio può autenticare l'ID utente utilizzando i dati trasmessi con il messaggio.Questo è un esempio di un servizio di identificazione e autenticazione.
- Un messaggio può essere codificato quando viene inserito su una coda da un'applicazione e decodificato quando viene richiamato dall'applicazione ricevente.Questo è un esempio di servizio di riservatezza.
- Un messaggio può essere controllato quando viene richiamato dall'applicazione ricevente. Questo controllo determina se il contenuto è stato deliberatamente modificato da quando è stato inserito per la prima volta in una coda dall'applicazione mittente.Questo è un esempio di un servizio di integrità dati.

pianificazione per Advanced Message Security
IBM WebSphere MQ Advanced Message Security (AMS) è un componente con licenza separata di IBM WebSphere MQ che fornisce un livello elevato di protezione per i dati sensibili che passano attraverso la rete IBM WebSphere MQ , senza influire sulle applicazioni finali.

Se si stanno spostando informazioni altamente sensibili o preziose, in particolare informazioni riservate o relative ai pagamenti, come i dati dei pazienti o i dati della carta di credito, è necessario prestare particolare attenzione alla sicurezza delle informazioni. Garantire che le informazioni che si spostano all'interno dell'azienda conservino la sua integrità e siano protette da accessi non autorizzati è una sfida e una responsabilità continua. È anche probabile che ti venga richiesto di rispettare le norme di sicurezza, a rischio di sanzioni per non conformità.

È possibile sviluppare le proprie estensioni di sicurezza in IBM WebSphere MQ. Tuttavia, tali soluzioni richiedono competenze specialistiche e possono essere complicate e costose da mantenere. IBM WebSphere MQ Advanced Message Security consente di rispondere a queste sfide quando si spostano le informazioni all'interno dell'azienda, praticamente tra ogni tipo di sistema IT commerciale.

IBM WebSphere MQ Advanced Message Security estende le funzioni di protezione di IBM WebSphere MQ nei modi seguenti:

- Fornisce protezione dei dati end-to-end a livello di applicazione per l'infrastruttura di messaggistica point - to - point, utilizzando la crittografia o la firma digitale dei messaggi.
- Fornisce una sicurezza completa senza scrivere codice di sicurezza complesso o modificare o ricompilare le applicazioni esistenti.
- Utilizza la tecnologia PKI (Public Key Infrastructure) per fornire servizi di autenticazione, autorizzazione, riservatezza e integrità dei dati per i messaggi.
- Fornisce la gestione delle policy di sicurezza per mainframe e server distribuiti.
- Supporta sia i client che i server IBM WebSphere MQ .
- Si integra con IBM WebSphere MQ Managed File Transfer per fornire una soluzione di messaggistica sicura end - to - end.

Per ulteriori informazioni, vedere “IBM WebSphere MQ Advanced Message Security” a pagina 271.

Come fornire la propria sicurezza a livello di applicazione
Questa raccolta di argomenti descrive come è possibile fornire i servizi di sicurezza a livello di applicazione.

Per facilitare l'implementazione della protezione a livello di applicazione, IBM WebSphere MQ fornisce due uscite, l'uscita API e l'uscita incrociata API.

Queste uscite possono fornire l'identificazione e l'autenticazione, il controllo degli accessi, la riservatezza, l'integrità dei dati e i servizi di non rifiuto e altre funzioni non correlate alla sicurezza.

Se l'uscita API o l'uscita incrociata API non è supportata nel proprio ambiente di sistema, è possibile considerare altri modi per fornire la propria sicurezza a livello di applicazione. Un modo è sviluppare un'API di livello superiore che incapsula MQI. I programmatori utilizzano quindi questa API, invece di MQI, per scrivere applicazioni IBM WebSphere MQ .

I motivi più comuni per utilizzare un'API di livello superiore sono:

- Per nascondere le funzioni più avanzate di MQI ai programmatori.
- Per applicare gli standard nell'utilizzo di MQI.
- Per aggiungere una funzione a MQI. Questa funzione aggiuntiva può essere servizi di sicurezza.

Alcuni prodotti del fornitore utilizzano questa tecnologia per fornire la sicurezza a livello di applicazione per IBM WebSphere MQ.

Se si intende fornire servizi di sicurezza in questo modo, tenere presente quanto segue per quanto riguarda la conversione dei dati:

- Se un token di sicurezza, come una firma digitale, è stato aggiunto ai dati dell'applicazione in un messaggio, qualsiasi codice che esegue la conversione dei dati deve essere consapevole della presenza di questo token.
- Un token di sicurezza potrebbe essere stato derivato da un'immagine binaria dei dati dell'applicazione. Pertanto, qualsiasi controllo del token deve essere eseguito prima della conversione dei dati.
- Se i dati dell'applicazione in un messaggio sono stati codificati, devono essere decodificati prima della conversione dei dati.

Programmi di uscita canale

I *programmi di uscita del canale* sono programmi richiamati in posizioni definite nella sequenza di elaborazione di un MCA. Gli utenti e i fornitori possono scrivere i propri programmi di uscita del canale. Alcuni sono forniti da IBM.

Esistono diversi tipi di programmi di uscita del canale, ma solo quattro hanno un ruolo nel fornire la sicurezza a livello di collegamento:

- Uscita di sicurezza
- Uscita messaggi
- Uscita invio
- Uscita ricezione

Questi quattro tipi di programmi di uscita canale sono illustrati in Figura 9 a pagina 66 e sono descritti nei seguenti argomenti.

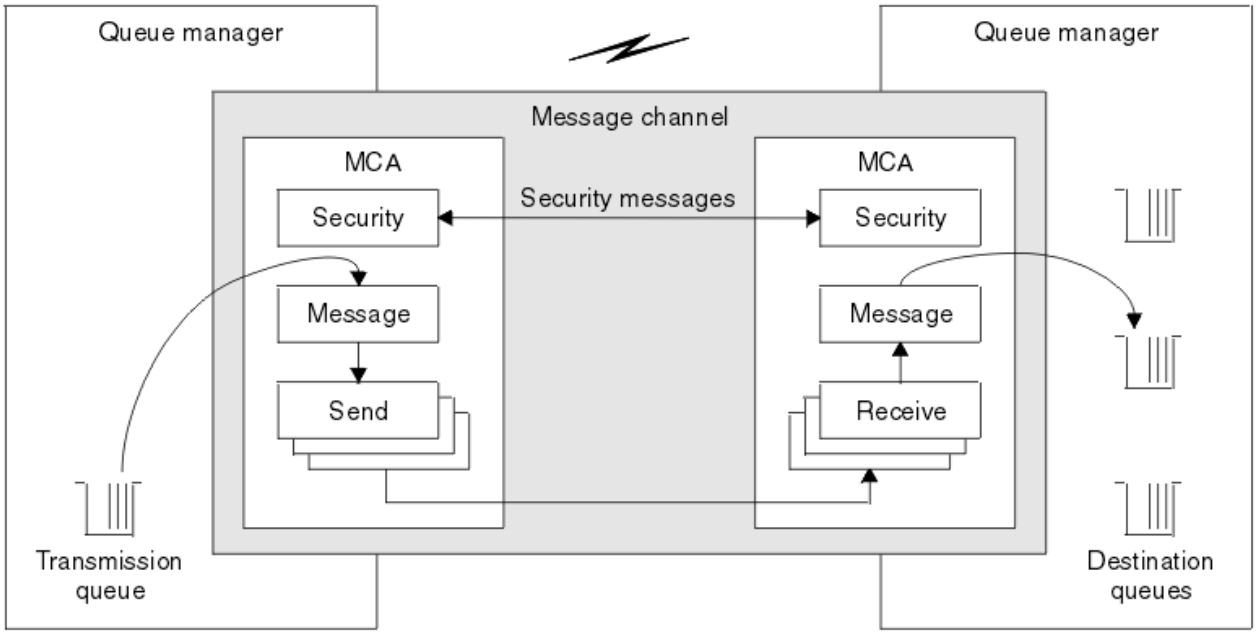


Figura 9. Uscite di sicurezza, messaggio, invio e ricezione su un canale di messaggi

Concetti correlati

Programmi di uscita canale per canali di messaggistica

Panoramica sull'uscita di sicurezza

Le uscite di sicurezza normalmente funzionano a coppie. Vengono richiamati prima del flusso di messaggi e il loro scopo è quello di permettere a un MCA di autenticare il proprio partner.

Le *uscite di sicurezza* normalmente funzionano a coppie; una ad ogni estremità di un canale. Vengono richiamati immediatamente dopo che la negoziazione dei dati iniziali è stata completata all'avvio del canale, ma prima che i messaggi inizino a fluire. Lo scopo principale dell'uscita di sicurezza è abilitare l'MCA ad ogni estremità di un canale per autenticare il relativo partner. Tuttavia, non c'è nulla che impedisca a un'uscita di sicurezza di eseguire altre funzioni, anche se non ha niente a che fare con la sicurezza.

Le uscite di sicurezza possono comunicare tra loro inviando *messaggi di sicurezza*. Il formato di un messaggio di sicurezza non è definito ed è determinato dall'utente. Un possibile risultato dello scambio di messaggi di sicurezza è che una delle uscite di sicurezza potrebbe decidere di non procedere ulteriormente. In tal caso, il canale viene chiuso e i messaggi non vengono trasmessi. Se c'è un'uscita di sicurezza ad una sola estremità di un canale, l'uscita viene ancora richiamata e può scegliere se continuare o chiudere il canale.

Le uscite di sicurezza possono essere richiamate su entrambi i canali MQI e messaggi. Il nome di un'uscita di sicurezza viene specificato come parametro nella definizione di canale ad ogni estremità di un canale.

Per ulteriori informazioni sulle uscite di sicurezza, consultare “Sicurezza del livello di collegamento utilizzando un'uscita di sicurezza” a pagina 63.

Uscita messaggi

Le uscite dei messaggi funzionano solo su canali di messaggi e normalmente funzionano a coppie. Un'uscita messaggio può operare sull'intero messaggio e apportare varie modifiche.

Le uscite dei messaggi alle estremità di invio e ricezione di un canale normalmente funzionano a coppie. Un'uscita messaggio all'estremità di invio di un canale viene richiamata dopo che l'MCA ha ricevuto un messaggio dalla coda di trasmissione. All'estremità di ricezione di un canale, viene richiamata un'uscita del messaggio prima che l'MCA inserisce un messaggio nella coda di destinazione.

Un'uscita del messaggio ha accesso sia all'intestazione della coda di trasmissione, MQXQH, che comprende il descrittore del messaggio incorporato, sia ai dati dell'applicazione in un messaggio. Un'uscita messaggio può modificarne il contenuto e la lunghezza. Una modifica della lunghezza potrebbe essere il risultato della compressione, decompressione, codifica o decodifica del messaggio. Potrebbe anche essere il risultato dell'aggiunta di dati al messaggio o della rimozione di dati da esso.

Le uscite dei messaggi possono essere utilizzate per qualsiasi scopo che richieda l'accesso all'intero messaggio, piuttosto che a una parte di esso, e non necessariamente per la sicurezza.

Un'uscita del messaggio può determinare che il messaggio che sta attualmente elaborando non deve procedere ulteriormente verso la sua destinazione. L'MCA inserisce il messaggio nella coda di messaggi non recapitabili. Un'uscita messaggio può anche chiudere il canale.

Le uscite dei messaggi possono essere richiamate solo sui canali dei messaggi, non sui canali MQI. Ciò è dovuto al fatto che lo scopo di un canale MQI è abilitare i parametri di input e output delle chiamate MQI al flusso tra l'applicazione client IBM WebSphere MQ MQI e il gestore code.

Il nome di un'uscita messaggio è specificato come parametro nella definizione di canale ad ogni estremità di un canale. È inoltre possibile specificare un elenco di uscite di messaggi da eseguire in successione.

Per ulteriori informazioni sulle uscite dei messaggi, consultare “Sicurezza del livello di collegamento utilizzando un'uscita messaggio” a pagina 63.

Uscite di invio e ricezione

Le uscite di invio e ricezione generalmente funzionano in coppie. Operano su segmenti di trasmissione e sono utilizzati al meglio quando la struttura dei dati che stanno elaborando non è pertinente.

Un' *uscita di trasmissione* ad un'estremità di un canale e un' *uscita di ricezione* all'altra estremità normalmente funzionano a coppie. Un'uscita di invio viene richiamata appena prima che un MCA emani un invio di comunicazioni per inviare dati su una connessione di comunicazione. Un'uscita di ricezione viene richiamata subito dopo che un MCA ha riacquistato il controllo dopo una ricezione di comunicazioni e ha ricevuto dati da una connessione di comunicazione. Se la condivisione delle conversazioni è in uso, su un canale MQI, viene richiamata una diversa istanza di un'uscita di invio e ricezione per ciascuna conversazione.

I flussi del protocollo del canale IBM WebSphere MQ tra due MCA su un canale di messaggi contengono informazioni di controllo e dati del messaggio. Allo stesso modo, su un canale MQI, i flussi contengono informazioni di controllo e i parametri delle chiamate MQI. Le uscite di invio e ricezione vengono richiamate per tutti i tipi di dati.

I flussi di dati dei messaggi in una sola direzione su un canale di messaggi ma, su un canale MQI, i parametri di input di un flusso di chiamate MQI in una direzione e i parametri di output nell'altra. Su entrambi i canali MQI e messaggi, controllare il flusso di informazioni in entrambe le direzioni. Di conseguenza, le uscite di invio e ricezione possono essere richiamate ad entrambe le estremità di un canale.

L'unità di dati trasmessa in un singolo flusso tra due MCA è denominata *segmento trasmissione*. Le uscite di invio e ricezione hanno accesso a ciascun segmento di trasmissione. Essi possono modificarne il contenuto e la lunghezza. Un'uscita di invio, tuttavia, non deve modificare i primi 8 byte di un segmento di trasmissione. Questi 8 byte fanno parte dell'intestazione del protocollo del canale IBM WebSphere MQ . Ci sono anche restrizioni su quanto un'uscita di invio può aumentare la lunghezza di un segmento di trasmissione. In particolare, un'uscita di invio non può aumentare la sua lunghezza oltre il valore massimo negoziato tra i due MCA all'avvio del canale.

Su un canale di messaggi, se un messaggio è troppo grande per essere inviato in un singolo segmento di trasmissione, l'MCA mittente suddivide il messaggio e lo invia in più di un segmento di trasmissione. Di conseguenza, viene richiamata un'exit di invio per ogni segmento di trasmissione contenente una parte del messaggio e, all'estremità ricevente, viene richiamata un'exit di ricezione per ogni segmento. L'MCA di ricezione ricostituisce il messaggio dai segmenti di trasmissione dopo che sono stati elaborati dall'uscita di ricezione.

Allo stesso modo, su un canale MQI, i parametri di input o output di una chiamata MQI vengono inviati in più di un segmento di trasmissione se sono troppo grandi. Ciò potrebbe verificarsi, ad esempio, in una chiamata MQPUT, MQPUT1o MQGET se i dati dell'applicazione sono sufficientemente grandi.

Tenendo conto di queste considerazioni, è più appropriato utilizzare le uscite di invio e ricezione per scopi in cui non hanno bisogno di comprendere la struttura dei dati che stanno gestendo e possono quindi trattare ogni segmento di trasmissione come un oggetto binario.

Un'uscita di invio o di ricezione può chiudere un canale.

I nomi di un'uscita di invio e di un'uscita di ricezione sono specificati come parametri nella definizione del canale ad ogni estremità di un canale. È inoltre possibile specificare un elenco di uscite di invio da eseguire in successione. Allo stesso modo, è possibile specificare un elenco di uscite di ricezione.

Per ulteriori informazioni sulle uscite di invio e ricezione, consultare “Sicurezza a livello di collegamento mediante uscite di invio e ricezione” a pagina 63.

Pianificazione dell'integrità dei dati

Pianificare come preservare l'integrità dei dati.

È possibile implementare l'integrità dei dati a livello di applicazione o a livello di collegamento.

A livello di applicazione, è possibile scegliere di utilizzare IBM WebSphere MQ Advanced Message Security per firmare digitalmente i messaggi al fine di proteggerli da modifiche non autorizzate. È anche possibile utilizzare i programmi di uscita API se le funzioni standard non soddisfano i requisiti.

A livello di link, puoi scegliere di utilizzare SSL o TLS, nel qual caso devi pianificare il tuo utilizzo dei certificati digitali. È anche possibile utilizzare i programmi di uscita del canale se le funzioni standard non soddisfano i requisiti.

Concetti correlati

“Protezione dei canali con SSL” a pagina 72
Il supporto SSL in IBM WebSphere MQ utilizza l'oggetto delle informazioni di autenticazione del gestore code e vari comandi MQSC. È inoltre necessario considerare il proprio utilizzo di certificati digitali.

“Integrità dei dati in IBM WebSphere MQ” a pagina 22
È possibile utilizzare un servizio di integrità dati per rilevare se un messaggio è stato modificato.

“pianificazione per Advanced Message Security” a pagina 64
IBM WebSphere MQ Advanced Message Security (AMS) è un componente con licenza separata di IBM WebSphere MQ che fornisce un livello elevato di protezione per i dati sensibili che passano attraverso la rete IBM WebSphere MQ , senza influire sulle applicazioni finali.

Riferimenti correlati

Riferimento uscita API

Chiamate di uscita canale e strutture dati

Controllo pianificazione

Decidere quali dati è necessario controllare e come si acquisiranno ed elaboreranno le informazioni di controllo. Considerare come verificare che il sistema sia configurato correttamente.

Il monitoraggio delle attività presenta diversi aspetti. Gli aspetti da considerare sono spesso definiti dai requisiti del revisore, e questi requisiti sono spesso guidati da standard normativi come HIPAA (Health Insurance Portability and Accountability Act) o SOX (Sarbanes-Oxley). IBM WebSphere MQ fornisce funzioni destinate a facilitare la conformità a tali standard.

Considerare se si è interessati solo alle eccezioni o se si è interessati a tutti i comportamenti del sistema.

Alcuni aspetti della verifica possono anche essere considerati come monitoraggio operativo; una distinzione per la verifica è che spesso si stanno esaminando i dati storici, non solo gli avvisi in tempo reale. Il monitoraggio è descritto nella sezione Monitoraggio e prestazioni .

Quali dati controllare

Considerare quali tipi di dati o attività è necessario controllare, come descritto nelle seguenti sezioni:

Modifiche apportate a IBM WebSphere MQ utilizzando le interfacce IBM WebSphere MQ
Configurare IBM WebSphere MQ per emettere eventi di strumentazione, in particolare eventi di comando ed eventi di configurazione.

Le modifiche apportate a IBM WebSphere MQ al di fuori del suo controllo
Alcune modifiche possono influire sul funzionamento di IBM WebSphere MQ , ma non possono essere monitorate direttamente da IBM WebSphere MQ. Esempi di tali modifiche includono le modifiche ai file di configurazione `mqs.ini`, `qm.inie` `mqclient.ini`, la creazione e l'eliminazione dei gestori code, l'installazione di file binari come i programmi di uscita utente e le modifiche alle autorizzazioni file. Per monitorare queste attività, è necessario utilizzare strumenti in esecuzione a livello del sistema operativo. Diversi strumenti sono disponibili e appropriati per i diversi sistemi operativi. Potresti anche avere dei log creati dagli strumenti associati come *sudo*.

Controllo operativo di IBM WebSphere MQ
Potrebbe essere necessario utilizzare gli strumenti del sistema operativo per controllare le attività come l'avvio e l'arresto dei gestori code. In alcuni casi, IBM WebSphere MQ può essere configurato per emettere eventi di strumentazione.

Attività dell'applicazione all'interno di IBM WebSphere MQ
Per controllare le azioni delle applicazioni, ad esempio l'apertura di code e l'inserimento e il richiamo di messaggi, configurare IBM WebSphere MQ per emettere eventi appropriati.

Avvisi intruso
Per controllare i tentativi di violazione della protezione, configurare il proprio sistema per emettere gli eventi di autorizzazione. Gli eventi del canale possono essere utili anche per mostrare l'attività, in particolare se un canale termina in modo imprevisto.

Pianificazione dell'acquisizione, visualizzazione e archiviazione dei dati di verifica

Molti degli elementi necessari vengono riportati come messaggi di evento IBM WebSphere MQ . È necessario scegliere gli strumenti che possono leggere e formattare questi messaggi. Se si è interessati alla memoria a lungo termine e all'analisi, è necessario spostarli in un meccanismo di memoria ausiliaria come un database. Se non si elaborano questi messaggi, essi rimangono nella coda eventi, probabilmente riempiendo la coda. Potresti decidere di implementare uno strumento che esegue automaticamente delle azioni in base ad alcuni eventi; ad esempio, per emettere un avviso quando si verifica un errore di sicurezza.

Verifica della corretta configurazione del sistema

Una serie di test viene fornita con IBM WebSphere MQ Explorer. Utilizzare queste informazioni per verificare la presenza di problemi nelle definizioni degli oggetti.

Inoltre, controllare periodicamente che la configurazione del sistema sia quella prevista. Sebbene i comandi e gli eventi di configurazione possano segnalare quando qualcosa viene modificato, è utile anche eseguire il dump della configurazione e confrontarla con una buona copia nota.

Pianificazione della sicurezza per topologia

Questa sezione riguarda la sicurezza in situazioni specifiche, in particolare per canali, cluster di gestori code, applicazioni di pubblicazione / sottoscrizione e multicast e quando si utilizza un firewall.

Per ulteriori informazioni, consultare i seguenti argomenti secondari:

Autorizzazione canale

Quando si invia o si riceve un messaggio tramite un canale, è necessario un ID utente che abbia accesso a varie risorse IBM WebSphere MQ .

Per ricevere i messaggi in fase di PUT per gli MCA, è possibile utilizzare l'ID utente associato all'MCA o l'ID utente associato al messaggio.

Al momento di CONNECT è possibile associare l'ID utente asserito ad un utente alternativo, utilizzando i record di autenticazione di canale **CHLAUTH** .

In WebSphere MQ, i canali possono essere protetti dal supporto SSL o TLS.

Gli ID utente associati ai canali di invio e ricezione, escluso il canale mittente in cui l'attributo MCAUSER non è utilizzato, richiedono l'accesso alle seguenti risorse:

- L'ID utente associato a un canale di invio richiede l'accesso al gestore code, alla coda di trasmissione, alla coda di messaggi non recapitabili e l'accesso a tutte le altre risorse richieste dalle uscite del canale.
- L'ID utente MCAUSER di un canale ricevente necessita dell'autorità *+ setall* .

  Il motivo è che il canale destinatario deve creare l'MQMD completo, inclusi tutti i campi di contesto, utilizzando i dati ricevuti dal canale mittente remoto.

  Il gestore code richiede quindi che l'utente che esegue questa attività disponga dell'autorizzazione *+ setall* . Questa autorizzazione *+ setall* deve essere concessa all'utente per:

  – Tutte le code in cui il canale ricevente inserisce i messaggi in modo valido.
  – L'oggetto gestore code. Per ulteriori informazioni, vedi Autorizzazioni per il contesto .
- L'ID utente MCAUSER di un canale destinatario in cui il mittente ha richiesto un messaggio di report COA richiede l'autorizzazione *+ passid* sulla coda di trasmissione che restituisce il messaggio di report. Senza questa autorità, vengono registrati i messaggi di errore AMQ8077 .
- Con l'ID utente associato al canale ricevente, è possibile aprire le code di destinazione per inserire i messaggi nelle code.

  Ciò implica l'interfaccia MQI (Message queuing Interface), pertanto potrebbe essere necessario effettuare ulteriori controlli del controllo accessi se non si utilizza OAM (Object Authority Manager) WebSphere MQ . È possibile specificare se le verifiche di autorizzazione vengono effettuate sull'ID utente associato all'MCA (come descritto in questo argomento) o sull'ID utente associato al messaggio (dal campo MQMD UserIdentifier).

  Per i tipi di canale a cui si applica, il parametro **PUTAUT** di una definizione di canale specifica quale ID utente viene utilizzato per questi controlli.

  – Per impostazione predefinita, il canale utilizza l'account di servizio del gestore code, che avrà tutti i diritti di gestione e non richiede autorizzazioni speciali

    Nel caso di canali di connessione server, le connessioni amministrative sono bloccate per impostazione predefinita dalle regole CHLAUTH e richiedono un provisioning esplicito.

    I canali di tipo ricevente, richiedente e ricevente del cluster consentono la gestione locale da qualsiasi gestore code adiacente, a meno che l'amministratore non effettui delle operazioni per limitare questo accesso.
  – Se si utilizza un ID utente che non dispone dei privilegi amministrativi di WebSphere , è necessario concedere l'autorizzazione dsp e ctrlx per il canale a tale ID utente per il funzionamento del canale. L'attributo MCAUSER non è utilizzato per il tipo di canale SDR.
  – Se si utilizza l'ID utente associato al messaggio, è probabile che l'ID utente provenga da un sistema remoto.

    Questo id utente del sistema remoto deve essere riconosciuto dal sistema di destinazione. Ad esempio, immettere i seguenti comandi:

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

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

dove *Profilo* è un canale.

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

dove *Profilo* è una coda di messaggi non instradabili, se impostata.

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

dove *Profilo* è un elenco di code autorizzate.

Attenzione: Prestare attenzione quando si autorizza un ID utente a inserire messaggi nella coda comandi o in altre code di sistema sensibili.

L'ID utente associato all'MCA dipende dal tipo di MCA. Esistono due tipi di MCA:

MCA chiamante
MCA che avviano un canale. Gli MCA del chiamante possono essere avviati come singoli processi, come thread dell'iniziatore del canale o come thread di un pool di processi. L'ID utente utilizzato è l'ID utente associato al processo principale (l'iniziatore del canale) o l'ID utente associato al processo che avvia l'MCA.

MCA responder
Gli MCA responder sono MCA avviati come risultato di una richiesta da parte di un MCA chiamante. Gli MCA responder possono essere avviati come singoli processi, come thread del listener o come thread di un pool di processi. L'ID utente può essere uno dei seguenti tipi (in questo ordine di preferenza):

1. Su APPC, l'MCA del chiamante può indicare l'ID utente da utilizzare per l'MCA del rispondente. Questo viene chiamato ID utente di rete e si applica solo ai canali avviati come singoli processi. Impostare l'ID utente di rete utilizzando il parametro USERID della definizione del canale.
2. Se il parametro **USERID** non viene utilizzato, la definizione del canale dell'MCA del responder può specificare l'ID utente che l'MCA deve utilizzare. Impostare l'ID utente utilizzando il parametro **MCAUSER** della definizione di canale.
3. Se l'ID utente non è stato impostato con uno dei due metodi precedenti, viene utilizzato l'ID utente del processo che avvia l'MCA o l'ID utente del processo parent (il listener).

Concetti correlati

“Record di autenticazione di canale” a pagina 39
Per esercitare un controllo più preciso sull'accesso concesso ai sistemi di connessione a livello di canale, è possibile utilizzare i record di autenticazione di canale.

Proprietà record di autenticazione di canale

Protezione delle definizioni dell'iniziatore di canali

Solo i membri del gruppo mqm possono modificare gli iniziatori di canali.

Gli iniziatori di canale IBM WebSphere MQ non sono oggetti IBM WebSphere MQ ; l'accesso ad essi non è controllato da OAM. IBM WebSphere MQ non permette agli utenti o alle applicazioni di manipolare questi oggetti, a meno che il loro ID utente non sia un membro del gruppo mqm. Se si dispone di un'applicazione che immette il comando PCF `StartChannelInitiator`, l'ID utente specificato nel descrittore del messaggio PCF deve essere un membro del gruppo mqm sul gestore code di destinazione.

Un ID utente deve essere anche un membro del gruppo mqm sulla macchina di destinazione per emettere i comandi MQSC equivalenti tramite il comando Escape PCF o utilizzando `runmqsc` in modalità indiretta.

Code di trasmissione

I gestori code inserano automaticamente i messaggi remoti su una coda di trasmissione; per questo non è richiesta alcuna autorizzazione speciale.

Tuttavia, se è necessario inserire un messaggio direttamente in una coda di trasmissione, ciò richiede un'autorizzazione speciale; consultare Tabella 10 a pagina 90.

Uscite canale

Se i record di autenticazione di canale non sono adatti, è possibile utilizzare le uscite di canale per una maggiore sicurezza. Un'uscita di sicurezza costituisce una connessione sicura tra due programmi di uscita di sicurezza. Un programma è per l'MCA (message channel agent) di invio e uno è per l'MCA di ricezione.

Consultare “Programmi di uscita canale” a pagina 65 per ulteriori informazioni sulle uscite dei canali.

Protezione dei canali con SSL

Il supporto SSL in IBM WebSphere MQ utilizza l'oggetto delle informazioni di autenticazione del gestore code e vari comandi MQSC. È inoltre necessario considerare il proprio utilizzo di certificati digitali.

Comandi e attributi per il supporto SSL

Il protocollo SSL (Secure Sockets Layer) fornisce la sicurezza del canale, con protezione da intercettazioni, manomissioni e imitazioni. Il supporto IBM WebSphere MQ per SSL consente di specificare, sulla definizione di canale, che un particolare canale utilizza la sicurezza SSL. È anche possibile specificare i dettagli del tipo di sicurezza che si desidera, come l'algoritmo di codifica che si desidera utilizzare.

I seguenti comandi MQSC supportano SSL:

MODIFICA AUTHINFO
Modifica gli attributi di un oggetto delle informazioni di autenticazione.

DEFINE AUTINFO
Crea un oggetto delle informazioni di autenticazione.

DELETE AUTINFO
Elimina un oggetto delle informazioni di autenticazione.

VISUALIZZA AUTHINFO
Visualizza gli attributi per un determinato oggetto delle informazioni di autenticazione.

I seguenti parametri del gestore code supportano SSL:

SSLCRLNL
L'attributo SSLCRLNL specifica un elenco dei nomi degli oggetti delle informazioni di autenticazione che vengono utilizzati per fornire le ubicazioni di revoca del certificato per consentire il controllo del certificato TLS/SSL avanzato.

SSLCRYP
Su sistemi Windows, UNIX and Linux , imposta l'attributo del gestore code SSLCryptoHardware . Questo attributo rappresenta il nome della stringa di parametri che è possibile utilizzare per configurare l'hardware crittografico presente sul sistema.

SSLEV
Determina se viene riportato un messaggio di evento SSL se un canale che utilizza SSL non riesce a stabilire una connessione SSL.

SSLFIPS
Specifica se devono essere utilizzati solo algoritmi certificati FIPS se la crittografia viene eseguita in IBM WebSphere MQ, piuttosto che nell'hardware di crittografia. Se l'hardware di crittografia è configurato, vengono utilizzati i moduli di crittografia forniti dal prodotto hardware, che potrebbero essere certificati FIPS a un determinato livello. Ciò dipende dal prodotto hardware in uso.

SSLKEYR
Su sistemi Windows,UNIX and Linux , associa un repository delle chiavi a un gestore code. Il database delle chiavi è contenuto in un database delle chiavi *GSKit* . IBM Global Security Kit (GSKit) consente di utilizzare la protezione SSL su sistemi Windows,UNIX and Linux .

SSLRKEYC
Il numero di byte da inviare e ricevere in una conversazione SSL prima che la chiave segreta venga rinegoziata. Il numero di byte include le informazioni di controllo inviate da MCA.

I seguenti parametri del canale supportano SSL:

SSLCAUTH
Definisce se IBM WebSphere MQ richiede e convalida un certificato dal client SSL.

SSLCIPH
Specifica il livello di crittografia e la funzione (CipherSpec), ad esempio NULL_MD5 o RC4_MD5_US. La CipherSpec deve corrispondere a entrambe le estremità del canale.

SSLPEER
Specifica il DN (distinguished name) (identificativo univoco) dei partner consentiti.

Questa sezione descrive i comandi `setmqaut`, `dspmqaut`, `dmpmqaut`, `rcrmqobj`, `rcdmqimge dspmqfls` per supportare l'oggetto delle informazioni di autenticazione. Inoltre, descrive il comando iKeycmd per la gestione dei certificati su sistemi UNIX and Linux e lo strumento runmqakm per la gestione dei certificati su sistemi UNIX, Linux e Windows . Consultare le seguenti sezioni:

- setmqaut
- dspmqaut
- dmpmqaut
- rcrmqobj
- rcdmqimg
- dspmqfls
- Gestione di chiavi e certificati

Per una panoramica sulla sicurezza del canale utilizzando SSL, consultare

- “Supporto IBM WebSphere MQ per SSL e TLS” a pagina 23

Per i dettagli dei comandi MQSC associati a SSL, consultare

- MODIFICA AUTHINFO
- DEFINE AUTHINFO
- DELETE AUTINFO
- VISUALIZZA AUTHINFO

Per i dettagli sui comandi PCF associati a SSL, consultare

- Modificare, copiare e creare l'oggetto delle informazioni di autenticazione
- Elimina oggetto informazioni di autenticazione
- Richiedi oggetto informazioni di autenticazione

Certificati autofirmati e certificati firmati dalla CA

È importante pianificare l'utilizzo dei certificati digitali, sia quando si sviluppa e si verifica l'applicazione, sia per il suo utilizzo in produzione. È possibile utilizzare i certificati firmati dalla CA o i certificati autofirmati, a seconda dell'utilizzo dei gestori code e delle applicazioni client.

Certificati firmati dalla CA
Per i sistemi di produzione, ottenere i certificati da una CA (Certificate Authority) attendibile. Quando ottieni un certificato da una CA esterna, paghi per il servizio.

certificati autofirmati
Mentre si sta sviluppando l'applicazione, è possibile utilizzare certificati autofirmati o certificati emessi da una CA locale, a seconda della piattaforma:

Su sistemi Windows, UNIXe Linux , è possibile utilizzare certificati autofirmati. Per istruzioni, consultare “Creazione di un certificato personale autofirmato su sistemi UNIX, Linux, and Windows” a pagina 122.

I certificati autofirmati non sono adatti per l'uso in produzione, per i motivi seguenti:

- I certificati autofirmati non possono essere revocati, il che potrebbe consentire ad un aggressore di falsificare un'identità dopo che una chiave privata è stata compromessa. Le CA possono revocare un certificato compromesso, che ne impedisce l'ulteriore utilizzo. I certificati firmati dalla CA sono quindi più sicuri da utilizzare in un ambiente di produzione, anche se i certificati autofirmati sono più convenienti per un sistema di test.
- I certificati autofirmati non scadono mai. Ciò è conveniente e sicuro in un ambiente di test, ma in un ambiente di produzione li lascia aperti a eventuali violazioni della sicurezza. Il rischio è aggravato dal fatto che i certificati autofirmati non possono essere revocati.

- Un certificato autofirmato viene utilizzato sia come certificato personale che come certificato CA root (o di ancoraggio sicuro). Un utente con un certificato personale autofirmato potrebbe essere in grado di utilizzarlo per firmare altri certificati personali. In generale, ciò non è vero per i certificati personali emessi da una CA e rappresenta un'esposizione significativa.

CipherSpecs e certificati digitali

Solo un sottoinsieme dei CipherSpecs supportati può essere utilizzato con tutti i tipi supportati di certificato digitale. È quindi necessario scegliere un CipherSpec appropriato per il certificato digitale. Allo stesso modo, se la politica di sicurezza della propria organizzazione richiede l'utilizzo di un particolare CipherSpec , è necessario ottenere un certificato digitale adatto.

Per ulteriori informazioni sulla relazione tra CipherSpecs e certificati digitali, fare riferimento a “Certificati digitali e compatibilità CipherSpec in IBM WebSphere MQ” a pagina 34

Politiche di convalida certificato

Lo standard IETF RFC 5280 specifica una serie di regole di convalida del certificato che il software dell'applicazione conforme deve implementare per prevenire attacchi di impersonificazione. Una serie di regole di convalida del certificato è nota come politica di convalida del certificato. Per ulteriori informazioni sulle politiche di convalida dei certificati in WebSphere MQ, consultare “Politiche di convalida dei certificati in IBM WebSphere MQ” a pagina 33.

Servizi di sicurezza SNA LU 6.2

SNA LU 6.2 offre la crittografia a livello di sessione, l'autenticazione a livello di sessione e l'autenticazione a livello di conversazione.

Nota: Questa raccolta di argomenti presuppone che l'utente abbia una conoscenza di base di SNA (Systems Network Architecture). L'altra documentazione a cui si fa riferimento in questa sezione contiene una breve introduzione ai concetti e alla terminologia pertinenti. Se hai bisogno di un'introduzione tecnica più completa a SNA, vedi *Systems Network Architecture Technical Overview*, GC30-3073.

SNA LU 6.2 fornisce tre servizi di sicurezza:

- Crittografia a livello di sessione
- Autenticazione a livello di sessione
- Autenticazione a livello di conversazione

Per la crittografia a livello di sessione e l'autenticazione a livello di sessione, SNA utilizza l'algoritmo *DES (Data Encryption Standard)* . L'algoritmo DES è un algoritmo di cifratura a blocchi, che utilizza una chiave simmetrica per codificare e decodificare i dati. Sia il blocco che la chiave hanno una lunghezza di 8 byte.

Crittografia a livello di sessione
La *crittografia a livello di sessione* codifica e decodifica i dati di sessione utilizzando l'algoritmo DES. Può quindi essere utilizzato per fornire un servizio di riservatezza a livello di collegamento sui canali SNA LU 6.2 .

Le LU (logical unit) possono fornire la crittografia dei dati obbligatoria (o richiesta), la crittografia dei dati selettiva o nessuna crittografia dei dati.

Su una *sessione crittografica obbligatoria*, una LU codifica tutte le unità di richiesta dati in uscita e decodifica tutte le unità di richiesta dati in ingresso.

Su una *sessione di crittografia selettiva*, una LU codifica solo le unità di richiesta dati specificate dal TP (transaction program) di invio. La LU di invio segnala che i dati sono codificati impostando un indicatore nell'intestazione della richiesta. Controllando questo indicatore, la LU ricevente può indicare quali unità di richiesta decodificare prima di trasmetterle al TP ricevente.

In una rete SNA, WebSphere MQ MCA sono programmi di transazioni. Gli MCA non richiedono la codifica per i dati che inviano. La crittografia selettiva dei dati non è pertanto un'opzione; solo la crittografia dei dati obbligatoria o nessuna crittografia dei dati è possibile in una sessione.

Per informazioni su come implementare la crittografia dei dati obbligatoria, consultare la documentazione per il sottosistema SNA. Fare riferimento alla stessa documentazione per informazioni su forme di crittografia più forti che potrebbero essere disponibili per l'utilizzo sulla piattaforma, come la crittografia Triple DES a 24 byte su z/OS.

Per informazioni più generali sulla crittografia a livello di sessione, consultare *Systems Network Architecture LU 6.2 Riferimento: protocolli peer*, SC31-6808.

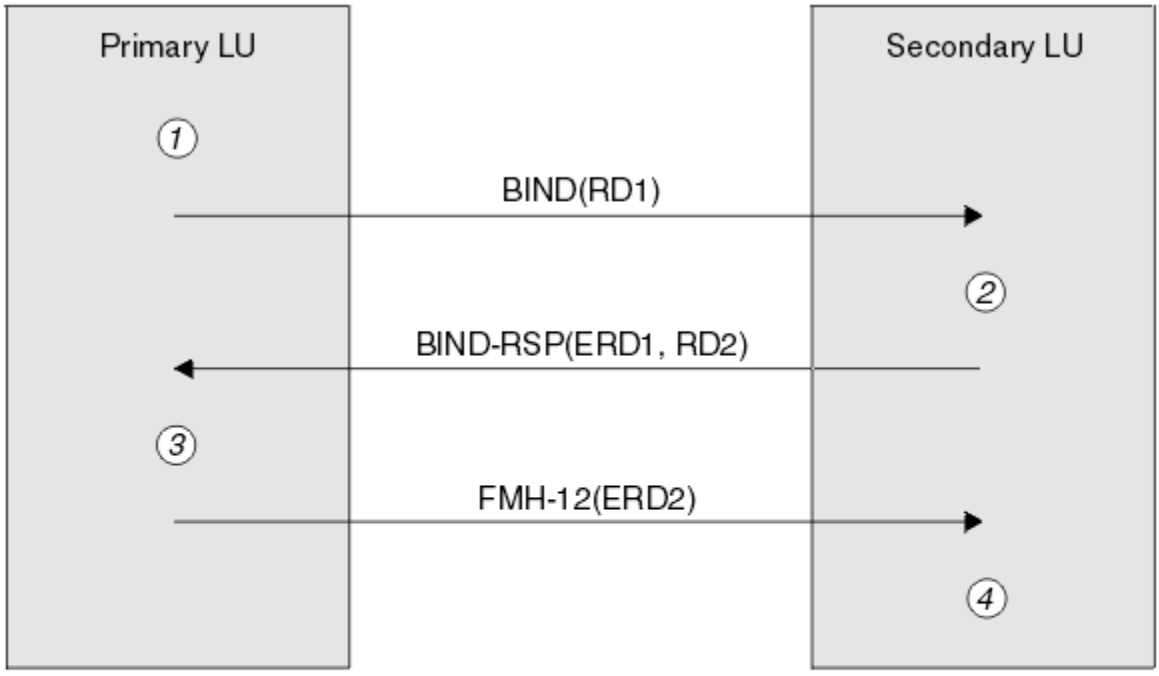
Autenticazione a livello di sessione

L' *autenticazione a livello di sessione* è un protocollo di sicurezza a livello di sessione che consente a due LU di autenticarsi reciprocamente mentre attivano una sessione. È anche noto come *Verifica LU - LU*.

Poiché una LU è effettivamente il "gateway" in un sistema dalla rete, è possibile considerare questo livello di autenticazione sufficiente in determinate circostanze. Ad esempio, se il gestore code deve scambiare i messaggi con un gestore code remoto in esecuzione in un ambiente controllato e attendibile, è possibile considerare attendibili le identità dei restanti componenti del sistema remoto dopo l'autenticazione della LU.

L'autenticazione a livello di sessione viene ottenuta da ogni LU che verifica la password del partner. La parola d'ordine viene denominata *parola d'ordine LU - LU* poiché viene stabilita una parola d'ordine tra ciascuna coppia di LU. Il modo in cui viene stabilita una parola d'ordine LU - LU dipende dall'implementazione ed è al di fuori dell'ambito di SNA.

Figura 10 a pagina 75 illustra i flussi per l'autenticazione a livello di sessione.



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. Flussi per l'autenticazione a livello di sessione

Il protocollo per l'autenticazione a livello di sessione è il seguente. I numeri nella procedura corrispondono ai numeri in Figura 10 a pagina 75.

1. La LU primaria genera un valore di dati casuale (RD1) e lo invia alla LU secondaria nella richiesta BIND.
2. Quando la LU secondaria riceve la richiesta BIND con i dati casuali, codifica i dati utilizzando l'algoritmo DES con la copia della parola d'ordine LU - LU come chiave. La LU secondaria genera quindi un secondo valore di dati casuale (RD2) e lo invia, con i dati codificati (ERD1), alla LU primaria nella risposta BIND.
3. Quando la LU primaria riceve la risposta BIND, calcola la propria versione dei dati codificati dai dati casuali generati in origine. Esegue questa operazione utilizzando l'algoritmo DES con la relativa copia della parola d'ordine LU - LU come chiave. Confronta quindi la versione con i dati crittografati ricevuti nella risposta BIND. Se i due valori sono gli stessi, la LU principale sa che la LU secondaria ha la stessa parola d'ordine e la LU secondaria è autenticata. Se i due valori non corrispondono, la LU principale termina la sessione.

   La LU primaria codifica quindi i dati casuali che ha ricevuto nella risposta BIND e invia i dati codificati (ERD2) alla LU secondaria in una Function Management Header 12 (FMH-12).
4. Quando la LU secondaria riceve FMH-12, calcola la propria versione dei dati codificati dai dati casuali che ha generato. Confronta quindi la propria versione con i dati crittografati ricevuti in FMH-12. Se i due valori sono gli stessi, la LU primaria viene autenticata. Se i due valori non corrispondono, la LU secondaria termina la sessione.

In una versione migliorata del protocollo, che fornisce una migliore protezione contro gli attacchi man in the middle, la LU secondaria calcola un MAC (DES Message Authentication Code) da RD1, RD2e il nome completo della LU secondaria, utilizzando la relativa copia della password LU - LU come chiave. La LU secondaria invia il MAC alla LU primaria nella risposta BIND invece di ERD1.

La LU primaria autentica la LU secondaria calcolando la propria versione del MAC, che confronta con il MAC ricevuto nella risposta BIND. La LU primaria calcola quindi un secondo MAC da RD1 e RD2e invia il MAC alla LU secondaria in FMH-12 invece di ERD2.

La LU secondaria autentica la LU primaria calcolando la propria versione del secondo MAC, che confronta con il MAC ricevuto in FMH-12.

Per informazioni su come configurare l'autenticazione del livello di sessione, consultare la documentazione per il sottosistema SNA. Per informazioni più generali sull'autenticazione del livello di sessione, vedi *Systems Network Architecture LU 6.2 Reference: Peer Protocols*, SC31-6808.

Autenticazione a livello di conversazione
Quando un TP locale tenta di assegnare una conversazione con un TP partner, la LU locale invia una richiesta di collegamento alla LU partner, chiedendogli di collegare il TP partner. In determinate circostanze, la richiesta di collegamento può contenere informazioni di sicurezza, che la LU partner può utilizzare per autenticare il TP locale. Questa operazione è nota come *autenticazione del livello di conversazione*o *verifica dell'utente finale*.

I seguenti argomenti descrivono in che modo IBM WebSphere MQ fornisce il supporto per l'autenticazione a livello di conversazione.

Per ulteriori informazioni sull'autenticazione del livello di conversazione, vedi *Systems Network Architecture LU 6.2 Reference: Peer Protocols*, SC31-6808. Per informazioni specifiche su z/OS, consultare *z/OS MVS Planning: APPC/MVS Management*, SA22-7599.

Per ulteriori informazioni su CPI-C, consultare *Common Programming Interface Communications CPI-C Specification*, SC31-6180. Per ulteriori informazioni su APPC/MVS TP Conversation Callable Services, consultare *z/OS MVS Programming: Writing Transaction Programs for APPC/MVS*, SA22-7621.

Supporto per l'autenticazione del livello di conversazione in IBM WebSphere MQ su UNIX e sistemi Windows
Utilizzare questo argomento per ottenere una panoramica su come funziona l'autenticazione del livello di conversazione, su UNIX, Linux, and Windows.

Il supporto per l'autenticazione a livello di conversazione in IBM WebSphere MQ per WebSphere MQ su sistemi UNIX e WebSphere MQ per Windows è illustrato in Figura 11 a pagina 77. I numeri nel diagramma corrispondono ai numeri nella descrizione che segue.

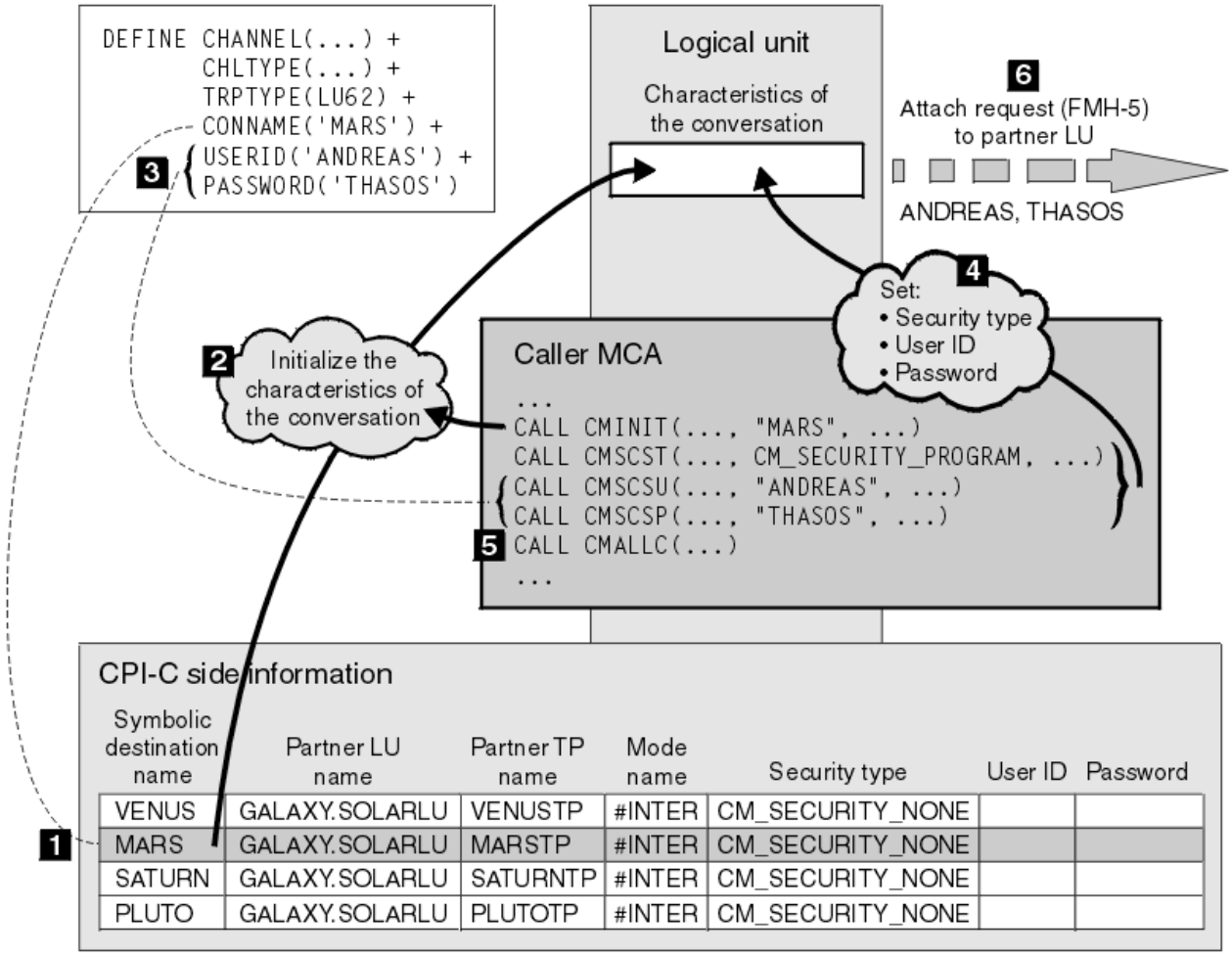


Figura 11. Supporto WebSphere MQ per l'autenticazione a livello di conversazione

Su sistemi IBM i, UNIX e Windows , un MCA utilizza chiamate CPI-C (Common Programming Interface Communications) per comunicare con un MCA partner attraverso una rete SNA. Nella definizione di canale all'estremità del chiamante di un canale, il valore del parametro CONNAME è un nome di destinazione simbolico, che identifica una voce di informazioni lato CPI-C (1). Questa voce specifica:

- Il nome della LU partner
- Il nome del TP partner, che è un MCA rispondente
- Il nome della modalità da utilizzare per la conversazione

Una voce di informazioni laterali può anche indicare le seguenti informazioni di sicurezza:

- Un tipo di sicurezza.

  I tipi di sicurezza comunemente implementati sono CM_SECURITY_NONE, CM_SECURITY_PROGRAM e CM_SECURITY_SAME, ma altri sono definiti nella specifica CPI-C.
- Un ID utente.
- Una password.

Un MCA chiamante prepara ad assegnare una conversazione con un MCA rispondente emettendo la chiamata CPI-C CMINIT, utilizzando il valore di CONNAME come uno dei parametri sulla chiamata. La chiamata CMINIT identifica, a vantaggio della LU locale, la voce di informazioni laterali che MCA intende utilizzare per la conversazione. La LU locale utilizza i valori in questa voce per inizializzare le caratteristiche della conversazione (2).

Il chiamante MCA verifica quindi i valori dei parametri USERID e PASSWORD nella definizione di canale (3). Se USERID è impostato, l'MCA del chiamante emette le seguenti chiamate CPI-C (4):

- CMSCST, per impostare il tipo di sicurezza per la conversazione su CM_SECURITY_PROGRAM.
- CMSCSU, per impostare l'ID utente per la conversazione sul valore USERID.
- CMSCSP, per impostare la password per la conversazione sul valore di PASSWORD. CMSCSP non viene richiamato a meno che PASSWORD non sia impostato.

Il tipo di sicurezza, l'ID utente e la parola d'ordine impostati da queste chiamate sovrascrivono i valori acquisiti precedentemente dalla voce di informazioni laterali.

Il chiamante MCA emette quindi la chiamata CPI-C CMALLC per allocare la conversazione (5). In risposta a questa chiamata, la LU locale invia una richiesta di collegamento (Function Management Header 5 o FMH-5) alla LU partner (6).

Se la LU partner accetta un ID utente e una password, i valori USERID e PASSWORD vengono inclusi nella richiesta di collegamento. Se la LU partner non accetterà un ID utente e una password, i valori non vengono inclusi nella richiesta di collegamento. La LU locale rileva se la LU partner accetterà un ID utente e una password come parte di uno scambio di informazioni quando le LU si collegano per formare una sessione.

In una versione successiva della richiesta di collegamento, un sostituto della password può fluire tra le LU invece di una password chiara.Un sostituto della password è un MAC (Message Authentication Code) DES o un digest del messaggio SHA-1 , formato dalla password. I sostituti della parola d'ordine possono essere utilizzati solo se entrambe le LU li supportano.

Quando la LU partner riceve una richiesta di collegamento in entrata contenente un ID utente e una parola d'ordine, potrebbe utilizzare l'ID utente e la parola d'ordine per scopi di identificazione e autenticazione. Facendo riferimento agli elenchi di controllo accessi, la LU partner potrebbe anche determinare se l'ID utente dispone dell'autorizzazione per allocare una conversazione e collegare l'MCA del responder.

Inoltre, l'MCA del risponditore potrebbe essere eseguito con l'ID utente incluso nella richiesta di collegamento.In questo caso, l'ID utente diventa l'ID utente predefinito per l'MCA del responder e viene utilizzato per i controlli delle autorizzazioni quando l'MCA tenta di collegarsi al gestore code. Potrebbe anche essere utilizzato per i controlli delle autorizzazioni successivamente quando l'MCA tenta di accedere alle risorse del gestore code.

Il modo in cui un ID utente e una parola d'ordine in una richiesta di collegamento possono essere utilizzati per l'identificazione, l'autenticazione e il controllo accessi dipende dall'implementazione. Per informazioni specifiche per il sottosistema SNA, fare riferimento alla documentazione appropriata.

Se USERID non è impostato, l'MCA del chiamante non chiama CMSCST, CMSCSU e CMSCSP. In tal caso, le informazioni di sicurezza che fluiscono in una richiesta di collegamento sono determinate esclusivamente da quanto specificato nella voce delle informazioni laterali e da quanto la LU partner accetterà.

Sicurezza per i cluster del gestore code

Sebbene i cluster di gestore code possano essere utili da utilizzare, è necessario prestare particolare attenzione alla relativa sicurezza.

Il *cluster di gestori code* è una rete di gestori code associati in modo logico. Un gestore code membro di un cluster viene denominato *gestore code cluster*.

Una coda che appartiene a un gestore code cluster può essere resa nota ad altri gestori code nel cluster. Tale coda viene denominata *coda cluster*. Qualsiasi gestore code in un cluster può inviare messaggi alle code del cluster senza che sia necessario quanto segue:

- Una definizione di coda remota esplicita per ogni coda cluster
- Canali definiti esplicitamente da e verso ogni gestore code remoto
- Una coda di trasmissione separata per ogni canale in uscita

È possibile creare un cluster in cui due o più gestori code sono cloni. Ciò significa che dispongono di istanze delle stesse code locali, incluse le code locali dichiarate come code cluster, e possono supportare istanze delle stesse applicazioni server.

Quando un'applicazione connessa a un gestore code del cluster invia un messaggio a una coda del cluster che dispone di un'istanza su ciascuno dei gestori code clonati, IBM WebSphere MQ decide a quale gestore code inviarlo. Quando molte applicazioni inviano messaggi alla coda del cluster, WebSphere MQ bilancia il workload su ciascuno dei gestori code che hanno un'istanza della coda. Se uno dei sistemi su cui si trova un gestore code clonato ha esito negativo, WebSphere MQ continua a bilanciare il workload tra i restanti gestori code fino a quando il sistema in errore non viene riavviato.

Se si utilizzano i cluster di gestori code, è necessario considerare i problemi di sicurezza riportati di seguito:

- Consentire solo ai gestori code selezionati di inviare messaggi al proprio gestore code
- Consentire solo agli utenti selezionati di un gestore code remoto di inviare messaggi a una coda sul gestore code
- Consentire alle applicazioni connesse al gestore code di inviare messaggi solo alle code remote selezionate

Queste considerazioni sono rilevanti anche se non si utilizzano i cluster, ma diventano più importanti se si utilizzano i cluster.

Se un'applicazione può inviare messaggi a una coda cluster, può inviare messaggi a qualsiasi altra coda cluster senza richiedere ulteriori definizioni di code remote, code di trasmissione o canali. Diventa quindi più importante considerare se è necessario limitare l'accesso alle code del cluster sul gestore code e limitare le code del cluster a cui le applicazioni possono inviare messaggi.

Ci sono alcune considerazioni sulla sicurezza aggiuntive, che sono rilevanti solo se si utilizzano i cluster del gestore code:

- Come consentire solo ai gestori code selezionati di unirsi a un cluster
- Forzare i gestori code indesiderati a lasciare un cluster

Per ulteriori informazioni su tutte queste considerazioni, vedi Conservazione dei cluster protetti.

Attività correlate

“Come impedire ai gestori code di ricevere messaggi” a pagina 248
È possibile evitare che un gestore code del cluster riceva messaggi che non è autorizzato a ricevere utilizzando i programmi di uscita.

Sicurezza per la pubblicazione / sottoscrizione IBM WebSphere MQ

Vi sono ulteriori considerazioni sulla sicurezza se si utilizza IBM WebSphere MQ Publish / Subscribe.

In un sistema di pubblicazione / sottoscrizione, esistono due tipi di applicazione: publisher e subscriber. I *Publisher* forniscono informazioni sotto forma di messaggi IBM WebSphere MQ . Quando un publisher pubblica un messaggio, specifica un *argomento*, che identifica l'oggetto delle informazioni all'interno del messaggio.

I *Sottoscrittori* sono i consumatori delle informazioni pubblicate. Un sottoscrittore specifica gli argomenti a cui è interessato effettuando la sottoscrizione.

Il *gestore code* è un'applicazione fornita con IBM WebSphere MQ Pubblicazione / Sottoscrizione. Riceve i messaggi pubblicati dai publisher e le richieste di sottoscrizione dai sottoscrittori e instrada i messaggi pubblicati ai sottoscrittori. A un sottoscrittore vengono inviati messaggi solo sugli argomenti per i quali ha effettuato la sottoscrizione.

Per ulteriori informazioni, consultare Sicurezza di pubblicazione / sottoscrizione.

Sicurezza multicast

Utilizzare queste informazioni per comprendere il motivo per cui i processi di sicurezza potrebbero essere necessari con IBM WebSphere MQ Multicast.

IBM WebSphere MQ Multicast non dispone di una sicurezza integrata. I controlli di sicurezza vengono gestiti nel gestore code in fase MQOPEN e l'impostazione del campo MQMD viene gestita dal client.

Alcune applicazioni nella rete potrebbero non essere applicazioni IBM WebSphere MQ (ad esempio, applicazioni LLM, consultare Interoperabilità multicast con WebSphere MQ Messaggistica a bassa latenza per ulteriori informazioni), pertanto potrebbe essere necessario implementare le proprie procedure di sicurezza poiché la ricezione di applicazioni non può essere certa della validità dei campi di contesto.

Vi sono tre processi di sicurezza da considerare:

Controllo accessi

Il controllo accessi in IBM WebSphere MQ è basato sugli ID utente. Per ulteriori informazioni su questo argomento, consultare “Controllo accessi per client” a pagina 57.

Sicurezza di rete

Una rete isolata potrebbe essere un'opzione di sicurezza valida per impedire messaggi falsi. È possibile che un'applicazione sull'indirizzo del gruppo multicast pubblichi messaggi dannosi utilizzando le funzioni di comunicazione native, che sono indistinguibili dai messaggi MQ perché provengono da un'applicazione sullo stesso indirizzo del gruppo multicast.

È anche possibile che un client sull'indirizzo del gruppo multicast riceva messaggi destinati ad altri client sullo stesso indirizzo del gruppo multicast.

L'isolamento della rete multicast garantisce l'accesso solo a client e applicazioni validi. Questa precauzione di sicurezza può evitare l'arrivo di messaggi dannosi e l'uscita di informazioni riservate.

Per informazioni sugli indirizzi di rete dei gruppi multicast, consultare: Impostazione della rete appropriata per il traffico multicast

Firme digitali

Una firma digitale è formata dalla codifica di una rappresentazione di un messaggio. La crittografia utilizza la chiave privata del firmatario e, per efficienza, di solito opera su un digest del messaggio piuttosto che sul messaggio stesso. La firma digitale di un messaggio prima di un MQPUT è una buona precauzione di sicurezza, ma questo processo potrebbe avere un effetto negativo sulle prestazioni se c'è un grande volume di messaggi.

Le firme digitali variano con i dati che vengono firmati. Se due messaggi diversi sono firmati digitalmente dalla stessa entità, le due firme differiscono, ma entrambe le firme possono essere verificate con la stessa chiave pubblica, ovvero la chiave pubblica dell'entità che ha firmato i messaggi.

Come indicato in precedenza in questa sezione, è possibile che un'applicazione sull'indirizzo del gruppo multicast pubblichi messaggi dolosi utilizzando funzioni di comunicazione native, che sono indistinguibili dai messaggi di MQ . Le firme digitali forniscono la prova di origine e solo il mittente conosce la chiave privata, il che fornisce una prova forte che il mittente è l'autore del messaggio.

Per ulteriori informazioni su questo argomento, consultare “Concetti crittografici” a pagina 7.

Firewall e pass-thru Internet

Normalmente si utilizza un firewall per evitare l'accesso da indirizzi IP ostili, ad esempio in un attacco Denial of Service. Tuttavia, potrebbe essere necessario bloccare temporaneamente gli indirizzi IP all'interno di IBM WebSphere MQ, forse mentre si attende che un amministratore della sicurezza aggiorni le regole del firewall.

Per bloccare uno o più indirizzi IP, creare un record di autenticazione di canale di tipo BLOCKADDR o ADDRESSMAP. Per ulteriori informazioni, fare riferimento a “Blocco di specifici indirizzi IP” a pagina 182.

Sicurezza per IBM WebSphere MQ Internet pass - thru

Internet pass - thru può semplificare la comunicazione attraverso un firewall, ma questo ha implicazioni di sicurezza.

IBM WebSphere MQ internet pass - thru è un'estensione del prodotto di base IBM WebSphere MQ fornita in SupportPac MS81.

WebSphere MQ Internet pass - thru consente a due gestori code di scambiarsi messaggi o a un'applicazione client WebSphere MQ per connettersi a un gestore code su Internet senza richiedere una connessione TCP/IP diretta. Ciò è utile se un firewall non consente una connessione TCP/IP diretta tra due sistemi. Rende il passaggio del protocollo del canale WebSphere MQ in entrata e in uscita da un firewall più semplice e gestibile eseguendo il tunnelling dei flussi all'interno di HTTP o agendo come un proxy. Utilizzando SSL (Secure Sockets Layer), può essere utilizzato anche per codificare e decodificare i messaggi inviati su Internet.

Quando il sistema WebSphere MQ comunica con IPT, a meno che non si stia utilizzando SSLProxyMode in IPT, assicurarsi che il CipherSpec utilizzato da WebSphere MQ corrisponda al CipherSuite utilizzato da IPT:

- Quando IPT funge da server SSL o TLS e WebSphere MQ si connette come client SSL o TLS, la CipherSpec utilizzata da WebSphere MQ deve corrispondere a una CipherSuite abilitata nel keyring IPT pertinente.
- Quando IPT funge da client SSL o TLS e si connette a un server WebSphere MQ SSL o TLS, la CipherSuite IPT deve corrispondere alla CipherSpec definita sul canale WebSphere MQ ricevente.

Se si esegue la migrazione da IPT al supporto SSL e TLS WebSphere MQ integrato, trasferire i certificati digitali da IPT utilizzando iKeyman.

Per ulteriori informazioni, consultare WebSphere MQ Internet Pass - Thru (SupportPac MS81).

Configurazione della sicurezza

Questa raccolta di argomenti contiene informazioni specifiche per i diversi sistemi operativi e per l'utilizzo dei client.

Impostazione della protezione sui sistemi UNIX, Linux, and Windows

Considerazioni sulla sicurezza specifiche per i sistemi UNIX, Linux, and Windows .

I gestori code IBM WebSphere MQ trasferiscono informazioni potenzialmente preziose, pertanto è necessario utilizzare un sistema di autorità per garantire che gli utenti non autorizzati non possano accedere ai gestori code. Considerare i seguenti tipi di controlli di sicurezza:

Chi può amministrare IBM WebSphere MQ
È possibile definire la serie di utenti che possono immettere comandi per amministrare IBM WebSphere MQ.

Chi può utilizzare gli oggetti IBM WebSphere MQ
È possibile definire quali utenti (di solito applicazioni) possono utilizzare chiamate MQI e comandi PCF per effettuare le seguenti operazioni:

- Chi può connettersi a un gestore code.
- Chi può accedere agli oggetti (code, definizioni di processi, elenchi di nomi, canali, canali di connessione client, listener, servizi e oggetti di informazioni di autenticazione) e quale tipo di accesso hanno a tali oggetti.
- Chi può accedere ai messaggi IBM WebSphere MQ .
- Chi può accedere alle informazioni di contesto associate a un messaggio.

Sicurezza canale
È necessario assicurarsi che i canali utilizzati per inviare messaggi ai sistemi remoti possano accedere alle risorse richieste.

È possibile utilizzare funzioni operative standard per concedere l'accesso alle librerie di programmi, alle librerie di collegamenti MQI e ai comandi. Tuttavia, la directory contenente le code e altri dati del gestore code è privata per IBM WebSphere MQ; non utilizzare i comandi del sistema operativo standard per concedere o revocare le autorizzazioni alle risorse MQI.

Connessione a IBM WebSphere MQ utilizzando Terminal Services

Il diritto utente **Create global objects** può causare problemi se si utilizzano i Servizi terminal.

Se ci si connette a un sistema Windows utilizzando Servizi terminal e si hanno problemi nella creazione o nell'avvio di un gestore code, ciò potrebbe essere dovuto al diritto utente, **Create global objects**, nelle versioni recenti di Windows.

Il diritto utente **Create global objects** limita gli utenti autorizzati a creare oggetti nello spazio dei nomi globale. Affinché un'applicazione possa creare un oggetto globale, deve essere in esecuzione nello spazio dei nomi globale oppure l'utente con cui è in esecuzione l'applicazione deve avere il diritto utente **Create global objects** ad esso applicato.

Gli amministratori hanno il diritto utente **Create global objects** applicato per impostazione predefinita, quindi un amministratore può creare e avviare i gestori code quando sono connessi utilizzando Servizi terminal senza modificare i diritti utente.

Se i vari metodi di gestione di WebSphere MQ non funzionano quando si utilizzano i servizi di terminale, provare a impostare il diritto utente **Create global objects** :

1. Aprire il pannello Strumenti di amministrazione:

   Windows 2003 e Windows XP
   Accedere a questo pannello utilizzando **Pannello di controllo** > **Strumenti di amministrazione**.

   Windows Vista e Windows Server 2008
   Accedere a questo pannello utilizzando **Pannello di controllo** > **Sistema e manutenzione** > **Strumenti di amministrazione**.
2. Fare doppio clic su **Criteri di protezione locale**.
3. Espandere `Local Policies`.
4. Fare clic su `User Rights Assignment`.
5. Aggiungere il nuovo utente o gruppo alla politica **Create global objects** .

Creazione e gestione dei gruppi in Windows

Queste istruzioni guidano l'utente nel processo di gestione dei gruppi su una stazione di lavoro o su una macchina server membro.

Per i controller di dominio, gli utenti e i gruppi vengono gestiti tramite Active Directory. Per ulteriori dettagli sull'utilizzo di Active Directory , fare riferimento alle istruzioni appropriate del sistema operativo.

Tutte le modifiche apportate all'appartenenza a un gruppo del principal non vengono riconosciute fino a quando il gestore code non viene riavviato o fino a quando non si immette il comando MQSC REFRESH SECURITY (o l'equivalente PCF).

Utilizzare il pannello Gestione computer per gestire utenti e gruppi. Le modifiche apportate all'utente collegato corrente potrebbero non essere effettive fino a quando l'utente non si collega di nuovo.

Windows 2003 e Windows XP
Accedere a questo pannello utilizzando **Pannello di controllo** > **Strumenti di amministrazione** > **Gestione computer**.

Windows Vista e Windows Server 2008
Accedere a questo pannello utilizzando **Pannello di controllo** > **Sistema e manutenzione** > **Strumenti di amministrazione** > **Gestione computer**.

Windows 7
Accedere a questo pannello utilizzando **Strumenti di amministrazione** > **Gestione computer**

Creazione di un gruppo in Windows

Creare un gruppo utilizzando il pannello di controllo.

Procedura

1. Aprire il pannello di controllo
2. Fare doppio clic su **Strumenti di amministrazione**.
   Si apre il pannello Strumenti di amministrazione.
3. Fare doppio clic su **Gestione computer**.
   Viene aperto il pannello Gestione computer.
4. Espandere **Utenti e gruppi locali**.
5. Fare clic con il pulsante destro del mouse su **Gruppi**e selezionare **Nuovo gruppo ...**.
   Viene visualizzato il pannello Nuovo gruppo.
6. Immettere un nome appropriato nel campo Nome gruppo, quindi fare clic su **Crea**.
7. Fai clic su **Chiudi**.

Aggiunta di un utente a un gruppo su Windows

Aggiungere un utente ad un gruppo utilizzando il pannello di controllo.

Procedura

1. Aprire il pannello di controllo
2. Fare doppio clic su **Strumenti di amministrazione**.
   Si apre il pannello Strumenti di amministrazione.
3. Fare doppio clic su **Gestione computer**.
   Viene aperto il pannello Gestione computer.
4. Dal pannello Gestione computer, espandere **Utenti e gruppi locali**.
5. Selezionare **Utenti**
6. Fare doppio clic sull'utente che si desidera aggiungere a un gruppo.
   Viene visualizzato il pannello delle proprietà utente.
7. Selezionare la scheda **Membro di** .
8. Selezionare il gruppo a cui si desidera aggiungere l'utente. Se il gruppo desiderato non è visibile:
   a) Fare clic su **Aggiungi...**.
      Viene visualizzato il pannello Seleziona gruppi.
   b) Fare clic su **Ubicazioni ...**.
      Viene visualizzato il pannello Ubicazioni.
   c) Selezionare dall'elenco l'ubicazione del gruppo a cui si desidera aggiungere l'utente e fare clic su **OK**.
   d) Immettere il nome gruppo nel campo fornito.

      In alternativa, fare clic su **Avanzate ...** e poi **Trova ora** per elencare i gruppi disponibili nell'ubicazione attualmente selezionata. Da qui, selezionare il gruppo a cui si desidera aggiungere l'utente e fare clic su **OK**.
   e) Fare clic su **OK**.
      Viene visualizzato il pannello delle proprietà utente, che mostra il gruppo aggiunto.
   f) Selezionare il gruppo.
9. Fare clic su **OK**.
   Viene visualizzato il pannello Gestione computer.

Visualizzazione di chi si trova in un gruppo su Windows

Visualizzare i membri di un gruppo utilizzando il pannello di controllo.

Procedura

1. Aprire il pannello di controllo
2. Fare doppio clic su **Strumenti di amministrazione**.
   Si apre il pannello Strumenti di amministrazione.
3. Fare doppio clic su **Gestione computer**.
   Viene aperto il pannello Gestione computer.
4. Dal pannello Gestione computer, espandere **Utenti e gruppi locali**.
5. Selezionare **Gruppi**.
6. Fare doppio clic su un gruppo. Viene visualizzato il pannello delle proprietà del gruppo.
   Viene visualizzato il pannello delle proprietà del gruppo.

Risultati

Vengono visualizzati i membri del gruppo.

Rimozione di un utente da un gruppo su Windows

Rimuovere un utente da un gruppo utilizzando il pannello di controllo.

Procedura

1. Aprire il pannello di controllo
2. Fare doppio clic su **Strumenti di amministrazione**.
   Si apre il pannello Strumenti di amministrazione.
3. Fare doppio clic su **Gestione computer**.
   Viene aperto il pannello Gestione computer.
4. Dal pannello Gestione computer, espandere **Utenti e gruppi locali**.
5. Selezionare **Utenti**.
6. Fare doppio clic sull'utente che si desidera aggiungere a un gruppo.
   Viene visualizzato il pannello delle proprietà utente.
7. Selezionare la scheda **Membro di** .
8. Selezionare il gruppo da cui si desidera rimuovere l'utente, quindi fare clic su **Rimuovi**.
9. Fare clic su **OK**.
   Viene visualizzato il pannello Gestione computer.

Risultati

L'utente è stato rimosso dal gruppo.

Creazione e gestione di gruppi su HP-UX

Su HP-UX, se non si utilizza NIS o NIS +, utilizzare SAM (System Administration Manager) per gestire i gruppi.

Creazione di un gruppo su HP-UX

Aggiunta di un utente a un gruppo utilizzando System Administration Manager

Procedura

1. Da SAM (System Administration Manager), fare doppio clic su Account per utenti e gruppi.
2. Fare doppio clic su Gruppi.
3. Selezionare Aggiungi dal menu a discesa Azioni per visualizzare il pannello Aggiungi un nuovo gruppo.
4. Immettere il nome del gruppo e selezionare gli utenti che si desidera aggiungere al gruppo.

5. Fare clic su Applica per creare il gruppo.

Risultati
A questo punto è stato creato un gruppo.

Aggiunta di un utente a un gruppo su HP-UX
Aggiungere un utente a un gruppo utilizzando System Administration Manager.

Procedura
1. Da SAM (System Administration Manager), fare doppio clic su Account per utenti e gruppi.
2. Fare doppio clic su Gruppi.
3. Evidenziare il nome del gruppo e selezionare Modifica dal menu a discesa Azioni per visualizzare il pannello Modifica un gruppo esistente.
4. Selezionare un utente che si desidera aggiungere al gruppo e fare clic su Aggiungi.
5. Se si desidera aggiungere altri utenti al gruppo, ripetere il passo 4 per ciascun utente.
6. Una volta terminata l'aggiunta di nomi all'elenco, fare clic su OK.

Risultati
È stato aggiunto un utente a un gruppo.

Visualizzazione di chi si trova in un gruppo su HP-UX
Visualizzare chi si trova in un gruppo utilizzando System Administration Manager

Procedura
1. Da SAM (System Administration Manager), fare doppio clic su Account per utenti e gruppi.
2. Fare doppio clic su Gruppi.
3. Evidenziare il nome del gruppo e selezionare Modifica dal menu a discesa Azioni per visualizzare il pannello Modifica un gruppo esistente, che visualizza un elenco di utenti del gruppo.

Risultati
Vengono visualizzati i membri del gruppo.

Rimozione di un utente da un gruppo su HP-UX
Rimuovere un utente da un gruppo utilizzando System Administration Manager.

Procedura
1. Da SAM (System Administration Manager), fare doppio clic su Account per utenti e gruppi.
2. Fare doppio clic su Gruppi.
3. Evidenziare il nome del gruppo e selezionare Modifica dal menu a discesa Azioni per visualizzare il pannello Modifica un gruppo esistente.
4. Selezionare un utente che si desidera rimuovere dal gruppo e fare clic su Rimuovi.
5. Se si desidera rimuovere altri utenti dal gruppo, ripetere il passo 4 per ciascun utente.
6. Una volta terminata la rimozione dei nomi dall'elenco, fare clic su OK.

Risultati
È stato rimosso un utente da un gruppo

Creazione e gestione di gruppi su AIX

Su AIX, se non si utilizza NIS o NIS +, utilizzare SMITTY per gestire i gruppi.

Creazione di un gruppo
Creare un gruppo utilizzando SMITTY.

Procedura

1. Da SMITTY, selezionare Sicurezza e Utenti e premere Invio.
2. Selezionare Gruppi e premere Invio.
3. Selezionare Aggiungi un Gruppo e premere Invio.
4. Immettere il nome del gruppo e i nomi degli utenti che si desidera aggiungere al gruppo, separati da virgole.
5. Premere Invio per creare il gruppo.

Risultati
A questo punto è stato creato un gruppo.

Aggiunta di un utente a un gruppo
Aggiungere un utente ad un gruppo utilizzando SMITTY.

Procedura

1. Da SMITTY, selezionare Sicurezza e Utenti e premere Invio.
2. Selezionare Gruppi e premere Invio.
3. Selezionare Modifica / Mostra caratteristiche dei gruppi e premere Invio.
4. Immettere il nome del gruppo per visualizzare un elenco dei membri del gruppo.
5. Aggiungere i nomi degli utenti che si desidera aggiungere al gruppo, separati da virgole.
6. Premere Invio per aggiungere i nomi al gruppo.

Visualizzazione di chi fa parte di un gruppo
Visualizzare chi si trova in un gruppo utilizzando SMITTY.

Procedura

1. Da SMITTY, selezionare Sicurezza e Utenti e premere Invio.
2. Selezionare Gruppi e premere Invio.
3. Selezionare Modifica / Mostra caratteristiche dei gruppi e premere Invio.
4. Immettere il nome del gruppo per visualizzare un elenco dei membri del gruppo.

Risultati
Vengono visualizzati i membri del gruppo.

Rimozione di un utente da un gruppo
Rimuovere un utente da un gruppo utilizzando SMITTY.

Procedura

1. Da SMITTY, selezionare Sicurezza e Utenti e premere Invio.
2. Selezionare Gruppi e premere Invio.
3. Selezionare Modifica / Mostra caratteristiche dei gruppi e premere Invio.
4. Immettere il nome del gruppo per visualizzare un elenco dei membri del gruppo.
5. Eliminare i nomi degli utenti che si desidera rimuovere dal gruppo.
6. Premere Invio per rimuovere i nomi dal gruppo.

Risultati
È stato rimosso un utente da un gruppo.

Creazione e gestione di gruppi su Solaris

Su Solaris, se non si utilizza NIS o NIS +, utilizzare il file `/etc/group` per gestire i gruppi.

Creazione di un gruppo su Solaris
Creazione di un gruppo utilizzando il comando **groupadd** .

Procedura

Immettere il seguente comando: `groupadd` *`group-name`*
dove *nome - gruppo* è il nome del gruppo.

Risultati
Il file `/etc/group` contiene informazioni sul gruppo.

Aggiunta di un utente a un gruppo su Solaris
Aggiungere un utente a un gruppo utilizzando il comando **usermod** .

Procedura

Per aggiungere un membro a un gruppo supplementare, eseguire il comando `usermod` ed elencare i gruppi supplementari di cui l'utente è attualmente membro e i gruppi supplementari di cui l'utente deve diventare membro.
Ad esempio, se l'utente è un membro del gruppo `groupae` deve diventare un membro anche di `groupb` , utilizzare il seguente comando: `usermod -G groupa,groupb` *`user-name`* , dove *user-name* è il nome utente.

Visualizzazione di chi fa parte di un gruppo su Solaris
Per rilevare chi è un membro di un gruppo, esaminare la voce per tale gruppo nel file `/etc/group` .

Rimozione di un utente da un gruppo su Solaris
Rimuovere un utente da un gruppo utilizzando il comando **usermod** .

Procedura

Per rimuovere un membro da un gruppo supplementare, eseguire il comando **usermod** elencando i gruppi supplementari di cui si desidera che l'utente rimanga membro.
Ad esempio, se il gruppo primario dell'utente è `users` e l'utente è anche un membro dei gruppi `mqm`, `groupa` e `groupb`, per rimuovere l'utente dal gruppo `mqm` , viene utilizzato il seguente comando: `usermod -G groupa,groupb` *`user-name`*, dove *`user-name`* è il nome utente.

Creazione e gestione di gruppi su Linux

Su Linux, se non si utilizza NIS o NIS +, utilizzare il file `/etc/group` per gestire i gruppi.

Creazione di un gruppo su Linux
Creare un gruppo utilizzando il comando **groupadd** .

Procedura

Per creare un nuovo gruppo, immettere il seguente comando: `groupadd -g` *`group-ID group-name`*
, dove *group - ID* è l'identificativo numerico del gruppo e *group - name* è il nome del gruppo.

Risultati
Il file `/etc/group` contiene informazioni sul gruppo.

Aggiunta di un utente a un gruppo su Linux
Aggiungere un utente a un gruppo utilizzando il comando **usermod** .

Procedura

Per aggiungere un membro a un gruppo supplementare, eseguire il comando `usermod` ed elencare i gruppi supplementari di cui l'utente è attualmente membro e i gruppi supplementari di cui l'utente deve diventare membro.
Ad esempio, se l'utente è membro del gruppo `groupa`e deve diventare anche membro di `groupb` , viene utilizzato il seguente comando: `usermod -G groupa,groupb` *`user-name`*
, dove *`user-name`* è il nome utente.

Visualizzazione degli utenti in un gruppo su Linux
Visualizzare chi si trova in un gruppo utilizzando il comando **getent** .

Procedura

Per visualizzare chi è membro di un gruppo, immettere il seguente comando: `getent group` *`group-name`*
, dove *nome - gruppo* è il nome del gruppo.

Rimozione di un utente da un gruppo
Rimuovere un utente da un gruppo utilizzando il comando **usermod** .

Procedura

Per rimuovere un membro da un gruppo supplementare, eseguire il comando **usermod** elencando i gruppi supplementari di cui si desidera che l'utente rimanga membro.
Ad esempio, se il gruppo principale dell'utente è `users` e l'utente è anche membro dei gruppi `mqm`, `groupa` e `groupb`, per rimuovere l'utente dal gruppo `mqm` , viene utilizzato il seguente comando: `usermod -G groupa,groupb` *`user-name`*
, dove *user-name* è il nome utente.

Funzionamento delle autorizzazioni

Le tabelle di specifiche di autorizzazione negli argomenti in questa sezione definiscono con precisione il funzionamento delle autorizzazioni e le relative limitazioni.

Le tabelle si applicano a queste situazioni:

- Applicazioni che emettono chiamate MQI
- Programmi di gestione che immettono comandi MQSC come PCF di escape
- Programmi di gestione che immettono comandi PCF

In questa sezione, le informazioni vengono presentate come una serie di tabelle che specificano quanto segue:

Azione da eseguire
Opzione MQI, comando MQSC o comando PCF.

Oggetto controllo accessi
Coda, processo, gestore code, elenco nomi, informazioni di autenticazione, canale, canale di connessione client, listener o servizio.

Autorizzazione richiesta
Espresso come costante MQZAO_.

Nelle tabelle, le costanti con prefisso MQZAO_ corrispondono alle parole chiave nell'elenco di autorizzazioni per il comando `setmqaut` per la particolare entità. Ad esempio, MQZAO_BROWSE corrisponde alla parola chiave `+browse`, MQZAO_SET_ALL_CONTEXT corrisponde alla parola chiave `+setalle` così via. Queste costanti sono definite nel file di intestazione cmqzc.h, fornito con il prodotto.

Autorizzazioni per le chiamate MQI

MQCONN, **MQOPEN**, **MQPUT1**e **MQCLOSE** potrebbero richiedere controlli di autorizzazione. Le tabelle in questo argomento riassumono le autorizzazioni necessarie per ogni chiamata.

Un'applicazione può emettere specifiche chiamate e opzioni MQI solo se all'identificativo utente con cui è in esecuzione (o alle cui autorizzazioni è in grado di presumere) è stata concessa l'autorizzazione pertinente.

Quattro chiamate MQI potrebbero richiedere verifiche di autorizzazione: **MQCONN**, **MQOPEN**, **MQPUT1**e **MQCLOSE**.

Per MQOPEN e MQPUT1, il controllo dell'autorizzazione viene eseguito sul nome dell'oggetto che si sta aprendo e non sul nome o sui nomi, come risultato dopo la risoluzione di un nome. Ad esempio, ad un'applicazione potrebbe essere concessa l'autorizzazione ad aprire una coda alias senza avere l'autorizzazione ad aprire la coda di base su cui l'alias si risolve. La regola è che il controllo viene eseguito sulla prima definizione rilevata durante il processo di risoluzione di un nome che non è un alias del gestore code, a meno che la definizione dell'alias del gestore code non venga aperta direttamente; ovvero, il suo nome viene visualizzato nel campo *ObjectName* del descrittore dell'oggetto. L'autorizzazione è sempre necessaria per l'oggetto da aprire. In alcuni casi è richiesta un'autorizzazione aggiuntiva indipendente dalla coda, ottenuta tramite un'autorizzazione per l'oggetto gestore code.

Tabella 8 a pagina 89, Tabella 9 a pagina 89, Tabella 10 a pagina 90e Tabella 11 a pagina 91 riepilogano le autorizzazioni necessarie per ogni chiamata. Nelle tabelle *Non applicabile* significa che il controllo di autorizzazione non è rilevante per questa operazione; *Nessun controllo* significa che non viene eseguito alcun controllo di autorizzazione.

Nota: In queste tabelle non vengono menzionati gli elenchi nomi, i canali, i canali di connessione client, i listener, i servizi o gli oggetti delle informazioni di autenticazione. Ciò è dovuto al fatto che nessuna delle autorizzazioni si applica a questi oggetti, ad eccezione di MQOO_INQUIRE, per il quale si applicano le stesse autorizzazioni degli altri oggetti.

L'autorizzazione speciale MQZAO_ALL_MQI include tutte le autorizzazioni nelle tabelle rilevanti per il tipo di oggetto, tranne MQZAO_DELETE e MQZAO_DISPLAY, che sono classificate come autorizzazioni di gestione.

Per modificare le opzioni di contesto del messaggio, è necessario disporre delle autorizzazioni appropriate per emettere la chiamata. Ad esempio, per utilizzare MQOO_SET_IDENTITY_CONTEXT o MQPMO_SET_IDENTITY_CONTEXT, è necessario disporre dell'autorizzazione `+setid` .

Tabella 8. Autorizzazione di sicurezza necessaria per chiamate MQCONN

Autorizzazione richiesta per:	**Oggetto coda (“1” a pagina 91)**	**Oggetto processo**	**Oggetto gestore code**
MQCONN	Non applicabile	Non applicabile	CONNECT MQZAO_

Tabella 9. Autorizzazione di sicurezza necessaria per chiamate MQOPEN

Autorizzazione richiesta per:	**Oggetto coda (“1” a pagina 91)**	**Oggetto processo**	**Oggetto gestore code**
MQOO_INQUIRE	INQUIRE MQZAO_	INQUIRE MQZAO_	INQUIRE MQZAO_
MQOO_SFOGLIA	MQZAO_BROWSE	Non applicabile	Nessun controllo
MQOO_INPUT_ *	INPUT MQZAO_	Non applicabile	Nessun controllo

Tabella 9. Autorizzazione di sicurezza necessaria per chiamate MQOPEN (Continua)			
Autorizzazione richiesta per:	**Oggetto coda (“1” a pagina 91)**	**Oggetto processo**	**Oggetto gestore code**
MQOO_SAVE_ ALL_CONTEXT **(“2” a pagina 91)**	INPUT MQZAO_	Non applicabile	Non applicabile
MQOO_OUTPUT (Coda normale) **(“3” a pagina 91)**	OUTPUT MQZAO_	Non applicabile	Non applicabile
MQOO_PASS_ IDENTITY_CONTEXT **(“4” a pagina 91)**	MQZAO_PASS_ CONTESTO_IDENTITÀXX_ ENCODE_CASE_CAPS_LO CK_OFF	Non applicabile	Nessun controllo
MQOO_PASS_ALL_ CONTEXT **(“4” a pagina 91, “5” a pagina 91)**	MQZAO_PASS _TUTTO_CONTESTO	Non applicabile	Nessun controllo
MQOO_SET_ IDENTITY_CONTEXT **(“4” a pagina 91, “5” a pagina 91)**	MQZAO_SET_ CONTESTO_IDENTITÀ	Non applicabile	MQZAO_SET_ IDENTITY_CONTEXT **(“6” a pagina 91)**
MQOO_SET_ ALL_CONTEXT **(“4” a pagina 91, “7” a pagina 91)**	MQZAO_SET_ TUTTO_CONTESTO	Non applicabile	MQZAO_SET_ ALL_CONTEXT **(“6” a pagina 91)**
MQOO_OUTPUT (Coda di trasmissione) **(“8” a pagina 91)**	MQZAO_SET_ TUTTO_CONTESTO	Non applicabile	MQZAO_SET_ ALL_CONTEXT **(“6” a pagina 91)**
SET MQOO	MQZAO_SET	Non applicabile	Nessun controllo
MQOO_ALTERNATE_ AUTORITÀ_UTENTE	**(“9” a pagina 91)**	**(“9” a pagina 91)**	MQZAO_ALTERNATE_ USER_AUTHORITY **(“9” a pagina 91, “10” a pagina 91)**

Tabella 10. Autorizzazione di sicurezza necessaria per le chiamate MQPUT1			
Autorizzazione richiesta per:	**Oggetto coda (“1” a pagina 91)**	**Oggetto processo**	**Oggetto gestore code**
MQPMO_PASS_ CONTESTO_IDENTITÀ	MQZAO_PASS_ IDENTITY_CONTEXT **(“11” a pagina 91)**	Non applicabile	Nessun controllo
MQPMO_PASS_ALL _CONTESTO	MQZAO_PASS_ ALL_CONTEXT **(“11” a pagina 91)**	Non applicabile	Nessun controllo
MQPMO_SET_ CONTESTO_IDENTITÀ	MQZAO_SET_ IDENTITY_CONTEXT **(“11” a pagina 91)**	Non applicabile	MQZAO_SET_ IDENTITY_CONTEXT **(“6” a pagina 91)**
MQPMO_SET_ TUTTO_CONTESTO	MQZAO_SET_ ALL_CONTEXT **(“11” a pagina 91)**	Non applicabile	MQZAO_SET_ ALL_CONTEXT **(“6” a pagina 91)**

Tabella 10. Autorizzazione di sicurezza necessaria per le chiamate MQPUT1 (Continua)			
Autorizzazione richiesta per:	**Oggetto coda ("1" a pagina 91)**	**Oggetto processo**	**Oggetto gestore code**
(Coda di trasmissione) **("8" a pagina 91)**	MQZAO_SET_ TUTTO_CONTESTO	Non applicabile	MQZAO_SET_ ALL_CONTEXT **("6" a pagina 91)**
AUTORIZZAZIONE_UTENTE_MQPMO_ALTERNATE_	**("12" a pagina 92)**	Non applicabile	MQZAO_ALTERNATE_ USER_AUTHORITY **("10" a pagina 91)**

Tabella 11. Autorizzazione di sicurezza necessaria per le chiamate MQCLOSE			
Autorizzazione richiesta per:	**Oggetto coda ("1" a pagina 91)**	**Oggetto processo**	**Oggetto gestore code**
MQCO_DELETE	MQZAO_DELETE **("13" a pagina 92)**	Non applicabile	Non applicabile
MQCO_DELETE _PURGE	MQZAO_DELETE **("13" a pagina 92)**	Non applicabile	Non applicabile

Note per le tabelle:

1. Se si apre una coda modello:
   - L'autorità MQZAO_DISPLAY è necessaria per la coda modello, oltre all'autorizzazione per aprire la coda modello per il tipo di accesso per cui si sta aprendo.
   - L'autorizzazione MQZAO_CREATE non è necessaria per creare la coda dinamica.
   - All'identificativo utente utilizzato per aprire la coda modello vengono automaticamente concesse tutte le autorizzazioni specifiche della coda (equivalente a MQZAO_ALL) per la coda dinamica creata.
2. È necessario specificare anche MQOO_INPUT_ *. È valido per una coda locale, modello o alias.
3. Questo controllo viene eseguito per tutti i casi di output, tranne le code di trasmissione (consultare la nota "8" a pagina 91).
4. È necessario specificare anche MQOO_OUTPUT.
5. MQOO_PASS_IDENTITY_CONTEXT è implicito anche da questa opzione.
6. Questa autorizzazione è richiesta sia per l'oggetto gestore code che per la particolare coda.
7. Anche MQOO_PASS_IDENTITY_CONTEXT, MQOO_PASS_ALL_CONTEXT e MQOO_SET_IDENTITY_CONTEXT sono impliciti in questa opzione.
8. Questo controllo viene eseguito per una coda locale o modello che ha un attributo di coda *Utilizzo* di MQUS_TRANSMISSION e viene aperto direttamente per l'output. Non si applica se una coda remota viene aperta (specificando i nomi del gestore code remoto e della coda remota o specificando il nome di una definizione locale della coda remota).
9. È necessario specificare anche almeno uno tra MQOO_INQUIRE (per qualsiasi tipo di oggetto) o MQOO_BROWSE, MQOO_INPUT_ *, MQOO_OUTPUT o MQOO_SET (per le code). Il controllo eseguito è quello per le altre opzioni specificate, utilizzando l'identificativo utente alternativo fornito per l'autorizzazione dell'oggetto con nome specifico e l'autorità dell'applicazione corrente per il controllo MQZAO_ALTERNATE_USER_IDENTIFIER.
10. Questa autorizzazione consente di specificare qualsiasi *AlternateUserId* .
11. Viene eseguito anche un controllo MQZAO_OUTPUT se la coda non dispone di un attributo della coda *Utilizzo* di MQUS_TRANSMISSION.

12. Il controllo effettuato è come per le altre opzioni specificate, utilizzando l'identificativo utente alternativo fornito per l'autorità della coda con nome specifico e l'autorità dell'applicazione corrente per il controllo MQZAO_ALTERNATE_USER_IDENTIFIER.
13. Il controllo viene eseguito solo se si verificano entrambe le seguenti condizioni:
    - Una coda dinamica permanente è in fase di chiusura ed eliminazione.
    - La coda non è stata creata dalla chiamata MQOPEN che ha restituito l'handle dell'oggetto utilizzato.

    In caso contrario, non c'è alcun controllo.

Autorizzazioni per i comandi MQSC nei PCF di escape

Queste informazioni riassumono le autorizzazioni necessarie per ogni comando MQSC contenuto in Escape PCF.

Non applicabile significa che questa operazione non è rilevante per questo tipo di oggetto.

L'ID utente con cui il programma che inoltra il comando è in esecuzione deve avere anche le seguenti autorizzazioni:

- Autorizzazione MQZAO_CONNECT per il gestore code
- Autorizzazione MQZAO_DISPLAY sul gestore code per eseguire i comandi PCF
- Autorizzazione a emettere il comando MQSC all'interno del testo del comando Escape PCF

ALTER *oggetto*

Oggetto	**Autorizzazione richiesta**
Coda	MODIFICA_MQZO
Argomento	MODIFICA_MQZO
Processo	MODIFICA_MQZO
Gestore code	MODIFICA_MQZO
Elenco nomi	MODIFICA_MQZO
Informazioni di autenticazione	MODIFICA_MQZO
Canale	MODIFICA_MQZO
Canale connessione client	MODIFICA_MQZO
Listener	MODIFICA_MQZO
Servizio	MODIFICA_MQZO
Informazioni di comunicazione	MODIFICA_MQZO

CLEAR *oggetto*

Oggetto	**Autorizzazione richiesta**
Coda	CLEAR MQZAO_
Argomento	CLEAR MQZAO_
Processo	Non applicabile
Gestore code	Non applicabile
Elenco nomi	Non applicabile
Informazioni di autenticazione	Non applicabile
Canale	Non applicabile

Oggetto	Autorizzazione richiesta
Canale connessione client	Non applicabile
Listener	Non applicabile
Servizio	Non applicabile
Informazioni di comunicazione	Non applicabile

DEFINE *oggetto* NOREPLACE (“1” a pagina 96)

Oggetto	Autorizzazione richiesta
Coda	**MQZAO_CREATE (“2” a pagina 96)**
Argomento	**MQZAO_CREATE (“2” a pagina 96)**
Processo	**MQZAO_CREATE (“2” a pagina 96)**
Gestore code	Non applicabile
Elenco nomi	**MQZAO_CREATE (“2” a pagina 96)**
Informazioni di autenticazione	**MQZAO_CREATE (“2” a pagina 96)**
Canale	**MQZAO_CREATE (“2” a pagina 96)**
Canale connessione client	**MQZAO_CREATE (“2” a pagina 96)**
Listener	**MQZAO_CREATE (“2” a pagina 96)**
Servizio	**MQZAO_CREATE (“2” a pagina 96)**
Informazioni di comunicazione	**MQZAO_CREATE (“2” a pagina 96)**

DEFINE *oggetto* REPLACE (“1” a pagina 96, “3” a pagina 97)

Oggetto	Autorizzazione richiesta
Coda	MODIFICA_MQZO
Argomento	MODIFICA_MQZO
Processo	MODIFICA_MQZO
Gestore code	Non applicabile
Elenco nomi	MODIFICA_MQZO
Informazioni di autenticazione	MODIFICA_MQZO
Canale	MODIFICA_MQZO
Canale connessione client	MODIFICA_MQZO
Listener	MODIFICA_MQZO
Servizio	MODIFICA_MQZO
Informazioni di comunicazione	MODIFICA_MQZO

DELETE *oggetto*

Oggetto	Autorizzazione richiesta
Coda	MQZAO_DELETE
Argomento	MQZAO_DELETE

Oggetto	Autorizzazione richiesta
Processo	MQZAO_DELETE
Gestore code	Non applicabile
Elenco nomi	MQZAO_DELETE
Informazioni di autenticazione	MQZAO_DELETE
Canale	MQZAO_DELETE
Canale connessione client	MQZAO_DELETE
Listener	MQZAO_DELETE
Servizio	MQZAO_DELETE
Informazioni di comunicazione	MQZAO_DELETE

VISUALIZZA *oggetto*

Oggetto	Autorizzazione richiesta
Coda	DISPLAY MQZAO_
Argomento	DISPLAY MQZAO_
Processo	DISPLAY MQZAO_
Gestore code	DISPLAY MQZAO_
Elenco nomi	DISPLAY MQZAO_
Informazioni di autenticazione	DISPLAY MQZAO_
Canale	DISPLAY MQZAO_
Canale connessione client	DISPLAY MQZAO_
Listener	DISPLAY MQZAO_
Servizio	DISPLAY MQZAO_
Informazioni di comunicazione	DISPLAY MQZAO_

START *oggetto*

Oggetto	Autorizzazione richiesta
Coda	Non applicabile
Argomento	Non applicabile
Processo	Non applicabile
Gestore code	Non applicabile
Elenco nomi	Non applicabile
Informazioni di autenticazione	Non applicabile
Canale	CONTROL MQZAO_
Canale connessione client	Non applicabile
Listener	CONTROL MQZAO_
Servizio	CONTROL MQZAO_

Oggetto	Autorizzazione richiesta
Informazioni di comunicazione	Non applicabile

STOP *oggetto*

Oggetto	Autorizzazione richiesta
Coda	Non applicabile
Argomento	Non applicabile
Processo	Non applicabile
Gestore code	Non applicabile
Elenco nomi	Non applicabile
Informazioni di autenticazione	Non applicabile
Canale	CONTROL MQZAO_
Canale connessione client	Non applicabile
Listener	CONTROL MQZAO_
Servizio	CONTROL MQZAO_
Informazioni di comunicazione	Non applicabile

Comandi per i canali

Comando	Oggetto	Autorizzazione richiesta
Ping canale	Canale	CONTROL MQZAO_
Reimpostazione canale	Canale	MQZAO_CONTROL_XX_ENCODE_CASE_ONE uscita
Risoluzione canale	Canale	MQZAO_CONTROL_XX_ENCODE_CASE_ONE uscita

Comandi sottoscrizione

Comando	Oggetto	Autorizzazione richiesta
MODIFICA SUB	Argomento	CONTROL MQZAO_
DEFINE SUB	Argomento	CONTROL MQZAO_
ELIMINA SUB	Argomento	CONTROL MQZAO_
VISUALIZZA SECONDARIO	Argomento	DISPLAY MQZAO_

Comandi sicurezza

Comando	Oggetto	Autorizzazione richiesta
SET AUTHREC	Gestore code	MODIFICA_MQZO
ELIMINA AUTHREC	Gestore code	MODIFICA_MQZO
VISUALIZZARE AUTHREC	Gestore code	DISPLAY MQZAO_
VISUALIZZA AUTHSERV	Gestore code	DISPLAY MQZAO_
VISUALIZZA ENTAUTH	Gestore code	DISPLAY MQZAO_

Comando	Oggetto	Autorizzazione richiesta
SET CHLAUTH	Gestore code	MODIFICA_MQZO
VISUALIZZA CHLAUTH	Gestore code	DISPLAY MQZAO_
Aggiorna sicurezza	Gestore code	MODIFICA_MQZO

Visualizzazioni stato

Comando	Oggetto	Autorizzazione richiesta
VISUALIZZA CHSTATUS	Gestore code	DISPLAY MQZAO_ Tenere presente che l'autorizzazione `+inq` (o in modo equivalente MQZAO_INQUIRE) è richiesta sulla coda di trasmissione se il tipo di canale è CLUSSDR.
VISUALIZZAZIONE LSSTATUS	Gestore code	DISPLAY MQZAO_
VISUALIZZA PUBSUB	Gestore code	DISPLAY MQZAO_
VISUALIZZAZIONE STATO SB	Gestore code	DISPLAY MQZAO_
VISUALIZZA SVSTATUS	Gestore code	DISPLAY MQZAO_
VISUALIZZA TPSTATUS	Gestore code	DISPLAY MQZAO_

Comandi per i Cluster

Comando	Oggetto	Autorizzazione richiesta
VISUALIZZA CLUSQMGR	Gestore code	DISPLAY MQZAO_
Aggiornamento cluster	È richiesta l'appartenenza al gruppo 'mqm'	
Reimposta cluster	È richiesta l'appartenenza al gruppo 'mqm'	
Gestore code in stato SUSPEND	È richiesta l'appartenenza al gruppo 'mqm'	
RESUME QMGR	È richiesta l'appartenenza al gruppo 'mqm'	

Altri comandi di gestione

Comando	Oggetto	Autorizzazione richiesta
QMGR PING	Gestore code	DISPLAY MQZAO_
AGGIORNA QMGR	Gestore code	MODIFICA_MQZO
RESET QMGR	Gestore code	MODIFICA_MQZO
VISUALIZZA CONN	Gestore code	DISPLAY MQZAO_
CONN STOP	Gestore code	MODIFICA_MQZO

Nota:

1. Per i comandi DEFINE, l'autorizzazione MQZAO_DISPLAY è necessaria anche per l'oggetto LIKE, se ne è specificato uno o sul SYSTEM.DEFAULT.xxx se LIKE viene omesso.
2. L'autorizzazione MQZAO_CREATE non è specifica per un particolare oggetto o tipo di oggetto. L'autorizzazione alla creazione viene concessa per tutti gli oggetti per un gestore code specificato, specificando un tipo di oggetto QMGR nel comando `setmqaut` .

3. Ciò si applica se l'oggetto da sostituire esiste già. In caso contrario, il controllo è quello per DEFINE *oggetto* NOREPLACE.

Informazioni correlate

Cluster: utilizzo delle procedure consigliate per REFRESH CLUSTER

Autorizzazioni per comandi PCF

Questa sezione riepiloga le autorizzazioni necessarie per ogni comando PCF.

Nessuna verifica indica che non viene eseguita alcuna verifica dell'autorizzazione; *Non applicabile* indica che questa operazione non è rilevante per questo tipo di oggetto.

L'ID utente con cui il programma che inoltra il comando è in esecuzione deve avere anche le seguenti autorizzazioni:

- Autorizzazione MQZAO_CONNECT per il gestore code
- Autorizzazione MQZAO_DISPLAY sul gestore code per eseguire i comandi PCF

L'autorizzazione speciale MQZAO_ALL_ADMIN include tutte le autorizzazioni nel seguente elenco che sono rilevanti per il tipo di oggetto, tranne MQZAO_CREATE, che non è specifico per un particolare oggetto o tipo di oggetto.

Modifica *oggetto*

Oggetto	**Autorizzazione richiesta**
Coda	MODIFICA_MQZO
Argomento	MODIFICA_MQZO
Processo	MODIFICA_MQZO
Gestore code	MODIFICA_MQZO
Elenco nomi	MODIFICA_MQZO
Informazioni di autenticazione	MODIFICA_MQZO
Canale	MODIFICA_MQZO
Canale connessione client	MODIFICA_MQZO
Listener	MODIFICA_MQZO
Servizio	MODIFICA_MQZO
Informazioni di comunicazione	MODIFICA_MQZO

Cancella *oggetto*

Oggetto	**Autorizzazione richiesta**
Coda	CLEAR MQZAO_
Argomento	CLEAR MQZAO_
Processo	Non applicabile
Gestore code	Non applicabile
Elenco nomi	Non applicabile
Informazioni di autenticazione	Non applicabile
Canale	Non applicabile
Canale connessione client	Non applicabile

Oggetto	Autorizzazione richiesta
Listener	Non applicabile
Servizio	Non applicabile
Informazioni di comunicazione	Non applicabile

Copia oggetto (senza sostituzione) (1)

Oggetto	Autorizzazione richiesta
Coda	MQZAO_CREATE **(2)**
Argomento	MQZAO_CREATE **(2)**
Processo	MQZAO_CREATE **(2)**
Gestore code	Non applicabile
Elenco nomi	MQZAO_CREATE **(2)**
Informazioni di autenticazione	MQZAO_CREATE **(2)**
Canale	MQZAO_CREATE **(2)**
Canale connessione client	MQZAO_CREATE **(2)**
Listener	MQZAO_CREATE **(2)**
Servizio	MQZAO_CREATE **(2)**
Informazioni di comunicazione	**MQZAO_CREATE (“2” a pagina 103)**

Copiare *oggetto* (con sostituzione) (1, 4)

Oggetto	Autorizzazione richiesta
Coda	MODIFICA_MQZO
Argomento	MODIFICA_MQZO
Processo	MODIFICA_MQZO
Gestore code	Non applicabile
Elenco nomi	MODIFICA_MQZO
Informazioni di autenticazione	MODIFICA_MQZO
Canale	MODIFICA_MQZO
Canale connessione client	MODIFICA_MQZO
Listener	MODIFICA_MQZO
Servizio	MODIFICA_MQZO
Informazioni di comunicazione	MODIFICA_MQZO

Crea oggetto (senza sostituzione) (3)

Oggetto	Autorizzazione richiesta
Coda	MQZAO_CREATE **(2)**
Argomento	MQZAO_CREATE **(2)**
Processo	MQZAO_CREATE **(2)**

Oggetto	Autorizzazione richiesta
Gestore code	Non applicabile
Elenco nomi	MQZAO_CREATE **(2)**
Informazioni di autenticazione	MQZAO_CREATE **(2)**
Canale	MQZAO_CREATE **(2)**
Canale connessione client	MQZAO_CREATE **(2)**
Listener	MQZAO_CREATE **(2)**
Servizio	MQZAO_CREATE **(2)**
Informazioni di comunicazione	MQZAO_CREATE **(2)**

Crea oggetto (con sostituzione) (3, 4)

Oggetto	Autorizzazione richiesta
Coda	MODIFICA_MQZO
Argomento	MODIFICA_MQZO
Processo	MODIFICA_MQZO
Gestore code	Non applicabile
Elenco nomi	MODIFICA_MQZO
Informazioni di autenticazione	MODIFICA_MQZO
Canale	MODIFICA_MQZO
Canale connessione client	MODIFICA_MQZO
Listener	MODIFICA_MQZO
Servizio	MODIFICA_MQZO
Informazioni di comunicazione	MODIFICA_MQZO

Elimina *oggetto*

Oggetto	Autorizzazione richiesta
Coda	MQZAO_DELETE
Argomento	MQZAO_DELETE
Processo	MQZAO_DELETE
Gestore code	Non applicabile
Elenco nomi	MQZAO_DELETE
Informazioni di autenticazione	MQZAO_DELETE
Canale	MQZAO_DELETE
Canale connessione client	MQZAO_DELETE
Listener	MQZAO_DELETE
Servizio	MQZAO_DELETE
Informazioni di comunicazione	MQZAO_DELETE

Interrogazione *oggetto*

Oggetto	**Autorizzazione richiesta**
Coda	DISPLAY MQZAO_
Argomento	DISPLAY MQZAO_
Processo	DISPLAY MQZAO_
Gestore code	DISPLAY MQZAO_
Elenco nomi	DISPLAY MQZAO_
Informazioni di autenticazione	DISPLAY MQZAO_
Canale	DISPLAY MQZAO_
Canale connessione client	DISPLAY MQZAO_
Listener	DISPLAY MQZAO_
Servizio	DISPLAY MQZAO_
Informazioni di comunicazione	DISPLAY MQZAO_

Interrogazione *oggetto* nomi

Oggetto	**Autorizzazione richiesta**
Coda	Nessun controllo
Argomento	Nessun controllo
Processo	Nessun controllo
Gestore code	Nessun controllo
Elenco nomi	Nessun controllo
Informazioni di autenticazione	Nessun controllo
Canale	Nessun controllo
Canale connessione client	Nessun controllo
Listener	Nessun controllo
Servizio	Nessun controllo
Informazioni di comunicazione	Nessun controllo

Avviare l' *oggetto*

Oggetto	**Autorizzazione richiesta**
Coda	Non applicabile
Argomento	Non applicabile
Processo	Non applicabile
Gestore code	Non applicabile
Elenco nomi	Non applicabile
Informazioni di autenticazione	Non applicabile
Canale	CONTROL MQZAO_

Oggetto	Autorizzazione richiesta
Canale connessione client	Non applicabile
Listener	CONTROL MQZAO_
Servizio	CONTROL MQZAO_
Informazioni di comunicazione	Non applicabile

Arresta *oggetto*

Oggetto	Autorizzazione richiesta
Coda	Non applicabile
Argomento	Non applicabile
Processo	Non applicabile
Gestore code	Non applicabile
Elenco nomi	Non applicabile
Informazioni di autenticazione	Non applicabile
Canale	CONTROL MQZAO_
Canale connessione client	Non applicabile
Listener	CONTROL MQZAO_
Servizio	CONTROL MQZAO_
Informazioni di comunicazione	Non applicabile

Comandi per i canali

Comando	Oggetto	Autorizzazione richiesta
Ping canale	Canale	CONTROL MQZAO_
Reimposta canale	Canale	MQZAO_CONTROL_XX_ENCODE_CASE_ONE uscita
Risolvi canale	Canale	MQZAO_CONTROL_XX_ENCODE_CASE_ONE uscita

Comandi sottoscrizione

Comando	Oggetto	Autorizzazione richiesta
Modifica sottoscrizione	Argomento	CONTROL MQZAO_
Crea sottoscrizione	Argomento	CONTROL MQZAO_
Elimina sottoscrizione	Argomento	CONTROL MQZAO_
Interroga sottoscrizione	Argomento	DISPLAY MQZAO_

Comandi sicurezza

Comando	Oggetto	Autorizzazione richiesta
Imposta record di autorizzazione	Gestore code	MODIFICA_MQZO

Comando	Oggetto	Autorizzazione richiesta
Eliminare il record di autorizzazione	Gestore code	MODIFICA_MQZO
Interrogazione record autorizzazione	Gestore code	DISPLAY MQZAO_
Interrogazione servizio autorizzazione	Gestore code	DISPLAY MQZAO_
Richiedi autorità entità	Gestore code	DISPLAY MQZAO_
Imposta record autenticazione canale	Gestore code	MODIFICA_MQZO
Interroga record autenticazione canale	Gestore code	DISPLAY MQZAO_
Aggiornamento sicurezza	Gestore code	MODIFICA_MQZO

Visualizzazioni stato

Comando	Oggetto	Autorizzazione richiesta
Interrogazione stato canale	Gestore code	DISPLAY MQZAO_ Tenere presente che l'autorizzazione `+inq` (o in modo equivalente MQZAO_INQUIRE) è richiesta sulla coda di trasmissione se il tipo di canale è CLUSSDR.
Richiedi stato listener canale	Gestore code	DISPLAY MQZAO_
Interroga stato pubblicazione / sottoscrizione	Gestore code	DISPLAY MQZAO_
Richiedi stato sottoscrizione	Gestore code	DISPLAY MQZAO_
Interrogazione stato servizio	Gestore code	DISPLAY MQZAO_
Interroga stato argomento	Gestore code	DISPLAY MQZAO_

Comandi per i Cluster

Comando	Oggetto	Autorizzazione richiesta
Interrogazione gestore code cluster	Gestore code	DISPLAY MQZAO_
Aggiornamento cluster	È richiesta l'appartenenza al gruppo 'mqm'	
Reimposta cluster	È richiesta l'appartenenza al gruppo 'mqm'	
Sospensione cluster gestore code	È richiesta l'appartenenza al gruppo 'mqm'	
Ripristino cluster gestore code	È richiesta l'appartenenza al gruppo 'mqm'	

Altri comandi di gestione

Comando	Oggetto	Autorizzazione richiesta
Ping gestore code	Gestore code	DISPLAY MQZAO_

Comando	Oggetto	Autorizzazione richiesta
Aggiornamento gestore code	Gestore code	MODIFICA_MQZO
Reimpostazione gestore code	Gestore code	MODIFICA_MQZO
Reimposta statistiche coda	Coda	MQZAO_DISPLAY e MQZAO_CHANGE
Interrogazione connessione	Gestore code	DISPLAY MQZAO_
Arresta connessione	Gestore code	MODIFICA_MQZO

Nota:

1. Per i comandi di copia, è necessaria anche l'autorizzazione MQZAO_DISPLAY per l'oggetto From.
2. L'autorizzazione MQZAO_CREATE non è specifica per un particolare oggetto o tipo di oggetto. L'autorizzazione alla creazione viene concessa per tutti gli oggetti per un gestore code specificato, specificando un tipo di oggetto QMGR nel comando `setmqaut` .
3. Per i comandi di creazione, è necessaria anche l'autorizzazione MQZAO_DISPLAY per il SISTEMA SYSTEM.DEFAULT.* dell'oggetto.
4. Ciò si applica se l'oggetto da sostituire esiste già. In caso contrario, il controllo è come per Copia o Crea senza sostituzione.

Considerazioni speciali per la protezione in Windows

Alcune funzioni di sicurezza si comportano in modo diverso su versioni differenti di Windows.

La sicurezza IBM WebSphere MQ si basa sulle chiamate all'API del sistema operativo per informazioni sulle autorizzazioni utente e sulle appartenenze ai gruppi. Alcune funzioni non funzionano in modo identico sui sistemi Windows . Questa raccolta di argomenti include descrizioni del modo in cui tali differenze potrebbero influire sulla sicurezza di IBM WebSphere MQ quando si esegue IBM WebSphere MQ in ambiente Windows .

Il programma di uscita del canale SSPI

WebSphere MQ per Windows fornisce un programma di uscita di sicurezza, che può essere utilizzato su entrambi i canali MQI e messaggi. L'uscita viene fornita come codice oggetto e origine e fornisce l'autenticazione unidirezionale e bidirezionale.

L'uscita di sicurezza utilizza SSPI (Security Support Provider Interface), che fornisce le funzionalità di protezione integrate delle piattaforme Windows .

L'uscita di sicurezza fornisce i seguenti servizi di identificazione e autenticazione:

autenticazione a una via

Utilizza il supporto di autenticazione NTLM (Windows NT LAN Manager). NTLM consente ai server di autenticare i propri client. Non consente a un client di autenticare un server o a un server di autenticarne un altro. NTLM è stato progettato per un ambiente di rete in cui si presume che i server siano originali. NTLM è supportato su tutte le piattaforme Windows supportate da WebSphere MQ Versione 7.0.

Questo servizio è generalmente utilizzato su un canale MQI per consentire a un gestore code del server di autenticare un'applicazione client WebSphere MQ MQI. Un'applicazione client viene identificata dall'ID utente associato al processo in esecuzione.

Per eseguire l'autenticazione, l'exit di sicurezza all'estremità client di un canale acquisisce un token di autenticazione da NTLM e invia il token in un messaggio di sicurezza al proprio partner all'altra estremità del canale. L'uscita di sicurezza partner passa il token a NTLM, che controlla che il token sia autentico. Se l'uscita di sicurezza del partner non è soddisfatta dell'autenticità del token, indica all'MCA di chiudere il canale.

Autenticazione a due vie o reciproca

Utilizza i servizi di autenticazione Kerberos . Il protocollo Kerberos non presuppone che i server in un ambiente di rete siano originali. I server possono autenticare i client e altri server e i client possono autenticare i server. Kerberos è supportato su tutte le piattaforme Windows supportate da WebSphere MQ Versione 7.0.

Questo servizio può essere utilizzato su entrambi i canali MQI e messaggi. Su un canale di messaggi, fornisce l'autenticazione reciproca dei due gestori code. Su un canale MQI, abilita il gestore code del server e l'applicazione client WebSphere MQ MQI per l'autenticazione reciproca. Un gestore code è identificato dal suo nome preceduto dalla stringa `ibmMQSeries/`. Un'applicazione client viene identificata dall'ID utente associato al processo in esecuzione.

Per eseguire l'autenticazione reciproca, l'uscita di sicurezza iniziale acquisisce un token di autenticazione dal server di protezione Kerberos e invia il token in un messaggio di sicurezza al partner. L'uscita di sicurezza del partner passa il token al server Kerberos , che controlla che sia autentico. Il server di sicurezza Kerberos genera un secondo token, che il partner invia in un messaggio di sicurezza all'uscita di sicurezza di avvio. L'uscita di sicurezza iniziale richiede quindi al server Kerberos di controllare che il secondo token sia autentico. Durante questo scambio, se l'uscita di sicurezza non è soddisfatta dell'autenticità del token inviato dall'altro, indica all'MCA di chiudere il canale.

L'uscita di sicurezza viene fornita sia in formato origine che in formato oggetto. È possibile utilizzare il codice sorgente come punto di partenza per la scrittura dei propri programmi di uscita del canale oppure è possibile utilizzare il modulo oggetto come fornito. Il modulo oggetto dispone di due punti di ingresso, uno per l'autenticazione a una via utilizzando il supporto di autenticazione NTLM e l'altro per l'autenticazione a due vie utilizzando i servizi di autenticazione Kerberos .

Per ulteriori informazioni su come funziona il programma di uscita del canale SSPI e per istruzioni su come implementarlo, consultare Utilizzo dell'uscita di sicurezza SSPI sui sistemi Windows.

Quando si riceve un errore 'gruppo non trovato' in Windows

Questo problema può verificarsi perché WebSphere MQ perde l'accesso al gruppo mqm locale quando i server Windows vengono promossi o retrocessi da controller di dominio. Per risolvere questo problema, creare nuovamente il gruppo mqm locale.

Il sintomo è un errore che indica la mancanza di un gruppo mqm locale, ad esempio:

```
>crtmqm qm0
AMQ8066:Local mqm group not found.
```

La modifica dello stato di una macchina tra server e controller di dominio può influire sul funzionamento di WebSphere MQ, poiché WebSphere MQ utilizza un gruppo mqm definito localmente. Quando un server viene promosso a controller di dominio, l'ambito cambia da locale a locale di dominio. Quando la macchina viene retrocessa al server, tutti i gruppi locali del dominio vengono rimossi. Ciò significa che la modifica di una macchina da server a controller di dominio e di nuovo a server perde l'accesso a un gruppo mqm locale.

Per risolvere questo problema, creare nuovamente il gruppo mqm locale utilizzando gli strumenti di gestione Windows standard. Poiché tutte le informazioni di appartenenza al gruppo vengono perse, è necessario ripristinare gli utenti con privilegi WebSphere MQ nel gruppo mqm locale appena creato. Se la macchina è un membro del dominio, è necessario aggiungere anche il gruppo mqm del dominio al gruppo mqm locale per concedere agli ID utente WebSphere MQ del dominio privilegiato il livello di autorizzazione richiesto.

In caso di problemi con IBM WebSphere MQ e controller di dominio su Windows

Alcuni problemi possono verificarsi con le impostazioni di sicurezza quando i server Windows vengono promossi a controller di dominio.

Quando si promuovono i server Windows 2000, Windows 2003 o Windows Server 2008 ai controller di dominio, viene visualizzata l'opzione di selezione di un'impostazione di sicurezza predefinita o non predefinita relativa alle autorizzazioni di utenti e gruppi. Questa opzione controlla se gli utenti arbitrari sono in grado di richiamare le appartenenze ai gruppi da Active Directory. Dal momento che WebSphere

MQ si basa sulle informazioni di appartenenza al gruppo per implementare la propria politica di sicurezza, è importante che l'ID utente che sta eseguendo le operazioni WebSphere MQ possa determinare le appartenenze al gruppo di altri utenti.

In Windows 2000, quando un dominio viene creato utilizzando l'opzione di sicurezza predefinita, l'ID utente predefinito creato da WebSphere MQ durante il processo di installazione può ottenere le appartenenze ai gruppi per altri utenti, come richiesto. Il prodotto viene quindi installato normalmente, creando oggetti predefiniti e il gestore code può determinare l'autorizzazione di accesso degli utenti locali e di dominio, se necessario.

Su Windows 2000, quando un dominio viene creato utilizzando l'opzione di protezione non predefinita, oppure su Windows 2003 e Windows Server 2008 quando un dominio viene creato utilizzando l'opzione di sicurezza predefinita, l'ID utente creato da WebSphere MQ durante l'installazione non può sempre determinare le appartenenze al gruppo richieste. In questo caso, è necessario conoscere:

- Modalità di funzionamento di Windows 2000 con non predefinito o Windows 2003 e Windows Server 2008 con predefinito, autorizzazioni di sicurezza
- Come consentire ai membri del gruppo mqm del dominio di leggere l'appartenenza al gruppo
- Modalità di configurazione di un servizio IBM WebSphere MQ Windows da eseguire sotto un utente del dominio

Windows 2000 con dominio non predefinito o Windows 2003 e Windows Server 2008 con autorizzazioni di sicurezza predefinite
L'installazione di WebSphere MQ si comporta in modo diverso su questi sistemi operativi a seconda che un utente locale o un utente di dominio esegua l'installazione.

Se un utente **locale** installa WebSphere MQ, la procedura guidata Prepara WebSphere MQ rileva che l'utente locale creato per il servizio IBM WebSphere MQ Windows può richiamare le informazioni di appartenenza al gruppo dell'utente di installazione. La procedura guidata Prepara WebSphere MQ richiede all'utente informazioni sulla configurazione di rete per determinare se sono presenti altri account utente definiti sui controller di dominio in esecuzione su Windows 2000 o versioni successive. In tal caso, il servizio IBM WebSphere MQ Windows deve essere eseguito con un account utente del dominio con impostazioni e autorizzazioni particolari. La procedura guidata Prepara WebSphere MQ richiede all'utente i dettagli dell'account di questo utente. La sua guida in linea fornisce i dettagli dell'account utente del dominio richiesto che può essere inviato all'amministratore del dominio.

Se un utente di **dominio** installa WebSphere MQ, la procedura guidata Prepara WebSphere MQ rileva che l'utente locale creato per il servizio IBM WebSphere MQ Windows non può richiamare le informazioni di appartenenza al gruppo dell'utente di installazione. In questo caso, la procedura guidata Prepara WebSphere MQ richiede sempre all'utente i dettagli dell'account utente del dominio da utilizzare per il servizio IBM WebSphere MQ Windows .

Quando il servizio IBM WebSphere MQ Windows deve utilizzare un account utente del dominio, WebSphere MQ non può funzionare correttamente fino a quando non viene configurato utilizzando la procedura guidata Prepara WebSphere MQ . La procedura guidata Prepara WebSphere MQ non consente all'utente di continuare con altre attività, fino a quando il servizio Windows non è stato configurato con un account appropriato.

Se un dominio Windows 2000 è stato configurato con autorizzazioni di sicurezza non predefinite, la soluzione usuale per abilitare WebSphere MQ al corretto funzionamento consiste nel configurarlo con un account utente di dominio appropriato, come descritto in precedenza.

Per ulteriori informazioni, consultare Creazione e impostazione di account di dominio per WebSphere MQ WebSphere MQ .

Configurazione dei servizi IBM WebSphere MQ per l'esecuzione in un utente di dominio su Windows
Utilizzare la procedura guidata Prepara IBM WebSphere MQ per immettere i dettagli dell'account utente del dominio. In alternativa, è possibile utilizzare il pannello Gestione computer per modificare i dettagli di **Accesso** per il servizio IBM WebSphere MQ specifico dell'installazione.

Per ulteriori informazioni, consultare Modifica della password dell'account utente del servizio IBM WebSphere MQ Windows

Applicazione dei file del modello di protezione a Windows

L'applicazione di un modello potrebbe influenzare le impostazioni di sicurezza applicate ai file e alle directory di WebSphere MQ . Se si utilizza il modello altamente sicuro, applicarlo prima di installare WebSphere MQ.

Windows supporta i file di modello di sicurezza basati su testo che è possibile utilizzare per applicare impostazioni di sicurezza uniformi a uno o più computer con lo snap-in MMC di configurazione e analisi della sicurezza. In particolare, Windows fornisce diversi modelli che includono una serie di impostazioni di sicurezza con l'obiettivo di fornire livelli specifici di sicurezza. Questi modelli includono Compatibile, Sicuro e Altamente Sicuro.

L'applicazione di uno di questi modelli potrebbe influenzare le impostazioni di sicurezza applicate ai file e alle directory WebSphere MQ . Se si desidera utilizzare il modello Highly Secure, configurare la macchina prima di installare WebSphere MQ.

Se si applica il modello altamente sicuro a una macchina su cui è già installato WebSphere MQ , tutte le autorizzazioni impostate sui file e le directory WebSphere MQ vengono rimosse. Poiché queste autorizzazioni vengono rimosse, si perde l'accesso *Amministratore*, *mqm*e, quando applicabile, il gruppo *Tutti* dalle directory degli errori.

Gruppi nidificati

Esistono restrizioni sull'utilizzo dei gruppi nidificati. Questi risultati derivano in parte dal livello funzionale del dominio e in parte dalle limitazioni di WebSphere MQ .

Active Directory può supportare diversi tipi di gruppi all'interno di un contesto di dominio a seconda del livello funzionale del dominio. Per impostazione predefinita, i domini Windows 2003 si trovano nel livello funzionale *Windows 2000 misto* . (Windows server 2003, Windows XP, Windows Vista e Windows Server 2008 seguono tutti il modello di dominio di Windows 2003.) Il livello funzionale del dominio determina i tipi di gruppo supportati e il livello di nidificazione consentiti durante la configurazione degli ID utente in un ambiente di dominio. Fare riferimento alla documentazione Active Directory per dettagli sull'ambito del gruppo e sui criteri di inclusione.

Oltre ai requisiti di Active Directory , vengono imposte ulteriori limitazioni agli ID utilizzati da WebSphere MQ. Le API di rete utilizzate da WebSphere MQ non supportano tutte le configurazioni supportate dal livello funzionale del dominio. Di conseguenza, WebSphere MQ non è in grado di interrogare le appartenenze al gruppo di ID dominio presenti in un gruppo locale di dominio che viene quindi nidificato in un gruppo locale. Inoltre, la nidificazione multipla di gruppi globali e universali non è supportata. Tuttavia, sono supportati i gruppi globali o universali immediatamente nidificati.

Configurazione dell'autorizzazione aggiuntiva per le applicazioni Windows che si collegano a IBM WebSphere MQ

L'account con cui vengono eseguiti i processi IBM WebSphere MQ potrebbe richiedere ulteriori autorizzazioni prima di poter concedere l'accesso SYNCHRONIZE ai processi dell'applicazione.

Potrebbero verificarsi problemi se si dispone di applicazioni Windows , ad esempio pagine ASP, che si collegano a IBM WebSphere MQ configurate per essere eseguite ad un livello di sicurezza superiore al normale.

IBM WebSphere MQ richiede l'accesso SYNCHRONIZE ai processi dell'applicazione per coordinare alcune azioni. APAR IC35116 modificato IBM WebSphere MQ in modo che siano specificati i privilegi appropriati. Tuttavia, l'account con cui vengono eseguiti i processi IBM WebSphere MQ potrebbe richiedere un'ulteriore autorizzazione prima di poter concedere l'accesso richiesto.

Quando un'applicazione server tenta per la prima volta di connettersi a un gestore code, IBM WebSphere MQ modificherà il processo per concedere l'autorizzazione SYNCHRONIZE agli amministratori IBM WebSphere MQ . Per configurare l'autorizzazione aggiuntiva all'ID utente con cui sono in esecuzione i processi IBM WebSphere MQ , completare la seguente procedura:

1. Avviare lo strumento Criteri di sicurezza locali, fare clic su Impostazioni di sicurezza -> Criteri locali -> Assegnazioni diritti utente, fare clic su "Programmi di debug".

2. Fare doppio clic sul pulsante "Debug programmi", quindi aggiungere l'ID utente IBM WebSphere MQ all'elenco

Se il sistema si trova in un dominio Windows e l'impostazione della politica effettiva non è ancora impostata, anche se è impostata l'impostazione della politica locale, l'ID utente deve essere autorizzato allo stesso modo a livello di dominio, utilizzando lo strumento della politica di sicurezza del dominio.

Impostazione della sicurezza su HP Integrity NonStop Server

Considerazioni sulla sicurezza specifiche dei sistemi HP Integrity NonStop Server .

Il client IBM WebSphere MQ per HP Integrity NonStop Server supporta sia i protocolli TLS (Transport Layer Security) che SSL (Secure Sockets Layer) per fornire la sicurezza a livello di link durante la connessione a un gestore code. Questi protocolli vengono supportati utilizzando un'implementazione di OpenSSL. OpenSSL richiede un'origine di dati casuali per fornire operazioni crittografiche forti.

OpenSSL

Panoramica sulla protezione OpenSSL per il client IBM WebSphere MQ per HP Integrity NonStop Server.

Il toolkit OpenSSL è un'implementazione open source dei protocolli SSL (Secure Sockets Layer) e TLS (Transport Layer Security) per comunicazioni sicure su una rete.

Il toolkit è sviluppato dal progetto OpenSSL . Per ulteriori informazioni sul progetto OpenSSL , consultare https://www.openssl.org. IBM WebSphere MQ client for HP Integrity NonStop Server contiene le versioni modificate delle librerie OpenSSL e il comando **openssl** . Le librerie e il comando **openssl** vengono trasferiti dal toolkit OpenSSL 1.0.1ce vengono forniti solo come codice oggetto. Non viene fornito alcun codice sorgente.

Le librerie OpenSSL vengono caricate dai programmi applicativi client IBM WebSphere MQ in modo dinamico come richiesto. Solo le librerie OpenSSL fornite da IBM WebSphere MQ sono supportate per l'utilizzo con applicazioni client IBM WebSphere MQ .

Il comando **openssl** , che è possibile utilizzare per la gestione dei certificati, è installato nella directory OSS `opt_installation_path/opt/mqm/bin`.

Utilizzando il comando **openssl** , è possibile creare e gestire chiavi e certificati digitali con vari formati di dati comuni ed eseguire semplici attività della CA (Certificate Authority).

Il formato predefinito per i dati chiave e certificato elaborati da OpenSSL è il formato PEM (Privacy Enhanced Mail). I dati in formato PEM sono dati ASCII con codifica base64 . I dati possono quindi essere trasferiti utilizzando sistemi basati su testo come email, e possono essere tagliati e incollati utilizzando editor di testo e browser web. PEM è uno standard Internet per gli scambi crittografici basati su testo ed è specificato in Internet RFC 1421, 1422, 1423 e 1424. IBM WebSphere MQ presuppone che un file con estensione `.pem` contenga dati in formato PEM. Un file in formato PEM può contenere più certificati e altri oggetti codificati e può includere commenti.

Il supporto IBM WebSphere MQ SSL su altri sistemi operativi potrebbe richiedere la codifica dei dati di chiave e certificato nei file utilizzando DER (Distinguished Encoding Rules). DER è un insieme di regole di codifica per l'utilizzo della notazione ASN.1 nelle comunicazioni protette. I dati codificati utilizzando DER sono dati binari e il formato dei dati chiave e certificato codificati utilizzando DER è noto anche come PKCS#12 o PFX. Un file che contiene questi dati generalmente ha estensione `.p12` o `.pfx`. Il comando **openssl** può essere convertito tra il formato PEM e PKCS#12 .

Daemon entropia

OpenSSL richiede un'origine di dati casuali per fornire operazioni crittografiche forti. La generazione di numeri casuali è una funzionalità generalmente fornita dal sistema operativo o da un processo daemon a livello di sistema. Il sistema operativo HP Integrity NonStop Server non fornisce questa capacità all'interno del sistema operativo.

Quando si utilizza il supporto SSL e TLS fornito con il client IBM WebSphere MQ per HP Integrity NonStop Server, è necessario un processo denominato daemon entropy per fornire l'origine dei dati casuali.

Quando si avvia un canale client che richiede SSL o TLS, OpenSSL prevede che un daemon entropy sia in esecuzione e fornisca i relativi servizi su un socket nel file system OSS in `/etc/egd-pool`.

Un daemon entropy non è fornito dal client IBM WebSphere MQ per HP Integrity NonStop Server. Il client di IBM WebSphere MQ per HP Integrity NonStop Server viene verificato con i seguenti daemon di entropia:

- amqjkdm0 (come fornito da IBM WebSphere MQ 5.3 server)
- `/usr/local/bin/prngd` (Versione 0.9.27, come fornita da HP Integrity NonStop Server Open Source Technical Library)

Impostazione della sicurezza client IBM WebSphere MQ MQI

È necessario considerare la sicurezza del client IBM WebSphere MQ MQI, in modo che le applicazioni client non abbiano accesso illimitato alle risorse sul server.

Quando si esegue un'applicazione client, non eseguire l'applicazione utilizzando un ID utente che dispone di più diritti di accesso del necessario; ad esempio, un utente nel gruppo `mqm` o anche l'utente `mqm` stesso.

Eseguendo un'applicazione come utente con troppi diritti di accesso, si corre il rischio che l'applicazione acceda e modifichi parti del gestore code, per caso o in modo doloso.

La sicurezza tra un'applicazione client e il relativo server del gestore code presenta due aspetti: l'autenticazione e il controllo accessi.

- L'autenticazione può essere utilizzata per garantire che l'applicazione del client, in esecuzione come utente specifico, sia chi dicono di essere. Utilizzando l'autenticazione è possibile evitare che un aggressore ottenga l'accesso al gestore code impersonando una delle applicazioni.

  È necessario utilizzare l'autenticazione reciproca all'interno di SSL o TLS. Per ulteriori informazioni, consultare “Utilizzo di SSL o TLS” a pagina 111
- Il controllo accessi può essere utilizzato per fornire o rimuovere i diritti di accesso per un utente o un gruppo specifico di utenti. Eseguendo un'applicazione client con un utente creato in modo specifico (o un utente in un gruppo specifico), è possibile utilizzare i controlli di accesso per garantire che l'applicazione non possa accedere a parti del gestore code che l'applicazione non dovrebbe accedere.

  Quando si imposta il controllo accessi, è necessario considerare le regole di autenticazione del canale e il campo MCAUSER su un canale. Entrambe queste funzioni hanno la possibilità di modificare l'ID utente utilizzato per verificare i diritti di controllo accessi.

  Per ulteriori informazioni sul controllo accessi, consultare “Autorizzazione dell'accesso agli oggetti” a pagina 159.

Se è stata configurata un'applicazione client per connettersi a un canale specifico con un ID limitato, ma il canale ha un ID amministratore impostato nel relativo campo MCAUSER, se l'applicazione client si connette correttamente, l'ID amministratore viene utilizzato per le verifiche del controllo accessi. Pertanto, l'applicazione client disporrà dei diritti di accesso completi per il gestore code.

Per ulteriori informazioni sull'attributo MCAUSER, consultare “Associazione di un ID utente asserito dal client ad un ID utente MCAUSER” a pagina 185.

Le regole di autenticazione di canale possono essere utilizzate anche come metodo per controllare l'accesso a un gestore code, impostando regole specifiche e criteri per una connessione da accettare.

Per ulteriori informazioni sulle regole di autenticazione del canale, consultare: “Record di autenticazione di canale” a pagina 39.

Specifica che solo i CipherSpecs certificati FIPS vengono utilizzati al runtime sul client MQI

Creare i repository delle chiavi utilizzando il software conforme a FIPS, quindi specificare che il canale deve utilizzare CipherSpecscertificati FIPS.

Per essere compatibili con FIPS in fase di runtime, i repository delle chiavi devono essere stati creati e gestiti utilizzando solo software compatibile con FIPS, come runmqakm con l'opzione -fips.

È possibile specificare che un canale SSL o TLS deve utilizzare solo CipherSpecs certificati FIPS in tre modi, elencati in ordine di precedenza:

1. Impostare il campo FipsRequired nella struttura MQSCO su MQSSL_FIPS_YES.
2. Impostare la variabile di ambiente MQSSLFIPS su YES.
3. Impostare l'attributo SSLFipsRequired nel file di configurazione client su YES.

Per impostazione predefinita, CipherSpecs con certificazione FIPS non è richiesto.

Questi valori hanno gli stessi significati dei valori di parametro equivalenti su ALTER QMGR SSLFIPS (vedere ALTER QMGR). Se il processo client attualmente non ha connessioni SSL o TLS attive e un valore FipsRequired è specificato in modo valido su un SSL MQCONNX, tutte le connessioni SSL successive associate a questo processo devono utilizzare solo i CipherSpecs associati a tale valore. Ciò si applica fino a quando questa e tutte le altre connessioni SSL o TLS non vengono arrestate, a questo punto un MQCONNX successivo può fornire un nuovo valore per FipsRequired.

Se l'hardware di crittografia è presente, i moduli di crittografia utilizzati da WebSphere MQ possono essere configurati in modo da essere quei moduli forniti dal prodotto hardware e potrebbero essere certificati FIPS ad un particolare livello. I moduli configurabili e se sono certificati FIPS dipendono dal prodotto hardware in uso.

Laddove possibile, se è configurato CipherSpecs solo FIPS, il client MQI rifiuta le connessioni che specificano una CipherSpec non FIPS con MQRC_SSL_INITIALIZATION_ERROR. WebSphere MQ non garantisce il rifiuto di tutte queste connessioni ed è responsabilità dell'utente determinare se la configurazione di WebSphere MQ è conforme a FIPS.

Concetti correlati

“FIPS (Federal Information Processing Standards) per UNIX, Linuxe Windows” a pagina 26
Quando la codifica è richiesta su un canale SSL o TLS su sistemi Windows, UNIX and Linux , WebSphere MQ utilizza un package di crittografia denominato IBM Crypto for C (ICC). Sulle piattaforme Windows, UNIX and Linux , il software ICC ha passato il programma di convalida crittografico FIPS (Federal Information Processing Standards) dello US National Institute of Standards and Technology, al livello 140-2.

Stanza SSL del file di configurazione client

Riferimenti correlati

FipsRequired (MQLONG)

FIPS MQSSL

Esecuzione di applicazioni client SSL o TLS con più installazioni di GSKit V8.0 su AIX

Le applicazioni client SSL o TLS su AIX potrebbero riscontrare MQRC_CHANNEL_CONFIG_ERROR ed errore AMQ6175 durante l'esecuzione su sistemi AIX con più installazioni GSKit V8.0 .

Quando si eseguono applicazioni client su un sistema AIX con più installazioni GSKit V8.0 , le chiamate di connessione client possono restituire MQRC_CHANNEL_CONFIG_ERROR quando si utilizza SSL o TLS. I log /var/mqm/errors registrano l'errore AMQ6175 e AMQ9220 per l'applicazione client in errore, ad esempio:

```
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 comune di questo errore è che l'impostazione della variabile di ambiente LIBPATH o LD_LIBRARY_PATH ha causato il caricamento da parte del client IBM WebSphere MQ di una serie mista di librerie da due diverse installazioni di GSKit V8.0 . L'esecuzione di un'applicazione clientIBM WebSphere MQ in un ambiente Db2 può causare questo errore.

Per evitare questo errore, includere le directory della libreria IBM WebSphere MQ all'inizio del percorso della libreria in modo che le librerie IBM WebSphere MQ abbiano la precedenza. Ciò può essere ottenuto utilizzando il comando **setmqenv** con il parametro **-k** , ad esempio:

```
. /usr/mqm/bin/setmqenv -s -k
```

Per ulteriori informazioni sull'utilizzo del comando **setmqenv** , fare riferimento a setmqenv (set WebSphere MQ environment)

Impostazione delle comunicazioni per SSL o TLS su sistemi UNIX, Linux, and Windows

Le comunicazioni sicure che utilizzano i protocolli di sicurezza crittografica SSL o TLS richiedono l'impostazione dei canali di comunicazione e la gestione dei certificati digitali che verranno utilizzati per l'autenticazione.

Per configurare l'installazione SSL o TLS è necessario definire i canali per utilizzare SSL o TLS. È inoltre necessario creare e gestire i certificati digitali. Su sistemi UNIX, Linux e Windows , è possibile eseguire i test con certificati autofirmati.

I certificati autofirmati non possono essere revocati, il che potrebbe consentire a un aggressore di falsificare un'identità dopo che una chiave privata è stata compromessa. Le CA possono revocare un certificato compromesso, che ne impedisce l'ulteriore utilizzo. I certificati firmati dalla CA sono quindi più sicuri da utilizzare in un ambiente di produzione, anche se i certificati autofirmati sono più convenienti per un sistema di test.

Per informazioni complete sulla creazione e la gestione dei certificati, consultare “Utilizzo di SSL o TLS su sistemi UNIX, Linux, and Windows” a pagina 114.

Questa raccolta di argomenti introduce alcune delle attività coinvolte nella configurazione delle comunicazioni SSL e fornisce istruzioni dettagliate sul completamento di queste attività.

Si potrebbe anche voler verificare l'autenticazione client SSL o TLS, che sono una parte facoltativa dei protocolli. Durante l'handshake SSL o TLS, il client SSL o TLS ottiene e convalida sempre un certificato

digitale dal server. Con l'implementazione IBM WebSphere MQ , il server SSL o TLS richiede sempre un certificato dal client.

Sui sistemi UNIX, Linux e Windows , il client SSL o TLS invia un certificato solo se ne ha uno con etichetta nel formato IBM WebSphere MQ corretto:

- Per un gestore code, il formato è `ibmwebspheremq` seguito dal nome del gestore code modificato in minuscolo. Ad esempio, per `QM1`, `ibmwebspheremqqm1`
- Per un client IBM WebSphere MQ , `ibmwebspheremq` seguito dall'ID utente di collegamento modificato in minuscolo, ad esempio `ibmwebspheremqmyuserid`.

IBM WebSphere MQ utilizza il prefisso `ibmwebspheremq` su un'etichetta per evitare confusione con i certificati per altri prodotti. Assicurarsi di specificare l'intera etichetta del certificato in minuscolo.

Il server SSL o TLS convalida sempre il certificato client, se ne viene inviato uno. Se il client non invia un certificato, l'autenticazione non riesce solo se la fine del canale che funge da server SSL o TLS è definita con il parametro SSLCAUTH impostato su REQUIRED o con un valore di parametro SSLPEER impostato. Per ulteriori informazioni, consultare “Connessione di due gestori code mediante SSL o TLS” a pagina 206.

Utilizzo di SSL o TLS

Questi argomenti forniscono istruzioni per l'esecuzione di singole attività relative all'utilizzo di SSL o TLS con IBM WebSphere MQ.

Molti di essi vengono utilizzati come passi nelle attività di livello superiore descritte nelle seguenti sezioni:

- “Identificazione e autenticazione degli utenti” a pagina 144
- “Autorizzazione dell'accesso agli oggetti” a pagina 159
- “Riservatezza dei messaggi” a pagina 206
- “Integrità dei dati dei messaggi” a pagina 227
- “Proteggere i cluster” a pagina 245

Utilizzo di SSL o TLS su HP Integrity NonStop Server

Descrive l'implementazione della sicurezza IBM WebSphere MQ client per HP Integrity NonStop Server OpenSSL , inclusi servizi di sicurezza, componenti, versioni di protocollo supportate, CipherSpecssupportati e funzionalità di sicurezza non supportate.

IBM WebSphere MQ Il supporto SSL & TLS fornisce i seguenti servizi di sicurezza per i canali client:

- Autenticazione del server e, facoltativamente, autenticazione del client.
- Crittografia e decrittografia dei dati che fluiscono attraverso un canale.
- Controlli di integrità sui dati che fluiscono attraverso un canale.

Il supporto SSL e TLS fornito con il client IBM WebSphere MQ per HP Integrity NonStop Server comprende i seguenti componenti:

- Librerie OpenSSL e il comando **`openssl`** .
- IBM WebSphere MQ comando stash password, **`amqrsslc`**.

I seguenti componenti richiesti per l'operazione del canale client SSL o TLS non vengono forniti con il client IBM WebSphere MQ per HP Integrity NonStop Server:

- Un daemon entropy per fornire un'origine di dati casuali per la crittografia OpenSSL .

Versioni protocollo supportate

Il client di IBM WebSphere MQ per HP Integrity NonStop Server supporta le seguenti versioni di protocollo:

- SSL 3.0
- TLS 1.0

- TLS 1.2

CipherSpecs supportati

Il client IBM WebSphere MQ per HP Integrity NonStop Server supporta le seguenti versioni CipherSpecs :

- 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 (obsoleto)
- 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

Funzionalità di sicurezza non supportata

Il client IBM WebSphere MQ per HP Integrity NonStop Server non supporta attualmente:

- PKCS#11 Supporto hardware crittografico
- Controllo elenco di revoca certificati LDAP
- Controllo protocollo stato certificato in linea OCSP
- FIPS 140-2, controlli della suite di cifratura NSA SUITE B

Gestione certificato

Utilizzare una serie di file per memorizzare le informazioni sul certificato digitale e sulla revoca del certificato.

Il supporto IBM WebSphere MQ SSL e TLS utilizza una serie di file per memorizzare le informazioni di revoca del certificato e del certificato digitale. Questi file si trovano in una directory specificata in modo programmatico tramite il campo KeyRepository nella struttura MQSCO inoltrata sulla chiamata MQCONNX, dalla variabile di ambiente *MQSSLKEYR* o, nella stanza SSL di `mqclient.ini` utilizzando l'attributo SSLKeyRepository .

La struttura MQSCO ha la precedenza sulla variabile di ambiente MQSSLKEYR che ha la precedenza sul valore della stanza del file ini.

Importante: L'ubicazione del contenitore chiavi specifica un'ubicazione di directory e non un nome file sulla piattaforma HP Integrity NonStop Server .

Il client IBM WebSphere MQ per HP Integrity NonStop Server utilizza i seguenti file, sensibili al maiuscolo / minuscolo, denominati nell'ubicazione del repository delle chiavi:

- "Archivio certificati personali" a pagina 113
- "Truststore certificato" a pagina 113
- "File stash passphrase" a pagina 113
- "File CRL (Certificate Revocation List)" a pagina 114

Archivio certificati personali
Il file di archivio certificati personali, `cert.pem`.

Questo file contiene il certificato personale e la chiave privata codificata per il client da utilizzare, in formato PEM. L'esistenza di questo file è facoltativa quando si utilizzano canali SSL o TLS che non richiedono l'autenticazione client. Laddove l'autenticazione client è richiesta dal canale e SSLCAUTH (REQUIRED) è specificato nella definizione di canale, questo file deve esistere e contenere sia il certificato che la chiave privata codificata.

Le autorizzazioni file devono essere impostate su questo file per consentire l'accesso in lettura al proprietario dell'archivio certificati.

Un file `cert.pem` formattato correttamente deve contenere esattamente due sezioni con le seguenti intestazioni e piè di pagina:

```
-----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 passphrase per la chiave privata codificata viene memorizzata nel file stash della passphrase, `Stash.sth`.

Truststore certificato
Il file truststore del certificato, `trust.pem`.

Questo file contiene le certificazioni necessarie per convalidare le certificazioni personali utilizzate dai gestori code a cui si connette il client, in formato PEM. Il truststore del certificato è obbligatorio per tutti i canali client SSL o TLS.

Le autorizzazioni file devono essere impostate per limitare l'accesso in scrittura a questo file.

Un file `trust.pem` formattato correttamente deve contenere una o più sezioni con le seguenti intestazioni e piè di pagina:

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

File stash passphrase
Il file stash della passphrase, `Stash.sth`.

Questo file è un formato binario privato di IBM WebSphere MQ e contiene la passphrase codificata da utilizzare quando si accede alla chiave privata contenuta nel file `cert.pem` . La chiave privata stessa viene archiviata nell'archivio certificati `cert.pem` .

Questo file viene creato o modificato utilizzando lo strumento della riga comandi di IBM WebSphere MQ **amqrsslc** con il parametro **-s** . Ad esempio, dove la directory `/home/alice` contiene un file `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
```

Le autorizzazioni file devono essere impostate su questo file per consentire l'accesso in lettura al proprietario dell'archivio di certificati personali associato.

File CRL (Certificate Revocation List)
Il file CRL (Certificate Revocation List), `crl.pem`.

Questo file contiene i CRL (Certificate Revocation List) che il client utilizza per convalidare i certificati digitali, in formato PEM. L'esistenza di questo file è facoltativa. Se questo file non è presente, non vengono eseguiti controlli di revoca dei certificati quando si convalidano i certificati.

Le autorizzazioni file devono essere impostate per limitare l'accesso in scrittura a questo file.

Un file `crl.pem` formattato correttamente deve contenere una o più sezioni con le seguenti intestazioni e piè di pagina:

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

Utilizzo di SSL o TLS su sistemi UNIX, Linux, and Windows

Su sistemi UNIX, Linux e Windows , il supporto SSL (Secure Sockets Layer) è installato con IBM WebSphere MQ.

Per informazioni più dettagliate sulle politiche di convalida dei certificati, consultare Convalida dei certificati e progettazione delle politiche di attendibilità.

Utilizzo di iKeyman, iKeycmd, runmqakm e runmqckm

Sui sistemi UNIX, Linux e Windows , gestire le chiavi e i certificati digitali con la GUI iKeyman o dalla riga comandi utilizzando iKeycmd o runmqakm.

- Per sistemi **UNIX and Linux** :
  - Utilizzare il comando **strmqikm** per avviare la GUI iKeyman .
  - Utilizzare il comando **runmqckm** per eseguire attività con l'interfaccia della riga comandi iKeycmd .
  - Utilizzare il comando **runmqakm** per eseguire le attività con l'interfaccia della riga comandi runmqakm. La sintassi del comando per **runmqakm** è uguale a quella per **runmqckm**.

    Se è necessario gestire i certificati SSL in modo conforme a FIPS, utilizzare il comando **runmqakm** invece dei comandi **runmqckm** o **strmqikm** .

  Vedi Gestione delle chiavi e dei certificati per una descrizione completa delle interfacce della riga di comando per i comandi **runmqckm** e **runmqakm** .

  Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, si noti che iKeycmd e iKeyman sono programmi a 64 bit. I moduli esterni richiesti per il supporto PKCS #11 verranno caricati in un processo a 64 bit, pertanto è necessario disporre di una libreria PKCS #11 a 64 bit installata per l'amministrazione dell'hardware di crittografia. Le piattaforme Windows e Linux x86 a 32 bit sono le uniche eccezioni, poiché i programmi iKeyman e iKeycmd sono a 32 bit su tali piattaforme.

Sulle seguenti piattaforme, dove il JRE era a 32 bit nelle versioni precedenti del prodotto, ma è a 64 bit solo in IBM WebSphere MQ Version 7.5, potrebbe essere necessario installare ulteriori driver PKCS#11 appropriati per la modalità di indirizzamento di **iKeyman** e **iKeycmd** JRE. Ciò è dovuto al fatto che il programma di controllo PKCS#11 deve utilizzare la stessa modalità di indirizzamento di JRE. La seguente tabella mostra le modalità di indirizzamento JRE IBM WebSphere MQ Version 7.5 .

Tabella 12. IBM WebSphere MQ Version 7.5 Modalità di indirizzamento JRE

Piattaforma	**Modalità di indirizzamento JRE**
Windows (32 bit o 64 bit)	32
Linux per System x a 32 bit	32
Linux per System x a 64 bit	64
Linux per System p	64
Linux per System z	64
HP-UX	64
Solaris Sparc	64
Solaris x86-64	64
AIX	64

Prima di eseguire il comando **strmqikm** per avviare la GUI di iKeyman , accertarsi di utilizzare una macchina in grado di eseguire il sistema X Window e di effettuare le seguenti operazioni:

– Impostare la variabile di ambiente DISPLAY, ad esempio:

```
export DISPLAY=mypc:0
```

– Verificare che la variabile di ambiente PATH contenga **/usr/bin** e **/bin**. Ciò è richiesto anche per i comandi **runmqckm** e **runmqakm** . Ad esempio:

```
export PATH=$PATH:/usr/bin:/bin
```

- Per sistemi **Windows** :
  – Utilizzare il comando **strmqikm** per avviare la GUI iKeyman .
  – Utilizzare il comando **runmqckm** per eseguire attività con l'interfaccia della riga comandi iKeycmd .

    Se è necessario gestire i certificati SSL in modo conforme a FIPS, utilizzare il comando **runmqakm** invece dei comandi **runmqckm** o **strmqikm** .

Per richiedere la traccia SSL su sistemi UNIX, Linux o Windows , consultare strmqtrc.

Riferimenti correlati
comandi runmqckm e runmqakm

Configurazione di un repository di chiavi su sistemi UNIX, Linux, and Windows

È possibile configurare un repository delle chiavi utilizzando l'interfaccia utente iKeyman oppure utilizzando i comandi **iKeycmd** o **runmqakm** .

Informazioni su questa attività

Una connessione SSL o TLS richiede un *repository di chiavi* ad ogni estremità della connessione. Ogni gestore code IBM WebSphere MQ e IBM WebSphere MQ MQI client devono avere accesso a un repository delle chiavi. Per ulteriori informazioni, vedere “Il repository delle chiavi SSL o TLS” a pagina 23.

Su sistemi UNIX, Linux, and Windows , i certificati digitali vengono memorizzati in un file di database delle chiavi gestito utilizzando l'interfaccia utente **iKeyman** o utilizzando i comandi **iKeycmd** o **runmqakm** .

Questi certificati digitali hanno etichette. Un'etichetta specifica associa un certificato personale a un gestore code o a un IBM WebSphere MQ MQI client. SSL e TLS utilizzano tale certificato per scopi di autenticazione. Sui sistemi UNIX, Linux, and Windows , IBM WebSphere MQ utilizza `ibmwebspheremq` come prefisso di etichetta per evitare confusione con i certificati per altri prodotti. Il prefisso è seguito dal nome del gestore code o dall'ID di collegamento dell'utente IBM WebSphere MQ MQI client , modificato in minuscolo. Assicurarsi di specificare l'intera etichetta del certificato in minuscolo.

Il nome del file di database delle chiavi comprende un percorso e un nome di origine:

- Sui sistemi UNIX and Linux , il percorso predefinito per il gestore code (impostato quando è stato creato il gestore code) è `/var/mqm/qmgrs/<queue_manager_name>/ssl`.

  Su sistemi Windows , il percorso predefinito è *MQ_INSTALLATION_PATH*`\Qmgrs\`*queue_manager_name*`\ssl`, dove *MQ_INSTALLATION_PATH* è la directory in cui è installato IBM WebSphere MQ . Ad esempio, `C:\program files\IBM\WebSphere MQ\Qmgrs\QM1\ssl`.

  Il nome radice predefinito è `key`. Facoltativamente, è possibile scegliere il proprio percorso e il nome della radice, ma l'estensione deve essere `.kdb`.

  Se si sceglie il proprio percorso o nome file, impostare le autorizzazioni sul file per controllare strettamente l'accesso ad esso.
- Per un client WebSphere MQ , non esiste un percorso o un nome radice predefinito. Controllare strettamente l'accesso a questo file. L'estensione deve essere `.kdb`.

Non creare repository di chiavi su file system che non supporta blocchi a livello di file, ad esempio NFS versione 2 su sistemi Linux .

Consultare “Modifica dell'ubicazione del repository delle chiavi per un gestore code su sistemi UNIX, Linux o Windows” a pagina 120 per informazioni sul controllo e la specifica del nome file del database delle chiavi. È possibile specificare il nome del file di database delle chiavi prima o dopo la creazione del file di database delle chiavi.

L'ID utente da cui si eseguono i comandi **iKeyman** o **iKeycmd** deve disporre dell'autorizzazione alla scrittura per la directory in cui viene creato o aggiornato il file del database delle chiavi. Per un gestore code che utilizza la directory `ssl` predefinita, l'ID utente da cui si esegue **iKeyman** o **iKeycmd** deve essere un membro del gruppo mqm. Per un IBM WebSphere MQ MQI client, se si esegue **iKeyman** o **iKeycmd** da un ID utente diverso da quello con cui viene eseguito il client, è necessario modificare le autorizzazioni del file per consentire a IBM WebSphere MQ MQI client di accedere al file del database delle chiavi in fase di runtime. Per ulteriori informazioni, fare riferimento a “Accesso e protezione dei file del tuo database di chiavi su Windows” a pagina 118 o “Accesso e protezione dei file del database di chiavi sui sistemi UNIX and Linux” a pagina 118.

In **iKeyman** o **iKeycmd** versione 7.0, i nuovi database di chiavi vengono popolati automaticamente con una serie di certificati CA predefiniti. In **iKeyman** o **iKeycmd** versione 8.0, i database di chiavi non vengono popolati automaticamente, rendendo la configurazione iniziale più sicura perché si includono solo i certificati CA desiderati, nel proprio file di database di chiavi.

Nota: A causa di questa modifica del comportamento per la versione 8.0 di GSKit che non comporta più l'aggiunta automatica dei certificati CA al repository, è necessario aggiungere manualmente i propri certificati CA preferiti. Questa modifica di comportamento fornisce un controllo più granulare sui certificati CA utilizzati. Consultare “Aggiunta di certificati CA predefiniti in un repository chiavi vuoto su sistemi UNIX, Linux, and Windows con GSKit versione 8.0” a pagina 119.

Il database delle chiavi viene creato utilizzando la riga comandi o l'interfaccia utente **strmqikm** (iKeyman).

Nota: Se è necessario gestire i certificati TLS in un modo conforme a FIPS, utilizzare il comando **runmqakm** . L'interfaccia utente **strmqikm** non fornisce un'opzione compatibile con FIPS.

Procedura

Creare un database delle chiavi utilizzando la riga comandi.

1. Eseguire uno dei comandi riportati di seguito:
   - Su sistemi UNIX, Linux, and Windows :

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

   - Utilizzo di runmqakm:

     ```
     runmqakm -keydb -create -db filename -pw password -type cms
         -stash -fips -strong
     ```

   dove:

   -db *nomefile*
   Specifica il nome file completo di un database di chiavi CMS e deve avere estensione `.kdb`.

   -pw *password*
   Specifica la password per il database di chiavi CMS.

   -type *cms*
   Specifica il tipo di database. (Per IBM WebSphere MQ, deve essere `cms`.)

   -stash
   Salva la password del database delle chiavi in un file.

   -fips
   Disabilita l'utilizzo della libreria crittografica BSafe. Viene utilizzato solo il componente ICC e tale componente deve essere inizializzato correttamente in modalità FIPS. In modalità FIPS, il componente ICC utilizza algoritmi convalidati da FIPS 140-2. Se il componente ICC non viene inizializzato in modalità FIPS, il comando **runmqakm** non riesce.

   -forte
   Verifica che la parola d'ordine immessa soddisfi i requisiti minimi per la complessità della parola d'ordine. I requisiti minimi per una parola d'ordine sono i seguenti:
   - La password deve avere una lunghezza minima di 14 caratteri.
   - La password deve contenere almeno un carattere minuscolo, un carattere maiuscolo e una cifra o un carattere speciale. I caratteri speciali includono l'asterisco (*), il simbolo del dollaro ($), il cancelletto (#) e il simbolo di percentuale (%). Uno spazio viene classificato come carattere speciale.
   - Ogni carattere può essere presente al massimo tre volte in una password.
   - Un massimo di due caratteri consecutivi nella password può essere identico.
   - Tutti i caratteri sono nella serie di caratteri stampabili ASCII standard nell'intervallo 0x20 - 0x7E.

In alternativa, creare un database delle chiavi utilizzando l'interfaccia utente di **strmqikm** (iKeyman).

2. Sui sistemi UNIX and Linux , accedere come utente root. Sui sistemi Windows , accedere come Amministratore o come membro del gruppo MQM.
3. Avviare l'interfaccia utente iKeyman eseguendo il comando **strmqikm** .
4. Dal menu **File database di chiavi** , fare clic su **Nuovo**.
   Si apre la finestra Nuovo.
5. Fare clic su **Tipo di database delle chiavi** e selezionare **CMS** (Certificate Management System).
6. Nel campo **Nome file** , immettere un nome file.
   Questo campo contiene già il testo `key.kdb`. Se il nome della radice è `key`, lasciare questo campo invariato. Se è stato specificato un nome di radice diverso, sostituire `key` con il proprio nome di radice. Tuttavia, non è necessario modificare l'estensione `.kdb` .
7. Nel campo **Ubicazione** , immettere il percorso.
   Ad esempio:
   - Per un gestore code: `/var/mqm/qmgrs/QM1/ssl` (su sistemi di UNIX and Linux) o `C:\Program Files\IBM\WebSphere MQ\qmgrs\QM1\ssl` (su sistemi Windows).

Il percorso deve corrispondere al valore dell'attributo **SSLKeyRepository** del gestore code.

- Per un client IBM WebSphere MQ : `/var/mqm/ssl` (sui sistemi UNIX and Linux) o `C:\mqm\ssl` (sui sistemi Windows).

8. Fare clic su **Apri**.

   Viene visualizzata la finestra Richiesta password.

9. Immettere una password nel campo **Password** e immetterla nuovamente nel campo **Conferma password** .
10. Selezionare la casella di spunta **Stash della password in un file** .

    Nota: Se non si esegue lo stash della parola d'ordine, i tentativi di avviare i canali SSL o TLS non riescono perché non possono ottenere la parola d'ordine richiesta per accedere al file di database delle chiavi.

11. Fare clic su **OK**.

    Viene visualizzata la finestra Certificati personali.

12. Impostare le autorizzazioni di accesso come descritto in “Accesso e protezione dei file del tuo database di chiavi su Windows” a pagina 118 o “Accesso e protezione dei file del database di chiavi sui sistemi UNIX and Linux” a pagina 118.

Accesso e protezione dei file del tuo database di chiavi su Windows
I file del database delle chiavi potrebbero non disporre delle autorizzazioni di accesso appropriate. È necessario impostare l'accesso appropriato a questi file.

Impostare il controllo accessi ai file *key*`.kdb`, *key*`.sth`, *key*`.crl`e *key*`.rdb`, dove *key* è il nome d'origine del database delle chiavi, per concedere l'autorizzazione a una serie limitata di utenti.

Considerare la concessione dell'accesso come segue:

autorizzazione completa
BUILTIN\Administrators, NT AUTHORITY\SYSTEM e l'utente che ha creato i file di database.

autorizzazione di lettura
Per un gestore code, solo il gruppo mqm locale. Ciò presuppone che l'MCA sia in esecuzione con un ID utente nel gruppo mqm.

Per un client, l'ID utente con cui è in esecuzione il processo client.

Accesso e protezione dei file del database di chiavi sui sistemi UNIX and Linux
I file del database delle chiavi potrebbero non disporre delle autorizzazioni di accesso appropriate. È necessario impostare l'accesso appropriato a questi file.

Per un gestore code, impostare le autorizzazioni sui file del database delle chiavi in modo che il gestore code e i processi del canale possano leggerli quando necessario, ma gli altri utenti non possono leggerli o modificarli. Normalmente, l'utente mqm ha bisogno delle autorizzazioni di lettura. Se è stato creato il file di database delle chiavi accedendo come utente mqm, le autorizzazioni sono probabilmente sufficienti; se non si era l'utente mqm, ma un altro utente nel gruppo mqm, è probabilmente necessario concedere le autorizzazioni di lettura ad altri utenti nel gruppo mqm.

Allo stesso modo per un client, impostare le autorizzazioni sui file del database delle chiavi in modo che i processi dell'applicazione client possano leggerli quando necessario, ma altri utenti non possono leggerli o modificarli. Di solito, l'utente con cui viene eseguito il processo client necessita di autorizzazioni di lettura. Se il file di database delle chiavi è stato creato accedendo come tale utente, le autorizzazioni sono probabilmente sufficienti; se non si era l'utente del processo client, ma un altro utente in quel gruppo, è probabilmente necessario concedere le autorizzazioni di lettura ad altri utenti del gruppo.

Impostare le autorizzazioni sui file *key*`.kdb`, *key*`.sth`, *key*`.crl`e *key*`.rdb`, dove *key* è il nome della radice del database delle chiavi, su read e write per il proprietario del file e su read per il gruppo di utenti mqm o client (-rw - r -----).

Aggiunta di certificati CA predefiniti in un repository chiavi vuoto su sistemi UNIX, Linux, and Windows con GSKit versione 8.0
Seguire questa procedura per aggiungere uno o più certificati CA predefiniti a un repository chiavi vuoto con GSKit versione 8.

In GSKit versione 7.0, il comportamento durante la creazione di un nuovo repository delle chiavi era quello di aggiungere automaticamente una serie di certificati CA predefiniti per le autorità di certificazione comunemente utilizzate. Per GSKit versione 8, questo comportamento è stato modificato in modo che i certificati CA non siano più aggiunti automaticamente al repository. L'utente deve ora aggiungere manualmente i certificati CA nel repository delle chiavi.

Utilizzo di iKeyman

Sulla macchina su cui si desidera aggiungere il certificato della CA, effettuare le seguenti operazioni:

1. Avviare la GUI di iKeyman utilizzando il comando **strmqikm** (su sistemi UNIX Linux e Windows).
2. Dal menu **File del database delle chiavi**, fare clic su **Apri**. Viene visualizzata la finestra Apri.
3. Fare clic su **Tipo di database delle chiavi** e selezionare **CMS** (Certificate Management System).
4. Fare clic su **Sfoglia** per passare alla directory che contiene i file del database di chiavi.
5. Selezionare il file del database delle chiavi al quale aggiungere il certificato, ad esempio `key.kdb`.
6. Fare clic su **Apri**. Viene visualizzata la finestra Richiesta password.
7. Digitare la password impostata durante la creazione del database delle chiavi e fare clic su **OK**. Il nome del file del database delle chiavi viene visualizzato nel campo **Nome file**.
8. Nel campo **Contenuto database delle chiavi**, selezionare **Certificati del firmatario**.
9. Fare clic su **Popola**. Viene visualizzata la finestra Aggiungi certificato CA.
10. I certificati CA che sono disponibili per essere aggiunti al repository vengono visualizzati in una struttura ad albero gerarchica. Selezionare la voce di livello superiore per l'organizzazione di cui si desidera considerare affidabili i certificati CA per visualizzare l'elenco completo di certificati CA validi.
11. Selezionare i certificati CA che si desidera accreditare dall'elenco e fare clic su **OK**. I certificati vengono aggiunti al repository chiavi.

Utilizzo della riga di comando

Utilizzare i seguenti comandi per elencare, quindi aggiungere i certificati CA utilizzando iKeycmd:

- Immettere il comando riportato di seguito per elencare i certificati CA predefiniti insieme alle organizzazioni che li emettono:

```
runmqckm -cert -listsigners
```

- Immettere il seguente comando per aggiungere tutti i certificati CA per l'organizzazione specificata nel campo *label* :

```
runmqckm -cert -populate -db filename -pw password -label label
```

dove:

`-db` *filename*	è il nome percorso completo del database delle chiavi.
`-pw` *password*	è la password per il database delle chiavi.
`-label` *label*	è l'etichetta allegata al certificato.

Nota: L'aggiunta di un certificato CA ad un repository di chiavi risulta in WebSphere MQ che considera attendibili tutti i certificati personali firmati da tale certificato CA. Considerare attentamente quali autorità di certificazione si desidera considerare attendibili e aggiungere solo la serie di certificati CA necessari per

autenticare i client e i gestori. Non è consigliabile aggiungere la serie completa di certificati CA predefiniti a meno che questo non sia un requisito definitivo per la politica di protezione.

Individuazione di un repository delle chiavi per un gestore code su sistemi UNIX, Linux, and Windows

Utilizzare questa procedura per ottenere l'ubicazione del file di database delle chiavi del gestore code

Procedura

1. Visualizzare gli attributi del gestore code utilizzando uno dei seguenti comandi MQSC:

   ```
   DISPLAY QMGR ALL
   DISPLAY QMGR SSLKEYR
   ```

   È anche possibile visualizzare gli attributi del gestore code utilizzando i comandi IBM WebSphere MQ Explorer o PCF.
2. Esaminare l'output del comando per il percorso e il nome del file di database delle chiavi.
   Ad esempio,
   a. su sistemi UNIX and Linux : `/var/mqm/qmgrs/QM1/ssl/key`, dove `/var/mqm/qmgrs/QM1/ssl` è il percorso e `key` è il nome della radice
   b. in Windows: *MQ_INSTALLATION_PATH*`\qmgrs\QM1\ssl\key`, dove *MQ_INSTALLATION_PATH*`\qmgrs\QM1\ssl` è il percorso e `key` è il nome della radice. *MQ_INSTALLATION_PATH* rappresenta la directory di alto livello in cui è installato WebSphere MQ .

Modifica dell'ubicazione del repository delle chiavi per un gestore code su sistemi UNIX, Linux o Windows

È possibile modificare l'ubicazione del file di database delle chiavi del proprio gestore code in vari modi, incluso il comando MQSC ALTER QMGR.

È possibile cambiare l'ubicazione del file di database delle chiavi del gestore code utilizzando il comando MQSC ALTER QMGR per impostare l'attributo del repository delle chiavi del gestore code. Ad esempio, sui sistemi UNIX and Linux :

```
ALTER QMGR SSLKEYR('/var/mqm/qmgrs/QM1/ssl/MyKey')
```

Il file di database delle chiavi ha il nome file completo: `/var/mqm/qmgrs/QM1/ssl/MyKey.kdb`

Su Windows:

```
ALTER QMGR SSLKEYR('C:\Program Files\IBM\WebSphere MQ\Qmgrs\QM1\ssl\Mykey')
```

Il file di database delle chiavi ha il nome file completo: `C:\Program Files\IBM\WebSphere MQ\Qmgrs\QM1\ssl\Mykey.kdb`

Attenzione: Assicurarsi di non includere l'estensione `.kdb` nel nome file nella parola chiave SSLKEYR, poiché il gestore code accoda questa estensione automaticamente.

È anche possibile modificare gli attributi del proprio gestore code utilizzando i comandi WebSphere MQ Explorer o PCF.

Quando si modifica l'ubicazione di un file di database di chiavi del gestore code, i certificati non vengono trasferiti dalla precedente ubicazione. Se il file di database delle chiavi a cui si sta ora accedendo è un nuovo file di database delle chiavi, è necessario popolarlo con la CA e i certificati personali necessari, come descritto in “Importazione di un certificato personale in un repository di chiavi su sistemi UNIX, Linux, and Windows” a pagina 134.

Individuazione del repository chiavi per un client IBM WebSphere MQ MQI su sistemi UNIX, Linux, and Windows

L'ubicazione del repository delle chiavi viene fornita dalla variabile MQSSLKEYR o specificata nella chiamata MQCONNX.

Esaminare la variabile di ambiente MQSSLKEYR per ottenere l'ubicazione del file di database delle chiavi del client IBM WebSphere MQ MQI. Ad esempio:

```
echo $MQSSLKEYR
```

Controllare anche l'applicazione, poiché il nome del file database delle chiavi può essere impostato anche in una chiamata MQCONNX, come descritto in “Specifica dell'ubicazione del repository delle chiavi per un client IBM WebSphere MQ MQI su sistemi UNIX, Linux, and Windows” a pagina 121. Il valore impostato in una chiamata MQCONNX sostituisce il valore di MQSSLKEYR.

Specifica dell'ubicazione del repository delle chiavi per un client IBM WebSphere MQ MQI su sistemi UNIX, Linux, and Windows

Non esiste alcun archivio chiavi predefinito per un client MQI IBM WebSphere MQ . È possibile specificarne l'ubicazione in due modi. Assicurarsi che il file di database delle chiavi sia accessibile solo agli utenti o agli amministratori previsti per impedire la copia non autorizzata su altri sistemi.

È possibile specificare l'ubicazione del file di database delle chiavi del client MQI IBM WebSphere MQ in due modi:

- Impostazione della variabile di ambiente MQSSLKEYR. Ad esempio, sui sistemi UNIX and Linux :

  ```
  export MQSSLKEYR=/var/mqm/ssl/key
  ```

  Il file di database delle chiavi ha il nome file completo:

  ```
  /var/mqm/ssl/key.kdb
  ```

  Su Windows:

  ```
  set MQSSLKEYR=C:\Program Files\IBM\WebSphere MQ\ssl\key
  ```

  Il file di database delle chiavi ha il nome file completo:

  ```
  C:\Program Files\IBM\WebSphere MQ\ssl\key.kdb
  ```

  Nota: L'estensione `.kdb` è una parte obbligatoria del nome file, ma non è inclusa come parte del valore della variabile di ambiente.
- Fornire il percorso e il nome radice del file di database delle chiavi nel campo *KeyRepository* della struttura MQSCO quando un'applicazione effettua una chiamata MQCONNX. Per ulteriori informazioni sull'utilizzo della struttura MQSCO in MQCONNX, consultare Panoramica per MQSCO .

Quando le modifiche ai certificati o all'archivio certificati diventano effettive su sistemi UNIX, Linux o Windows.

Quando si modificano i certificati in un archivio di certificati, o l'ubicazione dell'archivio di certificati, le modifiche diventano effettive a seconda del tipo di canale e della modalità di esecuzione del canale.

Le modifiche ai certificati nel file di database delle chiavi e all'attributo del repository delle chiavi diventano effettive nelle seguenti situazioni:

- Quando un nuovo processo di un singolo canale in uscita esegue per la prima volta un canale SSL.
- Quando un nuovo processo di canale singolo TCP/IP in entrata riceve per la prima volta una richiesta di avvio di un canale SSL.

- Quando il comando MQSC REFRESH SECURITY TYPE (SSL) viene emesso per aggiornare l'ambiente Websphere MQ SSL.
- Per i processi dell'applicazione client, quando viene chiusa l'ultima connessione SSL nel processo. La successiva connessione SSL ritira le modifiche del certificato.
- Per i canali che vengono eseguiti come thread di un processo di pooling del processo (amqrmppa), quando il processo di pooling del processo viene avviato o riavviato ed esegue per la prima volta un canale SSL. Se il processo di pooling del processo ha già eseguito un canale SSL e si desidera che la modifica diventi immediatamente effettiva, eseguire il comando MQSC REFRESH SECURITY TYPE (SSL).
- Per i canali eseguiti come thread dell'iniziatore di canali, quando l'iniziatore di canali viene avviato o riavviato e viene eseguito per primo un canale SSL. Se il processo dell'iniziatore di canali ha già eseguito un canale SSL e si desidera che la modifica diventi immediatamente effettiva, eseguire il comando MQSC REFRESH SECURITY TYPE (SSL).
- Per i canali eseguiti come thread di un listener TCP/IP, quando il listener viene avviato o riavviato e riceve prima una richiesta di avviare un canale SSL. Se il listener ha già eseguito un canale SSL e si desidera che la modifica diventi immediatamente effettiva, eseguire il comando MQSC REFRESH SECURITY TYPE (SSL).

È anche possibile aggiornare l'ambiente SSL WebSphere MQ utilizzando i comandi IBM WebSphere MQ Explorer o PCF.

Creazione di un certificato personale autofirmato su sistemi UNIX, Linux, and Windows

È possibile creare un certificato autofirmato utilizzando iKeyman, iKeycmdo runmqakm.

Nota: IBM WebSphere MQ non supporta gli algoritmi SHA-3 o SHA-5 . È possibile utilizzare i nomi degli algoritmi di firma digitale SHA384WithRSA e SHA512WithRSA perché entrambi gli algoritmi sono membri della famiglia SHA-2 .

I nomi degli algoritmi di firma digitale SHA3WithRSA e SHA5WithRSA sono obsoleti perché sono abbreviati rispettivamente in SHA384WithRSA e SHA512WithRSA .

Per ulteriori informazioni sul motivo per cui si desidera utilizzare i certificati autofirmati, consultare “Utilizzo di certificati autofirmati per l'autenticazione reciproca di due gestori code” a pagina 207.

Non tutti i certificati digitali possono essere utilizzati con tutti i CipherSpecs. Assicurati di creare un certificato compatibile con i CipherSpecs che devi utilizzare. WebSphere MQ supporta tre diversi tipi di CipherSpec. Per i dettagli, consultare “Interoperabilità di Elliptic Curve e RSA CipherSpecs” a pagina 35 nell'argomento “Certificati digitali e compatibilità CipherSpec in IBM WebSphere MQ” a pagina 34 . Per utilizzare CipherSpecs di tipo 1 (quelli con i nomi che iniziano con ECDHE_ECDSA_) è necessario utilizzare il comando **runmqakm** per creare il certificato ed è necessario specificare un parametro dell'algoritmo di firma ECDSA della curva ellittica; ad esempio, **-sig_alg** EC_ecdsa_with_SHA384 .

Utilizzo di iKeyman

iKeyman non fornisce un'opzione conforme a FIPS. Se è necessario gestire i certificati SSL o TLS in modo compatibile con FIPS, utilizzare il comando **runmqakm** .

Utilizzare la seguente procedura per ottenere un certificato autofirmato per il gestore code o il client WebSphere MQ MQI:

1. Avviare la GUI iKeyman utilizzando il comando **strmqikm** .
2. Dal menu **File del database delle chiavi**, fare clic su **Apri**. Viene visualizzata la finestra Apri.
3. Fare clic su **Tipo di database delle chiavi** e selezionare **CMS** (Certificate Management System).
4. Fare clic su **Sfoglia** per passare alla directory che contiene i file del database di chiavi.
5. Selezionare il file database di chiavi in cui si desidera salvare il certificato, ad esempio key . kdb.
6. Fare clic su **Apri**. Viene visualizzata la finestra Richiesta password.

7. Digitare la password impostata durante la creazione del database delle chiavi e fare clic su **OK**. Il nome del file del database di chiavi viene visualizzato nel campo **Nome file** .
8. Nel menu **Crea** , fare clic su **Nuovo certificato autofirmato**. Viene visualizzata la finestra Crea nuovo certificato autofirmato.
9. Nel campo **Etichetta chiave** , immettere:
   - Per un gestore code, `ibmwebspheremq` seguito dal nome del gestore code in lettere minuscole. Ad esempio, per `QM1`, `ibmwebspheremqqm1`o,
   - Per un client WebSphere MQ , `ibmwebspheremq` seguito dall'ID utente di collegamento ridotto in minuscolo, ad esempio `ibmwebspheremqmyuserid` .
10. Immettere o selezionare un valore per qualsiasi campo in **`Distinguished name`**o in uno dei campi **`Subject alternative name`** .
11. Per i restanti campi, accettare i valori predefiniti oppure immettere o selezionare nuovi valori. Per ulteriori informazioni sui DN (Distinguished Name), consultare “Nomi distinti” a pagina 11.
12. Fare clic su **OK**. L'elenco **Certificati personali** mostra l'etichetta del certificato personale autofirmato creato.

Utilizzo della riga di comando

Utilizzare i comandi seguenti per creare un certificato personale autofirmato utilizzando iKeycmd o runmqakm:

- Utilizzando iKeycmd su sistemi UNIX, Linux e Windows :

```
runmqckm -cert -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -x509version version -expire days
 -sig_alg algorithm
```

  Invece di `-dn` *`distinguished_name`* , è possibile utilizzare `-san_dsname` *`DNS_names`* , `-san_emailaddr` *`email_addresses`* o `-san_ipaddr` *`IP_addresses`* .

- Utilizzo di 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`*	Il nome file completo di un database di chiavi CMS.
`-pw` *`password`*	La password per il database delle chiavi CMS.
`-label` *`label`*	L'etichetta della chiave allegata al certificato.
`-dn` *`distinguished_name`*	Il DN (distinguished name) X.500 racchiuso tra virgolette. È richiesto almeno un attributo. È possibile fornire più attributi OU o DC.
`-size` *`key_size`*	La dimensione della chiave. Per iKeycmd, il valore può essere 512 o 1024. Per runmqakm, il valore può essere 512, 1024, 2048 o 4096.
`-x509version` *`version`*	La versione del certificato X.509 da creare. Il valore può essere 1, 2 o 3. Il valore predefinito è 3.
`-expire` *`days`*	La scadenza in giorni del certificato. Il valore predefinito è 365 giorni per un certificato.

`-fips`	specifica che il comando viene eseguito in modalità FIPS. Con questa modalità viene disabilitato l'uso della libreria crittografica BSafe. Viene utilizzato solo il componente ICC e tale componente deve essere inizializzato correttamente in modalità FIPS. Quando è attiva la modalità FIPS, il componente ICC utilizza gli algoritmi convalidati con FIPS 140-2. Se il componente ICC non viene inizializzato in modalità FIPS, il comando **runmqakm** non riesce.
`-sig_alg`	Per runmqakm, l'algoritmo di hash utilizzato durante la creazione di un certificato autofirmato. Questo algoritmo di hash viene utilizzato per creare la firma associata al certificato autofirmato appena creato. Il valore può essere `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`. Il valore di default è `SHA1WithRSA`.
`-sig_alg`	Per iKeycmd, l'algoritmo di firma asimmetrico utilizzato per creare la coppia di chiavi della voce. Il valore può essere `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` , `SHAWithDSAo` `SHAWithRSA`. Il valore di default è `SHA1WithRSA`.
`-san_dnsname` *DNS_names*	Un elenco delimitato da virgole o spazi di nomi DNS per la voce che si sta creando.
`-san_emailaddr` *email_addresses*	Un elenco delimitato da virgole o spazi di indirizzi email per la voce che si sta creando.
`-san_ipaddr` *IP_addresses*	Un elenco delimitato da virgole o spazi di indirizzi IP per la voce che si sta creando.

distributed ***Richiesta di certificato personale su sistemi UNIX, Linux, and Windows***

È possibile richiedere un certificato personale utilizzando la **strmqikm** (iKeyman) GUI o dalla riga comandi utilizzando i comandi **runmqckm** o **runmqakm** . Se è necessario gestire i certificati SSL o TLS in modo compatibile con FIPS, utilizzare il comando **runmqakm** .

Informazioni su questa attività

Puoi richiedere un certificato personale utilizzando la GUI iKeyman o dalla riga di comando, in base alle seguenti considerazioni:

- WebSphere MQ non supporta gli algoritmi SHA-3 o SHA-5 . È possibile utilizzare i nomi degli algoritmi di firma digitale SHA384WithRSA e SHA512WithRSA perché entrambi gli algoritmi sono membri della famiglia SHA-2 .
- I nomi degli algoritmi di firma digitale SHA3WithRSA e SHA5WithRSA sono obsoleti perché sono abbreviati rispettivamente in SHA384WithRSA e SHA512WithRSA .
- Non tutti i certificati digitali possono essere utilizzati con tutti i CipherSpecs. Assicurati di richiedere un certificato compatibile con i CipherSpecs che devi utilizzare. WebSphere MQ supporta tre diversi tipi di

CipherSpec. Per i dettagli, consultare “Interoperabilità di Elliptic Curve e RSA CipherSpecs” a pagina 35 nell'argomento “Certificati digitali e compatibilità CipherSpec in IBM WebSphere MQ” a pagina 34 .

- Per utilizzare CipherSpecs di tipo 1 (con nomi che iniziano con ECDHE_ECDSA_) è necessario utilizzare il comando **runmqakm** per richiedere il certificato ed è necessario specificare un parametro dell'algoritmo di firma ECDSA della curva ellittica; ad esempio **-sig_alg** EC_ecdsa_with_SHA384.
- Solo il comando runmqakm fornisce un'opzione conforme a FIPS.
- Se si sta utilizzando l'hardware crittografico, consultare “Richiesta di un certificato personale per l'hardware PKCS #11” a pagina 141.

Utilizzo dell'interfaccia utente iKeyman

Informazioni su questa attività

iKeyman non fornisce un'opzione conforme a FIPS. Se è necessario gestire i certificati SSL o TLS in modo compatibile con FIPS, utilizzare il comando **runmqakm** .

Procedura

Completa la seguente procedura per applicare un certificato personale, utilizzando l'interfaccia utente iKeyman :

1. Avviare l'interfaccia utente iKeyman utilizzando il comando **strmqikm** .
2. Dal menu **File del database delle chiavi**, fare clic su **Apri**.

   Verrà visualizzata la finestra **Apri**.
3. Fare clic su **Tipo di database delle chiavi** e selezionare **CMS** (Certificate Management System).
4. Fare clic su **Sfoglia** per passare alla directory che contiene i file del database di chiavi.
5. Selezionare il file di database delle chiavi da cui si desidera creare la richiesta; ad esempio key.kdb.
6. Fare clic su **Apri**.

   Viene visualizzata la finestra **Richiesta password** .
7. Digitare la password impostata durante la creazione del database delle chiavi e fare clic su **OK**.

   Il nome del file del database delle chiavi viene visualizzato nel campo **Nome file** .
8. Dal menu **Crea** , fare clic su **Nuova richiesta certificato**. Viene visualizzata la finestra **Crea nuova chiave e richiesta certificato** .
9. Nel campo **Etichetta chiave** , immettere le seguenti etichette:
   - Per un gestore code, immettere ibmwebspheremq seguito dal nome del gestore code modificato in minuscolo. Ad esempio, per un gestore code denominato QM1, immettere ibmwebspheremqqm1.
   - Per un IBM WebSphere MQ MQI client, immettere ibmwebspheremq seguito dall'ID utente di accesso, tutto in minuscolo; ad esempio, ibmwebspheremqmyuserid .
10. Immettere o selezionare un valore per qualsiasi campo nel campo **DN (Distinguished Name)** o uno dei campi **Nome alternativo oggetto** . Per i restanti campi, accettare i valori predefiniti oppure immettere o selezionare nuovi valori.

    Per ulteriori informazioni sui DN (Distinguished Name), consultare “Nomi distinti” a pagina 11.
11. Nel campo **Immettere il nome di un file in cui memorizzare la richiesta di certificato** , accettare il valore predefinito certreq.armoppure immettere un nuovo valore con un percorso completo.
12. Fare clic su **OK**.

    Viene visualizzata una finestra di conferma.
13. Fare clic su **OK**.

    L'elenco **Richieste di certificati personali** mostra l'etichetta della nuova richiesta di certificato personale creata. La richiesta di certificato viene memorizzata nel file scelto nel passo “11” a pagina 125.
14. Richiedere il nuovo certificato personale inviando il file a un'autorità di certificazione (CA) o copiando il file nel modulo di richiesta sul sito web per l'autorità di certificazione.

Procedura

Utilizzare i seguenti comandi per richiedere un certificato personale utilizzando il comando **runmqckm** o **runmqakm** :

- Utilizzo di **runmqckm**:

```
runmqckm -certreq -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -file filename -sig_alg algorithm
```

  Invece di -dn *distinguished_name* , è possibile utilizzare -san_dsname *DNS_names* , -san_emailaddr *email_addresses* o -san_ipaddr *IP_addresses* .

- Utilizzo di runmqakm:

```
runmqakm -certreq -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -file filename -fips
        -sig_alg algorithm
```

dove:

-db *nomefile*
Specifica il nome file completo di un database di chiavi CMS.

-pw *password*
Specifica la password per il database di chiavi CMS.

-label *label*
Specifica l'etichetta chiave allegata al certificato.

-dn *nome_distinto*
Specifica il nome distinto X.500 racchiuso tra virgolette. È richiesto almeno un attributo. È possibile fornire più attributi OU e DC.

-size *dimensione_chiave*
Specifica la dimensione della chiave. Se si utilizza **runmqckm** , il valore può essere 512 o 1024. Se si sta utilizzando **runmqakm**, il valore può essere 512, 1024o 2048.

-file *nomefile*
Specifica il nome file per la richiesta di certificato.

-fips
specifica che il comando viene eseguito in modalità FIPS. Con questa modalità viene disabilitato l'uso della libreria crittografica BSafe. Viene utilizzato solo il componente ICC e tale componente deve essere inizializzato correttamente in modalità FIPS. In modalità FIPS, il componente ICC utilizza algoritmi convalidati da FIPS 140-2. Se il componente ICC non viene inizializzato in modalità FIPS, il comando **runmqakm** non riesce.

-sig_alg
Per **runmqckm**, specifica l'algoritmo di firma asimmetrico utilizzato per creare la coppia di chiavi della voce. Il valore può essere 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, SHAWithDSAo SHAWithRSA . Il valore predefinito è SHA1WithRSA

-sig_alg
In **runmqakm**, specifica l'algoritmo di hash utilizzato durante la creazione di una richiesta di certificato. Questo algoritmo di hash viene utilizzato per creare la firma associata alla richiesta di certificato appena creata. Il valore può essere 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. Il valore di default è SHA1WithRSA.

-san_dnsname ***nomi_DNS***
Specifica un elenco delimitato da virgole o da spazi di nomi DNS per la voce che si sta creando.

-san_emailaddr ***indirizzi_email***
Specifica un elenco delimitato da virgole o da spazi di indirizzi email per la voce che si sta creando.

-san_ipaddr ***indirizzi_IP***
Specifica un elenco delimitato da virgole o da spazi di indirizzi IP per la voce che si sta creando.

Rinnovo di un certificato personale esistente su sistemi UNIX, Linux, and Windows

È possibile rinnovare un certificato personale utilizzando l'interfaccia utente iKeyman oppure utilizzando i comandi **iKeycmd** o **runmqakm** .

Prima di iniziare

Se hai il requisito di utilizzare dimensioni di chiavi più grandi per i tuoi certificati personali, i passi di rinnovo descritti di seguito non funzionano, perché la richiesta di certificato ricreato viene generata da una chiave esistente.

Seguire i passi descritti in “Richiesta di certificato personale su sistemi UNIX, Linux, and Windows” a pagina 124 per creare una nuova richiesta certificato, utilizzando le dimensioni chiave richieste. Questo processo sostituisce la chiave esistente.

Informazioni su questa attività

Un certificato personale ha una data di scadenza, dopo la quale il certificato non può più essere utilizzato. In questa sezione viene illustrato come rinnovare un certificato personale esistente prima della scadenza.

Utilizzo dell'interfaccia utente iKeyman

Informazioni su questa attività

iKeyman non fornisce un'opzione conforme a FIPS. Se è necessario gestire i certificati SSL o TLS in modo compatibile con FIPS, utilizzare il comando **runmqakm** .

Procedura

Completa la seguente procedura per applicare un certificato personale, utilizzando l'interfaccia utente iKeyman :

1. Avviare l'interfaccia utente iKeyman utilizzando il comando **strmqikm** sui sistemi UNIX, Linux, and Windows .
2. Dal menu **File del database delle chiavi**, fare clic su **Apri**.
   Verrà visualizzata la finestra **Apri**.
3. Fare clic su **Tipo di database delle chiavi** e selezionare **CMS** (Certificate Management System).
4. Fare clic su **Sfoglia** per passare alla directory che contiene i file del database di chiavi.
5. Selezionare il file di database delle chiavi da cui si desidera creare la richiesta; ad esempio key.kdb.
6. Fare clic su **Apri**.
   Viene visualizzata la finestra **Richiesta password** .
7. Digitare la password impostata durante la creazione del database delle chiavi e fare clic su **OK**.
   Il nome del file del database delle chiavi viene visualizzato nel campo **Nome file** .

8. Selezionare **Certificati personali** dal menu a discesa e selezionare il certificato dall'elenco che si desidera rinnovare.
9. Fare clic su **Ricrea richiesta ...** pulsante.

   Viene visualizzata una finestra in cui è possibile immettere il nome file e le relative informazioni.
10. Nel campo **nome file**, accettare il valore predefinito `certreq.arm`oppure immettere un nuovo valore, incluso il percorso file completo.
11. Fare clic su **OK**. La richiesta di certificati è memorizzata nel file selezionato nel passo “9” a pagina 128.
12. Richiedere il nuovo certificato personale inviando il file a un'autorità di certificazione (CA) o copiando il file nel modulo di richiesta sul sito web per l'autorità di certificazione.

Utilizzo della riga di comando

Procedura

Utilizzare i seguenti comandi per richiedere un certificato personale utilizzando il comando **iKeycmd** o **runmqakm** :

- Utilizzo di **iKeycmd** su sistemi UNIX, Linux, and Windows :

```
runmqckm -certreq -recreate -db filename -pw
password -label label
 -target filename
```

- Utilizzo di runmqakm:

```
runmqakm -certreq -recreate -db filename -pw
password -label label
  -target filename
```

dove:

-db *nomefile*
Specifica il nome file completo di un database di chiavi CMS.

-pw *password*
Specifica la password per il database di chiavi CMS.

-target *nomefile*
Specifica il nome file per la richiesta di certificato.

Operazioni successive

Una volta ricevuto il certificato personale firmato dall'autorità di certificazione, è possibile aggiungerlo al database delle chiavi utilizzando la procedura descritta in “Ricezione di certificati personali in un archivio di chiavi su sistemi UNIX, Linux e Windows” a pagina 128.

Ricezione di certificati personali in un archivio di chiavi su sistemi UNIX, Linux e Windows

Utilizzare questa procedura per ricevere un certificato personale nel file database delle chiavi. Il repository delle chiavi deve essere lo stesso repository in cui è stata creata la richiesta di certificato.

Dopo che la CA ha inviato un nuovo certificato personale, lo si aggiunge al file database delle chiavi da cui è stata generata la nuova richiesta di certificato. Se la CA invia il certificato come parte di un messaggio email, copiare il certificato in un file separato.

Utilizzo di iKeyman

Se è necessario gestire i certificati SSL in modo conforme a FIPS, utilizzare il comando runmqakm. iKeyman non fornisce un'opzione conforme a FIPS.

Verificare che il file di certificato da importare disponga dell'autorizzazione di scrittura per l'utente corrente, quindi utilizzare la seguente procedura per un gestore code o un WebSphere MQ client MQI per ricevere un certificato personale nel file di database delle chiavi:

1. Avviare la GUI di iKeyman utilizzando il comando **strmqikm** (su Windows UNIX and Linux).
2. Dal menu **File del database delle chiavi**, fare clic su **Apri**. Viene visualizzata la finestra Apri.
3. Fare clic su **Tipo di database delle chiavi** e selezionare **CMS** (Certificate Management System).
4. Fare clic su **Sfoglia** per passare alla directory che contiene i file del database di chiavi.
5. Selezionare il file del database delle chiavi al quale aggiungere il certificato, ad esempio `key.kdb`.
6. Fare clic su **Apri**, quindi su **OK**. Viene visualizzata la finestra Richiesta password.
7. Digitare la password impostata durante la creazione del database delle chiavi e fare clic su **OK**. Il nome del file del database di chiavi viene visualizzato nel campo **Nome file** . Selezionare la vista **Certificati personali** .
8. Fare clic su **Ricevi**. Viene visualizzata la finestra Ricevi certificato da file.
9. Immettere il nome file del certificato e l'ubicazione per il nuovo certificato personale oppure fare clic su **Sfoglia** per selezionare il nome e l'ubicazione.
10. Fare clic su **OK**. Se si dispone già di un certificato personale nel proprio database delle chiavi, viene visualizzata una finestra in cui viene richiesto se si desidera impostare la chiave che si sta aggiungendo come chiave predefinita nel database.
11. Fare clic su **Sì** o **No**. Viene visualizzata la finestra Immettere un'etichetta.
12. Fare clic su **OK**. Il campo **Certificati personali** visualizza l'etichetta del nuovo certificato personale aggiunto.

Utilizzo della riga di comando

Utilizzare i seguenti comandi per aggiungere un certificato personale a un file di database delle chiavi utilizzando iKeycmd :

- Su UNIX, Linux e Windows, immettere il seguente comando:

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

dove:

`-file` *filename*	è il nome file completo del file contenente il certificato personale.
`-db` *filename*	è il nome file completo di un database di chiavi CMS.
`-pw` *password*	è la password per il database di chiavi CMS.
`-format` *ascii*	è il formato del certificato. Il valore può essere `ascii` per i dati codificati in Base64-encoded ASCII o `binary` per i dati DER binari. Il valore predefinito è `ascii` .

Se si sta utilizzando l'hardware crittografico, fare riferimento a “Importazione di un certificato personale nell'hardware PKCS #11” a pagina 143.

Estrazione di un certificato CA da un repository delle chiavi

Seguire questa procedura per estrarre un certificato CA.

Utilizzo di iKeyman

Se è necessario gestire i certificati SSL in modo conforme a FIPS, utilizzare il comando runmqakm. iKeyman non fornisce un'opzione conforme a FIPS.

Eseguire le seguenti operazioni sulla macchina da cui si desidera estrarre il certificato CA:

1. Avviare la GUI di iKeyman utilizzando il comando **strmqikm** ..
2. Dal menu **File del database delle chiavi**, fare clic su **Apri**. Viene visualizzata la finestra Apri.
3. Fare clic su **Tipo di database delle chiavi** e selezionare **CMS** (Certificate Management System).
4. Fare clic su **Sfoglia** per passare alla directory che contiene i file del database di chiavi.
5. Selezionare il file di database delle chiavi da cui si desidera estrarre, ad esempio `key.kdb`.
6. Fare clic su **Apri**. Viene visualizzata la finestra Richiesta password.
7. Digitare la password impostata durante la creazione del database delle chiavi e fare clic su **OK**. Il nome del file del database di chiavi viene visualizzato nel campo **Nome file** .
8. Nel campo **Contenuto database di chiavi** , selezionare **Certificati del firmatario** e selezionare il certificato che si desidera estrarre.
9. Fare clic su **Estrai**. Viene visualizzata la finestra Estrai un certificato in un file.
10. Selezionare il **Tipo di dati** del certificato, ad esempio **Base64-encoded** per un file con estensione `.arm` .
11. Immettere il nome file del certificato e l'ubicazione in cui si desidera memorizzare il certificato oppure fare clic su **Sfoglia** per selezionare il nome e l'ubicazione.
12. Fare clic su **OK**. Il certificato viene scritto nel file specificato.

Utilizzo della riga di comando

Utilizzare i seguenti comandi per estrarre un certificato CA utilizzando iKeycmd :

- In UNIX, Linux e Windows:

```
runmqckm -cert -extract -db filename -pw password -label label -target filename
         -format ascii
```

dove:

`-db` *`filename`*	è il nome percorso completo di un database di chiavi CMS.
`-pw` *`password`*	è la password per il database di chiavi CMS.
`-label` *`label`*	è l'etichetta allegata al certificato.
`-target` *`filename`*	è il nome del file di destinazione.
`-format` *`ascii`*	è il formato del certificato. Il valore può essere `ascii` per ASCII con codifica Base64 oppure `binary` per i dati binari DER. Il valore predefinito è `ascii`.

Estrazione della parte pubblica di un certificato autofirmato da un repository delle chiavi su sistemi UNIX, Linux e Windows

Seguire questa procedura per estrarre la parte pubblica di un certificato autofirmato.

Utilizzo di iKeyman

Se è necessario gestire i certificati SSL in modo conforme a FIPS, utilizzare il comando runmqakm. iKeyman non fornisce un'opzione conforme a FIPS.

Effettuare le seguenti operazioni sulla macchina da cui si desidera estrarre la parte pubblica di un certificato autofirmato:

1. Avviare la GUI iKeyman utilizzando il comando **strmqikm** (in UNIX, Linux e Windows).
2. Dal menu **File del database delle chiavi**, fare clic su **Apri**. Viene visualizzata la finestra Apri.
3. Fare clic su **Tipo di database delle chiavi** e selezionare **CMS** (Certificate Management System).

4. Fare clic su **Sfoglia** per passare alla directory che contiene i file del database di chiavi.
5. Selezionare il file database di chiavi da cui si desidera estrarre il certificato, ad esempio `key.kdb`.
6. Fare clic su **Apri**. Viene visualizzata la finestra Richiesta password.
7. Digitare la password impostata durante la creazione del database delle chiavi e fare clic su **OK**. Il nome del file del database di chiavi viene visualizzato nel campo **Nome file** .
8. Nel campo **Contenuto database delle chiavi** , selezionare **Certificati personali** e selezionare il certificato.
9. Fare clic su **Estrai certificato**. Viene visualizzata la finestra Estrai un certificato in un file.
10. Selezionare il **Tipo di dati** del certificato, ad esempio **Base64-encoded** per un file con estensione `.arm` .
11. Immettere il nome file del certificato e l'ubicazione in cui si desidera memorizzare il certificato oppure fare clic su **Sfoglia** per selezionare il nome e l'ubicazione.
12. Fare clic su **OK**. Il certificato viene scritto nel file specificato. Notare che quando si estrae (piuttosto che esportare) un certificato, viene inclusa solo la parte pubblica del certificato, quindi non è richiesta una password.

Utilizzo della riga di comando

Utilizzare i seguenti comandi per estrarre la parte pubblica di un certificato autofirmato utilizzando iKeycmd o runmqakm:

- In UNIX, Linux e Windows:

```
runmqckm -cert -extract -db filename -pw password -label label -target filename
         -format ascii
```

- Utilizzo di runmqakm:

```
runmqakm -cert -extract -db filename -pw password -label label
            -target filename -format ascii -fips
```

dove:

`-db` *`filename`*	è il nome percorso completo di un database di chiavi CMS.
`-pw` *`password`*	è la password per il database di chiavi CMS.
`-label` *`label`*	è l'etichetta allegata al certificato.
`-target` *`filename`*	è il nome del file di destinazione.
`-format` *`ascii`*	è il formato del certificato. Il valore può essere `ascii` per ASCII con codifica Base64 oppure `binary` per i dati binari DER. Il valore predefinito è `ascii`.

Aggiunta di un certificato CA (o la parte pubblica di un certificato autofirmato) in un repository delle chiavi su sistemi UNIX, Linux, and Windows

Attenersi alla seguente procedura per aggiungere un certificato CA o la parte pubblica di un certificato autofirmato a un repository di chiavi.

Se il certificato che si desidera aggiungere si trova in una catena di certificati, è anche necessario aggiungere tutti i certificati che lo precedono nella catena. I certificati devono essere aggiunti in ordine rigorosamente discendente, iniziando dalla root, con il certificato della CA immediatamente successivo e così via.

Laddove le seguenti istruzioni si riferiscono a un certificato CA, queste si applicheranno anche alla parte pubblica di un certificato autofirmato.

Nota: Se il certificato che si desidera aggiungere si trova in una catena di certificati, è necessario aggiungere anche tutti i certificati che lo precedono nella catena. È necessario assicurarsi che il certificato sia con codifica ASCII (UTF-8) o binaria (DER), poiché IBM GSKit (Global Secure Toolkit) non supporta i certificati con altri tipi di codifica. I certificati devono essere aggiunti in ordine rigorosamente discendente, iniziando dalla root, seguita immediatamente dal certificato CA nella catena.

Utilizzo di iKeyman

Se è necessario gestire i certificati SSL in modo conforme a FIPS, utilizzare il comando runmqakm. iKeyman non fornisce un'opzione conforme a FIPS.

Sulla macchina su cui si desidera aggiungere il certificato della CA, effettuare le seguenti operazioni:

1. Avviare la GUI di iKeyman utilizzando il comando **strmqikm** (su sistemi UNIX Linux e Windows).
2. Dal menu **File del database delle chiavi**, fare clic su **Apri**. Viene visualizzata la finestra Apri.
3. Fare clic su **Tipo di database delle chiavi** e selezionare **CMS** (Certificate Management System).
4. Fare clic su **Sfoglia** per passare alla directory che contiene i file del database di chiavi.
5. Selezionare il file del database delle chiavi al quale aggiungere il certificato, ad esempio key.kdb.
6. Fare clic su **Apri**. Viene visualizzata la finestra Richiesta password.
7. Digitare la password impostata durante la creazione del database delle chiavi e fare clic su **OK**. Il nome del file del database delle chiavi viene visualizzato nel campo **Nome file**.
8. Nel campo **Contenuto database delle chiavi**, selezionare **Certificati del firmatario**.
9. Fare clic su **Aggiungi**. Viene aperta la finestra per aggiungere la certificazione della CA (autorità di certificazione) da un file.
10. Digitare il nome del file e la posizione di memorizzazione del certificato oppure fare clic su **Sfoglia** per selezionare il nome e la posizione.
11. Fare clic su **OK**. Viene visualizzata la finestra Immettere un'etichetta.
12. Nella finestra Immettere un'etichetta, digitare il nome del certificato.
13. Fare clic su **OK**. Il certificato viene aggiunto al database di chiavi.

Utilizzo della riga di comando

Utilizzare i seguenti comandi per aggiungere un certificato CA mediante iKeycmd :

- Su UNIX, Linux e Windows, immettere il seguente comando:

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

dove:

-db *filename*	è il nome percorso completo del database di chiavi CMS.
-pw *password*	è la password per il database di chiavi CMS.
-label *label*	è l'etichetta allegata al certificato.
-file *filename*	è il nome del file contenente il certificato.
-format *ascii*	è il formato del certificato. Il valore può essere ascii per ASCII con codifica Base64 oppure binary per i dati binari DER. Il valore predefinito è ascii.

Esportazione di un certificato personale da un repository delle chiavi

Seguire questa procedura per esportare un certificato personale.

Utilizzo di iKeyman

Se è necessario gestire i certificati SSL in modo conforme a FIPS, utilizzare il comando runmqakm. iKeyman non fornisce un'opzione conforme a FIPS.

Effettuare le seguenti operazioni sulla macchina da cui si desidera esportare il certificato personale:

1. Avviare la GUI di iKeyman utilizzando il comando **strmqikm** (in Windows UNIX and Linux).
2. Dal menu **File del database delle chiavi**, fare clic su **Apri**. Viene visualizzata la finestra Apri.
3. Fare clic su **Tipo di database delle chiavi** e selezionare **CMS** (Certificate Management System).
4. Fare clic su **Sfoglia** per passare alla directory che contiene i file del database di chiavi.
5. Selezionare il file di database delle chiavi da cui si desidera esportare il certificato, ad esempio key.kdb.
6. Fare clic su **Apri**. Viene visualizzata la finestra Richiesta password.
7. Digitare la password impostata durante la creazione del database delle chiavi e fare clic su **OK**. Il nome del file del database di chiavi viene visualizzato nel campo **Nome file** .
8. Nel campo **Contenuto database di chiavi** , selezionare **Certificati personali** e selezionare il certificato che si desidera esportare.
9. Fare clic su **Esporta / Importa**. Viene visualizzata la finestra Esporta / Importa chiave.
10. Selezionare **Esporta chiave**.
11. Selezionare il **Tipo di file di chiavi** del certificato che si desidera esportare, ad esempio **PKCS12**.
12. Immettere il nome file e l'ubicazione in cui si desidera esportare il certificato oppure fare clic su **Sfoglia** per selezionare il nome e l'ubicazione.
13. Fare clic su **OK**. Viene visualizzata la finestra Richiesta password. Notare che quando si esporta (piuttosto che estrarre) un certificato, sono incluse sia le parti pubbliche che private del certificato. Questo è il motivo per cui il file esportato è protetto da una password. Quando si estrae un certificato, viene inclusa solo la parte pubblica del certificato, quindi non è richiesta una password.
14. Immettere una password nel campo **Password** e immetterla nuovamente nel campo **Conferma password** .
15. Fare clic su **OK**. Il certificato viene esportato nel file specificato.

Utilizzo della riga di comando

Utilizzare i seguenti comandi per esportare un certificato personale utilizzando iKeycmd:

- In UNIX, Linux e Windows:

```
runmqckm -cert -export -db filename -pw password -label label -type cms
        -target filename -target_pw password -target_type pkcs12
```

dove:

-db *filename*	è il nome percorso completo del database di chiavi CMS.
-pw *password*	è la password per il database di chiavi CMS.
-label *label*	è l'etichetta allegata al certificato.
-type *cms*	è il tipo di database.
-target *filename*	è il nome percorso completo del file di destinazione.
-target_pw *password*	è la parola d'ordine per codificare il certificato.
-target_type *pkcs12*	è il tipo di certificato.

Importazione di un certificato personale in un repository di chiavi su sistemi UNIX, Linux, and Windows

Seguire questa procedura per importare un certificato personale

Prima di importare un certificato personale in formato PKCS #12 nel file di database delle chiavi, è necessario aggiungere la catena valida completa di certificati CA di emissione al file di database delle chiavi (consultare "Aggiunta di un certificato CA (o la parte pubblica di un certificato autofirmato) in un repository delle chiavi su sistemi UNIX, Linux, and Windows" a pagina 131).

I file PKCS #12 devono essere considerati temporanei ed eliminati dopo l'utilizzo.

Utilizzo di iKeyman

Se è necessario gestire i certificati SSL in un modo conforme a FIPS, utilizzare il comando runmqakm. iKeyman non fornisce un'opzione conforme a FIPS.

Eseguire le seguenti operazioni sulla macchina su cui si desidera importare il certificato personale:

1. Avviare la GUI iKeyman utilizzando il comando **strmqikm** .
2. Dal menu **File del database delle chiavi**, fare clic su **Apri**. Viene visualizzata la finestra Apri.
3. Fare clic su **Tipo di database delle chiavi** e selezionare **CMS** (Certificate Management System).
4. Fare clic su **Sfoglia** per passare alla directory che contiene i file del database di chiavi.
5. Selezionare il file del database delle chiavi al quale aggiungere il certificato, ad esempio `key.kdb`.
6. Fare clic su **Apri**. Viene visualizzata la finestra Richiesta password.
7. Digitare la password impostata durante la creazione del database delle chiavi e fare clic su **OK**. Il nome del file del database delle chiavi viene visualizzato nel campo **Nome file**.
8. Nel campo **Contenuto database chiavi** , selezionare **Certificati personali**.
9. Se sono presenti certificati nella vista Certificati personali, attenersi alla seguente procedura:
   a. Fare clic su **Esporta / Importa**. Viene visualizzata la finestra Esporta / Importa chiave.
   b. Selezionare **Importa chiave**.
10. Se non ci sono certificati nella vista Certificati personali, fare clic su **Importa**.
11. Selezionare il **Tipo di file chiave** del certificato che si desidera importare, ad esempio PKCS12.
12. Digitare il nome del file e la posizione di memorizzazione del certificato oppure fare clic su **Sfoglia** per selezionare il nome e la posizione.
13. Fare clic su **OK**. Viene visualizzata la finestra Richiesta password.
14. Nel campo **Password** , immettere la password utilizzata quando è stato esportato il certificato.
15. Fare clic su **OK**. Viene visualizzata la finestra Modifica etichette. Questa finestra consente di modificare le etichette dei certificati importati se, ad esempio, un certificato con la stessa etichetta già esiste nel database delle chiavi di destinazione. La modifica delle etichette certificato non ha alcun effetto sulla convalida della catena di certificati. Può essere utilizzato per modificare l'etichetta del certificato personale in quella richiesta da WebSphere MQ per associare il certificato al particolare gestore code o client (`ibmwebspheremqqm1` ad esempio).
16. Per modificare un'etichetta, selezionare l'etichetta richiesta dall'elenco **Seleziona un'etichetta da modificare** . L'etichetta viene copiata nel campo di immissione **Immettere una nuova etichetta** . Sostituire il testo dell'etichetta con quello della nuova etichetta e fare clic su **Applica**.
17. Il testo nel campo di immissione **Immettere una nuova etichetta** viene copiato nuovamente nel campo **Selezionare un'etichetta da cambiare** , sostituendo l'etichetta originariamente selezionata e rietichettando il certificato corrispondente.
18. Dopo aver modificato tutte le etichette che dovevano essere modificate, fare clic su **OK**. La finestra Modifica etichette si chiude e la finestra IBM Key Management originale viene nuovamente visualizzata con i campi **Certificati personali** e **Certificati del firmatario** aggiornati con i certificati correttamente etichettati.

19. Il certificato viene importato nel database delle chiavi di destinazione.

Utilizzo della riga di comando

Per importare un certificato personale utilizzando iKeycmd, utilizzare i seguenti comandi:

- In UNIX, Linux e Windows:

```
runmqckm -cert -import -file filename -pw password -type pkcs12 -target filename
-target_pw password -target_type cms -label label
```

dove:

`-file filename`	è il nome file completo del file contenente il certificato PKCS #12 .
`-pw password`	è la parola d'ordine per il certificato #12 PKCS.
`-type pkcs12`	è il tipo di file.
`-target filename`	è il nome del database di chiavi CMS di destinazione.
`-target_pw password`	è la password per il database di chiavi CMS.
`-target_type cms`	è il tipo di database specificato da `-target`
`-label label`	è l'etichetta del certificato da importare dal database delle chiavi origine.
`-new_label label`	è l'etichetta che il certificato verrà assegnato nel database di destinazione. Se si omette l'opzione `-new_label` , il valore predefinito è utilizzare lo stesso dell'opzione `-label` .

iKeycmd non fornisce un comando per modificare direttamente le etichette dei certificati. Utilizzare la seguente procedura per modificare un'etichetta del certificato:

1. Esportare il certificato in un file PKCS #12 utilizzando il comando **`-cert -export`** . Specificare l'etichetta del certificato esistente per l'opzione `-label` .
2. Rimuovere la copia esistente del certificato dal database delle chiavi originale utilizzando il comando **`-cert -delete`** .
3. Importare il certificato dal file PKCS #12 utilizzando il comando **`-cert -import`** . Specificare la vecchia etichetta per l'opzione `-label` e la nuova etichetta richiesta per l'opzione `-new_label` . Il certificato verrà reimportato nel database di chiavi con l'etichetta richiesta.

Importazione da un file Microsoft .pfx

Utilizzare questa procedura per eseguire il mport da un file Microsoft .pfx utilizzando iKeyman. Non è possibile utilizzare runmqakm per importare un file .pfx.

Un file .pfx può contenere due certificati relativi alla stessa chiave. Uno è un certificato personale o del sito (contenente sia una chiave pubblica che una privata). L'altro è un certificato CA (firmatario) (contenente solo una chiave pubblica). Questi certificati non possono coesistere nello stesso file di database delle chiavi CMS, pertanto è possibile importarne solo uno. Inoltre, il "nome descrittivo" o l'etichetta sono allegati solo al certificato del firmatario.

Il certificato personale è identificato da un UUID (Unique User Identifier) generato dal sistema. Questa sezione mostra l'importazione di un certificato personale da un file pfx mentre lo etichetta con il nome descrittivo precedentemente assegnato al certificato CA (firmatario). I certificati CA (firmatario) di emissione devono essere già aggiunti al database delle chiavi di destinazione. Notare che i file PKCS#12 devono essere considerati temporanei ed eliminati dopo l'utilizzo.

Seguire questi passi per importare un certificato personale da un database di chiavi pfx di origine:

1. Avviare la GUI di iKeyman utilizzando il comando **`strmqikm`** (su Linux, UNIX o Windows). Viene visualizzata la finestra IBM Key Management.
2. Dal menu **File del database delle chiavi**, fare clic su **Apri**. Viene visualizzata la finestra Apri.

3. Selezionare un tipo di database di chiavi **PKCS12**.
4. **Si consiglia di eseguire un backup del database pfx prima di eseguire questa fase.** Selezionare il database di chiavi pfx che si desidera importare. Fare clic su **Apri**. Viene visualizzata la finestra Richiesta password.
5. Immettere la password del database delle chiavi e fare clic su **OK**. Viene visualizzata la finestra IBM Key Management. La barra del titolo mostra il nome del file di database delle chiavi pfx selezionato, indicando che il file è aperto e pronto.
6. Selezionare **Certificati firmatario** dall'elenco. Il "nome descrittivo" del certificato richiesto viene visualizzato come etichetta nel pannello Certificati del firmatario.
7. Selezionare la voce etichetta e fare clic su **Elimina** per eliminare il certificato del firmatario. Viene visualizzata la finestra Conferma.
8. Fare clic su **Sì**. L'etichetta selezionata non viene più visualizzata nel pannello Certificati del firmatario.
9. Ripetere i passi 6, 7 e 8 per tutti i certificati del firmatario.
10. Dal menu **File del database delle chiavi**, fare clic su **Apri**. Viene visualizzata la finestra Apri.
11. Selezionare il database CMS della chiave di destinazione in cui viene importato il file pfx. Fare clic su **Apri**. Viene visualizzata la finestra Richiesta password.
12. Immettere la password del database delle chiavi e fare clic su **OK**. Viene visualizzata la finestra IBM Key Management. La barra del titolo mostra il nome del file database delle chiavi selezionato, che indica che il file è aperto e pronto.
13. Selezionare **Certificati personali** dall'elenco.
14. Se sono presenti certificati nella vista Certificati personali, attenersi alla seguente procedura:
    a. Fare clic su **Esporta / Importa chiave**. Viene visualizzata la finestra Esporta / Importa chiave.
    b. Selezionare **Importa** da Scegli tipo di azione.
15. Se non ci sono certificati nella vista Certificati personali, fare clic su **Importa**.
16. Selezionare il file PKCS12 .
17. Immettere il nome del file pfx come utilizzato nel passo 4. Fare clic su **OK**. Viene visualizzata la finestra Richiesta password.
18. Specificare la stessa password specificata quando è stato eliminato il certificato del firmatario. Fare clic su **OK**.
19. Viene visualizzata la finestra Modifica etichette (poiché dovrebbe essere disponibile un unico certificato per l'importazione). L'etichetta del certificato deve essere un UUID con formato `xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx`.
20. Per modificare l'etichetta, selezionare l'UUID dal pannello **Selezionare un'etichetta da cambiare** . L'etichetta verrà replicata nel campo **Immettere una nuova etichetta:** . Sostituire il testo dell'etichetta con quello del nome descrittivo eliminato nel passo 7 e fare clic su **Applica**. Il nome descrittivo deve essere nel formato `ibmwebspheremq`, seguito dal nome del gestore code o dall'ID di collegamento del client WebSphere MQ MQI in lettere minuscole.
21. Fare clic su **OK**. La finestra Modifica etichette viene ora rimossa e la finestra di IBM Key Management originale viene visualizzata nuovamente con i pannelli Certificati personali e Certificati firmatario aggiornati con il certificato personale correttamente etichettato.
22. Il certificato personale pfx viene ora importato nel database (di destinazione).

Non è possibile modificare un'etichetta del certificato utilizzando iKeycmd

Importazione da un file PKCS #7

Gli strumenti iKeyman e iKeycmd non supportano i file PKCS #7 (.p7b). Utilizzare lo strumento runmqckm per importare i certificati da un file PKCS #7 .

Utilizzare il comando seguente per aggiungere un certificato CA da un file PKCS #7 :

```
runmqckm -cert -add -db filename -pw password -type cms -file filename
-label label
```

-db *filename*	è il nome file completo del database delle chiavi CMS.
-pw *password*	è la password per il database delle chiavi.
-type *cms*	è il tipo di database delle chiavi.
-file *filename*	è il nome del file PKCS #7 .
-label *label*	è l'etichetta assegnata al certificato nel database di destinazione. Il primo certificato prende l'etichetta fornita. Tutti gli altri certificati, se presenti, sono etichettati con il relativo nome oggetto.

Utilizzare il seguente comando per importare un certificato personale da un file 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*	è il nome file completo del file contenente il certificato PKCS #7 .
-pw *password*	è la parola d'ordine per il certificato #7 PKCS.
-type *pkcs7*	è il tipo di file.
-target *filename*	è il nome del database delle chiavi di destinazione.
-target_pw *password*	è la password per il database delle chiavi di destinazione.
-target_type *cms*	è il tipo di database specificato da -target
-label *label*	è l'etichetta del certificato da importare.
-new_label *label*	è l'etichetta che il certificato verrà assegnato nel database di destinazione. Se si omette l'opzione -new_label , il valore predefinito è utilizzare lo stesso dell'opzione -label .

Eliminazione di un certificato da un repository delle chiavi su sistemi UNIX, Linux, and Windows

Utilizzare questa procedura per rimuovere i certificati personali o CA.

Utilizzo di iKeyman

Se è necessario gestire i certificati SSL in modo conforme a FIPS, utilizzare il comando runmqakm. iKeyman non fornisce un'opzione conforme a FIPS.

1. Avviare la GUI di iKeyman utilizzando il comando **strmqikm** (su sistemi UNIX Linux e Windows).
2. Dal menu **File del database delle chiavi**, fare clic su **Apri**. Viene visualizzata la finestra Apri.
3. Fare clic su **Tipo di database delle chiavi** e selezionare **CMS** (Certificate Management System).
4. Fare clic su **Sfoglia** per passare alla directory che contiene i file del database di chiavi.
5. Selezionare il file database delle chiavi da cui si desidera eliminare il certificato, ad esempio key.kdb.
6. Fare clic su **Apri**. Viene visualizzata la finestra Richiesta password.
7. Digitare la password impostata durante la creazione del database delle chiavi e fare clic su **OK**. Il nome del file del database di chiavi viene visualizzato nel campo **Nome file** .
8. Dall'elenco a discesa, selezionare **Certificati personali** o **Certificati firmatario**
9. Selezionare il certificato che si desidera eliminare.

10. Se non si dispone già di una copia del certificato e si desidera salvarlo, fare clic su **Esporta / Importa** ed esportarlo (consultare “Esportazione di un certificato personale da un repository delle chiavi” a pagina 132).
11. Con il certificato selezionato, fare clic su **Elimina**. Si apre la finestra Conferma.
12. Fare clic su **Sì**. Il campo **Certificati personali** non mostra più l'etichetta del certificato eliminato.

Utilizzo della riga di comando

Utilizzare i comandi seguenti per eliminare un certificato utilizzando iKeycmd o runmqakm:

- In UNIX, Linux e Windows:

```
runmqckm -cert -delete -db filename -pw password -label label
```

dove:

`-db` *filename*	è il nome file completo di un database di chiavi CMS.
`-pw` *password*	è la password per il database di chiavi CMS.
`-label` *label*	è l'etichetta allegata al certificato personale.
`-fips`	specifica che il comando viene eseguito in modalità FIPS. Con questa modalità viene disabilitato l'uso della libreria crittografica BSafe. Viene utilizzato solo il componente ICC e tale componente deve essere inizializzato correttamente in modalità FIPS. Quando è attiva la modalità FIPS, il componente ICC utilizza gli algoritmi convalidati con FIPS 140-2. Se il componente ICC non viene inizializzato in modalità FIPS, il comando **runmqakm** non riesce.

Generazione di password complesse per la protezione del repository delle chiavi

È possibile generare password complesse per la protezione del repository chiavi utilizzando il comando **runmqakm** .

È possibile utilizzare il comando **runmqakm** con i seguenti parametri per creare una password complessa:

```
runmqakm -random -create -length 14 -strong -fips
```

Quando si utilizza la password generata sul parametro **-pw** dei successivi comandi di gestione dei certificati, racchiudere sempre la password tra virgolette doppie. Sui sistemi UNIX and Linux , è necessario utilizzare anche un carattere barra retroversa per eseguire l'escape dei seguenti caratteri se vengono visualizzati nella stringa della password:

```
!   \   "    '
```

Quando si immette la password in risposta ad una richiesta da **runmqckm**, **runmqakm** o dalla GUI iKeyman , non è necessario racchiudere tra virgolette o caratteri di escape la password. Non è necessario perché la shell del sistema operativo non influisce sull'immissione dei dati in questi casi.

Configurazione dell'hardware crittografico su sistemi UNIX, Linux, and Windows

È possibile configurare l'hardware di crittografia per un gestore code o client in diversi modi.

È possibile configurare l'hardware di crittografia per un gestore code su sistemi UNIX, Linux o Windows utilizzando uno dei seguenti metodi:

- Utilizzare il comando ALTER QMGR MQSC con il parametro SSLCRYP, come descritto in ALTER QMGR.
- Utilizzare Esplora risorse di IBM WebSphere MQ per configurare l'hardware di crittografia sul sistema UNIX, Linux o Windows . Per ulteriori informazioni, fare riferimento alla guida in linea.

È possibile configurare l'hardware di crittografia per un client WebSphere MQ su sistemi UNIX, Linux o Windows utilizzando uno dei seguenti metodi:

- Impostare la variabile di ambiente MQSSLCRYP. I valori consentiti per MQSSLCRYP sono gli stessi del parametro SSLCRYP, come descritto in ALTER QMGR. Se si utilizza la versione GSK_PCS11 del parametro SSLCRYP, l'etichetta del token PKCS #11 deve essere specificata interamente in minuscolo.
- Impostare il campo **CryptoHardware** della struttura di opzioni di configurazione SSL, MQSCO, su una chiamata MQCONNX. Per ulteriori informazioni, consultare Panoramica per MQSCO.

Se è stato configurato l'hardware crittografico che utilizza l'interfaccia PKCS #11 utilizzando uno di tali metodi, è necessario memorizzare il certificato personale da utilizzare sui canali nel file di database delle chiavi per il token crittografico configurato. Ciò è descritto in "Gestione dei certificati sull'hardware PKCS #11" a pagina 139.

Gestione dei certificati sull'hardware PKCS #11
È possibile gestire certificati digitali su hardware crittografico che supporta l'interfaccia PKCS #11 .

Informazioni su questa attività

È necessario creare un database di chiavi per preparare l'ambiente IBM WebSphere MQ , anche se non si intende memorizzare i certificati CA (Certificate Authority), ma memorizzerà tutti i certificati sull'hardware di crittografia. Un database delle chiavi è necessario per il gestore code a cui fare riferimento nel relativo campo SSLKEYR o per l'applicazione client a cui fare riferimento nella variabile di ambiente MQSSLKEYR. Questo database delle chiavi è richiesto anche se si sta creando una richiesta di certificato.

Il database delle chiavi viene creato utilizzando la riga comandi o l'interfaccia utente **strmqikm** (iKeyman).

Procedura

Creare un database delle chiavi utilizzando la riga comandi.

1. Eseguire uno dei comandi riportati di seguito:
   - Su sistemi UNIX, Linux, and Windows :

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

   - Utilizzo di runmqakm:

     ```
     runmqakm -keydb -create -db filename -pw password -type cms
         -stash -fips -strong
     ```

   dove:

   -db *nomefile*
   Specifica il nome file completo di un database di chiavi CMS e deve avere estensione .kdb.

   -pw *password*
   Specifica la password per il database di chiavi CMS.

   -type *cms*
   Specifica il tipo di database. (Per IBM WebSphere MQ, deve essere cms.)

   -stash
   Salva la password del database delle chiavi in un file.

   -fips
   Disabilita l'utilizzo della libreria crittografica BSafe. Viene utilizzato solo il componente ICC e tale componente deve essere inizializzato correttamente in modalità FIPS. In modalità FIPS, il componente ICC utilizza algoritmi convalidati da FIPS 140-2. Se il componente ICC non viene inizializzato in modalità FIPS, il comando **runmqakm** non riesce.

   -forte
   Verifica che la parola d'ordine immessa soddisfi i requisiti minimi per la complessità della parola d'ordine. I requisiti minimi per una parola d'ordine sono i seguenti:

   - La password deve avere una lunghezza minima di 14 caratteri.

- La password deve contenere almeno un carattere minuscolo, un carattere maiuscolo e una cifra o un carattere speciale. I caratteri speciali includono l'asterisco (*), il simbolo del dollaro ($), il cancelletto (#) e il simbolo di percentuale (%). Uno spazio viene classificato come carattere speciale.
- Ogni carattere può essere presente al massimo tre volte in una password.
- Un massimo di due caratteri consecutivi nella password può essere identico.
- Tutti i caratteri sono nella serie di caratteri stampabili ASCII standard nell'intervallo 0x20 - 0x7E.

In alternativa, creare un database delle chiavi utilizzando l'interfaccia utente di **strmqikm** (iKeyman).

2. Sui sistemi UNIX and Linux , accedere come utente root. Sui sistemi Windows , accedere come Amministratore o come membro del gruppo MQM.
3. Avviare l'interfaccia utente iKeyman eseguendo il comando **strmqikm** .
4. Fare clic su **File database di chiavi** > **Apri**.
5. Fare clic su **Tipo di database delle chiavi** e selezionare **PKCS11Direct**.
6. Nel campo **Nome file** , immettere il nome del modulo per la gestione dell'hardware crittografico; ad esempio, `PKCS11_API.so`.

   Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, si noti che iKeycmd e iKeyman sono programmi a 64 bit. I moduli esterni richiesti per il supporto PKCS #11 verranno caricati in un processo a 64 bit, pertanto è necessario disporre di una libreria PKCS #11 a 64 bit installata per l'amministrazione dell'hardware di crittografia. Le piattaforme Windows e Linux x86 a 32 bit sono le uniche eccezioni, poiché i programmi iKeyman e iKeycmd sono a 32 bit su tali piattaforme.
7. Nel campo **Ubicazione** , immettere il percorso:

   - Su sistemi UNIX and Linux , potrebbe essere `/usr/lib/pksc11`, ad esempio.
   - Sui sistemi Windows , è possibile immettere il nome della libreria; ad esempio, `cryptoki`.

   Fare clic su **OK**. Viene visualizzata la finestra Apri token crittografico.
8. Nel campo **Password token crittografico** , immettere la parola d'ordine impostata durante la configurazione dell'hardware crittografico.
9. Se il proprio hardware di crittografia ha la capacità di contenere i certificati del firmatario richiesti per ricevere o importare un certificato personale, deselezionare entrambe le caselle di spunta del database delle chiavi secondario e continuare dal passo “13” a pagina 141.

   Se si richiede un database di chiavi CMS secondario per contenere i certificati del firmatario, selezionare **Apri file database di chiavi secondario esistente** oppure **Crea nuovo file database di chiavi secondario**.
10. Nel campo **Nome file** , immettere un nome file. Questo campo contiene già il testo `key.kdb`. Se il nome della radice è `key`, lasciare questo campo invariato. Se è stato specificato un nome di radice diverso, sostituire `key` con il proprio nome di radice. Non è necessario modificare il suffisso `.kdb` .
11. Nel campo **Ubicazione** , immettere il percorso, ad esempio:

    - Per un gestore code: `/var/mqm/qmgrs/QM1/ssl`
    - Per un client IBM WebSphere MQ MQI: `/var/mqm/ssl`

    Fare clic su **OK**. Viene visualizzata la finestra Richiesta password.
12. Immetti una password.

    Se è stato selezionato **Apri file database di chiavi secondario esistente** nel passo “9” a pagina 140, immettere una password nel campo **Password** .

    Se si seleziona **Crea nuovo file di database delle chiavi secondario** nel passo “9” a pagina 140, completare i seguenti passi secondari:

    a) Immettere una password nel campo **Password** e immetterla nuovamente nel campo **Conferma password** .

b) Selezionare **Stash della password in un file**. Tenere presente che se non si esegue lo stash della password, i tentativi di avviare i canali SSL non riescono perché non possono ottenere la password richiesta per accedere al file del database delle chiavi.

c) Fare clic su **OK**. Viene aperta una finestra, che conferma che la password si trova nel file `key.sth` (a meno che non sia stato specificato un nome radice diverso).

13. Fare clic su **OK**. Viene visualizzato il frame di contenuto del database delle chiavi.

Richiesta di un certificato personale per l'hardware PKCS #11
Utilizzare questa procedura per un gestore code o un client IBM WebSphere MQ MQI per richiedere un certificato personale per l'hardware di crittografia.

Utilizzo dell'interfaccia utente iKeyman

Informazioni su questa attività

Nota: WebSphere MQ non supporta gli algoritmi SHA-3 o SHA-5 . È possibile utilizzare i nomi degli algoritmi di firma digitale SHA384WithRSA e SHA512WithRSA perché entrambi gli algoritmi sono membri della famiglia SHA-2 .

I nomi degli algoritmi di firma digitale SHA3WithRSA e SHA5WithRSA sono obsoleti perché sono abbreviati rispettivamente in SHA384WithRSA e SHA512WithRSA .

Procedura

Per richiedere un certificato personale dall'interfaccia utente iKeyman , completare la seguente procedura:

1. Completare la procedura per gestire l'hardware crittografico. Consultare “Gestione dei certificati sull'hardware PKCS #11” a pagina 139.
2. Dal menu **Crea** , fare clic su **Nuova richiesta certificato**.

   Viene visualizzata la finestra Crea nuova chiave e richiesta certificato.
3. Nel campo **Etichetta chiave** , immettere le seguenti etichette:

   - Per un gestore code, immettere `ibmwebspheremq` seguito dal nome del gestore code modificato in minuscolo. Ad esempio, per un gestore code denominato `QM1`, immettere `ibmwebspheremqqm1`.
   - Per un IBM WebSphere MQ MQI client, immettere `ibmwebspheremq` seguito dall'ID utente di collegamento, tutto in minuscolo; ad esempio, `ibmwebspheremqmyuserid` .
4. Immettere i valori per **Nome comune** e **Organizzazione**e selezionare un **Paese** . Per i restanti campi facoltativi, accettare i valori predefiniti oppure immettere o selezionare nuovi valori.

   È possibile fornire solo un nome nel campo **Unità organizzativa** . Per ulteriori informazioni su questi campi, consultare “Nomi distinti” a pagina 11.
5. Nel campo **Immettere il nome di un file in cui memorizzare la richiesta di certificato** , accettare il valore predefinito `certreq.arm`oppure immettere un nuovo valore con un percorso completo.
6. Fare clic su **OK**.

   Viene aperta una finestra di conferma.
7. Fare clic su **OK**.

   L'elenco **Richieste di certificati personali** mostra l'etichetta della nuova richiesta di certificato personale creata. La richiesta di certificato viene memorizzata nel file scelto nel passo “5” a pagina 141.
8. Richiedere il nuovo certificato personale inviando il file a un'autorità di certificazione (CA) o copiando il file nel modulo di richiesta sul sito web per l'autorità di certificazione.

Procedura

Utilizzare i seguenti comandi per richiedere un certificato personale utilizzando il comando **runmqckm** o **runmqakm** :

- Utilizzo di **runmqckm**:

```
runmqckm -certreq -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -file filename -sig_alg algorithm
```

Invece di -dn *distinguished_name* , è possibile utilizzare -san_dsname *DNS_names* , -san_emailaddr *email_addresses* o -san_ipaddr *IP_addresses* .

- Utilizzo di runmqakm:

```
runmqakm -certreq -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -file filename -fips
        -sig_alg algorithm
```

dove:

-db *nomefile*
Specifica il nome file completo di un database di chiavi CMS.

-pw *password*
Specifica la password per il database di chiavi CMS.

-label *label*
Specifica l'etichetta chiave allegata al certificato.

-dn *nome_distinto*
Specifica il nome distinto X.500 racchiuso tra virgolette. È richiesto almeno un attributo. È possibile fornire più attributi OU e DC.

-size *dimensione_chiave*
Specifica la dimensione della chiave. Se si utilizza **runmqckm** , il valore può essere 512 o 1024. Se si sta utilizzando **runmqakm**, il valore può essere 512, 1024o 2048.

-file *nomefile*
Specifica il nome file per la richiesta di certificato.

-fips
specifica che il comando viene eseguito in modalità FIPS. Con questa modalità viene disabilitato l'uso della libreria crittografica BSafe. Viene utilizzato solo il componente ICC e tale componente deve essere inizializzato correttamente in modalità FIPS. In modalità FIPS, il componente ICC utilizza algoritmi convalidati da FIPS 140-2. Se il componente ICC non viene inizializzato in modalità FIPS, il comando **runmqakm** non riesce.

-sig_alg
Per **runmqckm**, specifica l'algoritmo di firma asimmetrico utilizzato per creare la coppia di chiavi della voce. Il valore può essere 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, SHAWithDSAo SHAWithRSA . Il valore predefinito è SHA1WithRSA

-sig_alg
In **runmqakm**, specifica l'algoritmo di hash utilizzato durante la creazione di una richiesta di certificato. Questo algoritmo di hash viene utilizzato per creare la firma associata alla richiesta di certificato appena creata. Il valore può essere 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. Il valore di default è SHA1WithRSA.

-san_dnsname ***nomi_DNS***
Specifica un elenco delimitato da virgole o da spazi di nomi DNS per la voce che si sta creando.

-san_emailaddr ***indirizzi_email***
Specifica un elenco delimitato da virgole o da spazi di indirizzi email per la voce che si sta creando.

-san_ipaddr ***indirizzi_IP***
Specifica un elenco delimitato da virgole o da spazi di indirizzi IP per la voce che si sta creando.

Importazione di un certificato personale nell'hardware PKCS #11
Utilizzare questa procedura per un gestore code o un client IBM WebSphere MQ MQI per importare un certificato personale nel proprio hardware di crittografia.

Utilizzo di iKeyman

Procedura

Per richiedere un certificato personale dall'interfaccia utente iKeyman , completare la seguente procedura:

1. Completare la procedura per gestire l'hardware crittografico. Consultare “Gestione dei certificati sull'hardware PKCS #11” a pagina 139.
2. Fare clic su **Ricevi**. Viene visualizzata la finestra Ricevi certificato da file.
3. Selezionare il **tipo di dati** del nuovo certificato personale; ad esempio, `Base64-encoded ASCII data` per un file con estensione `.arm` .
4. Immettere il nome file del certificato e l'ubicazione per il nuovo certificato personale oppure fare clic su **Sfoglia** per selezionare il nome e l'ubicazione.
5. Fare clic su **OK**. Se si dispone già di un certificato personale nel proprio database di chiavi, viene visualizzata una finestra in cui viene richiesto se si desidera impostare la chiave che si sta aggiungendo come chiave predefinita nel database.
6. Fare clic su **Sì** o **No**. Viene visualizzata la finestra Immettere un'etichetta.
7. Immettere un'etichetta.

   Ad esempio, è possibile utilizzare la stessa etichetta di quando è stato richiesto il certificato personale. Notare che l'etichetta deve essere nel formato IBM WebSphere MQ corretto:

   - Per un gestore code, `ibmwebspheremq` seguito dal nome del gestore code in minuscolo. Ad esempio, per un gestore code denominato QM1, l'etichetta è: `ibmwebspheremqqm1`.
   - Per un client IBM WebSphere MQ MQI, `ibmwebspheremq` seguito dall'ID utente di collegamento in minuscolo. Ad esempio, per un ID utente `MyUserID`, l'etichetta è: `ibmwebspheremqmyuserid`.
8. Fare clic su **OK**. L'elenco **Certificati personali** mostra l'etichetta del nuovo certificato personale aggiunto. Questa etichetta è formata aggiungendo l'etichetta del token crittografico prima dell'etichetta fornita.

Utilizzo della riga di comando

Procedura

Per richiedere un certificato personale da una riga comandi, completare la seguente procedura:

1. Aprire una finestra comandi configurata per il proprio ambiente.
2. Immettere il comando appropriato per il sistema operativo e la configurazione:

- Su sistemi Windows, UNIX and Linux , utilizzare uno dei seguenti comandi:

```
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
```

dove:

-file *nomefile*
Specifica il nome file completo del file contenente il certificato personale.

-crypto *percorso*
Specifica il percorso completo della libreria PKCS #11 fornita con l'hardware.

-tokenlabel *token hardware*
Specifica l'etichetta fornita alla parte di memoria dell'hardware crittografico durante l'installazione.

-pw *password_hardware*
Specifica la parola d'ordine per accedere all'hardware.

-format *formato_cert*
Specifica il formato del certificato. Il valore può essere `ascii` per ASCII con codifica Base64 oppure `binary` per i dati binari DER. Il valore predefinito è ASCII.

-fips
Specifica che il comando viene eseguito in modalità FIPS. Questa modalità disabilita l'utilizzo della libreria crittografica BSafe. Viene utilizzato solo il componente ICC e tale componente deve essere inizializzato correttamente in modalità FIPS. In modalità FIPS, il componente ICC utilizza algoritmi convalidati da FIPS 140-2. Se il componente ICC non viene inizializzato in modalità FIPS, il comando **runmqakm** non riesce.

Identificazione e autenticazione degli utenti

È possibile identificare e autenticare gli utenti utilizzando la struttura MQCSP o in diversi tipi di programma di uscita utente.

Utilizzo della struttura MQCSP

Specificare la struttura dei parametri di sicurezza della connessione MQCSP su una chiamata MQCONNX; questa struttura contiene un ID utente e password. Se necessario, è possibile modificare MQCSP in un'uscita di sicurezza.

Nota: OAM (object authority manager) non usa la password. Tuttavia, l'OAM esegue alcune operazioni limitate con l'ID utente, che potrebbero essere considerate una forma banale di autenticazione. Questi controlli arrestano l'adozione di un altro ID utente, se si utilizzano tali parametri nelle applicazioni.

Implementazione dell'identificazione e dell'autenticazione nelle uscite di sicurezza

Lo scopo principale di un'uscita di sicurezza è abilitare l'MCA ad ogni estremità di un canale per autenticare il relativo partner. Ad ogni estremità di un canale di messaggi e all'estremità server di un canale MQI, un MCA generalmente agisce per conto del gestore code a cui è connesso. All'estremità client di un canale MQI, un MCA generalmente agisce per conto dell'utente dell'applicazione client WebSphere MQ . In questa situazione, l'autenticazione reciproca avviene tra due gestori code o tra un gestore code e l'utente di un'applicazione client WebSphere MQ .

L'uscita di sicurezza fornita (l'uscita del canale SSPI) illustra come è possibile implementare l'autenticazione reciproca scambiando i token di autenticazione generati e quindi controllati da un server di autenticazione attendibile come Kerberos. Per ulteriori dettagli, vedere “Il programma di uscita del canale SSPI” a pagina 103.

L'autenticazione reciproca può essere implementata anche utilizzando la tecnologia PKI (Public Key Infrastructure). Ogni uscita di sicurezza genera alcuni dati casuali, li firma utilizzando la chiave privata del

gestore code o dell'utente che rappresenta e invia i dati firmati al partner in un messaggio di sicurezza. L'uscita di sicurezza del partner esegue l'autenticazione controllando la firma digitale utilizzando la chiave pubblica del gestore code o dell'utente. Prima di scambiare firme digitali, le uscite di sicurezza potrebbero dover concordare l'algoritmo per la generazione di un digest del messaggio, se più di un algoritmo è disponibile per l'uso.

Quando un'uscita di sicurezza invia i dati firmati al relativo partner, deve anche inviare alcuni mezzi per identificare il gestore code o l'utente che rappresenta. Potrebbe essere un DN (Distinguished Name) o anche un certificato digitale. Se viene inviato un certificato digitale, l'uscita di sicurezza partner può convalidare il certificato utilizzando la catena di certificati per il certificato CA root. Ciò garantisce la proprietà della chiave pubblica utilizzata per controllare la firma digitale.

L'uscita di sicurezza partner può convalidare un certificato digitale solo se ha accesso a un repository di chiavi che contiene i restanti certificati nella catena di certificati. Se non viene inviato un certificato digitale per il gestore code o l'utente, deve essere disponibile nel repository delle chiavi a cui ha accesso l'uscita di sicurezza del partner. L'uscita di sicurezza del partner non può controllare la firma digitale a meno che non riesca a trovare la chiave pubblica del firmatario.

SSL (Secure Sockets Layer) e TLS (Transport Layer Security) utilizzano tecniche PKI come quelle appena descritte. Per ulteriori informazioni su come SSL e TLS eseguono l'autenticazione, consultare “Concetti SSL (Secure Sockets Layer) e TLS (Transport Layer Security)” a pagina 14.

Se non è disponibile un server di autenticazione attendibile o un supporto PKI, è possibile utilizzare altre tecniche. Una tecnica comune, che può essere implementata nelle uscite di sicurezza, utilizza un algoritmo di chiave simmetrica.

Una delle uscite di sicurezza, l'uscita A, genera un numero casuale e lo invia in un messaggio di sicurezza all'uscita di sicurezza del partner, l'uscita B. L'uscita B codifica il numero utilizzando la relativa copia di una chiave nota solo alle due uscite di sicurezza. L'uscita B invia il numero crittografato all'uscita A in un messaggio di sicurezza con un secondo numero casuale generato dall'uscita B. L'uscita A verifica che il primo numero casuale sia stato codificato correttamente, codifica il secondo numero casuale utilizzando la sua copia della chiave e invia il numero codificato all'uscita B in un messaggio di sicurezza. L'uscita B verifica quindi che il secondo numero casuale sia stato codificato correttamente. Durante questo scambio, se una delle uscite di sicurezza non è soddisfatta dell'autenticità di altre, può indicare all'MCA di chiudere il canale.

Un vantaggio di questa tecnica è che nessuna chiave o parola d'ordine viene inviata attraverso la connessione di comunicazione durante lo scambio. Uno svantaggio è che non fornisce una soluzione al problema di come distribuire la chiave condivisa in modo sicuro. Una soluzione a questo problema è descritta in “Implementazione della riservatezza nei programmi di uscita utente” a pagina 225. Una tecnica simile viene utilizzata in SNA per l'autenticazione reciproca di due LU quando si collegano per formare una sessione. La tecnica è descritta in “Autenticazione a livello di sessione” a pagina 75.

Tutte le tecniche precedenti per l'autenticazione reciproca possono essere adattate per fornire l'autenticazione unidirezionale.

Implementazione dell'identificazione e dell'autenticazione nelle uscite dei messaggi

Quando un'applicazione inserisce un messaggio su una coda, il campo *UserIdentifier* nel descrittore del messaggio contiene un ID utente associato all'applicazione. Tuttavia, non sono presenti dati che possono essere utilizzati per autenticare l'ID utente. Questi dati possono essere aggiunti da un'uscita messaggio all'estremità di invio di un canale e controllati da un'uscita messaggio all'estremità di ricezione del canale. I dati di autenticazione possono essere, ad esempio, una password codificata o una firma digitale.

Questo servizio potrebbe essere più efficace se implementato a livello di applicazione. Il requisito di base è che l'utente dell'applicazione che riceve il messaggio sia in grado di identificare e autenticare l'utente dell'applicazione che ha inviato il messaggio. È pertanto naturale considerare l'implementazione di questo servizio a livello di applicazione. Per ulteriori informazioni, vedere “Associazione di identità nell'uscita API e nell'uscita incrociata API” a pagina 149.

Implementazione dell'identificazione e dell'autenticazione nell'uscita API e nell'uscita API - crossing

A livello di singolo messaggio, l'identificazione e autenticazione è un servizio che coinvolge due utenti, il mittente e il destinatario del messaggio. Il requisito di base è che l'utente dell'applicazione che riceve il messaggio sia in grado di identificare e autenticare l'utente dell'applicazione che ha inviato il messaggio. Si noti che il requisito è per l'autenticazione unidirezionale, non bidirezionale.

A seconda di come viene implementato, gli utenti e le relative applicazioni potrebbero dover interagire, o anche interagire, con il servizio. Inoltre, quando e come viene utilizzato il servizio potrebbe dipendere dalla posizione in cui si trovano gli utenti e le loro applicazioni e dalla natura delle applicazioni stesse. È quindi naturale considerare l'implementazione del servizio a livello di applicazione piuttosto che a livello di collegamento.

Se si considera l'implementazione di questo servizio a livello di link, potrebbe essere necessario risolvere i seguenti problemi:

- Su un canale di messaggi, come si applica il servizio solo ai messaggi che lo richiedono?
- Come consentire agli utenti e alle loro applicazioni di interagire o interagire con il servizio, se questo è un requisito?
- In una situazione multi - hop, in cui un messaggio viene inviato su più di un canale di messaggi sulla strada verso la sua destinazione, dove si richiamano i componenti del servizio?

Di seguito sono riportati alcuni esempi di come il servizio di identificazione e autenticazione può essere implementato a livello dell'applicazione. Il termine *uscita API* indica un'uscita API o un'uscita incrociata API.

- Quando un'applicazione inserisce un messaggio su una coda, un'uscita API può ottenere un token di autenticazione da un server di autenticazione attendibile come Kerberos. L'uscita API può aggiungere questo token ai dati dell'applicazione nel messaggio. Quando il messaggio viene richiamato dall'applicazione ricevente, una seconda uscita API può richiedere al server di autenticazione di autenticare il mittente controllando il token.
- Quando un'applicazione inserisce un messaggio su una coda, un'uscita API può accodare i seguenti elementi ai dati di applicazione nel messaggio:
  - Il certificato digitale del mittente
  - La firma digitale del mittente

  Se sono disponibili diversi algoritmi per la creazione di un digest del messaggio da utilizzare, l'uscita API può includere il nome dell'algoritmo utilizzato.

  Quando il messaggio viene richiamato dall'applicazione ricevente, una seconda uscita API può eseguire i seguenti controlli:

  - L'uscita API può convalidare il certificato digitale utilizzando la catena di certificati per il certificato CA root. Per eseguire questa operazione, l'uscita API deve avere accesso a un repository di chiavi che contenga i restanti certificati nella catena di certificati. Questo controllo garantisce che il mittente, identificato dal DN (Distinguished Name), sia il proprietario autentico della chiave pubblica contenuta nel certificato.
  - L'uscita API può controllare la firma digitale utilizzando la chiave pubblica contenuta nel certificato. Questo controllo autentica il mittente.

  Il DN (Distinguished Name) del mittente può essere inviato al posto dell'intero certificato digitale. In questo caso, è necessario che il repository delle chiavi contenga il certificato del mittente in modo che la seconda uscita API possa individuare la chiave pubblica del mittente. Un'altra possibilità è quella di inviare tutti i certificati nella catena di certificati.
- Quando un'applicazione inserisce un messaggio su una coda, il campo *UserIdentifier* nel descrittore del messaggio contiene un ID utente associato all'applicazione. L'ID utente può essere usato per identificare il mittente. Per abilitare l'autenticazione, un'uscita API può accodare alcuni dati, come una password codificata, ai dati dell'applicazione nel messaggio. Quando il messaggio viene richiamato

dall'applicazione ricevente, una seconda uscita API può autenticare l'ID utente utilizzando i dati trasmessi con il messaggio.

Questa tecnica potrebbe essere considerata sufficiente per i messaggi che hanno origine in un ambiente controllato e attendibile e in circostanze in cui non è disponibile un server di autenticazione attendibile o un supporto PKI.

Utenti privilegiati

Un utente privilegiato dispone di autorizzazioni di gestione complete per WebSphere MQ.

Oltre agli utenti elencati nella seguente tabella, i membri di qualsiasi gruppo con l'autorizzazione `+crt` per le code sono indirettamente amministratori. Allo stesso modo, qualsiasi utente che abbia l'autorizzazione `+set` sul gestore code e l'autorizzazione `+put` sulla coda dei comandi è un amministratore.

Non concedere questi privilegi agli utenti e alle applicazioni ordinari.

Tabella 13. Utenti privilegiati per piattaforma.

Una tabella di utenti privilegiati. Su Windows, SYSTEM, tutti i membri del gruppo mqm e tutti i membri del gruppo Administrators sono utenti privilegiati. Su sistemi UNIX and Linux , tutti i membri del gruppo mqm sono utenti privilegiati. Su IBM i, i profili (utenti) qmqm e qmqmadm, tutti i membri del gruppo qmqmadm e qualsiasi utente definito con l'impostazione *ALLOBJ sono utenti privilegiati.

Piattaforma	**Utenti privilegiati**
Sistemi Windows	• SYSTEM • Membri del gruppo mqm • Membri del gruppo Amministratori
Sistemi UNIX and Linux	• Membri del gruppo mqm

Identificazione e autenticazione degli utenti utilizzando la struttura MQCSP

È possibile specificare la struttura dei parametri di sicurezza della connessione MQCSP in una chiamata MQCONNX.

La struttura dei parametri di sicurezza della connessione MQCSP contiene un ID utente e una password, che il servizio di autorizzazione può utilizzare per identificare e autenticare l'utente.

Il componente del servizio di autorizzazione fornito con IBM WebSphere MQ è denominato OAM (Object Authority Manager). OAM autorizza gli utenti in base all'ID contenuto in MQCSP ma non convalida la password. È possibile implementare la convalida della password nel servizio di autorizzazione utilizzando le uscite concatenate con l'OAM o sostituendo l'OAM con un servizio di autorizzazione alternativo.

È possibile modificare MQCSP in un'uscita di sicurezza.

Implementazione dell'identificazione e dell'autenticazione nelle uscite di sicurezza

È possibile utilizzare un'uscita di sicurezza per implementare l'autenticazione unidirezionale o reciproca.

Lo scopo principale di un'uscita di sicurezza è abilitare l'MCA ad ogni estremità di un canale per autenticare il relativo partner. Ad ogni estremità di un canale di messaggi e all'estremità server di un canale MQI, un MCA generalmente agisce per conto del gestore code a cui è connesso. All'estremità client di un canale MQI, un MCA generalmente agisce per conto dell'utente dell'applicazione client WebSphere MQ MQI. In questa situazione, l'autenticazione reciproca avviene tra due gestori code o tra un gestore code e l'utente di un'applicazione client WebSphere MQ .

L'uscita di sicurezza fornita (l'uscita del canale SSPI) illustra come è possibile implementare l'autenticazione reciproca scambiando i token di autenticazione generati e quindi controllati da un server di autenticazione attendibile come Kerberos. Per ulteriori dettagli, consultare “Il programma di uscita del canale SSPI” a pagina 103.

L'autenticazione reciproca può essere implementata anche utilizzando la tecnologia PKI (Public Key Infrastructure). Ogni uscita di sicurezza genera alcuni dati casuali, li firma utilizzando la chiave privata del gestore code o dell'utente che rappresenta e invia i dati firmati al partner in un messaggio di sicurezza. L'uscita di sicurezza del partner esegue l'autenticazione controllando la firma digitale utilizzando la chiave pubblica del gestore code o dell'utente. Prima di scambiare firme digitali, le uscite di sicurezza potrebbero dover concordare l'algoritmo per la generazione di un digest del messaggio, se più di un algoritmo è disponibile per l'uso.

Quando un'uscita di sicurezza invia i dati firmati al relativo partner, deve anche inviare alcuni mezzi per identificare il gestore code o l'utente che rappresenta. Potrebbe essere un DN (Distinguished Name) o anche un certificato digitale. Se viene inviato un certificato digitale, l'uscita di sicurezza partner può convalidare il certificato utilizzando la catena di certificati per il certificato CA root. Ciò garantisce la proprietà della chiave pubblica utilizzata per controllare la firma digitale.

L'uscita di sicurezza partner può convalidare un certificato digitale solo se ha accesso a un repository di chiavi che contiene i restanti certificati nella catena di certificati. Se non viene inviato un certificato digitale per il gestore code o l'utente, deve essere disponibile nel repository delle chiavi a cui ha accesso l'uscita di sicurezza del partner. L'uscita di sicurezza del partner non può controllare la firma digitale a meno che non riesca a trovare la chiave pubblica del firmatario.

SSL (Secure Sockets Layer) e TLS (Transport Layer Security) utilizzano tecniche PKI come quelle appena descritte. Per ulteriori informazioni su come SSL (Secure Sockets Layer) esegue l'autenticazione, consultare “Concetti SSL (Secure Sockets Layer) e TLS (Transport Layer Security)” a pagina 14.

Se non è disponibile un server di autenticazione attendibile o un supporto PKI, è possibile utilizzare altre tecniche. Una tecnica comune, che può essere implementata nelle uscite di sicurezza, utilizza un algoritmo di chiave simmetrica.

Una delle uscite di sicurezza, l'uscita A, genera un numero casuale e lo invia in un messaggio di sicurezza all'uscita di sicurezza del partner, l'uscita B. L'uscita B codifica il numero utilizzando la relativa copia di una chiave nota solo alle due uscite di sicurezza. L'uscita B invia il numero crittografato all'uscita A in un messaggio di sicurezza con un secondo numero casuale generato dall'uscita B. L'uscita A verifica che il primo numero casuale sia stato codificato correttamente, codifica il secondo numero casuale utilizzando la sua copia della chiave e invia il numero codificato all'uscita B in un messaggio di sicurezza. L'uscita B verifica quindi che il secondo numero casuale sia stato codificato correttamente. Durante questo scambio, se una delle uscite di sicurezza non è soddisfatta dell'autenticità di altre, può indicare all'MCA di chiudere il canale.

Un vantaggio di questa tecnica è che nessuna chiave o parola d'ordine viene inviata attraverso la connessione di comunicazione durante lo scambio. Uno svantaggio è che non fornisce una soluzione al problema di come distribuire la chiave condivisa in modo sicuro. Una soluzione a questo problema è descritta in “Implementazione della riservatezza nei programmi di uscita utente” a pagina 225. Una tecnica simile viene utilizzata in SNA per l'autenticazione reciproca di due LU quando si collegano per formare una sessione. La tecnica è descritta in “Autenticazione a livello di sessione” a pagina 75.

Tutte le tecniche precedenti per l'autenticazione reciproca possono essere adattate per fornire l'autenticazione unidirezionale.

Mappature di identità nelle uscite del messaggio

È possibile utilizzare le uscite dei messaggi per elaborare le informazioni per autenticare un ID utente, anche se potrebbe essere meglio implementare l'autenticazione a livello dell'applicazione.

Quando un'applicazione inserisce un messaggio su una coda, il campo *UserIdentifier* nel descrittore del messaggio contiene un ID utente associato all'applicazione. Tuttavia, non sono presenti dati che possono essere utilizzati per autenticare l'ID utente. Questi dati possono essere aggiunti da un'uscita messaggio

all'estremità di invio di un canale e controllati da un'uscita messaggio all'estremità di ricezione del canale. I dati di autenticazione possono essere, ad esempio, una password codificata o una firma digitale.

Questo servizio potrebbe essere più efficace se implementato a livello di applicazione. Il requisito di base è che l'utente dell'applicazione che riceve il messaggio sia in grado di identificare e autenticare l'utente dell'applicazione che ha inviato il messaggio. È pertanto naturale considerare l'implementazione di questo servizio a livello di applicazione. Per ulteriori informazioni, vedere "Associazione di identità nell'uscita API e nell'uscita incrociata API" a pagina 149.

Associazione di identità nell'uscita API e nell'uscita incrociata API

Un'applicazione che riceve un messaggio deve essere in grado di identificare e autenticare l'utente dell'applicazione che ha inviato il messaggio. Questo servizio è generalmente implementato al meglio a livello di applicazione. Le uscite API possono implementare il servizio in diversi modi.

A livello di singolo messaggio, l'identificazione e autenticazione è un servizio che coinvolge due utenti, il mittente e il destinatario del messaggio. Il requisito di base è che l'utente dell'applicazione che riceve il messaggio sia in grado di identificare e autenticare l'utente dell'applicazione che ha inviato il messaggio. Si noti che il requisito è per l'autenticazione unidirezionale, non bidirezionale.

A seconda di come viene implementato, gli utenti e le relative applicazioni potrebbero dover interagire, o anche interagire, con il servizio. Inoltre, quando e come viene utilizzato il servizio potrebbe dipendere dalla posizione in cui si trovano gli utenti e le loro applicazioni e dalla natura delle applicazioni stesse. È quindi naturale considerare l'implementazione del servizio a livello di applicazione piuttosto che a livello di collegamento.

Se si considera l'implementazione di questo servizio a livello di link, potrebbe essere necessario risolvere i seguenti problemi:

- Su un canale di messaggi, come si applica il servizio solo ai messaggi che lo richiedono?
- Come consentire agli utenti e alle loro applicazioni di interagire o interagire con il servizio, se questo è un requisito?
- In una situazione multi - hop, in cui un messaggio viene inviato su più di un canale di messaggi sulla strada verso la sua destinazione, dove si richiamano i componenti del servizio?

Di seguito sono riportati alcuni esempi di come il servizio di identificazione e autenticazione può essere implementato a livello dell'applicazione. Il termine *uscita API* indica un'uscita API o un'uscita incrociata API.

- Quando un'applicazione inserisce un messaggio su una coda, un'uscita API può ottenere un token di autenticazione da un server di autenticazione attendibile come Kerberos. L'uscita API può aggiungere questo token ai dati dell'applicazione nel messaggio. Quando il messaggio viene richiamato dall'applicazione ricevente, una seconda uscita API può richiedere al server di autenticazione di autenticare il mittente controllando il token.
- Quando un'applicazione inserisce un messaggio su una coda, un'uscita API può accodare i seguenti elementi ai dati di applicazione nel messaggio:
  - Il certificato digitale del mittente
  - La firma digitale del mittente

  Se sono disponibili diversi algoritmi per la creazione di un digest del messaggio da utilizzare, l'uscita API può includere il nome dell'algoritmo utilizzato.

  Quando il messaggio viene richiamato dall'applicazione ricevente, una seconda uscita API può eseguire i seguenti controlli:

  - L'uscita API può convalidare il certificato digitale utilizzando la catena di certificati per il certificato CA root. Per eseguire questa operazione, l'uscita API deve avere accesso a un repository di chiavi che contenga i restanti certificati nella catena di certificati. Questo controllo garantisce che il mittente, identificato dal DN (Distinguished Name), sia il proprietario autentico della chiave pubblica contenuta nel certificato.

– L'uscita API può controllare la firma digitale utilizzando la chiave pubblica contenuta nel certificato. Questo controllo autentica il mittente.

Il DN (Distinguished Name) del mittente può essere inviato al posto dell'intero certificato digitale. In questo caso, è necessario che il repository delle chiavi contenga il certificato del mittente in modo che la seconda uscita API possa individuare la chiave pubblica del mittente. Un'altra possibilità è quella di inviare tutti i certificati nella catena di certificati.

- Quando un'applicazione inserisce un messaggio su una coda, il campo *UserIdentifier* nel descrittore del messaggio contiene un ID utente associato all'applicazione. L'ID utente può essere usato per identificare il mittente. Per abilitare l'autenticazione, un'uscita API può accodare alcuni dati, come una password codificata, ai dati dell'applicazione nel messaggio. Quando il messaggio viene richiamato dall'applicazione ricevente, una seconda uscita API può autenticare l'ID utente utilizzando i dati trasmessi con il messaggio.

  Questa tecnica potrebbe essere considerata sufficiente per i messaggi che hanno origine in un ambiente controllato e attendibile e in circostanze in cui non è disponibile un server di autenticazione attendibile o un supporto PKI.

Utilizzo dei certificati revocati

I certificati digitali possono essere revocati dalle autorità di certificazione. È possibile controllare lo stato di revoca dei certificati utilizzando OCSP o CRL sui server LDAP, a seconda della piattaforma.

Durante l'handshake SSL, i partner di comunicazione si autenticano reciprocamente con i certificati digitali. L'autenticazione può includere una conferma che il certificato ricevuto sia ancora ancora sicuro. Le autorità di certificazione (CA) revocano i certificati per vari motivi, tra cui:

- Il proprietario è stato spostato in un'organizzazione diversa
- La chiave privata non è più segreta

Le CA pubblicano i certificati personali revocati in un CRL (Certificate Revocation List). I certificati AC revocati vengono pubblicati in un elenco ARL (Authority Revocation List).

Su sistemi UNIX, Linux e Windows , il supporto SSL WebSphere MQ verifica la presenza di certificati revocati utilizzando OCSP (Online Certificate Status Protocol) o CRL e ARL su server LDAP (Lightweight Directory Access Protocol). OCSP è il metodo preferito. Le classi IBM WebSphere MQ classes for Java e IBM WebSphere MQ classes for JMS non possono utilizzare le informazioni OCSP in un file tabella di definizione del canale client. Tuttavia, è possibile configurare OCSP come descritto nella sezione Utilizzo di Online Certificate Protocol.

Su z / Os e IBM i WebSphere MQ il supporto SSL verifica la presenza di certificati revocati utilizzando solo CRL e ARL su server LDAP.

Per ulteriori informazioni sul certificato

Autorizzazioni, consultare “Certificati digitali” a pagina 9.

OCSP e certificati revocati

IBM WebSphere MQ determina quale responder OCSP (Online Certificate Status Protocol) utilizzare e gestisce la risposta ricevuta. Potrebbero essere necessarie delle azioni per rendere accessibile il responder OCSP.

Nota: Queste informazioni si applicano solo a WebSphere MQ nei sistemi Windows, UNIX and Linux.

Per verificare lo stato di revoca di un certificato digitale utilizzando OCSP, WebSphere MQ può utilizzare due metodi per determinare quale responder OCSP contattare:

- Utilizzando l'estensione del certificato AuthorityInfoAccess (AIA) nel certificato da controllare.
- Utilizzando un URL specificato in un oggetto delle informazioni di autenticazione o specificato da un'applicazione client.

Un URL specificato in un oggetto delle informazioni di autenticazione o da un'applicazione client è prioritario rispetto a un URL in un'estensione del certificato AIA.

Se l'URL del responder OCSP si trova dietro un firewall, riconfigurare il firewall in modo da consentire l'accesso al responder OCSP o impostare un server proxy OCSP. Specificare il nome del server proxy utilizzando la variabile SSLHTTPProxyName nella stanza SSL. Nei sistemi client, è anche possibile specificare il nome del server proxy utilizzando la variabile di ambiente MQSSLPROXY. Per ulteriori dettagli, consultare le informazioni correlate.

Se non è importante sapere se i certificati TLS o SSL siano revocati, magari perché ci si trova in un ambiente di prova, è possibile impostare OCSPCheckExtensions su NO nella stanza SSL. Se si imposta questa variabile, viene ignorata qualsiasi estensione del certificato AIA. Questa soluzione non è probabilmente accettabile in un ambiente di produzione, dove non si desidera consentire l'accesso ad utenti che presentano certificati revocati.

La chiamata per accedere a OCSP può comportare uno dei tre seguenti risultati:

Buono
Il certificato è valido.

Revocato
Il certificato è revocato.

Sconosciuto
Questo risultato può essere emesso per uno dei seguenti motivi:

- IBM WebSphere MQ non può accedere al responder OCSP.
- Il responder OCSP ha inviato una risposta, ma WebSphere MQ non è in grado di verificare la firma digitale di tale risposta.
- Il responder OCSP ha inviato una risposta che indica che non dispone di dati di revoca per il certificato.

Se IBM WebSphere MQ riceve un esito OCSP di `Sconosciuto`, il suo comportamento dipende dall'impostazione dell'attributo OCSPAuthentication. Per i gestori code, questo attributo è contenuto nella stanza SSL del file `qm.ini` per i sistemi UNIX and Linux o nel registro di Windows. Può essere impostato utilizzando Esplora risorse di IBM WebSphere MQ . Per i client, è contenuto nella stanza SSL del file di configurazione client.

Se si riceve un risultato `Sconosciuto` e OCSPAuthentication è impostato su REQUIRED (valore predefinito), WebSphere MQ rifiuta la connessione ed emette un messaggio di errore di tipo AMQ9716. Se i messaggi di evento SSL del gestore code sono abilitati, viene generato un messaggio di evento SSL di tipo MQRC_CHANNEL_SSL_ERROR con ReasonQualifier impostato su MQRQ_SSL_HANDSHAKE_ERROR.

Se si riceve un risultato `Sconosciuto` e OCSPAuthentication è impostato su OPTIONAL, WebSphere MQ consente l'avvio del canale SSL e non vengono generate avvertenze o messaggi di evento SSL.

Se viene ricevuto un esito di `Sconosciuto` e OCSPAuthentication è impostato su WARN, viene avviato il canale SSL ma IBM WebSphere MQ genera un messaggio di avvertenza di tipo AMQ9717 nel log degli errori. Se i messaggi di evento SSL del gestore code sono abilitati, viene generato un messaggio di evento SSL di tipo MQRC_CHANNEL_SSL_WARNING con ReasonQualifier impostato su MQRQ_SSL_UNKNOWN_REVOCATION.

Firma digitale delle risposte OCSP

Un responder OCSP può firmare le proprie risposte in uno dei seguenti tre modi. Il responder informa l'utente del metodo utilizzato.

- La risposta OCSP può essere firmata digitalmente utilizzando lo stesso certificato CA che ha emesso il certificato che si sta controllando. In questo caso, non è necessario impostare altri certificati; i passi già completati per stabilire la connettività SSL sono sufficienti per verificare la risposta OCSP.
- La risposta OCSP può essere firmata digitalmente utilizzando un altro certificato firmato dalla stessa CA (Certificate Authority) che ha emesso il certificato che si sta controllando. In questo caso, il certificato

di firma viene inviato insieme alla risposta OCSP. Il certificato emesso dal responder OCSP deve avere una Extended Key Usage Extension impostata su `id-kp-OCSPSigning` per poter essere considerato sicuro per questo scopo. Poiché la risposta OCSP viene inviata con il certificato che l'ha firmata (e tale certificato è firmato da una CA già considerata sicura per la connettività SSL), non sono richieste impostazioni aggiuntive per il certificato.

- La risposta OCSP può essere firmata digitalmente utilizzando un altro certificato non correlato direttamente al certificato che si sta controllando. In questo caso, la risposta OCSP viene firmata da un certificato emesso dallo stesso responder OCSP. È necessario aggiungere una copia del certificato del responder OCSP al database delle chiavi del client o del gestore code che esegue la verifica OCSP; consultare "Aggiunta di un certificato CA (o la parte pubblica di un certificato autofirmato) in un repository delle chiavi su sistemi UNIX, Linux, and Windows" a pagina 131. Quando viene aggiunto un certificato CA, per impostazione predefinita viene aggiunto come root sicura, che rappresenta l'impostazione richiesta in questo contesto. Se questo certificato non viene aggiunto, WebSphere MQ non è in grado di verificare la firma digitale della risposta OCSP e la verifica OCSP produce un risultato Sconosciuto, che potrebbe causare la chiusura del canale da parte di IBM WebSphere MQ, a seconda del valore di OCSPAuthentication.

OCSP (Online Certificate Status Protocol) nelle applicazioni client Java e JMS

A causa di una limitazione dell'API Java, WebSphere MQ può utilizzare il controllo sulla revoca dei certificati OCSP (Online Certificate Status Protocol) per i socket sicuri SSL e TLS solo quando OCSP viene abilitato per l'intero processo JVM (Java virtual machine). Esistono due modi per abilitare OCSP per tutti i socket sicuri nella JVM:

- Modificare il file java.security JRE per includere le impostazioni di configurazione OCSP mostrate nella Tabella 1 e riavviare l'applicazione.
- Utilizzare l'API java.security.Security.setProperty(), soggetta a eventuali politiche di Java Security Manager in vigore.

Come minimo, è necessario specificare uno dei valori ocsp.enable e ocsp.responderURL.

Nome proprietà	Descrizione
ocsp.enable	Il valore di questa proprietà è `true` o `false`. Se è `true`, la verifica OCSP viene abilitata quando si esegue il controllo sulla revoca dei certificati; se è `false` o non è impostato, la verifica OCSP è disabilitata.
ocsp.responderURL	Il valore di questa proprietà è un URL che identifica l'ubicazione del responder OCSP. Di seguito è riportato un esempio: `ocsp.responderURL=http://ocsp.example.net:80`. Per impostazione predefinita, l'ubicazione del responder OCSP è determinata implicitamente dal certificato che viene convalidato. La proprietà viene utilizzata quando l'estensione Authority Information Access (definita in RFC 3280) non è presente nel certificato o quando richiede la sovrascrittura.
ocsp.responderCertSubjectName	Il valore di questa proprietà è il nome oggetto del certificato del responder OCSP. Di seguito è riportato un esempio: `ocsp.responderCertSubjectName="CN=OCSP Responder, O=XYZ Corp"`. Per impostazione predefinita, il certificato del responder OCSP è quello dell'emittente del certificato che viene convalidato. Questa proprietà identifica il certificato del responder OCSP quando non viene applicato il valore predefinito. Il suo valore è un DN (distinguished name) stringa (definito in RFC 2253) che identifica un certificato nella serie di certificati forniti durante la convalida del percorso del certificato. Nei casi in cui il solo nome oggetto non sia sufficiente a identificare univocamente il certificato, è necessario

Nome proprietà	Descrizione
	utilizzare entrambe le proprietà ocsp.responderCertIssuerName e ocsp.responderCertSerialNumber. Quando questa proprietà è impostata, le proprietà ocsp.responderCertIssuerName e ocsp.responderCertSerialNumber vengono ignorate.
ocsp.responderCertIssuerName	Il valore di questa proprietà è il nome emittente del certificato del responder OCSP. Di seguito è riportato un esempio: `ocsp.responderCertIssuerName="CN=Enterprise CA, O=XYZ Corp"`. Per impostazione predefinita, il certificato del responder OCSP è quello dell'emittente del certificato che viene convalidato. Questa proprietà identifica il certificato del responder OCSP quando non viene applicato il valore predefinito. Il suo valore è un DN (distinguished name) stringa (definito in RFC 2253) che identifica un certificato nella serie di certificati forniti durante la convalida del percorso del certificato. Quando questa proprietà è impostata, è necessario impostare anche la proprietà ocsp.responderCertSerialNumber. Questa proprietà viene ignorata quando è impostata la proprietà ocsp.responderCertSubjectName.
ocsp.responderCertSerialNumber	Il valore di questa proprietà è il numero di serie del certificato del responder OCSP. Di seguito è riportato un esempio: `ocsp.responderCertSerialNumber=2A:FF:00`. Per impostazione predefinita, il certificato del responder OCSP è quello dell'emittente del certificato che viene convalidato. Questa proprietà identifica il certificato del responder OCSP quando non viene applicato il valore predefinito. Questo valore è una stringa di cifre esadecimali (potrebbero essere presenti separatori due punti o spazio) che identifica un certificato nella serie di certificati forniti durante la convalida del percorso del certificato. Quando questa proprietà è impostata, è necessario impostare anche la proprietà ocsp.responderCertIssuerName. Questa proprietà viene ignorata quando è impostata la proprietà ocsp.responderCertSubjectName.

Prima di abilitare OCSP in questo modo, vi sono alcune considerazioni di cui tenere conto:

- L'impostazione della configurazione OCSP interessa tutti i socket sicuri nel processo JVM. In alcuni casi, questa configurazione può avere effetti secondari indesiderati quando la JVM viene condivisa con un altro codice applicazione che utilizza i socket sicuri SSL o TLS. Verificare che la configurazione OCSP scelta sia adeguata per tutte le applicazioni in esecuzione nella stessa JVM.
- L'applicazione della manutenzione al JRE potrebbe sovrascrivere il file java.security. Prestare attenzione quando si applicano le correzioni temporanee Java e la manutenzione del prodotto per evitare di sovrascrivere il file java.security. Potrebbe essere necessario riapplicare le modifiche a java.security dopo aver applicato la manutenzione. Per questo motivo, si potrebbe considerare di impostare la configurazione OCSP utilizzando invece l'API java.security.Security.setProperty().
- L'abilitazione della verifica OCSP ha effetto solo se è abilitato anche il controllo sulle revoche. Il controllo sulle revoche viene abilitato dal metodo `PKIXParameters.setRevocationEnabled()`.
- Se si utilizza l'intercettore Java AMS descritto in Abilitazione della verifica OCSP negli intercettori nativi, evitare di utilizzare una configurazione OCSP java.security in quanto causa dei conflitti con la configurazione OCSP AMS nel file di configurazione del keystore.

Utilizzo dei CRL (Certificate Revocation Lists) e degli elenchi di revoca dell'autorità

Il supporto di WebSphere MQper CRL e ARL varia a seconda della piattaforma.

Il supporto CRL e ARL su ciascuna piattaforma è il seguente:

- Su z/OS, System SSL supporta i CRL e gli ARL memorizzati nei server LDAP dal prodotto Tivoli Public Key Infrastructure.
- Su altre piattaforme, il supporto CRL e ARL è conforme ai suggerimenti del profilo CRL PKIX X.509 V2 .

WebSphere MQ conserva una cache di CRL e ARL a cui è stato effettuato l'accesso nelle 12 ore precedenti.

Quando un gestore code o un client WebSphere MQ MQI riceve un certificato, controlla il CRL per confermare che il certificato è ancora valido. WebSphere MQ esegue prima il check-in della cache, se è presente una cache. Se il CRL non è presente nella cache, WebSphere MQ interroga le ubicazioni del server CRL LDAP nell'ordine in cui si verificano nell'elenco nomi degli oggetti delle informazioni di autenticazione specificati dall'attributo *SSLCRLNamelist* , fino a quando WebSphere MQ non trova un CRL disponibile. Se l'elenco nomi non è specificato o è specificato con un valore vuoto, i CRL non vengono controllati.

Per ulteriori informazioni su LDAP, consultare Using lightweight directory access protocol services with WebSphere MQ for Windows.

Impostazione dei server LDAP

Configurare la struttura ad albero delle informazioni della directory LDAP per riflettere la gerarchia dei DN (Distinguished Name) delle CA. Eseguire questa operazione utilizzando i file LDAP Data Interchange Format.

Configurare la struttura DIT (Directory Information Tree) LDAP per utilizzare la gerarchia corrispondente ai DN (Distinguished Name) delle CA che emettono i certificati e i CRL. È possibile impostare la struttura DIT con un file che utilizza LDIF (LDAP Data Interchange Format). È inoltre possibile utilizzare i file LDIF per aggiornare una directory.

I file LDIF sono file di testo ASCII che contengono le informazioni richieste per definire gli oggetti all'interno di un indirizzario LDAP. I file LDIF contengono una o più voci, ognuna delle quali comprende un DN (Distinguished Name), almeno una definizione di classe oggetto e, facoltativamente, più definizioni di attributo.

L'attributo di `certificateRevocationList;binary` contiene un elenco, in formato binario, di certificati utente revocati. L'attributo `authorityRevocationList;binary` contiene un elenco binario di certificati CA che sono stati revocati. Per l'uso con WebSphere MQ SSL, i dati binari per questi attributi devono essere conformi al formato DER (Definite Encoding Rules). Per ulteriori informazioni sui file LDIF, fare riferimento alla documentazione fornita con il proprio server LDAP.

Figura 12 a pagina 154 mostra un file LDIF di esempio che potresti creare come input per il tuo server LDAP per caricare i CRL e gli ARL emessi da CA1, che è un'autorità di certificazione immaginaria con il DN (Distinguished Name) "CN=CA1, OU=Test, O=IBM, C=GB", configurato dall'organizzazione Test in 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. File LDIF di esempio per una CA (Certificate Authority). Ciò può variare da implementazione a implementazione.

Figura 13 a pagina 155 mostra la struttura DIT creata dal server LDAP quando si carica il file LDIF di esempio mostrato in Figura 12 a pagina 154 insieme a un file simile per CA2, un'autorità di certificazione immaginaria impostata dall'organizzazione PKI, anche all'interno di IBM.

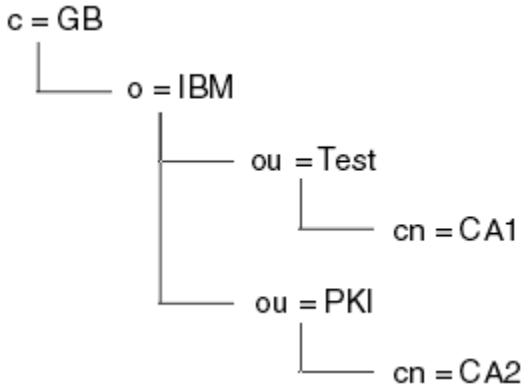


Figura 13. Esempio di una struttura ad albero di informazioni della directory LDAP

WebSphere MQ controlla i CRL e gli ARL.

Nota: Accertarsi che l'elenco di controllo accessi per il server LDAP consenta agli utenti autorizzati di leggere, ricercare e confrontare le voci che contengono i CRL e gli ARL. WebSphere MQ accede al server LDAP utilizzando le proprietà LDAPUSER e LDAPPWD dell'oggetto AUTHINFO.

Configurazione e aggiornamento dei server LDAP
Utilizzare questa procedura per configurare o aggiornare il server LDAP.

1. Ottenere i CRL e gli ARL in formato DER dall'autorità di certificazione o dalle autorità.
2. Utilizzando un editor di testo o lo strumento fornito con il proprio server LDAP, creare uno o più file LDIF che contengono il DN (Distinguished Name) della CA e le definizioni della classe di oggetti richieste. Copiare i dati del formato DER nel file LDIF come valori dell'attributo `certificateRevocationList;binary` per i CRL, dell'attributo `authorityRevocationList;binary` per gli ARL o entrambi.
3. Avviare il server LDAP.
4. Aggiungere le voci dal file o dai file LDIF creati al passo "2" a pagina 155.

Dopo aver configurato il server CRL LDAP, verificare che sia impostato correttamente. Per prima cosa, prova a utilizzare un certificato che non sia revocato sul canale e controlla che il canale si avvii correttamente. Quindi utilizzare un certificato revocato e verificare che il canale non venga avviato.

Ottenere frequentemente i CRL aggiornati dalle autorità di certificazione. Considerare la possibilità di eseguire questa operazione sui server LDAP ogni 12 ore.

Accesso a CRL e ARL con un gestore code

Un gestore code è associato a uno o più oggetti delle informazioni di autenticazione, che contengono l'indirizzo di un server CRL LDAP.

Si noti che in questa sezione, le informazioni sui CRL (Certificate Revocation Lists) si applicano anche agli ARL (Authority Revocation Lists).

Si indica al gestore code come accedere ai CRL fornendo al gestore code gli oggetti delle informazioni di autenticazione, ognuno dei quali contiene l'indirizzo di un server CRL LDAP. Gli oggetti delle informazioni di autenticazione sono contenuti in un elenco nomi, specificato nell'attributo del gestore code *SSLCRLNamelist* .

Nel seguente esempio, MQSC viene utilizzato per specificare i parametri:

1. Definire gli oggetti delle informazioni di autenticazione utilizzando il comando DEFINE AUTHINFO MQSC, con il parametro AUTHTYPE impostato su CRLLDAP.

Il valore CRLLDAP per il parametro AUTHTYPE indica che si accede ai CRL sui server LDAP. Ciascun oggetto delle informazioni di autenticazione con tipo CRLLDAP creato contiene l'indirizzo di un server LDAP. Quando si dispone di più di un oggetto delle informazioni di autenticazione, i server LDAP a cui puntano *devono* contenere informazioni identiche. Ciò fornisce la continuità del servizio se uno o più server LDAP hanno esito negativo.

Su tutte le piattaforme, l'ID utente e la password vengono inviati al server LDAP non codificati.

2. Utilizzando il comando DEFINE NAMELIST MQSC, definire un elenco nomi per i nomi degli oggetti delle informazioni di autenticazione.
3. Utilizzando il comando ALTER QMGR MQSC, fornire l'elenco nomi al gestore code. Ad esempio:

```
ALTER QMGR SSLCRLNL(sslcrlnlname)
```

dove `sslcrlnlname` è l'elenco nomi degli oggetti delle informazioni di autenticazione.

Questo comando imposta un attributo del gestore code denominato *SSLCRLNamelist*. Il valore iniziale del gestore code per questo attributo è vuoto.

È possibile aggiungere fino a 10 connessioni a server LDAP alternativi all'elenco nomi, per garantire la continuità del servizio in caso di errore di uno o più server LDAP. Tenere presente che i server LDAP *devono* contenere informazioni identiche.

Accesso a CRL e ARL mediante IBM WebSphere MQ Explorer
È possibile utilizzare IBM WebSphere MQ Explorer per indicare a un gestore code come accedere ai CRL.

Si noti che in questa sezione, le informazioni sui CRL (Certificate Revocation Lists) si applicano anche agli ARL (Authority Revocation Lists).

Utilizzare la procedura riportata di seguito per impostare una connessione LDAP a una CRL:

1. Assicurarsi di aver avviato il gestore code.
2. Fare clic con il pulsante destro del mouse sulla cartella **Informazioni di autenticazione** e fare clic su **Nuovo -> Informazioni di autenticazione**. Nel foglio delle proprietà che si apre:
   a. Nella prima pagina **Crea informazioni di autenticazione** , immettere un nome per l'oggetto CRL (LDAP).
   b. Nella pagina **Generale** di **Modifica proprietà**, selezionare il tipo di connessione. Facoltativamente, è possibile immettere una descrizione.
   c. Selezionare la pagina **CRL (LDAP)** di **Modifica proprietà**.
   d. Immettere il nome del server LDAP come nome di rete o indirizzo IP.
   e. Se il server richiede i dettagli di accesso, fornire un ID utente e, se necessario, una password.
   f. Fare clic su **OK**.
3. Fare clic con il tasto destro del mouse sulla cartella **Elenchi nomi** e selezionare **Nuovo -> Elenco nomi** . Nel foglio delle proprietà che si apre:
   a. Immettere un nome per l'elenco nomi.
   b. Aggiungere il nome dell'oggetto CRL (LDAP) (dal passo “2.a” a pagina 156) all'elenco.
   c. Fare clic su **OK**.
4. Fare clic con il tasto destro del mouse sul gestore code, selezionare **Proprietà**e selezionare la pagina **SSL** :
   a. Selezionare la casella di spunta **Verifica i certificati ricevuti da questo gestore code rispetto agli elenchi di revoca della certificazione** .
   b. Immettere il nome dell'elenco nomi (dal passo “3.a” a pagina 156) nel campo **Elenco nomi CRL** .

Accesso a CRL e ARL con un client IBM WebSphere MQ MQI

Sono disponibili tre opzioni per la specifica dei server LDAP che contengono i CRL per il controllo da parte di un client IBM WebSphere MQ MQI.

Si noti che in questa sezione, le informazioni sui CRL (Certificate Revocation Lists) si applicano anche agli ARL (Authority Revocation Lists).

I tre modi per specificare i server LDAP sono i seguenti:

- Utilizzo di una tabella di definizione di canale
- Utilizzo della struttura delle opzioni di configurazione SSL, MQSCO, su una chiamata MQCONNX
- Utilizzo di Active Directory (su sistemi Windows con supporto Active Directory)

Per ulteriori dettagli, fare riferimento alle informazioni correlate.

È possibile includere fino a 10 connessioni a server LDAP alternativi per garantire la continuità del servizio in caso di errore di uno o più server LDAP. Tenere presente che i server LDAP *devono* contenere informazioni identiche.

Non è possibile accedere ai CRL LDAP da un canale client WebSphere MQ in esecuzione su Linux (piattaforma zSeries).

Ubicazione di un responder OCSP e dei server LDAP che contengono i CRL
Su un sistema client IBM WebSphere MQ MQI, è possibile specificare l'ubicazione di un responder OCSP e dei server LDAP (Lightweight Directory Access Protocol) che contengono CRL (Certificate Revocation List).

È possibile specificare queste ubicazioni in tre modi, elencati qui in ordine decrescente di precedenza.

Quando un'applicazione client MQI WebSphere MQ emette una chiamata MQCONNX

È possibile specificare un responder OCSP o un server LDAP che contiene i CRL su una chiamata **MQCONNX** .

Su una chiamata **MQCONNX** , la struttura delle opzioni di connessione, MQCNO, può fare riferimento ad una struttura delle opzioni di configurazione SSL, MQSCO. A sua volta, la struttura MQSCO può fare riferimento a una o più strutture di record delle informazioni di autenticazione, MQAIR. Ogni struttura MQAIR contiene tutte le informazioni richieste dal client WebSphere MQ MQI per accedere a un responder OCSP o a un server LDAP che contiene i CRL. Ad esempio, uno dei campi in una struttura MQAIR è l'URL a cui è possibile contattare un responder. Per ulteriori informazioni sulla struttura MQAIR, consultare MQAIR - Record di informazioni di autenticazione.

Utilizzo di una ccdt (client channel definition table) per accedere a un responder OCSP o a server LDAP

In modo che un client WebSphere MQ MQI possa accedere a un responder OCSP o a server LDAP che contengono CRL, includere gli attributi di uno o più oggetti delle informazioni di autenticazione in una tabella di definizione del canale client.

Su un gestore code del server, è possibile definire uno o più oggetti delle informazioni di autenticazione. Gli attributi di un oggetto di autenticazione contengono tutte le informazioni richieste per accedere a un responder OCSP (sulle piattaforme in cui OCSP è supportato) o a un server LDAP che contiene i CRL. Uno degli attributi specifica l'URL del responder OCSP, un altro specifica l'indirizzo host o l'indirizzo IP di un sistema su cui viene eseguito un server LDAP.

Un oggetto delle informazioni di autenticazione con AUTHTYPE (OCSP) non si applica per l'utilizzo sui gestori code IBM i o z/OS , ma può essere specificato su tali piattaforme per essere copiato nella CCDT (Client Channel Definition Table) per l'utilizzo da parte del client.

Per consentire a un client WebSphere MQ MQI di accedere a un responder OCSP o a server LDAP che contengono CRL, gli attributi di uno o più oggetti delle informazioni di autenticazione possono essere inclusi in una tabella di definizione del canale client. È possibile includere tali attributi in uno dei seguenti modi:

Sulle piatteforme server AIX, HP-UX, Linux, Solaris e Windows

È possibile definire un elenco nomi che contiene i nomi di uno o più oggetti delle informazioni di autenticazione. È quindi possibile impostare l'attributo del gestore code, **SSLCRLNameList**, sul nome di questo elenco nomi.

Se si utilizzano i CRL, è possibile configurare più di un server LDAP per fornire una maggiore disponibilità. L'intenzione è che ciascun server LDAP conservi gli stessi CRL. Se un server LDAP non è disponibile quando è richiesto, un client WebSphere MQ MQI può tentare di accedervi.

Gli attributi degli oggetti delle informazioni di autenticazione identificati dall'elenco nomi vengono indicati collettivamente come *ubicazione di revoca del certificato*. Quando si imposta l'attributo del gestore code, **SSLCRLNameList**, sul nome dell'elenco nomi, l'ubicazione di revoca del certificato viene copiata nella tabella di definizione del canale client associata al gestore code. Se è possibile accedere alla CCDT da un sistema client come file condiviso o se la CCDT viene copiata su un sistema client, il client WebSphere MQ MQI su tale sistema può utilizzare l'ubicazione di revoca del certificato nella CCDT per accedere a un responder OCSP o ai server LDAP che contengono i CRL.

Se l'ubicazione di revoca del certificato del gestore code viene modificata successivamente, la modifica si riflette nel CCDT associato al gestore code. Se l'attributo del gestore code, **SSLCRLNameList**, è impostato su vuoto, l'ubicazione di revoca del certificato viene rimossa da CCDT. Queste modifiche non si riflettono in alcuna copia della tabella su un sistema client.

Se si richiede che l'ubicazione di revoca del certificato sul client e sulle estremità del server di un canale MQI sia diversa e il gestore code del server è quello utilizzato per creare l'ubicazione di revoca del certificato, è possibile effettuare le seguenti operazioni:

1. Sul gestore code del server, creare l'ubicazione di revoca del certificato da utilizzare sul sistema client.
2. Copiare la CCDT contenente l'ubicazione di revoca del certificato sul sistema client.
3. Sul gestore code del server, modificare l'ubicazione di revoca del certificato in ciò che è richiesto all'estremità del server del canale MQI.

Utilizzo di Active Directory in Windows

Su sistemi Windows , è possibile utilizzare il comando di controllo **setmqcrl** per pubblicare le informazioni CRL correnti in Active Directory.

Il comando **setmqcrl** non pubblica le informazioni OCSP.

Per informazioni su questo comando e la sua sintassi, consultare la sezione setmqcrl.

Accesso a CRL e ARL con classi IBM WebSphere MQ per Java e classi IBM WebSphere MQ per JMS

Le classi IBM WebSphere MQ per Java e le classi IBM WebSphere MQ per JMS accedono ai CRL in maniera diversa dalle altre piattaforme.

Per informazioni sulla gestione di CRL e ARL con classi IBM WebSphere MQ per Java, consultare Utilizzo degli elenchi di revoca dei certificati.

Per informazioni sull'utilizzo dei CRL e degli ARL con le classi IBM WebSphere MQ per JMS, consultare Proprietà oggetto SSLCERTSTORES.

Manipolazione degli oggetti delle informazioni di autenticazione

È possibile manipolare gli oggetti delle informazioni di autenticazione utilizzando i comandi MQSC o PCF o IBM WebSphere MQ Explorer.

I seguenti comandi MQSC agiscono sugli oggetti delle informazioni di autenticazione:

- DEFINE AUTINFO
- MODIFICA AUTHINFO

- DELETE AUTINFO
- VISUALIZZA AUTHINFO

Per una descrizione completa di questi comandi, consultare Comandi script (MQSC) .

I seguenti comandi PCF (Programmable Command Format) agiscono sugli oggetti delle informazioni di autenticazione:

- Creazione informazioni di autenticazione
- Copia informazioni di autenticazione
- Modifica informazioni di autenticazione
- Eliminazione informazioni di autenticazione
- Interrogazione informazioni di autenticazione
- Interrogazione nomi informazioni di autenticazione

Per una descrizione completa di questi comandi, vedere Definizioni dei formati dei comandi programmabili .

Sulle piattaforme in cui è disponibile, è anche possibile utilizzare WebSphere MQ Explorer.

Autorizzazione dell'accesso agli oggetti

Questa sezione contiene informazioni sull'utilizzo del gestore autorizzazioni oggetto e dei programmi di uscita canale per controllare l'accesso agli oggetti.

Su sistemi UNIX, Linux, and Windows . controllare l'accesso agli oggetti utilizzando OAM (object authority manager). Questa raccolta di argomenti contiene informazioni sull'utilizzo dell'interfaccia comandi per OAM. Contiene inoltre un elenco di controllo che è possibile utilizzare per determinare quali attività eseguire per applicare la sicurezza al proprio sistema e considerazioni per concedere agli utenti l'autorità di gestire IBM WebSphere MQ e gestire gli oggetti IBM WebSphere MQ . Se i meccanismi di sicurezza forniti non soddisfano le proprie esigenze, è possibile sviluppare i propri programmi di uscita canale.

Controllo dell'accesso agli oggetti utilizzando OAM su sistemi UNIX, Linux e Windows

OAM (object authority manager) fornisce un'interfaccia di comando per concedere e revocare l'autorizzazione agli oggetti WebSphere MQ .

È necessario essere autorizzati a utilizzare questi comandi, come descritto in “Autorizzazione per la gestione di IBM WebSphere MQ su sistemi UNIX, Linux, and Windows” a pagina 197. Gli ID utente che sono autorizzati a gestire WebSphere MQ hanno l'autorizzazione *super utente* per il gestore code, il che significa che non è necessario concedere loro ulteriori autorizzazioni per emettere richieste o comandi MQI.

Accesso a un oggetto IBM WebSphere MQ su un sistema UNIX, Linux, and Windows

Utilizzare il comando di controllo **setmqaut** o il comando PCF **MQCMD_SET_AUTH_REC** per fornire agli utenti e ai gruppi di utenti l'accesso agli oggetti IBM WebSphere MQ .

Per una definizione completa del comando di controllo **setmqaut** e della relativa sintassi, consultare setmqaute per una definizione completa del comando PCF **MQCMD_SET_AUTH_REC** e della relativa sintassi, consultare Imposta record di autorizzazioni.

Il gestore code deve essere in esecuzione per utilizzare questo comando. Una volta modificato l'accesso per un principal, le modifiche vengono riflesse immediatamente da OAM.

Per fornire agli utenti l'accesso a un oggetto, è necessario specificare:

- Il nome del gestore code che possiede gli oggetti utilizzati; se non si specifica il nome di un gestore code, viene utilizzato il gestore code predefinito.

- Il nome e il tipo dell'oggetto (per identificare l'oggetto in maniera univoca). Specificare il nome come *profilo*; questo è il nome esplicito dell'oggetto o un nome generico, inclusi i caratteri jolly. Per una descrizione dettagliata dei profili generici e l'utilizzo dei caratteri jolly al loro interno, consultare "Utilizzo di profili generici OAM su sistemi UNIX, Linux, and Windows" a pagina 161.
- Uno o più principal e nomi gruppo a cui si applica l'autorizzazione.

  Se un ID utente contiene spazi, racchiuderlo tra virgolette quando si utilizza questo comando. Sui sistemi Windows , è possibile qualificare un ID utente con un nome dominio. Se l'ID utente effettivo contiene un simbolo chiocciola (@), sostituirlo con @ @ per indicare che fa parte dell'ID utente e non il delimitatore tra l'ID utente e il nome dominio.
- Un elenco di autorizzazioni. Ogni elemento nell'elenco specifica un tipo di accesso che deve essere concesso a tale oggetto (o revocato ad esso). Ogni autorizzazione nell'elenco viene specificata come parola chiave, precedendo con un segno più (+) o con un segno meno (-). Utilizzare un segno più per aggiungere l'autorizzazione specificata e un segno meno per eliminare l'autorizzazione. Non devono essere presenti spazi tra il segno + o - e la parola chiave.

  È possibile specificare un numero qualsiasi di autorizzazioni in un singolo comando. Ad esempio, l'elenco di autorizzazioni per consentire a un utente o a un gruppo di inserire i messaggi in una coda e di sfogliarli, ma per revocare l'accesso per ottenere i messaggi è:

```
    +browse -get +put
```

Esempi di utilizzo del comando setmqaut

I seguenti esempi mostrano come utilizzare il comando `setmqaut` per concedere e revocare l'autorizzazione all'utilizzo di un oggetto:

```
setmqaut -m saturn.queue.manager -t queue -n RED.LOCAL.QUEUE
         -g groupa +browse -get +put
```

In questo esempio:

- `saturn.queue.manager` è il nome del gestore code
- `queue` è il tipo di oggetto
- `RED.LOCAL.QUEUE` è il nome dell'oggetto
- `groupa` è l'identificativo del gruppo con le autorizzazioni che devono essere modificate
- `+browse -get +put` è l'elenco di autorizzazioni per la coda specificata
  - `+browse` aggiunge l'autorizzazione per sfogliare i messaggi sulla coda (per emettere **MQGET** con l'opzione sfoglia)
  - `-get` rimuove l'autorizzazione a richiamare (**MQGET**) i messaggi dalla coda
  - `+put` aggiunge l'autorizzazione per inserire (**MQPUT**) messaggi nella coda

Il seguente comando revoca l'autorizzazione di inserimento sulla coda MyQueue dal principal fvuser e dai gruppi groupa e groupb. Su sistemi di UNIX and Linux , questo comando revoca anche l'autorizzazione di inserimento per tutti i principal nello stesso gruppo primario di fvuser.

```
setmqaut -m saturn.queue.manager -t queue -n MyQueue -p fvuser
         -g groupa -g groupb -put
```

Utilizzo del comando con un servizio di autorizzazione differente

Se si sta utilizzando il proprio servizio di autorizzazione invece di OAM, è possibile specificare il nome di questo servizio nel comando **setmqaut** per indirizzare il comando a questo servizio. È necessario specificare questo parametro se si dispone di più componenti installabili in esecuzione contemporaneamente; in caso contrario, l'aggiornamento viene effettuato al primo componente installabile per il servizio di autorizzazione. Per impostazione predefinita, questo è l'OAM fornito.

Utilizzo di profili generici OAM su sistemi UNIX, Linux, and Windows

I profili generici OAM consentono di impostare l'autorità di cui un utente dispone per molti oggetti contemporaneamente, piuttosto che dover immettere comandi **setmqaut** separati per ogni singolo oggetto quando viene creato.

L'uso di profili generici nel comando **setmqaut** consente di impostare un'autorizzazione generica per tutti gli oggetti che si adattano a tale profilo.

Questa raccolta di argomenti descrive in modo più dettagliato l'utilizzo di profili generici.

Utilizzo dei caratteri jolly nei profili OAM

Ciò che rende generico un profilo è l'uso di caratteri speciali (caratteri jolly) nel nome profilo. Ad esempio, il carattere jolly punto interrogativo (?) corrisponde a qualsiasi carattere singolo in un nome. Quindi, se si specifica `ABC.?EF`, l'autorizzazione che si concede a tale profilo si applica a tutti gli oggetti con i nomi `ABC.DEF` , `ABC.CEF`, `ABC.BEF`e così via.

I caratteri jolly disponibili sono:

?
Utilizzare il punto interrogativo (?) invece di qualsiasi carattere singolo. Ad esempio, `AB.?D` si riferisce agli oggetti `AB.CD` , `AB.ED`e `AB.FD`.

*****
Utilizzare l'asterisco (*) come:

- Un *qualificativo* in un nome profilo per corrispondere a un qualsiasi qualificativo in un nome oggetto. Un qualificatore è la parte di un nome di un oggetto delimitato da un punto. Ad esempio, in `ABC.DEF.GHI`, i qualificatori sono `ABC`, `DEF`e `GHI` .

  Ad esempio, `ABC.*.JKL` si applica agli oggetti `ABC.DEF.JKL`e `ABC.GHI.JKL`. (Si noti che **non** si applica a `ABC.JKL`; * utilizzato in questo contesto indica sempre un qualificatore).
- Un carattere all'interno di un qualificativo in un nome profilo che corrisponde a zero o più caratteri all'interno del qualificativo in un nome oggetto.

  Ad esempio, `ABC.DE*.JKL` si riferisce agli oggetti `ABC.DE.JKL`, `ABC.DEF.JKL`e `ABC.DEGH.JKL` .

******
Utilizzare il doppio asterisco (**) **una volta** in un nome profilo come:

- L'intero nome profilo deve corrispondere a tutti i nomi oggetto. Ad esempio, se si utilizza `-t prcs` per identificare i processi e si utilizza ** come nome profilo, si modificano le autorizzazioni per tutti i processi.
- Come qualificativo iniziale, centrale o finale in un nome profilo per corrispondere a zero o più qualificativi in un nome oggetto. Ad esempio `**.ABC` identifica tutti gli oggetti con il qualificatore finale ABC.

Nota: Quando si utilizzano i caratteri jolly sui sistemi UNIX and Linux , **è necessario** racchiudere il nome del profilo tra virgolette singole.

Priorità del profilo

Un punto importante da comprendere quando si utilizzano i profili generici è la priorità che i profili vengono dati quando si decide quali autorizzazioni applicare a un oggetto che si sta creando. Ad esempio, si supponga di aver immesso i seguenti comandi:

```
setmqaut -n AB.* -t q +put -p fred
setmqaut -n AB.C* -t q +get -p fred
```

Il primo fornisce l'autorità put a tutte le code per il principal fred con nomi che corrispondono al profilo AB. *; il secondo fornisce l'autorità get agli stessi tipi di coda che corrispondono al profilo AB.C*.

Si supponga di creare una coda denominata AB.CD. In base alle regole per la corrispondenza dei caratteri jolly, è possibile applicare setmqaut a tale coda. Quindi, ha messo o ottenuto l'autorità?

Per trovare la risposta, si applica la regola che, ogni volta che più profili possono essere applicati a un oggetto, **si applica solo il più specifico**. Il modo in cui si applica questa regola consiste nel confrontare i nomi dei profili da sinistra a destra. Laddove differiscono, un carattere non generico è più specifico di un carattere generico. Quindi, nell'esempio precedente, la coda AB.CD dispone dell'autorizzazione **get** (AB.C* è più specifico di AB. *).

Quando si confrontano caratteri generici, l'ordine di *specificità* è:

1. ?
2. *
3. **

Dump delle impostazioni del profilo

Per una definizione completa del comando di controllo **dmpmqaut** e della relativa sintassi, consultare dmpmqaute per una definizione completa del comando PCF **MQCMD_INQUIRE_AUTH_RECS** e della relativa sintassi, vedere Record di autorità di interrogazione .

I seguenti esempi mostrano l'utilizzo del comando di controllo **dmpmqaut** per eseguire il dump dei record di autorizzazione per profili generici:

1. Questo esempio esegue il dump di tutti record di autorizzazioni con un profilo che corrisponde alla coda a.b.c per il principal user1.

   ```
   dmpmqaut -m qm1 -n a.b.c -t q -p user1
   ```

   Il dump risultante è simile al seguente:

   ```
   profile:     a.b.*
   object type: queue
   entity:      user1
   type:        principal
   authority:   get, browse, put, inq
   ```

   Nota: Anche se gli utenti UNIX and Linux possono utilizzare l'opzione `-p` per il comando **dmpmqaut** , devono utilizzare invece `-g groupname` quando definiscono le autorizzazioni.
2. Questo esempio esegue il dump di tutti i record di autorizzazioni con un profilo che corrisponde alla coda a.b.c.

   ```
   dmpmqaut -m qmgr1 -n a.b.c -t q
   ```

   Il dump risultante è simile al seguente:

   ```
   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. Questo esempio esegue il dump di tutti i record di autorizzazioni per il profilo a.b. *, di tipo coda.

```
dmpmqaut -m qmgr1 -n a.b.* -t q
```

Il dump risultante è simile al seguente:

```
profile:     a.b.*
object type: queue
entity:      user1
type:        principal
authority:   get, browse, put, inq
```

4. Questo esempio esegue il dump di tutti i record di autorizzazioni per il gestore code qmX.

```
dmpmqaut -m qmX
```

Il dump risultante è simile al seguente:

```
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. Questo esempio esegue il dump di tutti i nomi profilo e i tipi di oggetto per il gestore code qmX.

```
dmpmqaut -m qmX -l
```

Il dump risultante è simile al seguente:

```
profile: q1, type: queue
profile: q*, type: queue
profile: name.*, type: namelist
profile: pr1, type: process
```

Nota: Per WebSphere MQ solo per Windows , tutti i principal visualizzati includono le informazioni sul dominio, ad esempio:

```
profile:     a.b.*
object type: queue
entity:      user1@domain1
type:        principal
authority:   get, browse, put, inq
```

Utilizzo dei caratteri jolly nei profili OAM

Utilizzare i caratteri jolly in un nome profilo OAM (object authority manager) per rendere tale profilo applicabile a più di un oggetto.

Ciò che rende generico un profilo è l'uso di caratteri speciali (caratteri jolly) nel nome profilo. Ad esempio, il carattere jolly punto interrogativo (?) corrisponde a qualsiasi carattere singolo in un nome. Quindi, se si

specifica ABC.?EF, l'autorizzazione che si concede a tale profilo si applica a tutti gli oggetti con i nomi ABC.DEF, ABC.CEF, ABC.BEFe così via.

I caratteri jolly disponibili sono:

?

Utilizzare il punto interrogativo (?) invece di qualsiasi carattere singolo. Ad esempio, AB.?D si riferisce agli oggetti AB.CD, AB.EDe AB.FD.

*****

Utilizzare l'asterisco (*) come:

- Un *qualificativo* in un nome profilo per corrispondere a un qualsiasi qualificativo in un nome oggetto. Un qualificatore è la parte di un nome di un oggetto delimitato da un punto. Ad esempio, in ABC.DEF.GHI, i qualificatori sono ABC, DEF e GHI.

  Ad esempio, ABC.*.JKL si applica agli oggetti ABC.DEF.JKLe ABC.GHI.JKL. (Si noti che **non** si applica a ABC.JKL; * utilizzato in questo contesto indica sempre un qualificatore).
- Un carattere all'interno di un qualificativo in un nome profilo che corrisponde a zero o più caratteri all'interno del qualificativo in un nome oggetto.

  Ad esempio, ABC.DE*.JKL si riferisce agli oggetti ABC.DE.JKL, ABC.DEF.JKLe ABC.DEGH.JKL.

******

Utilizzare il doppio asterisco (**) **una volta** in un nome profilo come:

- L'intero nome profilo deve corrispondere a tutti i nomi oggetto. Ad esempio, se si utilizza `-t prcs` per identificare i processi e si utilizza ** come nome profilo, si modificano le autorizzazioni per tutti i processi.
- Come qualificativo iniziale, centrale o finale in un nome profilo per corrispondere a zero o più qualificativi in un nome oggetto. Ad esempio **.ABC identifica tutti gli oggetti con il qualificatore finale ABC.

Nota: Quando si utilizzano i caratteri jolly sui sistemi UNIX and Linux , **è necessario** racchiudere il nome del profilo tra virgolette singole.

Priorità del profilo

Più di un profilo generico può essere applicato a un singolo oggetto. In questo caso si applica la regola più specifica.

Un punto importante da comprendere quando si utilizzano i profili generici è la priorità che i profili vengono dati quando si decide quali autorizzazioni applicare a un oggetto che si sta creando. Ad esempio, si supponga di aver immesso i seguenti comandi:

```
setmqaut -n AB.* -t q +put -p fred
setmqaut -n AB.C* -t q +get -p fred
```

Il primo fornisce l'autorità put a tutte le code per il principal fred con nomi che corrispondono al profilo AB. *; il secondo fornisce l'autorità get agli stessi tipi di coda che corrispondono al profilo AB.C*.

Si supponga di creare una coda denominata AB.CD. In base alle regole per la corrispondenza dei caratteri jolly, è possibile applicare setmqaut a tale coda. Quindi, ha messo o ottenuto l'autorità?

Per trovare la risposta, si applica la regola che, ogni volta che più profili possono essere applicati a un oggetto, **si applica solo il più specifico**. Il modo in cui si applica questa regola consiste nel confrontare i nomi dei profili da sinistra a destra. Laddove differiscono, un carattere non generico è più specifico di un carattere generico. Quindi, nell'esempio precedente, la coda AB.CD dispone dell'autorizzazione **get** (AB.C* è più specifico di AB. *).

Quando si confrontano caratteri generici, l'ordine di *specificità* è:

1. ?
2. *

3. **

Dump delle impostazioni del profilo

Utilizzare il comando di controllo **dmpmqaut** o il comando PCF **MQCMD_INQUIRE_AUTH_RECS** per eseguire il dump delle autorizzazioni correnti associate ad un profilo specificato.

Per una definizione completa del comando di controllo **dmpmqaut** e della relativa sintassi, consultare dmpmqaute per una definizione completa del comando PCF **MQCMD_INQUIRE_AUTH_RECS** e della relativa sintassi, vedere Record di autorità di interrogazione.

I seguenti esempi mostrano l'utilizzo del comando di controllo **dmpmqaut** per eseguire il dump dei record di autorizzazione per profili generici:

1. Questo esempio esegue il dump di tutti record di autorizzazioni con un profilo che corrisponde alla coda a.b.c per il principal user1.

   ```
   dmpmqaut -m qm1 -n a.b.c -t q -p user1
   ```

   Il dump risultante è simile al seguente esempio:

   ```
   profile:     a.b.*
   object type: queue
   entity:      user1
   type:        principal
   authority:   get, browse, put, inq
   ```

   Nota: Gli utenti UNIX and Linux non possono utilizzare l'opzione `-p` ; devono utilizzare invece `-g groupname` .
2. Questo esempio esegue il dump di tutti i record di autorizzazioni con un profilo che corrisponde alla coda a.b.c.

   ```
   dmpmqaut -m qmgr1 -n a.b.c -t q
   ```

   Il dump risultante è simile al seguente esempio:

   ```
   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. Questo esempio esegue il dump di tutti i record di autorizzazioni per il profilo a.b. *, di tipo coda.

   ```
   dmpmqaut -m qmgr1 -n a.b.* -t q
   ```

   Il dump risultante è simile al seguente esempio:

   ```
   profile:     a.b.*
   object type: queue
   entity:      user1
   type:        principal
   authority:   get, browse, put, inq
   ```
4. Questo esempio esegue il dump di tutti i record di autorizzazioni per il gestore code qmX.

   ```
   dmpmqaut -m qmX
   ```

Il dump risultante è simile al seguente esempio:

```
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. Questo esempio esegue il dump di tutti i nomi profilo e i tipi di oggetto per il gestore code qmX.

```
dmpmqaut -m qmX -l
```

Il dump risultante è simile al seguente esempio:

```
profile: q1, type: queue
profile: q*, type: queue
profile: name.*, type: namelist
profile: pr1, type: process
```

Nota: Per WebSphere MQ solo per Windows , tutti i principal visualizzati includono le informazioni sul dominio, ad esempio:

```
profile:     a.b.*
object type: queue
entity:      user1@domain1
type:        principal
authority:   get, browse, put, inq
```

Visualizzazione delle impostazioni di accesso

Utilizzare il comando di controllo **dspmqaut** o il comando PCF **MQCMD_INQUIRE_ENTITY_AUTH** per visualizzare le autorizzazioni di cui dispone un determinato principal o gruppo per un determinato oggetto.

Il gestore code deve essere in esecuzione per utilizzare questo comando. Quando si modifica l'accesso per un principal, le modifiche vengono riflesse immediatamente da OAM. L'autorizzazione può essere visualizzata solo per un gruppo o un principal alla volta. Per una definizione completa del comando di controllo **dmpmqaut** e la relativa sintassi, consultare dmpmqaute per una definizione completa del comando PCF **MQCMD_INQUIRE_ENTITY_AUTH** e la relativa sintassi, consultare Inquire Entity Authority.

Il seguente esempio mostra l'utilizzo del comando di controllo **dspmqaut** per visualizzare le autorizzazioni che il gruppo `GpAdmin` ha per una definizione di processo denominata `Annuities` che si trova sul gestore code `QueueMan1`.

```
dspmqaut -m QueueMan1 -t process -n Annuities -g GpAdmin
```

Modifica e revoca dell'accesso a un oggetto IBM WebSphere MQ

Per cambiare il livello di accesso di un utente o di un gruppo a un oggetto, utilizzare il comando **setmqaut** . Per revocare l'accesso di un particolare utente membro di un gruppo che dispone dell'autorizzazione, rimuovere l'utente dal gruppo.

Il processo di rimozione dell'utente da un gruppo è descritto in:

- “Creazione e gestione dei gruppi in Windows” a pagina 82
- “Creazione e gestione di gruppi su HP-UX” a pagina 84
- “Creazione e gestione di gruppi su AIX” a pagina 85
- “Creazione e gestione di gruppi su Solaris” a pagina 87
- “Creazione e gestione di gruppi su Linux” a pagina 87

.

All'ID utente che crea un oggetto IBM WebSphere MQ vengono concesse le autorizzazioni di controllo completo per tale oggetto. Se si rimuove questo ID utente dal gruppo mqm locale (o dal gruppo Amministratori su sistemi Windows), queste autorizzazioni non vengono revocate. Utilizzare il comando di controllo **setmqaut** o il comando PCF **MQCMD_DELETE_AUTH_REC** per revocare l'accesso a un oggetto per l'ID utente che lo ha creato, dopo averlo rimosso dal gruppo mqm o Administrators. Per una definizione completa del comando di controllo setmqaut e della sua sintassi, consultare setmqaute per una definizione completa del comando PCF **MQCMD_INQUIRE_ENTITY_AUTH** e della relativa sintassi, consultare Inquire Entity Authority.

Su Windows, eliminare le voci OAM corrispondenti ad uno specifico account utente Windows prima di eliminare il profilo utente. Impossibile rimuovere le voci OAM dopo aver rimosso l'account utente.

Come impedire i test di accesso di sicurezza sui sistemi UNIX, Linux, and Windows

Per disattivare tutte le verifiche di sicurezza, è possibile disattivare l'OAM. Ciò potrebbe essere adatto per un ambiente di test. Avendo disabilitato o rimosso l'OAM, non è possibile aggiungere un OAM a un gestore code esistente.

Se si decide che non si desidera eseguire controlli di sicurezza (ad esempio, in un ambiente di test), è possibile disabilitare l'OAM in uno dei seguenti due modi:

- Prima di creare un gestore code, impostare la variabile di ambiente del sistema operativo MQSNOAUT (in questo caso, non è possibile aggiungere un OAM in un secondo momento):

  Per ulteriori informazioni sulle implicazioni dell'impostazione della variabile MQSNOAUT, consultare Variabili di ambiente .
- Modificare il file di configurazione del gestore code per rimuovere il servizio. (Se si esegue questa operazione, non è possibile aggiungere un OAM in un secondo momento.)

Se si utilizza setmqaut o dspmqaut mentre l'OAM è disabilitato, notare i seguenti punti:

- OAM non convalida il principal o il gruppo specificato, il che significa che il comando può accettare valori non validi.
- OAM non esegue controlli di sicurezza e indica che tutti i principal e i gruppi sono autorizzati ad eseguire tutte le operazioni oggetto applicabili.

Avvertenza: Quando un OAM viene rimosso, non può essere reinserito in un gestore code esistente. Questo è perché l'OAM deve essere in posizione al momento della creazione dell'oggetto. Per utilizzare nuovamente l'OAM WebSphere MQ dopo che è stato rimosso, è necessario ricreare il gestore code.

Concetti correlati

Servizi installabili

Concessione dell'accesso richiesto alle risorse

Utilizzare questo argomento per determinare quali attività eseguire per applicare la sicurezza al sistema WebSphere MQ .

Informazioni su questa attività

Durante questa attività, si decide quali azioni sono necessarie per applicare il livello di sicurezza appropriato agli elementi dell'installazione di WebSphere MQ . Ogni singola attività a cui si fa riferimento fornisce istruzioni dettagliate per tutte le piattaforme.

Procedura

1. Devi limitare l'accesso al tuo gestore code a determinati utenti?
   a) No: non intraprendere ulteriori azioni.
   b) Sì: vai alla domanda successiva.
2. Questi utenti hanno bisogno di un accesso di gestione parziale su un sottoinsieme di risorse del gestore code?
   a) No: vai alla domanda successiva.
   b) Sì: consultare "Concessione di un accesso di gestione parziale su un sottoinsieme di risorse del gestore code" a pagina 168.
3. Questi utenti hanno bisogno di un accesso amministrativo completo su un sottoinsieme di risorse del gestore code?
   a) No: vai alla domanda successiva.
   b) Sì: consultare "Concessione dell'accesso di gestione completo su un sottoinsieme di risorse del gestore code" a pagina 173.
4. Questi utenti devono accedere in sola lettura a tutte le risorse del gestore code?
   a) No: vai alla domanda successiva.
   b) Sì: consultare "Concessione dell'accesso in sola lettura a tutte le risorse su un gestore code" a pagina 178.
5. Questi utenti hanno bisogno di un accesso amministrativo completo su tutte le risorse del gestore code?
   a) No: vai alla domanda successiva.
   b) Sì: consultare "Concessione dell'accesso amministrativo completo a tutte le risorse su un gestore code" a pagina 179.
6. Sono necessarie applicazioni utente per connettersi al gestore code?
   a) No: disabilitare la connettività, come descritto in "Rimozione della connettività al gestore code" a pagina 180
   b) Sì: consultare "Come consentire alle applicazioni utente di collegarsi al gestore code" a pagina 181.

Concessione di un accesso di gestione parziale su un sottoinsieme di risorse del gestore code

È necessario fornire a determinati utenti l'accesso di gestione parziale ad alcune risorse del gestore code, ma non a tutte. Utilizzare questa tabella per determinare le azioni da intraprendere.

Tabella 14. Concessione dell'accesso di gestione parziale a un sottoinsieme di risorse del gestore code

Gli utenti devono gestire gli oggetti di questo tipo	**Esegui questa azione**
Code	Concedere l'accesso di gestione parziale alle code richieste, come descritto in "Concessione di un accesso amministrativo limitato ad alcune code" a pagina 169

Tabella 14. Concessione dell'accesso di gestione parziale a un sottoinsieme di risorse del gestore code (Continua)	
Gli utenti devono gestire gli oggetti di questo tipo	**Esegui questa azione**
Argomenti	Concedere l'accesso di gestione parziale agli argomenti richiesti, come descritto in “Concessione di un accesso amministrativo limitato ad alcuni argomenti” a pagina 170
Canali	Concedere l'accesso amministrativo parziale ai canali richiesti, come descritto in “Concessione di un accesso amministrativo limitato ad alcuni canali” a pagina 170
Il gestore code	Concedere l'accesso di gestione parziale al gestore code, come descritto in “Concessione di un accesso di gestione limitato a un gestore code” a pagina 171
Processo padre	Concedere l'accesso amministrativo parziale ai processi richiesti, come descritto in “Concessione di un accesso amministrativo limitato ad alcuni processi” a pagina 172
Elenchi nomi	Concedere l'accesso amministrativo parziale agli elenchi nomi richiesti, come descritto in “Concessione di un accesso amministrativo limitato ad alcuni elenchi nomi” a pagina 172
Servizi	Concedere l'accesso amministrativo parziale ai servizi richiesti, come descritto in “Concessione di un accesso amministrativo limitato ad alcuni servizi” a pagina 173

Concessione di un accesso amministrativo limitato ad alcune code

Concedere l'accesso di gestione parziale ad alcune code su un gestore code, a ogni gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere un accesso amministrativo limitato ad alcune code per alcune azioni, utilizzare i comandi appropriati per il sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- I nomi delle variabili hanno i seguenti significati:

  QMgrName
  Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

  ObjectProfile
  Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

  GroupName
  Il nome del gruppo a cui concedere l'accesso.

ReqdAction
L'azione che si sta consentendo al gruppo di eseguire:

– Su sistemi UNIX, Linux e Windows , qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + dlt, + dsp. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.

Nota: La concessione di + crt per le code rende indirettamente l'utente o il gruppo un amministratore. Non utilizzare l'autorizzazione + crt per concedere un accesso di gestione limitato ad alcune code.

QTYPE

Per il comando DISPLAY, uno dei valori QUEUE, QLOCAL, QALIAS, QMODEL, QREMOTE o QCLUSTER.

Per altri valori di *ReqdAction*, uno dei valori QLOCAL, QALIAS, QMODEL o QREMOTE.

Concessione di un accesso amministrativo limitato ad alcuni argomenti

Concedere l'autorizzazione di gestione parziale ad alcuni argomenti su un gestore code a ciascun gruppo di utenti che ne hanno bisogno.

Informazioni su questa attività

Per concedere un accesso di gestione limitato ad alcuni argomenti per alcune azioni, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

ReqdAction
L'azione che si sta consentendo al gruppo di eseguire:

– Su sistemi UNIX, Linux e Windows , qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + crt, + dlt, + dsp. + ctrl. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.

Concessione di un accesso amministrativo limitato ad alcuni canali

Concedere l'accesso di gestione parziale ad alcuni canali su un gestore code a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere un accesso amministrativo limitato ad alcuni canali per alcune azioni, utilizzare i comandi appropriati per il sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

ReqdAction
L'azione che si sta consentendo al gruppo di eseguire:

– Su sistemi UNIX, Linux e Windows , qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + crt, + dlt, + dsp. + ctrl, + ctrlx. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.

Concessione di un accesso di gestione limitato a un gestore code

Concedere l'accesso di gestione parziale a un gestore code a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere un accesso di gestione limitato per eseguire alcune azioni sul gestore code, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

Risultati

Per determinare quali comandi MQSC l'utente può eseguire sul gestore code, immettere i seguenti comandi per ogni comando MQSC:

```
RDEFINE MQCMDS QMgrName.ReqdAction.QMGR UACC(NONE)
PERMIT QMgrName.ReqdAction.QMGR CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```

Per consentire all'utente di utilizzare il comando DISPLAY QMGR, emettere i seguenti comandi:

```
RDEFINE MQCMDS QMgrName.DISPLAY.QMGR UACC(NONE)
PERMIT QMgrName.DISPLAY.QMGR CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

ReqdAction
L'azione che si sta consentendo al gruppo di eseguire:

- Su sistemi UNIX, Linux e Windows , qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + crt, + dlt, + dsp. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.

Anche se + set è un'autorizzazione MQI e normalmente non è considerata amministrativa, la concessione di + set sul gestore code può indirettamente portare a un'autorizzazione di gestione completa. Non concedere + impostato a utenti e applicazioni comuni.

Concessione di un accesso amministrativo limitato ad alcuni processi

Concedere l'autorizzazione di gestione parziale ad alcuni processi su un gestore code a ciascun gruppo di utenti che ne hanno bisogno.

Informazioni su questa attività

Per concedere un accesso di gestione limitato ad alcuni processi per alcune azioni, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- I nomi delle variabili hanno i seguenti significati:

  QMgrName
  Il nome del gestore code.

  ObjectProfile
  Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

  GroupName
  Il nome del gruppo a cui concedere l'accesso.

  ReqdAction
  L'azione che si sta consentendo al gruppo di eseguire:

  – Su sistemi UNIX, Linux e Windows , qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + crt, + dlt, + dsp. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.

Concessione di un accesso amministrativo limitato ad alcuni elenchi nomi

Concedere l'accesso di gestione parziale ad alcuni elenchi nomi su un gestore code, a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere un accesso di gestione limitato ad alcuni elenchi nomi per alcune azioni, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- I nomi delle variabili hanno i seguenti significati:

  QMgrName
  Il nome del gestore code.

  ObjectProfile
  Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

  GroupName
  Il nome del gruppo a cui concedere l'accesso.

  ReqdAction
  L'azione che si sta consentendo al gruppo di eseguire:

– Su sistemi UNIX, Linux e Windows , qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + crt, + dlt, + ctrl, + ctrlx, + dsp. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.

Concessione di un accesso amministrativo limitato ad alcuni servizi

Concedere l'autorizzazione di gestione parziale ad alcuni servizi su un gestore code, a ciascun gruppo di utenti che ne hanno bisogno.

Informazioni su questa attività

Per concedere un accesso di gestione limitato ad alcuni servizi per alcune azioni, utilizzare i comandi appropriati per il proprio sistema operativo.

Nota: Gli oggetti di servizio non esistono su z/OS.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

Risultati

Questi comandi concedono l'accesso al servizio specificato. Per determinare quali comandi MQSC l'utente può eseguire sul servizio, immettere i seguenti comandi per ciascun comando MQSC:

```
RDEFINE MQCMDS QMgrName.ReqdAction.SERVICE UACC(NONE)
PERMIT QMgrName.ReqdAction.SERVICE CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```

Per consentire all'utente di utilizzare il comando DISPLAY SERVICE, immettere i seguenti comandi:

```
RDEFINE MQCMDS QMgrName.DISPLAY.SERVICE UACC(NONE)
PERMIT QMgrName.DISPLAY.SERVICE CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

ReqdAction
L'azione che si sta consentendo al gruppo di eseguire:

- Su sistemi UNIX, Linux e Windows , qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + crt, + dlt, + ctrl, + ctrlx, + dsp. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.

Concessione dell'accesso di gestione completo su un sottoinsieme di risorse del gestore code

È necessario fornire a determinati utenti l'accesso di gestione completo ad alcune, ma non tutte, le risorse del gestore code. Utilizzare queste tabelle per determinare le azioni da intraprendere.

Tabella 15. Concessione dell'accesso di gestione completo a un sottoinsieme di risorse del gestore code	
Gli utenti devono gestire gli oggetti di questo tipo	**Esegui questa azione**
Code	Concedere l'accesso amministrativo completo alle code richieste, come descritto in “Concessione di un accesso amministrativo completo ad alcune code” a pagina 174
Argomenti	Concedere l'accesso amministrativo completo agli argomenti richiesti, come descritto in “Concessione dell'accesso amministrativo completo ad alcuni argomenti” a pagina 175
Canali	Concedere l'accesso di gestione completo ai canali richiesti, come descritto in “Concessione di un accesso amministrativo completo ad alcuni canali” a pagina 175
Il gestore code	Concedere l'autorizzazione di gestione completa al gestore code, come descritto in “Concessione di un accesso di gestione completo a un gestore code” a pagina 176
Processo padre	Concedere l'accesso amministrativo completo ai processi richiesti, come descritto in “Concessione dell'accesso amministrativo completo ad alcuni processi” a pagina 176
Elenchi nomi	Concedere l'accesso amministrativo completo agli elenchi nomi richiesti, come descritto in “Concessione di accesso amministrativo completo ad alcuni elenchi nomi” a pagina 177
Servizi	Concedere l'accesso amministrativo completo ai servizi richiesti, come descritto in “Concessione di un accesso amministrativo completo ad alcuni servizi” a pagina 178

Concessione di un accesso amministrativo completo ad alcune code

Concedere l'accesso amministrativo completo ad alcune code su un gestore code, a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'accesso amministrativo completo ad alcune code, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQADMIN QMgrName.QUEUE.ObjectProfile UACC(NONE)
PERMIT QMgrName.QUEUE.ObjectProfile CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione dell'accesso amministrativo completo ad alcuni argomenti

Concedere l'autorizzazione di gestione completa ad alcuni argomenti su un gestore code, a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'accesso di gestione completo ad alcuni argomenti per alcune azioni, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQADMIN QMgrName.TOPIC.ObjectProfile UACC(NONE)
PERMIT QMgrName.TOPIC.ObjectProfile CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione di un accesso amministrativo completo ad alcuni canali

Concedere l'accesso di gestione completo ad alcuni canali su un gestore code a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'accesso amministrativo completo ad alcuni canali, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQADMIN QMgrName.CHANNEL.ObjectProfile UACC(NONE)
PERMIT QMgrName.CHANNEL.ObjectProfile CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione di un accesso di gestione completo a un gestore code

Concedere l'accesso di gestione completo a un gestore code a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'accesso di gestione completo al gestore code, utilizzare i comandi appropriati per il sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQADMIN QMgrName.QMGR UACC(NONE)
PERMIT QMgrName.QMGR CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione dell'accesso amministrativo completo ad alcuni processi

Concedere l'accesso di gestione completo ad alcuni processi su un gestore code, a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'accesso di gestione completo ad alcuni processi, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQADMIN QMgrName.CHANNEL.ObjectProfile UACC(NONE)
PERMIT QMgrName.PROCESS.ObjectProfile CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione di accesso amministrativo completo ad alcuni elenchi nomi

Concedere l'accesso amministrativo completo ad alcuni elenchi nomi su un gestore code, a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'accesso amministrativo completo ad alcuni elenchi nomi, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQADMIN QMgrName.NAMELIST.ObjectProfile UACC(NONE)
PERMIT QMgrName.NAMELIST.ObjectProfile CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione di un accesso amministrativo completo ad alcuni servizi

Concedere l'accesso di gestione completo ad alcuni servizi su un gestore code a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'accesso amministrativo completo ad alcuni servizi, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQADMIN QMgrName.SERVICE.ObjectProfile UACC(NONE)
PERMIT QMgrName.SERVICE.ObjectProfile CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione dell'accesso in sola lettura a tutte le risorse su un gestore code

Concedere l'accesso di sola lettura a tutte le risorse su un gestore code, a ciascun utente o gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Utilizzare la procedura guidata Aggiungi autorizzazioni basate sui ruoli o i comandi appropriati per il sistema operativo.

Procedura

- Utilizzando la procedura guidata:
  a) Nel riquadro WebSphere MQ Explorer Navigator , fare clic con il tasto destro del mouse sul gestore code e selezionare **Autorizzazioni oggetto** > **Aggiungi autorizzazioni basate sul ruolo**

  Si apre il wizard Aggiungi autorizzazioni basate sul ruolo.
- Per sistemi UNIX e Windows , immettere i comandi riportati di seguito:

```
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
```

Le autorizzazioni specifiche per SYSTEM.ADMIN.COMMAND.QUEUE e SYSTEM.MQEXPLORER.REPLY.MODEL MODELLO sono necessari solo se si desidera utilizzare MQ Explorer.

- Per IBM i, immettere i seguenti comandi:

```
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')
```

- Per z/OS, immettere i seguenti comandi:

```
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)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione dell'accesso amministrativo completo a tutte le risorse su un gestore code

Concedere l'accesso di gestione completo a tutte le risorse su un gestore code, a ciascun utente o gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Utilizzare la procedura guidata Aggiungi autorizzazioni basate sui ruoli o i comandi appropriati per il sistema operativo.

Procedura

- Utilizzando la procedura guidata:
  a) Nel riquadro WebSphere MQ Explorer Navigator , fare clic con il tasto destro del mouse sul gestore code e selezionare **Autorizzazioni oggetto** > **Aggiungi autorizzazioni basate sul ruolo**
  Si apre il wizard Aggiungi autorizzazioni basate sul ruolo.

- Per sistemi UNIX and Linux , immettere i seguenti comandi:

```
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
```

- Per i sistemi Windows, immettere gli stessi comandi dei sistemi UNIX and Linux , ma utilizzando il nome profilo @CLASS invece di @class.
- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQADMIN QMgrName.*.** UACC(NONE)
PERMIT QMgrName.*.** CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Rimozione della connettività al gestore code

Se non si desidera che le applicazioni utente si connettano al gestore code, rimuovere la relativa autorizzazione per connettersi ad esso.

Informazioni su questa attività

Revocare l'autorizzazione di tutti gli utenti a connettersi al gestore code utilizzando il comando appropriato per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
RDEFINE MQCONN QMgrName.CHIN UACC(NONE)
RDEFINE MQCONN QMgrName.CICS UACC(NONE)
RDEFINE MQCONN QMgrName.IMS UACC(NONE)
```

Non immettere alcun comando PERMIT.

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

GroupName
Il nome del gruppo a cui negare l'accesso.

Come consentire alle applicazioni utente di collegarsi al gestore code

Si desidera consentire all'applicazione utente di collegarsi al gestore code. Utilizzare le tabelle in questo argomento per determinare quali azioni intraprendere.

Innanzitutto, stabilire se le applicazioni client si connetteranno al gestore code.

Se nessuna delle applicazioni che si connetteranno al gestore code è un'applicazione client, disabilitare l'accesso remoto come descritto in “Disabilitazione dell'accesso remoto al gestore code” a pagina 188.

Se una o più applicazioni che si connetteranno al gestore code sono applicazioni client, proteggere la connettività remota come descritto in “Protezione della connettività remota al gestore code” a pagina 181.

In entrambi i casi, impostare la sicurezza della connessione come descritto in “Impostazione della sicurezza della connessione” a pagina 188

Se si desidera controllare l'accesso alle risorse per ogni utente che si connette al gestore code, fare riferimento alla seguente tabella. Se l'istruzione nella prima colonna è true, eseguire l'azione elencata nella seconda colonna.

Istruzione	Esegui questa azione
Si dispone di applicazioni che utilizzano code	Vedi “Controllo dell'accesso utente alle code” a pagina 189
Si dispone di applicazioni che utilizzano argomenti	Consultare “Controllo dell'accesso utente agli argomenti” a pagina 194.
Sono presenti applicazioni che interrogano l'oggetto gestore code	Consultare “Concessione dell'autorità per richiedere informazioni su un gestore code” a pagina 195.
Si dispone di applicazioni che utilizzano oggetti processo	Vedi “Concessione dell'autorità per accedere ai processi” a pagina 196
Si dispone di applicazioni che utilizzano elenchi nomi	Vedi “Concessione dell'autorità per accedere agli elenchi nomi” a pagina 196

Protezione della connettività remota al gestore code

È possibile proteggere la connettività remota al gestore code utilizzando SSL o TLS, un'uscita di sicurezza, i record di autenticazione di canale o una combinazione di questi metodi.

Informazioni su questa attività

Connettere un client al gestore code utilizzando un canale di connessione client sulla stazione di lavoro client e un canale di connessione server sul server. Proteggere tali connessioni in uno dei seguenti modi.

Procedura

1. Utilizzo di SSL o TLS con record di autenticazione di canale:
   a) Impedire a qualsiasi DN (Distinguished Name) di aprire un canale, utilizzando un record di autenticazione di canale SSLPEERMAP per associare tutti i DN a USERSRC (NOACCESS).

b) Consenti a specifici DN o serie di DN di aprire un canale utilizzando un record di autenticazione di canale SSLPEERMAP per associarli a USERSRC (CHANNEL).

2. Utilizzo di SSL o TLS con un'uscita di sicurezza:
   a) Impostare MCAUSER sul canale di connessione server su un identificativo utente senza privilegi.
   b) Scrivere un'uscita di protezione per assegnare un valore MCAUSER in base al valore del DN SSL che riceve nei campi SSLPeerNamePtr e SSLPeerNameLength passati all'uscita nella struttura MQCD.
3. Utilizzo di SSL o TLS con valori di definizione di canale fissi:
   a) Impostare SSLPEER sul canale di connessione server su un valore specifico o su un intervallo ristretto di valori.
   b) Impostare MCAUSER sul canale di connessione server sull'ID utente con cui deve essere eseguito il canale.
4. Utilizzo dei record di autenticazione di canale su canali che non utilizzano SSL o TLS:
   a) Impedire a qualsiasi indirizzo IP di aprire i canali, utilizzando un record di autenticazione di canale di associazione degli indirizzi con ADDRESS (*) e USERSRC (NOACCESS).
   b) Consentire agli indirizzi IP specifici di aprire i canali, utilizzando i record di autenticazione di canale di associazione indirizzi per tali indirizzi con USERSRC (CHANNEL).
5. Utilizzo di un'uscita di sicurezza:
   a) Scrivere un'uscita di sicurezza per autorizzare le connessioni in base a qualsiasi proprietà scelta, ad esempio, l'indirizzo IP di origine.
6. È anche possibile utilizzare i record di autenticazione di canale con un'uscita di sicurezza o utilizzare tutti e tre i metodi, se le circostanze particolari lo richiedono.

Blocco di specifici indirizzi IP
È possibile impedire a uno specifico canale di accettare una connessione in entrata da un indirizzo IP o impedire all'intero gestore code di consentire l'accesso da un indirizzo IP, utilizzando un record di autenticazione di canale.

Prima di iniziare

Abilitare i record di autenticazione di canale immettendo il seguente comando:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Informazioni su questa attività

Per impedire ai canali specifici di accettare una connessione in entrata e garantire che le connessioni vengano accettate solo quando si utilizza il nome canale corretto, è possibile utilizzare un tipo di regola per bloccare gli indirizzi IP. Per impedire a un indirizzo IP di accedere all'intero gestore code, normalmente si utilizza un firewall per bloccarlo in modo permanente. Tuttavia, è possibile utilizzare un altro tipo di regola per bloccare temporaneamente alcuni indirizzi, ad esempio mentre si è in attesa dell'aggiornamento del firewall.

Procedura

- Per bloccare gli indirizzi IP dall'utilizzo di un determinato canale, impostare un record di autenticazione di canale utilizzando il comando MQSC **SET CHLAUTH**o il comando PCF **Set Channel Authentication Record**.

```
SET CHLAUTH(generic-channel-name) TYPE(ADDRESSMAP) ADDRESS(generic-ip-address)
USERSRC(NOACCESS)
```

Esistono tre parti del comando:

SET CHLAUTH (*nome - canale generico*)
Utilizzare questa parte del comando per controllare se si desidera bloccare una connessione per l'intero gestore code, canale singolo o intervallo di canali. Ciò che si inserisce qui determina quali aree sono coperte.

Ad esempio:

- SET CHLAUTH('*') - blocca ogni canale su un gestore code, ossia l'intero gestore code
- SET CHLAUTH ('SYSTEM. *') - blocca ogni canale che inizia con SYSTEM.
- SET CHLAUTH ('SYSTEM.DEF.SVRCONN') - blocca il canale SYSTEM.DEF.SVRCONN

Tipo di regola CHLAUTH
Utilizzare questa parte del comando per specificare il tipo di comando e determinare se si desidera fornire un singolo indirizzo o un elenco di indirizzi.

Ad esempio:

- TYPE(ADDRESSMAP) - Utilizzare ADDRESSMAP se si desidera fornire un indirizzo singolo o un indirizzo jolly. Ad esempio, ADDRESS('192.168.*') blocca tutte le connessioni provenienti da un indirizzo IP che inizia in 192.168.

  Per ulteriori informazioni sul filtro degli indirizzi IP con modelli, consultare Indirizzi IP generici.
- TYPE(BLOCKADDR) - Utilizzare BLOCKADDR se si desidera fornire un elenco di indirizzi da bloccare.

Ulteriori parametri
Questi parametri dipendono dal tipo di regola utilizzato nella seconda parte del comando:

- Per TYPE(ADDRESSMAP) si utilizza ADDRESS
- Per TYPE(BLOCKADDR) si utilizza ADDRLIST

Riferimenti correlati
SET CHLAUTH

Blocco temporaneo di specifici indirizzi IP se il gestore code non è in esecuzione
È possibile che si desideri bloccare determinati indirizzi IP, o intervalli di indirizzi, quando il gestore code non è in esecuzione e quindi non è possibile emettere comandi MQSC. È possibile bloccare temporaneamente gli indirizzi IP su base eccezionale modificando il file `blockaddr.ini` .

Informazioni su questa attività

Il file `blockaddr.ini` contiene una copia delle definizioni BLOCKADDR utilizzate dal gestore code. Questo file viene letto dal listener se il listener viene avviato prima del gestore code. In tali circostanze, il listener utilizza tutti i valori che sono stati aggiunti manualmente al file `blockaddr.ini` .

Tuttavia, tenere presente che quando il gestore code viene avviato, scrive la serie di definizioni BLOCKADDR nel file `blockaddr.ini` , sovrascrivendo qualsiasi modifica manuale che potrebbe essere stata effettuata. Allo stesso modo, ogni volta che si aggiunge o si elimina una definizione BLOCKADDR utilizzando il comando **SET CHLAUTH** , il file `blockaddr.ini` viene aggiornato. È quindi possibile apportare modifiche permanenti alle definizioni BLOCKADDR solo utilizzando il comando **SET CHLAUTH** quando il gestore code è in esecuzione.

Procedura

1. Aprire il file `blockaddr.ini` in un editor di testo.
   Il file si trova nella directory dei dati del gestore code.
2. Aggiungere gli indirizzi IP come semplici coppie parola chiave - valore, dove la parola chiave è `Addr`.
   Per informazioni sul filtro degli indirizzi IP con modelli, consultare Indirizzi IP generici.
   Ad esempio:

```
Addr = 192.0.2.0
Addr = 192.0.*
Addr = 192.0.2.1-8
```

Attività correlate

“Blocco di specifici indirizzi IP” a pagina 182
È possibile impedire a uno specifico canale di accettare una connessione in entrata da un indirizzo IP o impedire all'intero gestore code di consentire l'accesso da un indirizzo IP, utilizzando un record di autenticazione di canale.

Riferimenti correlati

SET CHLAUTH

Blocco di ID utente specifici
È possibile impedire a utenti specifici di utilizzare un canale specificando ID utente che, se asseriti, causano l'arresto del canale. Eseguire questa operazione impostando un record di autenticazione di canale.

Prima di iniziare

Assicurarsi che i record di autenticazione di canali siano abilitati come segue:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procedura

Impostare un record di autenticazione di canale utilizzando il comando MQSC **SET CHLAUTH**o il comando PCF **Set Channel Authentication Record**. Ad esempio, è possibile emettere il comando MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE(BLOCKUSER) USERLIST(userID1, userID2)
```

generic - channel - name è il nome di un canale a cui si desidera controllare l'accesso oppure un pattern che include il simbolo asterisco (*) come carattere jolly che corrisponde al nome del canale.
L'elenco di utenti fornito su un TYPE(BLOCKUSER) si applica solo ai canali SVRCONN e non ai canali del gestore code.
userID1 e *userID2* sono l'ID di un utente a cui deve essere impedito l'utilizzo del canale. È anche possibile specificare il valore speciale *MQADMIN per fare riferimento agli utenti amministrativi privilegiati. Per ulteriori informazioni sugli utenti con privilegi, consultare “Utenti privilegiati” a pagina 147. Per ulteriori informazioni su *MQADMIN, consultare SET CHLAUTH.

Riferimenti correlati

SET CHLAUTH

Associazione di un gestore code remoto a un ID utente MCAUSER
È possibile utilizzare un record di autenticazione di canale per impostare l'attributo MCAUSER di un canale, in base al gestore code da cui si connette il canale.

Prima di iniziare

Assicurarsi che i record di autenticazione di canali siano abilitati come segue:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Informazioni su questa attività

Facoltativamente, è possibile limitare gli indirizzi IP a cui si applica la regola.

Notare che questa tecnica non si applica ai canali di connessione server. Se si specifica il nome di un canale di connessione server nei comandi mostrati di seguito, non ha alcun effetto.

Procedura

- Impostare un record di autenticazione di canale utilizzando il comando MQSC **SET CHLAUTH**o il comando PCF **Set Channel Authentication Record**. Ad esempio, è possibile emettere il comando MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE (QMGRMAP) QMNAME(generic-partner-qmgr-name
) USERSRC(MAP) MCAUSER(user)
```

  generic - channel - name è il nome di un canale a cui si desidera controllare l'accesso oppure un pattern che include il simbolo asterisco (*) come carattere jolly che corrisponde al nome del canale.
  generic - partner - qmgr - name è il nome del gestore code o un modello che include il simbolo asterisco (*) come carattere jolly che corrisponde al nome del gestore code.
  user è l'ID utente da utilizzare per tutte le connessioni dal gestore code specificato.

- Per limitare questo comando ad alcuni indirizzi IP, includere il parametro **ADDRESS** , nel modo seguente:

```
SET CHLAUTH('generic-channel-name') TYPE (QMGRMAP) QMNAME(generic-partner-qmgr-name
) USERSRC(MAP) MCAUSER(user) ADDRESS(
generic-ip-address)
```

  generic - channel - name è il nome di un canale a cui si desidera controllare l'accesso oppure un pattern che include il simbolo asterisco (*) come carattere jolly che corrisponde al nome del canale.
  generic - ip - address è un indirizzo singolo o un modello che include il simbolo asterisco (*) come carattere jolly o il trattino (-) per indicare un intervallo, che corrisponde all'indirizzo. Per ulteriori informazioni sugli indirizzi IP generici, consultare Indirizzi IP generici .

Riferimenti correlati
SET CHLAUTH

Associazione di un ID utente asserito dal client ad un ID utente MCAUSER
È possibile utilizzare un record di autenticazione di canale per modificare l'attributo MCAUSER di un canale di connessione server, in base all'ID utente originale ricevuto da un client.

Prima di iniziare

Assicurarsi che i record di autenticazione di canali siano abilitati come segue:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Informazioni su questa attività

Notare che questa tecnica si applica solo ai canali di connessione server. Non ha alcun effetto su altri tipi di canale.

Procedura

Impostare un record di autenticazione di canale utilizzando il comando MQSC **SET CHLAUTH**o il comando PCF **Set Channel Authentication Record** . Ad esempio, è possibile emettere il comando MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE (USERMAP) CLNTUSER(client-user-name) USERSRC(MAP)
MCAUSER(
user)
```

generic - channel - name è il nome di un canale a cui si desidera controllare l'accesso oppure un pattern che include il simbolo asterisco (*) come carattere jolly che corrisponde al nome del canale.
nome - utente - client è l'ID utente dichiarato dal client.
utente è l'ID utente da utilizzare al posto del nome utente client.

Riferimenti correlati
SET CHLAUTH

Associazione di un DN (Distinguished Name) SSL o TLS a un ID utente MCAUSER
È possibile utilizzare un record di autenticazione di canale per impostare l'attributo MCAUSER di un canale, in base al DN (Distinguished Name) ricevuto.

Prima di iniziare
Assicurarsi che i record di autenticazione di canali siano abilitati come segue:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procedura
Impostare un record di autenticazione di canale utilizzando il comando MQSC **SET CHLAUTH**o il comando PCF **Set Channel Authentication Record**. Ad esempio, è possibile emettere il comando MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE (SSLPEERMAP) SSLPEER(generic-ssl-peer-name
) USERSRC(MAP) MCAUSER(user)
```

generic - channel - name è il nome di un canale a cui si desidera controllare l'accesso oppure un pattern che include il simbolo asterisco (*) come carattere jolly che corrisponde al nome del canale.
generic - ssl - peer - name è una stringa che segue le regole standard di IBM WebSphere MQ per i valori SSLPEER. Consultare WebSphere MQ regole per i valori SSLPEER.
user è l'ID utente da utilizzare per tutte le connessioni che utilizzano il DN specificato.

Riferimenti correlati
SET CHLAUTH

Blocco dell'accesso da un gestore code remoto
È possibile utilizzare un record di autenticazione di canale per evitare che un gestore code remoto avvii canali.

Prima di iniziare
Assicurarsi che i record di autenticazione di canali siano abilitati come segue:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Informazioni su questa attività
Notare che questa tecnica non si applica ai canali di connessione server. Se si specifica il nome di un canale di connessione server nel comando mostrato di seguito, non ha alcun effetto.

Procedura
Impostare un record di autenticazione di canale utilizzando il comando MQSC **SET CHLAUTH**o il comando PCF **Set Channel Authentication Record**. Ad esempio, è possibile emettere il comando MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE(QMGRMAP) QMNAME('generic-partner-qmgr-name')
USERSRC(NOACCESS)
```

generic - channel - name è il nome di un canale a cui si desidera controllare l'accesso oppure un pattern che include il simbolo asterisco (*) come carattere jolly che corrisponde al nome del canale.
generic - partner - qmgr - name è il nome del gestore code o un modello che include il simbolo asterisco (*) come carattere jolly che corrisponde al nome del gestore code.

Riferimenti correlati
SET CHLAUTH

Blocco dell'accesso per un ID utente dichiarato dal client
È possibile utilizzare un record di autenticazione di canale per evitare che un ID utente dichiarato dal client avvii i canali.

Prima di iniziare

Assicurarsi che i record di autenticazione di canali siano abilitati come segue:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Informazioni su questa attività

Notare che questa tecnica si applica solo ai canali di connessione server. Non ha alcun effetto su altri tipi di canale.

Procedura

Impostare un record di autenticazione di canale utilizzando il comando MQSC **SET CHLAUTH**o il comando PCF **Set Channel Authentication Record**. Ad esempio, è possibile emettere il comando MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE(USERMAP) CLNTUSER('client-user-name') USERSRC(NOACCESS)
```

generic - channel - name è il nome di un canale a cui si desidera controllare l'accesso oppure un pattern che include il simbolo asterisco (*) come carattere jolly che corrisponde al nome del canale.
nome - utente - client è l'ID utente dichiarato dal client.

Riferimenti correlati

SET CHLAUTH

Blocco dell'accesso per un DN (Distinguished Name) SSL
È possibile utilizzare un record di autenticazione di canale per evitare che un DN (Distinguished Name) SSL avvii canali.

Prima di iniziare

Assicurarsi che i record di autenticazione di canali siano abilitati come segue:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procedura

Impostare un record di autenticazione di canale utilizzando il comando MQSC **SET CHLAUTH**o il comando PCF **Set Channel Authentication Record**. Ad esempio, è possibile emettere il comando MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE(SSLPEERMAP) SSLPEER('generic-ssl-peer-name')
USERSRC(NOACCESS)
```

generic - channel - name è il nome di un canale a cui si desidera controllare l'accesso oppure un pattern che include il simbolo asterisco (*) come carattere jolly che corrisponde al nome del canale.
generic - ssl - peer - name è una stringa che segue le regole standard di IBM WebSphere MQ per i valori SSLPEER. Consultare WebSphere MQ regole per i valori SSLPEER.

Riferimenti correlati

SET CHLAUTH

Associazione di un indirizzo IP a un ID utente MCAUSER
È possibile utilizzare un record di autenticazione di canale per impostare l'attributo MCAUSER di un canale, in base all'indirizzo IP da cui viene ricevuta la connessione.

Prima di iniziare

Assicurarsi che i record di autenticazione di canali siano abilitati come segue:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procedura

Impostare un record di autenticazione di canale utilizzando il comando MQSC **SET CHLAUTH**o il comando PCF **Set Channel Authentication Record**. Ad esempio, è possibile emettere il comando MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE(ADDRESSMAP) ADDRESS('generic-ip-address') USERSRC(MAP)
MCAUSER(user)
```

generic - channel - name è il nome di un canale a cui si desidera controllare l'accesso oppure un pattern che include il simbolo asterisco (*) come carattere jolly che corrisponde al nome del canale.
user è l'ID utente da utilizzare per tutte le connessioni che utilizzano il DN specificato.
generic - ip - address è l'indirizzo da cui viene effettuata la connessione o un modello che include l'asterisco (*) come carattere jolly o il trattino (-) per indicare un intervallo, che corrisponde all'indirizzo.

Riferimenti correlati

SET CHLAUTH

Disabilitazione dell'accesso remoto al gestore code

Se non si desidera che le applicazioni client si connettano al proprio gestore code, disabilitare l'accesso remoto ad esso.

Informazioni su questa attività

Impedire alle applicazioni client di connettersi al gestore code in uno dei seguenti modi:

Procedura

- Eliminare tutti i canali di connessione server utilizzando il comando MQSC **DELETE CHANNEL**.
- Impostare l'identificativo utente dell'agent del canale (MCAUSER) del canale su un ID utente senza diritti di accesso, utilizzando il comando MQSC **ALTER CHANNEL**.

Impostazione della sicurezza della connessione

Concedere l'autorità per connettersi al gestore code a ciascun utente o gruppo di utenti con necessità di business.

Informazioni su questa attività

Per impostare la sicurezza della connessione, utilizzare i comandi appropriati per il sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

  ```
  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)
```

Questi comandi forniscono l'autorità di connessione per batch, CICS, IMS e CHIN (channel initiator). Se non si utilizza un particolare tipo di connessione, omettere i comandi pertinenti.

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Controllo dell'accesso utente alle code

Si desidera controllare l'accesso dell'applicazione alle code. Utilizzare questo argomento per determinare quali azioni intraprendere.

Per ogni istruzione true nella prima colonna, eseguire l'azione indicata nella seconda colonna.

Istruzione	Azione
L'applicazione richiama i messaggi da una coda	Vedi “Concessione dell'autorità per richiamare i messaggi dalle code” a pagina 189
L'applicazione imposta il contesto	Vedi “Concessione dell'autorità per impostare il contesto” a pagina 190
L'applicazione passa il contesto	Vedi “Concessione dell'autorizzazione per passare il contesto” a pagina 191
L'applicazione inserisce i messaggi in una coda cluster	Vedi “Autorizzazione all'inserimento di messaggi nelle code del cluster remoto” a pagina 246
L'applicazione inserisce i messaggi su una coda locale	Vedi “Concessione dell'autorizzazione per inserire i messaggi in una coda locale” a pagina 191
L'applicazione inserisce i messaggi in una coda modello	Vedi “Concessione dell'autorizzazione per inserire i messaggi in una coda modello” a pagina 192
L'applicazione inserisce i messaggi su una coda remota	Vedi “Concessione dell'autorità per inserire i messaggi in una coda cluster remota” a pagina 193

Concessione dell'autorità per richiamare i messaggi dalle code

Concedere l'autorizzazione a richiamare i messaggi da una coda o da una serie di code, a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'autorizzazione a richiamare i messaggi da alcune code, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione dell'autorità per impostare il contesto
Concedere l'autorità per impostare il contesto su un messaggio che si sta inserendo, a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'autorità di impostare il contesto su alcune code, utilizzare i comandi appropriati per il sistema operativo.

Procedura

- Per sistemi UNIX, Linux e Windows , immettere uno dei comandi riportati di seguito:
  - Per impostare solo il contesto di identità:

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

  - Per impostare tutto il contesto:

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

- Per IBM i, immettere uno dei seguenti comandi:
  - Per impostare solo il contesto di identità:

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

  - Per impostare tutto il contesto:

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

- Per z/OS, immettere una delle seguenti serie di comandi:
  - Per impostare solo il contesto di identità:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
```

  - Per impostare tutto il contesto:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(CONTROL)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione dell'autorizzazione per passare il contesto
Concedere l'autorizzazione a trasmettere il contesto da un messaggio richiamato a uno che si sta inserendo, a ogni gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'autorizzazione a passare il contesto su alcune code, utilizzare i comandi appropriati per il sistema operativo.

Procedura

- Per sistemi UNIX, Linux e Windows , immettere uno dei comandi riportati di seguito:
  - Per passare solo il contesto di identità:

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

  - Per passare tutti i contesti:

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

- Per IBM i, immettere uno dei seguenti comandi:
  - Per passare solo il contesto di identità:

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

  - Per passare tutti i contesti:

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

- Per z/OS, immettere i seguenti comandi per passare il contesto di identità o tutto il contesto:

  ```
  RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
  PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
  ```

  I nomi delle variabili hanno i seguenti significati:

  QMgrName
  Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

  ObjectProfile
  Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

  GroupName
  Il nome del gruppo a cui concedere l'accesso.

Concessione dell'autorizzazione per inserire i messaggi in una coda locale
Concedere l'autorizzazione a inserire i messaggi in una coda locale o in una serie di code, a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'autorità di inserire i messaggi in alcune code locali, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
```

  I nomi delle variabili hanno i seguenti significati:

  QMgrName
  Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

  ObjectProfile
  Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

  GroupName
  Il nome del gruppo a cui concedere l'accesso.

Concessione dell'autorizzazione per inserire i messaggi in una coda modello
Concedere l'autorità di inserire i messaggi in una coda modello o in una serie di code modello a ciascun gruppo di utenti che ne hanno bisogno.

Informazioni su questa attività

Le code modello vengono utilizzate per creare code dinamiche. Pertanto, è necessario concedere l'autorità sia alle code modello che a quelle dinamiche. Per concedere queste autorizzazioni, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per sistemi UNIX, Linux e Windows , immettere i seguenti comandi:

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

- Per IBM i, immettere i seguenti comandi:

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

- Per z/OS, immettere i seguenti comandi:

```
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)
```

  I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

Nome ModelQueue
Il nome della coda modello su cui si basano le code dinamiche.

ObjectProfile
Il nome della coda dinamica o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione dell'autorità per inserire i messaggi in una coda cluster remota
Concedere l'autorità per inserire i messaggi in una coda cluster remota o in una serie di code, a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per inserire un messaggio in una coda cluster remota, è possibile inserirlo in una definizione locale di una coda remota o in una coda remota completa. Se si utilizza una definizione locale di una coda remota, è necessaria l'autorizzazione per inserire l'oggetto locale: consultare “Concessione dell'autorizzazione per inserire i messaggi in una coda locale” a pagina 191. Se si sta utilizzando una coda remota completa, è necessaria l'autorizzazione per inserire la coda remota. Concedere questa autorizzazione utilizzando i comandi appropriati per il proprio sistema operativo.

Il comportamento predefinito è quello di eseguire il controllo accessi su `SYSTEM.CLUSTER.TRANSMIT.QUEUE`. Notare che questo comportamento si applica, anche se si utilizzano più code di trasmissione.

Il comportamento specifico descritto in questo argomento si applica solo quando l'attributo **`ClusterQueueAccessControl`** nel file `qm.ini` è configurato come *RQMName*, come descritto nella sezione Stanza di sicurezza , e il gestore code è stato riavviato.

Su sistemi UNIX, Linuxe Windows , è possibile utilizzare anche il comando SET AUTHREC.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

```
setmqaut -m QMgrName -t rqmname -n
ObjectProfile -g GroupName +put
```

  Notare che è possibile utilizzare l'oggetto *rqmname* solo per le code cluster remote.

- Per IBM i, immettere il seguente comando:

```
GRTMQMAUT OBJTYPE(*RMTMQMNAME) OBJ('
ObjectProfile') USER(GroupName) AUT(*PUT) MQMNAME('
QMgrName')
```

  Si noti che è possibile utilizzare l'oggetto RMTMQMNAME solo per le code cluster remote.

- Per z/OS, immettere i seguenti comandi:

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

  Tenere presente che è possibile utilizzare il nome del gestore code remoto (o del gruppo di condivisione code) solo per le code del cluster remoto.

  I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome del gestore code remoto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Controllo dell'accesso utente agli argomenti

È necessario controllare l'accesso delle applicazioni agli argomenti. Utilizzare questo argomento per determinare quali azioni intraprendere.

Per ogni istruzione true nella prima colonna, eseguire l'azione indicata nella seconda colonna.

Tabella 16. Controllo dell'accesso utente agli argomenti

Istruzione	**Azione**
L'applicazione pubblica i messaggi in un argomento	Vedi “Concessione dell'autorizzazione a pubblicare messaggi in un argomento” a pagina 194
L'applicazione si sottoscrive a un argomento	Vedi “Concessione dell'autorizzazione alla sottoscrizione di argomenti” a pagina 194

Concessione dell'autorizzazione a pubblicare messaggi in un argomento

Concedere l'autorizzazione a pubblicare messaggi in un argomento o in una serie di argomenti, a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'autorizzazione a pubblicare i messaggi in alcuni argomenti, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQTOPIC QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQTOPIC) ID(GroupName) ACCESS(UPDATE)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione dell'autorizzazione alla sottoscrizione di argomenti

Concedere l'autorità di sottoscrivere un argomento o una serie di argomenti a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'autorità di sottoscrivere alcuni argomenti, utilizzare i comandi appropriati per il sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQTOPIC QMgrName.SUBSCRIBE.ObjectProfile UACC(NONE)
PERMIT QMgrName.SUBSCRIBE.ObjectProfile CLASS(MQTOPIC) ID(GroupName) ACCESS(UPDATE)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione dell'autorità per richiedere informazioni su un gestore code

Concedere l'autorità di interrogare un gestore code a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'autorità di indagare su un gestore code, utilizzare i comandi appropriati per il proprio sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

```
setmqaut -m QMgrName -n ObjectProfile -t qmgr -g GroupName +inq
```

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQCMDS QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName ObjectProfile CLASS(MQCMDS) ID(GroupName ) ACCESS(READ)
```

Questi comandi consentono l'accesso al gestore code specificato. Per permettere all'utente di utilizzare il comando MQINQ, immettere i seguenti comandi:

```
RDEFINE MQCMDS QMgrName.MQINQ.QMGR UACC(NONE)
PERMIT QMgrName.MQINQ.QMGR CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione dell'autorità per accedere ai processi

Concedere l'autorità per accedere a un processo o a una serie di processi, a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'autorizzazione ad accedere ad alcuni processi, utilizzare i comandi appropriati per il sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQPROC QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQPROC) ID(GroupName) ACCESS(READ)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Concessione dell'autorità per accedere agli elenchi nomi

Concedere l'autorizzazione per accedere a un elenco nomi o a una serie di elenchi nomi a ciascun gruppo di utenti con un'esigenza aziendale.

Informazioni su questa attività

Per concedere l'autorizzazione ad accedere ad alcuni elenchi nomi, utilizzare i comandi appropriati per il sistema operativo.

Procedura

- Per i sistemi UNIX, Linux e Windows , immettere il seguente comando:

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

- Per IBM i, immettere il seguente comando:

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

- Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQNLIST
QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile
 CLASS(MQNLIST) ID(GroupName) ACCESS(READ)
```

I nomi delle variabili hanno i seguenti significati:

QMgrName
Il nome del gestore code. Su z/OS, questo valore può anche essere il nome di un gruppo di condivisione code.

ObjectProfile
Il nome dell'oggetto o del profilo generico per cui modificare le autorizzazioni.

GroupName
Il nome del gruppo a cui concedere l'accesso.

Autorizzazione per la gestione di IBM WebSphere MQ su sistemi UNIX, Linux, and Windows

Gli amministratori IBM WebSphere MQ possono utilizzare tutti i comandi IBM WebSphere MQ e concedere le autorizzazioni per altri utenti. Quando gli amministratori immettono comandi ai gestori code remoti, devono disporre dell'autorizzazione richiesta sul gestore code remoto. Ulteriori considerazioni si applicano ai sistemi Windows .

Gli amministratori IBM WebSphere MQ hanno l'autorità per utilizzare tutti i comandi WebSphere MQ (inclusi i comandi per concedere le autorizzazioni WebSphere MQ per altri utenti)

Per essere un amministratore IBM WebSphere MQ , è necessario essere un membro di un gruppo speciale denominato gruppo *mqm* (o un membro del gruppo Amministratori su sistemi Windows). Il gruppo mqm viene creato automaticamente quando WebSphere MQ è installato; aggiungere ulteriori utenti al gruppo per consentire loro di eseguire l'amministrazione. Tutti i membri di questo gruppo hanno accesso a tutte le risorse. Questo accesso può essere revocato solo rimuovendo un utente dal gruppo mqm e immettendo il comando REFRESH SECURITY. Gli amministratori possono utilizzare i comandi di controllo per gestire WebSphere MQ. Uno di questi comandi di controllo è **setmqaut**, che viene utilizzato per concedere autorizzazioni ad altri utenti per consentire loro di accedere o controllare le risorse WebSphere MQ . I comandi PCF per la gestione dei record di autorizzazione sono disponibili per i non amministratori a cui sono state concesse le autorizzazioni dsp e chg sul gestore code. Per ulteriori informazioni sulla gestione delle autorizzazioni utilizzando i comandi PCF, consultare Programmable Command Formats.

Gli amministratori possono utilizzare il comando di controllo **runmqsc** per immettere i comandi IBM WebSphere MQ Script (MQSC). Quando **runmqsc** viene utilizzato in modalità indiretta per inviare comandi MQSC a un gestore code remoto, ciascun comando MQSC viene incapsulato all'interno di un comando PCF Escape. Gli amministratori devono disporre delle autorizzazioni richieste per i comandi MQSC che devono essere elaborati dal gestore code remoto. WebSphere MQ Explorer emette comandi PCF per eseguire attività di amministrazione. Gli amministratori non richiedono ulteriori autorizzazioni per utilizzare WebSphere MQ Explorer per gestire un gestore code sul sistema locale. Quando Esplora risorse di IBM WebSphere MQ viene utilizzato per gestire un gestore code su un altro sistema, gli amministratori devono disporre delle autorizzazioni richieste per l'elaborazione dei comandi PCF da parte del gestore code remoto.

Per ulteriori informazioni sui controlli di autorizzazione quando vengono elaborati i comandi PCF e MQSC, consultare i seguenti argomenti:

- Per i comandi PCF che operano su gestori code, code, processi, elenchi nomi e oggetti delle informazioni di autenticazione, consultare Authority to work with WebSphere MQ objects. Fare riferimento a questa sezione per i comandi MQSC equivalenti incapsulati nei comandi Escape PCF.

- Per i comandi PCF che operano su canali, iniziatori di canali, listener e cluster, consultare Sicurezza canale.
- Per i comandi PCF che operano sui record di autorizzazione, consultare Controllo autorizzazione per comandi PCF

Inoltre, su sistemi Windows , l'account SYSTEM ha accesso completo alle risorse WebSphere MQ .

Su piattaforme UNIX and Linux , viene creato anche un ID utente speciale di mqm, che può essere utilizzato solo dal prodotto. Non deve essere mai disponibile per utenti non privilegiati. Tutti gli oggetti WebSphere MQ sono di proprietà dell'ID utente mqm.

Sui sistemi Windows , i membri del gruppo Amministratori possono anche gestire qualsiasi gestore code, così come l'account SYSTEM. È anche possibile creare un gruppo mqm di dominio sul controller di dominio che contenga tutti gli ID utente privilegiati attivi nel dominio e aggiungerlo al gruppo mqm locale. Alcuni comandi, ad esempio **crtmqm**, manipolano le autorizzazioni sugli oggetti IBM WebSphere MQ e quindi necessitano dell'autorizzazione per gestire tali oggetti (come descritto nelle seguenti sezioni). I membri del gruppo mqm hanno l'autorizzazione a gestire tutti gli oggetti, ma su sistemi Windows potrebbero verificarsi casi in cui l'autorizzazione viene negata se si dispone di un utente locale e di un utente autenticato dal dominio con lo stesso nome. Ciò è descritto in “Principal e gruppi” a pagina 201.

Le versioni di Windows con una funzione UAC (User Account Control) limitano le azioni che gli utenti possono eseguire su determinate funzioni del sistema operativo, anche se sono membri del gruppo Administrators. Se l'ID utente si trova nel gruppo Administrators ma non nel gruppo mqm, è necessario utilizzare un prompt dei comandi elevato per immettere i comandi admin di WebSphere MQ come **crtmqm**, altrimenti viene generato l'errore "AMQ7077: Non si è autorizzati ad eseguire l'operazione richiesta". Per aprire un prompt dei comandi elevato, fare clic con il tasto destro del mouse sulla voce del menu di avvio o sull'icona per il prompt dei comandi e selezionare "Esegui come amministratore".

Non è necessario essere membri del gruppo mqm per effettuare le seguenti operazioni:

- Immettere i comandi da un programma applicativo che emette comandi PCF o i comandi MQSC all'interno di un comando PCF Escape, a meno che i comandi non manipolino gli iniziatori di canale. (Questi comandi sono descritti in “Protezione delle definizioni dell'iniziatore di canali” a pagina 71).
- Emettere chiamate MQI da un programma applicativo (a meno che non si desideri utilizzare i collegamenti rapidi sulla chiamata MQCONNX).
- Utilizzare il comando `crtmqcvx` per creare un frammento di codice che esegue la conversione dei dati sulle strutture dei tipi di dati.
- Utilizzare il comando `dspmq` per visualizzare i gestori code.
- Utilizzare il comando `dspmqtrc` per visualizzare l'output di traccia formattato WebSphere MQ .

Una limitazione di 12 caratteri si applica sia agli ID gruppo che agli ID utente.

Le piattaforme UNIX and Linux generalmente limitano la lunghezza di un ID utente a 12 caratteri. AIX Versione 5.3 ha aumentato questo limite, ma WebSphere MQ continua a osservare una limitazione di 12 caratteri su tutte le piattaforme UNIX and Linux . Se si utilizza un ID utente di lunghezza superiore a 12 caratteri, WebSphere MQ lo sostituisce con il valore `UNKNOWN`. Non definire un ID utente con valore `UNKNOWN`.

Gestione del gruppo mqm

Agli utenti del gruppo mqm vengono concessi privilegi di gestione completi su WebSphere MQ. Per questo motivo, non è necessario registrare le applicazioni e gli utenti ordinari nel gruppo mqm. Il gruppo mqm deve contenere solo gli account degli amministratori di WebSphere MQ .

Queste attività sono descritte in:

- Creazione e gestione di gruppi in Windows
- Creazione e gestione di gruppi in HP-UX
- Creazione e gestione di gruppi su AIX
- Creazione e gestione di gruppi su Solaris

- Creazione e gestione di gruppi su Linux

Se il controller di dominio viene eseguito su Windows 2000 o Windows 2003, l'amministratore del dominio potrebbe dover impostare un account speciale da utilizzare per WebSphere MQ . Ciò è descritto in Configurazione degli account WebSphere MQ.

Autorizzazione per gestire oggetti IBM WebSphere MQ su sistemi UNIX, Linux, and Windows

Tutti gli oggetti sono protetti da IBM WebSphere MQe ai principal deve essere fornita l'autorità appropriata per accedervi. Principal differenti richiedono diritti di accesso differenti per oggetti differenti.

I gestori code, le code, le definizioni dei processi, gli elenchi dei nomi, i canali, i canali di connessione client, i listener, i servizi e gli oggetti delle informazioni di autenticazione sono tutti accessibili dalle applicazioni che utilizzano chiamate MQI o comandi PCF. Queste risorse sono tutte protette da WebSphere MQe le applicazioni devono disporre dell'autorizzazione per accedervi. L'entità che effettua la richiesta può essere un utente, un programma applicativo che emette una chiamata MQI o un programma di amministrazione che emette un comando PCF. L'identificativo del richiedente viene definito *principal*.

A differenti gruppi di principal possono essere concessi diversi tipi di autorizzazioni di accesso allo stesso oggetto. Ad esempio, per una coda specifica, è possibile che ad un gruppo sia consentito eseguire operazioni di inserimento e di acquisizione; ad un altro gruppo potrebbe essere consentito solo di sfogliare la coda (MQGET con l'opzione di esplorazione). Allo stesso modo, alcuni gruppi potrebbero avere l'autorizzazione di inserimento e acquisizione per una coda, ma non possono modificare gli attributi della coda o eliminarla.

Alcune operazioni sono particolarmente sensibili e dovrebbero essere limitate agli utenti privilegiati. Ad esempio:

- Accesso ad alcune code speciali, come le code di trasmissione o la coda comandi SYSTEM.ADMIN.COMMAND.QUEUE
- Esecuzione di programmi che utilizzano le opzioni di contesto MQI complete
- Creazione ed eliminazione di code di applicazioni

L'autorizzazione di accesso completo per un oggetto viene fornita automaticamente all'ID utente che ha creato l'oggetto e a tutti i membri del gruppo mqm (e ai membri del gruppo Administrators locale su sistemi Windows).

Concetti correlati

“Autorizzazione per la gestione di IBM WebSphere MQ su sistemi UNIX, Linux, and Windows” a pagina 197

Gli amministratori IBM WebSphere MQ possono utilizzare tutti i comandi IBM WebSphere MQ e concedere le autorizzazioni per altri utenti. Quando gli amministratori immettono comandi ai gestori code remoti, devono disporre dell'autorizzazione richiesta sul gestore code remoto. Ulteriori considerazioni si applicano ai sistemi Windows .

Quando vengono eseguiti i controlli di sicurezza sui sistemi UNIX, Linux, and Windows

I controlli di sicurezza vengono generalmente eseguiti durante la connessione a un gestore code, l'apertura o la chiusura di oggetti e l'inserimento o il richiamo di messaggi.

I controlli di sicurezza effettuati per un'applicazione tipica sono i seguenti:

Connessione al gestore code (chiamate MQCONN o MQCONNX)

Questa è la prima volta che l'applicazione viene associata a un determinato gestore code. Il gestore code interroga l'ambiente operativo per rilevare l'ID utente associato con l'applicazione. WebSphere MQ verifica quindi che l'ID utente sia autorizzato a connettersi al gestore code e conserva l'ID utente per le verifiche future.

Gli utenti non devono collegarsi a WebSphere MQ; WebSphere MQ presuppone che gli utenti si siano collegati al sistema operativo sottostante e che siano stati autenticati da tale sistema.

Apertura dell'oggetto (chiamate MQOPEN o MQPUT1)
WebSphere Gli oggetti MQ vengono acceduti aprendo l'oggetto e immettendo i relativi comandi. Tutti i controlli delle risorse vengono eseguiti quando l'oggetto viene aperto, piuttosto che quando vi si accede effettivamente. Ciò significa che la richiesta **MQOPEN** deve specificare il tipo di accesso richiesto (ad esempio, se l'utente desidera solo sfogliare l'oggetto o eseguire un aggiornamento come l'inserimento di messaggi in una coda).

WebSphere MQ controlla la risorsa indicata nella richiesta **MQOPEN** . Per un alias o un oggetto della coda remota, l'autorizzazione utilizzata è quella dell'oggetto stesso, non la coda in cui si risolve l'alias o la coda remota. Ciò significa che l'utente non ha bisogno dell'autorizzazione per accedervi. Limitare l'autorizzazione a creare code per utenti privilegiati. In caso contrario, gli utenti potrebbero ignorare il normale controllo degli accessi semplicemente creando un alias. Se si fa riferimento esplicitamente a una coda remota con i nomi della coda e del gestore code, viene controllata la coda di trasmissione associata al gestore code remoto.

L'autorizzazione a una coda dinamica è basata sulla coda modello da cui è derivata, ma non è necessariamente la stessa. Ciò è descritto nella nota “1” a pagina 91.

L'ID utente utilizzato dal gestore code per i controlli di accesso è l'ID utente ottenuto dall'ambiente operativo dell'applicazione connessa al gestore code. Un'applicazione adeguatamente autorizzata può emettere una chiamata **MQOPEN** specificando un ID utente alternativo; le verifiche del controllo accessi vengono quindi effettuate sull'ID utente alternativo. Ciò non modifica l'ID utente associato con l'applicazione, ma solo quello utilizzato per le verifiche del controllo accessi.

Inserimento e ricezione di messaggi (chiamate MQPUT o MQGET)
Non viene eseguito alcun controllo di accesso.

Chiusura dell'oggetto (MQCLOSE)
Non viene eseguita alcuna verifica del controllo accessi, a meno che **MQCLOSE** non determini l'eliminazione di una coda dinamica. In questo caso, si verifica che l'ID utente sia autorizzato ad eliminare la coda.

Sottoscrizione a un argomento (MQSUB)
Quando un'applicazione sottoscrive un argomento, specifica il tipo di operazioni che deve eseguire. Si tratta di creare una nuova sottoscrizione, modificare una sottoscrizione esistente o riprendere una sottoscrizione esistente senza modificarla. Per ogni tipo di operazione, il gestore code verifica che l'ID utente associato all'applicazione disponga dell'autorizzazione per eseguire l'operazione.

Quando un'applicazione sottoscrive un argomento, i controlli di autorizzazione vengono eseguiti rispetto agli oggetti argomento che si trovano nella struttura ad albero degli argomenti al punto, o al punto superiore, nella struttura ad albero degli argomenti in cui l'applicazione ha effettuato la sottoscrizione. I controlli di autorizzazione potrebbero comportare controlli su più di un oggetto argomento.

L'ID utente che il gestore code utilizza per i controlli delle autorizzazioni è l'ID utente ottenuto dal sistema operativo quando l'applicazione si connette al gestore code.

Il gestore code esegue controlli di autorizzazione sulle code del sottoscrittore ma non sulle code gestite.

Modalità di implementazione del controllo degli accessi da parte di IBM WebSphere MQ su sistemi UNIX, Linux, and Windows

IBM WebSphere MQ utilizza i servizi di sicurezza forniti dal sistema operativo sottostante, utilizzando il gestore autorizzazioni oggetto. IBM WebSphere MQ fornisce comandi per la creazione e la gestione degli elenchi di controllo accessi.

Un'interfaccia di controllo accessi denominata Authorization Service Interface fa parte di WebSphere MQ. WebSphere MQ fornisce un'implementazione di un gestore controllo accessi (conforme all'Authorization Service Interface) noto come *OAM (object authority manager)*. Viene installato e abilitato automaticamente per ciascun gestore code creato, a meno che non venga specificato diversamente (come descritto in “Come impedire i test di accesso di sicurezza sui sistemi UNIX, Linux, and Windows” a

pagina 167). L'OAM può essere sostituito da qualsiasi componente scritto da un utente o da un fornitore conforme all'Authorization Service Interface.

OAM utilizza le funzioni di sicurezza del sistema operativo sottostante, utilizzando ID utente e gruppo del sistema operativo. Gli utenti possono accedere agli oggetti WebSphere MQ solo se dispongono dell'autorizzazione corretta. “Controllo dell'accesso agli oggetti utilizzando OAM su sistemi UNIX, Linux e Windows” a pagina 159 descrive come concedere e revocare questa autorizzazione.

OAM gestisce un ACL (access control list) per ogni risorsa che controlla. I dati di autorizzazione vengono memorizzati su una coda locale denominata SYSTEM.AUTH.DATA.QUEUE. L'accesso a questa coda è limitato agli utenti del gruppo mqm e, inoltre, su Windows, agli utenti del gruppo Administrators e agli utenti collegati con l'ID SYSTEM. L'accesso utente alla coda non può essere modificato.

WebSphere MQ fornisce i comandi per creare e gestire gli elenchi di controllo accessi. Per ulteriori informazioni su questi comandi, consultare “Controllo dell'accesso agli oggetti utilizzando OAM su sistemi UNIX, Linux e Windows” a pagina 159.

WebSphere MQ passa all'OAM una richiesta contenente un principal, un nome risorsa e un tipo di accesso. L'OAM concede o rifiuta l'accesso in base all'ACL che gestisce. WebSphere MQ segue la decisione di OAM; se OAM non può prendere una decisione, WebSphere MQ non consente l'accesso.

Identificazione dell'ID utente sui sistemi UNIX, Linux, and Windows

Il gestore autorizzazioni oggetto identifica il principal che sta richiedendo l'accesso a una risorsa. L'ID utente utilizzato come principal varia in base al contesto.

OAM (Object Authority Manager) deve essere in grado di identificare chi richiede l'accesso a una particolare risorsa. IBM WebSphere MQ utilizza il termine *principal* per fare riferimento a questo identificativo. Il principal viene stabilito quando l'applicazione si connette per la prima volta al gestore code; viene determinato dal gestore code dall'ID utente associato all'applicazione in fase di connessione. (Se l'applicazione emette chiamate XA senza connettersi al gestore code, l'ID utente associato all'applicazione che emette la chiamata xa_open viene utilizzato per i controlli delle autorità da parte del gestore code.)

Su sistemi UNIX and Linux , le routine di autorizzazione verificano l'ID utente reale (loggato) o l'ID utente effettivo associato all'applicazione. L'ID utente selezionato può essere dipendente dal tipo di bind, per i dettagli consultare Servizi installabili.

IBM WebSphere MQ trasmette l'ID utente ricevuto dal sistema nell'intestazione del messaggio (struttura MQMD) di ciascun messaggio come identificazione dell'utente. Questo identificativo fa parte delle informazioni di contesto del messaggio ed è descritto in “Autorizzazione contesto su sistemi UNIX, Linux e Windows” a pagina 203. Le applicazioni non possono modificare queste informazioni a meno che non siano state autorizzate a modificare le informazioni di contesto.

Principal e gruppi

I principal possono appartenere a gruppi. È possibile concedere l'accesso a una particolare risorsa ai gruppi piuttosto che agli individui, per ridurre la quantità di amministrazione richiesta. Su sistemi UNIX and Linux , tutti gli ACL (Access Control List) si basano su gruppi, ma su sistemi Windows , gli ACLS si basano su ID utente e gruppi.

Ad esempio, è possibile definire un gruppo composto da utenti che desiderano eseguire una particolare applicazione. Ad altri utenti è possibile fornire l'accesso a tutte le risorse richieste aggiungendo il proprio ID utente al gruppo appropriato. Questo processo è descritto in:

- Creazione e gestione di gruppi in Windows
- Creazione e gestione di gruppi in HP-UX
- Creazione e gestione di gruppi su AIX
- Creazione e gestione di gruppi su Solaris
- Creazione e gestione di gruppi su Linux

Un principal può appartenere a più di un gruppo (la sua serie di gruppi). Ha l'aggregato di tutte le autorità concesse a ciascun gruppo nella sua serie di gruppi. Tali autorizzazioni vengono memorizzate nella cache, quindi tutte le modifiche apportate all'appartenenza al gruppo del principal non vengono riconosciute fino a quando il gestore code non viene riavviato, a meno che non si immette il comando MQSC REFRESH SECURITY (o l'equivalente PCF).

Sistemi UNIX and Linux

Tutti gli ACL si basano sui gruppi. Quando a un utente viene concesso l'accesso a una particolare risorsa, il gruppo primario dell'ID utente viene incluso nell'ACL. L'ID utente individuale non viene incluso e l'autorizzazione viene concessa a tutti i membri di tale gruppo. Per questo motivo, è possibile modificare inavvertitamente l'autorizzazione di un principal modificando l'autorizzazione di un altro principal nello stesso gruppo. Tutti gli utenti sono assegnati nominalmente al gruppo utenti predefinito *nessuno* e, per impostazione predefinita, a questo gruppo non viene concessa alcuna autorizzazione. È possibile modificare l'autorizzazione nel gruppo *nobody* per concedere l'accesso alle risorse WebSphere MQ agli utenti senza autorizzazioni specifiche.

Non definire un ID utente con il valore "SCONOSCIUTO". Il valore "UNKNOWN" viene utilizzato quando un ID utente è troppo lungo, quindi ID utente arbitrari utilizzeranno le autorizzazioni di accesso di UNKNOWN.

Gli ID utente possono contenere fino a 12 caratteri e i nomi gruppo fino a 12 caratteri.

Sistemi Windows

Gli ACL si basano su ID utente e gruppi. I controlli sono gli stessi dei sistemi UNIX , ad eccezione del fatto che i singoli ID utente possono essere visualizzati anche nell'ACL. È possibile avere utenti differenti su domini differenti con lo stesso ID utente. WebSphere MQ consente agli ID utente di essere qualificati da un nome dominio in modo che a questi utenti possano essere assegnati diversi livelli di accesso.

Il nome del gruppo può facoltativamente includere un nome dominio, specificato nei formati seguenti:

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

I gruppi globali vengono controllati da OAM solo in due casi:

1. La stanza di sicurezza del gestore code include l'impostazione `GroupModel=GlobalGroups`; consultare Sicurezza.
2. Il gestore code sta utilizzando un gruppo di accesso di protezione alternativo; consultare **crtmqm**.

Gli ID utente possono contenere fino a 20 caratteri, nomi dominio fino a 15 caratteri e nomi gruppo fino a 64 caratteri.

L'OAM controlla prima il database di sicurezza locale, quindi il database del dominio primario e infine il database di tutti i domini attendibili. Il primo ID utente rilevato viene utilizzato da OAM per il controllo. Ognuno di questi ID utente potrebbe avere appartenenze a gruppi differenti su un particolare computer.

Alcuni comandi di controllo (ad esempio, `crtmqm`) modificano le autorizzazioni su oggetti WebSphere MQ utilizzando OAM (object authority manager). L'OAM ricerca i database di sicurezza nell'ordine indicato nel paragrafo precedente per stabilire i diritti di autorità per un particolare ID utente. Di conseguenza, l'autorizzazione determinata da OAM potrebbe sovrascrivere il fatto che un ID utente è membro del gruppo mqm locale. Ad esempio, se si immette il comando `crtmqm` da un ID utente autenticato da un controller di dominio che appartiene al gruppo mqm locale tramite un gruppo globale, il comando ha esito negativo se il sistema ha un utente locale con lo stesso nome che non fa parte del gruppo mqm locale.

Identificativi di sicurezza (SID) Windows

WebSphere MQ su Windows utilizza il SID in cui è disponibile. Se un SID Windows non viene fornito con una richiesta di autorizzazione, WebSphere MQ identifica l'utente in base al solo nome utente, ma ciò potrebbe comportare la concessione di un'autorizzazione non corretta.

Su sistemi Windows , il SID (security identifier) viene utilizzato per integrare l'ID utente. Il SID contiene le informazioni che identificano i dettagli completi dell'account utente sul database SAM (security account manager) Windows in cui è definito l'utente. Quando un messaggio viene creato su WebSphere MQ per Windows, WebSphere MQ memorizza il SID nel descrittore del messaggio. Quando WebSphere MQ on Finestre esegue i controlli di autorizzazione, utilizza il SID per interrogare tutte le informazioni dal database SAM. Il database SAM in cui è definito l'utente deve essere accessibile perché questa query abbia esito positivo.

Per impostazione predefinita, se un SID Windows non viene fornito con una richiesta di autorizzazione, WebSphere MQ identifica l'utente in base al solo nome utente. Esegue questa operazione effettuando una ricerca nei database di sicurezza nel seguente ordine:

1. Il database di sicurezza locale
2. Il database di sicurezza del dominio primario
3. Il database di sicurezza dei domini attendibili

Se il nome utente non è univoco, è possibile che venga concessa l'autorizzazione WebSphere MQ non corretta. Per prevenire questo problema, includere un SID in ogni richiesta di autorizzazione; il SID viene utilizzato da WebSphere MQ per stabilire credenziali utente.

Per indicare che tutte le richieste di autorizzazione devono includere un SID, utilizzare `regedit`. Impostare SecurityPolicy su NTSIDsRequired.

Autorizzazione utente alternativo su sistemi UNIX, Linux e Windows

È possibile indicare che un ID utente può utilizzare l'autorità di un altro utente quando accede a un oggetto WebSphere MQ . Questa è denominata *autorizzazione utente alternativo*ed è possibile utilizzarla su qualsiasi oggetto WebSphere MQ .

L'autorizzazione utente alternativo è essenziale quando un server riceve richieste da un programma e desidera assicurarsi che il programma disponga dell'autorità richiesta per la richiesta. Il server potrebbe disporre dell'autorizzazione necessaria, ma deve sapere se il programma dispone dell'autorizzazione per le azioni richieste.

Ad esempio, si supponga che un programma server in esecuzione con l'ID utente PAYSERV richiami un messaggio di richiesta da una coda che è stata inserita nella coda dall'ID utente USER1. Quando il programma del server richiama il messaggio di richiesta, elabora la richiesta e reinserisce la risposta nella coda di risposta specificata con il messaggio di richiesta. Invece di utilizzare il proprio ID utente (PAYSERV) per autorizzare l'apertura della coda di risposta, il server può specificare un ID utente differente, in questo caso, USER1. In questo esempio, è possibile utilizzare l'autorizzazione utente alternativo per controllare se PAYSERV può specificare USER1 come ID utente alternativo quando apre la coda di risposta.

L'ID utente alternativo viene specificato nel campo **`AlternateUserId`** del descrittore oggetto.

Autorizzazione contesto su sistemi UNIX, Linux e Windows

Il contesto è un'informazione che si applica a un particolare messaggio ed è contenuta nel descrittore del messaggio, MQMD, che fa parte del messaggio. Le applicazioni possono specificare i dati di contesto quando viene effettuata una chiamata `MQOPEN` o `MQPUT` .

Le informazioni di contesto si trovano in due sezioni:

Sezione Identità
Da chi proviene il messaggio. È costituito dai campi `UserIdentifier`, `AccountingToken`e `ApplIdentityData` .

Sezione Origine
Da dove proviene il messaggio e quando è stato inserito nella coda. È costituito dai campi `PutApplType`, `PutApplName`, `PutDate`, `PutTimee` `ApplOriginData` .

Le applicazioni possono specificare i dati di contesto quando viene effettuata una chiamata `MQOPEN` o `MQPUT` . Questi dati possono essere generati dall'applicazione, trasmessi da un altro messaggio o generati

dal gestore code per impostazione predefinita. Ad esempio, i dati di contesto possono essere utilizzati dai programmi del server per controllare l'identità del richiedente, verificando se il messaggio proviene da un'applicazione in esecuzione con un ID utente autorizzato.

Un programma server può utilizzare `UserIdentifier` per determinare l'ID utente di un utente alternativo. Si utilizza l'autorizzazione di contesto per controllare se l'utente può specificare una delle opzioni di contesto su una chiamata `MQOPEN` o `MQPUT1` .

Consultare Controllo delle informazioni di contesto per informazioni sulle opzioni di contesto e Panoramica per MQMD per le descrizioni dei campi del descrittore del messaggio relativi al contesto.

Implementazione del controllo accessi nelle uscite di sicurezza

È possibile implementare il controllo accessi in un'uscita di sicurezza utilizzando MCAUserIdentifier o il gestore autorizzazioni oggetto.

MCAUserIdentifier

Ogni istanza di un canale corrente ha una struttura di definizioni di canale associata, MQCD. I valori iniziali dei campi in MQCD sono determinati dalla definizione di canale creata da un amministratore WebSphere MQ . In particolare, il valore iniziale di uno dei campi, *MCAUserIdentifier*, è determinato dal valore del parametro MCAUSER nel comando DEFINE CHANNEL o dall'equivalente di MCAUSER se la definizione del canale viene creata in un altro modo. *MCAUserIdentifier* contiene i primi 12 byte dell'identificativo utente MCA. Se l'identificativo dell'utente MCA non è vuoto, specifica l'identificativo utente che deve essere utilizzato dall'agent del canale dei messaggi per l'autorizzazione ad accedere alle risorse MQ . Verificare che MCAUSER sia inferiore a 12 caratteri sulla piattaforma Windows.

La struttura MQCD viene inoltrata a un programma di uscita del canale quando viene richiamato da un MCA. Quando un'uscita di sicurezza viene richiamata da un MCA, l'uscita di sicurezza può modificare il valore di *MCAUserIdentifier*, sostituendo qualsiasi valore specificato nella definizione del canale.

Sui sistemi IBM i, UNIX, Linux e Windows , a meno che il valore di *MCAUserIdentifier* non sia vuoto, il gestore code utilizza *MCAUserIdentifier* come ID utente per i controlli di autorizzazione quando un MCA tenta di accedere alle risorse del gestore code dopo essersi connesso al gestore code. Se il valore di *MCAUserIdentifier* è vuoto, il gestore code utilizza l'ID utente predefinito dell'MCA. Ciò si applica ai canali RCVR, RQSTR, CLUSRCVR e SVRCONN. Per l'invio di MCA, l'ID utente predefinito viene sempre utilizzato per i controlli di autorizzazione, anche se il valore di *MCAUserIdentifier* non è vuoto.

Su z/OS, il gestore code potrebbe utilizzare il valore *MCAUserIdentifier* per i controlli dell'autorità, purché non sia vuoto. Per ricevere gli MCA e gli MCA di connessione del server, se il gestore code utilizza il valore di *MCAUserIdentifier* per i controlli delle autorizzazioni dipende da:

- Il valore del parametro PUTAUT nella definizione del canale
- Il profilo RACF utilizzato per le verifiche
- Il livello di accesso dell'ID utente dello spazio di indirizzo dell'iniziatore di canali al profilo RESLEVEL

Per l'invio di MCA, dipende da:

- Se l'MCA mittente è un chiamante o un responder
- Il livello di accesso dell'ID utente dello spazio di indirizzo dell'iniziatore di canali al profilo RESLEVEL

L'ID utente memorizzato da un'uscita di sicurezza in *MCAUserIdentifier* può essere acquisito in vari modi. Di seguito sono riportati alcuni esempi:

- Se non vi è alcuna uscita di sicurezza all'estremità client di un canale MQI, un ID utente associato all'applicazione client WebSphere MQ fluisce dall'MCA di connessione client all'MCA di connessione server quando l'applicazione client emette una chiamata MQCONN.L'MCA di connessione del server memorizza questo ID utente nel campo *RemoteUserIdentifier* nella struttura di definizione del canale, MQCD. Se il valore di *MCAUserIdentifier* è vuoto in questo momento, l'MCA memorizza lo stesso ID utente in *MCAUserIdentifier*. Se l'MCA non memorizza l'ID utente in *MCAUserIdentifier*, un'uscita di sicurezza può farlo successivamente impostando *MCAUserIdentifier* sul valore di *RemoteUserIdentifier*.

Se l'ID utente che fluisce dal sistema client sta immettendo un nuovo dominio di sicurezza e non è valido sul sistema server, l'uscita di sicurezza può sostituire l'ID utente con uno valido e memorizzare l'ID utente sostituito in *MCAUserIdentifier*.

- L'ID utente può essere inviato dall'uscita di sicurezza partner in un messaggio di sicurezza.

  Su un canale di messaggi, un'uscita di sicurezza richiamata dall'MCA mittente può inviare l'ID utente con cui è in esecuzione l'MCA mittente. Un'uscita di sicurezza richiamata dall'MCA ricevente può memorizzare l'ID utente in *MCAUserIdentifier*. Allo stesso modo, su un canale MQI, un'uscita di sicurezza all'estremità client del canale può inviare l'ID utente associato all'applicazione client MQI WebSphere MQ . Un'uscita di sicurezza all'estremità server del canale può quindi memorizzare l'ID utente in *MCAUserIdentifier*. Come nell'esempio precedente, se l'ID utente non è valido sul sistema di destinazione, l'uscita di sicurezza può sostituire l'ID utente con uno valido e memorizzare l'ID utente sostituito in *MCAUserIdentifier*.

  Se un certificato digitale viene ricevuto come parte del servizio di identificazione e autenticazione, un'uscita di sicurezza può associare il DN (Distinguished Name) nel certificato a un ID utente valido sul sistema di destinazione. Può quindi memorizzare l'ID utente in *MCAUserIdentifier*.

- Se si utilizza SSL sul canale, il DN (Distinguished Name) del partner viene passato all'uscita nel campo SSLPeerNamePtr di MQCD e il DN dell'emittente di tale certificato viene passato all'uscita nel campo SSLRemCertIssNamePtr di MQCXP.

Per ulteriori informazioni sul campo *MCAUserIdentifier* , la struttura di definizione del canale, MQCD e la struttura del parametro di uscita del canale, MQCXP, consultare Chiamate di uscita del canale e strutture dati. Per ulteriori informazioni sull'ID utente che fluisce da un sistema client su un canale MQI, consultare Controllo accessi.

Nota: Le applicazioni di uscita di sicurezza create prima della release di WebSphere MQ v7.1 potrebbero richiedere un aggiornamento. Per ulteriori informazioni, consultare Programmi di uscita di sicurezza del canale.

WebSphere MQ object authority manager autenticazione utente

Nelle connessioni client WebSphere MQ MQI, è possibile utilizzare le uscite di sicurezza per modificare o creare la struttura MQCSP utilizzata nell'autenticazione utente OAM (object authority manager). Ciò è descritto in Programmi di uscita canale per i canali di messaggistica

Implementazione del controllo accessi nelle uscite dei messaggi

Potrebbe essere necessario utilizzare un'uscita messaggio per sostituire un ID utente con un altro.

Considerare un'applicazione client che invia un messaggio a un'applicazione server. L'applicazione server può estrarre l'ID utente dal campo *UserIdentifier* nel descrittore del messaggio e, se dispone di autorizzazione utente alternativa, chiedere al gestore code di utilizzare questo ID utente per i controlli di autorizzazione quando accede alle risorse WebSphere MQ per conto del client.

Se il parametro PUTAUT è impostato su CTX (o ALTMCA su z/OS) nella definizione del canale, l'ID utente nel campo *UserIdentifier* di ciascun messaggio in entrata viene utilizzato per i controlli di autorizzazione quando l'MCA apre la coda di destinazione.

In alcune circostanze, quando viene generato un messaggio di report, viene inserito utilizzando l'autorizzazione dell'ID utente nel campo *UserIdentifier* del messaggio che causa il report. In particolare, i report COD (confirm - on - delivery) e i report di scadenza vengono sempre inseriti con questa autorizzazione.

A causa di queste situazioni, potrebbe essere necessario sostituire un ID utente con un altro nel campo *UserIdentifier* quando un messaggio entra in un nuovo dominio di protezione. Questa operazione può essere eseguita da un'exit dei messaggi all'estremità ricevente del canale. In alternativa, è possibile verificare che l'ID utente nel campo *UserIdentifier* di un messaggio in entrata sia definito nel nuovo dominio di sicurezza.

Se un messaggio in entrata contiene un certificato digitale per l'utente dell'applicazione che ha inviato il messaggio, un'uscita messaggio può convalidare il certificato e associare il DN (Distinguished Name) nel

certificato a un ID utente valido sul sistema ricevente. È quindi possibile impostare il campo *UserIdentifier* nel descrittore del messaggio su questo ID utente.

Se è necessario che un'uscita del messaggio modifichi il valore del campo *UserIdentifier* in un messaggio in arrivo, potrebbe essere appropriato che l'uscita del messaggio autentichi il mittente del messaggio contemporaneamente. Per ulteriori dettagli, vedere “Mappature di identità nelle uscite del messaggio” a pagina 148.

Implementazione del controllo accessi nell'uscita API e nell'uscita incrociata API

Un'API o un'API - crossing exit possono fornire controlli di accesso per integrare quelli forniti da WebSphere MQ. In particolare, l'uscita può fornire il controllo accessi a livello di messaggio. L'uscita può garantire che un'applicazione immetta in una coda o riceva da una coda solo i messaggi che soddisfano determinati criteri.

Considerare i seguenti esempi:

- Un messaggio contiene informazioni su un ordine. Quando un'applicazione tenta di inserire un messaggio in una coda, un'uscita API o di attraversamento API può verificare che il valore totale dell'ordine sia inferiore a qualche limite prescritto.
- I messaggi arrivano su una coda di destinazione dai gestori code remoti. Quando un'applicazione tenta di richiamare un messaggio dalla coda, un'uscita API o API può controllare che il mittente del messaggio sia autorizzato a inviare un messaggio alla coda.

Riservatezza dei messaggi

Per mantenere la riservatezza, crittografare i messaggi. Esistono diversi modi per codificare i messaggi in WebSphere MQ , in base alle proprie esigenze.

La scelta di CipherSpec determina il livello di riservatezza di cui si dispone.

Se è necessaria una protezione dei dati end-to-end a livello di applicazione per l'infrastruttura di messaggistica point - to - point, è possibile utilizzare WebSphere MQ Advanced Message Security per crittografare i messaggi o scrivere la propria uscita API o l'uscita incrociata API.

Se è necessario crittografare i messaggi solo mentre vengono trasportati attraverso un canale, poiché si dispone di una sicurezza adeguata sui gestori code, è possibile utilizzare SSL o TLS oppure scrivere la propria uscita di sicurezza, l'uscita del messaggio o i programmi di uscita di invio e ricezione.

Per ulteriori informazioni su WebSphere MQ Advanced Message Security, consultare “pianificazione per Advanced Message Security” a pagina 64.L'utilizzo di SSL e TLS con WebSphere MQ è descritto in “Supporto IBM WebSphere MQ per SSL e TLS” a pagina 23. L'utilizzo dei programmi di uscita nella codifica dei messaggi è descritto in “Implementazione della riservatezza nei programmi di uscita utente” a pagina 225.

Connessione di due gestori code mediante SSL o TLS

Le comunicazioni sicure che utilizzano i protocolli di sicurezza crittografica SSL o TLS richiedono l'impostazione dei canali di comunicazione e la gestione dei certificati digitali che verranno utilizzati per l'autenticazione.

Per configurare l'installazione SSL o TLS è necessario definire i canali per utilizzare SSL o TLS. È inoltre necessario ottenere e gestire i certificati digitali. Su un sistema di test, è possibile utilizzare certificati autofirmati o certificati emessi da un'autorità di certificazione (CA) locale. Su un sistema di produzione, non utilizzare certificati autofirmati. Per ulteriori informazioni, consultare ..//zs14140_.dita.

Per informazioni complete sulla creazione e la gestione dei certificati, consultare “Utilizzo di SSL o TLS su sistemi UNIX, Linux, and Windows” a pagina 114.

Questa raccolta di argomenti introduce le attività relative all'impostazione delle comunicazioni SSL e fornisce una guida dettagliata sul completamento di tali attività.

Si potrebbe anche voler verificare l'autenticazione client SSL o TLS, che sono una parte facoltativa dei protocolli. Durante l'handshake SSL o TLS, il client SSL o TLS ottiene e convalida sempre un certificato digitale dal server. Con l'implementazione WebSphere MQ , il server SSL o TLS richiede sempre un certificato dal client.

Note:

1. In questo contesto, un client SSL fa riferimento alla connessione che avvia l'handshake.
2. Consultare il Glossario per ulteriori dettagli.

Su sistemi UNIX, Linux e Windows , il client SSL o TLS invia un certificato solo se ha un'etichetta nel formato WebSphere MQ corretto, `ibmwebspheremq` seguito dal nome del gestore code modificato in minuscolo. Ad esempio, per `QM1`, `ibmwebspheremqqm1`.

WebSphere MQ utilizza il prefisso `ibmwebspheremq` su un'etichetta per evitare confusione con i certificati per altri prodotti. Assicurarsi di specificare l'intera etichetta del certificato in minuscolo.

Il server SSL o TLS convalida sempre il certificato client, se ne viene inviato uno. Se il client non invia un certificato, l'autenticazione non riesce solo se la fine del canale che agisce come server SSL o TLS è definita con il parametro SSLCAUTH impostato su REQUIRED o con un valore del parametro SSLPEER impostato. Per ulteriori informazioni sulla connessione anonima di un gestore code, ossia quando il client SSL o TLS non invia un certificato, consultare “Connessione di due gestori code utilizzando l'autenticazione unidirezionale” a pagina 211.

Utilizzo di certificati autofirmati per l'autenticazione reciproca di due gestori code

Seguire queste istruzioni di esempio per implementare l'autenticazione reciproca tra due gestori code, utilizzando certificati SSL o TLS autofirmati.

Informazioni su questa attività

Scenario:

- Sono presenti due gestori code, QM1 e QM2, che devono comunicare in modo sicuro. È necessaria l'autenticazione reciproca tra QM1 e QM2.
- Si è deciso di verificare la propria comunicazione protetta utilizzando certificati autofirmati.

La configurazione risultante è simile alla seguente:

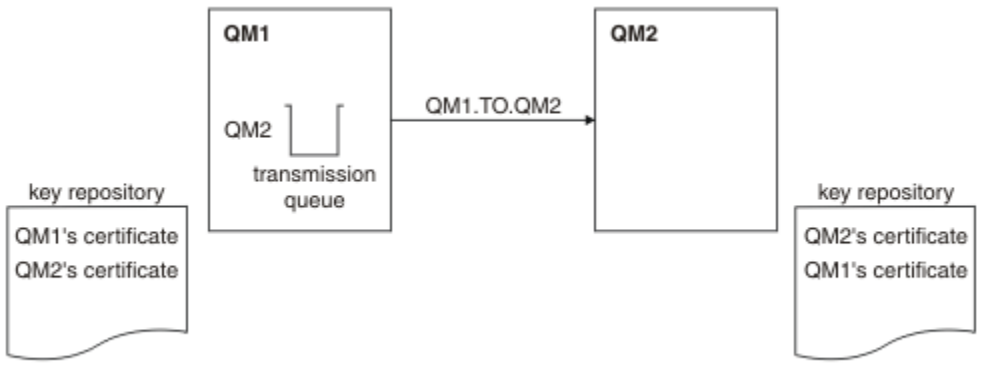


Figura 14. Configurazione risultante da questa attività

In Figura 14 a pagina 207, il repository delle chiavi per QM1 contiene il certificato per QM1 e il certificato pubblico da QM2. Il repository delle chiavi per QM2 contiene il certificato per QM2 e il certificato pubblico da QM1.

Procedura

1. Preparare il repository delle chiavi su ciascun gestore code, in base al sistema operativo:
   - Su sistemi UNIX, Linuxe Windows.
2. Creare un certificato autofirmato per ciascun gestore code:
   - Su sistemi UNIX, Linuxe Windows.
3. Estrarre una copia di ciascun certificato:
   - Su sistemi UNIX, Linuxe Windows.
4. Trasferire la parte pubblica del certificato QM1 sul sistema QM2 e viceversa, utilizzando un programma di utilità come FTP.
5. Aggiungere il certificato partner al repository delle chiavi per ciascun gestore code:
   - Su sistemi UNIX, Linuxe Windows.
6. In QM1, definire un canale mittente e la coda di trasmissione associata, immettendo comandi simili al seguente esempio:

   ```
   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)
   ```

   Questo esempio utilizza CipherSpec RC4_MD5. Le CipherSpecs ad ogni estremità del canale devono essere uguali.
7. Su QM2, definire un canale ricevente immettendo un comando simile al seguente esempio:

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

   Il canale deve avere lo stesso nome del canale mittente definito nel passo 6 e utilizzare lo stesso CipherSpec.
8. Avviare il canale.

Risultati

I repository e i canali chiave vengono creati come illustrato in Figura 14 a pagina 207

Operazioni successive

Verificare che l'attività sia stata completata correttamente utilizzando i comandi DISPLAY. Se l'attività ha avuto esito positivo, l'output risultante è simile a quello mostrato nei seguenti esempi.

Dal gestore code QM1, immettere il comando seguente:

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

L'output risultante è simile al seguente esempio:

```
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)
```

Dal gestore code QM2, immettere il seguente comando:

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

L'output risultante è simile al seguente esempio:

```
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( )
```

In ogni caso, il valore di SSLPE deve corrispondere a quello del DN nel certificato partner creato nel passo 2. Il nome degli emittenti corrisponde al nome peer perché il certificato è autofirmato.

SSLPEER è facoltativo. Se viene specificato, il valore deve essere impostato in modo che sia consentito il DN nel certificato partner (creato nel passo 2). Per ulteriori informazioni relative all'utilizzo di SSLPEER, consultare WebSphere MQ rules for SSLPEER values .

Utilizzo dei certificati firmati dalla CA per l'autenticazione reciproca di due gestori code

Segui queste istruzioni di esempio per implementare l'autenticazione reciproca tra due gestori code, utilizzando i certificati SSL o TLS firmati da CA.

Informazioni su questa attività

Scenario:

- Si hanno due gestori code denominati QMA e QMB, che devono comunicare in modo sicuro. È necessaria l'autenticazione reciproca tra QMA e QMB.
- In futuro si prevede di utilizzare questa rete in un ambiente di produzione e quindi si è deciso di utilizzare i certificati firmati dalla CA fin dall'inizio.

La configurazione risultante è simile alla seguente:

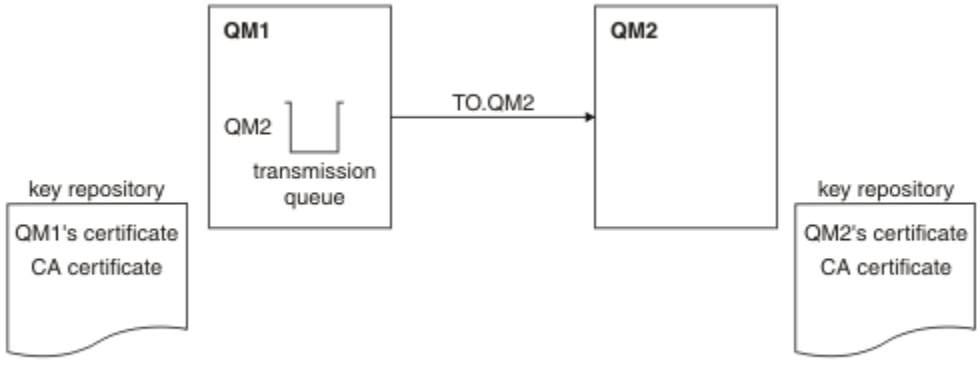


Figura 15. Configurazione risultante da questa attività

In Figura 15 a pagina 209, il repository delle chiavi per QMA contiene il certificato di QMA e il certificato CA. Il repository delle chiavi per QMB contiene il certificato QMB e il certificato CA. In questo esempio, sia il certificato QMA che il certificato QMB sono stati emessi dalla stessa CA. Se il certificato di QMA e il certificato di QMB sono stati emessi da AC differenti, i repository di chiavi per QMA e QMB devono contenere entrambi i certificati CA.

Procedura

1. Preparare il repository delle chiavi su ciascun gestore code, in base al sistema operativo:
   - Su sistemi UNIX, Linuxe Windows.
2. Richiedere un certificato firmato dalla CA per ciascun gestore code.
   È possibile utilizzare diverse CA per i due gestori code.
   - Su sistemi UNIX, Linuxe Windows.
3. Aggiungere il certificato dell'autorità di certificazione al repository delle chiavi per ciascun gestore code:
   Se i gestori code utilizzano autorità di certificazione differenti, il certificato CA per ogni autorità di certificazione deve essere aggiunto a entrambi i repository delle chiavi.
   - Su sistemi UNIX, Linuxe Windows.
4. Aggiungere il certificato firmato dalla CA al repository delle chiavi per ogni gestore code:
   - Su sistemi UNIX, Linuxe Windows.
5. In QMA, definire un canale mittente e la coda di trasmissione associata immettendo comandi come nel seguente esempio:

   ```
   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)
   ```

   Questo esempio utilizza CipherSpec RC4_MD5. Le CipherSpecs ad ogni estremità del canale devono essere uguali.
6. In QMB, definire un canale ricevente immettendo un comando simile al seguente esempio:

   ```
   DEFINE CHANNEL(TO.QMB) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(RC2_MD5_EXPORT)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL to QMB')
   ```

   Il canale deve avere lo stesso nome del canale mittente definito nel passo 6 e utilizzare lo stesso CipherSpec.
7. Avviare il canale:

Risultati

I repository e i canali chiave vengono creati come illustrato in Figura 15 a pagina 209.

Operazioni successive

Verificare che l'attività sia stata completata correttamente utilizzando i comandi DISPLAY. Se l'attività ha avuto esito positivo, l'output risultante è simile a quello mostrato nei seguenti esempi.

Dal gestore code QMA, immettere il comando seguente:

```
DISPLAY CHS(TO.QMB) SSLPEER SSLCERTI
```

L'output risultante è simile al seguente esempio:

```
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)
```

Dal gestore code QMB, immettere il seguente comando:

```
DISPLAY CHS(TO.QMB) SSLPEER SSLCERTI
```

L'output risultante è simile al seguente esempio:

```
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( )
```

In ogni caso, il valore di SSLPEER deve corrispondere a quello del DN (Distinguished Name) nel certificato partner creato nel passo 2. Il nome dell'emittente corrisponde al DN oggetto del certificato CA che ha firmato il certificato personale aggiunto nel Passo 4.

Connessione di due gestori code utilizzando l'autenticazione unidirezionale

Seguire queste istruzioni di esempio per modificare un sistema con l'autenticazione reciproca per consentire a un gestore code di connettersi utilizzando l'autenticazione unidirezionale a un altro; ovvero, quando il client SSL o TLS non invia un certificato.

Informazioni su questa attività

Scenario:

- I due gestori code (QM1 e QM2) sono stati configurati come in "Utilizzo dei certificati firmati dalla CA per l'autenticazione reciproca di due gestori code" a pagina 209.
- Si desidera modificare QM1 in modo che si connetta utilizzando l'autenticazione unidirezionale a QM2.

La configurazione risultante è simile alla seguente:

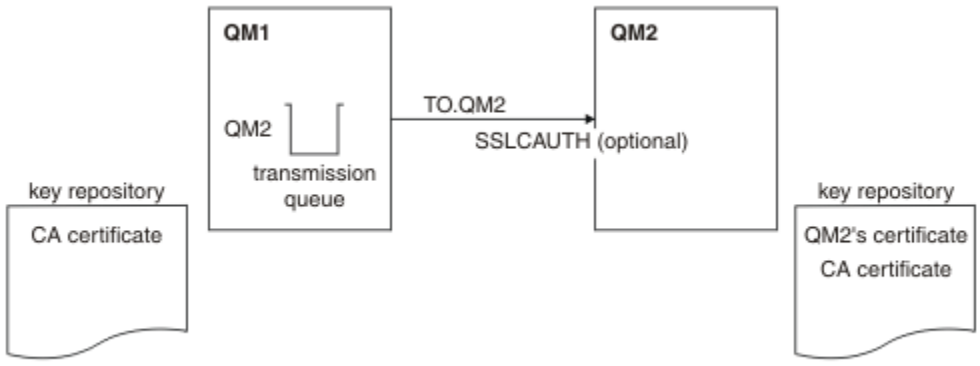


Figura 16. Gestori code che consentono l'autenticazione unidirezionale

Procedura

1. Rimuovere il certificato personale di QM1dal relativo repository delle chiavi, in base al sistema operativo:
   - Su sistemi UNIX, Linuxe Windows. Il certificato è etichettato come segue:
     - `ibmwebspheremq` seguito dal nome del gestore code ridotto in minuscolo. Ad esempio, per `QM1` , `ibmwebspheremqqm1`.

2. Opzionale: In QM1, se i canali SSL o TLS sono stati eseguiti in precedenza, aggiornare l'ambiente SSL o TLS.
3. Consentire connessioni anonime sul ricevitore.

Risultati

I repository chiave e i canali vengono modificati come illustrato in Figura 16 a pagina 211

Operazioni successive

Se il canale mittente era in esecuzione e si è immesso il comando REFRESH SECURITY TYPE (SSL) (nel passo 2), il canale viene riavviato automaticamente. Se il canale mittente non era in esecuzione, avviarlo.

All'estremità del server del canale, la presenza del valore del parametro del nome peer sul pannello di stato del canale indica che è stato trasmesso un certificato client.

Verificare che l'attività sia stata completata correttamente immettendo alcuni comandi DISPLAY. Se l'attività ha avuto esito positivo, l'output risultante è simile a quello mostrato nei seguenti esempi:

Dal gestore code QM1 , immettere il comando seguente:

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

L'output risultante sarà simile al seguente esempio:

```
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)
```

Dal gestore code QM2 , immettere il seguente comando:

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

L'output risultante sarà simile al seguente esempio:

```
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( )
```

In QM2, il campo SSLPE è vuoto e indica che QM1 non ha inviato un certificato. In QM1, il valore di SSLPEER corrisponde a quello del DN nel certificato personale di QM2.

Connessione sicura di un client a un gestore code

Le comunicazioni sicure che utilizzano i protocolli di sicurezza crittografica SSL o TLS richiedono l'impostazione dei canali di comunicazione e la gestione dei certificati digitali che verranno utilizzati per l'autenticazione.

Per configurare l'installazione SSL o TLS è necessario definire i canali per utilizzare SSL o TLS. È inoltre necessario ottenere e gestire i certificati digitali. Su un sistema di test, è possibile utilizzare certificati

autofirmati o certificati emessi da un'autorità di certificazione (CA) locale. Su un sistema di produzione, non utilizzare certificati autofirmati. Per ulteriori informazioni, consultare ..//zs14140_.dita.

Per informazioni complete sulla creazione e la gestione dei certificati, consultare “Utilizzo di SSL o TLS su sistemi UNIX, Linux, and Windows” a pagina 114.

Questa raccolta di argomenti introduce le attività relative all'impostazione delle comunicazioni SSL e fornisce una guida dettagliata sul completamento di tali attività.

Si potrebbe anche voler verificare l'autenticazione client SSL o TLS, che sono una parte facoltativa dei protocolli. Durante l'handshake SSL o TLS, il client SSL o TLS ottiene e convalida sempre un certificato digitale dal server. Con l'implementazione WebSphere MQ , il server SSL o TLS richiede sempre un certificato dal client.

Su sistemi UNIX, Linux, and Windows , il client SSL o TLS invia un certificato solo se ha un'etichetta nel formato WebSphere MQ corretto, che è `ibmwebspheremq` seguito dall'ID utente di collegamento modificato in minuscolo, ad esempio `ibmwebspheremqmyuserid`.

WebSphere MQ utilizza il prefisso `ibmwebspheremq` su un'etichetta per evitare confusione con i certificati per altri prodotti. Assicurarsi di specificare l'intera etichetta del certificato in minuscolo.

Il server SSL o TLS convalida sempre il certificato client, se ne viene inviato uno. Se il client non invia un certificato, l'autenticazione non riesce solo se la fine del canale che agisce come server SSL o TLS è definita con il parametro SSLCAUTH impostato su REQUIRED o con un valore del parametro SSLPEER impostato. Per ulteriori informazioni sulla connessione anonima di un gestore code, consultare “Connessione anonima di un client a un gestore code” a pagina 216.

Utilizzo di certificati autofirmati per l'autenticazione reciproca di un client e di un gestore code

Seguire queste istruzioni di esempio per implementare l'autenticazione reciproca tra un client e un gestore code, utilizzando certificati SSL o TLS autofirmati.

Informazioni su questa attività

Scenario:

- Si dispone di un client C1e di un gestore code, QM1, che devono comunicare in modo sicuro. È necessaria l'autenticazione reciproca tra C1 e QM1.
- Si è deciso di verificare la comunicazione protetta utilizzando i certificati autofirmati.

DCM su IBM i non supporta i certificati autofirmati, pertanto questa attività non è applicabile sui sistemi IBM i .

La configurazione risultante è simile alla seguente:

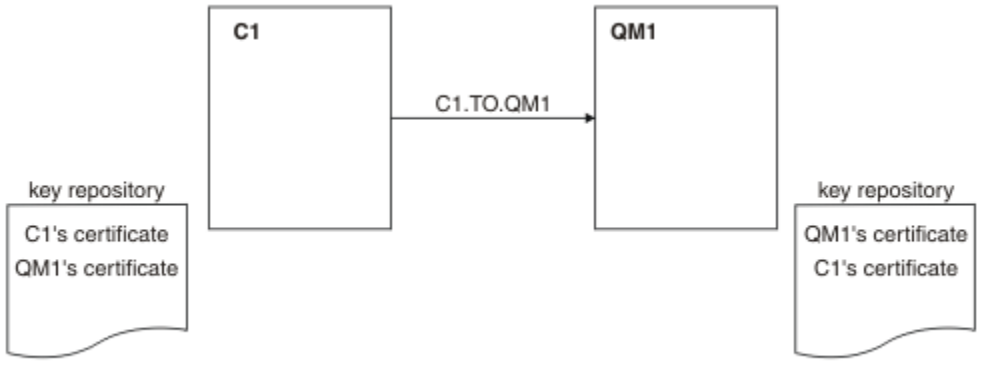


Figura 17. Configurazione risultante da questa attività

In Figura 17 a pagina 213, il repository delle chiavi per QM1 contiene il certificato per QM1 e il certificato pubblico da C1. Il repository delle chiavi per C1 contiene il certificato per C1 e il certificato pubblico da QM1.

Procedura

1. Preparare il repository chiavi sul client e sul gestore code, in base al sistema operativo:
   - Su sistemi UNIX, Linuxe Windows.
2. Creare certificati autofirmati per il client e il gestore code:
   - Su sistemi UNIX, Linuxe Windows.
3. Estrarre una copia di ciascun certificato:
   - Su sistemi UNIX, Linuxe Windows.
4. Trasferire la parte pubblica del certificato C1 al sistema QM1 e viceversa, utilizzando un programma di utilità come FTP.
5. Aggiungere il certificato partner al repository delle chiavi per client e gestore code:
   - Su sistemi UNIX, Linuxe Windows.
6. Immettere il comando REFRESH SECURITY TYPE (SSL) sul gestore code.
7. Definire un canale di connessione client in uno dei seguenti modi:
   - Utilizzando la chiamata MQCONNX con la struttura MQSCO su C1, come descritto in Creazione di un canale di connessione client sul client WebSphere MQ MQI.
   - Utilizzando una tabella di definizione di canale client, come descritto in Creazione di definizioni di connessione server e di connessione client nel server .
8. In QM1, definire un canale di connessione server, immettendo un comando simile al seguente esempio:

   ```
   DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(RC4_MD5_US)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from C1 to QM1')
   ```

   Il canale deve avere lo stesso nome del canale di collegamento client definito nel passo 6 e utilizzare lo stesso CipherSpec.

Risultati

I repository e i canali chiave vengono creati come illustrato in Figura 17 a pagina 213

Operazioni successive

Verificare che l'attività sia stata completata correttamente utilizzando i comandi DISPLAY. Se l'attività ha avuto esito positivo, l'output risultante è simile a quello mostrato nel seguente esempio.

Dal gestore code QM1, immettere il comando seguente:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

L'output risultante è simile al seguente esempio:

```
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)
```

È facoltativo impostare l'attributo filtro SSLPEER delle definizioni di canale. Se la definizione di canale SSLPEER è impostata, il suo valore deve corrispondere al DN soggetto nel certificato partner creato nel passo 2. Dopo una connessione riuscita, il campo SSLPEER nell'output DISPLAY CHSTATUS mostra il DN dell'oggetto del certificato client remoto.

Utilizzo di certificati firmati CA per l'autenticazione reciproca di un client e di un gestore code

Seguire queste istruzioni di esempio per implementare l'autenticazione reciproca tra un client e un gestore code, utilizzando i certificati SSL o TLS firmati dalla CA.

Informazioni su questa attività

Scenario:

- Si dispone di un client C1e di un gestore code, QM1, che devono comunicare in modo sicuro. È necessaria l'autenticazione reciproca tra C1 e QM1.
- In futuro si prevede di utilizzare questa rete in un ambiente di produzione e quindi si è deciso di utilizzare i certificati firmati dalla CA fin dall'inizio.

La configurazione risultante è simile alla seguente:

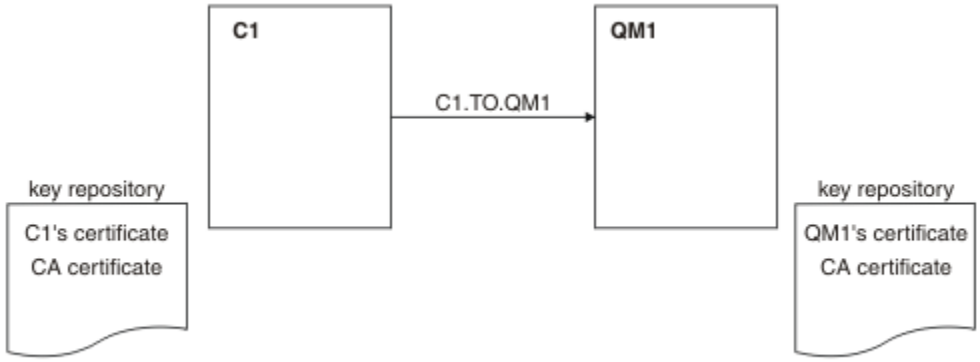


Figura 18. Configurazione risultante da questa attività

In Figura 18 a pagina 215, il repository delle chiavi per C1 contiene il certificato per C1 e il certificato CA. Il repository delle chiavi per QM1 contiene il certificato per QM1 e il certificato CA. In questo esempio, sia il certificato di C1che quello di QM1sono stati emessi dalla stessa CA. Se il certificato di C1e il certificato di QM1sono stati emessi da diverse CA, i repository delle chiavi per C1 e QM1 devono contenere entrambi i certificati CA.

Procedura

1. Preparare il repository chiavi sul client e sul gestore code, in base al sistema operativo:
   - Su sistemi UNIX, Linuxe Windows.
2. Richiedere un certificato firmato CA per il client e il gestore code.

   È possibile utilizzare diverse CA per il client e il gestore code.
   - Su sistemi UNIX, Linuxe Windows.
3. Aggiungere il certificato dell'autorità di certificazione al repository delle chiavi per client e gestore code.

   Se il client e il gestore code utilizzano autorità di certificazione differenti, il certificato CA per ogni autorità di certificazione deve essere aggiunto a entrambi i repository delle chiavi.
   - Su sistemi UNIX, Linuxe Windows.

4. Aggiungere il certificato firmato CA al repository delle chiavi per il client e il gestore code:
   - Su sistemi UNIX, Linuxe Windows.
5. Definire un canale di connessione client in uno dei seguenti modi:
   - Utilizzando la chiamata MQCONNX con la struttura MQSCO su C1, come descritto in Creazione di un canale di connessione client sul client WebSphere MQ MQI.
   - Utilizzando una tabella di definizione di canale client, come descritto in Creazione di definizioni di connessione server e di connessione client nel server .
6. In QM1, definire un canale di connessione server immettendo un comando simile al seguente esempio:

   ```
   DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(RC2_MD5_EXPORT)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from C1 to QM1')
   ```

   Il canale deve avere lo stesso nome del canale di collegamento client definito nel passo 6 e utilizzare lo stesso CipherSpec.

Risultati

I repository e i canali chiave vengono creati come illustrato in Figura 18 a pagina 215.

Operazioni successive

Verificare che l'attività sia stata completata correttamente utilizzando i comandi DISPLAY. Se l'attività ha avuto esito positivo, l'output risultante è simile a quello mostrato nel seguente esempio.

Dal gestore code QM1, immettere il comando seguente:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
```

L'output risultante è simile al seguente esempio:

```
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)
```

Il campo SSLPEER nell'output DISPLAY CHSTATUS mostra il DN oggetto del certificato client remoto creato nel passo 2. Il nome dell'emittente corrisponde al DN oggetto del certificato CA che ha firmato il certificato personale aggiunto nel Passo 4.

Connessione anonima di un client a un gestore code

Seguire queste istruzioni di esempio per modificare un sistema con autenticazione reciproca per consentire a un gestore code di connettersi in modo anonimo a un'altro.

Informazioni su questa attività

Scenario:

- Il gestore code e il client (QM1 e C1) sono stati impostati come in “Utilizzo di certificati firmati CA per l'autenticazione reciproca di un client e di un gestore code” a pagina 215.
- Si desidera modificare C1 in modo che si colleghi in modo anonimo a QM1.

La configurazione risultante è simile alla seguente:

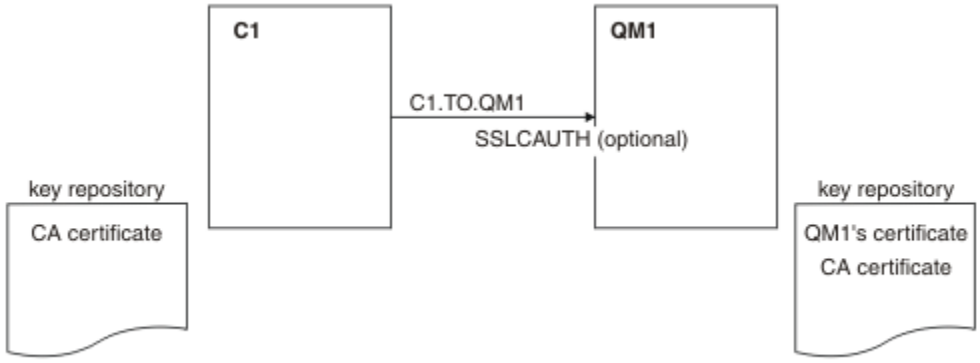


Figura 19. Client e gestore code che consentono la connessione anonima

Procedura

1. Rimuovere il certificato personale dal repository chiavi per C1, in base al sistema operativo:
   - Su sistemi UNIX, Linuxe Windows. Il certificato è etichettato come segue:
     - `ibmwebspheremq` seguito dall'ID utente di collegamento ripiegato in minuscolo, ad esempio `ibmwebspheremqmyuserid`.
2. Riavviare l'applicazione client o far sì che l'applicazione client chiuda e riapra tutte le connessioni SSL o TLS.
3. Consentire le connessioni anonime sul gestore code, immettendo il comando seguente:

```
ALTER CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) SSLCAUTH(OPTIONAL)
```

Risultati

I repository chiave e i canali vengono modificati come illustrato in Figura 19 a pagina 217

Operazioni successive

All'estremità del server del canale, la presenza del valore del parametro del nome peer sul pannello di stato del canale indica che è stato trasmesso un certificato client.

Verificare che l'attività sia stata completata correttamente immettendo alcuni comandi DISPLAY. Se l'attività ha avuto esito positivo, l'output risultante è simile a quello mostrato nel seguente esempio:

Dal gestore code QM1, immettere il comando seguente:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

L'output risultante sarà simile al seguente esempio:

```
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)
```

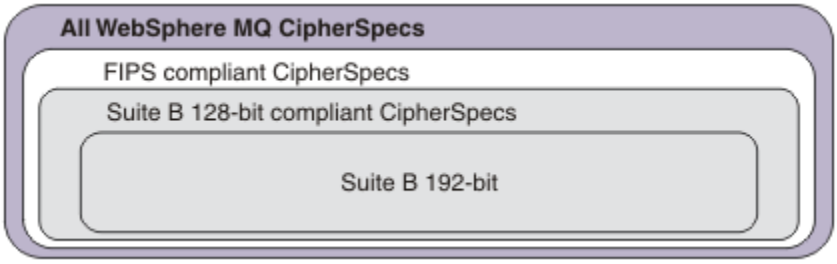
I campi SSLCERTI e SSLPEER sono vuoti e indicano che C1 non ha inviato un certificato.

Specifica di CipherSpecs

Specificare una CipherSpec utilizzando il parametro **SSLCIPH** nel comando MQSC **DEFINE CHANNEL** o nel comando MQSC **ALTER CHANNEL** .

Alcuni dei CipherSpecs che possono essere utilizzati con IBM WebSphere MQ sono compatibili con FIPS. Altri, come NULL_MD5, non lo sono. Allo stesso modo, alcuni CipherSpecs compatibili con FIPS sono compatibili con Suite B anche se altri non lo sono. Tutti i CipherSpecs compatibili con Suite B sono compatibili con FIPS. Tutti i CipherSpecs compatibili con la Suite B rientrano in due gruppi: 128 bit (ad esempio, ECDHE_ECDSA_AES_128_GCM_SHA256) e 192 bit (ad esempio ECDHE_ECDSA_AES_256_GCM_SHA384),

Il seguente diagramma illustra la relazione tra questi sottoinsiemi:



Le specifiche di cifratura che puoi utilizzare con il supporto IBM WebSphere MQ SSL e TLS sono elencate nella seguente tabella. Quando si richiede un certificato personale, si specifica una dimensione di chiave per la coppia di chiavi pubblica e privata. La dimensione della chiave utilizzata durante l'handshake SSL è la dimensione memorizzata nel certificato a meno che non sia determinata da CipherSpec, come indicato nella tabella.

Nome CipherSpec	Protocollo utilizzato	Algoritmo MAC	Algoritmo di codifica	Bit di codifica	FIPS[1]	Suite B a 128 bit	Suite B 192 bit
NULL_MD5 [a]	SSL 3.0	MD5	Nessuna	0	No	No	No
NULL_SHA [a]	SSL 3.0	SHA-1	Nessuna	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	Sì	No	No
TLS_RSA_WITH_AES_256_CBC_SHA [4] [a]	TLS 1.0	SHA-1	AES	256	Sì	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	Sì	No	No

Nome CipherSpec	Protocollo utilizzato	Algoritmo MAC	Algoritmo di codifica	Bit di codifica	FIPS[1]	Suite B a 128 bit	Suite B 192 bit
TLS_RSA_WITH_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sì	No	No
TLS_RSA_WITH_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sì	No	No
TLS_RSA_WITH_AES_256_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	256	Sì	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	Sì	No	No
ECDHE_ECDSA_AES_256_CBC_SHA384 [b]	TLS 1.2	SHA-384	AES	256	Sì	No	No
ECDHE_RSA_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sì	No	No
ECDHE_RSA_AES_256_CBC_SHA384 [b]	TLS 1.2	SHA-384	AES	256	Sì	No	No
ECDHE_ECDSA_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sì	Sì	No
ECDHE_ECDSA_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sì	No	Sì
ECDHE_RSA_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sì	No	No
ECDHE_RSA_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sì	No	No
TLS_RSA_WITH_NULL_SHA256 [b]	TLS 1.2	SHA-256	Nessuna	0	No	No	No
ECDHE_RSA_NULL_SHA256 [b]	TLS 1.2	SHA-1	Nessuna	0	No	No	No
ECDHE_ECDSA_NULL_SHA256 [b]	TLS 1.2	SHA-1	Nessuna	0	No	No	No
TLS_RSA_WITH_NULL_NULL [b]	TLS 1.2	Nessuna	Nessuna	0	No	No	No
TLS_RSA_WITH_RC4_128_SHA256 [b]	TLS 1.2	SHA-1	RC4	128	No	No	No

Nome CipherSpec	Protocollo utilizzato	Algoritmo MAC	Algoritmo di codifica	Bit di codifica	FIPS[1]	Suite B a 128 bit	Suite B 192 bit

Note:

1. Specifica se la CipherSpec è certificata da FIPS su una piattaforma certificata FIPS. Consultare Federal Information Processing Standards (FIPS) per una spiegazione di FIPS.
2. La dimensione massima della chiave di handshake è 512 bit. Se uno dei certificati scambiati durante l'handshake SSL ha una dimensione di chiave maggiore di 512 bit, viene creata una chiave temporanea di 512 bit da utilizzare durante l'handshake.
3. La dimensione della chiave di handshake è 1024 bit.
4. Questa CipherSpec non può essere utilizzata per proteggere una connessione da WebSphere MQ Explorer a un gestore code a meno che non vengano applicati i file delle politiche senza limitazioni appropriati al JRE utilizzato da Explorer.
5. Questa CipherSpec era certificata FIPS 140-2 prima del 19 maggio 2007.
6. Questa CipherSpec era certificata FIPS 140-2 prima del 19 maggio 2007. Il nome `FIPS_WITH_DES_CBC_SHA` è storico e riflette il fatto che questa CipherSpec era in precedenza (ma non è più) compatibile con FIPS. Questa CipherSpec è obsoleta e non se ne consiglia l'utilizzo.
7. Questa CipherSpec può essere utilizzata per trasferire fino a 32 GB di dati prima che la connessione venga terminata con l'errore AMQ9288. Per evitare questo errore, non utilizzare il triplo DES o abilitare la reimpostazione della chiave segreta quando si utilizza questa CipherSpec.

Supporto della piattaforma:

- a Disponibile su tutte le piattaforme supportate.
- b Disponibile solo su piattaforme UNIX, Linux, and Windows .

Concetti correlati

"Certificati digitali e compatibilità CipherSpec in IBM WebSphere MQ" a pagina 34
Questo argomento fornisce informazioni su come scegliere i CipherSpecs e i certificati digitali appropriati per la politica di sicurezza, evidenziando la relazione tra CipherSpecs e i certificati digitali in IBM WebSphere MQ.

Riferimenti correlati

Definire il canale
MODIFICA CANALE

CipherSpecs obsoleto

Un elenco di CipherSpecs obsoleti che è possibile utilizzare con WebSphere MQ , se necessario.

Consultare "Valori CipherSpec supportati in IBM WebSphere MQ" a pagina 39 per ulteriori informazioni su come abilitare i CipherSpecsobsoleti.

I CipherSpecs obsoleti che possono essere utilizzati con WebSphere MQ TLS support sono elencati nella tabella seguente:

Supporto piattaforma "1" a pagina 222	Nome CipherSpec	Protocollo utilizzato	Integrità dati	Algoritmo di codifica	Bit di codifica	FIPS "2" a pagina 222	Suite B	Aggiorna quando obsoleto
Tutto	DES_SHA_EXPORT"3" a pagina 222	SSL 3.0	SHA-1	DES	56	No	No	7.5.0.6

Supporto piattaforma "1" a pagina 222	Nome CipherSpec	Protocollo utilizzato	Integrità dati	Algoritmo di codifica	Bit di codifica	FIPS "2" a pagina 222	Suite B	Aggiorna quando obsoleto
UNIX Linux Windows	DES_SHA_EXPORT1024 "4" a pagina 222	SSL 3.0	SHA-1	DES	56	No	No	7.5.0.6
UNIX Linux Windows	FIPS_WITH_DES_CBC_SHA	SSL 3.0	SHA-1	DES	56	Nessun "6" a pagina 222	No	7.5.0.6
UNIX Linux Windows	FIPS_WITH_3DES_EDE_CBC_SHA	SSL 3.0	SHA-1	3DES	168	Nessun "7" a pagina 222	No	7.5.0.8
Tutto	NULL_MD5	SSL 3.0	MD5	Nessuno	0	No	No	7.5.0.6
Tutto	NULL_SHA	SSL 3.0	SHA-1	Nessuno	0	No	No	7.5.0.6
Tutto	RC2_MD5_EXPORT "3" a pagina 222	SSL 3.0	MD5	RC2	40	No	No	7.5.0.7
Tutto	RC4_MD5_EXPORT "3" a pagina 222	SSL 3.0	MD5	RC4	40	No	No	7.5.0.7
Tutto	RC4_MD5_US	SSL 3.0	MD5	RC4	128	No	No	7.5.0.7
Tutto	RC4_SHA_US	SSL 3.0	SHA-1	RC4	128	No	No	7.5.0.7
UNIX Linux Windows	RC4_56_SHA_EXPORT1024 "4" a pagina 222	SSL 3.0	SHA-1	RC4	56	No	No	7.5.0.7
Tutto	TRIPLE_DES_SHA_US	SSL 3.0	SHA-1	3DES	168	No	No	7.5.0.8
Tutto	TLS_RSA_WITH_DES_CBC_SHA	TLS 1.0	SHA-1	DES	56	Nessun "5" a pagina 222	No	7.5.0.6
UNIX Linux Windows	ECDHE_ECDSA_NULL_SHA256	TLS 1.2	SHA-1	Nessuno	0	No	No	7.5.0.6
UNIX Linux Windows	ECDHE_ECDSA_RC4_128_SHA256	TLS 1.2	SHA-1	RC4	128	No	No	7.5.0.7
UNIX Linux Windows	ECDHE_RSA_NULL_SHA256	TLS 1.2	SHA-1	Nessuno	0	No	No	7.5.0.6

Supporto piattaforma "1" a pagina 222	Nome CipherSpec	Protocollo utilizzato	Integrità dati	Algoritmo di codifica	Bit di codifica	FIPS "2" a pagina 222	Suite B	Aggiorna quando obsoleto
UNIX Linux Windows	ECDHE_RSA_RC4_128_SHA256	TLS 1.2	SHA-1	RC4	128	No	No	7.5.0.7
UNIX Linux Windows	TLS_RSA_WITH_NULL_NULL	TLS 1.2	Nessuno	Nessuno	0	No	No	7.5.0.6
Tutto	TLS_RSA_WITH_NULL_SHA256	TLS 1.2	SHA-256	Nessuno	0	No	No	7.5.0.6
UNIX Linux Windows	TLS_RSA_WITH_RC4_128_SHA256	TLS 1.2	SHA-1	RC4	128	No	No	7.5.0.7
Tutto	TLS_RSA_WITH_3DES_EDE_CBC_SHA"8" a pagina 222	TLS 1.0	SHA-1	3DES	168	Sì	No	7.5.0.8
UNIX Linux Windows	ECDHE_ECDSA_3DES_EDE_CBC_SHA256"8" a pagina 222	TLS 1.2	SHA-1	3DES	168	Sì	No	7.5.0.8
UNIX Linux Windows	ECDHE_RSA_3DES_EDE_CBC_SHA256"8" a pagina 222	TLS 1.2	SHA-1	3DES	168	Sì	No	7.5.0.8

Note:

1. Se non si nota una piattaforme specifica, il CipherSpec è disponibile su tutte le piattaforme.
2. Specifica se la CipherSpec è certificata da FIPS su una piattaforma certificata FIPS. Consultare Federal Information Processing Standards (FIPS) per una spiegazione di FIPS.
3. La dimensione massima della chiave di handshake è 512 bit. Se uno dei certificati scambiati durante l'handshake SSL ha una dimensione di chiave maggiore di 512 bit, viene creata una chiave temporanea di 512 bit da utilizzare durante l'handshake.
4. La dimensione della chiave di handshake è 1024 bit.
5. Questa CipherSpec era certificata FIPS 140-2 prima del 19 maggio 2007.
6. Questa CipherSpec era certificata FIPS 140-2 prima del 19 maggio 2007. Il nome FIPS_WITH_DES_CBC_SHA è storico e riflette il fatto che in precedenza questa CipherSpec fosse conforme a FIPS (ma non più). Questa CipherSpec è obsoleta e non se ne consiglia l'utilizzo.
7. Il nome FIPS_WITH_3DES_EDE_CBC_SHA è storico e riflette il fatto che in precedenza questa CipherSpec fosse conforme a FIPS (ma non più). L'utilizzo di questa CipherSpec è obsoleto.
8. Questa CipherSpec può essere utilizzata per trasferire fino a 32 GB di dati prima che la connessione venga terminata con l'errore AMQ9288. Per evitare questo errore, non utilizzare il triplo DES o abilitare la reimpostazione della chiave segreta quando si utilizza questa CipherSpec.

Acquisizione di informazioni su CipherSpecs utilizzando IBM WebSphere MQ Explorer

È possibile utilizzare IBM WebSphere MQ Explorer per visualizzare le descrizioni di CipherSpecs.

Utilizzare la seguente procedura per ottenere informazioni su CipherSpecs in “Specifica di CipherSpecs” a pagina 218:

1. Aprire **IBM WebSphere MQ Explorer** ed espandere la cartella **Gestori code** .
2. Assicurarsi di aver avviato il gestore code.
3. Selezionare il gestore code che si desidera utilizzare e fare clic su **Canali**.
4. Fare clic con il pulsante destro del mouse sul canale che si desidera utilizzare e selezionare **Proprietà**.
5. Selezionare la pagina delle proprietà **SSL** .
6. Selezionare dall'elenco la CipherSpec che si desidera utilizzare. Una descrizione viene visualizzata nella finestra sotto l'elenco.

Alternative per specificare CipherSpecs

Per le piattaforme in cui il sistema operativo fornisce il supporto SSL, il sistema potrebbe supportare i nuovi CipherSpecs. È possibile specificare un nuovo CipherSpec con il parametro SSLCIPH, ma il valore fornito dipende dalla piattaforma.

Nota: Questa sezione non si applica ai sistemi UNIX, Linux o Windows , perché i CipherSpecs vengono forniti con il prodotto WebSphere MQ , quindi i nuovi CipherSpecs non diventano disponibili dopo la spedizione.

Per le piattaforme in cui il sistema operativo fornisce il supporto SSL, il proprio sistema potrebbe supportare nuovi CipherSpecs non inclusi in “Specifica di CipherSpecs” a pagina 218. È possibile specificare un nuovo CipherSpec con il parametro SSLCIPH, ma il valore fornito dipende dalla piattaforma. In tutti i casi, la specifica *deve* corrispondere a un CipherSpec SSL valido e supportato dalla versione di SSL in esecuzione sul sistema.

IBM i

Una stringa di due caratteri che rappresenta un valore esadecimale.

Per ulteriori informazioni sui valori consentiti, fare riferimento alla documentazione del prodotto appropriata (ricercare *cipher_spec* nella documentazione del prodottoIBM i).

È possibile utilizzare il comando CHGMQMCHL o CRTMQMCHL per specificare il valore, ad esempio:

```
CRTMQMCHL CHLNAME('channel name') SSLCIPH('hexadecimal value')
```

È anche possibile utilizzare il comando ALTER QMGR MQSC per impostare il parametro SSLCIPH .

z/OS

Una stringa di due caratteri che rappresenta un valore esadecimale. I codici esadecimali corrispondono ai valori definiti nel protocollo SSL.

Per ulteriori informazioni, fare riferimento alla descrizione di gsk_environment_open () nel capitolo di riferimento API di *z/OS Cryptographic Services System SSL Programming*, SC24-5901, dove è presente un elenco di tutte le specifiche di cifratura SSL V3.0 e TLS V1.0 supportate sotto forma di codici esadecimali a due cifre.

Considerazioni sui cluster WebSphere MQ

Con i cluster WebSphere MQ è più sicuro utilizzare i nomi CipherSpec in “Specifica di CipherSpecs” a pagina 218. Se si utilizza una specifica alternativa, tenere presente che la specifica potrebbe non essere valida su altre piattaforme. Per ulteriori informazioni, fare riferimento a “SSL e cluster” a pagina 249.

Specifica di un CipherSpec per un client IBM WebSphere MQ MQI

Sono disponibili tre opzioni per specificare un CipherSpec per un client MQI IBM WebSphere MQ .

Le opzioni disponibili sono:

- Utilizzo di una tabella di definizione di canale
- Utilizzo del campo SSLCipherSpec nella struttura MQCD, in MQCD_VERSION_7 o superiore, su una chiamata MQCONNX.
- Utilizzo di Active Directory (su sistemi Windows con supporto Active Directory)

Specifica di una CipherSuite con classi IBM WebSphere MQ per Java e classi IBM WebSphere MQ per JMS

Le classi IBM WebSphere MQ per Java e IBM WebSphere MQ per JMS specificano CipherSuites in maniera diversa dalle altre piattaforme.

Per informazioni su come specificare una CipherSuite con classi IBM WebSphere MQ per Java, consultare Supporto SSL (Secure Sockets Layer).

Per informazioni su come specificare una CipherSuite con le classi IBM WebSphere MQ per JMS, consultare Utilizzo di SSL (Secure Sockets Layer) con WebSphere MQ classes for JMS.

Reimpostazione delle chiavi segrete SSL e TLS

IBM WebSphere MQ supporta la reimpostazione delle chiavi segrete su gestori code e client.

Le chiavi segrete vengono reimpostate quando un numero specificato di byte crittografati di dati è stato trasmesso attraverso il canale o dopo che il canale è stato inattivo per un periodo di tempo.

Il valore di reimpostazione della chiave viene sempre impostato dal lato di avvio del canale MQ .

Gestore code

Per un gestore code, utilizzare il comando **ALTER QMGR** con parametro **SSLRKEYC** per impostare i valori utilizzati durante la rinegoziazione della chiave.

Client MQI

Per impostazione predefinita, i client MQI non rinegoziano la chiave segreta. È possibile fare in modo che un client MQI rinegozii la chiave in uno dei tre modi. Nel seguente elenco, i metodi vengono mostrati in ordine di priorità. Se si specificano più valori, viene utilizzato il valore di priorità più alto.

1. Utilizzando il campo KeyResetCount nella struttura MQSCO su una chiamata MQCONNX
2. Utilizzando la variabile di ambiente MQSSLRESET
3. Impostando l'attributo SSLKeyResetCount nel file di configurazione client MQI

Queste variabili possono essere impostate su un numero intero compreso tra 0 e 999 999 999, che rappresenta il numero di byte non crittografati inviati e ricevuti all'interno di una conversazione SSL o TLS prima che la chiave segreta SSL o TLS venga rinegoziata. Specificando il valore 0 si indica che le chiavi segrete SSL o TLS non vengono mai rinegoziate. Se si specifica un conteggio di reimpostazione della chiave segreta SSL o TLS compreso tra 1 byte e 32 KB, i canali SSL o TLS utilizzeranno un conteggio di reimpostazione della chiave segreta di 32 KB. Ciò per evitare un numero eccessivo di reimpostazioni della chiave che si verificherebbe per valori di reimpostazione della chiave segreta SSL o TLS di piccole dimensioni.

Se viene specificato un valore maggiore di zero e gli heartbeat del canale sono abilitati per il canale, anche la chiave segreta viene rinegoziata prima che i dati del messaggio vengano inviati o ricevuti dopo un heartbeat del canale.

Il numero di byte fino alla successiva rinegoziazione della chiave segreta viene reimpostato dopo ogni rinegoziazione riuscita.

Per i dettagli completi della struttura MQSCO, consultare KeyResetCount (MQLONG). Per i dettagli completi di MQSSLRESET, vedere MQSSLRESET. Per ulteriori informazioni sull'utilizzo di SSL o TLS nel file di configurazione client, consultare Stanza SSL del file di configurazione client.

Java

Per IBM WebSphere MQ classes for Java, un'applicazione può reimpostare la chiave segreta in uno dei modi seguenti:

- Impostando il campo sslResetCount nella classe MQEnvironment.
- Impostando la proprietà di ambiente MQC.SSL_RESET_COUNT_PROPERTY in un oggetto Hashtable. L'applicazione, quindi, assegna l'hashtable al campo `properties` nella classe MQEnvironment o passa l'hashtable a un oggetto MQQueueManager sul relativo costruttore.

Se l'applicazione utilizza più di uno di questi modi, si applicano le solite regole di precedenza. Consultare Classe com.ibm.mq.MQEnvironment per le regole di precedenza.

Il valore del campo sslReseto della proprietà di ambiente MQC.SSL_RESET_COUNT_PROPERTY rappresenta il numero totale di byte inviati e ricevuti dalle classi WebSphere MQ per il codice client Java prima che la chiave segreta venga rinegoziata. Il numero di byte inviati è il numero prima della codifica e il numero di byte ricevuti è il numero dopo la decodifica. Il numero di byte include anche le informazioni di controllo inviate e ricevute dal client WebSphere MQ .

Se il conteggio di reimpostazione è zero, che è il valore predefinito, la chiave segreta non viene mai rinegoziata. Il conteggio di reimpostazioni viene ignorato se non viene specificato alcun CipherSuite .

JMS

Per IBM WebSphere MQ classes for JMS, la proprietà SSLRESETCOUNT rappresenta il numero totale di byte inviati e ricevuti da una connessione prima che la chiave segreta utilizzata per la codifica venga rinegoziata. Il numero di byte inviati è il numero prima della codifica e il numero di byte ricevuti è il numero dopo la decodifica. Il numero di byte include anche le informazioni di controllo inviate e ricevute da IBM WebSphere MQ classes for JMS. Ad esempio, per configurare un oggetto ConnectionFactory che può essere utilizzato per creare una connessione su un canale MQI abilitato SSL o TLS con una chiave segreta che viene rinegoziata dopo il flusso di 4 MB di dati, immettere il seguente comando per JMSAdmin:

```
ALTER CF(my.cf) SSLRESETCOUNT(4194304)
```

Se il valore di SSLRESETCOUNT è zero, che è il valore predefinito, la chiave segreta non viene mai rinegoziata. La proprietà SSLRESETCOUNT viene ignorata se SSLCIPHERSUITE non è impostata.

.NET

Per i client non gestiti .NET, la proprietà integer SSLKeyResetCount indica il numero di byte non codificati inviati e ricevuti all'interno di una conversazione SSL o TLS prima che la chiave segreta venga rinegoziata.

Per informazioni sull'utilizzo delle proprietà oggetto nelle classi IBM WebSphere MQ per .NET, consultare Acquisizione e impostazione dei valori degli attributi.

XMS .NET

Per i client XMS .NET non gestiti, fare riferimento a Connessioni sicure a un gestore code IBM WebSphere MQ.

Riferimenti correlati
Gestore code ALTER
VISUALIZZAZIONE QMGR

Implementazione della riservatezza nei programmi di uscita utente

Implementazione della riservatezza nelle uscite di sicurezza

Le uscite di sicurezza possono svolgere un ruolo nel servizio di riservatezza generando e distribuendo la chiave simmetrica per codificare e decodificare i dati che fluiscono sul canale. Una tecnica comune per fare questo utilizza la tecnologia PKI.

Un'uscita di sicurezza genera un valore di dati casuale, lo crittografa con la chiave pubblica del gestore code o dell'utente rappresentato dall'uscita di sicurezza del partner e invia i dati crittografati al relativo partner in un messaggio di sicurezza. L'uscita di sicurezza partner decodifica il valore dei dati casuali con la chiave privata del gestore code o dell'utente che rappresenta. Ogni uscita di sicurezza può ora utilizzare il valore dei dati casuali per derivare la chiave simmetrica indipendentemente dall'altro utilizzando un algoritmo noto a entrambi. In alternativa, possono utilizzare il valore dei dati casuali come chiave.

Se la prima uscita di sicurezza non ha autenticato il proprio partner in questo momento, il messaggio di sicurezza successivo inviato dal partner può contenere un valore previsto codificato con la chiave simmetrica. La prima uscita di sicurezza può ora autenticare il proprio partner controllando che l'uscita di sicurezza del partner sia stata in grado di codificare correttamente il valore previsto.

Le uscite di sicurezza possono anche utilizzare questa opportunità per concordare l'algoritmo per crittografare e decrittografare i dati che fluiscono sul canale, se più di un algoritmo è disponibile per l'uso.

Implementazione della riservatezza nelle uscite dei messaggi

Un'uscita messaggio all'estremità di invio di un canale può codificare i dati dell'applicazione in un messaggio e un'altra uscita messaggio all'estremità di ricezione del canale può decodificare i dati. Per motivi di prestazioni, un algoritmo di chiave simmetrica viene normalmente utilizzato per questo scopo. Per ulteriori informazioni su come generare e distribuire la chiave simmetrica, consultare “Implementazione della riservatezza nei programmi di uscita utente” a pagina 225.

Le intestazioni in un messaggio, come l'intestazione della coda di trasmissione, MQXQH, che include il descrittore del messaggio incorporato, non devono essere codificate da un'uscita messaggio. Ciò è dovuto al fatto che la conversione dei dati delle intestazioni del messaggio avviene dopo che un'uscita del messaggio viene richiamata all'estremità di invio o prima che un'uscita del messaggio venga richiamata all'estremità di ricezione. Se le intestazioni sono codificate, la conversione dei dati ha esito negativo e il canale si arresta.

Implementazione della riservatezza nelle uscite di invio e ricezione

Le uscite di invio e ricezione possono essere utilizzate per crittografare e decrittografare i dati che fluiscono su un canale. Sono più appropriati delle uscite dei messaggi per fornire questo servizio per i seguenti motivi:

- Su un canale di messaggi, è possibile codificare le intestazioni dei messaggi e i dati dell'applicazione nei messaggi.
- Le uscite di invio e ricezione possono essere utilizzate sui canali MQI e sui canali di messaggi. I parametri sulle chiamate MQI potrebbero contenere dati sensibili dell'applicazione che devono essere protetti durante il flusso su un canale MQI. È quindi possibile utilizzare le stesse uscite di invio e ricezione su entrambi i tipi di canali.

Implementazione della riservatezza nell'uscita API e nell'uscita incrociata API

I dati dell'applicazione in un messaggio possono essere codificati da un'API o da un'uscita incrociata API quando il messaggio viene inserito dall'applicazione mittente e decodificato da una seconda uscita quando il messaggio viene richiamato dall'applicazione ricevente. Per motivi di prestazioni, un algoritmo di chiave simmetrica viene generalmente utilizzato per questo scopo. Tuttavia, a livello dell'applicazione, in cui molti utenti potrebbero inviarsi messaggi l'uno all'altro, il problema è come garantire che solo il destinatario previsto di un messaggio sia in grado di decodificare il messaggio. Una soluzione consiste nell'utilizzare una diversa chiave simmetrica per ogni coppia di utenti che si inviano messaggi. Ma questa soluzione potrebbe essere difficile e dispendiosa in termini di tempo da amministrare, in particolare se

gli utenti appartengono a diverse organizzazioni. Un modo standard per risolvere questo problema è noto come *digital enveloping* e utilizza la tecnologia PKI.

Quando un'applicazione inserisce un messaggio in una coda, un'API o un'uscita API - crossing genera una chiave simmetrica casuale e utilizza la chiave per codificare i dati dell'applicazione nel messaggio. L'uscita codifica la chiave simmetrica con la chiave pubblica del destinatario previsto. Sostituisce quindi i dati dell'applicazione nel messaggio con i dati dell'applicazione codificati e la chiave simmetrica codificata. In questo modo, solo il destinatario previsto può decodificare la chiave simmetrica e quindi i dati dell'applicazione. Se un messaggio codificato ha più di un possibile destinatario previsto, l'uscita può codificare una copia della chiave simmetrica per ogni destinatario previsto.

Se sono disponibili diversi algoritmi per la codifica e la decodifica dei dati dell'applicazione, l'uscita può includere il nome dell'algoritmo utilizzato.

Integrità dei dati dei messaggi

Per mantenere l'integrità dei dati, è possibile utilizzare vari tipi di programmi di uscita utente per fornire digest di messaggi o firme digitali per i propri messaggi.

Integrità dati

Implementazione dell'integrità dei dati nei messaggi

Quando si utilizza SSL o TLS, la scelta di CipherSpec determina il livello di integrità dei dati nell'azienda. Se si utilizza il servizio WebSphere MQ Advanced Message Service (AMS), è possibile specificare l'integrità per un messaggio univoco.

Implementazione dell'integrità dei dati nelle uscite dei messaggi

Un messaggio può essere firmato digitalmente da un'uscita di messaggio all'estremità di invio di un canale. La firma digitale può quindi essere controllata da un'uscita del messaggio all'estremità ricevente di un canale per rilevare se il messaggio è stato deliberatamente modificato.

Alcune protezioni possono essere fornite utilizzando un digest del messaggio invece di una firma digitale. Un digest del messaggio potrebbe essere efficace contro la manomissione casuale o indiscriminata, ma non impedisce all'individuo più informato di modificare o sostituire il messaggio e generare un digest completamente nuovo per esso. Ciò è particolarmente vero se l'algoritmo utilizzato per generare il digest del messaggio è ben noto.

Implementazione dell'integrità dei dati nelle uscite di invio e ricezione

Su un canale di messaggi, le uscite di messaggi sono più appropriate per fornire questo servizio poiché un'uscita di messaggi ha accesso a un intero messaggio. Su un canale MQI, i parametri sulle chiamate MQI potrebbero contenere i dati dell'applicazione che devono essere protetti e solo le uscite di invio e ricezione possono fornire questa protezione.

Implementazione dell'integrità dei dati nell'uscita API o nell'uscita incrociata API

Un messaggio può essere firmato digitalmente da un'API o da un'uscita incrociata API quando il messaggio viene inserito dall'applicazione mittente. La firma digitale può quindi essere controllata da una seconda uscita quando il messaggio viene richiamato dall'applicazione ricevente per rilevare se il messaggio è stato deliberatamente modificato.

Alcune protezioni possono essere fornite utilizzando un digest del messaggio invece di una firma digitale. Un digest del messaggio potrebbe essere efficace contro la manomissione casuale o indiscriminata, ma non impedisce all'individuo più informato di modificare o sostituire il messaggio e generare un digest completamente nuovo per esso. Ciò è particolarmente vero se l'algoritmo utilizzato per generare il digest del messaggio è ben noto,

Connessione di due gestori code mediante SSL o TLS

Le comunicazioni sicure che utilizzano i protocolli di sicurezza crittografica SSL o TLS richiedono l'impostazione dei canali di comunicazione e la gestione dei certificati digitali che verranno utilizzati per l'autenticazione.

Per configurare l'installazione SSL o TLS è necessario definire i canali per utilizzare SSL o TLS. È inoltre necessario ottenere e gestire i certificati digitali. Su un sistema di test, è possibile utilizzare certificati autofirmati o certificati emessi da un'autorità di certificazione (CA) locale. Su un sistema di produzione, non utilizzare certificati autofirmati. Per ulteriori informazioni, consultare ../zs14140_.dita.

Per informazioni complete sulla creazione e la gestione dei certificati, consultare “Utilizzo di SSL o TLS su sistemi UNIX, Linux, and Windows” a pagina 114.

Questa raccolta di argomenti introduce le attività relative all'impostazione delle comunicazioni SSL e fornisce una guida dettagliata sul completamento di tali attività.

Si potrebbe anche voler verificare l'autenticazione client SSL o TLS, che sono una parte facoltativa dei protocolli. Durante l'handshake SSL o TLS, il client SSL o TLS ottiene e convalida sempre un certificato digitale dal server. Con l'implementazione WebSphere MQ , il server SSL o TLS richiede sempre un certificato dal client.

Note:

1. In questo contesto, un client SSL fa riferimento alla connessione che avvia l'handshake.
2. Consultare il Glossario per ulteriori dettagli.

Su sistemi UNIX, Linux e Windows , il client SSL o TLS invia un certificato solo se ha un'etichetta nel formato WebSphere MQ corretto, `ibmwebspheremq` seguito dal nome del gestore code modificato in minuscolo. Ad esempio, per `QM1`, `ibmwebspheremqqm1`.

WebSphere MQ utilizza il prefisso `ibmwebspheremq` su un'etichetta per evitare confusione con i certificati per altri prodotti. Assicurarsi di specificare l'intera etichetta del certificato in minuscolo.

Il server SSL o TLS convalida sempre il certificato client, se ne viene inviato uno. Se il client non invia un certificato, l'autenticazione non riesce solo se la fine del canale che agisce come server SSL o TLS è definita con il parametro SSLCAUTH impostato su REQUIRED o con un valore del parametro SSLPEER impostato. Per ulteriori informazioni sulla connessione anonima di un gestore code, ossia quando il client SSL o TLS non invia un certificato, consultare “Connessione di due gestori code utilizzando l'autenticazione unidirezionale” a pagina 211.

Etichette dei certificati digitali, comprensione dei requisiti

Quando si impostano SSL e TLS per utilizzare i certificati digitali, potrebbero essere presenti requisiti di etichetta specifici che è necessario seguire, a seconda della piattaforma utilizzata e del metodo utilizzato per la connessione.

Informazioni su questa attività

Cos' è l'etichetta del certificato?

Un'etichetta certificato è un identificativo univoco che rappresenta un certificato digitale memorizzato in un repository di chiavi e fornisce un nome leggibile con cui fare riferimento a un particolare certificato quando si eseguono funzioni di gestione chiavi. L'etichetta del certificato viene assegnata quando si aggiunge un certificato a un repository di chiavi per la prima volta.

L'etichetta del certificato è separata dai campi *Subject Distinguished Name* o *Subject Common Name* del certificato. Tenere presente che *Subject Distinguished Name* e *Subject Common Name* sono campi all'interno del certificato stesso. Questi sono definiti quando viene creato il certificato e non possono essere modificati. Tuttavia, è possibile modificare l'etichetta associata a un certificato digitale, se necessario.

Come viene utilizzata l'etichetta del certificato?

IBM WebSphere MQ utilizza le etichette di certificato per individuare un certificato personale inviato durante l'handshake SSL. Ciò elimina l'ambiguità quando esiste più di un certificato personale nel repository delle chiavi.

Le etichette dei certificati seguono una convenzione di denominazione; è necessario assicurarsi di utilizzare la convenzione di denominazione delle etichette corretta corrispondente alla piattaforma utilizzata.

In questo contesto, un client SSL o TLS fa riferimento al partner di connessione che avvia l'handshake, che potrebbe essere un client IBM WebSphere MQ o un altro gestore code.

Durante l'handshake SSL o TLS, il client SSL o TLS ottiene e convalida sempre un certificato digitale dal server. Con l'implementazione IBM WebSphere MQ , il server SSL o TLS richiede sempre un certificato dal client e il client fornisce sempre un certificato al server se ne trova uno. Se il client non è in grado di individuare un certificato personale, invia una risposta `no certificate` al server.

Il server SSL o TLS convalida sempre il certificato client, se ne viene inviato uno. Se il client non invia un certificato, l'autenticazione non riesce se la fine del canale che agisce come server SSL o TLS è definita con il parametro SSLCAUTH impostato su REQUIRED o con un valore del parametro SSLPEER impostato.

Per ulteriori informazioni relative alla connessione di un gestore code mediante l'autenticazione unidirezionale, ossia quando il client SSL o TLS non invia un certificato, consultare “Connessione di due gestori code utilizzando l'autenticazione unidirezionale” a pagina 211.

, UNIX, Linux, and Windows sistemi

Informazioni su questa attività

Su sistemi , UNIX, Linux, and Windows , il server SSL o TLS invia un certificato al client, solo se il server ne trova uno etichettato nel formato IBM WebSphere MQ corretto. Su questi sistemi, il formato corretto è `ibmwebspheremq`, seguito dal nome del gestore code modificato in minuscolo.

Ad esempio, per un gestore code denominato `QM1`, il requisito dell'etichetta del certificato è:

```
ibmwebspheremqqm1
```

Se non viene trovato alcun certificato nel repository delle chiavi del gestore code, che corrisponde all'etichetta richiesta nel formato e nel maiuscolo / minuscolo corretti, si verifica un errore e l'handshake SSL o TLS ha esito negativo.

IBM WebSphere MQ client

Informazioni su questa attività

Quando ci si connette da un'applicazione client IBM WebSphere MQ , il client SSL o TLS invia un certificato solo se dispone di un certificato con un'etichetta nel formato `ibmwebspheremq`, seguito dal nome utente dell'utente che esegue il processo dell'applicazione client.

Ad esempio, per il nome utente `wasadmin`, il requisito dell'etichetta del certificato è come mostrato, ripiegata in minuscolo:

```
  ibmwebspheremqwasadmin
```

Il requisito di etichetta precedente si applica a Message Service Client per C, o C + +, e .NET.

Client IBM WebSphere MQ Java o IBM WebSphere MQ JMS

Informazioni su questa attività

I client IBM WebSphere MQ Java o IBM WebSphere MQ JMS utilizzano le funzioni del proprio provider JSSE (Java Secure Socket Extension) per selezionare un certificato personale durante l'handshake SSL o TLS e quindi non sono soggetti ai requisiti dell'etichetta del certificato.

Il comportamento predefinito è che il client JSSE esegue l'iterazione attraverso i certificati nel repository delle chiavi, selezionando il primo certificato personale accettabile trovato. Tuttavia, questo comportamento è solo un valore predefinito e dipende dall'implementazione del provider JSSE.

Inoltre, l'interfaccia JSSE è altamente personalizzabile tramite la configurazione e l'accesso diretto al runtime da parte dell'applicazione. Consultare la documentazione fornita dal fornitore JSSE per dettagli specifici.

Per la risoluzione dei problemi o per comprendere meglio l'handshake eseguito dall'applicazione client Java IBM WebSphere MQ in combinazione con il provider JSSE specifico, è possibile abilitare il debug impostando

```
javax.net.debug=ssl
```

nell'ambiente JVM.

Puoi utilizzare `-Djavax.net.debug=ssl` sulla riga di comando o impostare la variabile all'interno dell'applicazione o tramite la configurazione.

Concetti correlati

"Importazione di un certificato personale in un repository di chiavi su sistemi UNIX, Linux, and Windows" a pagina 134

Seguire questa procedura per importare un certificato personale

Utilizzo di certificati autofirmati per l'autenticazione reciproca di due gestori code

Seguire queste istruzioni di esempio per implementare l'autenticazione reciproca tra due gestori code, utilizzando certificati SSL o TLS autofirmati.

Informazioni su questa attività

Scenario:

- Sono presenti due gestori code, QM1 e QM2, che devono comunicare in modo sicuro. È necessaria l'autenticazione reciproca tra QM1 e QM2.
- Si è deciso di verificare la propria comunicazione protetta utilizzando certificati autofirmati.

La configurazione risultante è simile alla seguente:

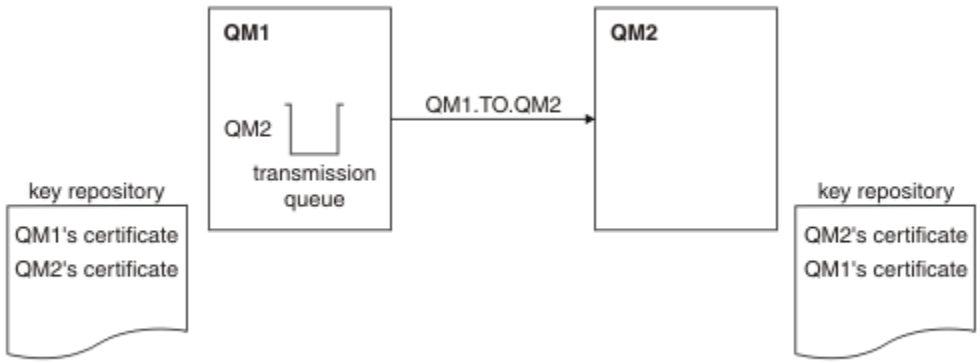


Figura 20. Configurazione risultante da questa attività

In Figura 14 a pagina 207, il repository delle chiavi per QM1 contiene il certificato per QM1 e il certificato pubblico da QM2. Il repository delle chiavi per QM2 contiene il certificato per QM2 e il certificato pubblico da QM1.

Procedura

1. Preparare il repository delle chiavi su ciascun gestore code, in base al sistema operativo:
   - Su sistemi UNIX, Linuxe Windows.

2. Creare un certificato autofirmato per ciascun gestore code:
   - Su sistemi UNIX, Linuxe Windows.
3. Estrarre una copia di ciascun certificato:
   - Su sistemi UNIX, Linuxe Windows.
4. Trasferire la parte pubblica del certificato QM1 sul sistema QM2 e viceversa, utilizzando un programma di utilità come FTP.
5. Aggiungere il certificato partner al repository delle chiavi per ciascun gestore code:
   - Su sistemi UNIX, Linuxe Windows.
6. In QM1, definire un canale mittente e la coda di trasmissione associata, immettendo comandi simili al seguente esempio:

```
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)
```

   Questo esempio utilizza CipherSpec RC4_MD5. Le CipherSpecs ad ogni estremità del canale devono essere uguali.
7. Su QM2, definire un canale ricevente immettendo un comando simile al seguente esempio:

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

   Il canale deve avere lo stesso nome del canale mittente definito nel passo 6 e utilizzare lo stesso CipherSpec.
8. Avviare il canale.

Risultati

I repository e i canali chiave vengono creati come illustrato in Figura 14 a pagina 207

Operazioni successive

Verificare che l'attività sia stata completata correttamente utilizzando i comandi DISPLAY. Se l'attività ha avuto esito positivo, l'output risultante è simile a quello mostrato nei seguenti esempi.

Dal gestore code QM1, immettere il comando seguente:

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

L'output risultante è simile al seguente esempio:

```
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)
```

Dal gestore code QM2, immettere il seguente comando:

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

L'output risultante è simile al seguente esempio:

```
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( )
```

In ogni caso, il valore di SSLPE deve corrispondere a quello del DN nel certificato partner creato nel passo 2. Il nome degli emittenti corrisponde al nome peer perché il certificato è autofirmato.

SSLPEER è facoltativo. Se viene specificato, il valore deve essere impostato in modo che sia consentito il DN nel certificato partner (creato nel passo 2). Per ulteriori informazioni relative all'utilizzo di SSLPEER, consultare WebSphere MQ rules for SSLPEER values .

Utilizzo dei certificati firmati dalla CA per l'autenticazione reciproca di due gestori code

Segui queste istruzioni di esempio per implementare l'autenticazione reciproca tra due gestori code, utilizzando i certificati SSL o TLS firmati da CA.

Informazioni su questa attività

Scenario:

- Si hanno due gestori code denominati QMA e QMB, che devono comunicare in modo sicuro. È necessaria l'autenticazione reciproca tra QMA e QMB.
- In futuro si prevede di utilizzare questa rete in un ambiente di produzione e quindi si è deciso di utilizzare i certificati firmati dalla CA fin dall'inizio.

La configurazione risultante è simile alla seguente:

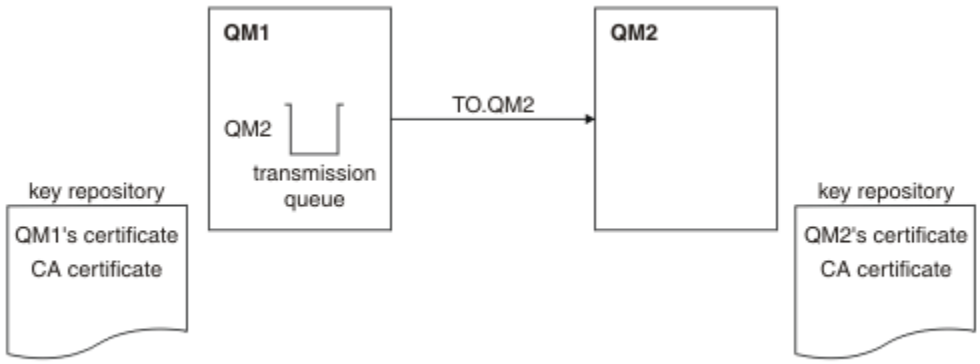


Figura 21. Configurazione risultante da questa attività

In Figura 15 a pagina 209, il repository delle chiavi per QMA contiene il certificato di QMA e il certificato CA. Il repository delle chiavi per QMB contiene il certificato QMB e il certificato CA. In questo esempio, sia il certificato QMA che il certificato QMB sono stati emessi dalla stessa CA. Se il certificato di QMA e il certificato di QMB sono stati emessi da AC differenti, i repository di chiavi per QMA e QMB devono contenere entrambi i certificati CA.

Procedura

1. Preparare il repository delle chiavi su ciascun gestore code, in base al sistema operativo:
   - Su sistemi UNIX, Linuxe Windows.
2. Richiedere un certificato firmato dalla CA per ciascun gestore code.

È possibile utilizzare diverse CA per i due gestori code.

- Su sistemi UNIX, Linuxe Windows.

3. Aggiungere il certificato dell'autorità di certificazione al repository delle chiavi per ciascun gestore code:

   Se i gestori code utilizzano autorità di certificazione differenti, il certificato CA per ogni autorità di certificazione deve essere aggiunto a entrambi i repository delle chiavi.

   - Su sistemi UNIX, Linuxe Windows.

4. Aggiungere il certificato firmato dalla CA al repository delle chiavi per ogni gestore code:

   - Su sistemi UNIX, Linuxe Windows.

5. In QMA, definire un canale mittente e la coda di trasmissione associata immettendo comandi come nel seguente esempio:

   ```
   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)
   ```

   Questo esempio utilizza CipherSpec RC4_MD5. Le CipherSpecs ad ogni estremità del canale devono essere uguali.

6. In QMB, definire un canale ricevente immettendo un comando simile al seguente esempio:

   ```
   DEFINE CHANNEL(TO.QMB) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(RC2_MD5_EXPORT)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL to QMB')
   ```

   Il canale deve avere lo stesso nome del canale mittente definito nel passo 6 e utilizzare lo stesso CipherSpec.

7. Avviare il canale:

Risultati

I repository e i canali chiave vengono creati come illustrato in Figura 15 a pagina 209.

Operazioni successive

Verificare che l'attività sia stata completata correttamente utilizzando i comandi DISPLAY. Se l'attività ha avuto esito positivo, l'output risultante è simile a quello mostrato nei seguenti esempi.

Dal gestore code QMA, immettere il comando seguente:

```
DISPLAY CHS(TO.QMB) SSLPEER SSLCERTI
```

L'output risultante è simile al seguente esempio:

```
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)
```

Dal gestore code QMB, immettere il seguente comando:

```
DISPLAY CHS(TO.QMB) SSLPEER SSLCERTI
```

L'output risultante è simile al seguente esempio:

```
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( )
```

In ogni caso, il valore di SSLPEER deve corrispondere a quello del DN (Distinguished Name) nel certificato partner creato nel passo 2. Il nome dell'emittente corrisponde al DN oggetto del certificato CA che ha firmato il certificato personale aggiunto nel Passo 4.

Connessione di due gestori code utilizzando l'autenticazione unidirezionale

Seguire queste istruzioni di esempio per modificare un sistema con l'autenticazione reciproca per consentire a un gestore code di connettersi utilizzando l'autenticazione unidirezionale a un altro; ovvero, quando il client SSL o TLS non invia un certificato.

Informazioni su questa attività

Scenario:

- I due gestori code (QM1 e QM2) sono stati configurati come in “Utilizzo dei certificati firmati dalla CA per l'autenticazione reciproca di due gestori code” a pagina 209.
- Si desidera modificare QM1 in modo che si connetta utilizzando l'autenticazione unidirezionale a QM2.

La configurazione risultante è simile alla seguente:

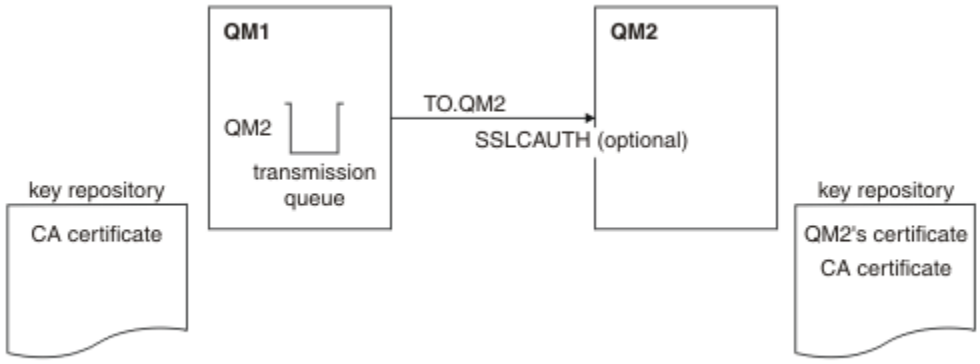


Figura 22. Gestori code che consentono l'autenticazione unidirezionale

Procedura

1. Rimuovere il certificato personale di QM1dal relativo repository delle chiavi, in base al sistema operativo:
   - Su sistemi UNIX, Linuxe Windows. Il certificato è etichettato come segue:
     - `ibmwebspheremq` seguito dal nome del gestore code ridotto in minuscolo. Ad esempio, per `QM1` , `ibmwebspheremqqm1`.
2. Opzionale: In QM1, se i canali SSL o TLS sono stati eseguiti in precedenza, aggiornare l'ambiente SSL o TLS.
3. Consentire connessioni anonime sul ricevitore.

Risultati

I repository chiave e i canali vengono modificati come illustrato in Figura 16 a pagina 211

Operazioni successive

Se il canale mittente era in esecuzione e si è immesso il comando REFRESH SECURITY TYPE (SSL) (nel passo 2), il canale viene riavviato automaticamente. Se il canale mittente non era in esecuzione, avviarlo.

All'estremità del server del canale, la presenza del valore del parametro del nome peer sul pannello di stato del canale indica che è stato trasmesso un certificato client.

Verificare che l'attività sia stata completata correttamente immettendo alcuni comandi DISPLAY. Se l'attività ha avuto esito positivo, l'output risultante è simile a quello mostrato nei seguenti esempi:

Dal gestore code QM1 , immettere il comando seguente:

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

L'output risultante sarà simile al seguente esempio:

```
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)
```

Dal gestore code QM2 , immettere il seguente comando:

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

L'output risultante sarà simile al seguente esempio:

```
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( )
```

In QM2, il campo SSLPE è vuoto e indica che QM1 non ha inviato un certificato. In QM1, il valore di SSLPEER corrisponde a quello del DN nel certificato personale di QM2.

Connessione sicura di un client a un gestore code

Le comunicazioni sicure che utilizzano i protocolli di sicurezza crittografica SSL o TLS richiedono l'impostazione dei canali di comunicazione e la gestione dei certificati digitali che verranno utilizzati per l'autenticazione.

Per configurare l'installazione SSL o TLS è necessario definire i canali per utilizzare SSL o TLS. È inoltre necessario ottenere e gestire i certificati digitali. Su un sistema di test, è possibile utilizzare certificati autofirmati o certificati emessi da un'autorità di certificazione (CA) locale. Su un sistema di produzione, non utilizzare certificati autofirmati. Per ulteriori informazioni, consultare ../zs14140_.dita.

Per informazioni complete sulla creazione e la gestione dei certificati, consultare “Utilizzo di SSL o TLS su sistemi UNIX, Linux, and Windows” a pagina 114.

Questa raccolta di argomenti introduce le attività relative all'impostazione delle comunicazioni SSL e fornisce una guida dettagliata sul completamento di tali attività.

Si potrebbe anche voler verificare l'autenticazione client SSL o TLS, che sono una parte facoltativa dei protocolli. Durante l'handshake SSL o TLS, il client SSL o TLS ottiene e convalida sempre un certificato digitale dal server. Con l'implementazione WebSphere MQ , il server SSL o TLS richiede sempre un certificato dal client.

Su sistemi UNIX, Linux, and Windows , il client SSL o TLS invia un certificato solo se ha un'etichetta nel formato WebSphere MQ corretto, che è `ibmwebspheremq` seguito dall'ID utente di collegamento modificato in minuscolo, ad esempio `ibmwebspheremqmyuserid`.

WebSphere MQ utilizza il prefisso `ibmwebspheremq` su un'etichetta per evitare confusione con i certificati per altri prodotti. Assicurarsi di specificare l'intera etichetta del certificato in minuscolo.

Il server SSL o TLS convalida sempre il certificato client, se ne viene inviato uno. Se il client non invia un certificato, l'autenticazione non riesce solo se la fine del canale che agisce come server SSL o TLS è definita con il parametro SSLCAUTH impostato su REQUIRED o con un valore del parametro SSLPEER impostato. Per ulteriori informazioni sulla connessione anonima di un gestore code, consultare “Connessione anonima di un client a un gestore code” a pagina 216.

Utilizzo di certificati autofirmati per l'autenticazione reciproca di un client e di un gestore code

Seguire queste istruzioni di esempio per implementare l'autenticazione reciproca tra un client e un gestore code, utilizzando certificati SSL o TLS autofirmati.

Informazioni su questa attività

Scenario:

- Si dispone di un client C1e di un gestore code, QM1, che devono comunicare in modo sicuro. È necessaria l'autenticazione reciproca tra C1 e QM1.
- Si è deciso di verificare la comunicazione protetta utilizzando i certificati autofirmati.

DCM su IBM i non supporta i certificati autofirmati, pertanto questa attività non è applicabile sui sistemi IBM i .

La configurazione risultante è simile alla seguente:

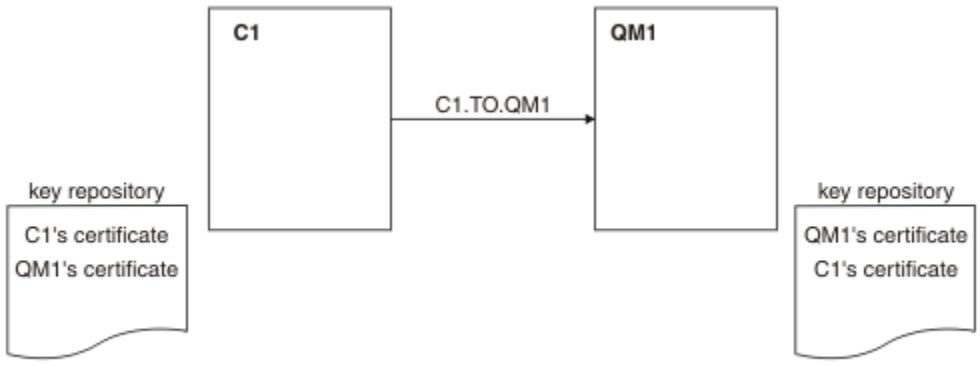


Figura 23. Configurazione risultante da questa attività

In Figura 17 a pagina 213, il repository delle chiavi per QM1 contiene il certificato per QM1 e il certificato pubblico da C1. Il repository delle chiavi per C1 contiene il certificato per C1 e il certificato pubblico da QM1.

Procedura

1. Preparare il repository chiavi sul client e sul gestore code, in base al sistema operativo:
   - Su sistemi UNIX, Linuxe Windows.
2. Creare certificati autofirmati per il client e il gestore code:
   - Su sistemi UNIX, Linuxe Windows.
3. Estrarre una copia di ciascun certificato:
   - Su sistemi UNIX, Linuxe Windows.
4. Trasferire la parte pubblica del certificato C1 al sistema QM1 e viceversa, utilizzando un programma di utilità come FTP.
5. Aggiungere il certificato partner al repository delle chiavi per client e gestore code:
   - Su sistemi UNIX, Linuxe Windows.
6. Immettere il comando REFRESH SECURITY TYPE (SSL) sul gestore code.
7. Definire un canale di connessione client in uno dei seguenti modi:
   - Utilizzando la chiamata MQCONNX con la struttura MQSCO su C1, come descritto in Creazione di un canale di connessione client sul client WebSphere MQ MQI.
   - Utilizzando una tabella di definizione di canale client, come descritto in Creazione di definizioni di connessione server e di connessione client nel server .
8. In QM1, definire un canale di connessione server, immettendo un comando simile al seguente esempio:

   ```
   DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(RC4_MD5_US)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from C1 to QM1')
   ```

   Il canale deve avere lo stesso nome del canale di collegamento client definito nel passo 6 e utilizzare lo stesso CipherSpec.

Risultati

I repository e i canali chiave vengono creati come illustrato in Figura 17 a pagina 213

Operazioni successive

Verificare che l'attività sia stata completata correttamente utilizzando i comandi DISPLAY. Se l'attività ha avuto esito positivo, l'output risultante è simile a quello mostrato nel seguente esempio.

Dal gestore code QM1, immettere il comando seguente:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

L'output risultante è simile al seguente esempio:

```
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)
```

È facoltativo impostare l'attributo filtro SSLPEER delle definizioni di canale. Se la definizione di canale SSLPEER è impostata, il suo valore deve corrispondere al DN soggetto nel certificato partner creato nel passo 2. Dopo una connessione riuscita, il campo SSLPEER nell'output DISPLAY CHSTATUS mostra il DN dell'oggetto del certificato client remoto.

Utilizzo di certificati firmati CA per l'autenticazione reciproca di un client e di un gestore code

Seguire queste istruzioni di esempio per implementare l'autenticazione reciproca tra un client e un gestore code, utilizzando i certificati SSL o TLS firmati dalla CA.

Informazioni su questa attività

Scenario:

- Si dispone di un client C1e di un gestore code, QM1, che devono comunicare in modo sicuro. È necessaria l'autenticazione reciproca tra C1 e QM1.
- In futuro si prevede di utilizzare questa rete in un ambiente di produzione e quindi si è deciso di utilizzare i certificati firmati dalla CA fin dall'inizio.

La configurazione risultante è simile alla seguente:

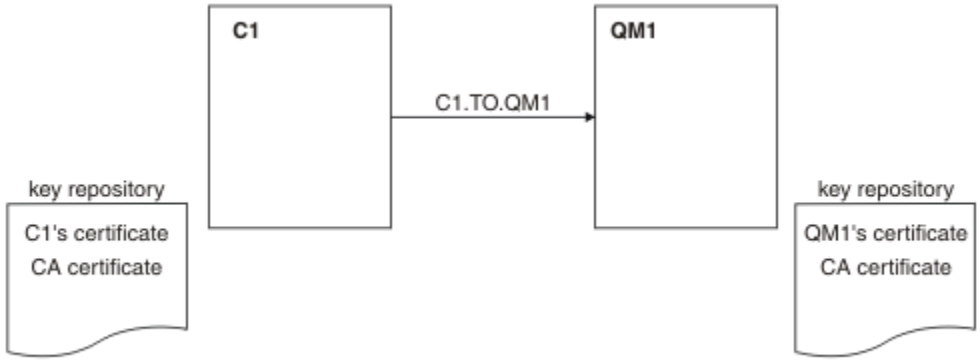


Figura 24. Configurazione risultante da questa attività

In Figura 18 a pagina 215, il repository delle chiavi per C1 contiene il certificato per C1 e il certificato CA. Il repository delle chiavi per QM1 contiene il certificato per QM1 e il certificato CA. In questo esempio, sia il certificato di C1che quello di QM1sono stati emessi dalla stessa CA. Se il certificato di C1e il certificato di QM1sono stati emessi da diverse CA, i repository delle chiavi per C1 e QM1 devono contenere entrambi i certificati CA.

Procedura

1. Preparare il repository chiavi sul client e sul gestore code, in base al sistema operativo:
   - Su sistemi UNIX, Linuxe Windows.
2. Richiedere un certificato firmato CA per il client e il gestore code.
   È possibile utilizzare diverse CA per il client e il gestore code.
   - Su sistemi UNIX, Linuxe Windows.
3. Aggiungere il certificato dell'autorità di certificazione al repository delle chiavi per client e gestore code.
   Se il client e il gestore code utilizzano autorità di certificazione differenti, il certificato CA per ogni autorità di certificazione deve essere aggiunto a entrambi i repository delle chiavi.
   - Su sistemi UNIX, Linuxe Windows.
4. Aggiungere il certificato firmato CA al repository delle chiavi per il client e il gestore code:
   - Su sistemi UNIX, Linuxe Windows.
5. Definire un canale di connessione client in uno dei seguenti modi:

- Utilizzando la chiamata MQCONNX con la struttura MQSCO su C1, come descritto in Creazione di un canale di connessione client sul client WebSphere MQ MQI.
- Utilizzando una tabella di definizione di canale client, come descritto in Creazione di definizioni di connessione server e di connessione client nel server .

6. In QM1, definire un canale di connessione server immettendo un comando simile al seguente esempio:

```
DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(RC2_MD5_EXPORT)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from C1 to QM1')
```

   Il canale deve avere lo stesso nome del canale di collegamento client definito nel passo 6 e utilizzare lo stesso CipherSpec.

Risultati

I repository e i canali chiave vengono creati come illustrato in Figura 18 a pagina 215.

Operazioni successive

Verificare che l'attività sia stata completata correttamente utilizzando i comandi DISPLAY. Se l'attività ha avuto esito positivo, l'output risultante è simile a quello mostrato nel seguente esempio.

Dal gestore code QM1, immettere il comando seguente:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
```

L'output risultante è simile al seguente esempio:

```
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)
```

Il campo SSLPEER nell'output DISPLAY CHSTATUS mostra il DN oggetto del certificato client remoto creato nel passo 2. Il nome dell'emittente corrisponde al DN oggetto del certificato CA che ha firmato il certificato personale aggiunto nel Passo 4.

Connessione anonima di un client a un gestore code

Seguire queste istruzioni di esempio per modificare un sistema con autenticazione reciproca per consentire a un gestore code di connettersi in modo anonimo a un'altro.

Informazioni su questa attività

Scenario:

- Il gestore code e il client (QM1 e C1) sono stati impostati come in “Utilizzo di certificati firmati CA per l'autenticazione reciproca di un client e di un gestore code” a pagina 215.
- Si desidera modificare C1 in modo che si colleghi in modo anonimo a QM1.

La configurazione risultante è simile alla seguente:

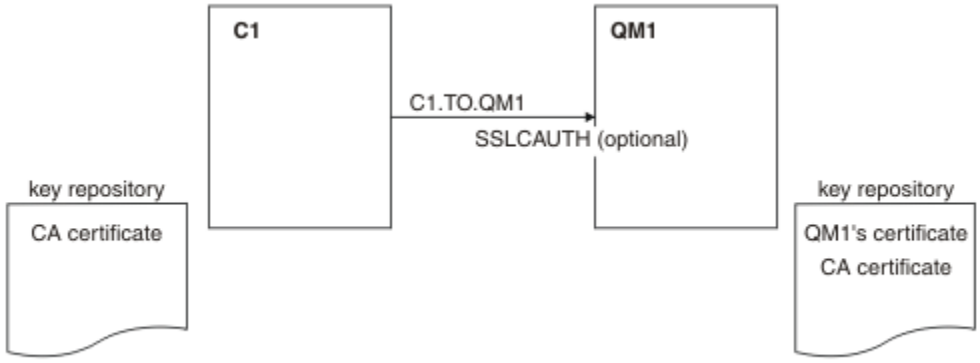


Figura 25. Client e gestore code che consentono la connessione anonima

Procedura

1. Rimuovere il certificato personale dal repository chiavi per C1, in base al sistema operativo:
   - Su sistemi UNIX, Linuxe Windows. Il certificato è etichettato come segue:
     - `ibmwebspheremq` seguito dall'ID utente di collegamento ripiegato in minuscolo, ad esempio `ibmwebspheremqmyuserid`.
2. Riavviare l'applicazione client o far sì che l'applicazione client chiuda e riapra tutte le connessioni SSL o TLS.
3. Consentire le connessioni anonime sul gestore code, immettendo il comando seguente:

```
ALTER CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) SSLCAUTH(OPTIONAL)
```

Risultati

I repository chiave e i canali vengono modificati come illustrato in Figura 19 a pagina 217

Operazioni successive

All'estremità del server del canale, la presenza del valore del parametro del nome peer sul pannello di stato del canale indica che è stato trasmesso un certificato client.

Verificare che l'attività sia stata completata correttamente immettendo alcuni comandi DISPLAY. Se l'attività ha avuto esito positivo, l'output risultante è simile a quello mostrato nel seguente esempio:

Dal gestore code QM1, immettere il comando seguente:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

L'output risultante sarà simile al seguente esempio:

```
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)
```

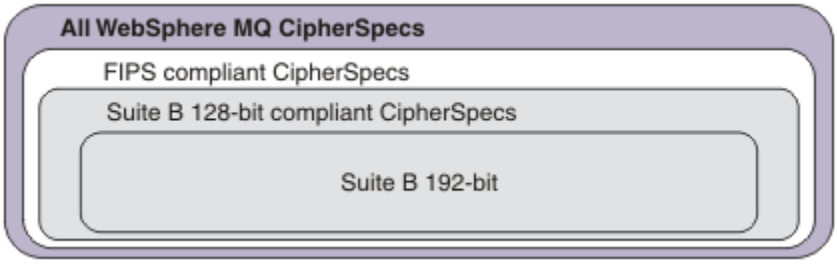
I campi SSLCERTI e SSLPEER sono vuoti e indicano che C1 non ha inviato un certificato.

Specifica di CipherSpecs

Specificare una CipherSpec utilizzando il parametro **SSLCIPH** nel comando MQSC **DEFINE CHANNEL** o nel comando MQSC **ALTER CHANNEL** .

Alcuni dei CipherSpecs che possono essere utilizzati con IBM WebSphere MQ sono compatibili con FIPS. Altri, come NULL_MD5, non lo sono. Allo stesso modo, alcuni CipherSpecs compatibili con FIPS sono compatibili con Suite B anche se altri non lo sono. Tutti i CipherSpecs compatibili con Suite B sono compatibili con FIPS. Tutti i CipherSpecs compatibili con la Suite B rientrano in due gruppi: 128 bit (ad esempio, ECDHE_ECDSA_AES_128_GCM_SHA256) e 192 bit (ad esempio ECDHE_ECDSA_AES_256_GCM_SHA384),

Il seguente diagramma illustra la relazione tra questi sottoinsiemi:



Le specifiche di cifratura che puoi utilizzare con il supporto IBM WebSphere MQ SSL e TLS sono elencate nella seguente tabella. Quando si richiede un certificato personale, si specifica una dimensione di chiave per la coppia di chiavi pubblica e privata. La dimensione della chiave utilizzata durante l'handshake SSL è la dimensione memorizzata nel certificato a meno che non sia determinata da CipherSpec, come indicato nella tabella.

Nome CipherSpec	Protocollo utilizzato	Algoritmo MAC	Algoritmo di codifica	Bit di codifica	FIPS[1]	Suite B a 128 bit	Suite B 192 bit
NULL_MD5 [a]	SSL 3.0	MD5	Nessuna	0	No	No	No
NULL_SHA [a]	SSL 3.0	SHA-1	Nessuna	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	Sì	No	No
TLS_RSA_WITH_AES_256_CBC_SHA [4] [a]	TLS 1.0	SHA-1	AES	256	Sì	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	Sì	No	No

Nome CipherSpec	Protocollo utilizzato	Algoritmo MAC	Algoritmo di codifica	Bit di codifica	FIPS[1]	Suite B a 128 bit	Suite B 192 bit
TLS_RSA_WITH_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sì	No	No
TLS_RSA_WITH_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sì	No	No
TLS_RSA_WITH_AES_256_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	256	Sì	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	Sì	No	No
ECDHE_ECDSA_AES_256_CBC_SHA384 [b]	TLS 1.2	SHA-384	AES	256	Sì	No	No
ECDHE_RSA_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sì	No	No
ECDHE_RSA_AES_256_CBC_SHA384 [b]	TLS 1.2	SHA-384	AES	256	Sì	No	No
ECDHE_ECDSA_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sì	Sì	No
ECDHE_ECDSA_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sì	No	Sì
ECDHE_RSA_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sì	No	No
ECDHE_RSA_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sì	No	No
TLS_RSA_WITH_NULL_SHA256 [b]	TLS 1.2	SHA-256	Nessuna	0	No	No	No
ECDHE_RSA_NULL_SHA256 [b]	TLS 1.2	SHA-1	Nessuna	0	No	No	No
ECDHE_ECDSA_NULL_SHA256 [b]	TLS 1.2	SHA-1	Nessuna	0	No	No	No
TLS_RSA_WITH_NULL_NULL [b]	TLS 1.2	Nessuna	Nessuna	0	No	No	No
TLS_RSA_WITH_RC4_128_SHA256 [b]	TLS 1.2	SHA-1	RC4	128	No	No	No

Nome CipherSpec	Protocollo utilizzato	Algoritmo MAC	Algoritmo di codifica	Bit di codifica	FIPS[1]	Suite B a 128 bit	Suite B 192 bit

Note:

1. Specifica se la CipherSpec è certificata da FIPS su una piattaforma certificata FIPS. Consultare Federal Information Processing Standards (FIPS) per una spiegazione di FIPS.
2. La dimensione massima della chiave di handshake è 512 bit. Se uno dei certificati scambiati durante l'handshake SSL ha una dimensione di chiave maggiore di 512 bit, viene creata una chiave temporanea di 512 bit da utilizzare durante l'handshake.
3. La dimensione della chiave di handshake è 1024 bit.
4. Questa CipherSpec non può essere utilizzata per proteggere una connessione da WebSphere MQ Explorer a un gestore code a meno che non vengano applicati i file delle politiche senza limitazioni appropriati al JRE utilizzato da Explorer.
5. Questa CipherSpec era certificata FIPS 140-2 prima del 19 maggio 2007.
6. Questa CipherSpec era certificata FIPS 140-2 prima del 19 maggio 2007. Il nome `FIPS_WITH_DES_CBC_SHA` è storico e riflette il fatto che questa CipherSpec era in precedenza (ma non è più) compatibile con FIPS. Questa CipherSpec è obsoleta e non se ne consiglia l'utilizzo.
7. Questa CipherSpec può essere utilizzata per trasferire fino a 32 GB di dati prima che la connessione venga terminata con l'errore AMQ9288. Per evitare questo errore, non utilizzare il triplo DES o abilitare la reimpostazione della chiave segreta quando si utilizza questa CipherSpec.

Supporto della piattaforma:

- a Disponibile su tutte le piattaforme supportate.
- b Disponibile solo su piattaforme UNIX, Linux, and Windows .

Concetti correlati

“Certificati digitali e compatibilità CipherSpec in IBM WebSphere MQ” a pagina 34
Questo argomento fornisce informazioni su come scegliere i CipherSpecs e i certificati digitali appropriati per la politica di sicurezza, evidenziando la relazione tra CipherSpecs e i certificati digitali in IBM WebSphere MQ.

Riferimenti correlati

Definire il canale
MODIFICA CANALE

Acquisizione di informazioni su CipherSpecs utilizzando IBM WebSphere MQ Explorer

È possibile utilizzare IBM WebSphere MQ Explorer per visualizzare le descrizioni di CipherSpecs.

Utilizzare la seguente procedura per ottenere informazioni su CipherSpecs in “Specifica di CipherSpecs” a pagina 218:

1. Aprire **IBM WebSphere MQ Explorer** ed espandere la cartella **Gestori code** .
2. Assicurarsi di aver avviato il gestore code.
3. Selezionare il gestore code che si desidera utilizzare e fare clic su **Canali**.
4. Fare clic con il pulsante destro del mouse sul canale che si desidera utilizzare e selezionare **Proprietà**.
5. Selezionare la pagina delle proprietà **SSL** .
6. Selezionare dall'elenco la CipherSpec che si desidera utilizzare. Una descrizione viene visualizzata nella finestra sotto l'elenco.

Alternative per specificare CipherSpecs

Per le piattaforme in cui il sistema operativo fornisce il supporto SSL, il sistema potrebbe supportare i nuovi CipherSpecs. È possibile specificare un nuovo CipherSpec con il parametro SSLCIPH, ma il valore fornito dipende dalla piattaforma.

Nota: Questa sezione non si applica ai sistemi UNIX, Linux o Windows , perché i CipherSpecs vengono forniti con il prodotto WebSphere MQ , quindi i nuovi CipherSpecs non diventano disponibili dopo la spedizione.

Per le piattaforme in cui il sistema operativo fornisce il supporto SSL, il proprio sistema potrebbe supportare nuovi CipherSpecs non inclusi in "Specifica di CipherSpecs" a pagina 218. È possibile specificare un nuovo CipherSpec con il parametro SSLCIPH, ma il valore fornito dipende dalla piattaforma. In tutti i casi, la specifica *deve* corrispondere a un CipherSpec SSL valido e supportato dalla versione di SSL in esecuzione sul sistema.

IBM i

Una stringa di due caratteri che rappresenta un valore esadecimale.

Per ulteriori informazioni sui valori consentiti, fare riferimento alla documentazione del prodotto appropriata (ricercare *cipher_spec* nella documentazione del prodottoIBM i).

È possibile utilizzare il comando CHGMQMCHL o CRTMQMCHL per specificare il valore, ad esempio:

```
CRTMQMCHL CHLNAME('channel name') SSLCIPH('hexadecimal value')
```

È anche possibile utilizzare il comando ALTER QMGR MQSC per impostare il parametro SSLCIPH .

z/OS

Una stringa di due caratteri che rappresenta un valore esadecimale. I codici esadecimali corrispondono ai valori definiti nel protocollo SSL.

Per ulteriori informazioni, fare riferimento alla descrizione di gsk_environment_open () nel capitolo di riferimento API di *z/OS Cryptographic Services System SSL Programming*, SC24-5901, dove è presente un elenco di tutte le specifiche di cifratura SSL V3.0 e TLS V1.0 supportate sotto forma di codici esadecimali a due cifre.

Considerazioni sui cluster WebSphere MQ

Con i cluster WebSphere MQ è più sicuro utilizzare i nomi CipherSpec in "Specifica di CipherSpecs" a pagina 218. Se si utilizza una specifica alternativa, tenere presente che la specifica potrebbe non essere valida su altre piattaforme. Per ulteriori informazioni, fare riferimento a "SSL e cluster" a pagina 249.

Specifica di un CipherSpec per un client IBM WebSphere MQ MQI

Sono disponibili tre opzioni per specificare un CipherSpec per un client MQI IBM WebSphere MQ .

Le opzioni disponibili sono:

- Utilizzo di una tabella di definizione di canale
- Utilizzo del campo SSLCipherSpec nella struttura MQCD, in MQCD_VERSION_7 o superiore, su una chiamata MQCONNX.
- Utilizzo di Active Directory (su sistemi Windows con supporto Active Directory)

Specifica di una CipherSuite con classi IBM WebSphere MQ per Java e classi IBM WebSphere MQ per JMS

Le classi IBM WebSphere MQ per Java e IBM WebSphere MQ per JMS specificano CipherSuites in maniera diversa dalle altre piattaforme.

Per informazioni su come specificare una CipherSuite con classi IBM WebSphere MQ per Java, consultare Supporto SSL (Secure Sockets Layer).

Per informazioni su come specificare una CipherSuite con le classi IBM WebSphere MQ per JMS, consultare Utilizzo di SSL (Secure Sockets Layer) con WebSphere MQ classes for JMS.

Audit

È possibile controllare le intrusioni di sicurezza o i tentativi di intrusioni utilizzando i messaggi di evento. È inoltre possibile verificare la sicurezza del sistema utilizzando IBM WebSphere MQ Explorer.

Per rilevare i tentativi di eseguire azioni non autorizzate, come la connessione a un gestore code o l'inserimento di un messaggio in una coda, esaminare i messaggi di evento prodotti dai gestori code, in particolare i messaggi di evento di autorizzazione. Per ulteriori informazioni sui messaggi di evento del gestore code, consultare Eventi del gestore codee, per ulteriori informazioni sul monitoraggio degli eventi in generale, fare riferimento a Controllo eventi .

Proteggere i cluster

Autorizzare o impedire ai gestori code di unirsi ai cluster o di inserire messaggi nelle code cluster. Forzare un gestore code a lasciare un cluster. Tenere conto di alcune considerazioni aggiuntive quando si configura SSL per i cluster.

Arresto dei messaggi di invio dei gestori code non autorizzati

Impedire ai gestori code non autorizzati di inviare messaggi al proprio gestore code utilizzando un'uscita di sicurezza del canale.

Prima di iniziare

Il clustering non ha alcun effetto sul modo in cui la sicurezza esce dal lavoro. È possibile limitare l'accesso a un gestore code nello stesso modo in cui si farebbe in un ambiente di accodamento distribuito.

Informazioni su questa attività

Impedire ai gestori code selezionati di inviare messaggi al proprio gestore code:

Procedura

1. Definire un programma di uscita di sicurezza del canale sulla definizione del canale CLUSRCVR .
2. Scrivere un programma che autentica i gestori code che tentano di inviare messaggi sul canale ricevente del cluster e nega loro l'accesso se non sono autorizzati.

Operazioni successive

I programmi di uscita di sicurezza del canale vengono richiamati all'inizio e alla fine di MCA.

Arresto dei gestori code non autorizzati che immettono messaggi nelle code

Utilizzare l'attributo di autorizzazione di inserimento del canale sul canale ricevente del cluster per arrestare i gestori code non autorizzati che inseriscono i messaggi nelle code. Autorizzare un gestore code remoto controllando l'ID utente nel messaggio utilizzando RACF su z/OSo l'OAM su altre piattaforme.

Informazioni su questa attività

Utilizzare le funzioni di sicurezza di una piattaforma e il meccanismo di controllo accessi in WebSphere MQ per controllare l'accesso alle code.

Procedura

1. Per impedire a determinati gestori code di inserire messaggi su una coda, utilizzare le funzioni di protezione disponibili sulla piattaforma.

   Ad esempio:

   - RACF o altri gestori di sicurezza esterni su WebSphere MQ for z/OS
   - L'OAM (object authority manager) su altre piattaforme.
2. Utilizzare l'attributo di immissione, PUTAUT, nella definizione del canale CLUSRCVR .

   L'attributo PUTAUT consente di specificare quali identificativi utente devono essere utilizzati per stabilire l'autorizzazione a inserire un messaggio in una coda.

   Le opzioni sull'attributo PUTAUT sono:

   DEF
   Utilizzare l'ID utente predefinito. Su z/OS, la verifica potrebbe implicare l'uso sia dell'ID utente ricevuto dalla rete che di quello derivato da MCAUSER.

   CTX
   Utilizzare l'ID utente nelle informazioni di contesto associate al messaggio. In z/OS la verifica potrebbe implicare l'utilizzo dell'ID utente ricevuto dalla rete o quello derivato da MCAUSERo entrambi. Utilizzare questa opzione se il link è attendibile e autenticato.

   ONLYMCA (soloz/OS)
   Come per DEF, ma qualsiasi ID utente ricevuto dalla rete non viene utilizzato. Utilizzare questa opzione se il link non è attendibile. Si desidera consentire solo una serie specifica di azioni, definite per MCAUSER.

   ALTMCA (soloz/OS)
   Come per CTX, ma qualsiasi ID utente ricevuto dalla rete non viene utilizzato.

Autorizzazione all'inserimento di messaggi nelle code del cluster remoto

Sulla piattaforma, autorizzare l'accesso per connettersi al gestore code e per inserirlo nella coda su tale gestore code.

Informazioni su questa attività

Il comportamento predefinito è quello di eseguire il controllo accessi su SYSTEM.CLUSTER.TRANSMIT.QUEUE. Notare che questo comportamento si applica, anche se si utilizzano più code di trasmissione.

Il comportamento specifico descritto in questo argomento si applica solo quando l'attributo **ClusterQueueAccessControl** nel file qm.ini è configurato come *RQMName*, come descritto nella sezione Stanza di sicurezza , e il gestore code è stato riavviato.

Procedura

- Per sistemi UNIX, Linux e Windows , immettere i seguenti comandi:

```
setmqaut -m QMgrName -t qmgr -g GroupName +connect
setmqaut -m QMgrName -t queue -n QueueName -g GroupName -all +put
```

  L'utente può inserire i messaggi solo nella coda cluster specificata e in nessun' altra coda cluster.

  I nomi delle variabili hanno i seguenti significati:

  QMgrName
  Il nome del gestore code.

  GroupName
  Il nome del gruppo a cui concedere l'accesso.

QueueName
Nome della coda o profilo generico per cui modificare le autorizzazioni.

Operazioni successive

Se si specifica una coda di risposta quando si inserisce un messaggio su una coda cluster, l'applicazione che utilizza deve disporre dell'autorizzazione per inviare la risposta. Impostare questa autorizzazione seguendo le istruzioni in “Concessione dell'autorità per inserire i messaggi in una coda cluster remota” a pagina 193.

Informazioni correlate
Stanza di sicurezza in qm.ini

Impedire ai gestori code di unirsi a un cluster

Se un gestore code anomalo si unisce a un cluster, è difficile impedirgli di ricevere i messaggi che non si desidera ricevere.

Procedura

Se si desidera assicurarsi che solo alcuni gestori code autorizzati si uniscano a un cluster, è possibile scegliere tra tre tecniche:

- Utilizzando i record di autenticazione di canale è possibile bloccare la connessione del canale cluster in base a: l'indirizzo IP remoto, il nome del gestore code remoto o il DN (Distinguished Name) SSL/TLS fornito dal sistema remoto.
- Scrivere un programma di uscita per impedire ai gestori code non autorizzati di scrivere in `SYSTEM.CLUSTER.COMMAND.QUEUE`. Non limitare l'accesso a `SYSTEM.CLUSTER.COMMAND.QUEUE` in modo che nessun gestore code possa scrivere su di esso, altrimenti si impedirebbe a qualsiasi gestore code di unirsi al cluster.
- Un programma di uscita di sicurezza sulla definizione di canale `CLUSRCVR` .

Uscite di sicurezza sui canali cluster

Considerazioni aggiuntive quando si utilizzano uscite di sicurezza sui canali cluster.

Informazioni su questa attività

Quando un canale mittente del cluster viene avviato per la prima volta, utilizza attributi definiti manualmente da un amministratore di sistema. Quando il canale viene arrestato e riavviato, prende gli attributi dalla corrispondente definizione di canale ricevente del cluster. La definizione del canale mittente del cluster originale viene sovrascritta con i nuovi attributi, incluso `SecurityExit` .

Procedura

1. È necessario definire un'uscita di sicurezza sia sull'estremità mittente del cluster che sull'estremità ricevente del cluster di un canale.

   La connessione iniziale deve essere effettuata con un handshake di uscita di sicurezza, anche se il nome dell'uscita di sicurezza viene inviato dalla definizione del ricevitore del cluster.
2. Convalidare il `PartnerName` nella struttura `MQCXP` nell'uscita di sicurezza.

   L'uscita deve consentire l'avvio del canale solo se il gestore code partner è autorizzato
3. Progettare l'uscita di sicurezza sulla definizione del ricevente del cluster da avviare.
4. Se lo si progetta come iniziato dal mittente, un gestore code non autorizzato senza un'uscita di sicurezza può unirsi al cluster perché non viene eseguito alcun controllo di sicurezza.

   Non fino a quando il canale non viene arrestato e riavviato, il nome `SCYEXIT` può essere inviato dalla definizione del ricevente del cluster e vengono eseguiti controlli di sicurezza completi.

5. Per visualizzare la definizione di canale mittente del cluster attualmente in uso, utilizzare il comando:

   ```
   DISPLAY CLUSQMGR(queue manager) ALL
   ```

   Il comando visualizza gli attributi che sono stati inviati dalla definizione ricevente del cluster.
6. Per visualizzare la definizione originale, utilizzare il comando:

   ```
   DISPLAY CHANNEL(channel name) ALL
   ```

7. Potrebbe essere necessario definire un'uscita di definizione automatica del canale, CHADEXIT, sul gestore code del mittente del cluster, se i gestori code si trovano su piattaforme differenti.

   Utilizzare l'uscita di definizione automatica del canale per impostare l'attributo SecurityExit su un formato appropriato per la piattaforma di destinazione.
8. Distribuire e configurare l'uscita di sicurezza.

   Windows, sistemi UNIX and Linux

   - La libreria di collegamento dinamico dell'uscita di sicurezza deve trovarsi nel percorso specificato nell'attributo SCYEXIT della definizione del canale.
   - La libreria di link dinamici di uscita di definizione automatica del canale deve trovarsi nel percorso specificato nell'attributo CHADEXIT della definizione del gestore code.

Forzare i gestori code indesiderati a lasciare un cluster

Forzare un gestore code indesiderato a lasciare un cluster immettendo il comando RESET CLUSTER su un gestore code del repository completo.

Informazioni su questa attività

È possibile forzare un gestore code indesiderato a lasciare un cluster. Se, ad esempio, un gestore code viene eliminato ma i relativi canali riceventi del cluster sono ancora definiti per il cluster. Potresti voler riordinare.

Solo i gestori code del repository completo sono autorizzati ad espellere un gestore code da un cluster.

Seguire questa procedura per espellere il gestore code OSLO dal cluster NORWAY:

Procedura

1. Su un gestore code del repository completo, immettere il comando:

   ```
   RESET CLUSTER(NORWAY) QMNAME(OSLO) ACTION(FORCEREMOVE)
   ```

2. In alternativa, utilizzare QMID invece di QMNAME nel comando:

   ```
   RESET CLUSTER(NORWAY) QMID(qmid) ACTION(FORCEREMOVE)
   ```

Risultati

Il gestore code che viene rimosso forzatamente non cambia: le sue definizioni di cluster locale mostrano che si trova nel cluster. Le definizioni in tutti gli altri gestori code non vengono visualizzate nel cluster.

Come impedire ai gestori code di ricevere messaggi

È possibile evitare che un gestore code del cluster riceva messaggi che non è autorizzato a ricevere utilizzando i programmi di uscita.

Informazioni su questa attività

È difficile impedire a un gestore code membro di un cluster di definire una coda. Esiste il pericolo che un gestore code non valido si unisca a un cluster e definisca la propria istanza di una delle code nel

cluster. Ora può ricevere messaggi che non è autorizzato a ricevere. Per evitare che un gestore code riceva messaggi, utilizzare una delle seguenti opzioni fornite nella procedura.

Procedura

- Un programma di uscita canale su ogni canale mittente del cluster. Il programma di uscita utilizza il nome connessione per determinare l'idoneità del gestore code di destinazione a inviare i messaggi.
- Un programma di uscita del carico di lavoro del cluster, che utilizza i record di destinazione per stabilire l'idoneità della coda di destinazione e del gestore code a inviare i messaggi.

SSL e cluster

Quando si configura SSL per i cluster, tenere presente che una definizione di canale CLUSRCVR viene propagata ad altri gestori code come canale CLUSSDR definito automaticamente. Se un canale CLUSRCVR utilizza SSL, è necessario configurare SSL su tutti i gestori code che comunicano utilizzando il canale.

Per ulteriori informazioni su SSL, consultare Supporto diWebSphere MQ per SSL e TLS. Il consiglio è generalmente applicabile ai canali cluster, ma è possibile considerare in modo particolare quanto segue:

In un cluster IBM WebSphere MQ una particolare definizione di canale CLUSRCVR viene spesso propagata a molti altri gestori code in cui viene trasformata in un CLUSSDRdefinito automaticamente. Successivamente, il CLUSSDR definito automaticamente viene utilizzato per avviare un canale per CLUSRCVR. Se CLUSRCVR è configurato per la connettività SSL, si applicano le seguenti considerazioni:

- Tutti i gestori code che desiderano comunicare con questo CLUSRCVR devono avere accesso al supporto SSL. Questo provisioning SSL deve supportare CipherSpec per il canale.
- I diversi gestori code a cui sono stati propagati i canali mittenti del cluster definiti automaticamente avranno ciascuno un DN differente associato. Se il controllo peer del DN (distinguished name) deve essere utilizzato su CLUSRCVR , deve essere impostato in modo che tutti i DN che possono essere ricevuti corrispondano correttamente.

  Ad esempio, si supponga che tutti i gestori code che ospiteranno i canali mittenti del cluster che si connetteranno a un particolare CLUSRCVR, abbiano certificati associati. Si supponga inoltre che i DN (distinguished name) in tutti questi certificati definiscano il paese come Regno Unito, l'organizzazione come IBM, l'unità organizzativa come IBM WebSphere MQ Development e tutti abbiano nomi comuni nel formato DEVT.QMnnn, dove nnn è numerico.

  In questo caso, un valore SSLPEER di `C=UK, O=IBM, OU=WebSphere MQ Development, CN=DEVT.QM*` su CLUSRCVR consentirà a tutti i canali mittenti del cluster richiesti di connettersi correttamente, ma impedirà la connessione di canali mittenti del cluster indesiderati.
- Se vengono utilizzate le stringhe CipherSpec personalizzate, tenere presente che i formati stringa personalizzati non sono consentiti su tutte le piattaforme. Un esempio di ciò è che la CipherSpec stringa RC4_SHA_US ha il valore 05 su IBM i ma non è una specifica valida su sistemi UNIX, Linux o Windows . Quindi, se i parametri SSLCIPH personalizzati vengono utilizzati su un CLUSRCVR, tutti i canali mittente del cluster definiti automaticamente risultanti devono risiedere su piattaforme su cui il supporto SSL sottostante implementa questo CipherSpec e su cui è possibile specificarlo con il valore personalizzato. Se non è possibile selezionare un valore per il parametro SSLCIPH che verrà compreso in tutto il cluster, sarà necessaria un'uscita di definizione automatica del canale per modificarla in qualcosa che le piattaforme utilizzate comprenderanno. Utilizzare le stringhe di testo CipherSpec dove possibile (ad esempio RC4_MD5_US).

Un parametro SSLCRLNL si applica a un singolo gestore code e non viene propagato ad altri gestori code all'interno di un cluster.

Aggiornamento dei canali e dei gestori code in cluster a SSL

Aggiornare i canali cluster uno alla volta, modificando tutti i canali CLUSRCVR prima dei canali CLUSSDR .

Prima di iniziare

Considerare le seguenti considerazioni, poiché potrebbero influire sulla scelta di CipherSpec per un cluster:

- Alcuni CipherSpecs non sono disponibili su tutte le piattaforme. Scegliere una CipherSpec supportata da tutti i gestori code nel cluster.
- Alcuni CipherSpecs potrebbero essere nuovi nella release corrente di WebSphere MQ e non sono supportati nelle release precedenti. Un cluster che contiene gestori code in esecuzione in release differenti di MQ è in grado di utilizzare solo i CipherSpecs supportati da ciascuna release.

  Per utilizzare un nuovo CipherSpec all'interno di un cluster, è necessario prima migrare tutti i gestori code del cluster alla versione corrente.
- Alcuni CipherSpecs richiedono un tipo specifico di certificato digitale da utilizzare, in particolare quelli che utilizzano Elliptic Curve Cryptography.

Aggiornare tutti i gestori code nel cluster a WebSphere MQ V6 o superiore, se non sono già a questi livelli. Distribuire i certificati e le chiavi in modo che SSL funzioni da ciascuno di essi.

Informazioni su questa attività

Modificare un CLUSRCVR alla volta e consentire il flusso delle modifiche nel cluster prima di modificare il successivo. Assicurarsi di non modificare il percorso inverso fino a quando le modifiche per il canale corrente non sono state distribuite in tutto il cluster.

Procedura

1. Passare i canali CLUSRCVR a SSL in qualsiasi ordine.

   Le modifiche fluiscono nella direzione opposta sui canali che non vengono modificati in SSL.
2. Commutare tutti i canali CLUSSDR manuali in SSL.

   Ciò non ha alcun effetto sul funzionamento del cluster, a meno che non si utilizzi il comando REFRESH CLUSTER con l'opzione REPOS(YES).

   Nota: Per i cluster di grandi dimensioni, l'utilizzo del comando **REFRESH CLUSTER** può danneggiare il cluster mentre è in esecuzione e, di nuovo, a intervalli di 27 giorni, quando gli oggetti del cluster inviano automaticamente gli aggiornamenti di stato a tutti i gestori code interessati. Consultare Refreshing in a large cluster can affect performance and availability of the cluster.

Concetti correlati

“Specifica di CipherSpecs” a pagina 218
Specificare una CipherSpec utilizzando il parametro **SSLCIPH** nel comando MQSC **DEFINE CHANNEL** o nel comando MQSC **ALTER CHANNEL** .

“Certificati digitali e compatibilità CipherSpec in IBM WebSphere MQ” a pagina 34
Questo argomento fornisce informazioni su come scegliere i CipherSpecs e i certificati digitali appropriati per la politica di sicurezza, evidenziando la relazione tra CipherSpecs e i certificati digitali in IBM WebSphere MQ.

Informazioni correlate

Cluster: utilizzo delle procedure consigliate per REFRESH CLUSTER

Disabilitazione di SSL o TLS su canali e gestori code con cluster

Per disattivare SSL o TLS, impostare il parametro SSLCIPH su ' '. Disabilitare il TLS sui canali cluster singolarmente, modificando tutti i canali riceventi del cluster prima dei canali mittente del cluster.

Informazioni su questa attività

Modificare un canale ricevente del cluster alla volta e consentire il flusso delle modifiche nel cluster prima di modificare il successivo.

Importante: Assicurarsi di non modificare il percorso inverso fino a quando le modifiche per il canale corrente non sono state distribuite in tutto il cluster.

Procedura

1. Impostare il valore del parametro SSLCIPH su `' '`, una stringa vuota racchiusa tra virgolette singole .
   È possibile disattivare SSL o TLS sui canali riceventi del cluster in qualsiasi ordine desiderato.

   Nota che le modifiche fluiscono nella direzione opposta sui canali su cui lasci SSL o TLS attivi.
2. Verificare che il nuovo valore si rifletta in tutti i gestori code utilizzando il comando **DISPLAY CLUSQMGR(*)** ALL.
3. Disattivare SSL o TLS su tutti i canali mittenti del cluster manuali.
   Ciò non ha alcun effetto sul funzionamento del cluster, a meno che non si utilizzi il comando **REFRESH CLUSTER** con l'opzione REPOS （YES）.

   Per i cluster di grandi dimensioni, l'utilizzo del comando **REFRESH CLUSTER** può essere disruttivo per il cluster mentre è in corso e di nuovo a intervalli regolari in seguito, quando gli oggetti cluster inviano automaticamente gli aggiornamenti di stato a tutti i gestori code interessati. Consultare L'aggiornamento in un cluster di grandi dimensioni può influire sulle prestazioni e sulla disponibilità del cluster per ulteriori informazioni.
4. Arrestare e riavviare i canali mittente del cluster.

Sicurezza di pubblicazione/sottoscrizione

I componenti e le interazioni coinvolti nella pubblicazione / sottoscrizione sono descritti come un'introduzione alle spiegazioni e agli esempi più dettagliati che seguono.

Esistono diversi componenti coinvolti nella pubblicazione e sottoscrizione di un argomento. Alcune delle relazioni di sicurezza tra di loro sono illustrate in Figura 26 a pagina 252 e descritte nel seguente esempio.

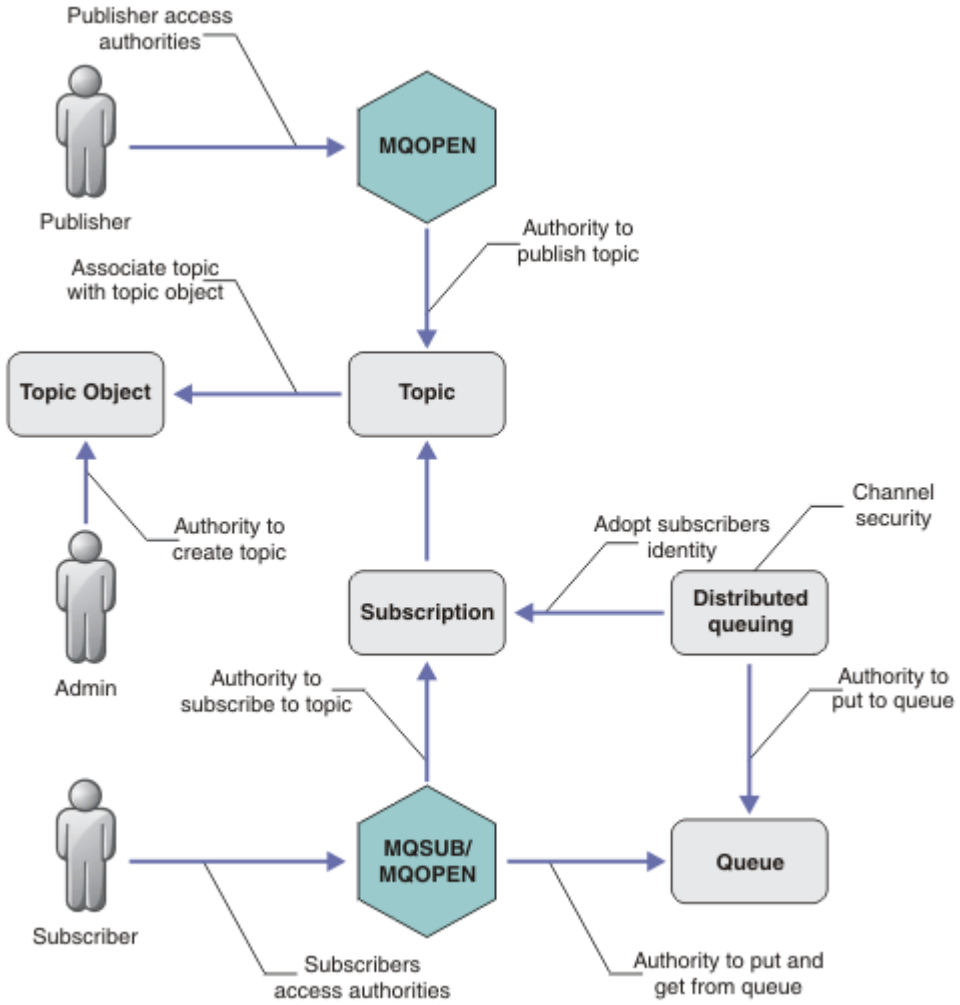


Figura 26. Relazioni di sicurezza di pubblicazione / sottoscrizione

Argomenti

Gli argomenti sono identificati da stringhe di argomento e sono generalmente organizzati in strutture ad albero, consultare Alberi degli argomenti. È necessario associare un argomento a un oggetto argomento per controllare l'accesso all'argomento. “Modello di sicurezza argomento” a pagina 254 spiega come proteggere gli argomenti utilizzando gli oggetti argomento.

Oggetti argomento di gestione

È possibile controllare chi ha accesso a un argomento e per quale scopo, utilizzando il comando **setmqaut** con un elenco di oggetti argomento di gestione. Consultare gli esempi, “Concedi accesso a un utente per sottoscrivere un argomento” a pagina 259 e “Concedi l'accesso a un utente per la pubblicazione in un argomento” a pagina 265.

Sottoscrizione

Sottoscrivere uno o più argomenti creando una sottoscrizione che fornisce una stringa di argomenti, che può includere caratteri jolly, da confrontare con le stringhe di argomenti delle pubblicazioni. Per ulteriori dettagli, consultare:

Sottoscrivi utilizzando un oggetto argomento

“Sottoscrizione utilizzando il nome oggetto argomento” a pagina 255

Sottoscrivi utilizzando un argomento

“Sottoscrizione utilizzando una stringa di argomenti in cui il nodo di argomenti non esiste” a pagina 256

Sottoscrivi utilizzando un argomento con caratteri jolly
"Sottoscrizione utilizzando una stringa di argomenti che contiene caratteri jolly" a pagina 257

Una sottoscrizione contiene informazioni sull'identità del sottoscrittore e sull'identità della coda di destinazione in cui devono essere inserite le pubblicazioni. Contiene inoltre informazioni su come la pubblicazione deve essere posizionata nella coda di destinazione.

Oltre a definire quali sottoscrittori hanno l'autorizzazione per sottoscrivere determinati argomenti, è possibile limitare l'utilizzo delle sottoscrizioni da parte di un singolo sottoscrittore. È inoltre possibile controllare quali informazioni sul sottoscrittore vengono utilizzate dal gestore code quando le pubblicazioni vengono inserite nella coda di destinazione. Consultare "Sicurezza sottoscrizione" a pagina 269.

Code

La coda di destinazione è una coda importante da proteggere. È locale per il sottoscrittore e le pubblicazioni che corrispondono alla sottoscrizione vengono inserite su di esso. È necessario considerare l'accesso alla coda di destinazione da due prospettive:

1. Inserimento di una pubblicazione sulla coda di destinazione.
2. Richiamo della pubblicazione dalla coda di destinazione.

Il gestore code inserisce una pubblicazione nella coda di destinazione utilizzando un'identità fornita dal sottoscrittore. Il sottoscrittore o un programma a cui è stata delegata l'attività di richiamo delle pubblicazioni, toglie i messaggi dalla coda. Consultare "Autorizzazione alle code di destinazione" a pagina 257.

Non sono presenti alias di oggetti argomento, ma è possibile utilizzare una coda alias come alias per un oggetto argomento. In questo caso, oltre a controllare l'autorizzazione per utilizzare l'argomento per la pubblicazione o la sottoscrizione, il gestore code controlla l'autorizzazione per utilizzare la coda.

Sicurezza di pubblicazione / sottoscrizione tra gestori code

L'autorizzazione alla pubblicazione o alla sottoscrizione di un argomento viene controllata sul gestore code locale utilizzando le identità e autorizzazioni locali. L'autorizzazione non dipende dal fatto che l'argomento sia definito o meno, né dal punto in cui è definito. Di conseguenza, è necessario eseguire l'autorizzazione dell'argomento su ogni gestore code in un cluster quando vengono utilizzati argomenti in cluster.

Nota: Il modello di sicurezza per gli argomenti differisce dal modello di sicurezza per le code. È possibile ottenere lo stesso risultato per le code definendo un alias della coda localmente per ogni coda cluster.

I gestori code si scambiano le sottoscrizioni in un cluster. Nella maggior parte delle configurazioni cluster di WebSphere MQ , i canali sono configurati con PUTAUT=DEF per posizionare i messaggi nelle code di destinazione utilizzando l'autorizzazione del processo del canale. È possibile modificare la configurazione del canale per utilizzare PUTAUT=CTX per richiedere all'utente sottoscrittore di disporre dell'autorizzazione per propagare una sottoscrizione su un altro gestore code in un cluster.

La sezione Sicurezza di pubblicazione / sottoscrizione tra gestori code descrive come modificare le definizioni dei canali per controllare a chi è consentito propagare le sottoscrizioni su altri server nel cluster.

Authorization

È possibile applicare l'autorizzazione agli oggetti argomento, come le code e altri oggetti. Esistono tre operazioni di autorizzazione, pub, sube `resume` che è possibile applicare solo agli argomenti. I dettagli sono descritti in Specifica delle autorizzazioni per i diversi tipi di oggetto.

Chiamate della funzione

Nei programmi di pubblicazione e sottoscrizione, come nei programmi in coda, i controlli di autorizzazione vengono eseguiti quando gli oggetti vengono aperti, creati, modificati o eliminati. I controlli non vengono eseguiti quando vengono effettuate chiamate MQI `MQPUT` o `MQGET` per inserire e ottenere pubblicazioni.

Per pubblicare un argomento, eseguire un MQOPEN sull'argomento, che esegue i controlli di autorizzazione. Pubblicare i messaggi nella gestione argomenti utilizzando il comando MQPUT , che non esegue alcun controllo di autorizzazione.

Per sottoscrivere un argomento, in genere si esegue un comando MQSUB per creare o riprendere la sottoscrizione e anche per aprire la coda di destinazione per ricevere le pubblicazioni. In alternativa, eseguire un MQOPEN separato per aprire la coda di destinazione, quindi eseguire MQSUB per creare o riprendere la sottoscrizione.

Indipendentemente dalle chiamate utilizzate, il gestore code verifica che sia possibile sottoscrivere l'argomento e ottenere le pubblicazioni risultanti dalla coda di destinazione. Se la coda di destinazione non è gestita, vengono eseguiti anche controlli di autorizzazione che il gestore code è in grado di inserire le pubblicazioni nella coda di destinazione. Utilizza l'identità che ha adottato da una sottoscrizione corrispondente. Si presuppone che il gestore code sia sempre in grado di inserire le pubblicazioni nelle code di destinazione gestite.

Ruoli

Gli utenti sono coinvolti in quattro ruoli nell'esecuzione delle applicazioni di pubblicazione / sottoscrizione:

1. Publisher
2. Abbonato
3. Amministratore argomenti
4. WebSphere MQ Administrator - membro del gruppo mqm

Definire i gruppi con autorizzazioni appropriate corrispondenti ai ruoli di pubblicazione, sottoscrizione e gestione argomenti. È quindi possibile assegnare i principal a questi gruppi autorizzandoli ad eseguire attività di pubblicazione e sottoscrizione specifiche.

Inoltre, è necessario estendere le autorizzazioni delle operazioni di amministrazione all'amministratore delle code e dei canali responsabili dello spostamento delle pubblicazioni e delle sottoscrizioni.

Modello di sicurezza argomento

Solo gli oggetti argomento definiti possono avere attributi di sicurezza associati. Per una descrizione degli oggetti argomento, consultare Oggetti argomento di gestione. Gli attributi di sicurezza specificano se a un ID utente o a un gruppo di sicurezza specificato è consentito eseguire un'operazione di sottoscrizione o pubblicazione su ciascun oggetto argomento.

Gli attributi di sicurezza sono associati al nodo di gestione appropriato nella struttura ad albero degli argomenti. Quando viene effettuato un controllo di autorizzazione per un particolare ID utente durante un'operazione di sottoscrizione o di pubblicazione, l'autorizzazione concessa si basa sugli attributi di sicurezza del nodo della struttura ad albero dell'argomento associato.

Gli attributi di sicurezza sono un elenco di controllo accessi, che indica quale autorizzazione ha un determinato ID utente o gruppo di sicurezza del sistema operativo per l'oggetto argomento.

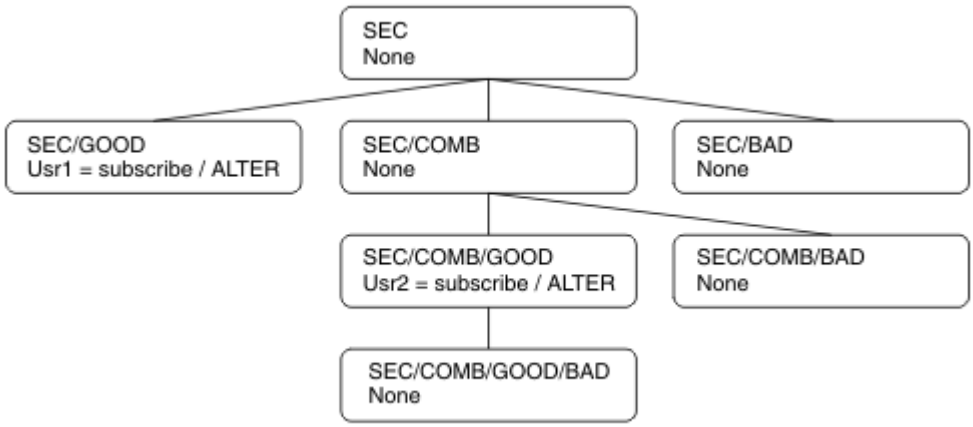
Considerare il seguente esempio in cui gli oggetti argomento sono stati definiti con gli attributi di sicurezza o le autorizzazioni mostrate:

Tabella 17. Autorizzazioni oggetto argomento di esempio

Nome argomento	**Stringa argomento**	**Autorizzazioni - non z/OS**	**Autorizzazioni z/OS**
SECROOT	SEC	Nessuno	Nessuno
SECGOOD	SEC/GOOD	usr1+subscribe	ALTER HLQ.SUBSCRIBE.SECGOOD

Tabella 17. Autorizzazioni oggetto argomento di esempio (Continua)			
Nome argomento	**Stringa argomento**	**Autorizzazioni - non z/OS**	**Autorizzazioni z/OS**
SECBAD	SEC/BAD	Nessuno	Nessuno HLQ.SUBSCRIBE.SECBAD
SECCOMB	SEC/COMB	Nessuno	Nessuno HLQ.SUBSCRIBE.SECCOMB
SECCOMBB	SEC/COMB/ GOOD/BAD	Nessuno	Nessuno HLQ.SUBSCRIBE.SECCOMBB
SECCOMBG	SEC/COMB/GOOD	usr2+subscribe	ALTER HLQ.SUBSCRIBE.SECCOMBG
SECCOMBN	SEC/COMB/BAD	Nessuno	Nessuno HLQ.SUBSCRIBE.SECCOMBN

La struttura ad albero degli argomenti con gli attributi di sicurezza associati a ciascun nodo può essere rappresentata come segue:



Gli esempi elencati forniscono le seguenti autorizzazioni:

- Sul nodo root dell'albero /SEC, nessun utente dispone dell'autorità su tale nodo.
- usr1 è stata concessa l'autorizzazione di sottoscrizione all'oggetto /SEC/GOOD
- usr2 è stata concessa l'autorizzazione di sottoscrizione all'oggetto /SEC/COMB/GOOD

Sottoscrizione utilizzando il nome oggetto argomento

Quando si sottoscrive un oggetto argomento specificando il nome MQCHAR48 , viene individuato il nodo corrispondente nella struttura ad albero dell'argomento. Se gli attributi di sicurezza associati al nodo indicano che l'utente dispone dell'autorizzazione per la sottoscrizione, l'accesso viene concesso.

Se all'utente non è concesso l'accesso, il nodo parent nella struttura ad albero determina se l'utente dispone dell'autorizzazione per la sottoscrizione a livello di nodo parent. In tal caso, viene concesso l'accesso. In caso contrario, viene considerato il parent di tale nodo. La ripetizione continua finché non viene individuato un nodo che concede l'autorizzazione di sottoscrizione all'utente. La ricorrenza si arresta quando il nodo root viene considerato senza che sia stata concessa l'autorizzazione. In quest' ultimo caso l'accesso è negato.

In breve, se un nodo nel percorso concede l'autorità di sottoscrizione a tale utente o applicazione, al sottoscrittore è consentito sottoscrivere su tale nodo o in un punto qualsiasi al di sotto di tale nodo nella struttura ad albero dell'argomento.

Il nodo root nell'esempio è SEC.

All'utente viene concessa l'autorizzazione di sottoscrizione se l'elenco di controllo accessi indica che l'ID utente stesso dispone dell'autorizzazione o che un gruppo di sicurezza del sistema operativo di cui l'ID utente è un membro dispone dell'autorizzazione.

Quindi, ad esempio:

- Se usr1 prova a sottoscrivere, utilizzando una stringa di argomenti di SEC/GOOD, la sottoscrizione sarà consentita poiché l'ID utente ha accesso al nodo associato a tale argomento. Tuttavia, se usr1 si tentasse di sottoscrivere utilizzando la stringa di argomenti SEC/COMB/GOOD la sottoscrizione non sarebbe consentita poiché l'ID utente non dispone dell'accesso al nodo associato.
- Se usr2 tenta di sottoscrivere, utilizzando una stringa di argomenti di SEC/COMB/GOOD , la sottoscrizione sarà consentita poiché l'ID utente ha accesso al nodo associato all'argomento. Tuttavia, se usr2 tentasse di sottoscrivere SEC/GOOD , la sottoscrizione non sarebbe consentita poiché l'ID utente non ha accesso al nodo associato.
- Se usr2 tenta di sottoscrivere utilizzando una stringa di argomenti di SEC/COMB/GOOD/BAD , la sottoscrizione sarà consentita perché l'ID utente ha accesso al nodo parent SEC/COMB/GOOD.
- Se usr1 o usr2 tenta di sottoscrivere utilizzando una stringa di argomenti di /SEC/COMB/BAD, non saranno consentiti né perché non hanno accesso al nodo di argomenti ad esso associato, né ai nodi parent di tale argomento.

Un'operazione di sottoscrizione che specifica il nome di un oggetto argomento che non esiste causa un errore MQRC_UNKNOWN_OBJECT_NAME.

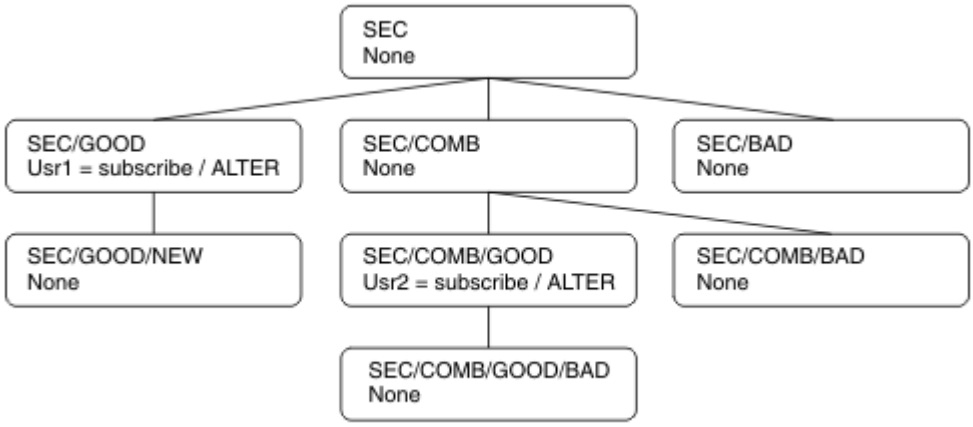
Sottoscrizione utilizzando una stringa di argomenti in cui esiste il nodo di argomenti

Il comportamento è lo stesso di quando si specifica l'argomento mediante il nome oggetto MQCHAR48 .

Sottoscrizione utilizzando una stringa di argomenti in cui il nodo di argomenti non esiste

Considerare il caso di un'applicazione che esegue la sottoscrizione, specificando una stringa di argomenti che rappresenta un nodo di argomenti che attualmente non esiste nella struttura di argomenti. Il controllo dell'autorità viene eseguito come descritto nella sezione precedente. Il controllo inizia con il nodo parent di quello rappresentato dalla stringa di argomenti. Se l'autorizzazione viene concessa, viene creato un nuovo nodo che rappresenta la stringa di argomenti nella struttura ad albero degli argomenti.

Ad esempio, usr1 tenta di sottoscrivere un argomento SEC/GOOD/NEW. L'autorizzazione viene concessa in quanto usr1 ha accesso al nodo parent SEC/GOOD. Viene creato un nuovo nodo di argomenti nella struttura ad albero, come mostrato nel seguente diagramma. Il nuovo nodo argomento non è un oggetto argomento a cui non è associato direttamente alcun attributo di sicurezza; gli attributi vengono ereditati dal relativo parent.



Sottoscrizione utilizzando una stringa di argomenti che contiene caratteri jolly

Considerare il caso della sottoscrizione utilizzando una stringa di argomenti che contiene un carattere jolly. Il controllo dell'autorizzazione viene effettuato sul nodo nella struttura ad albero dell'argomento che corrisponde alla parte completa della stringa dell'argomento.

Quindi, se un'applicazione si sottoscrive a SEC/COMB/GOOD/*, viene effettuato un controllo dell'autorizzazione come descritto nelle due sezioni precedenti sul nodo SEC/COMB/GOOD nella struttura ad albero degli argomenti.

Allo stesso modo, se un'applicazione deve sottoscrivere SEC/COMB/*/GOOD, viene eseguito un controllo dell'autorizzazione sul nodo SEC/COMB.

Autorizzazione alle code di destinazione

Quando si esegue la sottoscrizione a un argomento, uno dei parametri è la gestione hobj di una coda che è stata aperta per l'emissione per ricevere le pubblicazioni.

Se hobj non è specificato, ma è vuoto, viene creata una coda gestita se si applicano le seguenti condizioni:

- L'opzione MQSO_MANAGED è stata specificata.
- La sottoscrizione non esiste.
- La creazione è specificata.

Se hobj è vuoto e si sta modificando o ripristinando una sottoscrizione esistente, la coda di destinazione precedentemente fornita potrebbe essere gestita o non gestita.

L'applicazione o l'utente che effettua la richiesta MQSUB deve disporre dell'autorizzazione per inserire i messaggi nella coda di destinazione che ha fornito; in effetti, l'autorizzazione per pubblicare i messaggi su tale coda. Il controllo dell'autorità segue le regole esistenti per il controllo della sicurezza della coda.

Il controllo di sicurezza include l'ID utente alternativo e i controlli di sicurezza del contesto, se richiesti. Per poter impostare i campi del contesto di identità, è necessario specificare l'opzione MQSO_SET_IDENTITY_CONTEXT e l'opzione MQSO_CREATE o MQSO_ALTER . Non è possibile impostare nessuno dei campi del contesto di identità su una richiesta MQSO_RESUME .

Se la destinazione è una coda gestita, non viene eseguito alcun controllo di sicurezza sulla destinazione gestita. Se si è autorizzati a sottoscrivere un argomento, si presuppone che sia possibile utilizzare le destinazioni gestite.

Pubblicazione utilizzando il nome argomento o la stringa argomento in cui esiste il nodo argomento

Il modello di sicurezza per la pubblicazione è uguale a quello per la sottoscrizione, ad eccezione dei caratteri jolly. Le pubblicazioni non contengono caratteri jolly; pertanto, non esiste alcun caso di una stringa di argomenti contenente caratteri jolly da considerare.

Le autorizzazioni di pubblicazione e sottoscrizione sono distinte. Un utente o un gruppo può avere l'autorità di eseguire una operazione senza necessariamente essere in grado di eseguire l'altra.

Durante la pubblicazione in un oggetto argomento specificando il nome MQCHAR48 o la stringa di argomenti, viene individuato il nodo corrispondente nella struttura ad albero degli argomenti. Se gli attributi di sicurezza associati al nodo dell'argomento indicano che l'utente dispone dell'autorizzazione per la pubblicazione, l'accesso viene concesso.

Se l'accesso non viene concesso, il nodo principale nella struttura ad albero determina se l'utente dispone dell'autorizzazione per la pubblicazione a tale livello. In tal caso, viene concesso l'accesso. In caso contrario, la ripetizione continua finché non viene individuato un nodo che concede l'autorizzazione di pubblicazione all'utente. La ricorrenza si arresta quando il nodo root viene considerato senza che sia stata concessa l'autorizzazione. In quest' ultimo caso l'accesso è negato.

In breve, se un nodo nel percorso concede l'autorità di pubblicazione a tale utente o applicazione, il publisher è autorizzato a pubblicare in tale nodo o in qualsiasi punto al di sotto di tale nodo nella struttura ad albero degli argomenti.

Pubblicazione utilizzando il nome argomento o la stringa argomento in cui il nodo argomento non esiste

Come con l'operazione di sottoscrizione, quando un'applicazione pubblica, specificando una stringa di argomenti che rappresenta un nodo di argomenti che attualmente non esiste nella struttura ad albero degli argomenti, il controllo dell'autorizzazione viene eseguito a partire dall'elemento principale del nodo rappresentato dalla stringa di argomenti. Se l'autorizzazione viene concessa, viene creato un nuovo nodo che rappresenta la stringa di argomenti nella struttura ad albero degli argomenti.

Pubblicazione mediante una coda alias che si risolve in un oggetto argomento

Se si pubblica utilizzando una coda alias che si risolve in un oggetto argomento, il controllo di sicurezza si verifica sia sulla coda alias che sull'argomento sottostante in cui si risolve.

Il controllo di sicurezza sulla coda alias verifica che l'utente disponga dell'autorizzazione per inserire i messaggi su tale coda alias e il controllo di sicurezza sull'argomento verifica che l'utente possa pubblicare su tale argomento. Quando una coda alias viene risolta in un'altra coda, i controlli *non* vengono eseguiti sulla coda sottostante. Il controllo dell'autorizzazione viene eseguito in modo diverso per argomenti e code.

Chiusura di una sottoscrizione

Vi è un ulteriore controllo di sicurezza se si chiude una sottoscrizione utilizzando l'opzione MQCO_REMOVE_SUB se la sottoscrizione non è stata creata sotto questo handle.

Viene eseguito un controllo di sicurezza per assicurarsi di disporre dell'autorità corretta per eseguire questa operazione poiché l'azione risulta nella rimozione della sottoscrizione. Se gli attributi di sicurezza associati al nodo argomento indicano che l'utente dispone dell'autorizzazione, l'accesso viene concesso. In caso contrario, il nodo principale nella struttura ad albero viene considerato per stabilire se l'utente dispone dell'autorizzazione per chiudere la sottoscrizione. La ripetizione continua fino a quando non viene concessa l'autorizzazione o viene raggiunto il nodo root.

Definizione, modifica ed eliminazione di una sottoscrizione

Non viene eseguito alcun controllo di sicurezza della sottoscrizione quando una sottoscrizione viene creata amministrativamente, piuttosto che utilizzare una richiesta API MQSUB . Al responsabile è già stata concessa questa autorizzazione tramite il comando.

I controlli di sicurezza vengono eseguiti per garantire che le pubblicazioni possano essere inserite nella coda di destinazione associata alla sottoscrizione. I controlli vengono eseguiti come per una richiesta MQSUB .

L'ID utente utilizzato per questi controlli di sicurezza dipende dal comando immesso. Se il parametro **SUBUSER** viene specificato, influisce sul modo in cui viene eseguito il controllo, come mostrato in Tabella 18 a pagina 259:

Tabella 18. ID utente utilizzati per i controlli di sicurezza per i comandi

Comando	SUBUSER specificato e vuoto	SUBUSER specificato e completato	SUBUSER non specificato
	Utilizza l'ID amministratore		Utilizza l'ID amministratore
	Utilizza l'ID amministratore		Utilizza l'ID utente dalla sottoscrizione esistente

L'unico controllo di sicurezza eseguito quando si cancellano le sottoscrizioni utilizzando il comando DELETE SUB è il controllo di sicurezza del comando.

Impostazione della sicurezza di pubblicazione / sottoscrizione di esempio

Questa sezione descrive uno scenario che dispone di una configurazione del controllo accessi sugli argomenti in modo da consentire l'applicazione del controllo di sicurezza come richiesto.

Concedi accesso a un utente per sottoscrivere un argomento

Questo argomento è il primo di un elenco di attività che indica come concedere l'accesso agli argomenti a più di un utente.

Informazioni su questa attività

Questa attivit ... presuppone che non esistano oggetti argomento di gestione e che non sia stato definito alcun profilo per la sottoscrizione o la pubblicazione. Le applicazioni stanno creando nuove sottoscrizioni, piuttosto che riprendere quelle esistenti, e lo stanno facendo utilizzando solo la stringa argomento.

Un'applicazione può effettuare una sottoscrizione fornendo un oggetto argomento, una stringa argomento o una combinazione di entrambi. Qualunque sia il modo in cui l'applicazione seleziona, l'effetto è quello di effettuare una sottoscrizione in un determinato momento nella struttura ad albero degli argomenti. Se questo punto nella struttura ad albero degli argomenti è rappresentato da un oggetto argomento di gestione, viene controllato un profilo di sicurezza in base al nome di tale oggetto argomento.

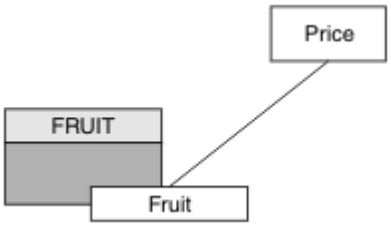


Figura 27. Esempio di accesso all'oggetto argomento

Tabella 19. Accesso oggetto argomento di esempio		
Argomento	**Accesso di sottoscrizione richiesto**	**Oggetto sezione**
Prezzo	Nessun utente	Nessuno
Prezzo / Frutta	USER1	frutta

Definire un nuovo oggetto argomento come segue:

Procedura

1. Immettere il comando MQSC `DEF TOPIC(FRUIT) TOPICSTR('Price/Fruit')`.
2. Concedere l'accesso come segue:
   - Altre piattaforme:

     Concedi l'accesso a USER1 per sottoscrivere l'argomento `"Price/Fruit"` concedendo all'utente l'accesso all'oggetto FRUIT . Eseguire questa operazione, utilizzando il comando di autorizzazione per la piattaforma:

     UNIX Linux Windows **Windows, sistemi UNIX and Linux**

     ```
     setmqaut -t topic -n FRUIT -p USER1 +sub
     ```

Risultati

Quando USER1 tenta di sottoscrivere l'argomento `"Price/Fruit"` , il risultato è positivo.

Quando USER2 tenta di sottoscrivere l'argomento `"Price/Fruit"` il risultato è un errore con un messaggio `MQRC_NOT_AUTHORIZED` , insieme a:

- UNIX Linux Windows Su altre piattaforme, il seguente evento di autorizzazione:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit"
```

Si noti che questa è un'illustrazione di ciò che si vede; non tutti i campi.

Concedere l'accesso a un utente per sottoscrivere un argomento più in profondità all'interno della struttura ad albero

Questo argomento è il secondo di un elenco di attività che indica come concedere l'accesso agli argomenti da più di un utente.

Prima di iniziare

Questo argomento utilizza la configurazione descritta in “Concedi accesso a un utente per sottoscrivere un argomento” a pagina 259.

Informazioni su questa attività

Se il punto nella struttura ad albero degli argomenti in cui l'applicazione effettua la sottoscrizione non è rappresentato da un oggetto argomento di gestione, spostare la struttura ad albero verso l'alto fino a quando non si trova l'oggetto argomento di gestione principale più vicino. Il profilo di sicurezza viene controllato, in base al nome dell'oggetto argomento.

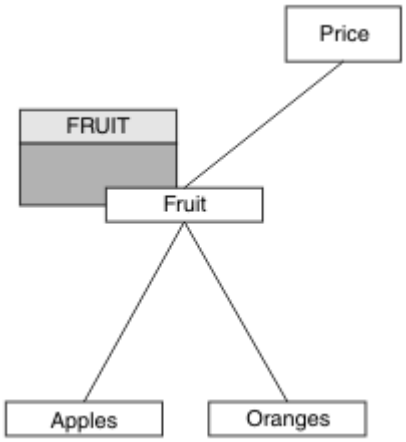


Figura 28. Esempio di concessione dell'accesso a un argomento all'interno di una struttura ad albero degli argomenti

Tabella 20. Requisiti di accesso per argomenti e oggetti argomento di esempio

Argomento	Accesso di sottoscrizione richiesto	Oggetto sezione
Prezzo	Nessun utente	Nessuno
Prezzo / Frutta	USER1	frutta
Prezzo / Frutta / Mele	USER1	
Prezzo / Frutta / Arance	USER1	

Nell'attività precedente USER1 era stato concesso l'accesso per la sottoscrizione all'argomento `"Price/Fruit"` concedendo l'accesso al profilo `hlq.SUBSCRIBE.FRUIT` in z/OS e l'accesso per la sottoscrizione al profilo FRUIT su altre piattaforme. Questo singolo profilo concede anche l'accesso USER1 per sottoscrivere `"Price/Fruit/Apples"`, `"Price/Fruit/Oranges"` e `"Price/Fruit/#"`.

Quando USER1 tenta di sottoscrivere l'argomento `"Price/Fruit/Apples"` , il risultato è positivo.

Quando USER2 tenta di sottoscrivere l'argomento `"Price/Fruit/Apples"` il risultato è un errore con un messaggio `MQRC_NOT_AUTHORIZED` , insieme a:

- Su z/OS, i seguenti messaggi visualizzati sulla console che mostrano il percorso di sicurezza completo attraverso la struttura ad albero degli argomenti tentata:

```
ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.FRUIT ...

ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.SYSTEM.BASE.TOPIC ...
```

- Su altre piattaforme, il seguente evento di autorizzazione:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit/Apples"
```

Tieni presente quanto segue:

- I messaggi ricevuti su z/OS sono identici a quelli ricevuti nell'attività precedente poiché gli stessi oggetti argomento e profili controllano l'accesso.
- Il messaggio di evento ricevuto su altre piattaforme è simile a quello ricevuto nell'attività precedente, ma la stringa di argomenti effettiva è diversa.

Concedi a un altro utente l'accesso per sottoscrivere solo l'argomento più profondo all'interno della struttura ad albero

Questo argomento è il terzo di un elenco di attività che indica come concedere l'accesso per la sottoscrizione agli argomenti da parte di più di un utente.

Prima di iniziare

Questo argomento utilizza la configurazione descritta in “Concedere l'accesso a un utente per sottoscrivere un argomento più in profondità all'interno della struttura ad albero” a pagina 260.

Informazioni su questa attività

Nell'attività precedente USER2 è stato rifiutato l'accesso all'argomento `"Price/Fruit/Apples"`. Questo argomento indica come concedere l'accesso a tale argomento, ma non ad altri argomenti.

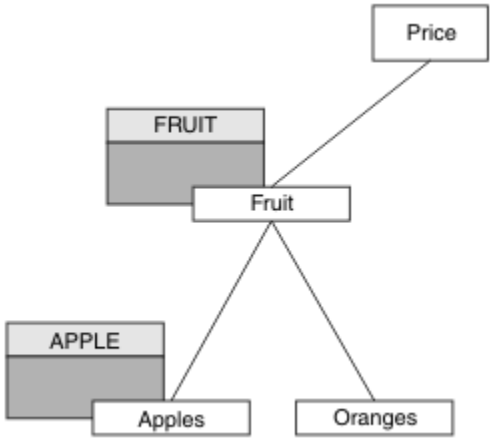


Figura 29. Concessione dell'accesso a specifici argomenti all'interno di una struttura ad albero degli argomenti

Tabella 21. Requisiti di accesso per argomenti e oggetti argomento di esempio

Argomento	Accesso di sottoscrizione richiesto	Oggetto sezione
Prezzo	Nessun utente	Nessuno
Prezzo / Frutta	USER1	frutta
Prezzo / Frutta / Mele	USER1 e USER2	Apple

Tabella 21. Requisiti di accesso per argomenti e oggetti argomento di esempio (Continua)		
Argomento	**Accesso di sottoscrizione richiesto**	**Oggetto sezione**
Prezzo / Frutta / Arance	USER1	

Definire un nuovo oggetto argomento come segue:

Procedura

1. Immettere il comando MQSC `DEF TOPIC(APPLE) TOPICSTR('Price/Fruit/Apples')`.
2. Concedere l'accesso come segue:
   - Altre piattaforme:

     Nell'attività precedente USER1 è stato concesso l'accesso alla sottoscrizione all'argomento `"Price/Fruit/Apples"` concedendo all'utente l'accesso alla sottoscrizione al profilo FRUIT .

     Questo singolo profilo ha concesso anche l'accesso USER1 per sottoscrivere `"Price/Fruit/Oranges"` e `"Price/Fruit/#"`, e questo accesso rimane anche con l'aggiunta del nuovo oggetto argomento e dei profili associati ad esso.

     Concedere l'accesso a USER2 per sottoscrivere l'argomento `"Price/Fruit/Apples"` concedendo all'utente l'accesso di sottoscrizione al profilo APPLE . Eseguire questa operazione, utilizzando il comando di autorizzazione per la piattaforma:

     
     Windows, sistemi UNIX and Linux

     ```
     setmqaut -t topic -n APPLE -p USER2 +sub
     ```

Risultati

Su z/OS, quando USER1 tenta di sottoscrivere l'argomento `"Price/Fruit/Apples"` il primo controllo di sicurezza sul profilo `hlq.SUBSCRIBE.APPLE` non riesce, ma quando si sposta in alto la struttura ad albero il profilo `hlq.SUBSCRIBE.FRUIT` consente a USER1 di sottoscrivere, in modo che la sottoscrizione abbia esito positivo e non venga inviato alcun codice di ritorno alla chiamata MQSUB. Tuttavia, viene generato un messaggio RACF `ICH` per la prima verifica:

```
ICH408I USER(USER1   ) ...
  hlq.SUBSCRIBE.APPLE ...
```

Quando USER2 tenta di sottoscrivere l'argomento `"Price/Fruit/Apples"` il risultato è positivo perché il controllo di sicurezza supera il primo profilo.

Quando USER2 tenta di sottoscrivere l'argomento `"Price/Fruit/Oranges"` il risultato è un errore con un messaggio `MQRC_NOT_AUTHORIZED` , insieme a:

- 
  Sulle piattaforme Windows, UNIX e Linux , il seguente evento di autorizzazione:

  ```
  MQRC_NOT_AUTHORIZED
  ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
  UserIdentifier        USER2
  AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
  TopicString           "Price/Fruit/Oranges"
  ```

Lo svantaggio di questa configurazione è che, su z/OS, si ricevono ulteriori messaggi `ICH` sulla console. È possibile evitare questa situazione se si protegge la struttura ad albero degli argomenti in modo diverso.

Modificare il controllo di accesso per evitare ulteriori messaggi

Questo argomento è il quarto di un elenco di attività che indica come concedere l'accesso agli argomenti da più di un utente ed evitare ulteriori messaggi RACF ICH408I su z/OS.

Prima di iniziare

Questo argomento migliora la configurazione descritta in "Concedi a un altro utente l'accesso per sottoscrivere solo l'argomento più profondo all'interno della struttura ad albero" a pagina 262 in modo da evitare ulteriori messaggi di errore.

Informazioni su questa attività

Questo argomento indica come concedere l'accesso agli argomenti più in profondità nella struttura ad albero e come rimuovere l'accesso all'argomento più in basso nella struttura ad albero quando nessun utente lo richiede.

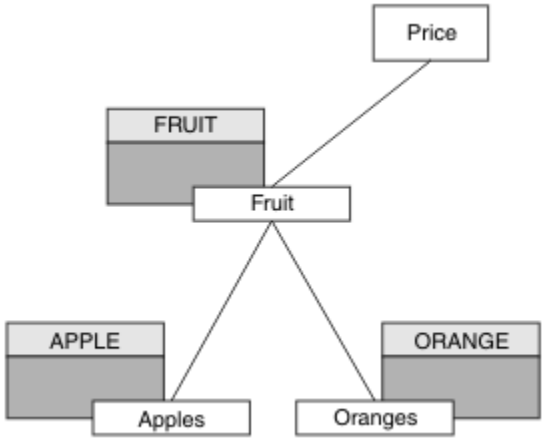


Figura 30. Esempio di concessione del controllo accessi per evitare ulteriori messaggi.

Definire un nuovo oggetto argomento come segue:

Procedura

1. Immettere il comando MQSC `DEF TOPIC(ORANGE) TOPICSTR('Price/Fruit/Oranges')`.
2. Concedere l'accesso come segue:
   - Altre piattaforme:

     Impostare l'accesso equivalente utilizzando i comandi di autorizzazione per la piattaforma:

     UNIX Linux Windows **Windows, sistemi UNIX and Linux**

     ```
     setmqaut -t topic -n ORANGE -p USER1 +sub
     setmqaut -t topic -n APPLE -p USER1 +sub
     ```

Risultati

Su z/OS, quando USER1 tenta di sottoscrivere l'argomento `"Price/Fruit/Apples"` il primo controllo di sicurezza sul profilo `hlq.SUBSCRIBE.APPLE` ha esito positivo.

Allo stesso modo, quando USER2 tenta di sottoscrivere un argomento `"Price/Fruit/Apples"` , il risultato è positivo perché il controllo di sicurezza supera il primo profilo.

Quando USER2 tenta di sottoscrivere l'argomento `"Price/Fruit/Oranges"` il risultato è un errore con un messaggio `MQRC_NOT_AUTHORIZED` , insieme a:

- UNIX Linux Windows Su altre piattaforme, il seguente evento di autorizzazione:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      ORANGE, FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit/Oranges"
```

Concedi l'accesso a un utente per la pubblicazione in un argomento

Questo argomento è il primo di un elenco di attività che indica come concedere l'accesso agli argomenti di pubblicazione a più di un utente.

Informazioni su questa attività

Questa attivit ... presuppone che non esistano oggetti argomento di gestione sul lato destro della struttura ad albero degli argomenti e che non siano stati definiti profili per la pubblicazione. L'ipotesi utilizzata è che i publisher stiano utilizzando solo la stringa di argomento.

Un'applicazione può pubblicare in un argomento fornendo un oggetto argomento, una stringa argomento o una combinazione di entrambi. Indipendentemente dal modo in cui viene selezionata l'applicazione, l'effetto è di pubblicare in un determinato momento nella struttura ad albero degli argomenti. Se questo punto nella struttura ad albero degli argomenti è rappresentato da un oggetto argomento di gestione, viene controllato un profilo di sicurezza in base al nome di tale oggetto argomento. Ad esempio:

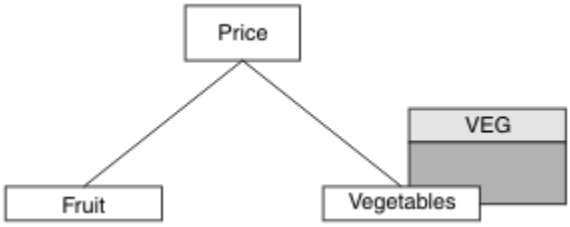


Figura 31. Concessione dell'accesso di pubblicazione a un argomento

Tabella 22. Requisiti di accesso alla pubblicazione di esempio

Argomento	È richiesto l'accesso alla pubblicazione	Oggetto sezione
Prezzo	Nessun utente	Nessuno
Prezzo / Verdure	USER1	VEG

Definire un nuovo oggetto argomento come segue:

Procedura

1. Immettere il comando MQSC `DEF TOPIC(VEG) TOPICSTR('Price/Vegetables')`.
2. Concedere l'accesso come segue:
   - Altre piattaforme:

     Concedere l'accesso a USER1 per pubblicare nell'argomento `"Price/Vegetables"` concedendo all'utente l'accesso al profilo VEG . Eseguire questa operazione, utilizzando il comando di autorizzazione per la piattaforma:

     Windows, sistemi UNIX and Linux

     ```
     setmqaut -t topic -n VEG -p USER1 +pub
     ```

Risultati

Quando USER1 tenta di pubblicare l'argomento `"Price/Vegetables"` , il risultato è corretto, ossia la chiamata MQOPEN ha esito positivo.

Quando USER2 tenta di pubblicare l'argomento `"Price/Vegetables"` la chiamata MQOPEN ha esito negativo con un messaggio `MQRC_NOT_AUTHORIZED` , insieme a:

- UNIX Linux Windows Su altre piattaforme, il seguente evento di autorizzazione:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      VEG, SYSTEM.BASE.TOPIC
TopicString           "Price/Vegetables"
```

Si noti che questa è un'illustrazione di ciò che si vede; non tutti i campi.

Concedere l'accesso a un utente per pubblicare un argomento più in profondità nella struttura ad albero

Questo argomento è il secondo di un elenco di attività che indica come concedere l'accesso alla pubblicazione degli argomenti a più di un utente.

Prima di iniziare

Questo argomento utilizza la configurazione descritta in “Concedi l'accesso a un utente per la pubblicazione in un argomento” a pagina 265.

Informazioni su questa attività

Se il punto nella struttura ad albero degli argomenti in cui l'applicazione pubblica non è rappresentato da un oggetto argomento di gestione, spostare la struttura ad albero verso l'alto fino a quando non si trova l'oggetto argomento di gestione principale più vicino. Il profilo di sicurezza viene controllato, in base al nome dell'oggetto argomento.

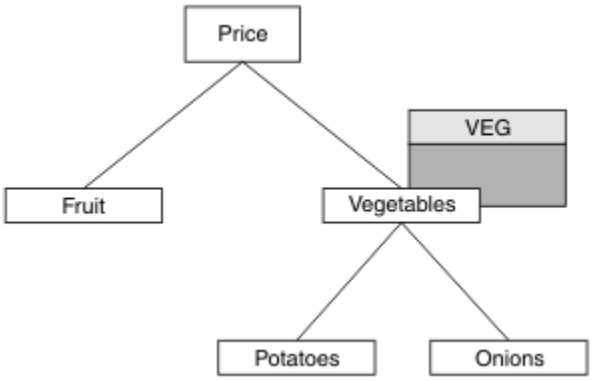


Figura 32. Concessione dell'accesso di pubblicazione a un argomento all'interno di una struttura ad albero degli argomenti

Tabella 23. Requisiti di accesso alla pubblicazione di esempio

Argomento	Accesso di sottoscrizione richiesto	Oggetto sezione
Prezzo	Nessun utente	Nessuno
Prezzo / Verdure	USER1	VEG

Tabella 23. Requisiti di accesso alla pubblicazione di esempio (Continua)		
Argomento	**Accesso di sottoscrizione richiesto**	**Oggetto sezione**
Prezzo / Verdura / Patate	USER1	
Prezzo / Verdure / Cipolle	USER1	

Nell'attività precedente, a USER1 è stato concesso l'accesso all'argomento di pubblicazione `"Price/Vegetables/Potatoes"` concedendo l'accesso al profilo `hlq.PUBLISH.VEG` su z/OS o l'accesso di pubblicazione al profilo VEG su altre piattaforme. Questo singolo profilo concede anche l' USER1 accesso alla pubblicazione in `"Price/Vegetables/Onions"`.

Quando USER1 tenta di pubblicare l'argomento `"Price/Vegetables/Potatoes"` il risultato è un esito positivo, ossia la chiamata MQOPEN ha esito positivo.

Quando USER2 tenta di sottoscrivere l'argomento `"Price/Vegetables/Potatoes"` il risultato è un errore; ovvero, la chiamata MQOPEN non riesce con un messaggio `MQRC_NOT_AUTHORIZED` , insieme a:

- Su z/OS, i seguenti messaggi visualizzati sulla console che mostrano il percorso di sicurezza completo attraverso la struttura ad albero degli argomenti tentata:

```
ICH408I USER(USER2   ) ...
  hlq.PUBLISH.VEG ...

ICH408I USER(USER2   ) ...
  hlq.PUBLISH.SYSTEM.BASE.TOPIC ...
```

- Su altre piattaforme, il seguente evento di autorizzazione:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      VEG, SYSTEM.BASE.TOPIC
TopicString           "Price/Vegetables/Potatoes"
```

Tieni presente quanto segue:

- I messaggi ricevuti su z/OS sono identici a quelli ricevuti nell'attività precedente poiché gli stessi oggetti argomento e profili controllano l'accesso.
- Il messaggio di evento ricevuto su altre piattaforme è simile a quello ricevuto nell'attività precedente, ma la stringa di argomenti effettiva è diversa.

Concedi accesso per pubblicazione e sottoscrizione

Questo argomento è l'ultimo di un elenco di attività che indica come concedere l'accesso alla pubblicazione e alla sottoscrizione di argomenti a più di un utente.

Prima di iniziare

Questo argomento utilizza la configurazione descritta in “Concedere l'accesso a un utente per pubblicare un argomento più in profondità nella struttura ad albero” a pagina 266.

Informazioni su questa attività

In un'attività precedente a USER1 è stato fornito l'accesso per la sottoscrizione all'argomento `"Price/Fruit"`. Questo argomento indica come concedere l'accesso a tale utente da pubblicare su tale argomento.

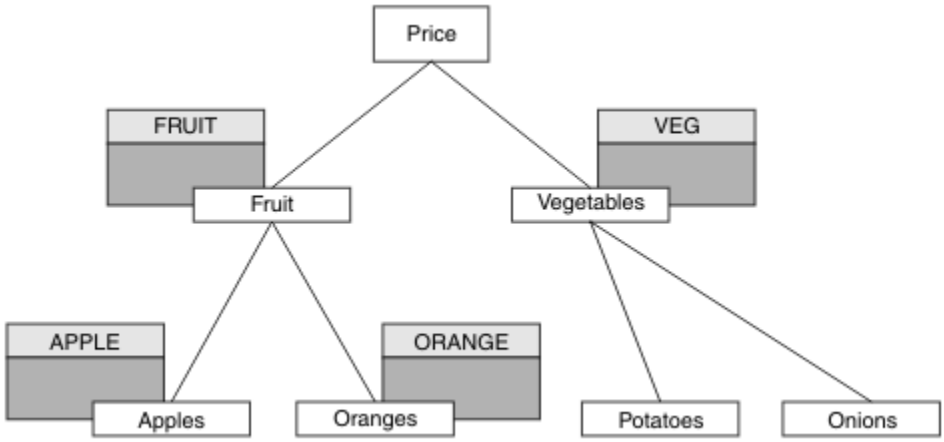


Figura 33. Concessione dell'accesso per la pubblicazione e la sottoscrizione

Tabella 24. Esempio di pubblicazione e sottoscrizione dei requisiti di accesso

Argomento	Accesso di sottoscrizione richiesto	È richiesto l'accesso alla pubblicazione	Oggetto sezione
Prezzo	Nessun utente	Nessun utente	Nessuno
Prezzo / Frutta	USER1	USER1	frutta
Prezzo / Frutta / Mele	USER1 e USER2		Apple
Prezzo / Frutta / Arance	USER1		ARANCIO

Procedura

Concedere l'accesso come segue:

- Altre piattaforme:

  Concedere l'accesso a USER1 per la pubblicazione nell'argomento `"Price/Fruit"` concedendo all'utente l'accesso di pubblicazione al profilo FRUIT . Eseguire questa operazione, utilizzando il comando di autorizzazione per la piattaforma:

  UNIX Linux Windows **Windows, sistemi UNIX and Linux**

  ```
  setmqaut -t topic -n FRUIT -p USER1 +pub
  ```

Risultati

Su z/OS, quando USER1 tenta di pubblicare l'argomento `"Price/Fruit"` il controllo di sicurezza sulla chiamata MQOPEN ha esito positivo.

Quando USER2 tenta di pubblicare nell'argomento `"Price/Fruit"` il risultato è un errore con un messaggio `MQRC_NOT_AUTHORIZED` , insieme a:

- UNIX Linux Windows Su piattaforme Windows, UNIX e Linux , il seguente evento di autorizzazione:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit"
```

Dopo la serie completa di queste attività, fornisce a USER1 e USER2 le seguenti autorizzazioni di accesso per la pubblicazione e la sottoscrizione agli argomenti elencati:

Tabella 25. Elenco completo delle autorizzazioni di accesso risultanti da esempi di sicurezza

Argomento	Accesso di sottoscrizione richiesto	È richiesto l'accesso alla pubblicazione	Oggetto sezione
Prezzo	Nessun utente	Nessun utente	Nessuno
Prezzo / Frutta	USER1	USER1	frutta
Prezzo / Frutta / Mele	USER1 e USER2		Apple
Prezzo / Frutta / Arance	USER1		ARANCIO
Prezzo / Verdure		USER1	VEG
Prezzo / Verdura / Patate			
Prezzo / Verdure / Cipolle			

Quando si hanno requisiti differenti per l'accesso alla sicurezza a livelli differenti all'interno della struttura ad albero degli argomenti, un'attenta pianificazione garantisce che non si ricevano avvertenze di sicurezza estranee nel log della console z/OS . L'impostazione della sicurezza al livello corretto all'interno della struttura ad albero evita messaggi di sicurezza fuorvianti.

Sicurezza sottoscrizione

MQSO_ALTERNATE_USER_AUTHORITY

Il campo ID AlternateUsercontiene un identificativo utente da utilizzare per convalidare questa chiamata MQSUB. La chiamata può riuscire solo se questo ID AlternateUserè autorizzato a sottoscrivere l'argomento con le opzioni di accesso specificate, indipendentemente dal fatto che l'identificativo utente con cui è in esecuzione l'applicazione sia autorizzato a farlo.

MQSO_SET_IDENTITY_CONTEXT

La sottoscrizione utilizza il token di account e i dati di identità dell'applicazione forniti nei campi PubAccountingToken e PubApplIdentityData .

Se viene specificata questa opzione, viene eseguito lo stesso controllo di autorizzazione come se si accedesse alla coda di destinazione utilizzando una chiamata MQOPEN con MQOO_SET_IDENTITY_CONTEXT, tranne nel caso in cui venga utilizzata anche l'opzione MQSO_MANAGED, nel qual caso non vi è alcun controllo di autorizzazione sulla coda di destinazione.

Se questa opzione non viene specificata, le pubblicazioni inviate a questo sottoscrittore hanno le informazioni di contesto predefinite associate come segue:

Tabella 26. Informazioni sul contesto di pubblicazione predefinito

Campo in MQMD	**Valore utilizzato**
UserIdentifier	L'ID utente associato alla sottoscrizione (vedere il campo SUBUSER su DISPLAY SBSTATUS) al momento della pubblicazione.
AccountingToken	Determinato dall'ambiente, se possibile; altrimenti, impostare su MQACT_NONE.
ApplIdentityData	Impostare su spazi vuoti.

Questa opzione è valida solo con MQSO_CREATE e MQSO_ALTER. Se utilizzato con MQSO_RESUME, i campi PubAccountingToken e PubApplIdentityData vengono ignorati, quindi questa opzione non ha alcun effetto.

Se una sottoscrizione viene modificata senza utilizzare questa opzione, dove in precedenza la sottoscrizione aveva fornito le informazioni sul contesto di identità, vengono generate le informazioni sul contesto predefinito per la sottoscrizione modificata.

Se una sottoscrizione che consente a ID utente differenti di utilizzarla con l'opzione MQSO_ANY_USERID, viene ripresa da un ID utente differente, viene generato il contesto di identità predefinito per il nuovo ID utente che ora possiede la sottoscrizione e vengono consegnate le pubblicazioni successive contenenti il nuovo contesto di identità.

AlternateSecurityId

Questo è un identificativo di sicurezza che viene passato con l'ID AlternateUseral servizio di autorizzazione per consentire l'esecuzione di controlli di autorizzazione appropriati. L'ID AlternateSecurityviene utilizzato solo se viene specificato MQSO_ALTERNATE_USER_AUTHORITY e il campo ID AlternateUsernon è completamente vuoto fino al primo carattere null o alla fine del campo.

opzione sottoscrizione MQSO_ANY_USERID

Quando viene specificato MQSO_ANY_USERID, l'identit ... del sottoscrittore non Š limitata a un singolo ID utente. Ciò consente a qualsiasi utente di modificare o riprendere la sottoscrizione quando dispone dell'autorizzazione appropriata. Solo un singolo utente può avere la sottoscrizione in qualsiasi momento. Un tentativo di riprendere l'utilizzo di una sottoscrizione attualmente in uso da parte di un'altra applicazione provocherà l'esito negativo della chiamata con MQRC_SUBSCRIPTION_IN_USE.

Per aggiungere questa opzione a una sottoscrizione esistente, la chiamata MQSUB (utilizzando MQSO_ALTER) deve provenire dallo stesso ID utente della sottoscrizione originale.

Se una chiamata MQSUB fa riferimento a una sottoscrizione esistente con MQSO_ANY_USERID impostato e l'ID utente differisce dalla sottoscrizione originale, la chiamata ha esito positivo solo se il nuovo ID utente dispone dell'autorizzazione per sottoscrivere l'argomento. Una volta completato correttamente, le pubblicazioni future per questo sottoscrittore vengono inserite nella coda del sottoscrittore con il nuovo ID utente impostato nella pubblicazione.

IDUSER_FIX_MQSO

Quando viene specificato MQSO_FIXED_USERID, la sottoscrizione può essere modificata o ripresa solo da un singolo ID utente proprietario. Questo ID utente è l'ultimo ID utente a modificare la sottoscrizione che ha impostato questa opzione, rimuovendo l'opzione MQSO_ANY_USERID oppure, se non sono state effettuate modifiche, è l'ID utente che ha creato la sottoscrizione.

Se un verbo MQSUB fa riferimento a una sottoscrizione esistente con MQSO_ANY_USERID impostato e modifica la sottoscrizione (utilizzando MQSO_ALTER) per utilizzare l'opzione MQSO_FIXED_USERID, l'ID utente della sottoscrizione è ora fisso su questo nuovo ID utente. La chiamata ha esito positivo solo se il nuovo ID utente dispone dell'autorizzazione per sottoscrivere l'argomento.

Se un ID utente diverso da quello registrato come proprietario di una sottoscrizione effettua la ripresa o la modifica di una sottoscrizione MQSO_FIXED_USERID, la chiamata avrà esito negativo con MQRC_IDENTITY_MISMATCH. L'ID utente proprietario di una sottoscrizione può essere visualizzato utilizzando il comando DISPLAY SBSTATUS.

Se non viene specificato né MQSO_ANY_USERID né MQSO_FIXED_USERID, il valore predefinito è MQSO_FIXED_USERID.

IBM WebSphere MQ Advanced Message Security

IBM WebSphere MQ Advanced Message Security (AMS) è un componente con licenza separata di IBM WebSphere MQ Advanced Message Security che fornisce un livello elevato di protezione per i dati sensibili che passano attraverso la rete IBM WebSphere MQ Advanced Message Security , senza influire sulle applicazioni finali.

IBM WebSphere MQ Advanced Message SecurityPanoramica

Le applicazioni IBM WebSphere MQ possono utilizzare IBM WebSphere MQ Advanced Message Security per inviare dati sensibili, come transazioni finanziarie di alto valore e informazioni personali, con diversi livelli di protezione utilizzando un modello crittografico a chiave pubblica.

Riferimenti correlati

Codici di ritorno GSKit utilizzati nei messaggi IBM WebSphere MQ AMS

Funzionamento che è stato modificato tra la versione 7.0.1 e la versione 7.5

Poiché IBM Advanced Message Security è diventato un componente in WebSphere MQ 7.5, alcuni aspetti della funzione IBM WebSphere MQ AMS sono stati modificati, ciò che potrebbe influenzare le applicazioni esistenti, gli script di gestione o le procedure di gestione.

Esaminare attentamente il seguente elenco di modifiche prima di eseguire l'aggiornamento dei gestori code alla versione 7.5. Decidere se è necessario pianificare le modifiche alle applicazioni, agli script e alle procedure esistenti prima di avviare la migrazione dei sistemi alla versione IBM WebSphere MQ 7.5:

- L'installazione di IBM WebSphere MQ AMS fa parte del processo di installazione di WebSphere MQ .
- Le funzionalità di sicurezza IBM WebSphere MQ AMS sono abilitate con la relativa installazione e controllate con le politiche di sicurezza. Non è necessario abilitare gli intercettatori per consentire a IBM WebSphere MQ AMS di avviare l'intercettazione dei dati.
- IBM WebSphere MQ AMS in WebSphere MQ versione 7.5 non richiede l'utilizzo del comandi **cfgmqs** come nella versione autonoma di IBM WebSphere MQ AMS.

Caratteristiche e funzioni di IBM WebSphere MQ Advanced Message Security

Advanced Message Security espande WebSphere MQ i servizi di sicurezza per fornire la firma e la codifica dei dati a livello di messaggio. I servizi espansi garantiscono che i dati del messaggio non siano stati modificati tra il momento in cui sono stati originariamente collocati su una coda e il momento in cui sono stati richiamati. Inoltre, IBM WebSphere MQ AMS verifica che un mittente dei dati del messaggio sia autorizzato a inserire i messaggi firmati su una coda di destinazione.

Di seguito è riportato un elenco completo delle funzioni IBM WebSphere MQ AMS :

- Protegge le transazioni sensibili o di valore elevato elaborate da WebSphere MQ.
- Rileva e rimuove i messaggi non autorizzati o non autorizzati prima che vengano elaborati da un'applicazione ricevente.
- Verifica che i messaggi non siano stati modificati durante il transito dalla coda alla coda.
- Protegge i dati non solo quando passano attraverso la rete ma anche quando vengono inseriti in una coda.
- Protegge le applicazioni proprietarie e scritte dal cliente esistenti per WebSphere MQ.

Gestione degli errori

Advanced Message Security definisce una coda di gestione degli errori per gestire i messaggi che contengono errori o i messaggi che non possono essere non protetti.

I messaggi difettosi sono trattati come casi eccezionali. Se un messaggio ricevuto non soddisfa i requisiti di sicurezza per la coda in cui si trova, ad esempio, se il messaggio è firmato quando deve essere codificato o se la decodifica o la verifica della firma non riesce, il messaggio viene inviato alla coda di gestione degli errori. Un messaggio potrebbe essere inviato alla coda di gestione degli errori per i seguenti motivi:

- Mancata corrispondenza della qualità della protezione - esiste una mancata corrispondenza della qualità della protezione (QOP) tra il messaggio ricevuto e la definizione QOP nella normativa di sicurezza.
- Errore di decodifica - non è possibile decodificare il messaggio.
- Errore intestazione PDMQ - impossibile accedere all'intestazione del messaggio AMS WebSphere MQ .
- Mancata corrispondenza della dimensione - la lunghezza di un messaggio dopo la decodifica è diversa da quella prevista.
- Mancata corrispondenza del livello dell'algoritmo di codifica - l'algoritmo di crittografia del messaggio è più debole di quanto richiesto.
- Errore sconosciuto - si è verificato un errore non previsto.

WebSphere MQ AMS utilizza SYSTEM.PROTECTION.ERROR.QUEUE come coda di gestione errori. Tutti i messaggi inseriti da IBM WebSphere MQ AMS nel SISTEMA SYSTEM.PROTECTION.ERROR.QUEUE sono precedute dall'intestazione MQDLH.

L'amministratore WebSphere MQ può definire anche il SISTEMA SYSTEM.PROTECTION.ERROR.QUEUE come coda alias che punta ad un'altra coda.

Concetti principali

Informazioni sui concetti chiave in Advanced Message Security per comprendere come funziona lo strumento e come gestirlo in modo efficace.

Infrastruttura chiave pubblica

PKI (Public Key Infrastructure) è un sistema di strutture, politiche e servizi che supporta l'utilizzo della crittografia a chiave pubblica per ottenere una comunicazione sicura.

Non esiste un singolo standard che definisce i componenti di una infrastruttura di chiavi pubbliche, ma un PKI in genere implica l'uso di certificati di chiavi pubbliche e comprende autorità di certificazione (CA) e altre autorità di registrazione (RA) che forniscono i seguenti servizi:

- Emissione di certificati digitali
- Convalida dei certificati digitali
- Revoca di certificati digitali
- Distribuzione dei certificati

L'identità degli utenti e delle applicazioni è rappresentata dal campo **DN (distinguished name)** in un certificato associato a messaggi firmati o codificati. Advanced Message Security utilizza questa identità per rappresentare un utente o un'applicazione. Per autenticare questa identità, l'utente o l'applicazione

deve avere accesso al keystore in cui sono memorizzati il certificato e la chiave privata associata. Ogni certificato è rappresentato da un'etichetta nel keystore.

Concetti correlati

“Utilizzo di keystore e certificati” a pagina 296
Per fornire una protezione crittografica trasparente per le applicazioni WebSphere MQ , Advanced Message Security utilizza il file keystore, in cui sono memorizzati i certificati della chiave pubblica e una chiave privata.

Certificati digitali

Advanced Message Security associa utenti e applicazioni ai certificati digitali standard X.509 . I certificati X.509 sono generalmente firmati da una CA (Certificate Authority) attendibile e implicano chiavi pubbliche e private utilizzate per la codifica e la decodifica.

I certificati digitali forniscono protezione contro l'impersonificazione collegando una chiave pubblica al suo proprietario, se tale proprietario è un individuo, un gestore code o un'altra entità. I certificati digitali sono anche noti come certificati di chiave pubblica, perché ti garantiscono la proprietà di una chiave pubblica quando utilizzi uno schema di chiave asimmetrica. Questo schema richiede la generazione di una chiave pubblica e di una chiave privata per un'applicazione. I dati codificati con la chiave pubblica possono essere decodificati solo utilizzando la corrispondente chiave privata mentre i dati codificati con la chiave privata possono essere decodificati solo utilizzando la chiave pubblica corrispondente. La chiave privata è memorizzata in un file database di chiavi protetto da password. Solo il proprietario ha accesso alla chiave privata utilizzata per decodificare i messaggi codificati utilizzando la chiave pubblica corrispondente.

Se le chiavi pubbliche vengono inviate direttamente dal loro proprietario a un'altra entità, c'è il rischio che il messaggio possa essere intercettato e la chiave pubblica sostituita da un'altra. Questo è conosciuto come un attacco "man - in - the - middle". La soluzione è scambiare chiavi pubbliche tramite una terza parte attendibile, dando all'utente una forte garanzia che la chiave pubblica appartiene all'entità con cui si sta comunicando. Invece di inviare la tua chiave pubblica direttamente, chiedi a una terza parte attendibile di incorporarla in un certificato digitale. La terza parte attendibile che emette certificati digitali è denominata CA (Certificate Authority).

Per ulteriori informazioni sui certificati digitali, consultare Cosa si trova in un certificato digitale.

Un certificato digitale contiene la chiave pubblica per un'entità e indica che la chiave pubblica appartiene a tale entità:

- quando un certificato è per una singola entità, viene denominato *certificato personale* o *certificato utente*.
- quando un certificato è per un'autorità di certificazione, il certificato viene denominato *certificato CA* o *certificato del firmatario*.

Nota: Advanced Message Security supporta i certificati autofirmati nelle applicazioni Java e native

Concetti correlati

“Crittografia” a pagina 7
La crittografia è il processo di conversione tra un testo leggibile, denominato *testo semplice*, e un formato illeggibile, denominato *testo crittografico*.

Gestore Autorizzazione Oggetto

OAM (Object Authority Manager) è il componente del servizio di autorizzazione fornito con i prodotti WebSphere MQ .

L'accesso alle entità Advanced Message Security è controllato tramite i gruppi di utenti WebSphere MQ e OAM. Gli amministratori possono utilizzare la CLI (command - line interface) per concedere o revocare le autorizzazioni come richiesto. Diversi gruppi di utenti possono avere diversi tipi di autorizzazione di accesso agli stessi oggetti. Ad esempio, un gruppo può eseguire operazioni PUT e GET per una coda specifica mentre un altro gruppo può solo sfogliare la coda. Allo stesso modo, alcuni gruppi potrebbero avere l'autorizzazione GET e PUT per una coda, ma non sono autorizzati a modificare o eliminare la coda.

Attraverso l'OAM, è possibile controllare:

- Accesso agli oggetti Advanced Message Security tramite MQI. Quando un programma applicativo tenta di accedere agli oggetti, OAM verifica se il profilo utente che effettua la richiesta dispone dell'autorizzazione per l'operazione richiesta. Ciò significa che le code e i messaggi sulle code possono essere protetti da accessi non autorizzati.
- Autorizzazione per utilizzare comandi PCF e MQSC.

Concetti correlati

Gestore Autorizzazione Oggetto

Tecnologia supportata

Advanced Message Security dipende da diversi componenti tecnologici per fornire un'infrastruttura di sicurezza.

Advanced Message Security supporta le seguenti API (application programming interface) WebSphere MQ :

- Message Queue Interface (MQI)
- WebSphere MQ JMS (Java Message Service) 1.0.2 e 1.1.
- Classi di base WebSphere MQ per Java
- Classi WebSphere MQ per. Net in modalità non gestita

Nota: Advanced Message Security supporta le autorità di certificazione X.509 .

Limitazioni note

Informazioni sulle limitazioni di IBM WebSphere MQ Advanced Message Security.

- Le seguenti opzioni IBM WebSphere MQ non sono supportate:
  - Pubblicazione / sottoscrizione.
  - Conversione dati del canale.
  - Elenchi di distribuzione.
  - Segmentazione del messaggio dell'applicazione
  - L'utilizzo di applicazioni non in thread utilizzando l'uscita API su piattaforme HP-UX .
  - Classi IBM WebSphere MQ per .NET in una modalità gestita (connessioni client o bind).
  - Applicazioni del client Message Service per .NET (XMS).
  - Client del servizio messaggi per applicazioni C/C++ (XMS supportPac IA94).
- Tutte le applicazioni Java dipendono da IBM Java Runtime.

  IBM WebSphere MQ Advanced Message Security non supporta il JRE fornito da altri fornitori.
- Applicazioni client JMS e Java che utilizzano IBM WebSphere MQ Advanced Message Security in modalità client.

  Qualsiasi applicazione client JMS o Java(inclusi gli agent IBM WebSphere MQ Explorer e IBM WebSphere MQ Managed File Transfer) non può utilizzare IBM WebSphere MQ Advanced Message Security in modalità client con un gestore code WebSphere MQ precedente a Version 7.5.

  Per utilizzare le politiche di protezione dei messaggi, queste applicazioni devono interagire con un gestore code IBM WebSphere MQ Version 7.5 o connettersi in modalità di bind locale a un gestore code sulla stessa macchina dell'applicazione.
- Si consiglia di evitare di inserire due o più certificati con gli stessi DN (Distinguished Name), in un singolo file keystore, poiché il funzionamento dell'intereceptor IBM WebSphere MQ Advanced Message Security con tali certificati non è definito.
- L'adattatore di risorse IBM WebSphere MQ Version 7.5 non supporta IBM WebSphere MQ Advanced Message Security. Se è necessario utilizzare la protezione dei messaggi con le applicazioni IBM WebSphere MQ classes for JMS o IBM WebSphere MQ classes for Java in esecuzione in un ambiente del server delle applicazioni:

– Il server delle applicazioni deve essere configurato per utilizzare l'adattatore di risorse Version 8.0 o successivo.
– Oppure è necessario utilizzare l'intercettazione MCA (Message Channel Agent).

Scenari utente

Familiarizzare con gli scenari possibili per comprendere quali obiettivi di business è possibile raggiungere con Advanced Message Security.

Guida rapida per piattaforme Windows

Utilizzare questa guida per configurare rapidamente IBM Advanced Message Security per fornire la sicurezza dei messaggi sulle piattaforme Windows . Al momento del completamento, sarà stato creato un database di chiavi per verificare le identità utente e le politiche di firma / crittografia definite per il gestore code.

Prima di iniziare

Sul sistema devono essere installate almeno le seguenti funzioni:

- Server
- Development Toolkit (per programmi di esempio)
- Advanced Message Security

Per i dettagli, fare riferimento alle funzioni IBM WebSphere MQ per i sistemi Windows .

Per informazioni sull'utilizzo del comando **setmqenv** per inizializzare l'ambiente corrente in modo che i comandi WebSphere MQ appropriati possano essere individuati ed eseguiti dal sistema operativo, consultare setmqenv.

1. Creazione di un gestore code e di una coda

Informazioni su questa attività

Tutti i seguenti esempi utilizzano una coda denominata TEST.Q per trasmettere i messaggi tra le applicazioni. Advanced Message Security utilizza gli intercettatori per firmare e codificare i messaggi nel punto in cui entrano nell'infrastruttura WebSphere MQ tramite l'interfaccia WebSphere MQ standard. La configurazione di base viene eseguita in WebSphere MQ e viene configurata nei seguenti passi.

È possibile utilizzare WebSphere MQ Explorer per creare il gestore code QM_VERIFY_AMS e la relativa coda locale denominata TEST.Q utilizzando tutte le impostazioni predefinite della procedura guidata oppure è possibile utilizzare i comandi disponibili in \WebSphere MQ\bin. Tenere presente che è necessario essere un membro del gruppo di utenti mqm per eseguire i seguenti comandi di gestione.

Procedura

1. Creare un gestore code

```
crtmqm QM_VERIFY_AMS
```

2. Avvia il gestore code

```
strmqm QM_VERIFY_AMS
```

3. Creare una coda denominata TEST.Q immettendo il comando seguente in **runmqsc** per il gestore code QM_VERIFY_AMS

```
DEFINE QLOCAL(TEST.Q)
```

Risultati

Se la procedura è stata completata, il comando immesso in **runmqsc** visualizzerà i dettagli su TEST.Q:

```
DISPLAY Q(TEST.Q)
```

2. Creazione e autorizzazione degli utenti

Informazioni su questa attività

In questo esempio vengono visualizzati due utenti: alice, il mittente e bob, il destinatario. Per utilizzare la coda dell'applicazione, a questi utenti deve essere concessa l'autorizzazione per utilizzarla. Inoltre, per utilizzare con successo le politiche di protezione che definiremo, a questi utenti deve essere concesso l'accesso ad alcune code di sistema. Per ulteriori informazioni sul comando **setmqaut** fare riferimento a **setmqaut** .

Procedura

1. Creare i due utenti e assicurarsi che HOMEPATH e HOMEDRIVE siano impostati per entrambi.
2. Autorizzare gli utenti a connettersi al gestore code e a lavorare con la coda

   ```
   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. È inoltre necessario consentire ai due utenti di esplorare la coda della normativa di sistema e inserire i messaggi nella coda di errore.

   ```
   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
   ```

Risultati

Gli utenti vengono ora creati e le autorizzazioni richieste vengono loro concesse.

Operazioni successive

Per verificare se le operazioni sono state eseguite correttamente, utilizzare gli esempi amqsput e amqsget come descritto nella sezione “7. Verifica della configurazione” a pagina 279.

3. Creazione di certificati e database di chiavi

Informazioni su questa attività

Interceptor richiede la chiave pubblica degli utenti di invio per codificare il messaggio. Pertanto, è necessario creare il database delle chiavi delle identità utente associate alle chiavi pubbliche e private. Nel sistema reale, dove gli utenti e le applicazioni sono distribuiti su diversi computer, ogni utente avrebbe il proprio keystore privato. Allo stesso modo, in questa guida, vengono creati database di chiavi per alice e bob e condivisi i certificati utente tra di loro.

Nota: In questa guida, vengono utilizzate le applicazioni di esempio scritte in C che si collegano utilizzando i bind locali. Se si intende utilizzare le applicazioni Java utilizzando i collegamenti client, è necessario creare un keystore JKS e i certificati utilizzando il comando **keytool** , che fa parte del JRE (consultare “Guida rapida per i clienti Java” a pagina 286 per ulteriori dettagli). Per tutti gli altri linguaggi e per le applicazioni Java che utilizzano i collegamenti locali, i passi in questa guida sono corretti.

Procedura

1. Utilizzare IBM Key Management GUI (strmqikm.exe) per creare un nuovo database delle chiavi per l'utente alice.

```
Type:       CMS
Filename:   alicekey.kdb
Location:   C:/Documents and Settings/alice/AMS
```

Nota:

- Si consiglia di utilizzare una password complessa per proteggere il database.
- Verificare che la check box **Stash password in un file** sia selezionata.

2. Modificare la vista del contenuto del database di chiavi in **Certificati personali**.
3. Selezionare **Nuovo autofirmato**; in questo scenario vengono utilizzati i certificati autofirmati.
4. Creare un certificato che identifica l'utente `alice` da utilizzare nella crittografia, utilizzando i seguenti campi:

```
Key label: Alice_Cert
Common Name: alice
Organisation: IBM
Country: GB
```

Nota:

- Ai fini di questa guida, utilizziamo un certificato autofirmato che può essere creato senza utilizzare un'autorità di certificazione. Per i sistemi di produzione, è consigliabile non utilizzare certificati autofirmati, ma affidarsi invece ai certificati firmati da una CA (Certificate Authority).
- Il parametro **`Key label`** specifica il nome per il certificato, che gli intercettatori ricercherà per ricevere le informazioni necessarie.
- I parametri **`Common Name`** e facoltativi specificano i dettagli del **DN (Distinguished Name)** , che deve essere univoco per ciascun utente.

5. Ripetere il passo 1-4 per l'utente bob

Risultati

I due utenti `alice` e bob hanno ora un certificato autofirmato.

4. Creazione di keystore.conf

Informazioni su questa attività

È necessario puntare gli intercettatori Advanced Message Security alla directory in cui si trovano located.This viene eseguita tramite il file `keystore.conf` , che contiene tali informazioni in formato di testo semplice. Ciascun utente deve avere un file `keystore.conf` separato. Questa operazione deve essere eseguita sia per `alice` che per bob.

Il contenuto di `keystore.conf` deve essere nel formato:

```
cms.keystore = <dir>/keystore_file
cms.certificate = certificate_label
```

Esempio

Per questo scenario, il contenuto di `keystore.conf` sarà il seguente:

```
cms.keystore = C:/Documents and Settings/alice/AMS/alicekey
cms.certificate = Alice_Cert
```

Nota:

- Il percorso del file keystore deve essere fornito senza estensione file.
- L'etichetta del certificato può includere spazi, quindi "Alice_Cert" e "Alice_Cert" ad esempio, sono riconosciuti come etichette di due certificati differenti. Tuttavia, per evitare confusione, è meglio non utilizzare spazi nel nome dell'etichetta.

- Esistono i seguenti formati di memorizzazione chiave: CMS (Cryptographic Message Syntax), JKS (Java Keystore) e JCEKS (Java Cryptographic Extension Keystore). Per ulteriori informazioni, fare riferimento a “Struttura del file di configurazione keystore (keystore.conf)” a pagina 296.
- *%HOMEDRIVE%\%HOMEPATH%*\.mqs\keystore.conf (ad es. C:\Documents and Settings\alice\.mqs\keystore.conf) è l'ubicazione predefinita in cui Advanced Message Security ricerca il file keystore.conf . Per informazioni su come utilizzare un'ubicazione non predefinita per keystore.conf, consultare “Utilizzo di keystore e certificati” a pagina 296.
- Per creare la directory .mqs , è necessario utilizzare il prompt dei comandi.

5. Condivisione dei certificati

Informazioni su questa attività

Condividere i certificati tra i due database di chiavi in modo che ogni utente possa identificare correttamente l'altro. Questa operazione viene eseguita estraendo il certificato pubblico di ciascun utente in un file, che viene quindi aggiunto al database delle chiavi dell'altro utente.

Nota: Utilizzare l'opzione *extract* e non l'opzione *export* . *Extract* ottiene la chiave pubblica dell'utente, mentre *export* ottiene sia la chiave pubblica che quella privata. L'utilizzo di *export* per errore comprometterebbe completamente la tua applicazione, passando la sua chiave privata.

Procedura

1. Estrarre il certificato che identifica alice in un file esterno:

```
runmqakm -cert -extract -db "C:/Documents and Settings/alice/AMS/alicekey.kdb" -pw passw0rd
-label Alice_Cert -target alice_public.arm
```

2. Aggiungere il certificato al keystore bob's :

```
runmqakm -cert -add -db "C:/Documents and Settings/bob/AMS/bobkey.kdb" -pw passw0rd -label
Alice_Cert -file alice_public.arm
```

3. Ripetere i passi per 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
```

Risultati

I due utenti alice e bob sono ora in grado di identificarsi correttamente l'uno con l'altro avendo creato e condiviso certificati autofirmati.

Operazioni successive

Verificare che un certificato si trovi nel keystore esplorandolo utilizzando la GUI o eseguendo i seguenti comandi che ne stampano i dettagli:

```
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. Definizione della politica della coda

Informazioni su questa attività

Con il gestore code creato e gli intercettatori preparati per intercettare i messaggi e accedere alle chiavi di crittografia, è possibile iniziare a definire le politiche di protezione su QM_VERIFY_AMS utilizzando il

comando `setmqspl` . Fare riferimento a setmqspl per ulteriori informazioni su questo comando. Ogni nome di politica deve essere uguale al nome della coda a cui deve essere applicato.

Esempio

Questo è un esempio di politica definita per la coda `TEST.Q` . Nell'esempio, i messaggi vengono firmati con l'algoritmo SHA1 e codificati con l'algoritmo AES256 . `alice` è l'unico mittente valido e bob è l'unico destinatario dei messaggi su questa coda:

```
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: I DN corrispondono esattamente a quelli specificati nel rispettivo certificato utente dal database delle chiavi.

Operazioni successive

Per verificare la politica definita, immettere il seguente comando:

```
dspmqspl -m QM_VERIFY_AMS
```

Per stampare i dettagli della politica come una serie di comandi `setmqspl` , l'indicatore `-export` . Ciò consente di memorizzare le politiche già definite:

```
dspmqspl -m QM_VERIFY_AMS -export >restore_my_policies.bat
```

7. Verifica della configurazione

Informazioni su questa attività

Eseguendo diversi programmi sotto utenti differenti è possibile verificare se l'applicazione è stata configurata correttamente.

Procedura

1. Cambia utente da eseguire come utente `alice`

   Fare clic con il pulsante destro del mouse su `cmd.exe` e selezionare **Esegui come ...**. Quando richiesto, accedere come utente `alice`.
2. Man mano che l'utente `alice` inserisce un messaggio utilizzando un'applicazione di esempio:

   ```
   amqsput TEST.Q QM_VERIFY_AMS
   ```
3. Immettere il testo del messaggio, quindi premere Invio.
4. Cambia utente da eseguire come utente bob

   Aprire un'altra finestra facendo clic con il pulsante destro del mouse su `cmd.exe` e selezionando **Esegui come ...**. Quando richiesto, accedere come utente bob.
5. Come l'utente Bob riceve un messaggio utilizzando un'applicazione di esempio:

   ```
   amqsget TEST.Q QM_VERIFY_AMS
   ```

Risultati

Se l'applicazione è stata configurata correttamente per entrambi gli utenti, il messaggio dell'utente `alice`viene visualizzato quando bob esegue l'applicazione di richiamo.

8. Verifica della codifica

Informazioni su questa attività

Per verificare che la codifica si stia verificando come previsto, creare una coda alias che faccia riferimento alla coda originale TEST.Q. Questa coda alias non avrà alcuna politica di protezione e quindi nessun utente avrà le informazioni per decodificare il messaggio e quindi verranno visualizzati i dati codificati.

Procedura

1. Utilizzando il comando **runmqsc** per il gestore code QM_VERIFY_AMS, creare una coda alias.

```
DEFINE QALIAS(TEST.ALIAS) TARGET(TEST.Q)
```

2. Concedere a bob l'accesso per sfogliare dalla coda alias

```
setmqaut -m QM_VERIFY_AMS -n TEST.ALIAS -t queue -p bob +browse
```

3. Come utente alice, inserire un altro messaggio utilizzando un'applicazione di esempio come prima:

```
amqsput TEST.Q QM_VERIFY_AMS
```

4. Come utente bob, sfogliare il messaggio utilizzando un'applicazione di esempio tramite la coda alias questa volta:

```
amqsbcg TEST.ALIAS QM_VERIFY_AMS
```

5. Come utente bob, richiamare il messaggio utilizzando un'applicazione di esempio dalla coda locale:

```
amqsget TEST.Q QM_VERIFY_AMS
```

Risultati

L'output dell'applicazione amqsbcg mostra i dati codificati presenti nella coda che dimostrano che il messaggio è stato codificato.

Guida rapida per piattaforme UNIX

Utilizzare questa guida per configurare rapidamente IBM Advanced Message Security per fornire la sicurezza dei messaggi su piattaforme UNIX . Al momento del completamento, sarà stato creato un database di chiavi per verificare le identità utente e le politiche di firma / crittografia definite per il gestore code.

Prima di iniziare

Sul sistema devono essere installati almeno i seguenti componenti:

- Runtime
- Server
- Programmi di esempio
- IBM Global Security Kit
- MQ Advanced Message Security

Fare riferimento ai seguenti argomenti per i nomi componente su ciascuna piattaforma specifica:

- ComponentiIBM WebSphere MQ per sistemi Linux .
- ComponentiIBM WebSphere MQ per sistemi HP-UX .
- ComponentiIBM WebSphere MQ per sistemi AIX .
- ComponentiIBM WebSphere MQ per sistemi Solaris .

1. Creazione di un gestore code e di una coda

Informazioni su questa attività

Tutti i seguenti esempi utilizzano una coda denominata TEST.Q per trasmettere i messaggi tra le applicazioni. Advanced Message Security utilizza gli intercettatori per firmare e codificare i messaggi nel punto in cui entrano nell'infrastruttura WebSphere MQ tramite l'interfaccia WebSphere MQ standard. La configurazione di base viene eseguita in WebSphere MQ e viene configurata nei seguenti passi.

È possibile utilizzare WebSphere MQ Explorer per creare il gestore code QM_VERIFY_AMS e la relativa coda locale denominata TEST.Q utilizzando tutte le impostazioni predefinite della procedura guidata oppure è possibile utilizzare i comandi disponibili in <MQ_INSTALL_PATH>/bin. Tenere presente che è necessario essere un membro del gruppo di utenti mqm per eseguire i seguenti comandi di gestione.

Procedura

1. Creare un gestore code

   ```
   crtmqm QM_VERIFY_AMS
   ```

2. Avvia il gestore code

   ```
   strmqm QM_VERIFY_AMS
   ```

3. Creare una coda denominata TEST.Q immettendo il comando seguente in **runmqsc** per il gestore code QM_VERIFY_AMS

   ```
   DEFINE QLOCAL(TEST.Q)
   ```

Risultati

Se la procedura è stata completata correttamente, il seguente comando immesso in **runmqsc** visualizzerà i dettagli su TEST.Q:

```
DISPLAY Q(TEST.Q)
```

2. Creazione e autorizzazione degli utenti

Informazioni su questa attività

In questo esempio vengono visualizzati due utenti: alice, il mittente e bob, il destinatario. Per utilizzare la coda dell'applicazione, a questi utenti deve essere concessa l'autorizzazione per utilizzarla. Inoltre, per utilizzare con successo le politiche di protezione che definiremo, a questi utenti deve essere concesso l'accesso ad alcune code di sistema. Per ulteriori informazioni sul comando **setmqaut** fare riferimento a **setmqaut** .

Procedura

1. Creare i due utenti

   ```
   useradd alice
   useradd bob
   ```

2. Autorizzare gli utenti a connettersi al gestore code e a lavorare con la coda

   ```
   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. È inoltre necessario consentire ai due utenti di esplorare la coda della normativa di sistema e inserire i messaggi nella coda di errore.

```
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
```

Risultati

I gruppi utenti vengono ora creati e le autorizzazioni richieste vengono loro concesse. In questo modo gli utenti assegnati a tali gruppi avranno anche l'autorizzazione per connettersi al gestore code e per inserire e ottenere dalla coda.

Operazioni successive

Per verificare se le operazioni sono state eseguite correttamente, utilizzare gli esempi `amqsput` e `amqsget` come descritto nella sezione “8. Verifica della codifica” a pagina 285.

3. Creazione di certificati e database di chiavi

Informazioni su questa attività

Per codificare il messaggio, l'intercettatore richiede la chiave privata dell'utente mittente e le chiavi pubbliche del destinatario. Pertanto, è necessario creare il database delle chiavi delle identità utente associate alle chiavi pubbliche e private. Nel sistema reale, dove gli utenti e le applicazioni sono distribuiti su diversi computer, ogni utente avrebbe il proprio keystore privato. Allo stesso modo, in questa guida, vengono creati database di chiavi per `alice` e bob e condivisi i certificati utente tra di loro.

Nota: In questa guida, vengono utilizzate le applicazioni di esempio scritte in C che si collegano utilizzando i bind locali. Se si intende utilizzare le applicazioni Java utilizzando i collegamenti client, è necessario creare un keystore JKS e i certificati utilizzando il comando **`keytool`** , che fa parte del JRE (consultare “Guida rapida per i clienti Java” a pagina 286 per ulteriori dettagli). Per tutti gli altri linguaggi e per le applicazioni Java che utilizzano i collegamenti locali, i passi in questa guida sono corretti.

Procedura

1. Creare un nuovo database di chiavi per l'utente `alice`

   ```
   mkdir /home/alice/.mqs -p
   runmqakm -keydb -create -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -stash
   ```

   Nota:

   - Si consiglia di utilizzare una password complessa per proteggere il database.
   - Il parametro `stash` memorizza la password nel file `key.sth` , che gli intercettatori possono utilizzare per aprire il database.
2. Verificare che sia possibile leggere il database delle chiavi

   ```
   chmod +r /home/alice/.mqs/alicekey.kdb
   ```

3. Crea un certificato che identifica l'utente `alice` da utilizzare nella cifratura

   ```
   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:

   - Ai fini di questa guida, utilizziamo un certificato autofirmato che può essere creato senza utilizzare un'autorità di certificazione. Per i sistemi di produzione, è consigliabile non utilizzare certificati autofirmati, ma affidarsi invece ai certificati firmati da una CA (Certificate Authority).
   - Il parametro `label` specifica il nome per il certificato, che gli intercettatori ricercherà per ricevere le informazioni necessarie.
   - Il parametro DN specifica i dettagli del **DN (Distinguished Name)** , che deve essere univoco per ogni utente.

4. Ora abbiamo creato il database delle chiavi, dovremmo impostarne la proprietà e assicurarci che sia illeggibile da parte di tutti gli altri utenti.

```
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. Ripetere il passo 1-4 per l'utente bob

Risultati

I due utenti `alice` e bob hanno ora un certificato autofirmato.

4. Creazione di keystore.conf

Informazioni su questa attività

È necessario puntare gli intercettatori Advanced Message Security alla directory in cui si trovano i database delle chiavi e i certificati. Ciò viene eseguito tramite il file `keystore.conf` , che contiene tali informazioni in formato di testo semplice. Ciascun utente deve avere un file `keystore.conf` separato nella cartella `.mqs` . Questa operazione deve essere eseguita sia per `alice` che per bob.

Il contenuto di `keystore.conf` deve essere nel formato:

```
cms.keystore = <dir>/keystore_file
cms.certificate = certificate_label
```

Esempio

Per questo scenario, il contenuto di `keystore.conf` sarà il seguente:

```
cms.keystore = /home/alice/.mqs/alicekey
cms.certificate = Alice_Cert
```

Nota:

- Il percorso del file keystore deve essere fornito senza estensione file.
- Esistono i seguenti formati di memorizzazione chiave: `CMS` (Cryptographic Message Syntax), `JKS` (Java Keystore) e `JCEKS` (Java Cryptographic Extension Keystore). Per ulteriori informazioni, fare riferimento a “Struttura del file di configurazione keystore (keystore.conf)” a pagina 296.
- `HOME/.mqs/keystore.conf` è l'ubicazione predefinita in cui Advanced Message Security ricerca il file `keystore.conf` . Per informazioni su come utilizzare un'ubicazione non predefinita per `keystore.conf`, consultare “Utilizzo di keystore e certificati” a pagina 296.

5. Condivisione dei certificati

Informazioni su questa attività

Condividere i certificati tra i due database di chiavi in modo che ogni utente possa identificare correttamente l'altro. Questa operazione viene eseguita estraendo il certificato pubblico di ciascun utente in un file, che viene quindi aggiunto al database delle chiavi dell'altro utente.

Nota: Utilizzare l'opzione *extract* e non l'opzione *export* . *Extract* ottiene la chiave pubblica dell'utente, mentre *export* ottiene sia la chiave pubblica che quella privata. L'utilizzo di *export* per errore comprometterebbe completamente la tua applicazione, passando la sua chiave privata.

Procedura

1. Estrarre il certificato che identifica `alice` in un file esterno:

```
runmqakm -cert -extract -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -label Alice_Cert
-target alice_public.arm
```

2. Aggiungere il certificato al keystore bob ' s :

```
runmqakm -cert -add -db /home/bob/.mqs/bobkey.kdb -pw passw0rd -label Alice_Cert -file
alice_public.arm
```

3. Ripetere il passo per bob:

```
 runmqakm -cert -extract -db /home/bob/.mqs/bobkey.kdb -pw passw0rd -label Bob_Cert -target
bob_public.arm
```

4. Aggiungere il certificato per bob al keystore `alice's` :

```
runmqakm -cert -add -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -label Bob_Cert -file
bob_public.arm
```

Risultati

I due utenti `alice` e bob sono ora in grado di identificarsi correttamente l'uno con l'altro avendo creato e condiviso certificati autofirmati.

Operazioni successive

Verificare che un certificato si trovi nel keystore eseguendo i seguenti comandi che ne stampano i dettagli:

```
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. Definizione della politica della coda

Informazioni su questa attività

Con il gestore code creato e gli intercettatori preparati per intercettare i messaggi e accedere alle chiavi di crittografia, è possibile iniziare a definire le politiche di protezione su `QM_VERIFY_AMS` utilizzando il comando `setmqspl` . Fare riferimento a setmqspl per ulteriori informazioni su questo comando. Ogni nome di politica deve essere uguale al nome della coda a cui deve essere applicato.

Esempio

Questo è un esempio di politica definita per la coda `TEST.Q` . In questo esempio, i messaggi vengono firmati dall'utente `alice` utilizzando l'algoritmo SHA1 e codificati utilizzando l'algoritmo AES a 256 bit.`alice` è l'unico mittente valido e bob è l'unico destinatario dei messaggi su questa coda:

```
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: I DN corrispondono esattamente a quelli specificati nel rispettivo certificato utente dal database delle chiavi.

Operazioni successive

Per verificare la politica definita, immettere il seguente comando:

```
dspmqspl -m QM_VERIFY_AMS
```

Per stampare i dettagli della politica come una serie di comandi `setmqspl` , l'indicatore `-export` . Ciò consente di memorizzare le politiche già definite:

```
dspmqspl -m QM_VERIFY_AMS -export >restore_my_policies.bat
```

7. Verifica della configurazione

Informazioni su questa attività

Eseguendo diversi programmi sotto utenti differenti è possibile verificare se l'applicazione è stata configurata correttamente.

Procedura

1. Passare alla directory contenente gli esempi. Se MQ è installato in un'ubicazione non predefinita, è possibile che si trovi in un'ubicazione diversa.

```
cd /opt/mqm/samp/bin
```

2. Cambia utente da eseguire come utente `alice`

```
su alice
```

3. Come utente `alice`, inserire un messaggio utilizzando un'applicazione di esempio:

```
./amqsput TEST.Q QM_VERIFY_AMS
```

4. Immettere il testo del messaggio, quindi premere Invio.
5. Arresta esecuzione come utente `alice`

```
exit
```

6. Cambia utente da eseguire come utente bob

```
su bob
```

7. Come utente bob, ottenere un messaggio utilizzando un'applicazione di esempio:

```
./amqsget TEST.Q QM_VERIFY_AMS
```

Risultati

Se l'applicazione è stata configurata correttamente per entrambi gli utenti, il messaggio dell'utente `alice`viene visualizzato quando bob esegue l'applicazione di richiamo.

8. Verifica della codifica

Informazioni su questa attività

Per verificare che la codifica si stia verificando come previsto, creare una coda alias che faccia riferimento alla coda originale `TEST.Q`. Questa coda alias non avrà alcuna politica di protezione e quindi nessun utente avrà le informazioni per decodificare il messaggio e quindi verranno visualizzati i dati codificati.

Procedura

1. Utilizzando il comando **runmqsc** per il gestore code QM_VERIFY_AMS, creare una coda alias.

```
DEFINE QALIAS(TEST.ALIAS) TARGET(TEST.Q)
```

2. Concedere a bob l'accesso per sfogliare dalla coda alias

```
setmqaut -m QM_VERIFY_AMS -n TEST.ALIAS -t queue -p bob +browse
```

3. Come utente `alice`, inserire un altro messaggio utilizzando un'applicazione di esempio come prima:

```
./amqsput TEST.Q QM_VERIFY_AMS
```

4. Come utente bob, sfogliare il messaggio utilizzando un'applicazione di esempio tramite la coda alias questa volta:

```
./amqsbcg TEST.ALIAS QM_VERIFY_AMS
```

5. Come utente bob, richiamare il messaggio utilizzando un'applicazione di esempio dalla coda locale:

```
./amqsget TEST.Q QM_VERIFY_AMS
```

Risultati

L'output dell'applicazione `amqsbcg` mostrerà i dati codificati presenti nella coda che dimostrano che il messaggio è stato codificato.

Guida rapida per i clienti Java

Utilizzare questa guida per configurare rapidamente IBM Advanced Message Security per fornire la sicurezza dei messaggi per le applicazioni Java che si collegano utilizzando i collegamenti client. Al momento del completamento, verrà creato un keystore per verificare le identità utente e le politiche di firma / crittografia definite per il gestore code.

Prima di iniziare

Accertarsi di disporre dei componenti appropriati installati come descritto nella **Guida rapida** (Windows o UNIX).

1. Creazione di un gestore code e di una coda

Informazioni su questa attività

Tutti i seguenti esempi utilizzano una coda denominata `TEST.Q` per trasmettere i messaggi tra le applicazioni. Advanced Message Security utilizza gli intercettatori per firmare e codificare i messaggi nel punto in cui entrano nell'infrastruttura WebSphere MQ tramite l'interfaccia WebSphere MQ standard. La configurazione di base viene eseguita in WebSphere MQ e viene configurata nei seguenti passi.

Procedura

1. Creare un gestore code

   ```
   crtmqm QM_VERIFY_AMS
   ```

2. Avvia il gestore code

   ```
   strmqm QM_VERIFY_AMS
   ```

3. Creare e avviare un listener immettendo i seguenti comandi in **runmqsc** per gestore code `QM_VERIFY_AMS`

   ```
   DEFINE LISTENER(AMS.LSTR) TRPTYPE(TCP) PORT(1414) CONTROL(QMGR)
   START LISTENER(AMS.LSTR)
   ```

4. Crea un canale per la connessione delle nostre applicazioni immettendo il seguente comando in **runmqsc** for queue manager `QM_VERIFY_AMS`

   ```
   DEFINE CHANNEL(AMS.SVRCONN) CHLTYPE(SVRCONN)
   ```

5. Creare una coda denominata `TEST.Q` immettendo il comando seguente in **runmqsc** per il gestore code `QM_VERIFY_AMS`

   ```
   DEFINE QLOCAL(TEST.Q)
   ```

Risultati

Se la procedura è stata completata correttamente, il seguente comando immesso in **runmqsc** visualizzerà i dettagli su `TEST.Q`:

```
DISPLAY Q(TEST.Q)
```

2. Creazione e autorizzazione degli utenti

Informazioni su questa attività

Ci sono due utenti che vengono visualizzati in questo scenario: `alice`, il mittente e `bob`, il destinatario. Per utilizzare la coda dell'applicazione, a questi utenti deve essere concessa l'autorizzazione per

utilizzarla. Inoltre, per utilizzare con successo le politiche di protezione che definiremo, a questi utenti deve essere concesso l'accesso ad alcune code di sistema. Per ulteriori informazioni sul comando **setmqaut** fare riferimento a **setmqaut** .

Procedura

1. Creare i due utenti come descritto nella **Guida di avvio rapido** (Windows o UNIX) per la propria piattaforma.
2. Autorizzare gli utenti a connettersi al gestore code e a lavorare con la coda

```
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. È inoltre necessario consentire ai due utenti di esplorare la coda della normativa di sistema e inserire i messaggi nella coda di errore.

```
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
```

Risultati

Gli utenti vengono ora creati e le autorizzazioni richieste vengono loro concesse.

Operazioni successive

Per verificare se le operazioni sono state eseguite correttamente, utilizzare gli esempi `JmsProducer` e `JmsConsumer` come descritto nella sezione “7. Verifica della configurazione” a pagina 290.

3. Creazione di certificati e database di chiavi

Informazioni su questa attività

Per codificare il messaggio all'intercettatore è necessaria la chiave pubblica degli utenti che inviano. Pertanto, è necessario creare il database delle chiavi delle identità utente associate alle chiavi pubbliche e private. Nel sistema reale, in cui gli utenti e le applicazioni sono distribuiti su più computer, ogni utente avrebbe il proprio keystore privato. Allo stesso modo, in questa guida, vengono creati database di chiavi per `alice` e bob e condivisi i certificati utente tra di loro.

Nota: In questa guida, vengono utilizzate le applicazioni di esempio scritte in Java che si collegano utilizzando i collegamenti client. Se si prevede di utilizzare le applicazioni Java utilizzando i bind locali o le applicazioni C, è necessario creare un keystore e i certificati CMS utilizzando il comando **runmqakm** . Ciò viene mostrato nella **Guida di avvio rapido** (Windows o UNIX).

Procedura

1. Creare una directory in cui creare il keystore, ad esempio `/home/alice/.mqs`. È possibile crearlo nella stessa directory utilizzata dalla **Guida rapida** (Windows o UNIX) per la propria piattaforma.

   Nota: Questa directory verrà indicata come *keystore-dir* nei seguenti passi
2. Creare un nuovo keystore e certificato che identifichi l'utente `alice` da utilizzare nella codifica

   Nota: Il comando **keytool** fa parte di JRE.

```
keytool -genkey -alias Alice_Java_Cert -keyalg RSA -keystore keystore-dir/keystore.jks
-storepass passw0rd
-dname "CN=alice, O=IBM, C=GB" -keypass passw0rd
```

   Nota:

   - Se la *keystore-dir* contiene spazi, è necessario racchiudere tra virgolette il nome completo del keystore

- Si consiglia di utilizzare una password complessa per proteggere il keystore.
- Ai fini di questa guida, utilizziamo un certificato autofirmato che può essere creato senza utilizzare un'autorità di certificazione. Per i sistemi di produzione, è consigliabile non utilizzare certificati autofirmati, ma affidarsi ai certificati firmati da una CA (Certificate Authority).
- Il parametro `alias` specifica il nome per il certificato, che gli intercettatori ricercherà per ricevere le informazioni necessarie.
- Il parametro `dname` specifica i dettagli del **DN (Distinguished Name)** , che deve essere univoco per ogni utente.

3. In UNIX, accertarsi che l'archivio chiavi sia leggibile

```
chmod +r keystore-dir/keystore.jks
```

4. Ripetere step1-4 per l'utente bob

Risultati

I due utenti `alice` e bob hanno ora un certificato autofirmato.

4. Creazione di keystore.conf

Informazioni su questa attività

È necessario puntare gli intercettatori Advanced Message Security alla directory in cui si trovano i database delle chiavi e i certificati. Questa operazione viene effettuata tramite il file `keystore.conf` , che contiene tali informazioni in formato testo semplice. Ciascun utente deve avere un file `keystore.conf` separato. Questa operazione deve essere eseguita sia per `alice` che per bob.

Esempio

Per questo scenario, il contenuto di `keystore.conf` per `alice` sarà il seguente:

```
JKS.keystore = keystore-dir/keystore
JKS.certificate = Alice_Java_Cert
JKS.encrypted = no
JKS.keystore_pass = passw0rd
JKS.key_pass = passw0rd
JKS.provider = IBMJCE
```

Per questo scenario, il contenuto di `keystore.conf` per bob sarà il seguente:

```
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:

- Il percorso del file keystore deve essere fornito senza estensione file.
- Se si dispone già di un `keystore.conf` perché è stata seguita la **Guida di avvio rapido** (Windows o UNIX), è possibile modificare quello esistente per aggiungerlo nelle righe precedenti.
- Per ulteriori informazioni, consultare “Struttura del file di configurazione keystore (keystore.conf)” a pagina 296.

5. Condivisione dei certificati

Informazioni su questa attività

Condividi i certificati tra i due keystore in modo che ogni utente possa identificare correttamente l'altro. Questa operazione viene eseguita estraendo il certificato di ciascun utente e importandolo nel keystore dell'altro utente.

Nota: I termini *extract* e *export* vengono utilizzati in maniera diversa da diversi strumenti di certificazione. Ad esempio, lo strumento IBM GSKit Keyman (ikeyman) fa una distinzione tra l' *estrazione* di certificati (chiavi pubbliche) e l' *esportazione* di chiavi private. Questa distinzione è estremamente importante per strumenti che offrono entrambe le opzioni, poiché l'uso di *export* per errore comprometterebbe completamente la tua applicazione trasmettendo la sua chiave privata. Poiché la distinzione è così importante, la documentazione di WebSphere MQ si sforza di utilizzare questi termini in modo coerente. Tuttavia, Java keytool fornisce un'opzione della riga comandi denominata *exportcert* che estrae solo la chiave pubblica. Per questi motivi, la seguente procedura fa riferimento all' *estrazione* dei certificati utilizzando l'opzione *exportcert* .

Procedura

1. Estrarre il certificato che identifica `alice`.

```
keytool -exportcert -keystore alice-keystore-dir/keystore.jks -storepass passw0rd
-alias Alice_Java_Cert -file alice-keystore-dir/Alice_Java_Cert.cer
```

2. Importare il certificato che identifica `alice` nel keystore che verrà utilizzato da bob . Quando richiesto, indicare che questo certificato verrà accreditato.

```
keytool -importcert -file alice-keystore-dir/Alice_Java_Cert.cer -alias Alice_Java_Cert
-keystore bob-keystore-dir/keystore.jks -storepass passw0rd
```

3. Ripetere i passi per bob

Risultati

I due utenti `alice` e bob sono ora in grado di identificarsi correttamente l'uno con l'altro avendo creato e condiviso certificati autofirmati.

Operazioni successive

Verificare che un certificato si trovi nel keystore eseguendo i seguenti comandi che ne stampano i dettagli:

```
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. Definizione della politica della coda

Informazioni su questa attività

Con il gestore code creato e gli intercettatori preparati per intercettare i messaggi e accedere alle chiavi di crittografia, è possibile iniziare a definire le politiche di protezione su `QM_VERIFY_AMS` utilizzando il comando `setmqspl` . Fare riferimento a setmqspl per ulteriori informazioni su questo comando. Ogni nome di politica deve essere uguale al nome della coda a cui deve essere applicato.

Esempio

Questo è un esempio di politica definita sulla coda `TEST.Q` , firmata dall'utente `alice` utilizzando l'algoritmo SHA1 e codificata utilizzando l'algoritmo AES a 256 bit per l'utente 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: I DN corrispondono esattamente a quelli specificati nel rispettivo certificato utente dal database delle chiavi.

Operazioni successive

Per verificare la politica definita, immettere il seguente comando:

```
dspmqspl -m QM_VERIFY_AMS
```

Per stampare i dettagli della politica come una serie di comandi `setmqspl` , l'indicatore `-export` . Ciò consente di memorizzare le politiche già definite:

```
dspmqspl -m QM_VERIFY_AMS -export >restore_my_policies.bat
```

7. Verifica della configurazione

Prima di iniziare

Assicurarsi che la versione di Java che si sta utilizzando abbia i file di politica JCE senza limitazioni installati.

Nota: La versione di Java fornita nell'installazione di WebSphere MQ contiene già questi file di politica. Può essere trovato in *MQ_INSTALLATION_PATH*/java/bin.

Informazioni su questa attività

Eseguendo diversi programmi sotto utenti differenti è possibile verificare se l'applicazione è stata configurata correttamente. Fare riferimento alla **Guida di avvio rapido** (Windows o UNIX) per la propria piattaforma, per i dettagli sull'esecuzione dei programmi da parte di utenti differenti.

Procedura

1. Per eseguire queste applicazioni di esempio JMS, utilizzare l'impostazione CLASSPATH per la piattaforma come mostrato in Variabili di ambiente utilizzate dalle classi IBM WebSphere MQ per JMS per assicurarsi che la directory degli esempi sia inclusa.
2. Come utente `alice`, inserire un messaggio utilizzando un'applicazione di esempio, collegandosi come client:

   ```
   java JMSProducer -m QM_VERIFY_AMS -d TEST.Q -h localhost -p 1414 -l AMS.SVRCONN
   ```

3. Come utente bob, ottenere un messaggio utilizzando un'applicazione di esempio, collegandosi come client:

   ```
   java JMSConsumer -m QM_VERIFY_AMS -d TEST.Q -h localhost -p 1414 -l AMS.SVRCONN
   ```

Risultati

Se l'applicazione è stata configurata correttamente per entrambi gli utenti, il messaggio dell'utente `alice`viene visualizzato quando bob esegue l'applicazione di richiamo.

Protezione di code remote

Per proteggere completamente le connessioni della coda remota, è necessario impostare la stessa politica sulla coda remota e sulla coda locale a cui vengono trasmessi i messaggi.

Quando un messaggio viene inserito in una coda remota, Advanced Message Security intercetta l'operazione ed elabora il messaggio in base ad una serie di criteri per la coda remota. Ad esempio, per una politica di crittografia, il messaggio viene crittografato prima di essere passato a WebSphere MQ per gestirlo. Una volta che Advanced Message Security ha elaborato il messaggio inserito in una coda remota, WebSphere MQ lo inserisce nella coda di trasmissione associata e lo inoltra al gestore code di destinazione e alla coda di destinazione.

Quando un'operazione GET viene eseguita sulla coda locale, Advanced Message Security tenta di decodificare il messaggio in base alla serie di politiche sulla coda locale. Affinché l'operazione riesca, la politica utilizzata per decodificare il messaggio deve essere identica a quella utilizzata per codificarlo. Qualsiasi discrepanza causerà il rifiuto del messaggio.

Se per qualsiasi motivo non è possibile impostare entrambe le politiche contemporaneamente, viene fornito un supporto di roll-out a fasi. La politica può essere impostata su una coda locale con indicatore

di tolleranza attivo, che indica che una politica associata ad una coda può essere ignorata quando un tentativo di richiamare un messaggio dalla coda implica un messaggio che non ha la politica di sicurezza impostata. In questo caso, GET tenterà di decodificare il messaggio, ma consentirà la consegna di messaggi non codificati. In questo modo le politiche sulle code remote possono essere impostate dopo che le code locali sono state protette (e verificate).

Attenzione: Rimuovere l'indicatore di tolleranza una volta completato il rollout di Advanced Message Security .

Riferimenti correlati

setmqspl (impostazione politica di sicurezza)

Instradamento dei messaggi protetti mediante WebSphere Message Broker

IBM Advanced Message Security è in grado di proteggere i messaggi in un'infrastruttura in cui è installato WebSphere Message Broker versione 8.0.0.1 (o successiva). È necessario comprendere la natura di entrambi i prodotti prima di applicare la sicurezza nell'ambiente WebSphere Message Broker.

Informazioni su questa attività

Advanced Message Security fornisce la sicurezza end-to-end del payload del messaggio. Ciò significa che solo le parti specificate come mittenti e destinatari validi di un messaggio sono in grado di produrlo o riceverlo. Ciò implica che, per proteggere i messaggi che passano attraverso WebSphere Message Broker, è possibile consentire a WebSphere Message Broker di elaborare i messaggi senza conoscerne il contenuto (Scenario 1) o renderlo un utente autorizzato in grado di ricevere e inviare messaggi (Scenario 2).

Scenario 1 - Message Broker non può visualizzare il contenuto del messaggio

Prima di iniziare

È necessario che WebSphere Message Broker sia connesso a un gestore code esistente. Sostituire *QMgrName* con questo nome gestore code esistente nei comandi che seguono.

Informazioni su questa attività

In questo scenario, Alice inserisce un messaggio protetto in una coda di input `QIN`. In base alla proprietà del messaggio `routeTo`, il messaggio viene instradato a *bob* (`QBOB`),[1](`QCECIL`) o la coda predefinita (`QDEF`). L'instradamento è possibile perché Advanced Message Security protegge solo il payload del messaggio e non le intestazioni e le proprietà che rimangono non protette e che possono essere lette da WebSphere Message Broker. Advanced Message Security è utilizzato solo da *alice*, *bob* e *cecil*. Non è necessario installarlo o configurarlo per WebSphere Message Broker.

WebSphere Message Broker riceve il messaggio protetto dalla coda alias non protetta per evitare qualsiasi tentativo di decodifica del messaggio. Se dovesse utilizzare direttamente la coda protetta, il messaggio verrebbe inserito nella coda DEAD LETTER come impossibile da decodificare. Il messaggio viene instradato da WebSphere Message Broker e arriva sulla coda di destinazione non modificata. Pertanto è ancora firmato dall'autore originale (sia *bob* che *cecil* accettano solo i messaggi inviati da *alice*) e protetto come prima (solo *bob* e *cecil* possono leggerlo). WebSphere Message Broker inserisce il messaggio instradato in un alias non protetto. I destinatari richiamano il messaggio da una coda di output protetta in cui IBM WebSphere MQ AMS decodificherà in modo trasparente il messaggio.

Procedura

1. Configurare *alice*, *bob* e *cecil* per utilizzare Advanced Message Security come descritto nella **Guida rapida** (Windows o UNIX).

   Accertarsi che siano state completate le seguenti operazioni:

   - Creazione e autorizzazione di utenti
   - Creazione di database di chiavi e certificati

[1] cecil

- Creazione di keystore.conf

2. Fornire il certificato *alice* a *bob* e *cecil*, in modo che *alice* possa essere identificato da loro quando controllano le firme digitali sui messaggi.

   Eseguire questa operazione estraendo il certificato che identifica *alice* in un file esterno, quindi aggiungendo il certificato estratto ai keystore *bob* e *cecil* . È importante utilizzare il metodo descritto in **Attività 5. Condivisione dei certificati** nella **Guida di avvio rapido** (Windows o UNIX).
3. Fornisci i certificati *bob* e *cecil* a *alice*, in modo che *alice* possa inviare messaggi codificati per *bob* e *cecil*.

   Eseguire questa operazione utilizzando il metodo specificato nel passo precedente.
4. Sul proprio gestore code, definire le code locali denominate QIN, QBOB, QCECIL e QDEF.

   ```
   DEFINE QLOCAL(QIN)
   ```
5. Impostare la politica di sicurezza per la coda QIN su una configurazione idonea. Utilizzare la configurazione identica per le code QBOB, QCECIL e QDEF .

   ```
   setmqspl -m QMgrName -p QIN -s SHA1 -a "CN=alice,O=IBM,C=GB"
   -e AES256 -r "CN=bob,O=IBM,C=GB" -r "CN=cecil,O=IBM,C=GB"
   ```

   Questo scenario presuppone la politica di sicurezza dove *alice* è l'unico mittente autorizzato e *bob* e *cecil* sono i destinatari.
6. Definire le code alias AIN, ABOB e ACECIL che fanno riferimento rispettivamente alle code locali QIN, QBOB e QCECIL .

   ```
   DEFINE QALIAS(AIN) TARGET(QIN)
   ```
7. Verificare che la configurazione di sicurezza per gli alias specificati nel passo precedente non sia presente; altrimenti impostare la relativa politica su NONE.

   ```
   dspmqspl -m QMgrName -p AIN
   ```
8. In WebSphere Message Broker creare un flusso di messaggi per instradare i messaggi in arrivo sulla coda alias AIN al nodo BOB, CECIL o DEF in base alla proprietà routeTo del messaggio. Per farlo:
   a) Creare un MQInput nodo denominato IN e assegnare l'alias AIN come nome coda.
   b) Creare i nodi MQOutput denominati BOB, CECIL e DEF e assegnare le code alias ABOB, ACECIL e ADEF come rispettivi nomi coda.
   c) Crea un nodo di instradamento e chiamalo TEST.
   d) Connettere il nodo di IN al terminale di input del nodo TEST .
   e) Creare terminali di output bobe cecil per il nodo TEST .
   f) Connettere il terminal di output bob al nodo BOB .
   g) Connettere il terminal di output cecil al nodo CECIL .
   h) Connetti il nodo DEF al terminale di output predefinito.
   i) Applicare le regole seguenti:

      ```
      $Root/MQRFH2/usr/routeTo/text()="bob"
      $Root/MQRFH2/usr/routeTo/text()="cecil"
      ```
9. Distribuire il flusso di messaggi al componente di runtime di WebSphere Message Broker.
10. L'esecuzione come utente Alice inserisce un messaggio che contiene anche una proprietà del messaggio denominata routeTo con un valore di bob o cecil. L'esecuzione dell'applicazione di esempio **amqsstm** consente di eseguire questa operazione.

    ```
    Sample AMQSSTMA start
    target queue is TEST.Q
    Enter property name
    routeTo
    Enter property value
    ```

```
bob
Enter property name

Enter message text
My Message to Bob
Sample AMQSSTMA end
```

11. L'esecuzione come utente *bob* richiama il messaggio dalla coda QBOB utilizzando l'applicazione di esempio **amqsget**.

Risultati

Quando *alice* inserisce un messaggio nella coda QIN , il messaggio è protetto. Viene richiamato in forma protetta da WebSphere Message Broker dalla coda alias AIN . WebSphere Message Broker decide dove instradare il messaggio leggendo la proprietà routeTo che, come tutte le proprietà, non è codificata. WebSphere Message Broker posiziona il messaggio sull'alias non protetto appropriato evitando la sua ulteriore protezione. Quando viene ricevuto da *bob* o *cecil* dalla coda, il messaggio viene decodificato e la firma digitale viene verificata.

Scenario 2 - Message Broker può visualizzare il contenuto del messaggio

Informazioni su questa attività

In questo scenario, a un gruppo di persone è consentito inviare messaggi a WebSphere Message Broker. Un altro gruppo è autorizzato a ricevere i messaggi creati da WebSphere Message Broker. La trasmissione tra le parti e WebSphere Message Broker non può essere intercettata.

Tenere presente che WebSphere Message Broker legge le politiche di protezione e i certificati solo quando viene aperta una coda, quindi è necessario ricaricare il gruppo di esecuzione dopo aver apportato gli aggiornamenti alle politiche di protezione per rendere effettive le modifiche.

```
mqsireload execution-group-name
```

Se WebSphere Message Broker è considerato una parte autorizzata a leggere o firmare il payload del messaggio, è necessario configurare Advanced Message Security per l'utente che avvia il servizio WebSphere Message Broker. Tenere presente che non è necessariamente lo stesso utente che inserisce / richiama i messaggi nelle code né l'utente che crea e distribuisce le applicazioni WebSphere Message Broker.

Procedura

1. Configurare *alice*, *bob*, *cecil* e *dave* e l'utente del servizio WebSphere Message Broker, per utilizzare Advanced Message Security come descritto nella **Guida rapida** (Windows o UNIX).

   Accertarsi che siano state completate le seguenti operazioni:

   - Creazione e autorizzazione di utenti
   - Creazione di database di chiavi e certificati
   - Creazione di keystore.conf
2. Fornire i certificati *alice*, *bob*, *cecil* e *dave* all'utente del servizio WebSphere Message Broker.

   Eseguire questa operazione estraendo in file esterni ciascuno dei certificati che identificano *alice*, *bob*, *cecil* e *dave*, quindi aggiungendo i certificati estratti al keystore di WebSphere Message Broker. È importante utilizzare il metodo descritto in **Attività 5. Condivisione dei certificati** nella **Guida di avvio rapido** (Windows o UNIX).
3. Fornire il certificato utente del servizio WebSphere Message Broker a *alice*, *bob*, *cecil* e *dave*.

   Eseguire questa operazione utilizzando il metodo specificato nel passo precedente.

   Nota: *Alice* e *bob* hanno bisogno del certificato dell'utente del servizio WebSphere Message Broker per codificare correttamente i messaggi. L'utente del servizio WebSphere Message Broker necessita dei certificati *alice* e *bob* per verificare gli autori dei messaggi. L'utente del servizio WebSphere Message Broker ha bisogno dei certificati *cecil* e *dave* per crittografare i messaggi. *cecil* e *dave* hanno bisogno

del certificato dell'utente del servizio WebSphere Message Broker per verificare se il messaggio proviene da WebSphere Message Broker.

4. Definire una coda locale denominata IN e definire la politica di protezione con *alice* e *bob* specificati come autori e WebSphere l'utente del servizio di Message Broker specificato come 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. Definire una coda locale denominata OUT e definire la politica di protezione con l'utente del servizio di WebSphere Message Broker specificato come autore e *cecil* e *dave* specificato come destinatari:

```
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. In WebSphere Message Broker creare un flusso di messaggi con un nodo MQInput e MQOutput . Configurare il nodo MQInput per utilizzare la coda IN e il nodo MQOutput per utilizzare la coda OUT .
7. Distribuire il flusso di messaggi al componente di runtime di WebSphere Message Broker.
8. L'esecuzione come utente *alice* o *bob* inserisce un messaggio nella coda IN utilizzando l'applicazione di esempio **amqsput**.
9. L'esecuzione come utente *cecil* o *dave* richiama il messaggio dalla coda OUT utilizzando l'applicazione di esempio **amqsget**.

Risultati

I messaggi inviati da *alice* o *bob* alla coda di input IN vengono codificati consentendo solo a WebSphere Message Broker di leggerli. WebSphere Message Broker accetterà solo i messaggi da *alice* e *bob* e rifiuterà tutti gli altri. I messaggi accettati verranno elaborati in modo appropriato, quindi firmati e codificati con le chiavi *cecil* e *dave* prima di essere inseriti nella coda di output OUT. Solo *cecil* e *dave* sono in grado di leggerlo, i messaggi non firmati da WebSphere Message Broker vengono rifiutati.

Utilizzo di IBM WebSphere MQ Advanced Message Security con IBM WebSphere MQ Managed File Transfer

Questo scenario spiega come configurare Advanced Message Security per fornire la riservatezza dei messaggi per i dati inviati tramite un IBM WebSphere MQ Managed File Transfer.

Prima di iniziare

Accertarsi che il componente Advanced Message Security sia installato nell'installazione WebSphere MQ che ospita le code utilizzate da IBM WebSphere MQ Managed File Transfer che si desidera proteggere.

Se gli agent IBM WebSphere MQ Managed File Transfer si collegano in modalità bind, verificare che il componente GSKit sia installato nella relativa installazione locale.

Informazioni su questa attività

Quando il trasferimento dei dati tra due agent IBM WebSphere MQ Managed File Transfer viene interrotto, è possibile che i dati riservati non siano protetti sulle code WebSphere MQ sottostanti utilizzate per gestire il trasferimento. Questo scenario spiega come configurare e utilizzare Advanced Message Security per proteggere tali dati sulle code IBM WebSphere MQ Managed File Transfer .

In questo scenario si considera una semplice topologia che comprende una macchina con due code IBM WebSphere MQ Managed File Transfer e due agent, AGENT1 e AGENT2, che condividono un unico gestore code, hubQM, come descritto nello scenario Trasferimento file di base utilizzando gli script. Entrambi gli agent si collegano nello stesso modo, in modalità bind o in modalità client.

1. Creazione di certificati

Prima di iniziare

Questo scenario utilizza un modello semplice in cui un utente ftagent in un gruppo FTAGENTS viene utilizzato per eseguire i processi dell'agente IBM WebSphere MQ Managed File Transfer . Se si utilizzano i propri nomi utente e gruppo, modificare i comandi di conseguenza.

Informazioni su questa attività

Advanced Message Security utilizza la crittografia a chiave pubblica per firmare e / o codificare i messaggi sulle code protette.

Nota:

- Se gli agent IBM WebSphere MQ Managed File Transfer sono in esecuzione in modalità bind, i comandi utilizzati per creare un keystore CMS (Cryptographic Message Syntax) sono descritti in dettaglio nella **Guida rapida** (Windows o UNIX) per la piattaforma.
- Se gli agent IBM WebSphere MQ Managed File Transfer sono in esecuzione in modalità client, i comandi necessari per creare un JKS (Java Keystore) sono descritti in dettaglio in "Guida rapida per i clienti Java" a pagina 286.

Procedura

1. Creare un certificato autofirmato per identificare l'utente `ftagent` come descritto nella Guida di avvio rapido appropriata.
   Utilizzare un DN (Distinguished Name) come segue:

   ```
   CN=ftagent, OU=MFT, O=IBM, L=Hursley, ST=Hampshire, C=GB
   ```

2. Creare un file `keystore.conf` per identificare l'ubicazione del keystore e il relativo certificato, come descritto nella Guida di avvio rapido appropriata.

2. Configurazione della protezione dei messaggi

Informazioni su questa attività

È necessario definire una politica di sicurezza per la coda dati utilizzata da AGENT2, utilizzando il comando **setmqspl** . In questo scenario lo stesso utente viene utilizzato per avviare entrambi gli agenti e quindi il DN del firmatario e del destinatario sono uguali e corrispondono al certificato generato.

Procedura

1. Arrestare gli agent IBM WebSphere MQ Managed File Transfer in preparazione della protezione utilizzando il comando **fteStopAgent** .
2. Creare una politica di sicurezza per proteggere la coda 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. Accertarsi che l'utente che esegue il processo dell'agente IBM WebSphere MQ Managed File Transfer abbia accesso per esplorare la coda delle politiche di sistema e inserire i messaggi nella coda di errore.

   ```
   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. Riavviare gli agent IBM WebSphere MQ Managed File Transfer utilizzando il comando **fteStartAgent** .
5. Confermare che gli agent siano stati riavviati correttamente utilizzando il comando **fteListAgents** e verificando che gli agent si trovino nello stato READY .

Risultati

Ora è possibile inoltrare i trasferimenti da AGENT1 a AGENT2e il contenuto del file verrà trasmesso in modo sicuro tra due agenti.

InstallazioneIBM WebSphere MQ Advanced Message Security

Installare il componente IBM WebSphere MQ Advanced Message Security su varie piattaforme.

Informazioni su questa attività

Per le procedure di installazione complete, vedere Installazione di IBM WebSphere MQ Advanced Message Security.

Attività correlate

disinstallazioneIBM WebSphere MQ Advanced Message Security

Utilizzo di keystore e certificati

Per fornire una protezione crittografica trasparente per le applicazioni WebSphere MQ , Advanced Message Security utilizza il file keystore, in cui sono memorizzati i certificati della chiave pubblica e una chiave privata.

In Advanced Message Security, utenti e applicazioni sono rappresentati dalle identità PKI (Public Key Infrastructure). Questo tipo di identità viene utilizzato per firmare e codificare i messaggi. L'identità PKI è rappresentata dal campo **DN (distinguished name)** dell'oggetto in un certificato associato ai messaggi firmati e codificati. Per un utente o un'applicazione per codificare i propri messaggi, è necessario accedere al file keystore in cui sono memorizzati i certificati e le chiavi pubbliche e private associate.

L'ubicazione del keystore viene fornita nel relativo file di configurazione, che per impostazione predefinita è `keystore.conf` . Ogni utente Advanced Message Security deve avere il file di configurazione keystore che punta a un file keystore. Advanced Message Security accetta il seguente formato di file keystore: `.kdb`, `.jceks`, `.jks`.

L'ubicazione predefinita del file `keystore.conf` è:

- Su piattaforme UNIX : `$HOME/.mqs/keystore.conf`
- Su piattaforme Windows : *%HOMEDRIVE%%HOMEPATH%*`\.mqs\keystore.conf`

Se si sta utilizzando un percorso e un nome file keystore specificati, è necessario utilizzare i seguenti comandi

- Per Java `java -D` *MQS_KEYSTORE_CONF=path/filename app_name* :
- Per client e server C:
  - Su UNIX and Linux: `export MQS_KEYSTORE_CONF=`*path/filename*
  - Su Windows: `set MQS_KEYSTORE_CONF=`*path\filename*

    Nota: Il percorso su Windows può e deve specificare la lettera dell'unità se è disponibile più di una lettera dell'unità.

Concetti correlati

“Nomi distinti del mittente” a pagina 309

I DN (distinguished name) del mittente identificano gli utenti autorizzati a inserire messaggi in una coda.

“Nomi distinti del destinatario” a pagina 310

Il DN (distinguished name) del destinatario identifica gli utenti autorizzati a richiamare i messaggi da una coda.

Struttura del file di configurazione keystore (keystore.conf)

Il file di configurazione del keystore (`keystore.conf`) punta Advanced Message Security all'ubicazione del keystore appropriato.

Esistono due tipi di configurazione CMS e Java (JKS e JCEKS). Le voci di configurazione CMS hanno come prefisso `cms.` e Java hanno come prefisso `jks.` o `jceks.` a seconda del tipo di keystore.

Il file di configurazione, a seconda del tipo di file di configurazione, può avere una delle seguenti strutture:

```
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
```

I parametri del file di configurazione sono definiti come segue:

keystore
Percorso del file keystore.

Importante:

- Il percorso del file keystore non deve includere l'estensione file.
- Per i file keystore Java, IBM WebSphere MQ AMS supporta i formati file seguenti: `.jks`, `.jceks`, `.jck`.

certificate
Etichetta del certificato.

encrypted
Stato della password.

keystore_pass
Password per il file keystore.

Nota:

- Per il keystore CMS, IBM WebSphere MQ AMS si basa sui file stash (`.sth`), mentre JKS e JCEKS potrebbero richiedere una password sia per il certificato che per la chiave privata dell'utente.
- La memorizzazione delle password in testo semplice rappresenta un rischio per la sicurezza.

key_pass
Password per la chiave privata dell'utente.

Importante: La memorizzazione delle password in formato di testo semplice può rappresentare un rischio per la sicurezza.

provider
Il provider di sicurezza Java che implementa gli algoritmi crittografici richiesti dal certificato keystore.

Nota: Attualmente IBMJCE è l'unico provider supportato da Advanced Message Security.

Importante: Le informazioni memorizzate nel keystore sono fondamentali per il flusso sicuro dei dati inviati utilizzando WebSphere MQ, motivo per cui gli amministratori della sicurezza devono prestare particolare attenzione quando assegnano autorizzazioni file a questi file.

Di seguito è riportato un esempio del file `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
```

Attività correlate

“Protezione delle password in Java” a pagina 307
La memorizzazione delle password del keystore e della chiave privata come testo semplice rappresenta un rischio per la sicurezza, pertanto Advanced Message Security fornisce uno strumento che può codificare tali password utilizzando una chiave utente, disponibile nel file keystore.

Intercettazione MCA (Message Channel Agent)

L'intercettazione MCA consente a un gestore code in esecuzione in IBM WebSphere MQ di abilitare in modo selettivo le politiche da applicare ai canali di connessione server.

L'intercettazione MCA consente ai client esterni a IBM WebSphere MQ AMS di essere ancora connessi a un gestore code e ai relativi messaggi di essere codificati e decodificati.

L'intercettazione MCA è progettata per fornire la funzionalità IBM WebSphere MQ AMS quando IBM WebSphere MQ AMS non è abilitato sul client. Si noti che l'utilizzo dell'intercettazione MCA e di un client abilitato a IBM WebSphere MQ AMSporta a una doppia protezione dei messaggi che potrebbe essere problematica per la ricezione di applicazioni.

Se viene riportato un messaggio di errore `2085 (MQRC_UNKNOWN_OBJECT_NAME)` se si utilizza un client Version 7.5 o successivo per connettersi a un gestore code da una versione precedente del prodotto, è necessario disabilitare IBM WebSphere MQ Advanced Message Security sul client. Per ulteriori informazioni, consultare “Disabilitazione di IBM WebSphere MQ Advanced Message Security sul client” a pagina 300.

File di configurazione keystore

Per impostazione predefinita, il file di configurazione del keystore per l'intercettazione MCA è `keystore.conf` e si trova nella directory `.mqs` nel percorso della directory HOME dell'utente che ha avviato il gestore code o il listener. Il keystore può essere configurato anche utilizzando la variabile di ambiente MQS_KEYSTORE_CONF. Per ulteriori informazioni sulla configurazione del keystore IBM WebSphere MQ AMS , consultare “Utilizzo di keystore e certificati” a pagina 296.

Per abilitare l'intercettazione MCA, è necessario specificare un nome di canale che si desidera utilizzare nel file di configurazione del keystore. Per l'intercettazione MCA, è possibile utilizzare solo un tipo di keystore cms.

Per un esempio di impostazione dell'intercettazione MCA, consultare “IBM WebSphere MQ AMS Esempio di intercettazione MCA” a pagina 298.

Attenzione: È necessario completare l'autenticazione del client e la crittografia sui canali selezionati, ad esempio, utilizzando SSL e SSLPEER o CHLAUTH TYPE (SSLPEERMAP), per garantire che solo i client autorizzati possano connettersi e utilizzare questa funzionalità.

IBM WebSphere MQ AMS Esempio di intercettazione MCA

Un'attività di esempio su come impostare un'intercettazione MCA IBM WebSphere MQ AMS .

Prima di iniziare

Attenzione: È necessario completare l'autenticazione del client e la crittografia sui canali selezionati, ad esempio, utilizzando SSL e SSLPEER o CHLAUTH TYPE (SSLPEERMAP), per garantire che solo i client autorizzati possano connettersi e utilizzare questa funzionalità.

Informazioni su questa attività

Questa attività consente di eseguire il processo di configurazione del sistema per utilizzare l'intercettazione MCA, quindi di verificare la configurazione.

Nota: Prima di IBM WebSphere MQ Version 7.5, IBM WebSphere MQ AMS era un prodotto aggiuntivo che doveva essere installato separatamente e gli intercettatori configurati per proteggere le applicazioni. Da Version 7.5 in poi, gli intercettatori vengono automaticamente inclusi e abilitati dinamicamente negli ambienti di runtime client e server MQ . In questo esempio di intercettazione MCA, gli intercettatori vengono forniti all'estremità del server del canale e viene utilizzato un runtime client precedente (nel passo 12) per inserire un messaggio non protetto nel canale in modo che possa essere visualizzato come protetto dagli intercettatori MCA. Se questo esempio avesse utilizzato un Version 7.5 o un client successivo, causerebbe la protezione del messaggio due volte, poiché l'intercettatore di runtime del client MQ e l'intercettatore MCA proteggerebbero entrambi il messaggio man mano che entra in MQ.

Attenzione: Sostituire `userID` nel codice con il proprio ID utente.

Procedura

1. Creare il database delle chiavi e i certificati utilizzando i seguenti comandi per creare uno script shell.

   Inoltre, modificare **INSTLOC** e **KEYSTORELOC** oppure eseguire i comandi richiesti. Tenere presente che potrebbe non essere necessario creare il certificato per 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. Condividere i certificati tra i due database di chiavi in modo che ogni utente possa identificare correttamente l'altro.

   È importante utilizzare il metodo descritto in **Attività 5. Condivisione dei certificati** nella **Guida di avvio rapido** (Windows o UNIX).

3. Creare `keystore.conf` con la configurazione seguente: `Keystore.conf location: /home/userID/ssl/ams1/`

```
cms.keystore = /home/userID/ssl/ams1/alicekey
cms.certificate.channel.SYSTEM.DEF.SVRCONN = alice_cert
```

4. Creare e avviare il gestore code AMSQMGR1
5. Definire un listener con *port* 14567 e *control* QMGR
6. Disabilitare l'autorizzazione del canale o impostare le regole per l'autorizzazione del canale.

   Consultare SET CHLAUTH per ulteriori informazioni.

7. Chiudere il gestore code.
8. Impostare il keystore:

```
export MQS_KEYSTORE_CONF=/home/userID/ssl/ams1/keystore.conf
```

9. Avviare il gestore code sulla stessa shell.
10. Impostare la politica di sicurezza e verificare:

```
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
```

    Per ulteriori informazioni, consultare setmqspl e dspmqspl .

11. Impostare la configurazione del canale:

```
export MQSERVER='SYSTEM.DEF.SVRCONN/TCP/127.0.0.1(14567)'
```

12. Eseguire **amqsputc** da un client MQ che non abilita automaticamente un intercettatore MCA, ad esempio un client IBM WebSphere MQ Version 7.1 o precedente. Inserire i seguenti due messaggi:

```
/opt/mqm/samp/bin/amqsputc TESTQ TESTQMGR
```

13. Rimuovere la politica di sicurezza e verificare il risultato:

```
setmqspl -m AMSQMGR1 -p TESTQ -remove
dspmqspl -m AMSQMGR1
```

14. Sfogliare la coda dall'installazione di IBM WebSphere MQ Version 7.5 :

```
/opt/mq75/samp/bin/amqsbcg TESTQ AMSQMGR1
```

L'output di ricerca mostra i messaggi in formato codificato.

15. Impostare la politica di sicurezza e verificare il risultato:

```
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. Eseguire **amqsgetc** dall'installazione di IBM WebSphere MQ Version 7.5 :

```
/opt/mqm/samp/bin/amqsgetc TESTQ TESTQMGR
```

Attività correlate

“Guida rapida per i clienti Java” a pagina 286
Utilizzare questa guida per configurare rapidamente IBM Advanced Message Security per fornire la sicurezza dei messaggi per le applicazioni Java che si collegano utilizzando i collegamenti client. Al momento del completamento, verrà creato un keystore per verificare le identità utente e le politiche di firma / crittografia definite per il gestore code.

Riferimenti correlati

“Limitazioni note” a pagina 274
Informazioni sulle limitazioni di IBM WebSphere MQ Advanced Message Security.

Disabilitazione di IBM WebSphere MQ Advanced Message Security sul client

È necessario disabilitare IBM WebSphere MQ Advanced Message Security (AMS) sul client se si utilizza un Version 7.5 o un client successivo per connettersi a un gestore code da una versione precedente del prodotto e viene riportato un errore 2085 (MQRC_UNKNOWN_OBJECT_NAME).

Informazioni su questa attività

Da Version 7.5, IBM WebSphere MQ Advanced Message Security (AMS) viene abilitato automaticamente in un client IBM WebSphere MQ , quindi, per impostazione predefinita, il client tenta di controllare le politiche di sicurezza per gli oggetti nel gestore code. Tuttavia, i server sulle versioni precedenti del prodotto, ad esempio Version 7.1, non hanno AMS abilitato, che causa un errore 2085 (MQRC_UNKNOWN_OBJECT_NAME).

Se questo errore viene riportato, quando si tenta di connettersi a un gestore code da una versione precedente del prodotto, è possibile disabilitare AMS sul client nel modo seguente:

- Per i client Java , in uno dei seguenti modi:
  - V7.5.0.4 Impostando una variabile di ambiente AMQ_DISABLE_CLIENT_AMS.
  - V7.5.0.4 Impostando la proprietà di sistema Java com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS.
  - V7.5.0.5 Utilizzando la proprietà AMS DisableClient, nella stanza **Security** del file mqclient.ini .
- Per i client C, in uno dei modi riportati di seguito:
  - V7.5.0.4 Impostando una variabile di ambiente AMQ_DISABLE_CLIENT_AMS.
  - V7.5.0.5 Utilizzando la proprietà AMS DisableClient, nella stanza **Security** del file mqclient.ini .

Procedura

- Per disabilitare AMS sul client, utilizzare una delle seguenti opzioni:

  V 7.5.0.4 Variabile di ambiente AMQ_DISABLE_CLIENT_AMS

  È necessario impostare questa variabile nei seguenti casi:

  – Se si utilizza JRE (Java Runtime Environment) diverso da JRE (IBM Java Runtime Environment)

  – Se si utilizza il client Version 7.5, o successivo, IBM WebSphere MQ classes for Java o IBM WebSphere MQ classes for JMS .

  Inoltre, è possibile utilizzare AMQ_DISABLE_CLIENT_AMS per disattivare la funzione AMS per i client C.

  Creare la variabile di ambiente AMQ_DISABLE_CLIENT_AMS e impostarla su TRUE nell'ambiente in cui è in esecuzione l'applicazione. Ad esempio:

  ```
  export AMQ_DISABLE_CLIENT_AMS=TRUE
  ```

  V 7.5.0.4 Proprietà di sistema com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS

  Per i client IBM WebSphere MQ classes for JMS e IBM WebSphere MQ classes for Java , impostare la proprietà di sistema Java com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS sul valore TRUE per l'applicazione Java .

  Ad esempio, è possibile impostare la proprietà di sistema Java come opzione `-D` quando viene richiamato il comando Java :

  ```
  java -Dcom.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS=TRUE -cp <MQ_INSTALLATION_PATH>/java/lib/
  com.ibm.mqjms.jar  my.java.applicationClass
  ```

  In alternativa, è possibile specificare la proprietà di sistema Java all'interno di un JMS file di configurazione, `jms.config`, se l'applicazione utilizza questo file.

  V 7.5.0.5 Proprietà `DisableClientAMS` nel file `mqclient.ini`

  Per i client IBM WebSphere MQ classes for JMS e IBM WebSphere MQ classes for Java , e per i client C, aggiungere il nome proprietà `DisableClientAMS` nella sezione **`Security`** del file `mqclient.ini` come mostrato nel seguente esempio:

  ```
  Security:
  DisableClientAMS=Yes
  ```

  È anche possibile abilitare AMS come mostrato nel seguente esempio:

  ```
  Security:
  DisableClientAMS=No
  ```

Operazioni successive

Per ulteriori informazioni sui problemi relativi all'apertura delle code protette AMS , consultare “Problemi nell'apertura di code protette quando si usa JMS” a pagina 325.

Concetti correlati

“Intercettazione MCA (Message Channel Agent)” a pagina 298

L'intercettazione MCA consente a un gestore code in esecuzione in IBM WebSphere MQ di abilitare in modo selettivo le politiche da applicare ai canali di connessione server.

Attività correlate

Configurazione di un client utilizzando un file di configurazione

Riferimenti correlati

Il file di configurazione IBM WebSphere MQ classes for JMS

Requisiti del certificato per AMS

I certificati devono avere una chiave pubblica RSA per poter essere utilizzati con Advanced Message Security.

Per ulteriori informazioni sui diversi tipi di chiave pubblica e su come crearli, consultare “Certificati digitali e compatibilità CipherSpec in IBM WebSphere MQ” a pagina 34.

Estensioni utilizzo chiave

Le estensioni di utilizzo delle chiavi pongono ulteriori restrizioni sul modo in cui un certificato può essere utilizzato.

In Advanced Message Security, l'utilizzo della chiave deve essere impostato come segue: per i certificati nello standard X.509 V3 o successivo utilizzati per la qualità dell'integrità della protezione, se le estensioni di utilizzo della chiave sono impostate, devono includere almeno uno dei due:

- **`nonRepudiation`**
- **`digitalSignature`**

Per la qualità della protezione della privacy, se sono impostate le estensioni di utilizzo chiave, devono includere anche l'estensione **`keyEncipherment`** .

Concetti correlati

“QoP (Quality of protection)” a pagina 312
Le politiche di protezione dati Advanced Message Security implicano una qualità di protezione (QOP).

Metodi di convalida dei certificati in IBM WebSphere MQ Advanced Message Security

È possibile utilizzare IBM WebSphere MQ Advanced Message Security per rilevare e rifiutare certificati revocati in modo che i messaggi sulle code non siano protetti utilizzando certificati che non soddisfano gli standard di sicurezza.

IBM WebSphere MQ AMS consente di verificare la validità di un certificato utilizzando OCSP (Online Certificate Status Protocol) o CRL (Certificate Revocation List).

IBM WebSphere MQ AMS può essere configurato per il controllo OCSP o CRL o per entrambi. Se entrambi i metodi sono abilitati, per motivi di prestazioni, IBM WebSphere MQ AMS utilizza prima OCSP per lo stato di revoca. Se lo stato di revoca di un certificato non è determinato dopo il controllo OCSP, IBM WebSphere MQ AMS utilizza il controllo CRL.

Concetti correlati

“OCSP (Certificate Status Protocol Online)” a pagina 302
L'OCSP (Online Certificate Status Protocol) determina se un certificato è stato revocato e, pertanto, consente di determinare se il certificato può essere considerato attendibile.

“CRL (Certificate Revocation Lists)” a pagina 304
I CRL contengono un elenco di certificati che sono stati contrassegnati dalla CA (Certificate Authority) come non più attendibili per una serie di motivi, ad esempio, la chiave privata è stata persa o compromessa.

OCSP (Certificate Status Protocol Online)

L'OCSP (Online Certificate Status Protocol) determina se un certificato è stato revocato e, pertanto, consente di determinare se il certificato può essere considerato attendibile.

Abilitazione della verifica OCSP negli intercettori nativi
Per abilitare la verifica OCSP (Online Certificate Status Protocol) in Advanced Message Security, è necessario modificare il file di configurazione del keystore.

Procedura

Aggiungere le seguenti opzioni al file di configurazione del keystore:

Nota: I valori per le singole opzioni forniti nella tabella sono predefiniti.

È necessario specificare uno dei seguenti valori:

- `ocsp.enable=on`
- `ocsp.url=<reposnder_URL>`
- `ocsp.http.proxy.host=<OCSP_proxy>`

Opzione	Descrizione
`ocsp.enable=off`	Abilitare la verifica OCSP se il certificato da verificare ha un'estensione Authority Info Access con un metodo di accesso PKIX_AD_OCSP contenente un URI dell'ubicazione del responder OCSP. Valori possibili: on/`off`.
`ocsp.url=<reposnder_URL>`	L'indirizzo URL del responder OCSP.
`ocsp.http.proxy.host=<OCSP_proxy>`	L'indirizzo URL del server proxy OCSP.
`ocsp.http.proxy.port=<port_number>`	Il numero di porta del server proxy OCSP.
`ocsp.nonce.generation=on/off`	Generare un parametro nonce durante le query su OCSP. Il valore predefinito è `off`.
`ocsp.nonce.check=on/off`	Verificare il parametro nonce dopo la ricezione di una risposta da OCSP. Il valore predefinito è `off`.
`ocsp.nonce.size=8`	Dimensione nonce in byte.
`ocsp.http.get=on/off`	Specificare HTTP GET come metodo di richiesta. Se questa opzione è impostata su `off`, viene utilizzato HTTP POST.
`ocsp.max_response_size=20480`	Dimensione massima della risposta dal responder OCSP fornita in byte.
`ocsp.cache_size=100`	Abilitare la memorizzazione della risposta OCSP interna nella cache e impostare il limite per il numero di voci della cache.
`ocsp.timeout=30`	Tempo di attesa per una risposta del server, in secondi, dopo il quale Advanced Message Security va in timeout.

Abilitazione del controllo OCSP in Java
Per abilitare il checkin OCSP per Java in Advanced Message Security, modificare il file `java.security` o il file di configurazione del keystore.

Informazioni su questa attività

Esistono due modi per abilitare il controllo OCSP in Advanced Message Security:

Utilizzo di java.security
Verificare se il certificato dispone di AIA (Authority Information Access) impostato.

Procedura

1. Se AIA non è impostato o se si desidera sovrascrivere il proprio certificato, modificare il file `$JAVA_HOME/lib/security/java.security` con le seguenti proprietà:

   ```
   ocsp.responderURL=http://url.to.responder:port
   ocsp.responderCertSubjectName=CN=Example CA,O=IBM,C=US
   ```

   e abilitare il controllo OCSP modificando il file `$JAVA_HOME/lib/security/java.security` con la seguente riga:

   ```
   ocsp.enable=true
   ```

2. Se AIA è impostato, abilitare il controllo OCSP modificando il file `$JAVA_HOME/lib/security/java.security` con la riga seguente:

   ```
   ocsp.enable=true
   ```

Operazioni successive

Se si sta utilizzando Java Security Manager, completare troppo la configurazione, aggiungere la seguente autorizzazione Java a `lib/security/java.policy`

```
permission java.security.SecurityPermission "getProperty.ocsp.enable";
```

Utilizzo di keystore.conf

Procedura

Aggiungere il seguente attributo al file di configurazione:

```
ocsp.enable=true
```

Importante: L'impostazione di questo attributo nel file di configurazione sostituisce le impostazioni java.security .

Operazioni successive

Per completare la configurazione, aggiungere le seguenti autorizzazioni Java a `lib/security/java.policy`:

```
permission java.security.SecurityPermission "getProperty.ocsp.enable";
permission java.security.SecurityPermission "setProperty.ocsp.enable";
```

CRL (Certificate Revocation Lists)

I CRL contengono un elenco di certificati che sono stati contrassegnati dalla CA (Certificate Authority) come non più attendibili per una serie di motivi, ad esempio, la chiave privata è stata persa o compromessa.

Per convalidare i certificati, Advanced Message Security crea una catena di certificati costituita dal certificato del firmatario e dalla catena di certificati della CA (Certificate Authority) fino a un ancoraggio di trust. Un ancoraggio di trust è un file keystore sicuro che contiene un certificato attendibile o un certificato root attendibile utilizzato per asserire l'attendibilità di un certificato.IBM WebSphere MQ AMS verifica il percorso del certificato utilizzando un algoritmo di convalida PKIX. Quando il concatenamento viene creato e verificato, IBM WebSphere MQ AMS completa la convalida del certificato che include la convalida della data di emissione e di scadenza di ciascun certificato nel concatenamento rispetto alla data corrente, controllando se l'estensione di utilizzo della chiave è presente nel certificato di entità finale. Se l'estensione viene aggiunta al certificato, IBM WebSphere MQ AMS verifica se sono impostati anche **digitalSignature** o **nonRepudiation** . In caso contrario, il `MQRC_SECURITY_ERROR` viene riportato e registrato. In seguito, IBM WebSphere MQ AMS scarica i CRL dai file o da LDAP in base ai valori specificati nel file di configurazione. Solo i CRL codificati in formato DER sono supportati da IBM

WebSphere MQ AMS. Se non viene trovata alcuna configurazione relativa al CRL nel file di configurazione del keystore, IBM WebSphere MQ AMS non esegue alcun controllo di validità CRL. Per ogni certificato CA, IBM WebSphere MQ AMS esegue una query LDAP per i CRL utilizzando i DN (Distinguished Name) di una CA per trovare il relativo CRL. I seguenti attributi sono inclusi nella query LDAP:

```
certificateRevocationList,
certificateRevocationList;binary,
authorityRevocationList,
authorityRevocationList;binary
deltaRevocationList
deltaRevocationList;binary,
```

Nota: `deltaRevocationList` è supportata solo quando è specificato come punti di distribuzione.

Abilitazione della convalida del certificato e del supporto CRL (Certificate Revocation List) negli intercettatori nativi
È necessario modificare il file di configurazione del keystore in modo che Advanced Message Security possa scaricare i CLR dal server LDAP (Lightweight Directory Access Protocol).

Procedura

Aggiungere le seguenti opzioni al file di configurazione:

Nota: I valori per le singole opzioni forniti nella tabella sono predefiniti.

Opzione	**Descrizione**
`crl.ldap.host=<`*`host_name`*`>`	Nome host del server LDAP.
`crl.ldap.port=<`*`port_number`*`>`	Numero di porta del server LDAP. È possibile specificare fino a 11 server. Sono utilizzati più host LDAP per assicurare il failover trasparente in caso di errore di collegamento LDAP. Si prevede che tutti i server LDAP siano repliche e contengano gli stessi dati. Quando l'interceptor Java di IBM WebSphere MQ AMS si connette correttamente a un server LDAP, non tenta di scaricare i CRL dai restanti server forniti.
`crl.cdp=off`	Utilizzare questa opzione per controllare o utilizzare le estensioni CRLDistributionPoints nei certificati.
`crl.ldap.version=3`	Numero di versione protocollo LDAP. Valori possibili: 2 o 3.
`crl.ldap.user=cn=<`*`username`*`>`	Accedere al server LDAP. Se questo valore non è specificato, gli attributi CRL in LDAP devono essere leggibili
`crl.ldap.pass=<`*`password`*`>`	Password per il server LDAP.
`crl.ldap.cache_lifetime=0`	Durata della cache LDAP in secondi. Valori possibili: 0-86400.
`crl.ldap.cache_size=50`	Dimensione cache LDAP. Questa opzione può essere specificata solo se il valore `crl.ldap.cache_lifetime` è maggiore di `0`.
`crl.http.proxy.host=some.host.com`	Porta del server proxy Http per il recupero CRL CDP.
`crl.http.proxy.port=8080`	Numero di porta del server proxy Http.

Opzione	Descrizione
`crl.http.max_response_size=204800`	La dimensione massima di CRL, in byte, che può essere richiamata da un server HTTP accettato da GSKit.
`crl.http.timeout=30`	Il tempo di attesa per una risposta del server, espresso in secondi, dopo il quale IBM WebSphere MQ AMS va in timeout.
`crl.http.cache_size=0`	Dimensione cache HTTP, in byte.

Abilitazione del supporto CRL (Certificate Revocation List) in Java
Per abilitare il supporto CRL in Advanced Message Security, è necessario modificare il file di configurazione del keystore per consentire a IBM WebSphere MQ AMS di scaricare i CRL dal server LDAP (Lightweight Directory Access Protocol) e configurare il file java.security .

Procedura

1. Aggiungere le seguenti opzioni al file di configurazione:

Intestazione	Descrizione
`crl.ldap.host=<`*`host_name`*`>`	Nome host LDAP.
`crl.ldap.port=<`*`port_number`*`>`	Numero di porta del server LDAP. È possibile specificare fino a 11 server. Sono utilizzati più host LDAP per assicurare il failover trasparente in caso di errore di collegamento LDAP. Si prevede che tutti i server LDAP siano repliche e contengano gli stessi dati. Quando l'interceptor Java di IBM WebSphere MQ AMS si connette correttamente a un server LDAP, non tenta di scaricare i CRL dai restanti server forniti. Java non utilizza valori `crl.ldap.user` e crl.ldaworldp.pass . Non utilizza un utente e una password durante la connessione a un server LDAP. Di conseguenza, gli attributi CRL in LDAP devono essere leggibili.
`crl.cdp=on/off`	Utilizzare questa opzione per controllare o utilizzare le estensioni CRLDistributionPoints nei certificati.

2. Modificare il file `JRE/lib/security/java.security` con le seguenti proprietà:

Nome proprietà	Descrizione
`com.ibm.security.enableCRLDP`	Questa proprietà assume i seguenti valori: `true`, `false`. Se è impostato su `true`, durante il controllo della revoca del certificato, i CRL vengono individuati utilizzando l'URL dall'estensione dei punti di distribuzione CRL del certificato. Se è impostato su `false` o non è impostato, il controllo di CRL utilizzando l'estensione dei punti di distribuzione CRL è disabilitato.

Nome proprietà	Descrizione
`ibm.security.certpath.ldap.cache.lifetime`	Questa proprietà può essere utilizzata per impostare la durata delle voci nella cache di memoria di LDAP CertStore su un valore in secondi. Il valore 0 disabilita la cache; -1 indica una durata illimitata. Se non è impostato, la durata predefinita è di 30 secondi.
`com.ibm.security.enableAIAEXT`	Questa proprietà assume i seguenti valori: `true`, `false`. Se è impostato su `true`, tutte le estensioni di accesso alle informazioni dell'autorità che si trovano all'interno dei certificati del percorso del certificato in fase di creazione vengono esaminate per determinare se contengono URI LDAP. Per ciascun URI LDAP trovato, viene creato un oggetto LDAPCertStore e aggiunto alla raccolta di CertStores utilizzata per individuare altri certificati richiesti per creare il percorso del certificato. Se è impostato su `false` o non è impostato, non vengono creati ulteriori oggetti LDAPCertStore .

Protezione delle password in Java

La memorizzazione delle password del keystore e della chiave privata come testo semplice rappresenta un rischio per la sicurezza, pertanto Advanced Message Security fornisce uno strumento che può codificare tali password utilizzando una chiave utente, disponibile nel file keystore.

Prima di iniziare

Il proprietario del file `keystore.conf` deve garantire che solo il proprietario del file sia autorizzato a leggerlo. La protezione delle password descritta in questo capitolo è solo una misura aggiuntiva di protezione.

Procedura

1. Modificare i file `keystore.conf` per includere il percorso al keystore e l'etichetta utente.

   ```
   jceks.keystore = c:/Documents and Setting/Alice/AliceKeystore
   jceks.certificate = AliceCert
   jceks.provider = IBMJCE
   ```

2. Per eseguire lo strumento, immettere:

   ```
   java -cp com.ibm.mq.jmqi.jar com.ibm.mq.ese.config.KeyStoreConfigProtector keystore_password
   private_key_password
   ```

   Viene generato un output con password codificate che è possibile copiare nel file `keystore.conf` .

   Per copiare automaticamente l'output nel file `keystore.conf` , eseguire:

   ```
   java -cp com.ibm.mq.jmqi.jar com.ibm.mq.ese.config.KeyStoreConfigProtector keystore_password
   private_key_password >> ~/<path_to_keystore>/keystore.conf
   ```

   Nota:

   Per un elenco di ubicazioni predefinite di `keystore.conf` su varie piattaforme, consultare “Utilizzo di keystore e certificati” a pagina 296.

Esempio

Di seguito è riportato un esempio di tale output:

```
#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
```

Amministrazione delle politiche di protezione IBM WebSphere MQ Advanced Message Security

IBM WebSphere MQ Advanced Message Security utilizza le politiche di sicurezza per specificare la crittografia e gli algoritmi di firma per codificare e autenticare i messaggi che passano attraverso le code.

Panoramica delle politiche di sicurezza

Le politiche di sicurezza IBM Advanced Message Security sono oggetti concettuali che descrivono il modo in cui un messaggio viene crittografato e firmato.

Per i dettagli relativi agli attributi della politica di sicurezza, consultare i seguenti topic secondari:

Concetti correlati

"QoP (Quality of protection)" a pagina 312
Le politiche di protezione dati Advanced Message Security implicano una qualità di protezione (QOP).

"Attributi della politica di protezione" a pagina 311
È possibile utilizzare Advanced Message Security per selezionare un particolare algoritmo o metodo per proteggere i dati.

Nome politica

Il nome della politica è un nome univoco che identifica una politica Advanced Message Security specifica e la coda a cui si applica.

Il nome della politica deve corrispondere al nome della coda a cui si applica. Esiste un'associazione uno a uno tra una politica Advanced Message Security (IBM WebSphere MQ AMS) e una coda.

Creando una politica con lo stesso nome di una coda, si attiva la politica per tale coda. Le code senza nomi di politica corrispondenti non vengono protette da IBM WebSphere MQ AMS.

L'ambito della politica è rilevante per il gestore code locale e le relative code. I gestori code remoti devono avere le proprie politiche definite localmente per le code che gestiscono.

Algoritmo della firma

L'algoritmo di firma indica l'algoritmo da utilizzare quando si firmano i messaggi di dati.

Valori validi includono:

- MD5
- SHA-1
- SHA-2 Famiglia:
  - SHA256
  - SHA384 (lunghezza minima della chiave accettabile - 768 bit)
  - SHA512 (lunghezza chiave minima accettabile - 768 bit)

Una politica che non specifica un algoritmo di firma o un algoritmo di NONE, implica che i messaggi collocati nella coda associata alla politica non siano firmati.

Nota: La qualità di protezione utilizzata per le funzioni put e get del messaggio deve corrispondere. Se c'è una mancata corrispondenza della qualità di protezione della politica tra la coda e il messaggio nella coda, il messaggio non viene accettato e viene inviato alla coda di gestione errori. Questa regola si applica sia alle code locali che a quelle remote.

Algoritmo di codifica

L'algoritmo di codifica indica l'algoritmo da utilizzare quando si codificano i messaggi di dati posizionati sulla coda associata alla politica.

Valori validi includono:

- RC2
- DES
- 3DES
- AES128
- AES256

Una politica che non specifica un algoritmo di codifica o un algoritmo di NONE implica che i messaggi collocati nella coda associata alla politica non vengono codificati.

Tenere presente che una normativa che specifica un algoritmo di cifratura diverso da NONE deve specificare anche almeno un DN destinatario e un algoritmo di firma poiché anche i messaggi cifrati Advanced Message Security sono firmati.

Importante: La qualità di protezione utilizzata per le funzioni put e get del messaggio deve corrispondere. Se c'è una mancata corrispondenza della qualità di protezione della politica tra la coda e il messaggio nella coda, il messaggio non viene accettato e viene inviato alla coda di gestione errori. Questa regola si applica sia alle code locali che a quelle remote.

Tolleranza

L'attributo di tolleranza indica se IBM Advanced Message Security può accettare messaggi senza alcuna politica di sicurezza specificata.

Quando si richiama un messaggio da una coda con una politica per codificare i messaggi, se il messaggio non è codificato, viene restituito all'applicazione chiamante. Valori validi includono:

0
No (**predefinito**).

1
Sì.

Una politica che non specifica un valore di tolleranza o specifica 0, implica che i messaggi collocati nella coda associata alla politica devono corrispondere alle regole della politica.

La tolleranza è facoltativa ed esiste per facilitare l'implementazione della configurazione, dove le politiche sono state applicate alle code ma quelle code contengono già messaggi che non hanno una politica di sicurezza specificata.

Nomi distinti del mittente

I DN (distinguished name) del mittente identificano gli utenti autorizzati a inserire messaggi in una coda.

IBM Advanced Message Security (IBM WebSphere MQ AMS) non controlla se un messaggio è stato inserito in una coda protetta da dati da un utente valido fino a quando il messaggio non viene richiamato. A questo punto, se la politica stabilisce uno o più mittenti validi e l'utente che ha inserito il messaggio nella coda non si trova nell'elenco di mittenti validi, IBM WebSphere MQ AMS restituisce un errore all'applicazione di acquisizione e inserisce il messaggio nella sua coda errori.

In una politica possono essere specificati 0 o più DN mittente. Se per la politica non viene specificato alcun DN mittente, qualsiasi utente può inserire i messaggi con dati protetti alla coda a condizione che il certificato dell'utente sia attendibile.

I nomi distinti del mittente hanno la seguente forma:

```
CN=Common Name,O=Organization,C=Country
```

Importante:

- Tutti i DN devono essere in maiuscolo. Tutti gli identificativi di nome componente nel DN devono essere specificati nell'ordine mostrato nella tabella seguente:

Nome componente	Valore
CN	Il nome comune per l'oggetto di questo DN, ad esempio un nome completo o lo scopo previsto di una periferica.
OU	L'unità all'interno dell'organizzazione con cui è affiliato l'oggetto del DN, ad esempio una divisione aziendale o un nome prodotto.
O	L'organizzazione a cui è affiliato l'oggetto del DN, ad esempio una società.
L	La località (città o comune) in cui si trova l'oggetto del DN.
ST	Il nome dello stato o della provincia in cui si trova l'oggetto del DN.
C	Il paese in cui si trova l'oggetto del DN (distinguished name).

- Se per la politica vengono specificati uno o più DN mittente, solo quegli utenti possono inserire i messaggi alla coda associata alla politica.
- I DN del mittente, quando specificati, devono corrispondere esattamente al DN contenuto nel certificato digitale associato all'utente che inserisce il messaggio.
- IBM WebSphere MQ AMS supporta i DN con valori solo dalla serie di caratteri Latin-1 . Per creare DN con caratteri della serie, è necessario prima creare un certificato con un DN creato nella codifica UTF-8 utilizzando le piattaforme UNIX con la codifica UTF-8 attivata o con il programma di utilità iKeyman . È necessario quindi creare una politica da una piattaforma UNIX con la codifica UTF-8 attivata oppure utilizzare il plug-in IBM WebSphere MQ AMS per WebSphere MQ.

Concetti correlati

“Nomi distinti del destinatario” a pagina 310
Il DN (distinguished name) del destinatario identifica gli utenti autorizzati a richiamare i messaggi da una coda.

Nomi distinti del destinatario

Il DN (distinguished name) del destinatario identifica gli utenti autorizzati a richiamare i messaggi da una coda.

In una politica possono essere specificati zero o più DN destinatari. I nomi distinti dei destinatari hanno il seguente formato:

```
CN=Common Name,O=Organization,C=Country
```

Importante:

- Tutti i DN devono essere in maiuscolo. Tutti gli identificativi di nome componente nel DN devono essere specificati nell'ordine mostrato nella tabella seguente:

Nome componente	Valore
CN	Il nome comune per l'oggetto di questo DN, ad esempio un nome completo o lo scopo previsto di una periferica.
OU	L'unità all'interno dell'organizzazione con cui è affiliato l'oggetto del DN, ad esempio una divisione aziendale o un nome prodotto.
O	L'organizzazione a cui è affiliato l'oggetto del DN, ad esempio una società.
L	La località (città o comune) in cui si trova l'oggetto del DN.
ST	Il nome dello stato o della provincia in cui si trova l'oggetto del DN.
C	Il paese in cui si trova l'oggetto del DN (distinguished name).

- Se per la politica non viene specificato alcun DN destinatario, qualsiasi utente può acquisire i messaggi dalla coda associata alla politica.
- Se per la politica vengono specificati uno o più DN destinatario, solo quegli utenti possono recuperare i messaggi dalla coda associata alla politica.
- I DN del destinatario, quando specificati, devono corrispondere esattamente al DN contenuto nel certificato digitale associato all'utente che recupera il messaggio.
- Advanced Message Security supporta i DN con valori solo dalla serie di caratteri Latin-1 . Per creare DN con caratteri della serie, è necessario prima creare un certificato con un DN creato nella codifica UTF-8 utilizzando le piattaforme UNIX con la codifica UTF-8 attivata o con il programma di utilità iKeyman . È necessario quindi creare una politica da una piattaforma UNIX con la codifica UTF-8 attivata oppure utilizzare il plug-in Advanced Message Security per WebSphere MQ.

Concetti correlati

“Nomi distinti del mittente” a pagina 309
I DN (distinguished name) del mittente identificano gli utenti autorizzati a inserire messaggi in una coda.

Attributi della politica di protezione

È possibile utilizzare Advanced Message Security per selezionare un particolare algoritmo o metodo per proteggere i dati.

Una politica di protezione è un oggetto concettuale che descrive il modo in cui un messaggio viene crittograficamente crittografato e firmato. La seguente tabella presenta gli attributi della politica di sicurezza in Advanced Message Security:

Attributi	Descrizione
Nome politica	Nome univoco della politica per un gestore code.
Algoritmo della firma	Algoritmo crittografico utilizzato per firmare i messaggi prima dell'invio.
Algoritmo di codifica	Algoritmo crittografico utilizzato per codificare i messaggi prima dell'invio.
Elenco destinatari	Elenco dei DN (distinguished name) dei potenziali destinatari di un messaggio.
Elenco di controllo DN firma	Elenco di DN firma da convalidare durante il recupero del messaggio.

In Advanced Message Security, i messaggi vengono codificati con una chiave simmetrica e la chiave simmetrica viene codificata con le chiavi pubbliche dei destinatari. Le chiavi pubbliche sono crittografate con l'algoritmo RSA, con chiavi di lunghezza effettiva fino a 2048 bit. L'effettiva codifica della chiave asimmetrica dipende dalla lunghezza della chiave del certificato.

Gli algoritmi di chiave simmetrica supportati sono i seguenti:

- RC2
- DES
- 3DES
- AES128
- AES256

Advanced Message Security supporta anche le seguenti funzioni hash crittografiche:

- MD5
- SHA-1
- SHA-2 Famiglia:
  - SHA256
  - SHA384 (lunghezza minima della chiave accettabile - 768 bit)
  - SHA512 (lunghezza chiave minima accettabile - 768 bit)

Nota: La qualità di protezione utilizzata per le funzioni put e get del messaggio deve corrispondere. Se c'è una mancata corrispondenza della qualità di protezione della politica tra la coda e il messaggio nella coda, il messaggio non viene accettato e viene inviato alla coda di gestione errori. Questa regola si applica sia alle code locali che a quelle remote.

QoP (Quality of protection)

Le politiche di protezione dati Advanced Message Security implicano una qualità di protezione (QOP).

I tre livelli di qualità di protezione in Advanced Message Security dipendono da algoritmi crittografici utilizzati per firmare e codificare i messaggi:

- Privacy - i messaggi inseriti nella coda devono essere firmati e codificati.
- Integrità - i messaggi inseriti nella coda devono essere firmati dal mittente.
- Nessuno - nessuna protezione dei dati è applicabile.

Una normativa che stabilisce che i messaggi devono essere firmati quando inseriti in una coda hanno un QOP di INTEGRITY. Un QOP di INTEGRITY significa che una normativa stabilisce un algoritmo di firma, ma non stabilisce un algoritmo di codifica. I messaggi protetti dall'integrità vengono anche indicati come "SIGNED".

Una normativa che stabilisce che i messaggi devono essere firmati e codificati quando vengono inseriti in una coda ha un QOP di PRIVACY. Un QOP di PRIVACY significa che quando una normativa stabilisce un algoritmo di firma e un algoritmo di cifratura. I messaggi protetti dalla privacy sono anche indicati come "SIGILLATI".

Una normativa che non stipula un algoritmo di firma o un algoritmo di crittografia ha un QOP di NONE. Advanced Message Security non fornisce protezione dati per le code che hanno una normativa con QOP NONE.

Gestione delle politiche di sicurezza

Una politica di protezione è un oggetto concettuale che descrive il modo in cui un messaggio viene crittograficamente crittografato e firmato.

Tutte le attività amministrative relative alle politiche di sicurezza vengono eseguite dalla seguente ubicazione:

- Su piattaforme UNIX : `<MQInstallRoot>/bin`

- Su piattaforme Windows , le attività di gestione possono essere eseguite da qualsiasi ubicazione, poiché la variabile di ambiente PATH viene aggiornata durante l'installazione.

Attività correlate

“Creazione di politiche di sicurezza” a pagina 313
Le politiche di sicurezza definiscono il modo in cui un messaggio viene protetto quando viene inserito o il modo in cui un messaggio deve essere protetto quando viene ricevuto.

“Modifica delle politiche di sicurezza” a pagina 314
È possibile utilizzare Advanced Message Security per modificare dettagli delle politiche di sicurezza già definite.

“Visualizzazione e dump delle politiche di sicurezza” a pagina 314
Utilizzare il comando `dspmqspl` per visualizzare un elenco di tutte le politiche di sicurezza o i dettagli di una politica definita in base ai parametri della riga comandi forniti.

“Rimozione delle politiche di sicurezza” a pagina 316
Per rimuovere le politiche di sicurezza in Advanced Message Security, è necessario utilizzare il comando `setmqspl` .

Creazione di politiche di sicurezza

Le politiche di sicurezza definiscono il modo in cui un messaggio viene protetto quando viene inserito o il modo in cui un messaggio deve essere protetto quando viene ricevuto.

Prima di iniziare

Ci sono alcune condizioni di entrata che devono essere soddisfatte quando si creano le politiche di sicurezza:

- Il gestore code deve essere in esecuzione.
- Il nome di una politica di sicurezza deve seguire le regole per la denominazione degli oggetti WebSphere MQ.
- È necessario disporre delle autorizzazioni `+connect +inq +chg` necessarie per creare una politica di sicurezza. Per la sintassi completa del comando di modifica autorizzazione, consultare **`setmqaut`** .
- Accertarsi di disporre delle autorizzazioni necessarie per operare sulle code e sui gestori code di WebSphere MQ . Per ulteriori informazioni, consultare “Concessione delle autorizzazioni OAM” a pagina 317

Esempio

Di seguito è riportato un esempio di creazione di un criterio sul gestore code `QMGR`. La politica specifica che i messaggi vengano firmati utilizzando l'algoritmo `SHA1` e codificati utilizzando l'algoritmo `AES256` per certificati con DN: `CN=joe, O=IBM, C=US` e DN: `CN=jane, O=IBM, C = US`. Questa politica è allegata 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
```

Di seguito è riportato un esempio di creazione della politica sul gestore code `QMGR`. La politica specifica che i messaggi vengono codificati utilizzando l'algoritmo `DES` per i certificati con DN: `CN=john, O=IBM, C=US` e `CN=jeff, O=IBM, C=US` e firmati con l'algoritmo `MD5` per i certificati con DN: `CN=phil, O=IBM, C=US`

```
$ setmqspl -m QMGR -p MY.OTHER.QUEUE -s MD5 -e DES -r CN=john,O=IBM,C=US -r CN=jeff,O=IBM,C=US
-a CN=phil,O=IBM,C=US
```

Nota:

- La qualità della protezione utilizzata per l'inserimento e l'acquisizione del messaggio deve corrispondere. Se la qualità di protezione della politica definita per il messaggio è più debole di quella definita per una coda, il messaggio viene inviato alla coda di gestione errori. Questa normativa è valida sia per le code locali che remote.

Riferimenti correlati

Elenco completo degli attributi del comando setmqspl

Modifica delle politiche di sicurezza

È possibile utilizzare Advanced Message Security per modificare dettagli delle politiche di sicurezza già definite.

Prima di iniziare

- Il gestore code su cui si desidera operare deve essere in esecuzione.
- È necessario disporre delle autorizzazioni `+connect +inq +chg` necessarie per creare le normative di sicurezza. Per la sintassi completa del comando di modifica autorizzazione, consultare **setmqaut** .

Informazioni su questa attività

Per modificare le politiche di sicurezza, applicare il comando `setmqspl` a una politica già esistente che fornisce nuovi attributi.

Esempio

Di seguito viene riportato un esempio di creazione di una politica denominata MYQUEUE su un gestore code denominato QMGR che specifica che i messaggi verranno codificati utilizzando l'algoritmo RC2 per i certificati con DN:CN=bob, O=IBM, C=US e firmati con l'algoritmo SHA1 per certificati con DN:CN=jeff, O=IBM, C = US.

```
setmqspl -m QMGR -p MYQUEUE -e RC2 -s SHA1 -a CN=jeff,O=IBM,C=US -r CN=alice,O=IBM,C=US
```

Per modificare questa politica, emettere il comando `setmqspl` con tutti gli attributi dell'esempio modificando solo i valori che si desidera modificare. In questo esempio, la politica creata in precedenza viene allegata a una nuova coda e il suo algoritmo di codifica viene modificato in AES256:

```
setmqspl -m QMGR -p MYQUEUE -e AES256 -s SHA1 -a CN=jeff,O=IBM,C=US -r CN=alice,O=IBM,C=US
```

Riferimenti correlati

setmqspl

Visualizzazione e dump delle politiche di sicurezza

Utilizzare il comando `dspmqspl` per visualizzare un elenco di tutte le politiche di sicurezza o i dettagli di una politica definita in base ai parametri della riga comandi forniti.

Prima di iniziare

- Per visualizzare i dettagli delle politiche di sicurezza, il gestore code deve esistere ed essere in esecuzione.
- È necessario disporre delle autorizzazioni `+connect +inq +dsp` necessarie applicate a un gestore code per visualizzare ed eseguire il dump delle politiche di sicurezza. Per la sintassi completa del comando di modifica autorizzazione, consultare **setmqaut** .

Informazioni su questa attività

Di seguito è riportato l'elenco di indicatori di comando `dspmqspl` :

Tabella 27. indicatori del comando `dspmqspl` .

Indicatore comando	**Spiegazione**
`-m`	Nome gestore code (**obbligatorio**).
`-p`	Il nome della politica.

Tabella 27. indicatori del comando dspmqspl . (Continua)	
Indicatore comando	**Spiegazione**
-export	L'aggiunta di questo indicatore genera un output che può essere facilmente applicato a un gestore code differente.

Esempio

In questo esempio verranno create due politiche di sicurezza per 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
```

Questo esempio mostra un comando che visualizza i dettagli di tutte le politiche definite per venus.queue.manager e l'output che 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
```

Questo esempio mostra un comando che visualizza i dettagli di una politica di sicurezza selezionata definita per venus.queue.manager e l'emissione che 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
```

Nell'esempio successivo, creiamo prima una politica di sicurezza e poi la esportiamo utilizzando l'indicatore -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]
```

Per importare una politica di sicurezza:

- Su piattaforme Windows , eseguire policies.bat
- Su piattaforme UNIX :
  1. Collegarsi come un utente che appartiene al gruppo di amministrazione mqm WebSphere MQ .
  2. Immettere . policies.sh.

Riferimenti correlati

Elenco completo degli attributi del comando dspmqspl

Rimozione delle politiche di sicurezza

Per rimuovere le politiche di sicurezza in Advanced Message Security, è necessario utilizzare il comando `setmqspl` .

Prima di iniziare

Esistono alcune condizioni di immissione che devono essere soddisfatte quando si gestiscono le politiche di sicurezza:

- Il gestore code deve essere in esecuzione.
- È necessario disporre delle autorizzazioni `+connect +inq +chg` necessarie per creare le normative di sicurezza. Per la sintassi completa del comando di modifica autorizzazione, consultare **`setmqaut`** .

Informazioni su questa attività

Utilizzare il comando `setmqspl` con l'opzione `-remove` .

Esempio

Di seguito è riportato un esempio di rimozione di una politica:

```
$ setmqspl -m QMGR -remove -p MY.OTHER.QUEUE
```

Riferimenti correlati

Elenco completo degli attributi del comando setmqspl

Protezione coda di sistema

Le code di sistema consentono la comunicazione tra WebSphere MQ e le relative applicazioni ausiliarie. Ogni volta che viene creato un gestore code, viene creata anche una coda di sistema per memorizzare i dati e i messaggi interni WebSphere MQ . È possibile proteggere le code di sistema con Advanced Message Security in modo che solo gli utenti autorizzati possano accedervi o decodificarle.

La protezione della coda di sistema segue lo stesso modello della protezione delle code regolari. Vedi “Creazione di politiche di sicurezza” a pagina 313.

Per utilizzare la protezione della coda di sistema sulle piatteforme Windows , copiare il file `keystore.conf` nella seguente directory:

```
c:\Documents and Settings\Default User\.mqs\keystore.conf
```

Per fornire la protezione per `SYSTEM.ADMIN.COMMAND.QUEUE`, il server dei comandi deve avere accesso a `keystore` e a `keystore.conf`, che contengono chiavi e una configurazione in modo che il server dei comandi possa accedere a chiavi e certificati. Tutte le modifiche apportate alla politica di sicurezza `SYSTEM.ADMIN.COMMAND.QUEUE` richiedono il riavvio del server dei comandi.

Tutti i messaggi inviati e ricevuti dalla coda comandi vengono firmati o firmati e codificati in base alle impostazioni della normativa. Se un amministratore definisce i firmatari autorizzati, i messaggi di comando che non passano il controllo DN (Distinguished Name) del firmatario non vengono eseguiti dal server dei comandi e non vengono instradati alla coda di gestione errori Advanced Message Security . I messaggi inviati come risposte alle code dinamiche temporanee di WebSphere MQ Explorer non sono protetti da WebSphere MQ AMS.

Le modifiche alle politiche di sicurezza di Advanced Message Security richiedono il riavvio del server dei comandi WebSphere MQ

Le politiche di sicurezza non hanno effetto sulle seguenti code SYSTEM:

- 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

Concessione delle autorizzazioni OAM

Le autorizzazioni file autorizzano tutti gli utenti a eseguire i comandi `setmqspl` e `dspmqspl` . Tuttavia, IBM Advanced Message Security si basa su OAM (Object Authority Manager) e ogni tentativo di eseguire questi comandi da parte di un utente che non appartiene al gruppo mqm, che è il gruppo di amministrazione WebSphere MQ , o che non dispone delle autorizzazioni per leggere le impostazioni della politica di sicurezza concesse, causa un errore.

Procedura

Per concedere le autorizzazioni necessarie a un utente, eseguire:

```
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
```

Eventi di comando e configurazione

Con Advanced Message Security, è possibile creare messaggi di eventi di configurazione e comandi, che possono essere registrati e utilizzati come record delle modifiche della politica per il controllo.

Gli eventi di comando e configurazione generati da WebSphere MQ sono messaggi del formato PCF inviati alle code dedicate.

I messaggi degli eventi di configurazione vengono inviati a SYSTEM.ADMIN.CONFIG.EVENT sul gestore code in cui si verifica l'evento.

I messaggi di eventi comando vengono inviati al SISTEMA SYSTEM.ADMIN.COMMAND.EVENT sul gestore code in cui si verifica l'evento.

Gli eventi vengono generati indipendentemente dagli strumenti che si stanno utilizzando per gestire le politiche di sicurezza Advanced Message Security .

In Advanced Message Security, esistono quattro tipi di eventi generati da diverse azioni sulle politiche di sicurezza:

- “Creazione di politiche di sicurezza” a pagina 313, che genera due messaggi di eventi WebSphere MQ :
  - Un evento di configurazione
  - Un evento di comando
- “Modifica delle politiche di sicurezza” a pagina 314, che genera tre messaggi di eventi WebSphere MQ :
  - Un evento di configurazione che contiene i valori della politica di sicurezza obsoleti
  - Un evento di configurazione che contiene nuovi valori della politica di sicurezza
  - Un evento di comando
- “Visualizzazione e dump delle politiche di sicurezza” a pagina 314, che crea un messaggio evento WebSphere MQ :
  - Un evento di comando
- “Rimozione delle politiche di sicurezza” a pagina 316, che genera due messaggi di eventi WebSphere MQ :
  - Un evento di configurazione
  - Un evento di comando

Abilitazione e disabilitazione della registrazione eventi

Gli eventi di comando e configurazione vengono controllati utilizzando gli attributi del gestore code CONFIGEV e CMDEV. Per abilitare questi eventi, impostare l'attributo del gestore code appropriato su ENABLED. Per disabilitare questi eventi, impostare l'attributo del gestore code appropriato su DISABLED.

Procedura

Eventi di configurazione

Per abilitare gli eventi di configurazione, impostare CONFIGEV su ENABLED. Per disabilitare gli eventi di configurazione, impostare CONFIGEV su DISABLED. Ad esempio, è possibile abilitare gli eventi di configurazione utilizzando il seguente comando MQSC:

```
ALTER QMGR CONFIGEV (ENABLED)
```

Eventi di comandi

Per abilitare gli eventi comando, impostare CMDEV su ENABLED. Per abilitare gli eventi comando per i comandi tranne i comandi DISPLAY MQSC e Inquire PCF, impostare CMDEV su NODISPLAY. Per disabilitare gli eventi comando, impostare CMDEV su DISABLED. Ad esempio, è possibile abilitare gli eventi comando utilizzando il seguente comando MQSC:

```
ALTER QMGR CMDEV (ENABLED)
```

Attività correlate

Controllo degli eventi di configurazione, comando e programma di registrazione in Websphere MQ

Formato messaggio evento comando

Il messaggio di evento di comando è composto dalla struttura MQCFH e dai parametri PCF che lo seguono.

Di seguito sono riportati i valori MQCFH selezionati:

```
Type = MQCFT_EVENT;
Command = MQCMD_COMMAND_EVENT;
MsgSeqNumber = 1;
Control = MQCFC_LAST;
ParameterCount = 2;
CompCode = MQCC_WARNING;
Reason = MQRC_COMMAND_PCF;
```

Nota: Il valore ParameterCount è due perché ci sono sempre due parametri di tipo MQCFGR (gruppo). Ogni gruppo è costituito da parametri appropriati. I dati evento sono composti da due gruppi, CommandContext e CommandData.

CommandContext contiene:

EventUserID

Descrizione:	L'ID utente che ha emesso il comando o la chiamata che ha generato l'evento. (Questo è lo stesso ID utente utilizzato per controllare l'autorizzazione a emettere il comando o la chiamata; per i comandi ricevuti da una coda, questo è anche l'identificativo utente (UserIdentifier) dal MD del messaggio di comando).
Identificatore:	MQCACF_EVENT_USER_ID.
Tipo di dati:	MQCFST.
Lunghezza massima:	MQ_USER_ID_LENGTH.
Restituito:	Sempre.

EventOrigin

Descrizione:	L'origine dell'azione che causa l'evento.
Identificatore:	MQIACF_EVENT_ORIGIN.
Tipo di dati:	MQCFIN.
Valori:	**CONSOLE MQEVO** Riga comandi della console. **MQEVO_MSG** Messaggio di comando dal plugin Esplora risorse di WebSphere MQ .
Restituito:	Sempre.

EventQMgr

Descrizione:	Il gestore code in cui è stato immesso il comando o la chiamata. (Il gestore code in cui viene eseguito il comando e che genera l'evento si trova nel MD del messaggio di evento).
Identificatore:	MQCACF_EVENT_Q_MGR.
Tipo di dati:	MQCFST.
Lunghezza massima:	MQ_Q_MGR_NAME_LENGTH.
Restituito:	Sempre.

EventAccountingToken

Descrizione:	Per i comandi ricevuti come messaggio (MQEVO_MSG), il token di account (AccountingToken) dal MD del messaggio di comando.
Identificatore:	MQBACF_EVENT_ACCOUNTING_TOKEN.
Tipo di dati:	MQCFBS.
Lunghezza massima:	MQ_ACCOUNTING_TOKEN_LENGTH.
Restituito:	Solo se EventOrigin è MQEVO_MSG.

EventIdentityData

Descrizione:	Per i comandi ricevuti come messaggio (MQEVO_MSG), i dati di identità dell'applicazione (ApplIdentityData) dal MD del messaggio di comandi.
Identificatore:	MQCACF_EVENT_APPL_IDENTITY.
Tipo di dati:	MQCFST.
Lunghezza massima:	MQ_APPL_IDENTITY_DATA_LENGTH.
Restituito:	Solo se EventOrigin è MQEVO_MSG.

EventApplType

Descrizione:	Per i comandi ricevuti come messaggio (MQEVO_MSG), il tipo di applicazione (PutApplType) dal MD del messaggio di comando.
Identificatore:	MQIACF_EVENT_APPL_TYPE.
Tipo di dati:	MQCFIN.
Restituito:	Solo se EventOrigin è MQEVO_MSG.

EventApplName

Descrizione:	Per i comandi ricevuti come messaggio (MQEVO_MSG), il nome dell'applicazione (PutApplName) dal MD del messaggio di comando.
Identificatore:	MQCACF_EVENT_APPL_NAME.
Tipo di dati:	MQCFST.
Lunghezza massima:	LUNGHEZZA_NOME_APPL_MQ.
Restituito:	Solo se EventOrigin è MQEVO_MSG.

EventApplOrigin

Descrizione:	Per i comandi ricevuti come messaggio (MQEVO_MSG), i dati di origine dell'applicazione (ApplOriginData) dal MD del messaggio di comando.
Identificatore:	MQCACF_EVENT_APPL_ORIGIN.
Tipo di dati:	MQCFST.
Lunghezza massima:	MQ_APPL_ORIGIN_DATA_LENGTH.
Restituito:	Solo se EventOrigin è MQEVO_MSG.

Command

Descrizione:	Il codice di comando.
Identificatore:	COMANDO MQIACF.
Tipo di dati:	MQCFIN.
Valori:	**Valore numerico MQCMD_INQUIRE_PROT_POLICY 205** **Valore numerico 206 MQCMD_CREATE_PROT_POLICY** **Valore numerico MQCMD_DELETE_PROT_POLICY 207** **Valore numerico MQCMD_CHANGE_PROT_POLICY 208** Questi sono definiti in WebSphere MQ 7.5 `cmqcfc.h`
Restituito:	Sempre.

CommandData contiene elementi PCF che comprendono il comando PCF.

Formato messaggio evento di configurazione

Gli eventi di configurazione sono messaggi PCF in formato Advanced Message Security standard.

Per i valori possibili per il descrittore del messaggio MQMD, consultare Event message MQMD (message descriptor).

Di seguito sono riportati i valori MQMD selezionati:

```
Format = MQFMT_EVENT
Peristence = MQPER_PERSISTENCE_AS_Q_DEF
PutApplType = MQAT_QMGR                      //for both CLI and command server
```

Il buffer di messaggi è composto dalla struttura MQCFH e dalla struttura di parametro che la segue. Per i valori MQCFH possibili, consultare Messaggio evento MQCFH (intestazione PCF).

Di seguito sono riportati i valori MQCFH selezionati:

```
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}
```

I parametri che seguono MQCFH sono:

EventUserID

Descrizione: L'ID utente che ha emesso il comando o la chiamata che ha generato l'evento. (Questo è lo stesso ID utente utilizzato per controllare l'autorizzazione a emettere il comando o la chiamata; per i comandi ricevuti da una coda, questo è anche l'identificativo utente (UserIdentifier) dal MD del messaggio di comando).

Identificatore: **ID UTENTE MQCACF_EVENT_**

Tipo di dati: MQCFST.

Lunghezza massima: MQ_USER_ID_LENGTH.

Restituito: Sempre.

SecurityId

Descrizione: Valore di MQMD.AccountingToken in caso di messaggio del server dei comandi o SID Windows per il comando locale.

Identificatore: **ID_MQBACF_EVENT_SECURITY_ID**

Tipo di dati: MQCBS.

Lunghezza massima: MQ_SECURITY_ID_LENGTH.

Restituito: Sempre.

EventOrigin

Descrizione: L'origine dell'azione che causa l'evento.

Identificatore: **MQIACF_EVENT_ORIGIN**

Tipo di dati: MQCFIN.

Valori:

CONSOLE MQEVO
Riga comandi della console.

MQEVO_MSG
Messaggio di comandi dal plugin WebSphere MQ Explorer.

Restituito: Sempre.

EventQMgr

Descrizione: Il gestore code in cui è stato immesso il comando o la chiamata. (Il gestore code in cui viene eseguito il comando e che genera l'evento si trova nel MD del messaggio di evento).

Identificatore: **MQCACF_EVENT_Q_MGR**

Tipo di dati: MQCFST

Lunghezza massima: LUNGHEZZA_NOME_MQ_Q_MGR_

Restituito: Sempre.

ObjectType

Descrizione: Tipo di oggetto.

Identificatore: **TIPO_OGGETTO_MQIAC**

Tipo di dati: MQCFIN

Valore: **PROT_POLICY MQOT_**
Politica di protezione Advanced Message Security . **1019** - un valore numerico definito in WebSphere MQ 7.5 o nel file cmqc.h .

Restituito: Sempre.

PolicyName

Descrizione: Il nome della politica Advanced Message Security .

Identificatore: **NOME_POLITICA_MQCA**.

Tipo di dati: MQCFST.

Valore: **2112** - un valore numerico definito in WebSphere MQ 7.5 o nel file cmqc.h .

Lunghezza massima: MQ_OBJECT_NAME_LENGTH.

Restituito: Sempre.

PolicyVersion

Descrizione: La versione della politica Advanced Message Security .

Identificatore: **VERSIONE MQIA_POLICY_**

Tipo di dati: MQCFIN

Valore **238** - un valore numerico definito in WebSphere MQ 7.5 o nel file cmqc.h .

Restituito: Sempre

TolerateFlag

Descrizione: L'indicatore di tolleranza della politica Advanced Message Security .

Identificatore: **MQIA_TOLERATE_UNPROTECTED**

Tipo di dati: MQCFIN

Valore **235** - un valore numerico definito in WebSphere MQ 7.5 o nel file cmqc.h .

Restituito: Sempre.

SignatureAlgorithm

Descrizione: L'algoritmo di firma della politica Advanced Message Security .

Identificatore: **MQIA_SIGNATURE_ALGORITHM**

Tipo di dati: MQCFIN

Valore: **236** - un valore numerico definito in WebSphere MQ 7.5 o nel file cmqc.h .

Restituito: Ogni volta che è definito un algoritmo di firma nella politica Advanced Message Security

EncryptionAlgorithm

Descrizione: L'algoritmo di codifica della politica Advanced Message Security .

Identificatore: **ALGORITMO_CODIFICA_MQI**

Tipo di dati: MQCFIN

Valore: **237** - un valore numerico definito in WebSphere MQ 7.5 o nel file cmqc.h .

Restituito: Ogni volta che è presente un algoritmo di crittografia definito nella politica WebSphere MQ

SignerDNs

Descrizione: Oggetto DistinguishedName dei firmatari consentiti.

Identificatore: **DN_SIGNER_MQCA**

Tipo di dati: MQCFSL

Valore: **2113** - un valore numerico definito in WebSphere MQ 7.5 o nel file `cmqc.h` .

Lunghezza massima: DN firmatario più lungo nella politica, ma non più MQ_DISTINGUISHED_NAME_LENGTH

Restituito: Ogni volta che definito nella politica WebSphere MQ .

RecipientDNs

Descrizione: Oggetto DistinguishedName dei firmatari consentiti.

Identificatore: **DN_MQCA_RECIPIENT_**

Tipo di dati: MQCFSL

Valore: **2114** - un valore numerico definito in WebSphere MQ 7.5 o nel file `cmqc.h` .

Lunghezza massima: DN destinatario più lungo nella politica, ma non più MQ_DISTINGUISHED_NAME_LENGTH.

Restituito: Ogni volta che definito nella politica WebSphere MQ .

Problemi e soluzioni

Questa sezione descrive come risolvere i problemi che potrebbero verificarsi con l'installazione di IBM Utilizzare queste informazioni per identificare e risolvere eventuali problemi relativi a Advanced Message Security.

com.ibm.security.pkcsutil.PKCSException: Errore durante la codifica del contenuto

L'errore `com.ibm.security.pkcsutil.PKCSException: Error encrypting contents` suggerisce che IBM Advanced Message Security abbia problemi nell'accesso agli algoritmi crittografici.

Se il seguente errore viene restituito da Advanced Message Security:

```
DRQJP0103E   The IBM WebSphere MQ Advanced Message Security Java interceptor failed to protect
message.
  com.ibm.security.pkcsutil.PKCSException: Error encrypting contents
(java.security.InvalidKeyException: Illegal key size or default parameters)
```

verificare se la politica di sicurezza JCE in `JAVA_HOME/lib/security/local_policy.jar/*.policy` concede l'accesso agli algoritmi di firma utilizzati nella politica AMS di MQ .

Se l'algoritmo di firma che si desidera utilizzare non è specificato nella politica di sicurezza corrente, scaricare il file della politica Java corretto dalle seguenti ubicazioni:

- IBM SDK Policy files for Java 1.4.2.
- IBM File di politica SDK per Java 5.0
- IBM SDK Policy files for Java 6.0
- IBM SDK Policy files for Java 7.0.

Supporto OSGi

Per utilizzare il bundle OSGi con IBM Advanced Message Security sono richiesti ulteriori parametri.

Eseguire il seguente parametro durante l'avvio del bundle OSGi:

```
-Dorg.osgi.framework.system.packages.extra=com.ibm.security.pkcs7
```

Quando si utilizza la password codificata in keystore.conf, è necessario aggiungere la seguente istruzione quando è in esecuzione il bundle OSGi:

```
-Dorg.osgi.framework.system.packages.extra=com.ibm.security.pkcs7,com.ibm.misc
```

Limitazione: IBM WebSphere MQ AMS supporta la comunicazione utilizzando solo classi Java di base MQ per le code protette dall'interno del bundle OSGi.

Problemi nell'apertura di code protette quando si usa JMS

Possono verificarsi diversi problemi quando si aprono le code protette quando si utilizza IBM WebSphere MQ Advanced Message Security.

Si sta eseguendo JMS e si riceve l'errore 2085 (MQRC_UNKNOWN_OBJECT_NAME) insieme all'errore JMSMQ2008.

Hai verificato di aver configurato il tuo IBM WebSphere MQ Advanced Message Security come descritto in “Guida rapida per i clienti Java” a pagina 286.

Una possibile causa è che si sta utilizzando un ambiente di runtime nonIBM Java . Questa è una limitazione nota descritta in “Limitazioni note” a pagina 274.

Non è stata impostata la variabile di ambiente AMQ_DISABLE_CLIENT_AMS.

Risoluzione del problema

Esistono quattro opzioni per risolvere questo problema:

1. Avviare l'applicazione JMS in un JRE (IBM Java Runtime Environment) supportato.
2. Spostare l'applicazione sulla stessa macchina su cui è in esecuzione il proprio gestore code e fare in modo che si connetta utilizzando una connessione in modalità bind.

   Una connessione in modalità bind utilizza librerie native della piattaforma per eseguire le chiamate API IBM WebSphere MQ . Di conseguenza, l'intercettatore AMS nativo viene utilizzato per eseguire le operazioni AMS e non ci si affida alle funzionalità di JRE.
3. Utilizzare un intercettatore MCA, perché ciò consente la firma e la codifica dei messaggi non appena arrivano al gestore code, senza che il client debba eseguire alcuna elaborazione AMS.

   Poiché la protezione viene applicata al gestore code, è necessario utilizzare un meccanismo alternativo per proteggere i messaggi in transito dal client al gestore code. Più comunemente questo si ottiene configurando la cifratura SSL/TLS sul canale di connessione server utilizzato dall'applicazione.
4. Impostare la variabile di ambiente AMQ_DISABLE_CLIENT_AMS se non si desidera utilizzare IBM WebSphere MQ Advanced Message Security.

Consultare “Intercettazione MCA (Message Channel Agent)” a pagina 298 per ulteriori informazioni.

Nota: È necessario disporre di una politica di sicurezza per ogni coda in cui MCA Interceptor consegnerà i messaggi. In altre parole, la coda di destinazione deve disporre di una politica di sicurezza AMS con il DN (distinguished name) del firmatario e del destinatario corrispondente a quello del certificato assegnato a MCA Interceptor. Ovvero, il DN del certificato designato dalla proprietà di `cms.certificate.channel.SYSTEM.DEF.SVRCONN` nel `keystore.conf` utilizzato dal gestore code.

Informazioni particolari

Queste informazioni sono state sviluppate per i prodotti ed i servizi offerti negli Stati Uniti.

IBM potrebbe non offrire i prodotti, i servizi o le funzioni descritti in questo documento in altri paesi. Consultare il rappresentante IBM locale per informazioni sui prodotti e sui servizi disponibili nel proprio paese. Ogni riferimento relativo a prodotti, programmi o servizi IBM non implica che solo quei prodotti, programmi o servizi IBM possano essere utilizzati. In sostituzione a quelli forniti da IBM possono essere usati prodotti, programmi o servizi funzionalmente equivalenti che non comportino la violazione dei diritti di proprietà intellettuale o di altri diritti dell'IBM. È comunque responsabilità dell'utente valutare e verificare la possibilità di utilizzare altri programmi e/o prodotti, fatta eccezione per quelli espressamente indicati dall'IBM.

IBM potrebbe disporre di applicazioni di brevetti o brevetti in corso relativi all'argomento descritto in questo documento. La fornitura di tale documento non concede alcuna licenza a tali brevetti. Chi desiderasse ricevere informazioni relative a licenze può rivolgersi per iscritto a:

Director of Commercial Relations
IBM Corporation
Schoenaicher Str. 220
D-7030 Boeblingen
U.S.A.

Per richieste di licenze relative ad informazioni double-byte (DBCS), contattare il Dipartimento di Proprietà Intellettuale IBM nel proprio paese o inviare richieste per iscritto a:

Intellectual Property Licensing
Legge sulla proprietà intellettuale e legale
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

Il seguente paragrafo non si applica al Regno Unito o a qualunque altro paese in cui tali dichiarazioni sono incompatibili con le norme locali: INTERNATIONAL BUSINESS MACHINES CORPORATION FORNISCE LA PRESENTE PUBBLICAZIONE "NELLO STATO IN CUI SI TROVA" SENZA GARANZIE DI ALCUN TIPO, ESPRESSE O IMPLICITE, IVI INCLUSE, A TITOLO DI ESEMPIO, GARANZIE IMPLICITE DI NON VIOLAZIONE, DI COMMERCIABILITÀ E DI IDONEITÀ PER UNO SCOPO PARTICOLARE. Alcuni stati non consentono la rinuncia a garanzie esplicite o implicite in determinate transazioni; quindi la presente dichiarazione potrebbe non essere applicabile.

Questa pubblicazione potrebbe contenere imprecisioni tecniche o errori tipografici. Le informazioni incluse in questo documento vengono modificate su base periodica; tali modifiche vengono incorporate nelle nuove edizioni della pubblicazione. IBM si riserva il diritto di apportare miglioramenti o modifiche al prodotto/i e/o al programma/i descritti nella pubblicazione in qualsiasi momento e senza preavviso.

Qualsiasi riferimento a siti Web non IBM contenuto nelle presenti informazioni è fornito per consultazione e non vuole in alcun modo promuovere i suddetti siti Web. I materiali presenti in tali siti Web non sono parte dei materiali per questo prodotto IBM e l'utilizzo di tali siti Web è a proprio rischio.

Tutti i commenti e i suggerimenti inviati potranno essere utilizzati liberamente da IBM e diventeranno esclusiva della stessa.

Coloro che detengono la licenza su questo programma e desiderano avere informazioni su di esso allo scopo di consentire (i) uno scambio di informazioni tra programmi indipendenti ed altri (compreso questo) e (ii) l'uso reciproco di tali informazioni, dovrebbero rivolgersi a:

IBM Corporation
Coordinatore interoperabilità software, Dipartimento 49XA
Autostrada 3605 52 N

Rochester, MN 55901
U.S.A.

Queste informazioni possono essere rese disponibili secondo condizioni contrattuali appropriate, compreso, in alcuni casi, il pagamento di un addebito.

Il programma su licenza descritto in queste informazioni e tutto il materiale su licenza disponibile per esso sono forniti da IBM in base ai termini dell' IBM Customer Agreement, IBM International Program License Agreement o qualsiasi altro accordo equivalente tra le parti.

Tutti i dati relativi alle prestazioni contenuti in questo documento sono stati determinati in un ambiente controllato. Pertanto, i risultati ottenuti in altri ambienti operativi possono variare in modo significativo. Alcune misurazioni potrebbero essere state fatte su sistemi a livello di sviluppo e non vi è alcuna garanzia che queste misurazioni saranno le stesse sui sistemi generalmente disponibili. Inoltre, alcune misurazioni potrebbero essere state stimate mediante estrapolazione. I risultati quindi possono variare. Gli utenti di questo documento dovrebbero verificare i dati applicabili per il loro ambiente specifico.

Le informazioni relative a prodotti non IBM provengono dai fornitori di tali prodotti, dagli annunci pubblicati o da altre fonti pubblicamente disponibili. IBM non ha verificato tali prodotti e, pertanto, non può garantirne l'accuratezza delle prestazioni. Eventuali commenti relativi alle prestazioni dei prodotti non IBM devono essere indirizzati ai fornitori di tali prodotti.

Tutte le dichiarazioni riguardanti la direzione o l'intento futuro di IBM sono soggette a modifica o ritiro senza preavviso e rappresentano solo scopi e obiettivi.

Questa pubblicazione contiene esempi di dati e prospetti utilizzati quotidianamente nelle operazioni aziendali, Per illustrarle nel modo più completo possibile, gli esempi includono i nomi di individui, società, marchi e prodotti. Tutti questi nomi sono fittizi e qualsiasi somiglianza con nomi ed indirizzi adoperati da imprese realmente esistenti sono una mera coincidenza.

LICENZA SUL COPYRIGHT:

Queste informazioni contengono programmi applicativi di esempio in lingua originale, che illustrano le tecniche di programmazione su diverse piattaforme operative. È possibile copiare, modificare e distribuire questi programmi di esempio sotto qualsiasi forma senza alcun pagamento alla IBM, allo scopo di sviluppare, utilizzare, commercializzare o distribuire i programmi applicativi in conformità alle API (application programming interface) a seconda della piattaforma operativa per cui i programmi di esempio sono stati scritti. Questi esempi non sono stati testati approfonditamente tenendo conto di tutte le condizioni possibili. IBM, quindi, non può garantire o sottintendere l'affidabilità, l'utilità o il funzionamento di questi programmi.

Se si sta visualizzando queste informazioni in formato elettronico, le fotografie e le illustrazioni a colori potrebbero non apparire.

Informazioni sull'interfaccia di programmazione

Le informazioni sull'interfaccia di programmazione, se fornite, consentono di creare software applicativo da utilizzare con questo programma.

Questo manuale contiene informazioni sulle interfacce di programmazione che consentono al cliente di scrivere programmi per ottenere i servizi di IBM WebSphere MQ.

Queste informazioni, tuttavia, possono contenere diagnosi, modifica e regolazione delle informazioni. La diagnosi, la modifica e la regolazione delle informazioni vengono fornite per consentire il debug del software applicativo.

Importante: Non utilizzare queste informazioni di diagnosi, modifica e ottimizzazione come interfaccia di programmazione poiché sono soggette a modifica.

Marchi

IBM, il logo IBM , ibm.com, sono marchi di IBM Corporation, registrati in molte giurisdizioni nel mondo. Un elenco aggiornato dei marchi IBM è disponibile sul web in "Copyright and trademark

information"www.ibm.com/legal/copytrade.shtml. Altri nomi di prodotti e servizi potrebbero essere marchi di IBM o altre società.

Microsoft e Windows sono marchi di Microsoft Corporation negli Stati Uniti e/o in altri paesi.

UNIX è un marchio registrato di The Open Group negli Stati Uniti e/o in altri paesi.

Linux è un marchi registrato di Linus Torvalds negli Stati Uniti e/o in altri paesi.

Questo prodotto include il software sviluppato da Eclipse Project (http://www.eclipse.org/).

Java e tutti i marchi e i logo Java sono marchi registrati di Oracle e/o di società affiliate.

Numero parte:

(1P) P/N: