

9.0

*Protezione di IBM MQ*

**IBM**

**Nota**

Prima di utilizzare queste informazioni e il prodotto che supportano, leggere le informazioni in [“Informazioni particolari” a pagina 609](#).

Questa edizione si applica alla release 0 della versione 9 di IBM® 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.

© **Copyright International Business Machines Corporation 2007, 2023.**

---

# Indice

<b>Protezione.....</b>	<b>5</b>
Aggiornamenti di sicurezza.....	5
Panoramica della sicurezza.....	5
Concetti e meccanismi di sicurezza.....	5
Meccanismi di sicurezza IBM MQ.....	21
Pianificazione dei requisiti di sicurezza.....	71
Identificazione e autenticazione della pianificazione.....	72
Autorizzazione di pianificazione.....	75
Pianificazione della riservatezza.....	91
Pianificazione dell'integrità dei dati.....	99
Controllo pianificazione.....	99
Pianificazione della sicurezza per topologia.....	100
Firewall e pass-thru Internet.....	115
Elenco di controllo dell'implementazione della sicurezza IBM MQ for z/OS.....	115
Configurazione della sicurezza.....	118
Impostazione della sicurezza su UNIX, Linux, and Windows.....	118
Considerazioni speciali per la sicurezza su Windows.....	139
Impostazione della sicurezza su IBM i.....	147
Impostazione della sicurezza su z/OS.....	177
Impostazione della sicurezza IBM MQ MQI client.....	261
Impostazione delle comunicazioni per SSL o TLS su IBM i.....	264
Impostazione delle comunicazioni per SSL o TLS su UNIX, Linux o Windows.....	264
Impostazione delle comunicazioni per SSL o TLS su z/OS.....	265
Utilizzo di SSL/TLS.....	266
Identificazione e autenticazione degli utenti.....	321
Utenti privilegiati.....	324
Identificazione e autenticazione degli utenti utilizzando la struttura MQCSP.....	325
Implementazione dell'identificazione e dell'autenticazione nelle uscite di sicurezza.....	325
Mappature di identità nelle uscite del messaggio.....	326
Associazione di identità nell'uscita API e nell'uscita incrociata API.....	327
Utilizzo dei certificati revocati.....	328
Utilizzo di PAM (Pluggable Authentication Method).....	340
Autorizzazione dell'accesso agli oggetti.....	340
Controllo dell'accesso agli oggetti utilizzando OAM su UNIX, Linux, and Windows.....	340
Concessione dell'accesso richiesto alle risorse.....	351
Autorizzazione per gestire IBM MQ su UNIX, Linux, and Windows.....	391
Autorizzazione per gestire gli oggetti IBM MQ su UNIX, Linux, and Windows.....	393
Implementazione del controllo accessi nelle uscite di sicurezza.....	398
Implementazione del controllo accessi nelle uscite dei messaggi.....	400
Implementazione del controllo accessi nell'uscita API e nell'uscita incrociata API.....	400
Autorizzazione LDAP.....	401
Impostazione delle autorizzazioni.....	402
Visualizzazione delle autorizzazioni.....	404
Altre considerazioni sull'utilizzo dell'autorizzazione LDAP.....	404
Passaggio tra i modelli di autorizzazione SO e LDAP.....	405
Gestione LDAP.....	406
Riservatezza dei messaggi.....	407
Abilitazione di CipherSpecs.....	408
Reimpostazione delle chiavi segrete SSL e TLS.....	427
Implementazione della riservatezza nei programmi di uscita utente.....	429
Integrità dei dati dei messaggi.....	430
Audit.....	431

Proteggere i cluster.....	432
Arresto dei messaggi di invio dei gestori code non autorizzati.....	432
Arresto dei gestori code non autorizzati che immettono messaggi nelle code.....	432
Autorizzazione all'inserimento di messaggi nelle code del cluster remoto.....	433
Impedire ai gestori code di unirsi a un cluster.....	434
Forzare i gestori code indesiderati a lasciare un cluster.....	435
Come impedire ai gestori code di ricevere messaggi.....	436
SSL/TLS e cluster.....	436
Sicurezza di pubblicazione/sottoscrizione.....	438
Impostazione della sicurezza di pubblicazione / sottoscrizione di esempio.....	446
Sicurezza sottoscrizione.....	459
Sicurezza di pubblicazione / sottoscrizione tra i gestori code.....	461
Sicurezza IBM MQ Console e REST API.....	464
Configurazione di utenti e ruoli.....	466
Utilizzo dell'autenticazione del certificato client con REST API e IBM MQ Console.....	471
Utilizzo di autenticazione di base HTTP con REST API.....	475
Utilizzo dell'autenticazione basata su token con REST API.....	477
Configurazione di CORS per REST API.....	481
Audit.....	483
Considerazioni sulla protezione per IBM MQ Console e REST API su z/OS.....	483
Configurazione della sicurezza MFTREST API.....	491
Gestione di chiavi e certificati su UNIX, Linux, and Windows.....	492
comandi runmqckm e runmqakm.....	493
opzioni runmqckm e runmqakm.....	503
codici di errore runmqakm.....	506
Protezione dei dettagli di autenticazione del database.....	514
Protezione dei client AMQP.....	515
Limitazione del takeover del client AMQP.....	517
Configurazione di JAAS per canali AMQP.....	518
Advanced Message Security.....	520
Panoramica di AMS.....	520
Installazione Advanced Message Security.....	547
Controllo su z/OS.....	548
Utilizzo di keystore e certificati.....	549
Gestione delle politiche di sicurezza Advanced Message Security.....	575
Problemi e soluzioni.....	596
Configurazioni di esempio su z/OS.....	598
<b>Informazioni particolari.....</b>	<b>609</b>
Informazioni sull'interfaccia di programmazione.....	610
Marchi.....	610

# protezione IBM MQ

---

La sicurezza è una considerazione importante sia per gli sviluppatori di applicazioni IBM MQ che per gli amministratori di sistema IBM MQ .

## Aggiornamenti di sicurezza

---

Assicurarsi che tutto l'hardware e il software all'interno della zona protetta e sulle stazioni di lavoro dell'operatore siano nel loro ciclo di vita di supporto, siano stati aggiornati con aggiornamenti software obbligatori e siano stati applicati prontamente gli aggiornamenti di sicurezza.

È possibile trovare ulteriori informazioni sugli aggiornamenti della protezione per:

- Tutte le piattaforme su [IBM Security Bulletins](#)
- Sicurezza e integrità del sistema APAR su z/OS sul portale [IBM Z System Integrity](#).

## Panoramica della sicurezza

---

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

## Concetti e meccanismi di sicurezza

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

Gli aspetti comunemente accettati della sicurezza sono i seguenti:

- [“Identificazione e autenticazione”](#) a pagina 6
- [“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 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 71.

### Concetti correlati

[“Utilizzo di SSL/TLS”](#) a pagina 266

Questi argomenti forniscono istruzioni per eseguire singole attività relative all'utilizzo di TLS con IBM MQ.

### Informazioni correlate

[Connessione di due gestori code mediante TLS](#)

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

### Concetti correlati

[“Identificazione e autenticazione in IBM MQ” a pagina 21](#)

In IBM 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 MQ” a pagina 22](#)

È possibile utilizzare l'autorizzazione per limitare ciò che particolari individui o applicazioni possono fare nel proprio ambiente IBM 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 75 e gli argomenti secondari associati.

### **Concetti correlati**

[“Controllo in IBM MQ”](#) a pagina 22

IBM 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 IBM MQ.

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

### **Concetti correlati**

[“Crittografia in IBM MQ”](#) a pagina 23

IBM MQ fornisce la crittografia utilizzando il protocollo 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*).

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/TLS”](#) a pagina 20.

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 crittografia 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 crittografia della chiave asimmetrica, nota anche come *crittografia 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.

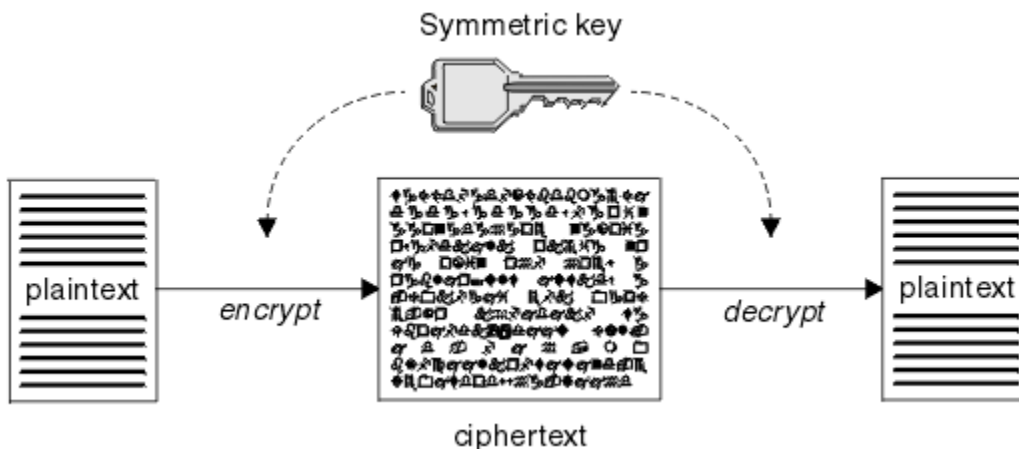


Figura 1. crittografia di chiavi simmetrica

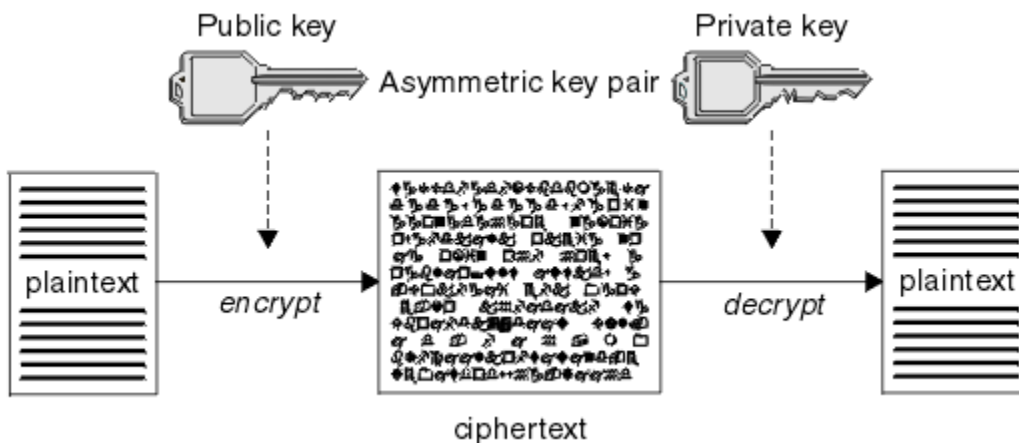


Figura 2. crittografia di chiavi asimmetrica

[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. Il digest del messaggio è calcolato da una funzione hash e può essere crittografato, formando una firma digitale.

La funzione hash utilizzata per calcolare un digest del messaggio deve soddisfare due criteri:

- Deve essere un modo. 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/TLS” a pagina 20](#)

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 IBM 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*

IBM MQ supporta certificati digitali conformi allo standard X.509 . Richiede l'opzione di autenticazione client.

Poiché IBM MQ è un sistema peer to peer, viene visualizzato come autenticazione client nella terminologia SSL/TLS. Pertanto, qualsiasi certificato personale utilizzato per l'autenticazione SSL/TLS 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 MQ, consultare [“Politiche di convalida dei certificati in IBM MQ”](#) a pagina 42.

#### *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, vedi [“Utilizzo dei certificati revocati”](#) a pagina 328
- 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 .



**Attenzione:** Solo gli attributi nella seguente tabella possono essere utilizzati in un filtro SSLPEER. I DN certificato possono contenere altri attributi, ma il filtro non è consentito su questi attributi.

<i>Tabella 1. Tipi di attributo trovati nel DN che possono essere utilizzati in un filtro SSLPEER</i>	
<b>Tipo di attributo</b>	<b>Descrizione</b>
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)

Tabella 1. Tipi di attributo trovati nel DN che possono essere utilizzati in un filtro SSLPEER (Continua)

Tipo di attributo	Descrizione
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 MQ tollera alcuni DN non corretti. Per ulteriori informazioni, consultare [IBM MQ regole 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 certificato vengono generalmente generate dallo strumento di gestione dei certificati utilizzato dal sistema; ad esempio:

-  Multi Lo strumento iKeyman su [Multiplatforme](#).
-  z/OS 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 13](#) illustra il processo per ottenere un certificato digitale da una CA.

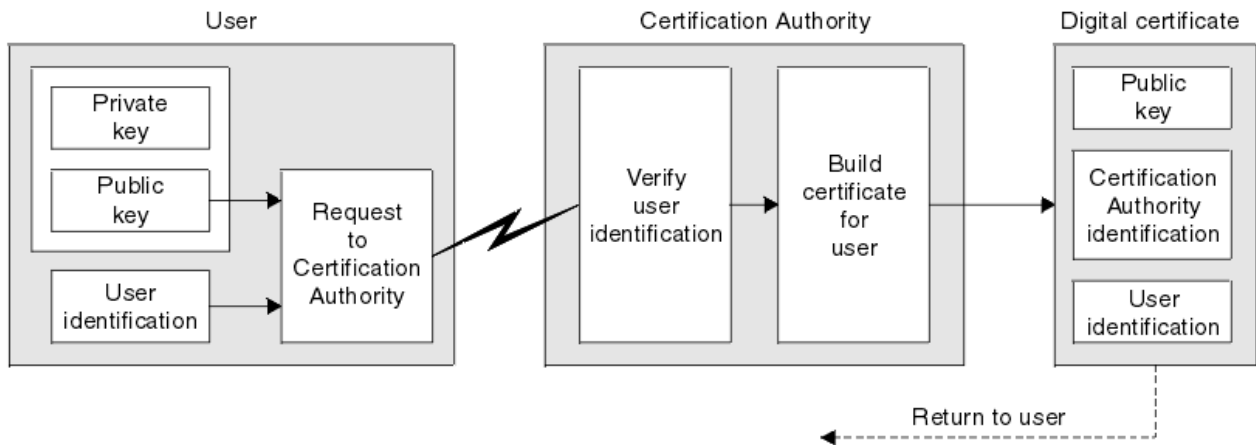


Figura 3. Ottenimento di un certificato digitale

Nel diagramma:

- L'identificazione utente include il DN (Distinguished Name) del soggetto.
- L'identificazione dell'autorità di certificazione include il DN (Distinguished Name) della CA che emette 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 14 illustra un percorso di certificazione dal proprietario del certificato alla CA root, dove inizia la catena di attendibilità.

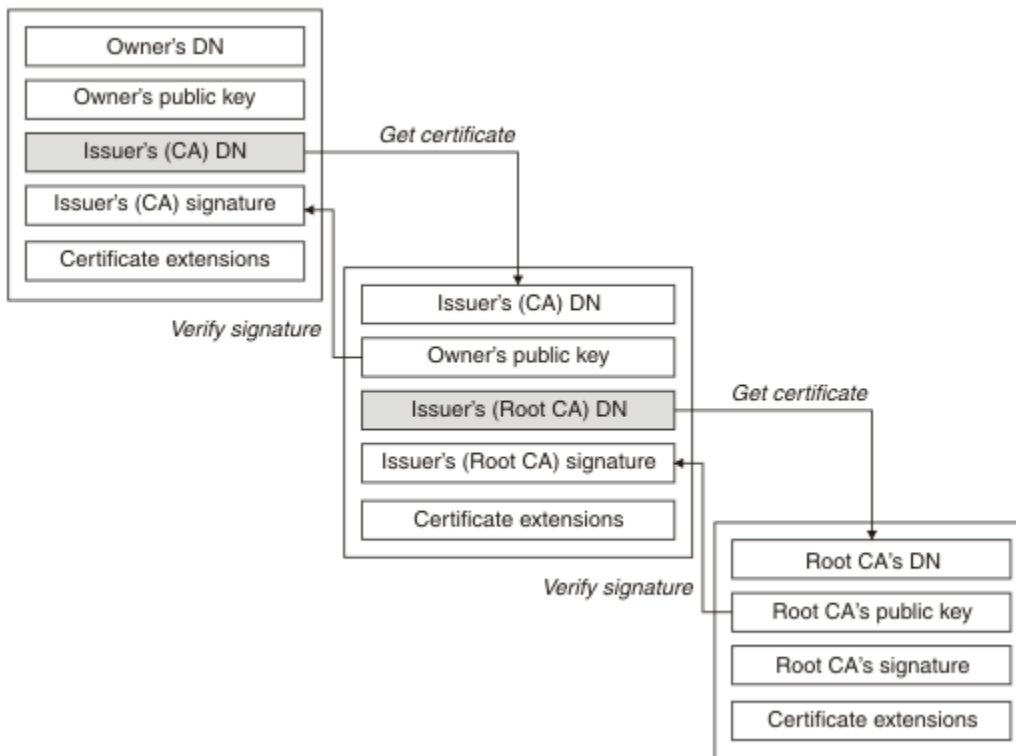


Figura 4. Catena di fiducia

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.

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.

IBM MQ può controllare se un certificato viene revocato inviando una richiesta a un responder OCSP (Online Certificate Status Protocol) (solo su UNIX, Linux®, and Windows). In alternativa, possono accedere a un CRL (Certificate Revocation List) 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 328.

### **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 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: 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 MQ supporta 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 TLS in IBM MQ”](#) a pagina 24

IBM MQ supporta il protocollo TLS (Transport Layer Security) per fornire la sicurezza a livello di link per canali di messaggi e canali MQI.

### **Concetti di TLS (Transport Layer Security)**

Il protocollo TLS consente a due parti di identificarsi e autenticarsi reciprocamente e di 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.

Il protocollo TLS fornisce la sicurezza delle comunicazioni su Internet e consente 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 TLS viene avviata da un'applicazione, che diventa il client TLS. L'applicazione che riceve la connessione diventa il server TLS. Ogni nuova sessione inizia con un handshake, come definito dai protocolli TLS.

Un elenco completo di CipherSpecs supportati da IBM MQ è disponibile all'indirizzo [“Abilitazione di CipherSpecs”](#) a pagina 408

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/TLS**

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

Questa sezione fornisce un riepilogo dei passi che consentono al client e al server 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. TLS utilizza quindi la chiave condivisa per la codifica simmetrica dei messaggi, che è più veloce della crittografia asimmetrica.

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

In panoramica, i passi coinvolti nell'handshake TLS sono i seguenti:

1. Il client TLS invia un messaggio "client hello" che elenca le informazioni crittografiche come la versione 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 TLS risponde con un messaggio "server hello" che contiene la CipherSuite scelta dal server dall'elenco fornito dal cliente, l'ID sessione e 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 TLS verifica il certificato digitale del server. Per ulteriori informazioni, consultare [“Come TLS fornisce identificazione, autenticazione, riservatezza e integrità” a pagina 17.](#)
4. Il client TLS invia la stringa di byte casuale che abilita sia il client che il server a 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 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 TLS verifica il certificato del client. Per ulteriori informazioni, consultare [“Come TLS fornisce identificazione, autenticazione, riservatezza e integrità” a pagina 17.](#)
7. Il client 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 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 TLS, il client e il server possono ora scambiare i messaggi che sono simmetricamente codificati con la chiave segreta condivisa.

[Figura 5 a pagina 17](#) illustra l'handshake TLS.



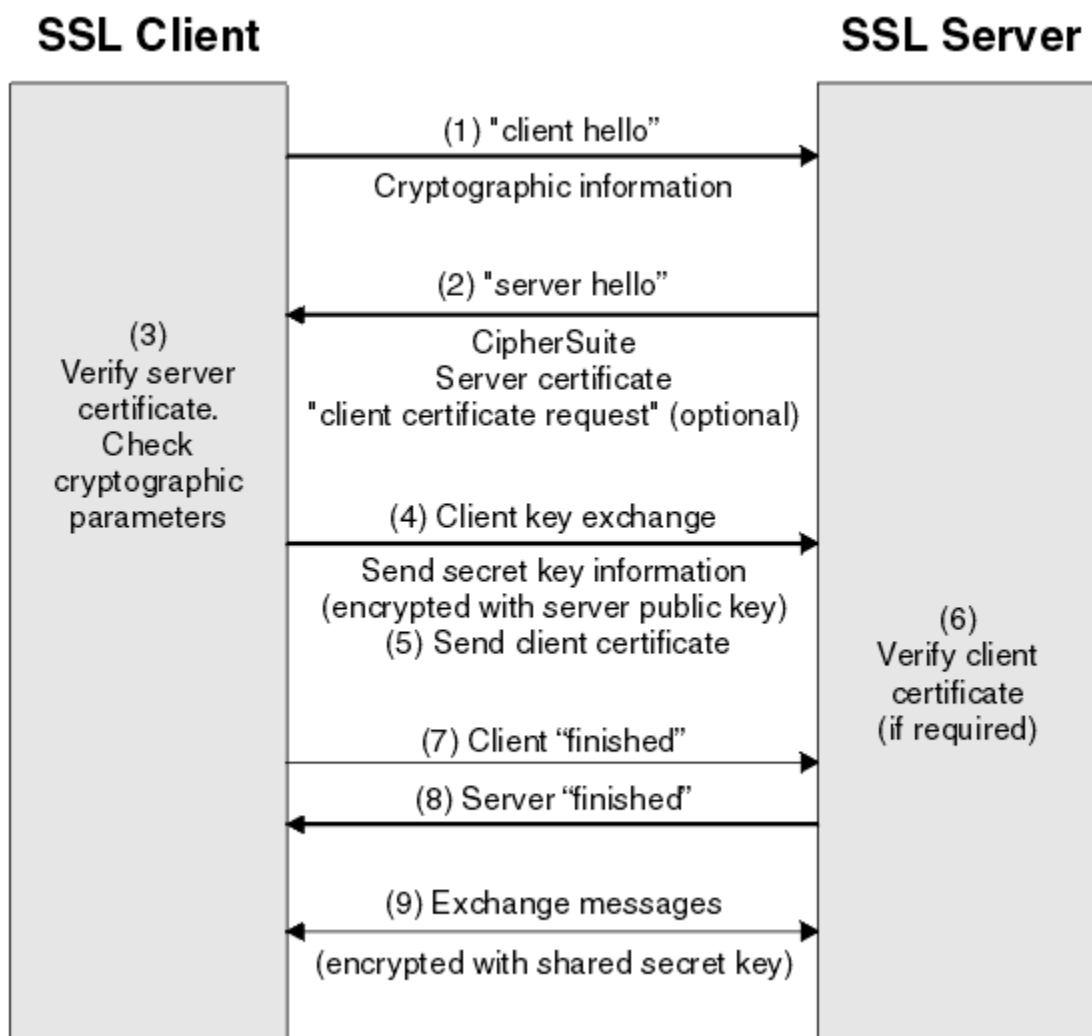


Figura 5. Panoramica dell'handshake TLS

### **Come TLS fornisce 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/TLS"](#) a pagina 15.

### **Come TLS fornisce 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 del client, il server utilizza la chiave pubblica nel certificato client per decodificare i dati che il client invia durante il passo ["5"](#) a pagina 16 dell'handshake. Lo scambio di messaggi terminati crittografati con la chiave segreta (passi ["7"](#) a pagina 16 e ["8"](#) a pagina 16 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 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 TLS e CA Y emette il certificato per il server TLS:

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

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

e il cliente TLS ha bisogno di:

- Il certificato CA per CA Y

Se il server 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 per il 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 server TLS che il client potrebbero aver bisogno di altri certificati CA per formare una catena di certificati al 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 16 e “6” a pagina 16 della panoramica, il client TLS verifica il certificato del server e il server TLS verifica il certificato del client. Questa verifica presenta quattro aspetti:

1. La firma digitale viene controllata (consultare [“Firme digitali in SSL/TLS”](#) a pagina 20).
2. La catena di certificati è controllata; è necessario disporre di certificati CA intermedi (vedere [“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 328).

## Reimpostazione chiave segreta

Durante un handshake TLS viene generata una *chiave segreta* per codificare i dati tra il client e il server TLS. 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 TLS fornisce la riservatezza

TLS utilizza una combinazione di codifica simmetrica e asimmetrica per garantire la privacy dei messaggi. Durante l'handshake TLS, il client e il server TLS concordano un algoritmo di crittografia e una chiave segreta condivisa da utilizzare solo per una sessione. Tutti i messaggi trasmessi tra il server e il client TLS vengono crittografati utilizzando tale algoritmo e chiave, garantendo che il messaggio rimanga privato anche se viene intercettato. Poiché TLS utilizza 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 crittografia, fare riferimento a [“Crittografia” a pagina 7](#).

## Come TLS fornisce l'integrità

TLS fornisce l'integrità dei dati calcolando un digest del messaggio. Per ulteriori informazioni, fare riferimento a [“Integrità dei dati dei messaggi” a pagina 430](#).

L'utilizzo di TLS garantisce l'integrità dei dati, purché CipherSpec nella definizione di canale utilizzi un algoritmo hash come descritto nella tabella in [“Abilitazione di CipherSpecs” a pagina 408](#).

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 devono concordare sullo stesso CipherSpec per poter comunicare.

Da IBM MQ 9.0.0 Fix Pack 3 e IBM MQ 9.0.5, IBM MQ supporta il protocollo TLSv1.2. Tuttavia, è possibile abilitare CipherSpecs obsoleti, se necessario.

Consultare [“Abilitazione di CipherSpecs” a pagina 408](#) per informazioni su:

- CipherSpecs supportati da IBM MQ
- Come abilitare SSLv3 e TLSv1.0 CipherSpecs obsoleti.

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

Una CipherSuite è una suite di algoritmi crittografici utilizzata da una connessione 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 ogni componente della suite, ma solo alcune combinazioni sono valide quando specificate per una connessione TLS. Il nome di una CipherSuite valida definisce la combinazione di algoritmi utilizzati. Ad esempio, CipherSuite TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA specifica:

- Lo scambio di chiavi RSA e l'algoritmo di autenticazione
- L'algoritmo di crittografia AES, che utilizza una chiave a 128 bit e la modalità CBC (cipher block chaining)
- MAC (Message Authentication Code) SHA-1

## Firme digitali in SSL/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 20 illustra questo processo.

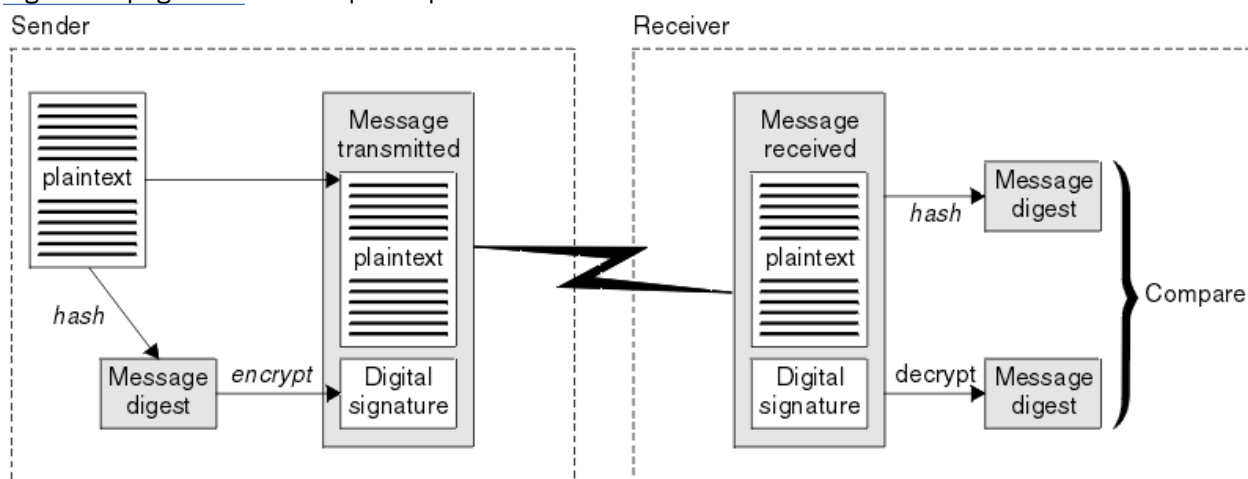


Figura 6. Il processo di firma digitale

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 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.

### **Concetti correlati**

[“Crittografia della National Security Agency \(NSA\) Suite B” a pagina 21](#)

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.

### **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 è configurato per l'operazione compatibile con Suite B, viene utilizzata solo la serie limitata di algoritmi di codifica 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 and Linux , IBM 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 MQ” a pagina 38](#).

### **Riferimenti correlati**

[“FIS \(Federal Information Processing Standards\)” a pagina 20](#)

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 MQ**

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

IBM 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 MQ**

In IBM 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 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 6](#)

*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 MQ

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

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

- Consentire solo a un amministratore autorizzato di immettere comandi per gestire le risorse IBM 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 75](#) 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 MQ

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

Di seguito sono riportati alcuni esempi di verifica in un ambiente IBM 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 MQ

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

La riservatezza può essere garantita in un ambiente IBM MQ nel modo seguente:

- Dopo che un MCA mittente riceve un messaggio da una coda di trasmissione, IBM MQ utilizza 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 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 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 MQ

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

L'integrità dei dati può essere garantita in un ambiente IBM MQ nel modo seguente:

- È possibile utilizzare TLS per rilevare se il contenuto di un messaggio è stato deliberatamente modificato mentre veniva trasmesso su una rete. In TLS, l'algoritmo digest del messaggio fornisce il rilevamento dei messaggi modificati in transito.

Tutti i IBM MQ CipherSpecs forniscono un algoritmo digest del messaggio, tranne TLS\_RSA\_WITH\_NULL\_NULL, che non fornisce l'integrità dei dati del messaggio.

IBM MQ rileva i messaggi modificati al momento della ricezione; al momento della ricezione di un messaggio modificato, IBM MQ emette un messaggio di errore AMQ9661 e il canale si arresta.

- Mentre i messaggi sono memorizzati in una coda locale, i meccanismi di controllo degli accessi forniti da IBM MQ potrebbero essere considerati sufficienti per impedire la modifica deliberata del contenuto dei messaggi.

Tuttavia, per un livello di sicurezza maggiore, è possibile utilizzare 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.

Dopo aver rilevato un messaggio modificato, l'applicazione che tenta di ricevere il messaggio riceve un codice di ritorno 2063 e, se utilizza una chiamata [MQGET](#), il messaggio viene spostato nel SISTEMA.SYSTEM.PROTECTION.ERROR.QUEUE

### 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 MQ

IBM MQ fornisce la crittografia utilizzando il protocollo TLS (Transport Security Layer).

Per ulteriori informazioni, fare riferimento a [“Protocolli di sicurezza TLS in IBM MQ” a pagina 24](#).

### Concetti correlati

[“Concetti crittografici” a pagina 7](#)

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

## Protocolli di sicurezza TLS in IBM MQ

IBM MQ supporta il protocollo TLS (Transport Layer Security) per fornire la sicurezza a livello di link per canali di messaggi e canali MQI.

I canali di messaggi e MQI possono utilizzare il protocollo TLS per fornire la sicurezza a livello di link. Un MCA chiamante è un client TLS e un MCA rispondente è un server TLS. IBM MQ supporta TLS 1.0 e TLS 1.2. È possibile specificare gli algoritmi crittografici utilizzati dal protocollo TLS fornendo una CipherSpec come parte della definizione di canale.

**Nota:** Da IBM MQ 8.0.0 Fix Pack 2, il protocollo SSLv3 e l'utilizzo di alcuni IBM MQ CipherSpecs è obsoleto. Per ulteriori informazioni, consultare [Deprecazione: protocollo SSLv3](#).

È possibile utilizzare il parametro `SECPROT` per visualizzare il protocollo di protezione in uso su un 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 TLS, l'MCA invia il certificato digitale del gestore code all'MCA partner all'altra estremità del canale. Il codice IBM MQ all'estremità client di un canale MQI agisce per conto dell'utente dell'applicazione client IBM MQ. Durante l'handshake TLS, il codice IBM MQ invia il certificato digitale dell'utente all'MCA all'estremità server del canale MQI.

Ai gestori code e agli utenti client IBM MQ non è richiesto di avere certificati digitali personali associati quando agiscono come client TLS, a meno che `SSLCAUTH (REQUIRED)` non sia specificata 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 contenente il certificato digitale del gestore code. Su un sistema client IBM MQ, la variabile di ambiente `MQSSLKEYR` specifica l'ubicazione del repository delle chiavi che contiene il certificato digitale dell'utente. In alternativa, un'applicazione client IBM MQ può specificare la relativa posizione nel campo **KeyRepository** della struttura delle opzioni di configurazione TLS, `MQSCO`, su una chiamata `MQCONN`. Consultare gli argomenti correlati per ulteriori informazioni sui repository chiave e su come specificare dove si trovano.

## Supporto per TLS

IBM MQ fornisce supporto per TLS 1.0 e TLS 1.2 in base alla piattaforma che stai utilizzando. Per ulteriori informazioni sul protocollo TLS, fare riferimento alle informazioni negli argomenti secondari.

### IBM i

Il supporto TLS è parte integrante del sistema operativo IBM i.

### Client Java e JMS

Questi client utilizzano la JVM per fornire il supporto TLS.

### UNIX, Linux, and Windows

Il supporto TLS è installato con IBM MQ.

### z/OS

Il supporto TLS è parte integrante del sistema operativo z/OS. Il supporto TLS su z/OS è noto come *SSL di sistema*.

Per informazioni su tutti i prerequisiti per il supporto TLS IBM MQ, consultare [Requisiti di sistema per IBM MQ](#).

### Concetti correlati

“Protocolli di sicurezza crittografici: TLS” a pagina 15




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 MQ supporta TLS.

### Il repository delle chiavi SSL/TLS

Una connessione TLS autenticata reciprocamente richiede un repository di chiavi a ciascuna 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. Al repository delle chiavi fanno riferimento nomi diversi su piattaforme e ambienti differenti che supportano TLS:

-  Su IBM i: *archivio certificati*
- Su Java e JMS: *keystore* e *truststore*
-  Su UNIX, Linux, and Windows: *file database delle chiavi*
-  Su z/OS: *keyring*

Per ulteriori informazioni, consultare [“Certificati digitali” a pagina 9](#) e [“Concetti di TLS \(Transport Layer Security\)” a pagina 15](#).

Una connessione TLS autenticata reciprocamente richiede un repository di chiavi a ciascuna estremità della connessione. Il repository delle chiavi può contenere i seguenti certificati e richieste:

- 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 IBM MQ MQI client. I certificati personali sono essenziali su un client TLS se è necessaria 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 tuo repository delle chiavi, vedi [“Protezione dei repository delle chiavi IBM MQ” a pagina 26](#).

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

#### **IBM i**

Il repository chiavi è un archivio certificati. La memorizzazione certificato di sistema predefinita si trova in `/QIBM/UserData/ICSS/Cert/Server/Default` nell'IFS (integrated file system). IBM MQ memorizza la password per l'archivio certificati in un *file stash delle password*. Ad esempio, il file stash per il gestore code QM1 è `/QIBM/UserData/mqm/qmgrs/QM1/ssl/Stash.sth`.

In alternativa, è possibile specificare che deve essere utilizzato l'archivio certificati del sistema IBM i. A tale scopo, modificare il valore dell'attributo **SSLKEYR** del gestore code in `*SYSTEM`. Questo valore indica che il gestore code deve utilizzare l'archivio di certificati di sistema e che il gestore code è registrato per l'utilizzo come applicazione con DCM (Digital Certificate Manager).

L'archivio certificati contiene anche la chiave privata per il gestore code.

#### **Sistemi UNIX, Linux, and Windows**

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 MQ è installato nell'ubicazione predefinita, il percorso equivalente su Windows è `C:\ProgramData\IBM\MQ\Qmgrs\QM1\ssl\key.kdb`.

A ciascun file di database di chiavi è associato un file stash delle password. 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:** Le schede hardware crittografiche PKCS #11 possono contenere i certificati e le chiavi che sono altrimenti contenute in un file di database delle chiavi. Quando certificati e chiavi sono conservati su schede PKCS #11, IBM MQ richiede ancora l'accesso sia a un file di database delle chiavi che a un file stash delle password.

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

## z/OS

I certificati sono conservati in un keyring in z/OS.

Altri gestori di sicurezza esterni (ESM) utilizzano anche i keyring per memorizzare i certificati.

Le chiavi private sono gestite da RACF.

### *Protezione dei repository delle chiavi IBM MQ*

Il repository delle chiavi per IBM 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 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.

### *Etichette dei certificati digitali, comprensione dei requisiti*

Quando si configura 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.



## Qual 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. Notare 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. Se necessario, tuttavia, è possibile modificare l'etichetta associata a un certificato digitale.

## Sintassi etichetta certificato

Un'etichetta di certificato può contenere lettere, numeri e punteggiatura con le seguenti condizioni:

-  **Multi** L'etichetta del certificato può contenere fino a 64 caratteri.
-  **z/OS** L'etichetta del certificato può contenere fino a 32 caratteri.
- L'etichetta del certificato non può contenere spazi.
- Le etichette sono sensibili al maiuscolo / minuscolo.
- Sui sistemi che utilizzano il katakana EBCDIC, non è possibile utilizzare caratteri minuscoli.

Ulteriori requisiti per i valori di etichetta del certificato sono specificati nelle seguenti sezioni.

## Come viene utilizzata l'etichetta del certificato?

IBM MQ utilizza le etichette del certificato per individuare un certificato personale inviato durante l'handshake TLS. Ciò elimina l'ambiguità quando esiste più di un certificato personale nel repository delle chiavi.

È possibile impostare l'etichetta del certificato su un valore a scelta. Se non si imposta un valore, viene utilizzata un'etichetta predefinita che segue una convenzione di denominazione a seconda della piattaforma utilizzata. Per i dettagli, consultare le sezioni che seguono, relative a particolari piattaforme.

### Note:

1. Non è possibile impostare l'etichetta del certificato su sistemi Java o JMS .
2. I canali definiti automaticamente creati da un'uscita CHAD (channel automatic definition) non possono impostare l'etichetta del certificato, poiché l'handshake TLS si è verificato al momento della creazione del canale. L'impostazione dell'etichetta del certificato in un'uscita CHAD per i canali in ingresso non ha alcun effetto.

In questo contesto, un client TLS fa riferimento al partner di connessione che avvia l'handshake, che può essere un client IBM MQ o un altro gestore code.

Durante l'handshake TLS, il client TLS ottiene e convalida sempre un certificato digitale dal server. Con l'implementazione IBM MQ , il server 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 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 funge da server TLS è definita con il parametro **SSLCAUTH** impostato su *REQUIRED* o con un valore di parametro **SSLPEER** impostato.

Tenere presente che i canali in entrata (inclusi ricevente, richiedente, ricevente del cluster, server non qualificato e canali di connessione server) inviano il certificato configurato solo se la versione IBM MQ del peer remoto supporta completamente la configurazione dell'etichetta del certificato e il canale utilizza un CipherSpecTLS.

Un canale server non qualificato è uno che non ha il campo CONNAME impostato.

In tutti gli altri casi, il parametro **CERTLABL** del gestore code determina il certificato inviato. In particolare, quanto segue riceve sempre il certificato configurato dal parametro **CERTLABL** del gestore code, indipendentemente dall'impostazione dell'etichetta specifica del canale:

- Tutti i client Java e JMS correnti.
- Versioni di IBM MQ precedenti a IBM MQ 8.0.

Inoltre, il certificato utilizzato da un canale deve essere appropriato per il canale CipherSpec - consultare [“Certificati digitali e compatibilità CipherSpec in IBM MQ”](#) a pagina 42 per ulteriori informazioni.

IBM MQ 8.0 supporta l'utilizzo di più certificati sullo stesso gestore code, utilizzando un attributo di etichetta certificato per canale. I canali in entrata per il gestore code (ad esempio, la connessione server o il destinatario) si basano sul rilevamento del nome del canale utilizzando SNI (Server Name Indication) TLS, al fine di presentare il certificato corretto dal gestore code.

Se si utilizza MQIPT con una rotta che ha sia *SSLServer* che *SSLClient* impostati, ci sono due sessioni TLS separate tra gli endpoint e i dati SNI non fluiranno attraverso l'interruzione della sessione.

È possibile utilizzare instradamenti MQIPT separati per ottenere il supporto di più certificati selezionando il certificato appropriato, ad esempio mediante le proprietà di instradamento *SSLServerSiteLabel* e *SSLClientSiteLabel* . In alternativa, utilizzare MQIPT *SSLProxyMode* che inoltra tutti i flussi di controllo TLS intatti, incluso il nome SNI.

Notare che più certificati in MQIPT funzionano solo se si utilizza la modalità proxy TLS.

Per ulteriori informazioni, consultare la sezione di supporto TLS della documentazione *IBM MQ Internet Pass - Thru* .

Per ulteriori informazioni sulla connessione di un gestore code mediante l'autenticazione unidirezionale, ossia quando il client TLS non invia un certificato, fare riferimento a [Connessione di due gestori code mediante l'autenticazione unidirezionale](#).

## Sistemi multiplatforme



Su [Multiplatforme](#), il server TLS invia un certificato al client.

Per i gestori code e i client, rispettivamente, le seguenti origini vengono ricercate in sequenza per un valore non vuoto. Il primo valore non vuoto determina l'etichetta del certificato. L'etichetta del certificato deve esistere nel repository delle chiavi. Se non viene trovato alcun certificato corrispondente nel formato e nel maiuscolo / minuscolo corretti che corrisponde a un'etichetta, si verifica un errore e l'handshake TLS ha esito negativo.

### Gestori code

1. Attributo etichetta certificato canale **CERTLABL**.
2. Attributo etichetta certificato gestore code **CERTLABL**.
3. Un valore predefinito, che è nel formato: `ibmwebspheremq` con il nome del gestore code aggiunto, tutto in minuscolo. Ad esempio, per un gestore code denominato QM1, l'etichetta del certificato predefinita è `ibmwebspheremqqm1`.

### IBM MQ client

1. Attributo dell'etichetta del certificato **CERTLABL** nella definizione del canale CLNTCONN.
2. Attributo della struttura MQSCO **CertificateLabel**.
3. Variabile di ambiente **MQCERTLABL**.
4. Attributo `.ini` file (nella sezione SSL) **CertificateLabel** del client
5. Un valore predefinito, che è nel seguente formato: `ibmwebspheremq` con l'ID utente che l'applicazione client sta eseguendo come accodato, tutto in minuscolo. Ad esempio, per un ID utente USER1, l'etichetta del certificato predefinita è `ibmwebspheremquser1`.

## z/OS sistemi



I client IBM MQ non sono supportati su z/OS. Tuttavia, un gestore code z/OS può agire come un client TLS quando inizia una connessione o come un server TLS quando accetta una richiesta di connessione. I requisiti dell'etichetta del certificato per i gestori code z/OS si applicano in entrambi questi ruoli e differiscono dai requisiti su [Multiplatforme](#).

Per i gestori code e i client, rispettivamente, le seguenti origini vengono ricercate in sequenza per un valore non vuoto. Il primo valore non vuoto determina l'etichetta del certificato. L'etichetta del certificato deve esistere nel repository delle chiavi. Se non viene trovato alcun certificato corrispondente nel formato e nel maiuscolo / minuscolo corretti che corrisponde a un'etichetta, si verifica un errore e l'handshake TLS ha esito negativo.

1. Attributo etichetta certificato canale, **CERTLABL**.
2. Se condiviso, l'attributo dell'etichetta del certificato del gruppo di condivisione code, **CERTQSGL**.  
Se non è condiviso, l'attributo dell'etichetta del certificato del gestore code, **CERTLABL**.
3. Un valore predefinito, che è nel formato: `ibmWebSphereMQ` con il nome del gestore code o del gruppo di condivisione code accodato. Notare che questa stringa è sensibile al maiuscolo / minuscolo e deve essere scritta come mostrato. Ad esempio, per un gestore code denominato QM1, l'etichetta del certificato predefinita è `ibmWebSphereMQQM1`.
4. Se non viene trovato alcun certificato con il formato nell'opzione [“3” a pagina 28](#), IBM MQ tenta di utilizzare il certificato contrassegnato come predefinito nel keyring.

Per informazioni su come visualizzare il repository delle chiavi, consultare [“Individuazione del repository delle chiavi per un gestore code su z/OS”](#) a pagina 311.

## **IBM MQ Java e client IBM MQ JMS**

I client IBM MQ Java e IBM MQ JMS utilizzano le funzionalità del provider JSSE (Java Secure Socket Extension) per selezionare un certificato personale durante l'handshake TLS e non sono pertanto soggetti ai requisiti di 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 IBM MQ Java in combinazione con il provider JSSE specifico, è possibile abilitare il debug impostando `javax.net.debug=ssl` nell'ambiente JVM.

È possibile impostare la variabile all'interno dell'applicazione, tramite la configurazione o immettendo `-Djavax.net.debug=ssl` sulla riga comandi.

### *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 IBM MQ Explorer. Per ulteriori informazioni, consultare il [Comando MQCMD\\_REFRESH\\_SECURITY](#) e l'argomento *Aggiornamento della sicurezza TLS* nella sezione IBM MQ Explorer di questa documentazione del prodotto.

### **Concetti correlati**

[“Aggiornamento di una vista del client del contenuto del repository di chiavi SSL/TLS e delle impostazioni SSL/TLS”](#) a pagina 29

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 di chiavi SSL/TLS e delle impostazioni SSL/TLS*

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 IBM MQ; non è presente alcun 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 29

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).

### **Protezione password MQCSP**

Da IBM MQ 8.0, puoi inviare le password incluse nella struttura MQCSP protette, utilizzando la funzionalità IBM MQ o crittografate, utilizzando la crittografia TLS.

**Importante:** La protezione della password MQCSP è utile per scopi di test e sviluppo in quanto l'utilizzo della protezione della password MQCSP è più semplice rispetto all'impostazione della crittografia TLS, ma non così sicuro. Per scopi di produzione, dovresti utilizzare la codifica TLS al posto della protezione della password IBM MQ, specialmente quando la rete tra il client e il gestore code non è attendibile, poiché la codifica TLS è più sicura.

Se ti interessa esattamente quale crittografia viene utilizzata e quanta protezione offre, devi utilizzare la crittografia TLS completa. In questa situazione, gli algoritmi sono pubblicamente noti ed è possibile selezionare quello appropriato per la propria azienda utilizzando l'attributo del canale **SSLCIPH**.

Per ulteriori informazioni sulla struttura MQCSP, consultare [Struttura MQCSP](#).

La protezione con password viene utilizzata quando vengono soddisfatte tutte le seguenti condizioni:

- Entrambe le estremità della connessione utilizzano IBM MQ 8.0 o versioni successive.
- Il canale non utilizza la codifica TLS. Un canale non utilizza la crittografia TLS se il canale ha un attributo **SSLCIPH** vuoto o se l'attributo **SSLCIPH** è impostato su CipherSpec che non fornisce la crittografia. Le cifrature null, ad esempio NULL\_SHA, non forniscono la crittografia.
- Impostare **MQCSP**. Da **AuthenticationType** a MQCSP\_AUTH\_USER\_ID\_AND\_PWD. L'impostazione di questo valore consente di valutare più controlli per decidere se la protezione della password è stata eseguita. Il valore predefinito di **MQCSP.AuthenticationType** è MQCSP\_AUTH\_NONE. Con l'impostazione predefinita, non viene eseguita alcuna protezione con password. Per ulteriori informazioni, consultare [AuthenticationType](#).
- Se il client è IBM MQ Explorer e la modalità di compatibilità di identificazione utente non è abilitata, che non è quella predefinita. Questa condizione è applicabile solo a Esplora risorse di IBM MQ.

Se queste condizioni non vengono soddisfatte, la parola d'ordine viene inviata in testo semplice a meno che non sia proibita dall'impostazione di configurazione **PasswordProtection**.

### **L'impostazione di configurazione PasswordProtection**

L'attributo **PasswordProtection** nella sezione Canali dei file di configurazione .ini del client e del gestore code può impedire l'invio delle password in testo semplice. L'attributo può assumere 1 di 3 valori. Il valore predefinito è `compatible`:

#### **Compatible**

La password può essere inviata in testo semplice se il gestore code o il client sta eseguendo una versione precedente a IBM MQ 8.0. In altre parole, le password in testo semplice sono consentite per la compatibilità.

Pertanto:

- La parola d'ordine viene inviata codificata da TLS CipherSpec se viene utilizzata la codifica TLS e CipherSpec non è null.
- La password viene inviata in testo semplice se il gestore code o il client sta eseguendo una versione precedente a IBM MQ 8.0 e la crittografia TLS non viene utilizzata. La parola d'ordine viene inviata in testo semplice poiché le versioni precedenti a IBM MQ 8.0 possono inviare le password solo in testo semplice.
- La password viene inviata protetta se sia il gestore code che il client sono in esecuzione su IBM MQ 8.0 o versioni successive e viene utilizzata una CipherSpec null o la codifica TLS non viene utilizzata. **MQCSP.AuthenticationType** deve essere impostato su MQCSP\_AUTH\_USER\_ID\_AND\_PWD.

- La connessione non riesce prima che la password venga inviata se sia il gestore code che il client sono in esecuzione su IBM MQ 8.0 o versioni successive e **MQCSP.AuthenticationType** non è impostato su MQCSP\_AUTH\_USER\_ID\_AND\_PWD.

#### **sempre**

La parola d'ordine deve essere codificata con un CipherSpec che non sia un CipherSpecnull o **MQCSP.AuthenticationType** deve essere impostato su MQCSP\_AUTH\_USER\_ID\_AND\_PWD. Altrimenti, la connessione non riesce. In altre parole, le password in testo semplice non sono consentite.

Pertanto:

- La parola d'ordine viene inviata codificata da TLS CipherSpec se viene utilizzata la codifica TLS e CipherSpec non è null.
- La password viene inviata protetta se sia il gestore code che il client eseguono una versione di IBM MQ alle IBM MQ 8.0 o successive e non viene utilizzata la crittografia TLS oppure se viene utilizzata una CipherSpec null. **MQCSP.AuthenticationType** deve essere impostato su MQCSP\_AUTH\_USER\_ID\_AND\_PWD.
- La connessione non riesce prima che la password venga inviata se il gestore code o il client eseguono una versione precedente a IBM MQ 8.0e la crittografia TLS non viene utilizzata. Poiché le versioni precedenti a IBM MQ 8.0 possono inviare le password solo in testo semplice e always richiede che la password sia codificata o protetta, la connessione non riesce.

#### **facoltativo**

La password può essere facoltativamente inviata protetta, ma viene inviata in testo semplice se **MQCSP.AuthenticationType** non è impostato su MQCSP\_AUTH\_USER\_ID\_AND\_PWD. In altre parole, le password in testo semplice possono essere inviate da qualsiasi client.

Pertanto:

- La parola d'ordine viene inviata codificata da TLS CipherSpec se viene utilizzata la codifica TLS e CipherSpec non è null.
- La password viene inviata in testo semplice se viene utilizzato un CipherSpec null e **MQCSP.AuthenticationType** non è impostato su MQCSP\_AUTH\_USER\_ID\_AND\_PWD.
- La password viene inviata in testo semplice se il gestore code o il client sta eseguendo una versione precedente a IBM MQ 8.0e la crittografia TLS non viene utilizzata. La parola d'ordine viene inviata in testo semplice poiché le versioni precedenti a IBM MQ 8.0 possono inviare le password solo in testo semplice.
- La password viene inviata protetta se sia il gestore code che il client sono in esecuzione su IBM MQ 8.0 o versioni successive, se la crittografia TLS non viene utilizzata o se viene utilizzata una CipherSpec null e **MQCSP.AuthenticationType** è impostato su MQCSP\_AUTH\_USER\_ID\_AND\_PWD.

Per i client Java e JMS, il funzionamento dell'attributo **PasswordProtection** cambia in base alla scelta di utilizzare la modalità di compatibilit ... o la modalit ... MQCSP:

- Se i client Java e JMS operano in modalità di compatibilità, una struttura MQCSP non viene eseguita durante l'elaborazione della connessione. Pertanto, il comportamento dell'attributo **PasswordProtection** è lo stesso descritto per i clienti che eseguono una versione di IBM MQ precedente a IBM MQ 8.0.
- Se i client Java e JMS funzionano in modalità MQCSP, il funzionamento dell'attributo **PasswordProtection** è quello descritto.

Per ulteriori informazioni sull'autenticazione della connessione con i client Java e JMS, consultare [“Autenticazione della connessione con il client Java” a pagina 69.](#)

### **Digital Certificate Manager (DCM)**

Utilizzare DCM per gestire i certificati digitali e le chiavi private su IBM i.

DCM (Digital Certificate Manager ) consente di gestire i certificati digitali e di utilizzarli in applicazioni sicure sul server IBM i . Con Digital Certificate Manager, è possibile richiedere ed elaborare certificati

digitali da CA (Certificate Authority) o altre terze parti. È anche possibile agire come una CA (Certificate Authority) locale per creare e gestire certificati digitali per gli utenti.

DCM supporta anche l'utilizzo di CRL (Certificate Revocation Lists) per fornire un processo di convalida di applicazioni e certificati più forte. È possibile utilizzare DCM per definire l'ubicazione in cui risiede uno specifico CRL CA (Certificate Authority) su un server LDAP in modo che IBM MQ possa verificare che un certificato specifico non sia stato revocato.

DCM supporta e può rilevare automaticamente i certificati in diversi formati. Quando DCM rileva un certificato codificato PKCS #12 o un certificato PKCS #7 che contiene dati codificati, richiede automaticamente all'utente di immettere la parola d'ordine utilizzata per codificare il certificato. DCM non richiede certificati PKCS #7 che non contengono dati codificati.

DCM fornisce un'interfaccia utente basata sul browser che è possibile utilizzare per gestire i certificati digitali per le applicazioni e gli utenti. L'interfaccia utente è divisa in due frame principali: un frame di navigazione e un frame di attività.

Si utilizza il frame di navigazione per selezionare le attività per gestire i certificati o le applicazioni che li utilizzano. Alcune singole attività vengono mostrate direttamente nel frame di navigazione principale, ma la maggior parte delle attività nel frame di navigazione sono organizzate in categorie. Ad esempio, Gestisci certificati è una categoria di attività che contiene diverse singole attività guidate, come Visualizza certificato, Rinnova certificato e Importa certificato. Se un elemento nel frame di navigazione è una categoria che contiene più di un'attività, viene visualizzata una freccia a sinistra. La freccia indica che quando si seleziona il link della categoria, viene visualizzato un elenco espanso di attività, che consente di scegliere quale attività eseguire.

Per informazioni importanti su DCM, consultare le seguenti pubblicazioni IBM Redbooks :

- *IBM i Sicurezza di rete cablato: OS/400 V5R1 DCM and Cryptographic Enhancements*, SG24-6168. In particolare, consultare le appendici per informazioni essenziali sull'impostazione del proprio sistema IBM i come CA locale.
- *AS/400 Sicurezza Internet: sviluppo di una DCA (Digital Certificate Infrastructure)*, SG24-5659. In particolare, si veda il capitolo 5. *Digital Certificate Manager per AS/400* , che spiega AS/400 DCM.


### **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 di crittografia che possono essere utilizzate sui canali TLS.

Queste informazioni si applicano alle piattaforme seguenti:

- Windows
- UNIX and Linux
- z/OS

La conformità FIPS 140-2 di una connessione TLS IBM MQ su UNIX, Linux, and Windows si trova qui [“FIPS \(Federal Information Processing Standards\) per UNIX, Linux, and Windows” a pagina 33.](#)

 La conformità FIPS 140-2 di una connessione TLS IBM MQ su z/OS si trova qui [“FIPS \(Federal Information Processing Standards\) per z/OS” a pagina 35.](#)

Se l'hardware di crittografia è presente, i moduli di codifica utilizzati da IBM 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 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 262](#)

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

[“Utilizzo di runmqckm, runmqakme strmqikm per gestire i certificati digitali” a pagina 278](#)

Su sistemi UNIX, Linux, and Windows , gestire chiavi e certificati digitali con **strmqikm** (iKeyman) Dalla GUI o dalla riga comandi utilizzando **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd).

## Riferimenti correlati

[“FIS \(Federal Information Processing Standards\)” a pagina 20](#)

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).

## Informazioni correlate

[Abilitazione di TLS in IBM MQ classes for Java](#)

[Proprietà TLS degli oggetti JMS](#)

[Utilizzo di TLS \(Transport Layer Security\) con IBM MQ classes for JMS](#)

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

Quando la crittografia è richiesta su un canale SSL/TLS su sistemi Windows, UNIX and Linux , IBM MQ utilizza un pacchetto di crittografia denominato ICC ( IBM Crypto for C). Sulle piattaforme Windows UNIX and Linux , il software ICC ha superato il programma di convalida crittografica FIPS (Federal Information Processing Standards) del National Institute of Standards and Technology degli Stati Uniti, al livello 140-2.

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

- Per tutti i canali di messaggi IBM 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 IBM MQ MQI client , 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 applicazioni IBM MQ classes for Java che utilizzano la modalità client, la connessione utilizza le implementazioni 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 applicazioni IBM MQ classes for JMS che utilizzano la modalità client, la connessione utilizza le implementazioni 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 JMS .
  - Tutti i repository di chiavi sono stati creati e manipolati utilizzando solo software compatibile con FIPS, come **runmqakm** con l'opzione **-fips** .
- Per le applicazioni client .NET non gestite, 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 che deve essere utilizzata solo la 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 le applicazioni client XMS .NET non gestite, 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 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 supportate sono certificate FIPS 140-2 tranne come indicato nel file readme incluso con ogni fix pack o pacchetto di aggiornamento.

Per le connessioni 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, 2023.
@ (#) 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 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 IBM 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, IBM MQ non reimposta la chiave di 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 427](#).

IBM 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 262](#)

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

[“Utilizzo di runmqckm, runmqakme strmqikm per gestire i certificati digitali” a pagina 278](#)

Su sistemi UNIX, Linux, and Windows , gestire chiavi e certificati digitali con **strmqikm** (iKeyman) Dalla GUI o dalla riga comandi utilizzando **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd).

### Riferimenti correlati

[“FIS \(Federal Information Processing Standards\)” a pagina 20](#)

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).

### Informazioni correlate

[Abilitazione di TLS in IBM MQ classes for Java](#)

[Proprietà TLS degli oggetti JMS](#)

[Utilizzo di TLS \(Transport Layer Security\) con IBM MQ classes for JMS](#)

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

Quando la crittografia è richiesta su un canale SSL/TLS su z/OS , IBM MQ utilizza un servizio denominato SSL di sistema. L'obiettivo di System SSL è di fornire la capacità di eseguire in modo sicuro in una modalità progettata per aderire al programma di convalida crittografico FIPS (Federal Information Processing Standards) del National Institute of Standards and Technology degli Stati Uniti, al livello 140-2.

Quando si implementano connessioni conformi a FIPS 140-2 con connessioni TLS IBM MQ , è necessario considerare diversi aspetti:

- Per abilitare i canali di messaggi IBM MQ per la conformità FIPS, verificare che siano soddisfatte le seguenti condizioni:
  - L'FMID di livello 3 della sicurezza SSL di sistema è installato e configurato (consultare [Pianificazione dell'installazione di IBM MQ](#) ).

- I moduli SSL di sistema vengono convalidati.
- L'attributo SSLFIPS del gestore code è stato impostato su **YES**.

Durante l'esecuzione in modalità FIPS, SSL di sistema utilizza CP Assist for Cryptographic Function (CPACF) quando disponibile. Le funzioni di crittografia eseguite dall'hardware supportato da ICSF durante l'esecuzione in modalità non FIPS continuano ad essere sfruttate durante l'esecuzione in modalità FIPS, ad eccezione della generazione della firma RSA che deve essere eseguita nel software.

*Tabella 2. Differenze tra la modalità FIPS e il supporto dell' algoritmo modalità non FIPS.*

Algoritmo	Non FIPS		FIPS	
	Dimensioni chiave	Hardware	Dimensioni chiave	Hardware
RC2	40 e 128			
RC4	40 e 128			
DES	56	x		
TDI	168	x	168	x
AES	128 e 256	x	128 e 256	x
MD5	48			
SHA-1	160	x	160	x
SHA-2	224, 256, 384 e 512	x	224, 256, 384 e 512	x
RSA	512-4096	x	1024-4096	x
DSA	512-1024		1024	
DH	512-2048		2048	

In modalità FIPS, System SSL può utilizzare solo i certificati che utilizzano gli algoritmi e le dimensioni delle chiavi mostrati nella Tabella 1. Durante la convalida del certificato X.509 se viene rilevato un algoritmo incompatibile con la modalità FIPS, il certificato non può essere utilizzato e viene considerato non valido.

Per le applicazioni delle classi IBM MQ che utilizzano la modalità client all'interno di WebSphere Application Server , fare riferimento a [Supporto FIPS \(Federal Information Processing Standard\)](#).

Per informazioni sulla configurazione del modulo SSL di sistema, vedere [Impostazione verifica modulo SSL di sistema](#) .

#### **Riferimenti correlati**

[“FIS \(Federal Information Processing Standards\)” a pagina 20](#)

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/TLS su IBM MQ MQI client**

IBM MQ supporta TLS sui client. È possibile personalizzare l'utilizzo di TLS in diversi modi.

IBM MQ fornisce il supporto TLS per IBM MQ MQI clients su sistemi Windows, UNIX and Linux . Se si utilizza IBM MQ classes for Java, consultare [Utilizzo di IBM MQ classes for Java](#) e se si utilizza IBM MQ classes for JMS, consultare [Utilizzo di IBM MQ classes for JMS](#). Il resto di questa sezione non si applica agli ambienti Java o JMS .

Puoi specificare il repository delle chiavi per un IBM MQ MQI client con il valore MQSSLKEYR nel tuo file di configurazione del client IBM MQ o quando la tua applicazione effettua una chiamata MQCONN. Hai tre opzioni per specificare che un canale utilizza TLS:

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

Non è possibile utilizzare la variabile di ambiente MQSERVER per specificare che il canale utilizza TLS.

Puoi continuare ad eseguire le tue applicazioni IBM MQ MQI client senza TLS, purché TLS non sia specificato all'altra estremità del canale.

Se vengono apportate modifiche su una macchina client al contenuto del repository delle chiavi TLS, all'ubicazione del repository delle chiavi TLS, alle informazioni di autenticazione o ai parametri hardware crittografici, è necessario terminare tutte le connessioni TLS 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 TLS. Vengono utilizzate tutte le nuove impostazioni TLS. Queste impostazioni sono analoghe a quelle aggiornate dal comando REFRESH SECURITY TYPE (SSL) sui sistemi del gestore code.

Quando IBM MQ MQI client 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 TLS e FIPS sono supportate su IBM MQ MQI clients. Per ulteriori informazioni, consultare [“Reimpostazione delle chiavi segrete SSL e TLS” a pagina 427](#) e [“FIPS \(Federal Information Processing Standards\) per UNIX, Linux, and Windows” a pagina 33](#).

Consultare [“Impostazione della sicurezza IBM MQ MQI client” a pagina 261](#) per ulteriori informazioni sul supporto TLS per IBM MQ MQI clients.

### **Informazioni correlate**

[Configurazione di un client utilizzando un file di configurazione](#)

#### *Specifica che un canale MQI utilizza SSL/TLS*

Per un canale MQI per utilizzare TLS, il valore dell'attributo *SSLCipherSpec* del canale di connessione client deve essere il nome di un CipherSpec supportato da IBM 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 IBM MQ MQI client emette una chiamata MQCONN.

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 MQCONN.

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 su Windows.

Sui sistemi Windows , è possibile utilizzare il comando di controllo **setmqscp** per pubblicare le definizioni di canale di collegamento 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 MQCONN, questa definizione viene utilizzata di preferenza rispetto a tutte le voci in una tabella di definizione di canale client a cui il client IBM MQ può accedere.

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

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 IBM MQ MQI client” a pagina 416

Si dispone di tre opzioni per specificare un CipherSpec per un IBM MQ MQI client.

### CipherSpecs e CipherSuites in IBM MQ

Da IBM MQ 9.0.0 Fix Pack 3 e IBM MQ 9.0.5, IBM MQ supporta gli algoritmi TLS V1.2 CipherSpecse RSA e Diffie - Hellman. Tuttavia, è possibile abilitare CipherSpecsobsoleti, se necessario.

Consultare “Abilitazione di CipherSpecs” a pagina 408 per informazioni su:

- CipherSpecs supportati da IBM MQ
- Come abilitare CipherSpecs obsoleti

IBM MQ supporta gli algoritmi di autenticazione e scambio di chiavi RSA e Diffie - Hellman. La dimensione della chiave utilizzata durante l'handshake TLS 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 19

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 MQ

Questo argomento fornisce informazioni su come configurare IBM MQ su 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 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 7.5 elenca la versione di Suite B applicata da ciascun livello di manutenzione del prodotto. Se si configura IBM MQ per applicare la conformità Suite B, consultare sempre il file readme quando si prevede di applicare la manutenzione. Consultare [Lecture del prodotto IBM MQ, WebSphere MQe Serie MQ](#).

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

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

Tabella 3. Livelli di sicurezza della suite B con CipherSpecs e algoritmi di firma digitale consentiti (Continua)

Livello di sicurezza	CipherSpecs consentiti	Algoritmi di firma digitale consentiti
192 bit	ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA con SHA-384
Entrambi <sup>1</sup>	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

Sebbene il comportamento predefinito di IBM MQ non sia conforme allo standard Suite B, IBM MQ può essere configurato per essere conforme a uno o entrambi i livelli di sicurezza sui sistemi Windows, UNIX and Linux . In seguito alla corretta configurazione di IBM 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 IBM MQ CipherSpecs, consultare [“Abilitazione di CipherSpecs”](#) a pagina 408

## 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 IBM 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:** I comandi **runmqckm** e **strmqikm** 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 UNIX, Linux, and Windows”](#) a pagina 285

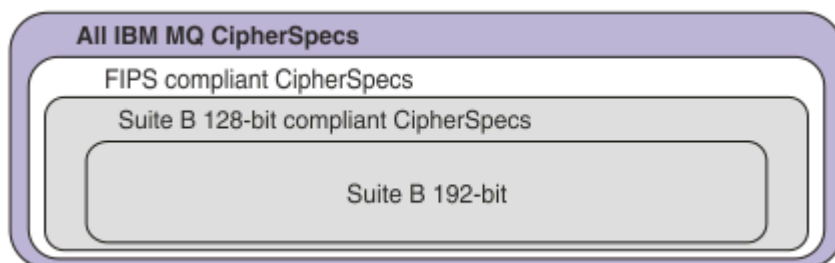
Per richiedere un certificato digitale firmato CA per l'utilizzo di produzione Suite B, consultare [“Richiesta di un certificato personale su UNIX, Linux, and Windows”](#) a pagina 288.

**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 MQ è configurata per operazioni compatibili con FIPS 140-2. È quindi possibile configurare IBM MQ per la conformità sia FIPS che Suite B contemporaneamente, nel qual caso si applicano entrambe le serie di restrizioni.

Il seguente diagramma illustra la relazione tra questi sottoinsiemi:



## Configurazione di IBM MQ per operazioni conformi a Suite B

Per informazioni su come configurare IBM MQ su Windows, UNIX and Linux per operazioni conformi a Suite B, consultare [“Configurazione di IBM MQ per Suite B”](#) a pagina 40.

IBM MQ non supporta operazioni conformi a Suite B sulle piattaforme IBM i e z/OS . Anche i clienti IBM 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 262

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

### Configurazione di IBM MQ per Suite B

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

La suite B limita la serie di algoritmi di crittografia abilitati per fornire un livello di sicurezza garantito. IBM MQ può essere configurato per operare in conformità con la Suite B per fornire un livello migliorato di sicurezza. Per ulteriori informazioni sulla Suite B, consultare [“Crittografia della National Security Agency \(NSA\) Suite B”](#) a pagina 21. Per ulteriori informazioni sulla configurazione della Suite B e sul suo effetto sui canali TLS, consultare [“NSA Suite B Crittografia in IBM MQ”](#) a pagina 38.

## 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 **MQCMD\_CHANGE\_Q\_MGR** PCF con il parametro **MQIA\_SUITE\_B\_STRENGTH** per configurare il gestore code per l'operazione conforme a Suite B.



**Nota:** Se si modificano le impostazioni della Suite B del gestore code, è necessario riavviare il servizio MQXR per rendere effettive tali impostazioni.

## Client MQI

Per impostazione predefinita, i client MQI non applicano la conformità Suite B. È possibile abilitare il client MQI per la conformità Suite B eseguendo una delle opzioni riportate di seguito:

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 .NET client non gestiti, la proprietà **MQC. ENCRYPTION\_POLICY\_SUITE\_B** indica il tipo di sicurezza Suite B richiesto.

Per informazioni sull'utilizzo di Suite B in IBM MQ classes for .NET, vedi [MQEnvironment .NET class](#).

## AMQP

► V 9.0.0

Le impostazioni dell'attributo Suite B per un gestore code si applicano ai canali AMQP su tale gestore code. Se si modificano le impostazioni della suite B del gestore code, è necessario riavviare il servizio AMQP per rendere effettive le modifiche.

## **Politiche di convalida dei certificati in IBM 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 applicazioni Java e JMS su tutte le piattaforme, la politica di convalida dei certificati 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 IBM i , la politica di convalida dei certificati dipende dalla libreria di socket sicuri fornita dal sistema operativo. Per ulteriori informazioni relative alla politica di convalida del certificato, consultare la documentazione per il sistema operativo.
- Per i sistemi z/OS , la politica di convalida del certificato dipende dal componente SSL di sistema fornito dal sistema operativo. Per ulteriori informazioni relative alla politica di convalida del certificato, consultare la documentazione per il sistema operativo.
- 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 del certificato su UNIX, Linux, and Windows, vedere [“Configurazione delle politiche di convalida dei certificati in IBM MQ”](#) a pagina 42. Per ulteriori informazioni sulle differenze tra le politiche di convalida dei certificati di base e standard, consultare [Certificate validation and trust policy design on UNIX, Linux, and Windows](#).

## **Configurazione delle politiche di convalida dei certificati in IBM MQ**

È possibile specificare quale politica di convalida del certificato 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 MQ”](#) a pagina 42.

## **Certificati digitali e compatibilità CipherSpec in IBM 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 MQ.

Nelle release precedenti a IBM WebSphere MQ 7.1, tutti i CipherSpecs TLS supportati utilizzavano l'algoritmo RSA per firme digitali e accordo chiave. Tutti i tipi di certificati digitali supportati erano

compatibili con tutti i CipherSpecs supportati, quindi è stato possibile modificare la CipherSpec per qualsiasi canale senza dover modificare i certificati digitali.

In IBM WebSphere MQ 7.1 e versioni successive, solo un sottoinsieme dei CipherSpecs supportati può essere utilizzato con tutti i tipi supportati di certificati digitali. 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. Per utilizzare CipherSpecs più recenti basati sul protocollo TLS 1.2, assicurarsi che i certificati digitali non utilizzino l'algoritmo MD5 nelle relative firme digitali. I CipherSpecs meno recenti che utilizzano i protocolli TLS 1.0 non sono soggetti a questa restrizione e possono continuare a utilizzare i certificati con le 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 della firma digitale da visualizzare. Per i dettagli, consultare [Etichetta certificato digitale](#).

**Nota:** Sebbene sia possibile utilizzare **runmqckm** (iKeycmd) e la GUI **stimqikm** (iKeyman) per visualizzare una selezione di algoritmi di firma digitale, lo strumento **runmqakm** fornisce un intervallo più ampio.

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

```
Label : ibmmqexample
Key Size : 1024
Version : X509 V3
Serial : 4e4e93f1
Issuer : CN=Old Certificate Authority,OU=Test,O=Example,C=US
Subject : CN=Example Queue Manager,OU=Test,O=Example,C=US
Not Before : August 19, 2011 5:48:49 PM GMT+01:00
Not After : August 18, 2012 5:48:49 PM GMT+01:00
Public Key
 30 81 9F 30 0D 06 09 2A 86 48 86 F7 0D 01 01 01
 05 00 03 81 8D 00 30 81 89 02 81 81 00 98 5A 7A
 F0 18 21 EE E4 8A 6E DE C8 01 4B 3A 1E 41 90 3D
 CE 01 3F E6 32 30 6C 23 59 F0 FE 78 6D C2 80 EF
 BC 83 54 7A EB 60 80 62 6B F1 52 FE 51 9D C1 61
 80 A5 1C D4 F0 76 C7 15 6D 1F 0D 4D 31 3E DC C6
 A9 20 84 6E 14 A1 46 7D 4C F5 79 4D 37 54 0A 3B
 A9 74 ED E7 8B 0F 80 31 63 1A 0B 20 A5 99 EE 0A
 30 A6 B6 8F 03 97 F6 99 DB 6A 58 89 7F 27 34 DE
 55 08 29 D8 A9 6B 46 E6 02 17 C3 13 D3 02 03 01
 00 01
Public Key Type : RSA (1.2.840.113549.1.1.1)
Fingerprint : SHA1 :
 09 4E 4F F2 1B CB C1 F4 4F 15 C9 2A F7 32 0A 82
 DA 45 92 9F
Fingerprint : MD5 :
 44 54 81 7C 58 68 08 3A 5D 75 96 40 D5 8C 7A CB
Fingerprint : SHA256 :
 3B 47 C6 E7 7B B0 FF 85 34 E7 48 BE 11 F2 D4 35
 B7 9A 79 53 2B 07 F5 E7 65 E8 F7 84 E0 2E 82 55
Signature Algorithm : MD5WithRSASignature (1.2.840.113549.1.1.4)
Value
 3B B9 56 E6 F2 77 94 69 5B 3F 17 EA 7B 19 D0 A2
 D7 10 38 F1 88 A4 44 1B 92 35 6F 3B ED 99 9B 3A
 A5 A4 FC 72 25 5A A9 E3 B1 96 88 FC 1E 9F 9B F1
 C5 E8 8E CF C4 8F 48 7B 0E A6 BB 13 AE 2B BD D8
 63 2C 03 38 EF DC 01 E1 1F 7A 6F FB 2F 65 74 D0
 FD 99 94 BA B2 3A D5 B4 89 6C C1 2B 43 6D E2 39
```

66 6A 65 CB C3 C4 E2 CC F5 49 39 A3 8B 93 5A DD  
B0 21 0B A8 B2 59 5B 24 59 50 44 89 DC 78 19 51  
Trust Status : Enabled

La 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 TLS IBM MQ, ma sono particolarmente rilevanti per gli utenti della crittografia Elliptic Curve.

La seguente tabella riassume le relazioni tra CipherSpecs e i certificati digitali:

Tipo	Prefisso nome CipherSpec	Descrizione	Tipo di chiave pubblica richiesto	Algoritmo di crittografia della firma digitale	Metodo di creazione e 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 chiavi pubbliche RSA, chiavi segrete Elliptic Curve e algoritmi di firma digitale RSA.	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 non sono supportati da gestori code IBM MQ e client MQI sulla piattaforma IBM i.

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.

È possibile configurare un canale con un CipherSpec che richiede un certificato EC (Elliptic Curve) e un'etichetta di certificato per un certificato RSA o viceversa. È necessario assicurarsi che il certificato indicato nell'etichetta del certificato sia appropriato per il canale CipherSpec.

Supponendo di aver configurato correttamente IBM MQ, è possibile avere:

- Un singolo gestore code con una combinazione di certificati RSA e EC.
- Canali differenti sullo stesso gestore code che utilizzano un certificato RSA o EC.

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 oppure 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.

I CipherSpecs di tipo 3 richiedono che il certificato personale abbia 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 IBM 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.

Se è necessario utilizzare diversi tipi di CipherSpec sullo stesso gestore code o sulla stessa applicazione client, configurare un'etichetta di certificato appropriata e una combinazione CipherSpec sulla definizione client.

I tre tipi di CipherSpec non interagiscono direttamente: questa è una limitazione degli standard 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 CipherSpec di tipo 1, quindi l'estremità del mittente remoto del canale TO.QM1 deve avere un certificato personale con una chiave Elliptic Curve e una firma digitale basata su ECDSA. Se il canale mittente remoto non soddisfa questi requisiti, l'avvio del canale non riesce.

Gli altri canali che si collegano al gestore code QM1 possono utilizzare altri CipherSpecs, purché ciascun canale utilizzi un certificato del tipo corretto per CipherSpec di quel canale. Ad esempio, si supponga che QM1 utilizzi un canale mittente denominato TO.QM2 per inviare messaggi a un altro gestore code denominato QM2. Il canale TO.QM2 potrebbe utilizzare il tipo 3 CipherSpec TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA256 purché entrambe le estremità del canale utilizzino certificati contenenti chiavi pubbliche RSA. L'attributo del canale dell'etichetta del certificato può essere utilizzato per configurare un certificato differente per ogni canale.

Quando pianifichi le tue reti IBM MQ , considera attentamente quali canali richiedono TLS e assicurati che il tipo di certificato utilizzato per ogni canale sia appropriato per l'utilizzo con la CipherSpec su tale canale.

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. Per i dettagli, consultare [Etichetta certificato digitale](#) .

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

```
Label : ibmmqexample
Key Size : 384
Version : X509 V3
Serial : 9ad5eeef5d756f41
Issuer : CN=Example Certificate Authority,OU=Test,O=Example,C=US
Subject : CN=Example Queue Manager,OU=Test,O=Example,C=US
Not Before : 21 August 2011 13:10:24 GMT+01:00
Not After : 21 August 2012 13:10:24 GMT+01:00
Public Key
 30 76 30 10 06 07 2A 86 48 CE 3D 02 01 06 05 2B
 81 04 00 22 03 62 00 04 3E 6F A9 06 B6 C3 A0 11
 F8 D6 22 78 FE EF 0A FE 34 52 C0 8E AB 5E 81 73
 D0 97 3B AB D6 80 08 E7 31 E9 18 3F 6B DE 06 A7
```

```

15 D6 9D 5B 6F 56 3B 7F 72 BB 6F 1E C9 45 1C 46
60 BE F2 DC 1B AD AC EC 64 4C 0E 06 65 6E ED 93
B8 F5 95 E0 F9 2A 05 D6 21 02 BD FB 06 63 A1 CC
66 C6 8A 0A 5C 3F F7 D3
Public Key Type : EC_ecPublicKey (1.2.840.10045.2.1)
Fingerprint : SHA1 :
3C 34 58 04 5B 63 5F 5C C9 7A E7 67 08 2B 84 43
3D 43 7A 79
Fingerprint : MD5 :
49 13 13 E1 B2 AC 18 9A 31 41 DC 8C B4 D6 06 68
Fingerprint : SHA256 :
6F 76 78 68 F3 70 F1 53 CE 39 31 D9 05 C5 C5 9F
F2 B8 EE 21 49 16 1D 90 64 6D AC EB 0C A7 74 17
Signature Algorithm : EC_ecdsa_with_SHA384 (1.2.840.10045.4.3.3)
Value
30 65 02 30 0A B0 2F 72 39 9E 24 5A 22 FE AC 95
0D 0C 6D 6C 2F B3 E7 81 F6 C1 36 1B 9A B0 6F 07
59 2A A1 4C 02 13 7E DD 06 D6 FE 4B E4 03 BC B1
AC 49 54 1E 02 31 00 90 0E 46 2B 04 37 EE 2C 5F
1B 9C 69 E5 99 60 84 84 10 71 1A DA 63 88 33 E2
22 CC E6 1A 4E F4 61 CC 51 F9 EE A0 8E F4 DC B5
0B B9 72 58 C3 C7 A4
Trust Status : Enabled

```

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 anche possibile utilizzare il comando **runmqckm** con gli stessi parametri. Inoltre, la GUI di **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, è necessario 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 IBM MQ è configurato per essere conforme al profilo TLS 1.2 conforme a Suite B, gli CipherSpecs consentiti e gli algoritmi di firma digitale sono limitati come descritto in “NSA Suite B Crittografia in IBM MQ” a pagina 38. 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 5. 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

“Abilitazione di CipherSpecs” a pagina 408

Abilitare 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 262

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

“NSA Suite B Crittografia in IBM MQ” a pagina 38

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

“Crittografia della National Security Agency (NSA) Suite B” a pagina 21

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.

## 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 impostato uno strumento di amministrazione come IBM MQ Explorer, e un canale per tale specifico uso, è consigliabile assicurarsi che solo specifici computer client 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**.

<b>Tipo di canale</b>	<b>MCA dove vengono applicate le regole CHLAUTH</b>
SDR-RCVR	RCVR
RQSTR-SVR (avviato a SVR)	RQSTR
RQSTR-SVR (avviato a RQSTR)	SVR
RQSTR-SDR (avviato a SDR)	RQSTR

<i>Tabella 6. Dove le regole CHLAUTH vengono applicate per diverse coppie di canali (Continua)</i>	
<b>Tipo di canale</b>	<b>MCA dove vengono applicate le regole CHLAUTH</b>
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, potrebbero esserci delle occasioni in cui si riscontrano dei tentativi di connessione da un indirizzo IP che non dovrebbe avere accesso al sistema IBM MQ ed è necessario bloccare temporaneamente l'indirizzo prima che il firewall possa essere aggiornato. Questi tentativi di connessione potrebbero non provenire dai canali IBM MQ; questi tentativi di connessione potrebbero provenire da altre applicazioni socket erroneamente configurate per avere come obiettivo il listener IBM MQ utilizzato. 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 373.



## **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 MQRChannel\_BLOCKED con identificativo motivo MQRChannel\_BLOCKED\_USERID, a condizione che gli eventi del canale siano abilitati.

Per un esempio, consultare [“Blocco di ID utente specifici” a pagina 375.](#)

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 MQRChannel\_BLOCKED con il qualificatore motivo MQRChannel\_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 client” a pagina 378.](#)

## **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 MQRChannel\_BLOCKED con il qualificatore motivo MQRChannel\_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 378.](#)

## **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 MQRChannel\_BLOCKED con il qualificatore motivo MQRChannel\_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 o TLS” a pagina 379.](#)

## **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 inoltramento 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 379.](#)

## **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 376.](#)

## **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 MQI IBM MQ è necessario utilizzare un MCAUSER specificato differente, impostare un record di autenticazione del canale di tipo USERMAP. L'associazione degli ID utente non utilizza caratteri jolly.

Per un esempio, consultare [“Associazione di un ID utente client a un ID utente MCAUSER” a pagina 376.](#)

## **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 377.](#)

## **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 376.](#)

## **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 host, nome del gestore code o DN SSL o TLS, viene utilizzata la corrispondenza più specifica. La corrispondenza viene considerata nel seguente modo:

- La più specifica è un nome senza caratteri jolly, ad esempio:
  - Un nome canale come A.B.C
  - Un indirizzo IP come 192.0.2.6
  - Un nome host come hursley.ibm.com
  - Un nome gestore code come 192.0.2.6
- La più generica è un asterisco (\*) che corrisponde, ad esempio, a:
  - Tutti i nomi canale
  - Tutti gli indirizzi IP
  - Tutti i nomi host
  - Tutti i nomi gestore code
- Un pattern con un asterisco all'inizio di una stringa è più generico di un valore definito all'inizio di una stringa:
  - Ad esempio, \*.B.C è più generico di A.\*
  - Per gli indirizzi IP, \*.0.2.6 è più generico di 192.\*
  - Per i nomi host, \*.ibm.com è più generico di hursley.\*
  - Per i nomi gestore code, \*QUEUEMANAGER è più generico di QUEUEMANAGER\*
- Un pattern con un asterisco in una specifica posizione in una stringa è più generico di un valore definito nella stessa posizione della stringa, lo stesso vale per ogni posizione successiva:
  - Per i canali, A.\*C è più generico di A.B.\*
  - Per gli indirizzi IP, 192.\*.2.6 è più generico di 192.0.\*.
  - Per i nomi host, hursley.\*.com è più generico di hursley.ibm.\*
  - Per i nomi gestore code, Q\*MANAGER è più generico di QUEUE\*
- Laddove due o più pattern contengono un asterisco in una specifica posizione in una stringa, quello con la quantità minore di nodi dopo l'asterisco è più generico:
  - Per i canali, A.\* è più generico di A.\*C
  - Per gli indirizzi IP, 192.\* è più generico di 192.\*.2.\*.
  - Per i nomi host, hursley.\* è più generico di hursley.\*.com

- Per i nomi gestore code, Q\* è più generico di Q\*MGR
- Inoltre, per un indirizzo IP:
  - 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 pattern può contenere asterischi aggiuntivi a condizione che nessun asterisco sia adiacente all'asterisco finale. Ad esempio, 192.\*.2.\* è valido, ma 192,0.\*\* non è valida.
  - Un pattern 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 DN (Distinguished Name) SSL o TLS, l'ordine di precedenza delle sottostringhe è il seguente:

<i>Tabella 7. Ordine di precedenza delle sottostringhe</i>		
<b>Ordina</b>	<b>Sottostringa DN</b>	<b>Nome</b>
1	SERIALNUMBER=	Numero di serie del certificato
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, IBM 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 86

Questa sezione tratta gli aspetti della messaggistica remota della sicurezza.

## Interazione di CHLAUTH e CONNAUTH

Il modo in cui i record di autenticazione di canale (CHLAUTH) e di connessione (CONNAUTH) interagiscono in IBM MQ, nel caso di una sola conversazione su un canale.

## Diversi tipi di binding

IBM MQ supporta due metodi per la connessione di un'applicazione:

### Bind locali

Si applica quando l'applicazione e il gestore code si trovano sulla stessa immagine operativa. CHLAUTH non è rilevante per questo tipo di connessione dell'applicazione.

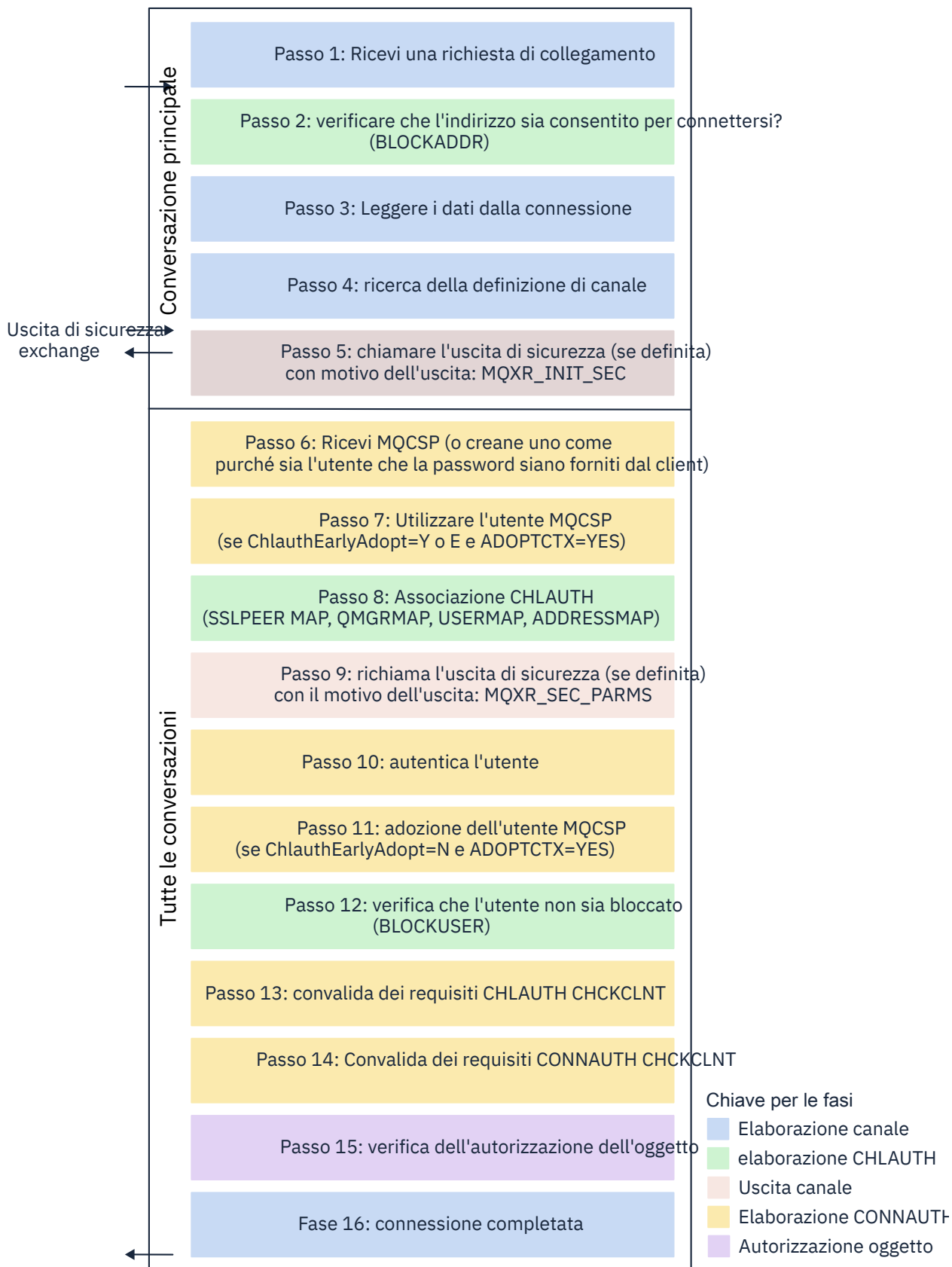
**Bind client**

Si applica quando l'applicazione e il gestore code utilizzano la rete per comunicare. L'applicazione e il gestore code possono essere in esecuzione sulla stessa macchina o su macchine differenti. In IBM MQ, una connessione client viene gestita sotto forma di un canale di connessione server (SVRCONN) e, in questa situazione, sono applicabili sia CONNAUTH che CHLAUTH.

**Passi di binding dell'estremità di ricezione di un canale**

Quando un'applicazione si connette a un gestore code, viene eseguita una quantità sostanziale di controllo per garantire che entrambe le estremità del canale comprendano ciò che è supportato dall'altra estremità. L'estremità di ricezione del canale esegue un ulteriore controllo, che coinvolge CHLAUTH e CONNAUTH, per garantire che il client sia autorizzato a connettersi e questo processo potrebbe includere anche un'uscita di sicurezza, poiché ciò potrebbe influire sul risultato. Questa fase di collegamento del canale viene anche indicata come *fase di bind*.

Il seguente diagramma elenca i passi che un canale SVRCONN passa attraverso quando viene avviato il server (nel gestore code):



### Passo 1: Ricevi una richiesta di connessione

L'iniziatore o il listener del canale riceve una richiesta di connessione da qualche parte sulla rete.

### Passo 2: L'indirizzo è autorizzato a connettersi?

Prima di leggere qualsiasi dato, IBM MQ controlla l'indirizzo IP del partner rispetto alle regole CHLAUTH, per vedere se l'indirizzo è nella regola BLOCKADDR. Se l'indirizzo non viene trovato e quindi non è bloccato, il flusso procede al passaggio successivo.

### Passo 3: Leggere i dati dal canale

IBM MQ ora legge i dati in un buffer e inizia a elaborare le informazioni inviate.

### Passo 4: ricerca della definizione di canale

Nel primo flusso di dati, IBM MQ invia, tra l'altro, il nome del canale che l'estremità mittente sta tentando di avviare. Il gestore code di ricezione può quindi ricercare la definizione di canale, che ha tutte le impostazioni specificate per il canale.

### Passo 5: chiamare l'uscita di sicurezza (se definita)

Se il canale ha un'uscita di sicurezza (SCYEXIT) definita, questa viene richiamata con il motivo dell'uscita (MQCXP.ExitReason) impostato su MQXR\_INIT\_SEC.

### Fase 6: ricevi MQCSP

Se necessario, crearne uno, purché l'ID utente e la password siano forniti dal client.

Se il client è un'applicazione Java o JMS in esecuzione in modalità di compatibilità, il client non trasmette una struttura MQCSP al gestore code. Invece, se l'applicazione ha fornito un ID utente e una password, viene creata qui una struttura MQCSP.

### Passo 7: adottare l'utente MQCSP (se ChlauthEarlyAdopt è Y e ADOPTCTX=YES)

L'ID utente dichiarato dal client è autenticato.

Se CONNAUTH sta utilizzando LDAP per associare un DN (distinguished name) asserito a un ID utente breve, l'associazione si verifica in questo passo.

Se l'autenticazione ha esito positivo, l'ID utente viene adottato dal canale e viene utilizzato dal passo di associazione CHLAUTH.

**Nota:** Da IBM MQ 9.0.4 il parametro **ChlauthEarlyAdopt= Y** viene aggiunto automaticamente alla stanza dei canali del file qm.ini per i nuovi gestori code.

### Passo 8: Associazione CHLAUTH

La cache CHLAUTH viene analizzata nuovamente per cercare le regole di associazione SSLPEERMAP, USERMAP, QMGRMAP e ADDRESSMAP.

Viene utilizzata la regola che corrisponde più specificamente al canale in ingresso. Se la regola ha USERSRC(CHANNEL) o (MAP), il canale continua il bind.

Se le regole CHLAUTH vengono valutate come una regola con USERSRC(NOACCESS), all'applicazione viene bloccato il collegamento al canale, a meno che le credenziali non vengano successivamente sovrascritte con un ID utente e una password validi nel passo 9.

### Passo 9: chiamare l'uscita di sicurezza (se definita)

Se il canale ha un'uscita di sicurezza (SCYEXIT) definita, questa viene richiamata con il motivo dell'uscita (MQCXP.ExitReason) impostare su MQXR\_SEC\_PARMS.

Un puntatore a MQCSP sarà presente nel campo SecurityParms della struttura MQCXP.

La struttura MQCSP ha dei puntatori all'ID utente (MQCSP.CSPUserIdPtr) e la password (MQCSP.CSPPasswordPtr).

È possibile modificare l'ID utente e la password nell'uscita. Il seguente esempio mostra come un'uscita di sicurezza dovrebbe stampare i valori ID utente e password in un log di verifica:

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



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


L'uscita può indicare a IBM MQ di chiudere il canale, restituendo *MQXCC\_CLOSE\_CHANNEL* in MQCXP.Campo **Exitresponse**. In alternativa, l'elaborazione del canale continua fino alla fase di autenticazione della connessione.

**Nota:** Se l'utente asserito viene modificato dall'uscita di sicurezza, le regole di associazione CHLAUTH non verranno riapplicate al nuovo utente.

### Passo 10: autentica l'utente

La fase di autenticazione si verifica se CONNAUTH è abilitato sul gestore code.

Per verificare ciò, immettere il comando MQSC 'DISPLAY QMGR CONNAUTH'.

 Il seguente esempio mostra l'output del comando **DISPLAY QMGR CONNAUTH** proveniente da un gestore code in esecuzione su IBM MQ for z/OS.


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

 Il seguente esempio mostra l'output del comando **DISPLAY QMGR CONNAUTH** da un gestore code in esecuzione su IBM MQ for Multiplatforms.

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

Il valore CONNAUTH è il nome di un oggetto **AUTHINFO** IBM MQ .

Poiché l'autenticazione del sistema operativo (**AUTHTYPE(IDPWOS)**) è valida sia su IBM MQ for Multiplatforms che su IBM MQ for z/OS, gli esempi utilizzano l'autenticazione del sistema operativo.

 Il seguente esempio mostra l'oggetto predefinito fornito per **AUTHTYPE(IDPWOS)** da un gestore code in esecuzione su IBM MQ for z/OS.

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

 Il seguente esempio mostra l'oggetto predefinito fornito per **AUTHTYPE(IDPWOS)** da un gestore code in esecuzione su IBM MQ for Multiplatforms.

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

AUTHINFO TYPE (IDPWOS) ha un attributo denominato CHCKCLNT. Se il valore viene modificato in REQUIRED, tutte le applicazioni client devono fornire un ID utente e una parola d'ordine validi.

Se l'utente è stato autenticato nel passo 7, l'utente non verrà nuovamente autenticato a meno che l'utente o la password nel campo SecurityParms della struttura MQCXP non siano stati modificati da un'uscita di sicurezza nel passo 9.

**Passo 11: adottare il contesto dell'utente MQCSP (se Ch1authEarlyAdopt=N e ADOPTCTX=YES)**

È possibile impostare l'attributo ADOPTCTX, che controlla se il canale viene eseguito in MCAUSER o l'ID utente fornito dall'applicazione.

Se l'ID utente asserito nel campo MQCSP o **SecurityParms** della struttura MQCXP è stato autenticato correttamente e **ADOPTCTX** è **YES**, il contesto dell'utente risultante dai passi 7 e 8 viene adottato come contesto da utilizzare per questa applicazione, a meno che l'utente o la password nel campo **SecurityParms** della struttura MQCXP non siano stati modificati da un'uscita di sicurezza nel passo 9.

Questo ID utente dichiarato è l'ID utente controllato per l'autorizzazione all'utilizzo delle risorse IBM MQ.

Ad esempio, non si dispone di MCAUSER impostato sul canale SVRCONN e il client è in esecuzione in 'johndoe' sulla macchina Linux. L'applicazione specifica l'utente 'fred' in MQCSP, quindi il canale inizia l'esecuzione con 'johndoe' come MCAUSER attivo. Dopo il controllo CONNAUTH, l'utente 'fred' viene adottato e il canale viene eseguito con 'fred' come MCAUSER attivo.

**Passo 12: Verifica che l'utente non sia bloccato (BLOCKUSER)**

Se il controllo **CONNAUTH** ha esito positivo, la cache CHLAUTH viene nuovamente ispezionata per controllare se MCAUSER attivo è bloccato da una regola BLOCKUSER. Se l'utente è bloccato, il canale termina.

**Step13: convalida i requisiti CHCKCLNT CHLAUTH**

Se la regola CHLAUTH che è stata selezionata nel passo 8 specifica anche un valore CHCKCLNT di REQUIRED o REQDADM, la convalida viene eseguita per garantire che sia stato fornito un ID utente CONNAUTH valido per soddisfare il requisito.

- Se CHCKCLNT (REQUIRED) è impostato, un utente deve essere stato autenticato nel passo 7 o 10. Altrimenti la connessione viene rifiutata.
- Se CHCKCLNT (REQDADM) è impostato, un utente deve essere stato autenticato nel passo 7 o 10 se si determina che questa connessione è privilegiata. Altrimenti la connessione viene rifiutata.
- Se è impostato CHCKCLNT (ASQMGR), questo passo viene ignorato.

**Note:**

1. Se CHCKCLNT (REQUIRED) o CHCKCLNT (REQDADM) è impostato, ma CONNAUTH non è abilitato sul gestore code, la connessione non riesce con un codice di ritorno MQRC\_SECURITY\_ERROR (2063) a causa del conflitto nella configurazione.
2. L'utente non viene riautenticato in questo passo.

**Passo 14: convalida dei requisiti CONNAUTH CHCKCLNT.**

La fase di autenticazione si verifica se CONNAUTH è abilitato sul gestore code.

Il valore CONNAUTH CHCKCLNT viene controllato per determinare quali requisiti sono impostati per le connessioni in entrata:

- Se CHCKCLNT (NONE) è impostato, questo passo viene ignorato
- Se CHCKCLNT (OPTIONAL) è impostato, questo passo viene ignorato.
- Se CHCKCLNT (REQUIRED) è impostato, un utente deve essere stato autenticato nel passo 7 o 10. Altrimenti la connessione viene rifiutata.
- Se CHCKCLNT (REQDADM) è impostato, un utente deve essere stato autenticato nel passo 7 o 10 se si determina che questa connessione è privilegiata. Altrimenti la connessione viene rifiutata.

**Nota:** L'utente non viene riautenticato in questo passo.

**Passo 15: controllo dell'autorizzazione dell'oggetto**

Viene eseguito un controllo per assicurarsi che il MCAUSER attivo disponga dell'autorizzazione appropriata per connettersi al gestore code.

ULW

Per ulteriori informazioni, consultare [Object Authority Manager](#).

IBM i

Per ulteriori informazioni, consultare [“Gestore autorizzazioni oggetto su IBM i”](#) a pagina 148.

**Passo 16: La connessione è stata completata**

Se i passi precedenti vengono completati correttamente, la connessione viene completata.

**Concetti correlati**

CONNAUTH

Un gestore code può essere configurato per utilizzare un ID utente e una password forniti per verificare se un utente dispone dell'autorizzazione per accedere alle risorse.

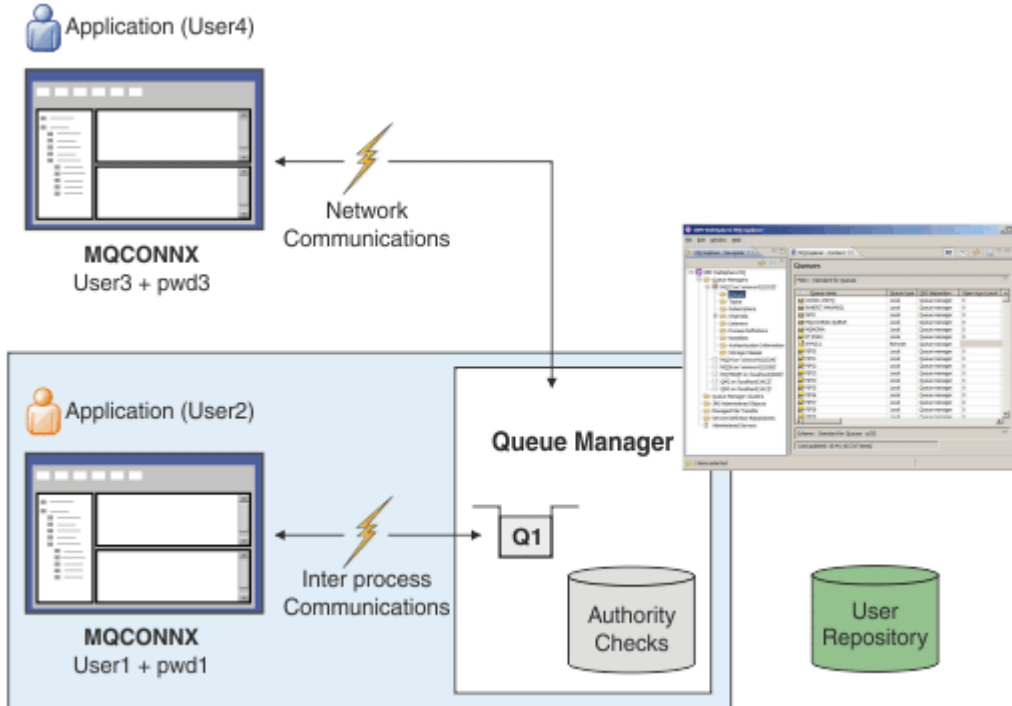
**Informazioni correlate**

SET CHLAUTH

[MODIFICA AUTHINFO](#)**Autenticazione connessione**

L'autenticazione della connessione può essere ottenuta in vari modi:

- Un'applicazione può fornire un ID utente e una password. L'applicazione può essere un client oppure può utilizzare i bind locali.
- Un gestore code può essere configurato per agire su un ID utente e password forniti.
- È possibile utilizzare un repository per stabilire se una combinazione di ID utente e password è valida.



Nel diagramma, due applicazioni stanno effettuando connessioni con un gestore code, un'applicazione come client e una utilizzando i bind locali. Le applicazioni possono utilizzare una varietà di API per connettersi al gestore code, ma tutte possono fornire un ID utente e una parola d'ordine. L'ID utente con cui è in esecuzione l'applicazione, User12 e User14 nel diagramma, che è il solito ID utente del sistema

operativo presentato a IBM MQ, potrebbe essere diverso dall'ID utente fornito dall'applicazione, User1 e User3.

Il gestore code riceve i comandi di configurazione (nel diagramma, IBM MQ Explorer è utilizzato) e gestisce l'apertura delle risorse e controlla l'autorità per accedere a tali risorse. Ci sono molte risorse differenti in IBM MQ a cui un'applicazione potrebbe richiedere l'autorità per accedere. Il diagramma illustra l'apertura di una coda per l'emissione, ma gli stessi principi si applicano anche ad altre risorse.

Consultare [Repository utente](#) per dettagli sul repository utilizzato per il controllo di ID utente e password.

### **Concetti correlati**

[“Autenticazione connessione: configurazione” a pagina 60](#)

Un gestore code può essere configurato per utilizzare un ID utente e una password forniti per verificare se un utente dispone dell'autorizzazione per accedere alle risorse.

[“Autenticazione della connessione: modifiche all'applicazione” a pagina 64](#)

[“Autenticazione connessione: repository utente” a pagina 65](#)

Per ciascun gestore code, è possibile scegliere diversi tipi di oggetti delle informazioni di autenticazione per l'autenticazione di ID utente e password.

### **Autenticazione connessione: configurazione**

Un gestore code può essere configurato per utilizzare un ID utente e una password forniti per verificare se un utente dispone dell'autorizzazione per accedere alle risorse.

### **Attivazione dell'autenticazione della connessione su un gestore code**

Su un oggetto gestore code, l'attributo **CONNAUTH** può essere impostato sul nome di un oggetto delle informazioni di autenticazione (AUTHINFO). Questo oggetto può essere di due tipi (attributo AUTHTYPE):

#### **IDPWOS**

Indica che il gestore code utilizza il sistema operativo locale per autenticare l'ID utente e password.

#### **IDPWLDP**

Indica che il gestore code utilizza il server LDAP per autenticare l'ID utente e password.

**Nota:** Non è possibile utilizzare nessun altro tipo di oggetto delle informazioni di autenticazione nel campo **CONNAUTH**.

IDPWOS e IDPWLDP sono simili in un certo numero di loro attributi, descritti qui. Altri attributi vengono considerati successivamente.

Per controllare le connessioni locali, utilizzare l'attributo AUTHINFO **CHCKLOCL** (controllare connessioni locali). Per verificare connessioni client, utilizzare l'attributo AUTHINFO **CHCKCLNT** (verifica connessioni client). La configurazione deve essere aggiornata prima che il gestore code riconosca le modifiche.

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

Dove USE.PW in CONNAUTH è una stringa che corrisponde alla definizione AUTHINFO.

Sia **CHCKLOCL** che **CHCKCLNT** hanno la stessa serie di valori possibili che consentono di variare la severità del controllo:

#### **nessuno**

Disattiva la verifica.

#### **Facoltativo**

Garantisce che se un ID utente e una password vengono forniti da un'applicazione, si tratta di una coppia valida, ma non è obbligatorio fornirli. Questa opzione potrebbe essere utile durante la migrazione, ad esempio.


**Importante:** FACOLTATIVO è il valore minimo che è possibile impostare per utilizzare regole CHLAUTH più rigorose.

Se si seleziona NONE e la connessione client corrisponde ad un record CHLAUTH con CHCKCLNT REQUIRED (o REQDADM su piattaforme diverse da z/OS), la connessione non riesce. Si riceve il messaggio AMQ9793 su piattaforme diverse da z/OS e il messaggio CSQX793E su z/OS.

### Obbligatorio

Richiede che tutte le applicazioni forniscano un ID utente e una password validi. Vedere anche la seguente nota.

### REQDADM

Gli utenti privilegiati devono fornire un ID e una password validi, ma gli utenti non privilegiati vengono trattati come con l'impostazione FACOLTATIVO . Vedere anche la seguente nota.  (Questa impostazione non è consentita su sistemi z/OS .

### Nota:

L'impostazione di **CHCKLOCL** su REQUIRED o REQDADM significa che non è possibile amministrare localmente il gestore code utilizzando **runmqsc** (errore AMQ8135: Non autorizzato) a meno che l'utente non specifichi il parametro `-u UserId` sulla riga di comando **runmqsc** . Con tale impostazione, **runmqsc** richiede la password dell'utente nella console.

In modo analogo, un utente che esegue IBM MQ Explorer sul sistema locale visualizzerà l'errore AMQ4036 quando tenta di connettersi al gestore code. Per specificare un nome utente e una parola d'ordine, fare clic con il tasto destro del mouse sull'oggetto gestore code locale e selezionare **Dettagli connessione > Proprietà ...** dal menu. Nella sezione **ID utente** , immettere il nome utente e la password da utilizzare e fare clic su **OK**.

Considerazioni simili si applicano alle connessioni remote con **CHCKCLNT**.

**CONNAUTH** è vuoto per i gestori code migrati ma è impostato su *SYSTEM.DEFAULT.AUTHINFO.IDPWOS* per nuovi gestori code. La definizione **AUTHINFO** precedente ha **CHCKCLNT** impostato su *REQDADM* per impostazione predefinita.

Pertanto, è necessario fornire la password del sistema operativo corretta per tutti i client esistenti che utilizzano un ID utente privilegiato per connettersi.

**Avviso:** In alcuni casi, la password in una struttura MQCSP per un'applicazione client verrà inviata attraverso una rete in testo semplice. Per assicurarsi che le password dell'applicazione del client siano protette in modo appropriato, consultare [“Protezione password MQCSP”](#) a pagina 30.

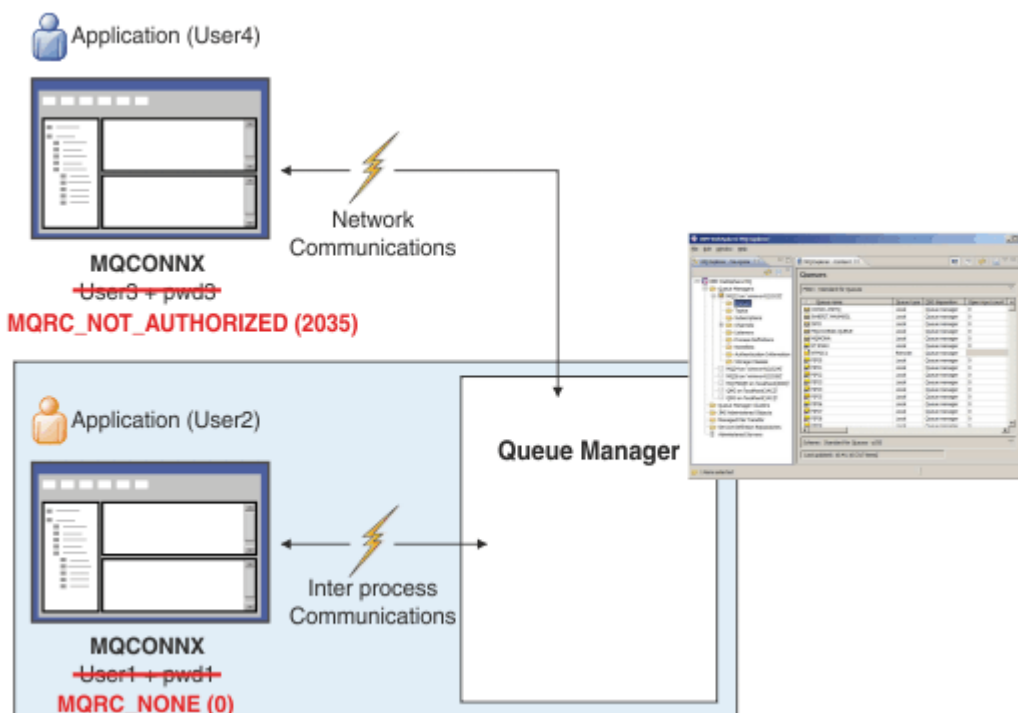
## Granularità della configurazione

Oltre a **CHCKLOCL** e **CHCKCLNT** che vengono utilizzati per attivare il controllo di ID utente e password, ci sono dei miglioramenti alle regole CHLAUTH in modo che sia possibile effettuare una configurazione più specifica utilizzando **CHCKCLNT**.

È possibile impostare il valore generale di **CHCKCLNT** su FACOLTATIVO, ad esempio, e quindi aggiornarlo in modo che sia più rigoroso per determinati canali impostando **CHCKCLNT** su OBBLIGATORIO o REQDADM sulla regola CHLAUTH . Per impostazione predefinita, le regole CHLAUTH verranno eseguite con CHCKCLNT (ASQMGR) , quindi non è necessario utilizzare questa granularità. Ad esempio:

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

## Notifica di errore



Viene registrato un errore se un'applicazione non fornisce un ID utente e una parola d'ordine quando richiesto o fornisce una combinazione non corretta anche quando è facoltativa.

**Nota:** Quando il controllo password è disattivato, utilizzando l'opzione NONE su **CHKCLOCL** o **CHKCLNT**, non vengono rilevate password non valide.

Le autenticazioni non riuscite vengono conservate per il numero di secondi specificato dall'attributo **FAILDLAY** prima che l'errore venga restituito all'applicazione. Ciò fornisce una protezione da un'applicazione che tenta ripetutamente di connettersi.

L'errore viene registrato in diversi modi:

### Applicazione

All'applicazione viene restituito l'errore di sicurezza IBM MQ standard, RC2035 - MQRC\_NOT\_AUTHORIZED.

### Amministratore

Un amministratore di IBM MQ visualizza l'evento riportato nel log degli errori e può quindi vedere che l'applicazione è stata rifiutata perché l'ID utente e password non hanno superato la verifica, piuttosto che perché, ad esempio, non vi era alcuna autorizzazione di connessione .

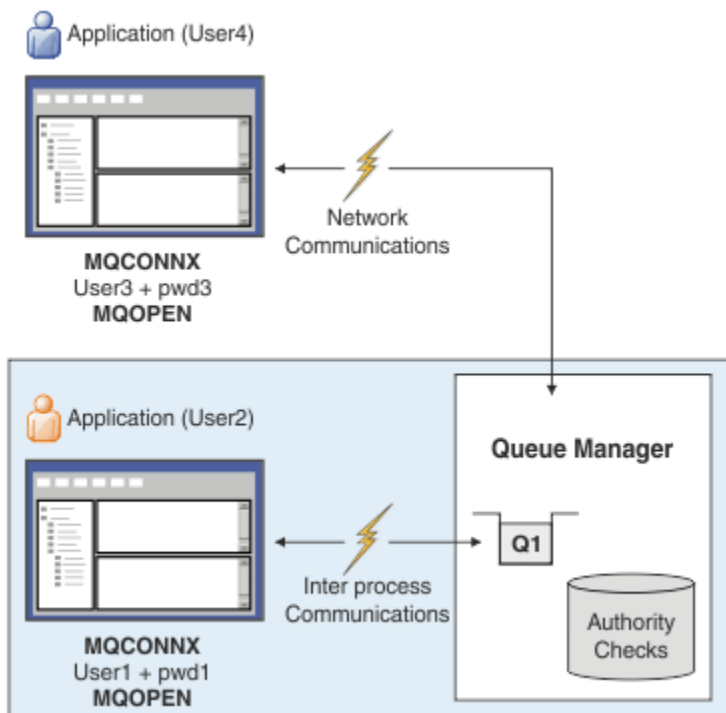
### Strumento di monitoraggio

Uno strumento di monitoraggio può anche essere notificato dell'errore, se si attivano gli eventi di autorizzazione inviando un messaggio di evento al SISTEMA SYSTEM.ADMIN.QMGR.EVENT :

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

Questo evento "Non autorizzato" è un evento di connessione di tipo 1 e fornisce gli stessi campi degli altri eventi di tipo 1, con un ulteriore campo, l'ID utente MQCSP fornito. La parola d'ordine non è fornita nel messaggio di eventi. Ciò significa che ci sono due ID utente nel messaggio di evento: l'ID con cui è in esecuzione l'applicazione e l'ID che l'applicazione ha presentato per il controllo dell'ID utente e della password.

## Relazione con l'autorizzazione



È possibile configurare un gestore code per richiedere che gli ID utente e le password vengano forniti da alcune applicazioni, poiché l'ID utente con cui l'applicazione è in esecuzione potrebbe non essere lo stesso ID utente che è stato presentato dall'applicazione insieme a una password quando l'applicazione apre una coda per l'output, ad esempio:

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

La modalità di gestione degli ID utente e delle password è controllata dall'attributo **ADOPTCTX** sull'oggetto delle informazioni di autenticazione.

### ADOPTCTX (Sì)

Tutti i controlli di autorizzazione per una applicazione vengono eseguiti con lo stesso ID utente autenticato dalla parola d'ordine, selezionando di adottare il contesto come contesto dell'applicazione per il resto della vita della connessione.



**Attenzione:** Quando si utilizzano ADOPTCTX (YES) e gli ID utente del sistema operativo, è necessario assicurarsi che l'ID utente adottato non superi la lunghezza massima degli ID utente. Per ulteriori informazioni, fare riferimento a [“ID utente”](#) a pagina 74.

### ADOPTCTX (NO)

Un'applicazione fornisce un ID utente e una password allo scopo di autenticarli al momento della connessione, ma poi continua utilizzando l'ID utente con cui è in esecuzione l'applicazione per futuri controlli di autorizzazione. Questa opzione potrebbe essere utile durante la migrazione o se si prevede di utilizzare altri meccanismi, come i record di autenticazione di canale, per assegnare l'identificativo utente dell'agent del canale dei messaggi (MCAUSER).



#### Attenzione:

Quando si utilizza il parametro **ADOPTCTX(YES)** su un oggetto delle informazioni di autenticazione, non è possibile adottare un altro contesto di sicurezza se non si imposta il parametro **Ch1authEarlyAdopt** nella stanza dei canali del file `qm.ini`.

Ad esempio, l'oggetto delle informazioni di autenticazione predefinito è impostato su **ADOPTCTX(YES)** e l'utente fred è collegato. Sono configurate le seguenti due regole CHLAUTH:

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

Il seguente comando viene emesso, con l'intenzione di autenticare il comando come contesto di sicurezza adottato dall'utente bob:

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

Infatti, il gestore code utilizza il contesto di sicurezza di fred, non bob, e la connessione non riesce.

Per ulteriori informazioni su **ChlauthEarlyAdopt**, consultare [Attributi della stanza dei canali](#).

### Concetti correlati

[“Autenticazione connessione” a pagina 59](#)

[“Autenticazione della connessione: modifiche all'applicazione” a pagina 64](#)

[“Autenticazione connessione: repository utente” a pagina 65](#)

Per ciascun gestore code, è possibile scegliere diversi tipi di oggetti delle informazioni di autenticazione per l'autenticazione di ID utente e password.

### ***Autenticazione della connessione: modifiche all'applicazione***

Un'applicazione può fornire un ID utente e una password all'interno della struttura MQCSP (connection security parameters) quando viene richiamato MQCONN. L'ID utente e la password vengono passati per il controllo al gestore autorizzazioni oggetto (OAM) fornito con il gestore code o al componente del servizio di autorizzazione fornito con il gestore code su sistemi z/OS. Non è necessario scrivere l'interfaccia personalizzata.

Se l'applicazione è in esecuzione come client, l'ID utente e la password vengono inoltrati anche alle uscite di sicurezza lato client e lato server per l'elaborazione. Possono essere utilizzati anche per impostare l'attributo MCAUSER (message channel agent user identifier) di una istanza del canale. L'uscita di sicurezza viene richiamata con motivo dell'uscita MQXR\_SEC\_PARMS per questa elaborazione. Le uscite di sicurezza lato client e l'uscita di preconnessione possono apportare modifiche a MQCONN prima che venga inviata al gestore code.

**Avviso:** In alcuni casi, la password in una struttura MQCSP per un'applicazione client verrà inviata attraverso una rete in testo semplice. Per assicurarsi che le password dell'applicazione del client siano protette in modo appropriato, consultare [“Protezione password MQCSP” a pagina 30](#).

Utilizzando la stringa XAOPEN per fornire ID utente e password, è possibile evitare di dover apportare modifiche al codice dell'applicazione.

### **Nota:**

Da IBM WebSphere MQ 6.0, l'uscita di sicurezza ha consentito l'impostazione di MQCSP. Pertanto, i client di questo livello o successivi non devono essere aggiornati.

Tuttavia, nelle versioni precedenti a IBM MQ 8.0, MQCSP non poneva alcuna limitazione all'ID utente e alla password forniti dall'applicazione. Quando si utilizzano questi valori con le funzioni fornite da IBM MQ ci sono dei limiti che si applicano all'utilizzo di queste funzioni, ma se si stanno solo passando alle proprie uscite, tali limiti non si applicano.

### Concetti correlati

[“Autenticazione connessione” a pagina 59](#)

[“Autenticazione connessione: configurazione” a pagina 60](#)

Un gestore code può essere configurato per utilizzare un ID utente e una password forniti per verificare se un utente dispone dell'autorizzazione per accedere alle risorse.



“Autenticazione connessione: repository utente” a pagina 65

Per ciascun gestore code, è possibile scegliere diversi tipi di oggetti delle informazioni di autenticazione per l'autenticazione di ID utente e password.

### **Autenticazione connessione: repository utente**

Per ciascun gestore code, è possibile scegliere diversi tipi di oggetti delle informazioni di autenticazione per l'autenticazione di ID utente e password.

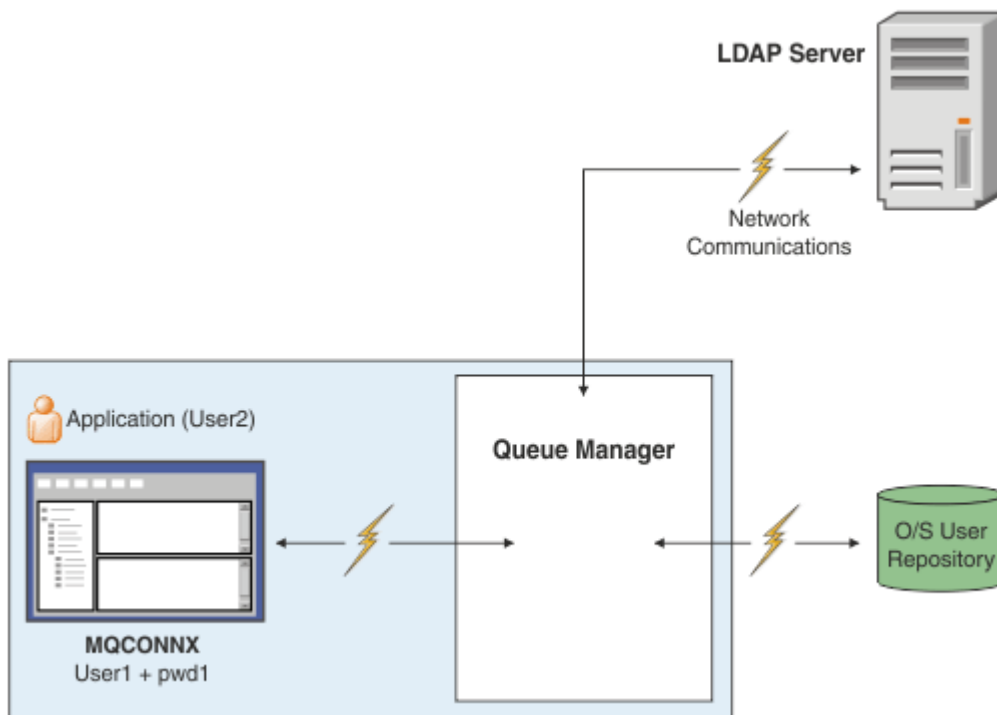


Figura 7. Tipi di oggetti delle informazioni di autenticazione

```
DEFINE AUTHINFO(USE.OS) AUTHTYPE(IDPWOS)
DEFINE AUTHINFO(USE.LDAP) +
AUTHTYPE(IDPWLDAP) +
CONNNAME('ldap1(389),ldap2(389)') +
LDAPUSER('CN=QMGR1') +
LDAPPWD('passwd1d') SECCOMM(YES)
```

Esistono due tipi di oggetti delle informazioni di autenticazione, rappresentati nel diagramma:

- IDPWOS viene utilizzato per indicare che il gestore code utilizza il sistema operativo locale per autenticare l'ID utente e password. Se si sceglie di utilizzare il sistema operativo locale, è necessario impostare gli attributi comuni, come descritto negli argomenti precedenti.
- IDPWLDAP viene utilizzato per specificare che il gestore code utilizza il server LDAP per autenticare l'ID utente e password. Se si sceglie di utilizzare un server LDAP, in questo argomento vengono fornite ulteriori informazioni.

È possibile scegliere un solo tipo di oggetto delle informazioni di autenticazione per ciascun gestore code da utilizzare, denominando l'oggetto appropriato nell'attributo **CONNAUTH** del gestore code.

### **Utilizzo di un server LDAP per autenticazione.**

Impostare il campo **CONNNAME** sull'indirizzo del server LDAP per il gestore code. È possibile fornire più indirizzi per il server LDAP in un elenco separato da virgole, che può aiutare con la ridondanza se il server LDAP non fornisce questa funzione.

Impostare l'ID e la password del server LDAP richiesti nei campi **LDAPUSER** e **LDAPPWD** in modo che il gestore code possa accedere al server LDAP e ricercare le informazioni sui record utente.

## Connessione sicura a un server LDAP

A differenza dei canali, non esiste alcun parametro **SSLCPH** per attivare l'utilizzo di TLS per la comunicazione con il server LDAP. In questo caso, IBM MQ agisce come un client per il server LDAP, quindi gran parte della configurazione viene effettuata sul server LDAP. Alcuni parametri esistenti in IBM MQ vengono utilizzati per configurare il funzionamento di tale connessione.

Impostare il campo **SECCOMM** per controllare se la connettività al server LDAP utilizza TLS.

Oltre a questo attributo, gli attributi del gestore code **SSLFIPS** e **SUITEB** limitano la serie di specifiche di cifratura scelte. Il certificato utilizzato per identificare il gestore code per il server LDAP è il certificato del gestore code, `ibmwebsphere.mq.qmgr-name` o il valore dell'attributo **CERTLABL**. Per i dettagli, consultare [Etichetta certificato digitale](#).

## Repository utenti LDAP

Quando si utilizza un repository utente LDAP, è necessario eseguire ulteriori operazioni di configurazione sul gestore code, oltre a indicare al gestore code dove trovare il server LDAP.

Gli ID definiti in un server LDAP hanno una struttura gerarchica che li identifica in modo univoco. Pertanto, un'applicazione può connettersi al gestore code e presentare il proprio ID utente come ID utente gerarchico completo.

Tuttavia, per semplificare le informazioni che un'applicazione deve fornire, è possibile configurare il gestore code in modo da presumere che la prima parte della gerarchia sia comune a tutti gli ID e aggiungerla automaticamente prima dell'ID abbreviato fornito dall'applicazione. Il gestore code può quindi presentare un ID completo al server LDAP.

Impostare **BASEDNU** sul punto iniziale in cui la ricerca LDAP ricerca l'ID nella gerarchia LDAP. Quando si imposta **BASEDNU**, è necessario assicurarsi che venga restituito un solo risultato quando si cerca l'ID nella gerarchia LDAP.

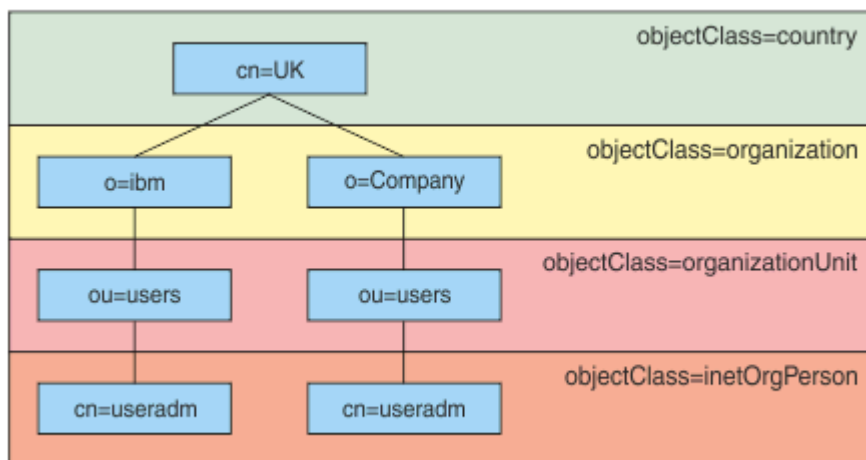


Figura 8. Una gerarchia LDAP di esempio

Ad esempio, in Figura 8 a pagina 66 **BASEDNU** può essere impostato su "ou=users, o=ibm, c = UK" o "o=ibm, c = UK". Tuttavia, poiché un DN che contiene "cn = useradm" esiste sia nel ramo "o = ibm" che nel ramo "o=Company", **BASEDNU** non può essere impostato su "c = UK". Per motivi di prestazioni e sicurezza, utilizzare il punto più alto nella gerarchia LDAP da cui è possibile fare riferimento a tutti gli ID utente necessari. In questo esempio, è "ou=users, o=ibm, c = UK".

L'applicazione potrebbe inoltrare al gestore code l'ID utente senza fornire il nome attributo LDAP, ad esempio CN= . Se si imposta **USRFIELD** sul nome attributo LDAP, questo valore viene aggiunto come

prefisso all'ID utente che proviene dall'applicazione. Questo potrebbe essere un aiuto migratorio utile quando si passa dagli ID utente del sistema operativo agli ID utente LDAP, poiché l'applicazione può presentare la stessa stringa in entrambi i casi ed è possibile evitare di modificare l'applicazione.

Pertanto, l'ID utente completo presentato al server LDAP è simile al seguente:

```
USRFIELD = ID_from_application BASEDNU
```

## Concetti correlati

[“Autenticazione connessione” a pagina 59](#)

[“Autenticazione connessione: configurazione” a pagina 60](#)

Un gestore code può essere configurato per utilizzare un ID utente e una password forniti per verificare se un utente dispone dell'autorizzazione per accedere alle risorse.

[“Autenticazione della connessione: modifiche all'applicazione” a pagina 64](#)

## Uscita di sicurezza lato client per inserire ID utente e password (mqccred)

Se si dispone di applicazioni client che sono richieste per inviare un ID utente o una parola d'ordine ma non è ancora possibile modificare l'origine, esiste un'uscita di sicurezza fornita con IBM MQ 8.0 denominata **mqccred** che è possibile utilizzare. **mqccred** fornisce un ID utente e una password per conto dell'applicazione client, da un file `.ini`. Questo ID utente e questa password vengono inviati al gestore code che, se configurato per eseguire tale operazione, li autenticherà.

## Panoramica

**mqccred** è un'uscita di sicurezza che viene eseguita sulla stessa macchina dell'applicazione client. Consente di fornire informazioni su ID utente e password per conto dell'applicazione client, laddove tali informazioni non vengono fornite dall'applicazione stessa. Le informazioni sull'ID utente e sulla password vengono fornite in una struttura nota come [Parametri di sicurezza della connessione \(MQCSP\)](#) e verranno autenticate dal gestore code se è configurata l'[autenticazione della connessione](#).

Le informazioni su ID utente e password vengono richiamate da un file `.ini` sulla macchina client. Le password nel file sono protette da offuscamento utilizzando il comando **runmqccred** e anche garantendo che le autorizzazioni file sul file `.ini` siano impostate in modo che solo l'ID utente che esegue l'applicazione client (e quindi l'uscita) siano in grado di leggerlo.

## Ubicazione

**mqccred** è installato:

### Windows piattaforma

Nella directory `installation_directory\Tools\c\Samples\mqccred\`

### UNIX piattaforma

Nella directory `installation_directory/samp/mqccred`

**Note:** L'uscita:

1. Agisce esclusivamente come un'uscita del canale di sicurezza e deve essere l'unica uscita di questo genere definita su un canale.
2. Generalmente viene denominato tramite la CCDT (Client Channel Definition Table), ma un client Java può avere l'uscita menzionata direttamente negli oggetti JNDI oppure l'exit potrebbe essere configurata per le applicazioni che creano manualmente la struttura [MQCD](#).
3. È necessario copiare i programmi **mqccred** e **mqccred\_r** nella directory `var/mqm/exits`.

Ad esempio, su una macchina della piattaforma UNIX a 64 bit, immettere il comando:

```
cp installation_directory/samp/mqccred/lib64/* /var/mqm/exits
```

Per ulteriori informazioni, consultare [Un esempio passo per passo di come verificare mqccred](#).

4. È in grado di essere eseguito su versioni precedenti di IBM MQ ; fino a IBM WebSphere MQ 7.0.1.

## Impostazione di ID utente e password

Il file `.ini` contiene le stanze per ogni gestore code, con un'impostazione globale per i gestori code non specificati. Ogni stanza contiene il nome del gestore code, un ID utente e una password in testo semplice o offuscata.

È necessario modificare il file `.ini` manualmente, utilizzando l'editor desiderato e aggiungere l'attributo `password` in testo semplice alle stanze. Eseguire il programma `runmqccred` fornito, che prende il file `.ini` e sostituisce l'attributo **Password** con l'attributo **OPW**, un formato offuscato della password.

Consultare [runmqccred](#) per una descrizione del comando e dei relativi parametri.

Il file `mqccred.ini` contiene le informazioni relative all'ID utente e alla password.

Un file modello `.ini` viene fornito nella stessa directory dell'uscita per fornire un punto iniziale per la propria azienda.

Per impostazione predefinita, questo file verrà ricercato in `$HOME/.mq/mqccred.ini`. Se si desidera individuarlo altrove, è possibile utilizzare la variabile di ambiente `MQCCRED` per puntare ad esso:

```
MQCCRED=C:\mydir\mqccred.ini
```

Se si utilizza `MQCCRED`, la variabile deve includere il nome completo del file di configurazione, incluso qualsiasi tipo di file `.ini`. Poiché questo file contiene password (anche se offuscate), si prevede di proteggere il file utilizzando i privilegi del sistema operativo per garantire che le persone non autorizzate non possano leggerlo. Se non si dispone dell'autorizzazione file corretta, l'uscita non verrà eseguita correttamente.

Se l'applicazione ha già fornito una struttura `MQCSP`, l'uscita normalmente lo rispetta e non inserirà alcuna informazione dal file `.ini`. Tuttavia, è possibile sovrascriverlo utilizzando l'attributo **Force** nella sezione.

L'impostazione di **Force** sul valore `TRUE` rimuove l'ID utente e la password forniti dall'applicazione e sostituisce quelli con la versione del file ini.

È anche possibile impostare l'attributo **Force** nella sezione globale del file per impostare il valore predefinito di tale file.

Il valore predefinito per **Force** è `FALSE`.

È possibile fornire un ID utente e una password per tutti i gestori code o per ogni singolo gestore code. Questo è un esempio di file `mqccred.ini`:

```
# comments are permitted
AllQueueManagers:
User=abc
OPW=%^&aerivrgtsr

QueueManager:
Name=QMA
User=user1
OPW=H&^dbgfh

Force=TRUE

QueueManager:
Name=QMB
User=user2
password=passw0rd
```

### Note:

1. Le singole definizioni del gestore code hanno la precedenza sull'impostazione globale.
2. Gli attributi non sono sensibili al maiuscolo / minuscolo.

## Vincoli

Quando questa uscita è in uso, l'ID utente locale della persona che esegue l'applicazione non passa dal client al server. Le uniche informazioni di identità disponibili provengono dal contenuto del file ini.

Pertanto, è necessario configurare il gestore code per utilizzare **ADOPTCTX(YES)** o associare la richiesta di connessione in entrata a un ID utente appropriato tramite uno dei meccanismi disponibili, ad esempio [“Record di autenticazione di canale” a pagina 47](#).

**Importante:** Se si aggiungono nuove password o si aggiornano quelle vecchie, il comando **runmqccred** elabora solo le password in testo semplice, lasciando inalterate quelle offuscate.

## Debug

L'uscita scrive sulla traccia IBM MQ standard quando è abilitata.

Per facilitare il debug dei problemi di configurazione, l'uscita può anche scrivere direttamente in stdout.

Nessun dato di uscita di sicurezza del canale ( **SCYDATA** ) La configurazione è normalmente richiesta per il canale. Tuttavia, è possibile specificare:

### ERRORE

Stampare solo le informazioni sulle condizioni di errore, come ad esempio non essere in grado di trovare il file di configurazione.

### DEBUG

Visualizza queste condizioni di errore e alcune ulteriori istruzioni di traccia.

### NOCHECK

Ignora i vincoli sulle autorizzazioni file e l'ulteriore vincolo in base al quale il file .ini non deve contenere password non protette.

È possibile inserire uno o più di questi elementi nel campo **SCYDATA** , separati da virgole, in qualsiasi ordine. Ad esempio, SCYDATA=(NOCHECKS, DEBUG).

Notare che gli elementi sono sensibili al maiuscolo / minuscolo e devono essere immessi in maiuscolo.

## Utilizzo di mqccred

Una volta impostato il file, è possibile richiamare l'exit del canale aggiornando la definizione del canale di connessione client per includere l'attributo SCYEXIT( 'mqccred(ChlExit) ' ) :

```
DEFINE CHANNEL(channelname) CHLTYPE(cIntconn) +  
CONNAME(remote machine) +  
QMNAME(remote qmgr) +  
SCYEXIT('mqccred(ChlExit)') +  
REPLACE
```

### Informazioni correlate

[SCYDATA](#)

[SCYEXIT](#)

[runmqccred](#)

### **Autenticazione della connessione con il client Java**

L'autenticazione della connessione è una funzione in IBM MQ che consente al gestore code di essere configurato per autenticare le applicazioni, utilizzando un ID utente e password forniti. Quando l'applicazione è un'applicazione Java che utilizza i bind del client, l'autenticazione della connessione può essere eseguita in modalità di compatibilità o in modalità di autenticazione MQCSP.

### Modalità di compatibilità

Prima di IBM MQ 8.0, il client Java poteva inviare un ID utente e una password attraverso il canale di connessione client al canale di connessione server e farli fornire a un'uscita di sicurezza nei campi

**RemoteUserIdentifier** e **RemotePassword** della struttura MQCD. In modalità di compatibilità, questo comportamento viene mantenuto.

È possibile utilizzare questa modalità in combinazione con l'autenticazione della connessione e migrare da qualsiasi uscita di sicurezza precedentemente utilizzata per eseguire lo stesso lavoro.

È necessario utilizzare ADOPTCTX (YES) o un altro metodo, ad esempio una regola CHLAUTH basata su un certificato TLS, per impostare MCAUSER in esecuzione quando si utilizza la modalità di compatibilità, poiché in questa modalità l'ID utente lato client non viene inviato al gestore code.

La modalità di compatibilità del funzionamento può essere abilitata su una base di connessione per connessione o globalmente:

- In IBM MQ classes for Java, impostare la proprietà `MQConstants.USE_MQCSP_AUTHENTICATION_PROPERTY` su `false` nella hashtable delle proprietà passata al costruttore `com.ibm.mq.MQQueueManager`.
- In IBM MQ classes for JMS, impostare la proprietà `JmsConstants.USER_AUTHENTICATION_MQCSP` su `false`, sul factory di connessione appropriato prima di creare la connessione.
- A livello globale, specificare la Java proprietà di sistema `-Dcom.ibm.mq.cfg.jmqi.useMQCSPauthentication=false` sulla riga comandi quando si avvia l'applicazione, come mostrato nel seguente esempio:

```
java -Dcom.ibm.mq.cfg.jmqi.useMQCSPauthentication=false application_name
```

La modalità di compatibilità è l'impostazione predefinita.

## Modalità di autenticazione MQCSP

In questa modalità, vengono inviati l'ID utente lato client e l'ID utente e password da autenticare, in modo da poter utilizzare ADOPTCTX (NO). L'ID utente e password sono disponibili per un'uscita di sicurezza della connessione server nella struttura `MQCSP` fornita nella struttura `MQCXP`.

Questa modalità di funzionamento può essere abilitata su una base di connessione per connessione o globalmente:

- In IBM MQ classes for Java, impostare la proprietà `MQConstants.USE_MQCSP_AUTHENTICATION_PROPERTY` su `true` nella hashtable delle proprietà passata al costruttore `com.ibm.mq.MQQueueManager`.
- In IBM MQ classes for JMS, impostare la proprietà `JmsConstants.USER_AUTHENTICATION_MQCSP` su `true`, sul factory di connessione appropriato prima di creare la connessione.
- Globalmente, impostare la proprietà di sistema `com.ibm.mq.cfg.jmqi.useMQCSPauthentication` su un valore che indica true, ad esempio, aggiungendo `-Dcom.ibm.mq.cfg.jmqi.useMQCSPauthentication=Y` alla riga comandi.

## Scelta della modalità di autenticazione in IBM MQ Explorer

IBM MQ Explorer è un'applicazione Java, quindi anche queste due modalità, la modalità di compatibilità e la modalità di autenticazione MQCSP, sono applicabili.

**V 9.0.4** Da IBM MQ 9.0.4, la modalità di autenticazione MQCSP è quella predefinita. Prima di IBM MQ 9.0.4, la modalità di compatibilità è quella predefinita.

Nei pannelli in cui viene fornita l'identificazione utente, è disponibile una casella di spunta per abilitare o disabilitare la modalità di compatibilità:

- **V 9.0.4** Da IBM MQ 9.0.4, per impostazione predefinita, questa casella di spunta non è selezionata. Per utilizzare la modalità di compatibilità, selezionare questa casella di controllo.
- Prima di IBM MQ 9.0.4, per impostazione predefinita, questa casella di spunta è abilitata. Per utilizzare l'autenticazione MQCSP, deselezionare la check box.

## Concetti correlati

[“Autenticazione connessione” a pagina 59](#)

[“Autenticazione della connessione: modifiche all'applicazione” a pagina 64](#)

[“Autenticazione connessione: repository utente” a pagina 65](#)

Per ciascun gestore code, è possibile scegliere diversi tipi di oggetti delle informazioni di autenticazione per l'autenticazione di ID utente e password.

## Sicurezza dei messaggi in IBM MQ

La sicurezza dei messaggi nell'infrastruttura IBM MQ viene fornita da un Advanced Message Security.

Advanced Message Security (AMS) espande i servizi di sicurezza IBM MQ 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, AMS verifica che un mittente dei dati del messaggio sia autorizzato a inserire i messaggi firmati su una coda di destinazione.

## Concetti correlati

[“Advanced Message Security” a pagina 520](#)

Advanced Message Security (AMS) è un componente di IBM MQ che fornisce un livello elevato di protezione per i dati sensibili che passano attraverso la rete IBM MQ, 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 MQ.

È possibile utilizzare IBM 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.

IBM MQ fornisce una gamma di servizi di sicurezza a livello di link, incluso il supporto per TLS (Transport Layer Security).

È necessario considerare alcuni aspetti della protezione quando si pianifica di installare IBM MQ:

- ▶ **Multi** Su [Multiplatforme](#), se si ignorano questi aspetti e non si fa nulla, non è possibile utilizzare IBM MQ.
- ▶ **z/OS** Su z/OS, l'effetto di ignorare questi aspetti è che le risorse IBM MQ non sono protette. Ovvero, tutti gli utenti possono accedere e modificare tutte le risorse IBM MQ.



## Autorizzazione per gestire IBM MQ

Gli amministratori IBM MQ hanno bisogno dell'autorità per:

- Immettere i comandi per gestire IBM MQ
- Utilizzare IBM MQ Explorer
- ▶ **IBM i** Utilizzare i comandi e i pannelli di gestione IBM i.
- ▶ **z/OS** Utilizzare le operazioni e i pannelli di controllo su z/OS
- ▶ **z/OS** Utilizzare il programma di utilità IBM MQ, CSQUTIL, su z/OS
- ▶ **z/OS** Accedere ai dataset del gestore code su z/OS

Per ulteriori informazioni, vedere:

- [“Autorizzazione per gestire IBM MQ su UNIX, Linux, and Windows” a pagina 391](#)

-  [“Autorizzazione per gestire IBM MQ su IBM i” a pagina 76](#)
-  [“Autorizzazione per gestire IBM MQ su z/OS” a pagina 77](#)

## Autorizzazione per gestire gli oggetti IBM MQ

Le applicazioni possono accedere ai seguenti oggetti IBM 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 IBM MQ e per accedere ai canali e agli oggetti delle informazioni di autenticazione. Questi oggetti possono essere protetti da IBM MQ , in modo che gli ID utente associati alle applicazioni abbiano l'autorizzazione per accedervi.

Per ulteriori informazioni, vedere [“Autorizzazione per le applicazioni ad utilizzare IBM MQ” a pagina 79.](#)

## Sicurezza canale

Gli ID utente associati agli agent MCA (message channel agent) necessitano dell'autorità per accedere a diverse risorse IBM 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 100.](#)

## Ulteriori considerazioni

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

- [“Sicurezza per i cluster del gestore code” a pagina 113](#)
- [“Sicurezza per la pubblicazione / sottoscrizione IBM MQ” a pagina 114](#)
- [“Sicurezza per IBM MQ Internet pass - thru” a pagina 115](#)

## 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 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 MQ che utilizzano 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 MQ. Per ulteriori informazioni, consultare:

- [“Associazione di un gestore code remoto a un ID utente MCAUSER” a pagina 376](#)
- [“Associazione di un ID utente client a un ID utente MCAUSER” a pagina 376](#)
- [“Associazione di un DN \(Distinguished Name\) SSL o TLS a un ID utente MCAUSER” a pagina 377](#)



- [“Associazione di un indirizzo IP a un ID utente MCAUSER” a pagina 379](#)

## 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 IBM MQ MQI client connesso a un server. La sicurezza viene applicata a quattro livelli, come descritto nel seguente testo. MCA è un agente del canale dei messaggi.

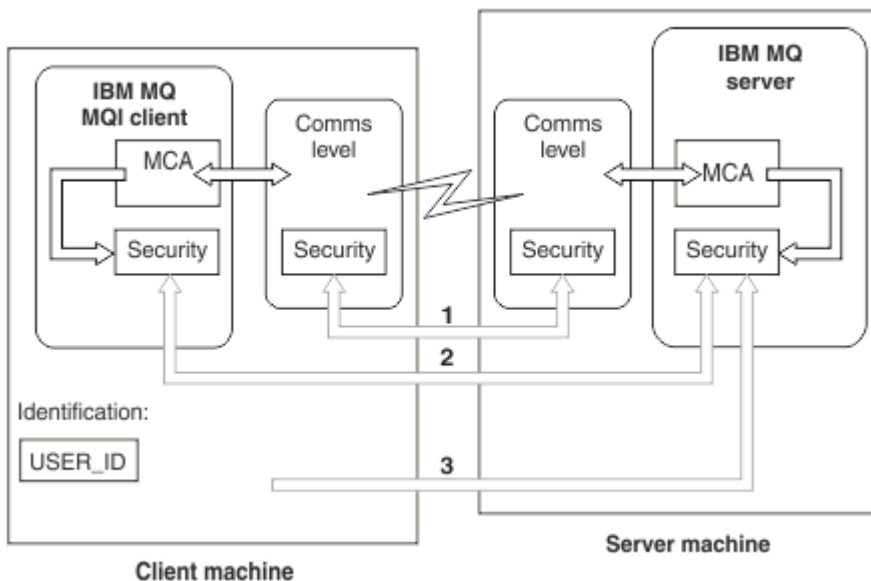


Figura 9. Sicurezza in una connessione client/server

### 1. Livello di comunicazione

Vedere la freccia 1. Per implementare la sicurezza a livello di comunicazioni, utilizzare TLS. Per ulteriori informazioni, vedi [“Protocolli di sicurezza crittografici: TLS” a pagina 15](#)

### 2. Record di autenticazione di canale

Vedere frecce 2 & 3. L'autenticazione può essere controllata utilizzando l'indirizzo IP o i DN (distinguished name) TLS a 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 47](#).

### 3. Autenticazione connessione

Vedere la freccia 3. Il client invia un ID e una password. Per ulteriori informazioni, vedere [“Autenticazione connessione: configurazione” a pagina 60](#).

### 4. 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](#).

### 5. 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 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.






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

Da IBM MQ 8.0, è possibile inviare le password incluse nella struttura MQCSP.

**Avviso:** In alcuni casi, la password in una struttura MQCSP per un'applicazione client verrà inviata attraverso una rete in testo semplice. Per assicurarsi che le password dell'applicazione del client siano protette in modo appropriato, consultare [“Protezione password MQCSP”](#) a pagina 30.

## ID utente

Quando si creano ID utente per le applicazioni client, gli ID utente non devono superare la lunghezza massima consentita. Non utilizzare gli ID utente riservati UNKNOWN e NOBODY. Se il server a cui si connette il client è un server IBM MQ for Windows, è necessario ignorare l'utilizzo del simbolo chiocciola @. La lunghezza consentita degli ID utente dipende dalla piattaforma utilizzata per il server:

-    Su z/OS e UNIX and Linux, la lunghezza massima di un ID utente è 12 caratteri.
-  Su IBM i, la lunghezza massima di un ID utente è 10 caratteri.
-  Su Windows, se sia il server IBM MQ MQI client che il server IBM MQ sono su Windows e il server ha accesso al dominio su cui è definito l'ID utente client, la lunghezza massima di un ID utente è di 20 caratteri. Tuttavia, se il server IBM MQ non è un server Windows, l'ID utente viene troncato a 12 caratteri.
- Se si utilizza la struttura MQCSP per passare le credenziali, la lunghezza massima di un ID utente è di 1024 caratteri. L'ID utente della struttura MQCSP non può essere utilizzato per aggirare la lunghezza massima dell'id utente utilizzata da IBM MQ per l'autorizzazione. Per ulteriori informazioni sulla struttura MQCSP, consultare [“Identificazione e autenticazione degli utenti utilizzando la struttura MQCSP”](#) a pagina 325.

Su sistemi UNIX and Linux, l'impostazione predefinita è che gli ID utente vengono utilizzati per l'autenticazione e i gruppi vengono utilizzati per l'autorizzazione. Tuttavia, è possibile configurare questi sistemi per autorizzare rispetto agli ID utente. Per ulteriori informazioni, consultare [“Autorizzazioni basate sull'utente OAM su UNIX and Linux”](#) a pagina 341. I sistemi Windows possono utilizzare sia ID utente per l'autenticazione e l'autorizzazione che gruppi per l'autorizzazione.

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.

## ID utente limitati

Gli ID utente UNKNOWN e il gruppo NOBODY hanno significati speciali per IBM MQ. La creazione di un ID utente nel sistema operativo denominato UNKNOWN o di un gruppo denominato NOBODY potrebbe avere risultati indesiderati.

## ID utente durante la connessione a un server IBM MQ for Windows



un server IBM MQ for Windows non supporta la connessione di un client Windows se il client è in esecuzione con un ID utente che contiene il carattere @, ad esempio, abc@d. Il codice di ritorno per la chiamata MQCONN al 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.

## Autorizzazione di pianificazione

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

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

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

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

Gli amministratori IBM MQ hanno bisogno dell'autorità per:

- Immettere i comandi per gestire IBM MQ.
-   Utilizza la IBM MQ Explorer.
-  Utilizzare le operazioni e i pannelli di controllo su z/OS.
-  Utilizzare il programma di utilità IBM MQ , CSQUTIL, su z/OS.
-  Accedere ai dataset del gestore code su z/OS.

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

### Autorizzazione per gestire IBM MQ su UNIX e sistemi Windows

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

Per essere un amministratore IBM MQ su sistemi UNIX e Windows , un utente deve essere un membro del *gruppo mqm*. Questo gruppo viene creato automaticamente quando si installa IBM MQ. Per consentire agli utenti di immettere comandi di controllo, è necessario aggiungerli al gruppo `mqm`. Ciò include l'utente `root` su 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 MQ e sono autorizzati ad eseguire solo i comandi per i quali è stato loro concesso l'accesso.


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

Tutti i membri del gruppo `mqm` hanno accesso a tutte le risorse IBM 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`. Sui sistemi Windows , i membri del gruppo `Administrators` hanno anche accesso a tutte le risorse IBM MQ .

Gli amministratori possono utilizzare il comando di controllo **runmqsc** per immettere comandi MQSC ( IBM MQ Script). 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.

IBM MQ Explorer emette comandi PCF per eseguire attività di gestione. Gli amministratori non richiedono ulteriori autorizzazioni per utilizzare IBM MQ Explorer per gestire un gestore code sul sistema locale. Quando IBM 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, vedere [“Autorizzazione per le applicazioni ad utilizzare IBM MQ”](#) a pagina 79.
- Per i comandi che operano su canali, iniziatori di canali, listener e cluster, consultare [Sicurezza canale](#).
-  Per i comandi MQSC elaborati dal server dei comandi su IBM MQ for z/OS, consultare [“Sicurezza dei comandi e sicurezza delle risorse dei comandi”](#) a pagina 77.

Per ulteriori informazioni sull'autorità necessaria per amministrare IBM MQ su sistemi UNIX e Windows, consultare le informazioni correlate.

## **Autorizzazione per gestire IBM MQ su IBM i**

Per essere un amministratore IBM MQ su IBM i, è necessario essere un membro del *Gruppo QMQMADM*. Questo gruppo ha proprietà simili a quelle del gruppo mqm sui sistemi UNIX e Windows. In particolare, il gruppo QMQMADM viene creato quando si installa IBM MQ for IBM i e i membri del gruppo QMQMADM hanno accesso alle risorse IBM MQ sul sistema. Inoltre, si ha accesso a tutte le risorse IBM MQ se si dispone dell'autorizzazione \*ALLOBJ.

Gli amministratori possono utilizzare i comandi CL per gestire IBM MQ. Uno di questi comandi è GRTMQMAUT, utilizzato per concedere le autorizzazioni ad altri utenti. Un altro comando, STRMQMMQSC, consente a un amministratore di emettere comandi MQSC per un gestore code locale.

Esistono due gruppi di comandi CL forniti da IBM MQ for IBM i:

### **Gruppo 1**

Per emettere un comando in questa categoria, un utente deve essere un membro del gruppo QMQMADM o disporre dell'autorizzazione \*ALLOBJ. GRTMQMAUT e STRMQMMQSC appartengono a questa categoria, ad esempio.

### **Gruppo 2**

Per immettere un comando in questa categoria, non è necessario che un utente sia membro del gruppo QMQMADM o che disponga dell'autorizzazione \*ALLOBJ. Sono invece richiesti due livelli di autorizzazione:

- L'utente richiede l'autorità IBM i per utilizzare il comando. Questa autorizzazione viene concessa utilizzando il comando GRTOBJAUT.
- L'utente richiede l'autorizzazione IBM MQ per accedere a qualsiasi oggetto IBM MQ associato al comando. Questa autorizzazione viene concessa utilizzando il comando GRTMQMAUT.

I seguenti esempi mostrano i comandi in questo gruppo:

- CRTMQMQ, Crea coda MQM
- CHGMQMPRC, Modifica processo MQM
- DLTMQMNL, Elimina elenco nomi MQM
- DSPMQMAUTI, Visualizzazione informazioni di autenticazione MQM
- CRTMQMCHL, Crea canale MQM

Per ulteriori informazioni su questo gruppo di comandi, consultare [“Autorizzazione per le applicazioni ad utilizzare IBM MQ”](#) a pagina 79.

Per un elenco completo dei comandi gruppo 1 e gruppo 2, consultare [“Autorizzazioni di accesso per oggetti IBM MQ su IBM i”](#) a pagina 149

Per ulteriori informazioni relative all'autorizzazione necessaria per amministrare IBM MQ su IBM i, consultare [Amministrazione IBM i](#).

## **Autorizzazione per gestire IBM MQ su z/OS**

Questa raccolta di argomenti descrive vari aspetti dell'autorizzazione necessaria per amministrare IBM MQ for z/OS.

### **z/OS** Controlli autorizzazione su z/OS

IBM MQ for z/OS utilizza SAF (System Authorization Facility) per instradare richieste di controlli di autorizzazione a un ESM (External Security Manager) come z/OS Security Server Resource Access Control Facility (RACF). IBM MQ non esegue controlli di autorizzazione propri.

Si presume che si stia utilizzando RACF come ESM. Se si utilizza un ESM diverso, potrebbe essere necessario interpretare le informazioni fornite per RACF in un modo pertinente per il proprio ESM.

È possibile specificare se si desidera che i controlli di autorizzazione siano attivati o disattivati per ogni gestore code singolarmente o per ogni gestore code in un gruppo di condivisione code. Questo livello di controllo è denominato *sicurezza del sottosistema*. Se si disattiva la sicurezza del sottosistema per un determinato gestore code, non vengono eseguiti controlli di autorizzazione per tale gestore code.

Se si attiva la sicurezza del sottosistema per un particolare gestore code, i controlli delle autorizzazioni possono essere eseguiti a due livelli:

#### **Sicurezza a livello di gruppo di condivisione code**

I controlli delle autorizzazioni utilizzano profili RACF condivisi da tutti i gestori code nel gruppo di condivisione code. Ciò significa che ci sono meno profili da definire e gestire, rendendo più semplice l'amministrazione della sicurezza.

#### **sicurezza a livello di gestore code**

I controlli delle autorizzazioni utilizzano profili RACF specifici del gestore code.

È possibile utilizzare una combinazione di gruppo di condivisione code e sicurezza a livello di gestore code. Ad esempio, è possibile disporre i profili specifici di un gestore code per sovrascrivere quelli del gruppo di condivisione code a cui appartiene.

La sicurezza del sottosistema, la sicurezza a livello di gruppo di condivisione code e la sicurezza a livello di gestore code vengono attivate o disattivate definendo *profili switch*. Un profilo switch è un profilo RACF normale che ha un significato speciale per IBM MQ.

#### *Sicurezza dei comandi e sicurezza delle risorse dei comandi*

La sicurezza del comando si riferisce all'autorità di emettere un comando; l'autorizzazione della risorsa comando si riferisce all'autorità di eseguire un'operazione su una risorsa. Entrambi sono implementati e utilizzando le classi RACF.

I controlli dell'autorizzazione vengono eseguiti quando un amministratore di IBM MQ immette un comando MQSC. Questa operazione è denominata *sicurezza del comando*.

Per implementare la sicurezza dei comandi, è necessario definire alcuni profili RACF e fornire ai gruppi e agli ID utente necessari l'accesso a questi profili ai livelli richiesti. Il nome di un profilo per la sicurezza del comando contiene il nome di un comando MQSC.

Alcuni comandi MQSC eseguono un'operazione su una risorsa IBM MQ, come il comando DEFINE QLOCAL per creare una coda locale. Quando un amministratore immette un comando MQSC, vengono eseguiti dei controlli di autorizzazione per stabilire se l'operazione richiesta può essere eseguita sulla risorsa specificata nel comando. Questa operazione è denominata *sicurezza della risorsa comando*.

Per implementare la sicurezza delle risorse del comando, è necessario definire alcuni profili RACF e fornire ai gruppi e ID utenti necessari l'accesso a tali profili ai livelli richiesti. Il nome di un profilo per la sicurezza della risorsa comando contiene il nome di una risorsa IBM MQ e il relativo tipo (QUEUE, PROCESS, NAMELIST, TOPIC, AUTHINFO o CHANNEL).

La sicurezza del comando e la sicurezza della risorsa comando sono indipendenti. Ad esempio, quando un amministratore immette il comando:

```
DEFINE QLOCAL(MOON.EUROPA)
```

vengono eseguiti i seguenti controlli di autorizzazione:

- La sicurezza del comando controlla che l'amministratore sia autorizzato a immettere il comando DEFINE QLOCAL.
- La sicurezza della risorsa comando controlla che l'amministratore sia autorizzato ad eseguire un'operazione sulla coda locale denominata MOON.EUROPA.

La sicurezza dei comandi e la sicurezza delle risorse dei comandi possono essere attivate o disattivate definendo i profili di commutazione.

#### Comandi MQSC e la coda di input dei comandi di sistema su z/OS

Utilizzare questo argomento per comprendere in che modo il server dei comandi elabora i comandi MQSC diretti alla coda di input dei comandi di sistema su z/OS.

La sicurezza del comando e la sicurezza della risorsa comando vengono utilizzate anche quando il server dei comandi richiama un messaggio contenente un comando MQSC dalla coda di input del comando di sistema. L'ID utente utilizzato per i controlli di autorizzazione è quello trovato nel campo *UserIdentifier* nel descrittore del messaggio contenente il comando MQSC. Questo ID utente deve disporre delle autorizzazioni richieste sul gestore code su cui viene elaborato il comando. Per ulteriori informazioni sul campo *UserIdentifier* e su come è impostato, consultare [Contesto del messaggio](#).

I messaggi che contengono i comandi MQSC vengono inviati alla coda di input dei comandi di sistema nelle seguenti circostanze:

- Le operazioni e i pannelli di controllo inviano comandi MQSC alla coda di input dei comandi di sistema del gestore code di destinazione. I comandi MQSC corrispondono alle azioni scelte sui pannelli. Il campo *UserIdentifier* in ogni messaggio è impostato sull'ID utente TSO dell'amministratore.
- La funzione COMMAND del programma di utilità IBM MQ, CSQUTIL, invia i comandi MQSC nel dataset di input alla coda di input del comando di sistema del gestore code di destinazione. Le funzioni COPY e EMPTY inviano i comandi DISPLAY QUEUE e DISPLAY STGCLASS. Il campo *UserIdentifier* in ogni messaggio è impostato sull'ID utente lavoro.
- I comandi MQSC nei dataset CSQINPX vengono inviati alla coda di input dei comandi di sistema del gestore code a cui è connesso l'iniziatore di canali. Il campo *UserIdentifier* in ogni messaggio è impostato sull'ID utente dello spazio di indirizzo dell'iniziatore di canali.

Nessun controllo di autorizzazione viene eseguito quando i comandi MQSC vengono emessi dai dataset CSQINP1 e CSQINP2. È possibile controllare a chi è consentito aggiornare questi dataset utilizzando la protezione dei dataset RACF.

- All'interno di un gruppo di condivisione code, un iniziatore di canali potrebbe inviare comandi START CHANNEL alla coda di input dei comandi di sistema del gestore code a cui è connesso. Un comando viene inviato quando un canale in uscita che utilizza una coda di trasmissione condivisa viene avviato dall'attivazione. Il campo *UserIdentifier* in ogni messaggio è impostato sull'ID utente dello spazio di indirizzo dell'iniziatore di canali.
- Un'applicazione può inviare comandi MQSC ad una coda di input di comandi di sistema. Per impostazione predefinita, il campo *UserIdentifier* in ogni messaggio è impostato sull'ID utente associato all'applicazione.
- Sui sistemi UNIX, Linux, and Windows, il comando di controllo **runmqsc** può essere usato in modalità indiretta per inviare comandi MQSC alla coda di input dei comandi di sistema di un gestore code su z/OS. Il campo *UserIdentifier* in ogni messaggio è impostato sull'ID dell'amministratore che ha immesso il comando **runmqsc**.

#### Accesso ai dataset del gestore code su z/OS

Gli amministratori IBM MQ for z/OS hanno bisogno dell'autorità per accedere ai dataset del gestore code. Utilizzare questo argomento per comprendere quali dataset necessitano della protezione RACF.

Questi dataset includono:

- I dataset a cui fanno riferimento CSQINP1, CSQINP2, CSQINPT e CSQXLIB nella procedura dell'attività avviata del gestore code
- I set di pagine del gestore code, i dataset di log attivi, i dataset di log di archiviazione e i dataset di bootstrap (BSDS)
- I dataset a cui fanno riferimento CSQXLIB e CSQINPX nella procedura dell'attività avviata dell'iniziatore di canali

È necessario proteggere i dataset in modo che nessun utente non autorizzato possa avviare un gestore code o ottenere l'accesso ai dati del gestore code. A tale scopo, utilizzare la protezione del dataset RACF .

## **Autorizzazione per le applicazioni ad utilizzare IBM 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 MQ emettendo chiamate MQI:

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


Le applicazioni possono anche utilizzare comandi PCF per gestire oggetti IBM 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 e quelle fornite con IBM MQ for z/OS. Le applicazioni fornite con IBM MQ for z/OS includono:

- Le operazioni e i pannelli di controllo
- Il programma di utilità IBM MQ , CSQUTIL
- Il programma di utilità gestore code di messaggi non recapitabili, CSQUDLQH

Le applicazioni che utilizzano IBM MQ classes for Java, IBM MQ classes for JMS, IBM MQ classes for .NET o i client del servizio messaggi per C/C++ e .NET utilizzano MQI indirettamente.

Gli MCA, inoltre, emettono chiamate MQI e gli ID utente associati agli MCA necessitano dell'autorizzazione per accedere a questi oggetti IBM MQ . Per ulteriori informazioni su questi ID utente e sulle autorizzazioni richieste, consultare [“Autorizzazione canale” a pagina 100](#).

Su z/OS, le applicazioni possono anche utilizzare i comandi MQSC per accedere a questi oggetti IBM MQ , ma la sicurezza dei comandi e la sicurezza delle risorse dei comandi forniscono i controlli di autorizzazione in queste circostanze.  Per ulteriori informazioni, consultare [“Sicurezza dei comandi e sicurezza delle risorse dei comandi” a pagina 77](#) e [“Comandi MQSC e la coda di input dei comandi di sistema su z/OS” a pagina 78](#).

Su IBM i, un utente che immette un comando CL nel gruppo 2 potrebbe richiedere l'autorizzazione per accedere ad un oggetto IBM MQ associato al comando. Per ulteriori informazioni, vedere [“Quando vengono eseguiti i controlli di autorizzazione” a pagina 79](#).

### **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.

Su IBM i, i controlli delle autorizzazioni possono essere eseguiti anche quando un utente immette un comando CL nel gruppo 2 che accede a uno qualsiasi di questi oggetti IBM MQ . 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 MQ. IBM 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 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 descrittore 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 z/OS, viene eseguito un controllo sul profilo coda RACF che corrisponde al nome del gestore code remoto. Su **Multiplatforme**, viene effettuato un controllo rispetto al profilo RQMNAME che corrisponde al nome del gestore code remoto, se viene utilizzato il cluster. Un'applicazione può fare riferimento esplicitamente a una coda cluster impostando il campo `ObjectName` nel descrittore dell'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 IBM 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 81](#).

#### **IBM i Su IBM i, quando un utente immette un comando CL nel Gruppo 2 che opera su un oggetto IBM MQ**

Questa regola include il caso in cui un comando CL nel Gruppo 2 opera su un oggetto delle informazioni di autenticazione.

I controlli vengono eseguiti per stabilire se l'utente dispone dell'autorizzazione per operare su un oggetto IBM MQ associato al comando. I controlli vengono eseguiti a meno che l'utente non sia un membro del gruppo QMQMADM o non disponga dell'autorità \*ALLOBJ . L'autorizzazione richiesta dipende dal tipo di operazione che il comando esegue sull'oggetto. Ad esempio, il comando **CHGMQM**, Change MQM Queue, richiede l'autorità per modificare gli attributi della coda specificata dal comando. Al contrario, il comando **DSPMQM**, Visualizza coda MQM, richiede l'autorità per visualizzare gli attributi della coda specificata dal comando.

Molti comandi operano su più di un oggetto. Ad esempio, per immettere il comando **DLTMQM**, Elimina coda MQM, sono richieste le seguenti autorizzazioni:

- L'autorizzazione per connettersi al gestore code specificato dal comando
- L'autorità per cancellare la coda specificata dal comando

Alcuni comandi operano su nessun oggetto. In questo caso, l'utente richiede solo l'autorità IBM i per immettere uno di tali comandi. **STRMQLSR**, Avviare MQM Listener, è un esempio di tale comando.

#### **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; una transazione CICS , ad esempio.

##### **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 7.0 o 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 a gestire gli oggetti IBM MQ su IBM i , sistemi UNIX, Linux, and Windows

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

### • Autenticazione.

Il controllo di autenticazione eseguito da OAM fornito con IBM 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 proprio controllo di autenticazione quando un'applicazione si connette a un gestore code e si verificano le seguenti condizioni:

- Se una struttura MQCSP è stata fornita dall'applicazione di connessione e
- l'attributo *AuthenticationType* nella struttura MQCSP ha il valore MQCSP\_AUTH\_USER\_ID\_AND\_PWD e
- Il valore CHCKLOCL o CHKCCLNT sull'oggetto AUTHINFO configurato non è 'NONE '


La procedura di autenticazione in OAM convalida la parola d'ordine utilizzando i servizi del sistema operativo, che potrebbero essere stati configurati per eseguire ulteriori controlli, come ad esempio verificare che il nome utente non abbia avuto troppi tentativi di test della parola d'ordine non corretti.

È possibile utilizzare meccanismi di autenticazione alternativi se si scrive un nuovo componente del servizio di autorizzazione o se se ne ottiene uno da un fornitore.

### • 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  IBM i , UNIX, Linux, and Windows , *servizio di autorizzazione* fornisce il controllo dell'accesso quando un'applicazione emette una chiamata MQI per accedere a un oggetto IBM 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 MQ , anche quando UAC è abilitato.

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

Il servizio di autorizzazione fornisce anche i controlli di autorizzazione quando un comando PCF opera su uno di tali oggetti IBM 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.

**IBM i** Su IBM i, a meno che l'utente non sia un membro del gruppo QMQMADM o non disponga dell'autorizzazione \*ALLOBJ, il servizio di autorizzazione fornisce anche i controlli di autorizzazione quando un utente immette un comando CL nel gruppo 2 che opera su uno qualsiasi di questi oggetti IBM MQ o un oggetto delle informazioni di autenticazione.

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 MQ.

Il componente del servizio di autorizzazione fornito con IBM MQ viene 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 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 **IBM i** IBM i e 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 MQ continua ad osservare una limitazione di 12 caratteri su tutte le piattaforme UNIX. Se si utilizza un ID utente superiore a 12 caratteri, IBM MQ lo sostituisce con il valore "SCONOSCIUTO". 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 MQCONN. La struttura viene trasmessa alla funzione OAM Authenticate User (MQZ\_AUTHENTICATE\_USER), che imposta i campi del contesto di identità appropriati. Se una connessione MQCONN da un client IBM 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](#).

**Avviso:** In alcuni casi, la password in una struttura MQCSP per un'applicazione client verrà inviata attraverso una rete in testo semplice. Per assicurarsi che le password dell'applicazione client siano protette in modo appropriato, consultare [IBM MQCSP password protection](#).

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.

**IBM i** Su IBM i, un amministratore utilizza il comando CL GRMQMAUT per concedere le autorizzazioni e il comando CL RVKMQMAUT per revocare le autorizzazioni. Possono essere utilizzati anche profili generici. Ad esempio, il comando CL:

```
GRMQMAUT MQMNAME(JUPITER) OBJTYPE(*Q) OBJ('MOON.*') USER(VOYAGER) AUT(*PUT)
```

fornisce la stessa funzione dell'esempio precedente di un comando **setmqaut**; consente ai membri del gruppo VOYAGER di inserire messaggi su qualsiasi coda con un nome che inizia con i caratteri MOON.

**IBM i** Il comando CL DSPMQMAUT visualizza le autorizzazioni correnti che l'utente o il gruppo ha per un oggetto specificato. I comandi CL WRKMQMAUT e WRKMQMAUTD sono disponibili anche per gestire le autorizzazioni correnti associate agli oggetti e 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 sistemi **IBM i** IBM i, UNIX, Linux, and Windows, è possibile utilizzare comandi PCF per accedere ai comandi di gestione OAM.

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

<i>Tabella 8. Comandi PCF e relativi comandi OAM equivalenti</i>	
<b>Comando PCF</b>	<b>comando OAM</b>
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](#).

### **Autorizzazione per gestire gli oggetti IBM MQ su z/OS**

Su z/OS, ci sono sette categorie di controllo dell'autorità associate alle chiamate a MQI. È necessario definire alcuni profili RACF e fornire l'accesso appropriato a tali profili. Utilizzare il profilo *RESLEVEL* per controllare il numero di ID utente controllati.

Le sette categorie di controllo dell'autorità associate alle chiamate alla MQI:

#### **Sicurezza connessione**

I controlli di autorizzazione eseguiti quando un'applicazione si connette a un gestore code

#### **Sicurezza coda**

I controlli di autorizzazione eseguiti quando un'applicazione apre una coda o elimina una coda dinamica permanente

### **Sicurezza del processo**

I controlli di autorizzazione eseguiti quando un'applicazione apre un oggetto processo

### **Sicurezza elenco nomi**

I controlli di autorizzazione eseguiti quando un'applicazione apre un oggetto elenco nomi

### **sicurezza utente alternativa**

I controlli di autorizzazione eseguiti quando un'applicazione richiede un'autorizzazione utente alternativa durante l'apertura di un oggetto

### **sicurezza del contesto**

I controlli di autorizzazione eseguiti quando un'applicazione apre una coda e specifica che intende impostare o trasmettere le informazioni di contesto nei messaggi che inserisce nella coda

### **sicurezza dell'argomento**

I controlli di autorizzazione eseguiti quando un'applicazione apre un argomento

Ogni categoria di controllo di autorizzazione viene implementata nello stesso modo in cui viene implementata la sicurezza del comando e della risorsa del comando. È necessario definire determinati profili RACF e fornire ai gruppi e ID utente necessari l'accesso a tali profili ai livelli richiesti. Per la sicurezza della coda, il livello di accesso determina i tipi di operazioni che l'applicazione può eseguire su una coda. Per la sicurezza del contesto, il livello di accesso determina se l'applicazione può:

- Passa tutti i campi di contesto
- Passare tutti i campi di contesto e impostare i campi di contesto identità
- Passare e impostare tutti i campi di contesto

Ogni categoria di controllo dell'autorizzazione può essere attivata o disattivata definendo i profili di switch.

Tutte le categorie, ad eccezione della sicurezza della connessione, sono note collettivamente come *sicurezza delle risorse API*.

Per impostazione predefinita, quando un controllo di sicurezza delle risorse API viene eseguito come risultato di una chiamata MQI da un'applicazione che utilizza una connessione batch, viene controllato solo un ID utente. Quando un controllo viene eseguito come risultato di una chiamata MQI da una applicazione CICS o IMS o dall'inziatore del canale, vengono controllati due ID utente.

Definendo un *profilo RESLEVEL*, tuttavia, è possibile controllare se vengono selezionati zero, uno o due ID utente. Il numero di ID utente che vengono controllati è determinato dall'ID utente associato al tipo di connessione quando un'applicazione si connette al gestore code e dal livello di accesso che l'ID utente ha al profilo RESLEVEL. L'ID utente associato a ciascun tipo di connessione è:

- L'ID utente dell'attività di connessione per connessioni batch
- L'ID utente dello spazio di indirizzo CICS per le connessioni CICS
- L'ID utente dello spazio di indirizzi della regione IMS per le connessioni IMS
- L'ID utente dello spazio di indirizzo dell'inziatore di canali per le connessioni dell'inziatore di canali

Per ulteriori informazioni sull'autorità per gestire gli oggetti IBM MQ su z/OS, consultare [“Autorizzazione per gestire IBM MQ su z/OS”](#) a pagina 77.

## **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 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 (Distinguished Name) oggetto 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 IBM MQ for IBM i**

Questa sezione tratta gli aspetti della messaggistica remota della sicurezza.

È necessario fornire agli utenti l'autorizzazione per utilizzare le strutture IBM MQ for IBM i . Questa autorizzazione è organizzata in base alle azioni da intraprendere rispetto a oggetti e 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

L'agente del canale dei messaggi su un sito remoto deve controllare che il messaggio che si sta consegnando derivi da un utente con l'autorità di inviare il messaggio su 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:

- Decreta nella definizione del canale che i messaggi devono contenere l'autorizzazione *context* accettabile, altrimenti vengono eliminati.
- Implementare i record di autenticazione del canale per rifiutare i tentativi di connessione non desiderati o per impostare un valore MCAUSER basato su uno dei seguenti valori: l'indirizzo IP remoto, l'ID utente remoto, il DN (Distinguished Name) TLS fornito o il nome del gestore code remoto.
- Implementare il controllo di sicurezza dell'uscita utente per assicurarsi che il canale di messaggi corrispondente 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.
- Implementare l'elaborazione dei messaggi di uscita utente per garantire che i singoli messaggi vengano controllati per l'autorizzazione.

Di seguito sono riportati alcuni fatti relativi al modo in cui IBM MQ for IBM i opera la sicurezza:

- Gli utenti sono identificati e autenticati da IBM i.
- I servizi del gestore code richiamati dalle applicazioni vengono eseguiti con l'autorizzazione del profilo utente del gestore code, ma nel processo dell'utente.
- I servizi del gestore code richiamate dai comandi utente vengono eseguiti con l'autorizzazione del profilo utente del gestore code.

## Sicurezza degli oggetti su 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 MQ .

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

### ID utente su 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 del gruppo mqm e del gruppo di amministratori sui sistemi Windows se questo ID utilizzerà i comandi di amministrazione IBM MQ .

### ID utente su sistemi Windows

Sui sistemi Windows , *se non è installata alcuna uscita 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

Le piattaforme diverse da Windows, 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, le seguenti conversioni vengono eseguite da MCA (message channel agent) 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 UNIX, Linux, and Windows per qualsiasi altro motivo.

## Utilizzo di un servizio di autorizzazione personalizzato

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

Il componente del servizio di autorizzazione fornito con IBM 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 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.

<i>Tabella 9. L'ID utente utilizzato da un canale di connessione server</i>	
<b>ID utente</b>	<b>Quando utilizzato</b>
L'ID utente impostato da un'uscita di sicurezza	Utilizzato a meno che non sia bloccato da una regola <b>CHLAUTH TYPE (BLOCKUSER)</b> . Per ulteriori informazioni, vedere la seguente sezione, <u>“Impostazione dell'ID utente in un'uscita di sicurezza”</u> a pagina 90 .
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 <b>MCAUSER</b> 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 utente è impostato con 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, <u>“L'ID utente che esegue il programma del canale”</u> a pagina 90 .

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 IBM MQ MQI clients, 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.

Il client IBM 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 (non z/OS ), l'ID utente dalla voce `inetd.conf` o l'ID utente che ha avviato il listener.
- Per SNA (non z/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 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 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 UNIX, Linux, and 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. Puoi scegliere di utilizzare 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

[“Confronto tra sicurezza a livello di collegamento e sicurezza a livello di applicazione” a pagina 91](#)

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 96](#)

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/TLS” a pagina 103](#)

Il supporto TLS in IBM 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 10 a pagina 91](#).

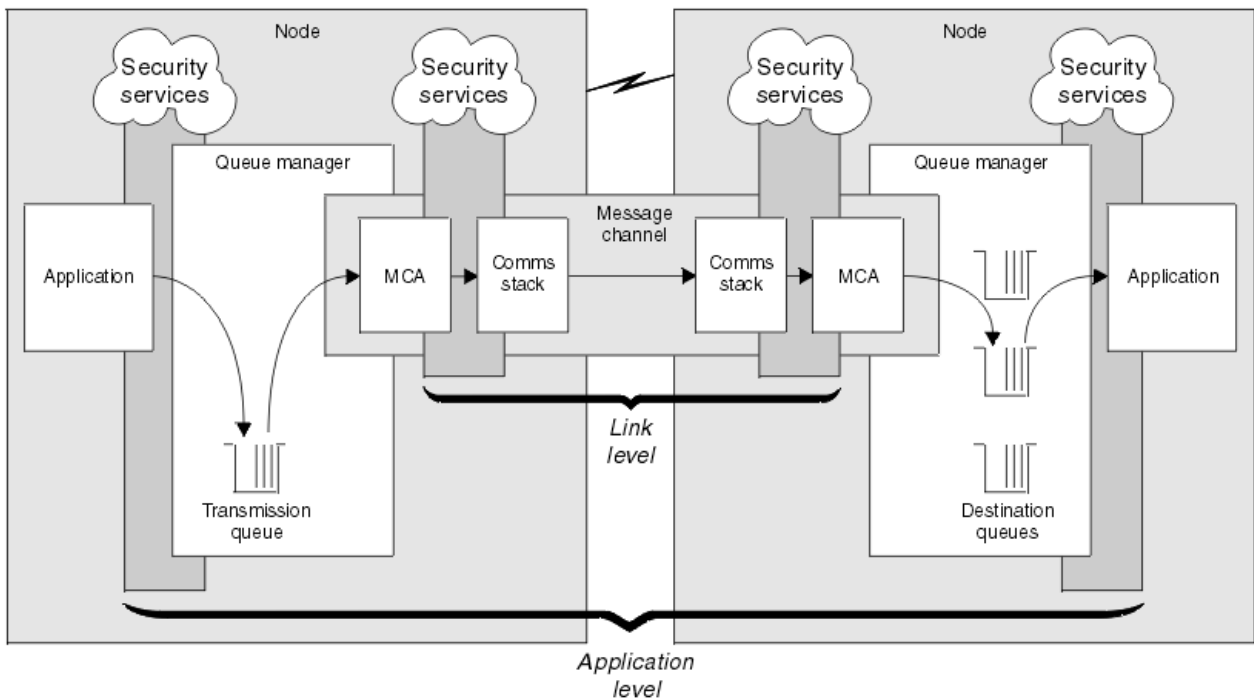


Figura 10. Sicurezza a livello di collegamento e sicurezza a livello di applicazione

## 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 10 a pagina 91](#).

## **Gestori code non in esecuzione in ambienti controllati e attendibili**

Se un gestore code è in esecuzione in un ambiente controllato e attendibile, i meccanismi di controllo accessi forniti da IBM 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 IBM 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 10 a pagina 91](#).

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 MQ**

Il mezzo principale per fornire riservatezza e integrità dei dati in IBM MQ è l'utilizzo di TLS. Per ulteriori informazioni sull'utilizzo di TLS in IBM MQ, consultare [“Protocolli di sicurezza TLS in IBM MQ” a pagina 24](#). Per l'autenticazione, IBM 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, vedere [“Record di autenticazione di canale” a pagina 47](#).

#### *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 96](#). Lo stesso argomento descrive anche il programma di uscita canale fornito con IBM MQ for 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.

### **Informazioni 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 IBM MQo da una combinazione di questi che lavorano insieme. La sicurezza a livello di applicazione viene illustrata in [Figura 10 a pagina 91](#).

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*

Advanced Message Security (AMS) è un componente di IBM MQ che fornisce un livello elevato di protezione per i dati sensibili che passano attraverso la rete IBM 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 MQ. Tuttavia, tali soluzioni richiedono competenze specialistiche e possono essere complicate e costose da mantenere. 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.

Advanced Message Security estende le funzioni di protezione di IBM 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 MQ.
- Si integra con Managed File Transfer per fornire una soluzione di messaggistica sicura end - to - end.

Per ulteriori informazioni, vedere [“Advanced Message Security” a pagina 520](#).

#### *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 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 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 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 programma di uscita canale sono illustrati in [Figura 11 a pagina 97](#) e sono descritti nei seguenti argomenti.



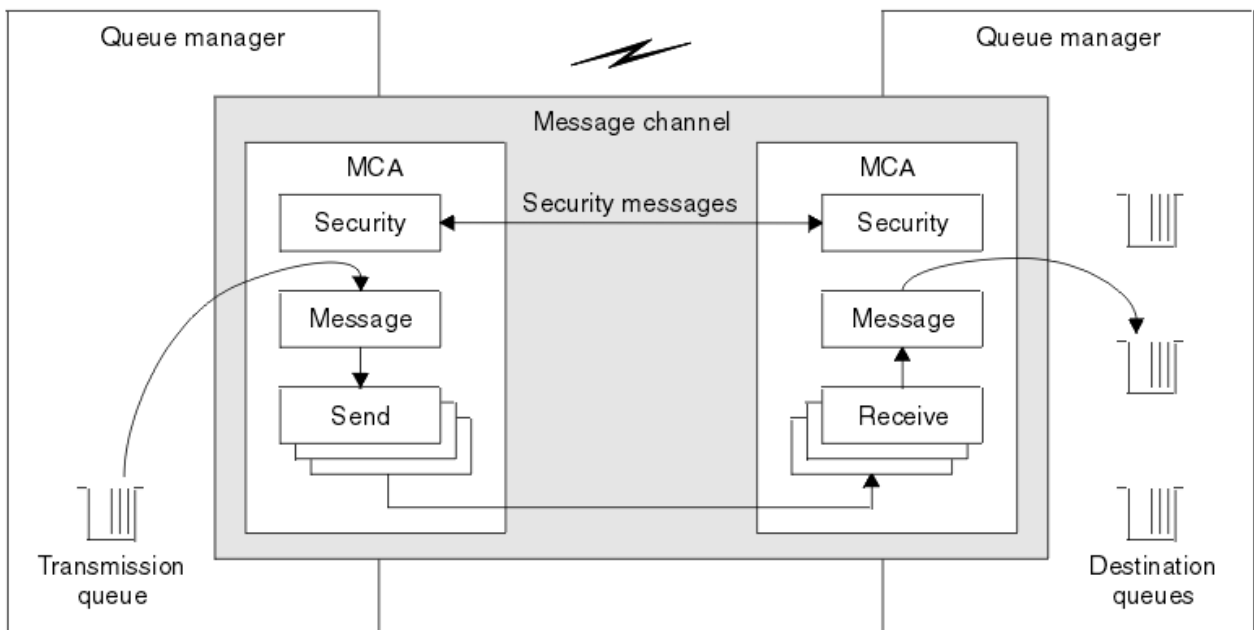


Figura 11. Uscite di sicurezza, messaggio, invio e ricezione su un canale di messaggi

### Informazioni correlate

[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 94.

### 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. Questo perché lo scopo di un canale MQI è abilitare i parametri di input e output delle chiamate MQI per il flusso tra l'applicazione IBM MQ MQI client 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 94.

### ***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 riacquisito 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 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 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 94.

## 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 dell'applicazione, è possibile utilizzare i programmi di uscita API se le funzioni standard non soddisfano i requisiti. È possibile scegliere di utilizzare Advanced Message Security (AMS) per firmare digitalmente i messaggi al fine di proteggere da modifiche non autorizzate.

A livello di link, è possibile scegliere di utilizzare TLS, nel qual caso è necessario pianificare l'uso 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/TLS”](#) a pagina 103

Il supporto TLS in IBM 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 MQ”](#) a pagina 23

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

[“pianificazione per Advanced Message Security”](#) a pagina 95

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

### Informazioni correlate

[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 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 MQ utilizzando le interfacce IBM MQ

Configurare IBM MQ per emettere eventi di strumentazione, in particolare eventi di comando ed eventi di configurazione.

### Le modifiche apportate a IBM MQ al di fuori del suo controllo

Alcune modifiche possono influenzare il funzionamento di IBM MQ, ma non possono essere monitorate direttamente da IBM MQ. Esempi di tali modifiche includono le modifiche ai file di configurazione `mqs.ini`, `qm.inie` e `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 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 MQ può essere configurato per emettere eventi di strumentazione.

### Attività dell'applicazione all'interno di IBM MQ

Per controllare le azioni delle applicazioni, ad esempio l'apertura di code e l'inserimento e l'acquisizione di messaggi, configurare IBM 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 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 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 fornire l'accesso a varie risorse IBM MQ. Gli MCA (Message Channel Agent) sono essenzialmente applicazioni IBM MQ che spostano i messaggi tra gestori code e, pertanto, richiedono l'accesso a varie risorse IBM MQ per funzionare correttamente.

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 IBM MQ, i canali possono essere protetti dal supporto 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 di ricezione, 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 di controllo degli accessi se non si utilizza OAM (Object Authority Manager) di IBM 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 dispone di diritti di gestione completi 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.
- Non è necessario concedere l'autorizzazione *dsp* e *ctrlx* per l'ID utente MCAUSER di un canale ricevente.
- Prima di IBM MQ 8.0.0 Fix Pack 4, se si utilizza un ID utente che non dispone dei privilegi amministrativi IBM MQ, è necessario concedere l'autorizzazione **dsp** e **ctrlx** per il canale a tale ID utente per il funzionamento del canale.

Da IBM MQ 8.0.0 Fix Pack 4, non ci sono controlli di autorizzazione quando un canale si risincronizza e corregge i numeri di sequenza.

Tuttavia, l'immissione manuale di un comando RESET CHANNEL richiede ancora **+dsp** e **+ctrlx** in tutte le release.



**Attenzione:** Quando è necessaria una reimpostazione del canale per la conferma batch del messaggio, IBM MQ tenta di interrogare il canale, che richiede l'autorizzazione **+dsp**.

- 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. I seguenti comandi sono esempi del tipo di comando che è possibile immettere per concedere l'autorizzazione a un ID utente da un sistema remoto:

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

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

dove *Profile* è un canale.

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

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

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

dove *Profile* 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 47](#)

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.

### **Informazioni correlate**

[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 MQ non sono oggetti IBM MQ ; l'accesso ad essi non è controllato da OAM. IBM 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 emette 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 anche essere un membro del gruppo mqm sulla macchina di destinazione per emettere i comandi MQSC equivalenti tramite il comando PCF Escape 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 12 a pagina 121](#).

### **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 96 per ulteriori informazioni sulle uscite dei canali.

### **Protezione dei canali con SSL/TLS**

Il supporto TLS in IBM 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.

### **Certificati digitali e archivi di chiavi**

Si consiglia di impostare l'attributo dell'etichetta del certificato del gestore code (**CERTLABL**) al nome del certificato personale da utilizzare per la maggior parte dei canali e sostituirlo per le eccezioni, impostando l'etichetta del certificato sui canali che richiedono certificati diversi.

Se sono necessari molti canali con certificati diversi dal certificato predefinito impostato sul gestore code, si consiglia di dividere i canali tra diversi gestori code o utilizzare un proxy MQIPT davanti al gestore code per presentare un certificato differente.

È possibile utilizzare un certificato differente per ogni canale, ma se si memorizzano troppi certificati in un repository delle chiavi, è possibile che le prestazioni vengano influenzate quando si avviano i canali TLS. Provare a mantenere il numero di certificati in un repository di chiavi inferiore a circa 50 e considerare 100 come un valore massimo poiché le prestazioni di GSKit diminuiscono notevolmente con repository di chiavi più grandi.

L'abilitazione di più certificati sullo stesso gestore code aumenta le possibilità che più certificati CA vengano utilizzati sullo stesso gestore code. Ciò aumenta le probabilità di conflitti dello spazio dei nomi del nome distinto del soggetto del certificato per i certificati emessi da autorità di certificazione separate.

Mentre le autorità di certificazione professionali sono probabilmente più attente, le autorità di certificazione interne spesso non hanno convenzioni di denominazione chiare e si potrebbe finire con corrispondenze indesiderate tra una CA e l'altra.

È necessario controllare il DN (Distinguished Name) dell'emittente del certificato oltre al DN (Distinguished Name) del soggetto. A tale scopo, utilizzare un record di autenticazione di canale SSLPEERMAP e impostare i campi **SSLPEER** e **SSLCERTI** in modo che corrispondano rispettivamente al DN soggetto e al DN emittente.

### **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 UNIX, Linux, and Windows”](#) a pagina 285.

**IBM i** Sui sistemi IBM i, è possibile utilizzare i certificati firmati dalla CA locale. Per istruzioni, consultare [“Richiesta di un certificato server su IBM i”](#) a pagina 270.

**z/OS** Su z/OS, è possibile utilizzare certificati autofirmati o certificati firmati dalla CA locale. Consultare [“Creazione di un certificato personale autofirmato su z/OS”](#) a pagina 313 o [“Richiesta di un certificato personale su z/OS”](#) a pagina 313 per istruzioni.

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 pertanto necessario scegliere un CipherSpec appropriato per i certificati digitali. Allo stesso modo, se la politica di sicurezza della propria organizzazione richiede l'utilizzo di una particolare CipherSpec, è necessario ottenere certificati digitali adeguati.

Per ulteriori informazioni sulla relazione tra CipherSpecs e certificati digitali, fare riferimento a [“Certificati digitali e compatibilità CipherSpec in IBM MQ”](#) a pagina 42

## 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 IBM MQ, consultare [“Politiche di convalida dei certificati in IBM MQ”](#) a pagina 42.

## Pianificazione del controllo della revoca del certificato

L'abilitazione di più certificati da diverse autorità di certificazione potrebbe causare un ulteriore controllo non necessario della revoca del certificato.

In particolare, se è stato esplicitamente configurato l'utilizzo di un server di revoca da una particolare CA, ad esempio utilizzando un oggetto AUTHINFO o una struttura MQAIR (Authentication information record), un controllo di revoca ha esito negativo quando viene presentato un certificato da una CA differente.

Si consiglia di evitare la configurazione esplicita del server di revoca certificati. Invece, è necessario abilitare il controllo implicito in cui ciascun certificato contiene la propria ubicazione del server di revoca in un'estensione del certificato, ad esempio, Punto di distribuzione CRL o Accesso OCSP AuthorityInfo.

Per ulteriori informazioni, consultare [OCSPCheckExtensions](#) e [CDPCheckExtensions](#).

## Comandi e attributi per il supporto TLS

Il protocollo TLS (Transport Layer Security) fornisce la sicurezza del canale, con protezione da intercettazioni, manomissioni e imitazioni. Il supporto IBM MQ per TLS consente di specificare, nella



definizione del canale, che un determinato canale utilizza la sicurezza TLS. 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 TLS:

**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 TLS:

**CERTLABL**

Definisce un'etichetta di certificato personale da utilizzare.

**SSLCRLNL**

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

**SSLCRYP**

Su sistemi Windows , UNIX and Linux , imposta l'attributo 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 TLS se un canale che utilizza TLS non riesce a stabilire una connessione TLS.

**SSLFIPS**

Specifica se devono essere utilizzati solo algoritmi certificati FIPS se la crittografia viene eseguita in IBM 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**

Sui sistemi UNIX, Linux, and Windows , associa un contenitore 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 sicurezza TLS su sistemi Windows , UNIX and Linux .

**SSLRKEYC**

Il numero di byte da inviare e ricevere in una conversazione TLS 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 TLS:

**CERTLABL**

Definisce un'etichetta di certificato personale da utilizzare.

**SSLCAUTH**

Definisce se IBM MQ richiede e convalida un certificato dal client TLS.

**SSLCIPH**

Specifica la funzione e il livello di codifica (CipherSpec), ad esempio TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA. 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**, **rcdmqimg** **dspmqfls** per supportare l'oggetto delle informazioni di autenticazione. Descrive anche il comando **runmqckm**

(iKeycmd) per la gestione dei certificati sui sistemi UNIX and Linux e lo strumento **runmqakm** per la gestione dei certificati su UNIX, Linux, and Windows. Consultare le seguenti sezioni:

- [setmqaut](#)
- [dspmqaut](#)
- [dmpmqaut](#)
- [rcrmqobj](#)
- [rcdmqimg](#)
- [dspmqfls](#)
- [Gestione di chiavi e certificati](#)

Per una panoramica sulla sicurezza del canale utilizzando TLS, consultare

- [“Protocolli di sicurezza TLS in IBM MQ” a pagina 24](#)

Per i dettagli dei comandi MQSC associati a TLS, consultare

- [MODIFICA AUTHINFO](#)
- [DEFINE AUTHINFO](#)
- [DELETE AUTINFO](#)
- [VISUALIZZA AUTHINFO](#)

Per i comandi PCF associati a TLS, consultare

- [Modificare, copiare e creare l'oggetto delle informazioni di autenticazione](#)
- [Elimina oggetto informazioni di autenticazione](#)
- [Richiedi oggetto informazioni di autenticazione](#)

### ***IBM MQ for z/OS Canale di connessione server***

Il canale SVRCONN IBM MQ for z/OS non è sicuro senza implementare l'autenticazione del canale o aggiungere un'uscita di sicurezza utilizzando TLS. I canali SVRCONN non dispongono di un'uscita di protezione definita per impostazione predefinita.

### **Problemi di sicurezza**

I canali SVRCONN non sono sicuri come inizialmente definito, SYSTEM.DEF.SVRCONN ad esempio. Per proteggere un canale SVRCONN, è necessario impostare l'autenticazione di canale utilizzando il comando [SET CHLAUTH](#) oppure installare un'uscita di sicurezza e implementare TLS.

È necessario utilizzare un'uscita di sicurezza di esempio pubblicamente disponibile, scrivere un'uscita di sicurezza personalmente o acquistare un'uscita di sicurezza.

Ci sono diversi esempi disponibili che è possibile utilizzare come un buon punto di partenza per scrivere la propria uscita di sicurezza del canale SVRCONN.

In IBM MQ for z/OS, il membro CSQ4BCX3 nella libreria hlq.SCSQC37S è un esempio di uscita di sicurezza scritto in linguaggio C. L'esempio CSQ4BCX3 viene anche fornito precompilato nella libreria hlq.SCSQAUTH .

È possibile implementare l'uscita di esempio CSQ4BCX3 copiando il membro compilato hlq.SCSQAUTH(CSQ4BCX3) in una libreria di caricamento assegnata a CSQXLIB DD nel proprio CHIN Proc. Notare che CHIN richiede che la libreria di caricamento sia impostata come "Programma controllato".

Modificare il canale SVRCONN per impostare CSQ4BCX3 come uscita di sicurezza.

Quando un client si connette utilizzando tale canale SVRCONN, CSQ4BCX3 eseguirà l'autenticazione utilizzando la coppia RemoteUserIdentifier e RemotePassword da MQCD. Se l'autenticazione ha esito positivo, copierà l'identificativo RemoteUserin MCAUserIdentifier, modificando il contesto di identità del thread.

Se si sta scrivendo un client MQ Java , è possibile utilizzare le finestre a comparsa per interrogare l'utente e impostare MQEnvironment.userID e MQEnvironment.password. Questi valori verranno trasmessi quando viene stabilita la connessione.

Ora che si dispone di un'uscita di sicurezza funzionale, vi è la preoccupazione aggiuntiva che l'id utente e la password vengano trasmessi in testo semplice nella rete quando viene effettuata la connessione, così come il contenuto di eventuali messaggi successivi di MQ . È possibile utilizzare TLS per crittografare queste informazioni di connessione iniziali e il contenuto di tutti i messaggi MQ .

## Esempio

Per proteggere il IBM MQ Explorer SVRCONN channel SYSTEM.ADMIN.SVRCONN completare i seguenti passi:

1. Copiare hlq.SCSQAUTH(CSQ4BCX3) in una libreria di caricamento assegnata a CSQXLIB DD nel programma CHINIT.
2. Verificare che la libreria di caricamento sia controllata dal programma.
3. Modificare SYSTEM ADMIN.SVRCONN per utilizzare l'uscita di sicurezza CSQ4BCX3.
4. In IBM MQ Explorer, fare clic con il pulsante destro del mouse su z/OS Nome gestore code, selezionare **Dettagli connessione > Proprietà > ID utente** e immettere l'ID utente z/OS .
5. Connettersi al gestore code z/OS immettendo una password.

## Ulteriori informazioni

Affinché l'uscita CSQ4BCX3 venga eseguita in un ambiente controllato dal programma, tutti gli elementi caricati nello spazio di indirizzo CHIN devono essere caricati da una libreria controllata dal programma, ad esempio, tutte le librerie in STEPLIB e tutte le librerie denominate in CSQXLIB DD. Per impostare una libreria di caricamento come Programma controllato, immettere i comandi RACF . Nel seguente esempio il nome della libreria di caricamento è MY.TEST.LOADLIB.

```
RALTER PROGRAM * ADDMEM('MY.TEST.LOADLIB'//NOPADCHK)
SETROPTS WHEN(PROGRAM)REFRESH
```

Per modificare il canale SVRCONN in modo da implementare CSQ4BCX3, immettere il seguente comando MQ :

```
ALTER CHANNEL( SYSTEM ADMIN.SVRCONN) CHLTYPE(SVRCONN) SECYEXIT(CSQ4BCX3)
```

Nell'esempio precedente, il nome del canale SVRCONN utilizzato è SYSTEM ADMIN.SVRCONN.

Consultare [“Programmi di uscita canale”](#) a pagina 96 per ulteriori informazioni sulle uscite dei canali.

### Informazioni correlate

[Scrittura dei programmi di uscita del canale su z/OS](#)

## 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, gli IBM MQ MCA sono programmi di transazione. 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 sulle forme più forti di crittografia 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.

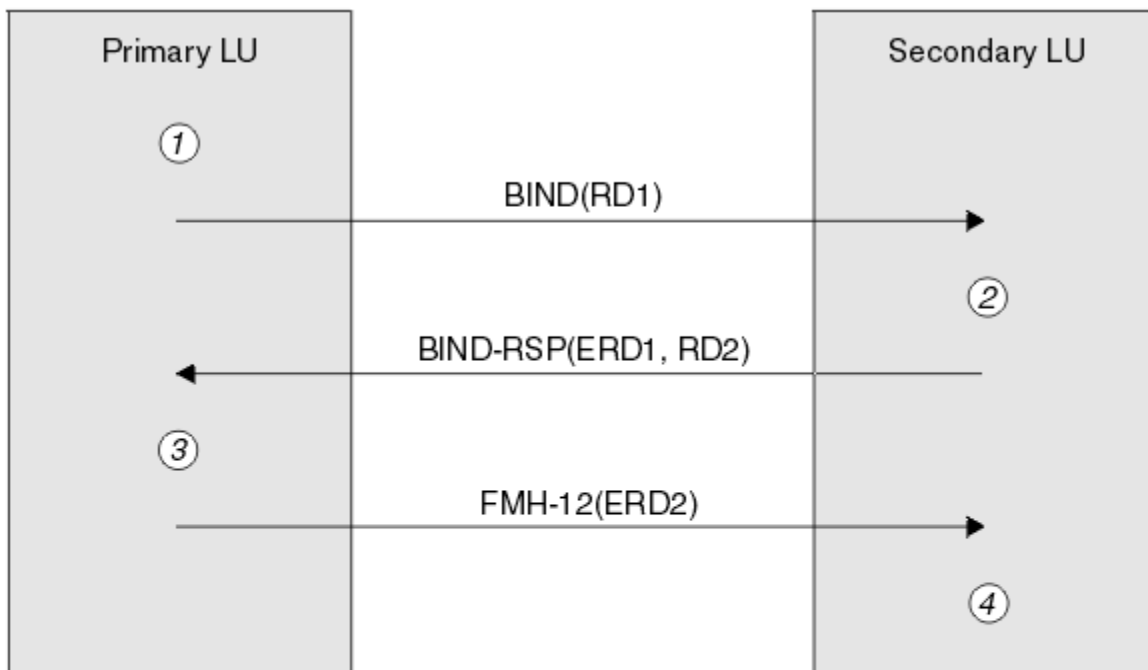
#### *Autenticazione a livello di sessione*

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 12 a pagina 109 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 12. 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 12 a pagina 109](#).

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, RD2 e 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 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 per 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 a livello di conversazione su IBM i, UNIXe Windows**

Utilizzare questo argomento per ottenere una panoramica del funzionamento dell'autenticazione a livello di conversazione su IBM i, UNIXe Windows.

Il supporto per l'autenticazione a livello di conversazione su IBM i, UNIXe Windows è illustrato in [Figura 13 a pagina 111](#). I numeri nel diagramma corrispondono ai numeri nella descrizione che segue.

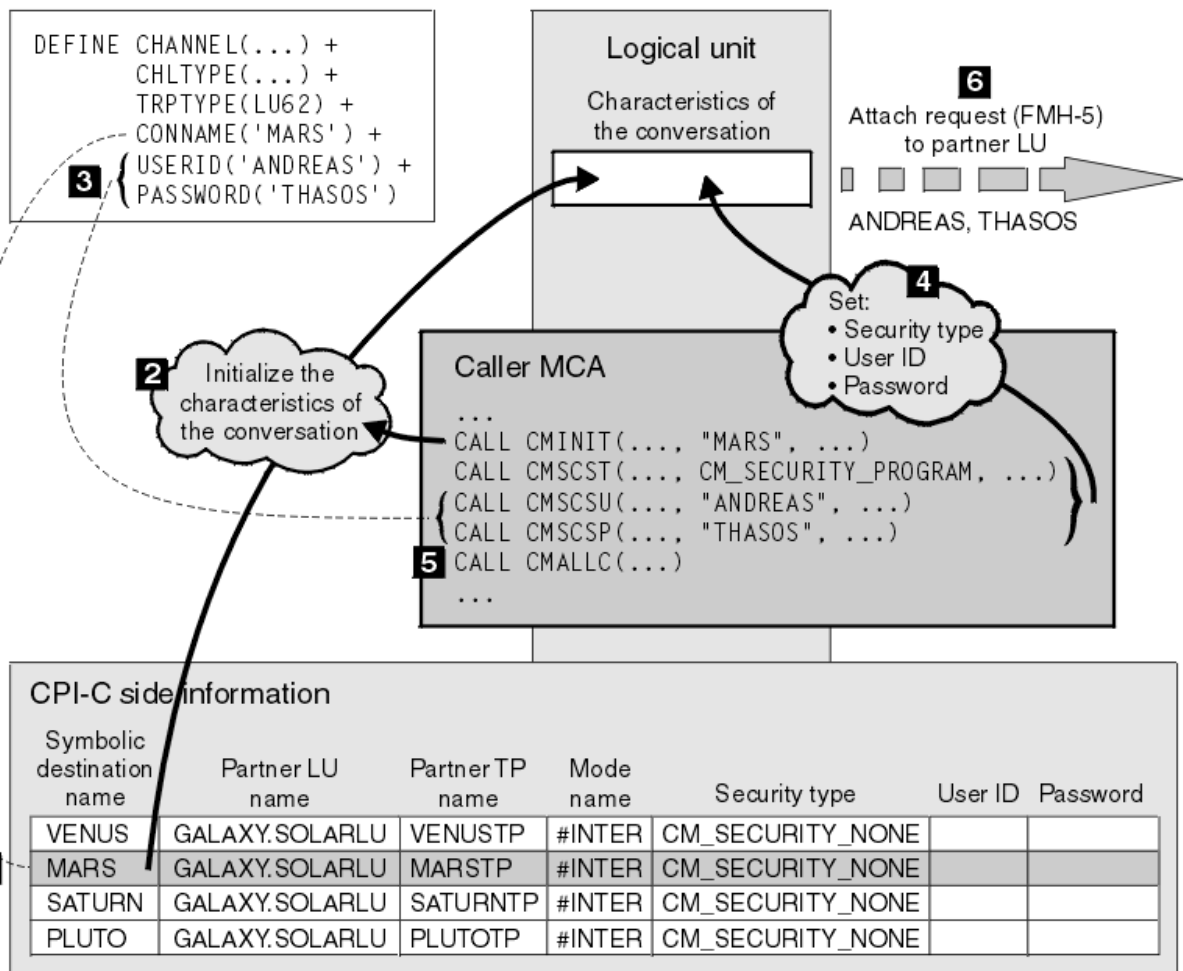


Figura 13. Supporto IBM MQ per l'autenticazione del livello di conversazione

Su IBM i, UNIXe 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.

#### *Autenticazione a livello di conversazione e IBM MQ for z/OS*

Utilizzare questo argomento per ottenere una panoramica su come funziona l'autenticazione del livello di conversazione, su z/OS.

Su IBM MQ for z/OS, gli MCA non utilizzano CPI-C. Invece, usano APPC/MVS TP Conversation Callable Services, un'implementazione di APPC (Advanced Program - to - Program Communication), che ha alcune funzionalità CPI-C. Quando un MCA chiamante assegna una conversazione, viene specificato un tipo di sicurezza SAME nella chiamata. Pertanto, poiché una LU APPC/MVS supporta la verifica persistente solo per conversazioni in entrata, non per conversazioni in uscita, esistono due possibilità:

- Se la LU partner considera attendibile la LU APPC/MVS e accetterà un ID utente già verificato, la LU APPC/MVS invia una richiesta di collegamento contenente:
  - L'ID utente dello spazio di indirizzo dell'iniziatore di canali
  - Un nome profilo di sicurezza, che, se viene utilizzato RACF, è il nome del gruppo di connessione corrente dell'ID utente dello spazio di indirizzo dell'iniziatore di canali
  - Un indicatore già verificato
- Se la LU partner non considera attendibile la LU APPC/MVS e non accetterà un ID utente già verificato, la LU APPC/MVS invia una richiesta di collegamento che non contiene informazioni di sicurezza.



Su IBM MQ for z/OS, i parametri USERID e PASSWORD nel comando DEFINE CHANNEL non possono essere utilizzati per un canale di messaggi e sono validi solo all'estremità di connessione client di un canale MQI. Pertanto, una richiesta di collegamento da una LU APPC / MVS non contiene mai i valori specificati da questi parametri.

## 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 MQ decide a quale gestore code inviarlo. Quando molte applicazioni inviano messaggi alla coda del cluster, IBM MQ bilancia il carico di lavoro su ciascuno dei gestori code che hanno un'istanza della coda. Se uno dei sistemi che ospitano un gestore code clonato ha esito negativo, IBM MQ continua a bilanciare il carico di lavoro tra i gestori code rimanenti 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](#).

 Per considerazioni specifiche su IBM MQ for z/OS, consultare [“Sicurezza nei cluster del gestore code su z/OS”](#) a pagina 256.

### Attività correlate

[“Come impedire ai gestori code di ricevere messaggi”](#) a pagina 436

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 MQ

Vi sono ulteriori considerazioni sulla sicurezza se si utilizza IBM 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 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 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 MQ Multicast.

IBM 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 MQ (ad esempio, applicazioni LLM, consultare [Interoperabilità multicast con IBM 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 MQ è basato sugli ID utente. Per ulteriori informazioni su questo argomento, consultare [“Controllo accessi per client”](#) a pagina 88.

### 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 pubblici 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, potresti dover bloccare temporaneamente gli indirizzi IP all'interno di IBM MQ, forse mentre attendi 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 373.

## Sicurezza per IBM MQ Internet pass - thru

Internet pass - thru può semplificare la comunicazione attraverso un firewall, ma questo ha implicazioni di sicurezza.

IBM MQ internet pass - thru è un'estensione del prodotto di base IBM MQ fornita in SupportPac MS81.

IBM MQ Internet pass - thru consente a due gestori code di scambiare messaggi o a un'applicazione client IBM MQ di 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 dei flussi del protocollo del canale IBM MQ all'interno e all'esterno di un firewall più semplice e gestibile eseguendo il tunnelling dei flussi all'interno di HTTP o agendo come un proxy. Utilizzando TLS (Transport Layer Security), può essere utilizzato anche per codificare e decodificare i messaggi inviati su Internet.

Quando il sistema IBM MQ comunica con IPT, a meno che non si stia utilizzando SSLProxyMode in IPT, assicurarsi che CipherSpec utilizzato da IBM MQ corrisponda a CipherSuite utilizzato da IPT:

- Quando IPT funge da server TLS e IBM MQ si connette come client TLS, la CipherSpec utilizzata da IBM MQ deve corrispondere a una CipherSuite abilitata nel keyring IPT pertinente.
- Quando IPT agisce come client TLS e si connette a un server TLS IBM MQ, la CipherSuite IPT deve corrispondere alla CipherSpec definita sul canale IBM MQ di ricezione.

Se si esegue la migrazione da IPT al supporto TLS IBM MQ integrato, trasferire i certificati digitali da IPT utilizzando `mqiPTKeyman` o `mqiPTKeycmd`.

Per ulteriori informazioni, consultare [IBM MQ Internet Pass-Thru \(SupportPac MS81\)](#).

## Elenco di controllo dell'implementazione della sicurezza IBM MQ for z/OS

Questo argomento fornisce una procedura dettagliata che è possibile utilizzare per elaborare e definire l'implementazione della sicurezza per ciascun gestore code IBM MQ.

RACF fornisce le definizioni per le classi di sicurezza IBM MQ nella relativa CDT (Class Descriptor Table) statica fornita. Mentre si utilizza l'elenco di controllo, è possibile determinare quali di queste classi sono richieste dall'impostazione. È necessario assicurarsi che siano attivati come descritto in [“Classi di sicurezza RACF”](#) a pagina 177.

Fare riferimento ad altre sezioni per dettagli, in particolare [“Profili utilizzati per controllare l'accesso alle risorse IBM MQ”](#) a pagina 187.

Se si richiede il controllo di sicurezza, seguire questo elenco di controllo per implementarlo:

1. Attivare la classe RACF MQADMIN (profili in maiuscolo) o MXADMIN (profili in maiuscolo e minuscolo).

- Si desidera la sicurezza a livello di gruppo di condivisione code, a livello di gestore code o a una combinazione di entrambi?

Consultare [“Profili per controllare il gruppo di condivisione code o la sicurezza a livello di gestore code”](#) a pagina 182.

## 2. Hai bisogno di sicurezza di connessione?

- **Si:** attivare la classe MQCONN. Definire profili di connessione appropriati a livello di gestore code o a livello di gruppo di condivisione code nella classe MQCONN. Quindi, consentire agli utenti o ai gruppi appropriati l'accesso a questi profili.

**Nota:** Solo gli utenti della richiesta API MQCONN o gli ID utente dello spazio di indirizzi CICS o IMS devono avere accesso al profilo di connessione corrispondente.

- **No:** definire un hlq.NO.CONNECT.CHECKS a livello di gestore code o di gruppo di condivisione code nella classe MQADMIN o MXADMIN.

## 3. Hai bisogno di controlli di sicurezza sui comandi?

- **Si:** attivare la classe MQCMD5. Definire i profili di comando appropriati a livello di gestore code o di gruppo di condivisione code nella classe MQCMD5. Quindi, consentire agli utenti o ai gruppi appropriati l'accesso a questi profili.

Se si utilizza un gruppo di condivisione code, potrebbe essere necessario includere gli ID utente utilizzati dal gestore code stesso e dall'iniziatore del canale. Consultare [“Impostazione della sicurezza delle risorse IBM MQ for z/OS”](#) a pagina 247.

- **No:** definire un hlq.NO.CMD.CHECKS per il gestore code o il gruppo di condivisione code richiesto nella classe MQADMIN o MXADMIN.

## 4. Hai bisogno di sicurezza sulle risorse utilizzate nei comandi?

- **Si:** assicurarsi che la classe MQADMIN o MXADMIN sia attiva. Definire profili appropriati per la protezione delle risorse sui comandi a livello di gestore code o a livello di gruppo di condivisione code nella classe MQADMIN o MXADMIN. Quindi, consentire agli utenti o ai gruppi appropriati l'accesso a questi profili. Impostare il parametro CMDUSER in CSQ6SYSP sull'ID utente predefinito da utilizzare per i controlli di sicurezza dei comandi.

Se si utilizza un gruppo di condivisione code, potrebbe essere necessario includere gli ID utente utilizzati dal gestore code stesso e dall'iniziatore del canale. Consultare [“Impostazione della sicurezza delle risorse IBM MQ for z/OS”](#) a pagina 247.

- **No:** definire un hlq.NO.CMD.RESC.CHECKS per il gestore code o il gruppo di condivisione code richiesto nella classe MQADMIN o MXADMIN.

## 5. Hai bisogno della sicurezza della coda?

- **Si:** attivare la classe MQQUEUE o MXQUEUE. Definire i profili di coda appropriati per il gestore code o il gruppo di condivisione code richiesto in MQQUEUE o MXQUEUEclass. Quindi, consentire agli utenti o ai gruppi appropriati l'accesso a questi profili.
- **No:** definire un hlq.NO.QUEUE.CHECKS per il gestore code o il gruppo di condivisione code richiesto nella classe MQADMIN o MXADMIN.

## 6. Hai bisogno di sicurezza di processo?

- **Si:** attivare la classe MQPROC o MXPROC. Definire i profili di processo appropriati a livello di gestore code o di gruppo di condivisione code e consentire agli utenti o ai gruppi appropriati l'accesso a tali profili.
- **No:** definire un hlq.NO.PROCESS.CHECKS per il gestore code o il gruppo di condivisione code appropriato nella classe MXADMIN o MQADMIN.

## 7. Hai bisogno della sicurezza dell'elenco nomi?

- **Si:** attivare MQNLIST o MXNLISTclass. Definire profili di elenco nomi appropriati a livello di gestore code o a livello di gruppo di condivisione code nella classe MQNLIST o MXNLIST. Quindi, consentire agli utenti o ai gruppi appropriati l'accesso a questi profili.

- **No:** definire un hlq.NO.NLIST.CHECKS per il gestore code o il gruppo di condivisione code richiesto nella classe MQADMIN o MXADMIN.
8. Hai bisogno di sicurezza per gli argomenti?
- **Sì:** attivare la classe MXTOPIC. Definire i profili argomento appropriati a livello di gestore code o a livello di gruppo di condivisione code nella classe MXTOPIC. Quindi, consentire agli utenti o ai gruppi appropriati l'accesso a questi profili.
  - **No:** definire un hlq.NO.TOPIC.CHECKS per il gestore code o il gruppo di condivisione code richiesto nella classe MQADMIN o MXADMIN.
9. Gli utenti devono proteggere l'utilizzo delle opzioni MQOPEN o MQPUT1 relative all'utilizzo del contesto?
- **Sì:** assicurarsi che la classe MQADMIN o MXADMIN sia attiva. Definire i profili hlq.CONTEXT.queueName a livello di coda, gestore code o gruppo di condivisione code nella classe MQADMIN o MXADMIN. Quindi, consentire agli utenti o ai gruppi appropriati l'accesso a questi profili.
  - **No:** definire un hlq.NO.CONTEXT.CHECKS per il gestore code o il gruppo di condivisione code richiesto nella classe MQADMIN o MXADMIN.
10. È necessario proteggere l'utilizzo di ID utente alternativi?
- **Sì:** assicurarsi che la classe MQADMIN o MXADMIN sia attiva. Definire il hlq.ALTERNATE.USER. I profili *alternateuserid* per il gestore code o il gruppo di condivisione code richiesto e consentono agli utenti o ai gruppi richiesti di accedere a tali profili.
  - **No:** definire il profilo hlq.NO.ALTERNATE.USER.CHECKS per il gestore code o per il gruppo di condivisione code richiesto nella classe MQADMIN o MXADMIN.
11. È necessario personalizzare gli ID utente da utilizzare per i controlli di sicurezza delle risorse tramite RESLEVEL?
- **Sì:** assicurarsi che la classe MQADMIN o MXADMIN sia attiva. Definire un profilo hlq.RESLEVEL a livello di gestore code o a livello di gruppo di condivisione code nella classe MQADMIN o MXADMIN. Quindi, consentire agli utenti o ai gruppi richiesti l'accesso al profilo.
  - **No:** verificare che non esista alcun profilo generico nella classe MQADMIN o MXADMIN che possa essere applicato a hlq.RESLEVEL. Definire un profilo hlq.RESLEVEL per il gestore code o il gruppo di condivisione code richiesto e assicurarsi che nessun utente o gruppo abbia accesso ad esso.
12. È necessario 'timeout' degli ID utente non utilizzati da IBM MQ ?
- **Sì:** determinare i valori di supero tempo che si desidera utilizzare ed emettere il comando MQSC ALTER SECURITY per modificare i parametri TIMEOUT e INTERVAL.
  - **No:** immettere il comando MQSC ALTER SECURITY per impostare il valore INTERVAL su zero.
- Nota:** Aggiornare il dataset di input di inizializzazione CSQINP1 utilizzato dal sottosistema in modo che il comando MQSC ALTER SECURITY venga emesso automaticamente all'avvio del gestore code.
13. Si utilizza l'accodamento distribuito?
- **Sì:** utilizzare i record di autenticazione di canale. Per ulteriori informazioni, consultare [“Record di autenticazione di canale”](#) a pagina 47.
  - È anche possibile determinare il valore dell'attributo MCAUSER appropriato per ogni canale o fornire uscite di sicurezza del canale adatte.
14. Si desidera utilizzare TLS (Transport Layer Security)?
- **Sì:** per specificare che qualsiasi utente che presenta un certificato personale 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.
  - Pianificare l'infrastruttura TLS. Installare la funzione SSL di sistema di z/OS. In RACF, configura i tuoi CNFs (Certificate Name Filters), se li stai utilizzando, e i tuoi certificati digitali. Impostare il

keyring SSL. Verificare che l'attributo del gestore code SSLKEYR non sia vuoto e che punti al keyring SSL. Assicurarsi inoltre che il valore di SSLTASKS sia almeno 2.

- **No:** assicurarsi che SSLKEYR sia vuoto e che SSLTASKS sia zero.

Per ulteriori dettagli su TLS, consultare [“Protocolli di sicurezza TLS in IBM MQ”](#) a pagina 24.

#### 15. Usi i clienti?

- **Si:** utilizzare i record di autenticazione di canale.
- È anche possibile determinare il valore dell'attributo MCAUSER appropriato per ogni canale di connessione server o fornire uscite di sicurezza del canale appropriate, se necessario.

#### 16. Controllare le impostazioni dello switch.

IBM MQ emette messaggi all'avvio del gestore code che visualizzano le proprie impostazioni di sicurezza. Utilizzare questi messaggi per determinare se gli switch sono impostati correttamente.

#### 17. Si inviano password dalle applicazioni client?

- **Si:** assicurarsi che la funzione z/OS sia installata e che ICSF (Integrated Cryptographic Service Facility) sia avviato per una protezione ottimale.
- **No:** è possibile ignorare il messaggio di errore che indica che ICSF non è stato avviato.

Per ulteriori informazioni su ICSF, consultare [“Utilizzo di ICSF \(Integrated Cryptographic Service Facility\)”](#) a pagina 256

## Configurazione della sicurezza

---

Questa raccolta di argomenti contiene informazioni specifiche per i diversi sistemi operativi e per l'utilizzo dei client.

### ULW

## Impostazione della sicurezza su UNIX, Linux, and Windows

Considerazioni sulla sicurezza specifiche dei sistemi UNIX, Linux, and Windows .

I gestori code IBM 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 MQ

È possibile definire la serie di utenti che possono immettere comandi per amministrare IBM MQ.

### Chi può utilizzare gli oggetti IBM 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 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 MQ; non utilizzare i comandi del sistema operativo standard per concedere o revocare le autorizzazioni alle risorse MQI.

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.

**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 effettuato sul nome dell'oggetto che si sta aprendo e non sul nome o sui nomi, risultante dalla 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 10 a pagina 120, Tabella 11 a pagina 120, Tabella 12 a pagina 121 e Tabella 13 a pagina 121 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.

<i>Tabella 10. Autorizzazione di sicurezza necessaria per chiamate MQCONN</i>			
<b>Autorizzazione richiesta per:</b>	<b>Oggetto coda ( “1” a pagina 121 )</b>	<b>Oggetto processo</b>	<b>Oggetto gestore code</b>
<b>MQCONN</b>	Non applicabile	Non applicabile	CONNECT MQZAO_

<i>Tabella 11. Autorizzazione di sicurezza necessaria per chiamate MQOPEN</i>			
<b>Autorizzazione richiesta per:</b>	<b>Oggetto coda ( “1” a pagina 121 )</b>	<b>Oggetto processo</b>	<b>Oggetto gestore code</b>
MQOO_INQUIRE	INQUIRE MQZAO_	INQUIRE MQZAO_	INQUIRE MQZAO_
MQOO_SFOGLIA	MQZAO_BROWSE	Non applicabile	Nessun controllo
MQOO_INPUT_*	INPUT MQZAO_	Non applicabile	Nessun controllo
MQOO_SAVE_ALL_CONTEXT ( “2” a pagina 121 )	INPUT MQZAO_	Non applicabile	Non applicabile
MQOO_OUTPUT (Coda normale) ( “3” a pagina 121 )	OUTPUT MQZAO_	Non applicabile	Non applicabile
MQOO_PASS_IDENTITY_CONTEXT ( “4” a pagina 121 )	MQZAO_PASS_CONTESTO_IDENTITÀXX_ENCODE_CASE_CAPS_LOCK_OFF	Non applicabile	Nessun controllo
MQOO_PASS_ALL_CONTEXT ( “4” a pagina 121, “5” a pagina 121 )	MQZAO_PASS_TUTTO_CONTESTO	Non applicabile	Nessun controllo
MQOO_SET_IDENTITY_CONTEXT ( “4” a pagina 121, “5” a pagina 121 )	MQZAO_SET_CONTESTO_IDENTITÀ	Non applicabile	MQZAO_SET_CONTESTO_IDENTITÀ ( “6” a pagina 121 )
MQOO_SET_ALL_CONTEXT ( “4” a pagina 121, “7” a pagina 121 )	MQZAO_SET_TUTTO_CONTESTO	Non applicabile	MQZAO_SET_ALL_CONTEXT ( “6” a pagina 121 )
MQOO_OUTPUT (Coda di trasmissione) ( “8” a pagina 122 )	MQZAO_SET_TUTTO_CONTESTO	Non applicabile	MQZAO_SET_ALL_CONTEXT ( “6” a pagina 121 )
SET MQOO	MQZAO_SET	Non applicabile	Nessun controllo
MQOO_ALTERNATE_AUTORITÀ_UTENTE	( “9” a pagina 122 )	( “9” a pagina 122 )	MQZAO_ALTERNATE_USER_AUTHORITY ( “9” a pagina 122, “10” a pagina 122 )



<i>Tabella 12. Autorizzazione di sicurezza necessaria per le chiamate MQPUT1</i>			
<b>Autorizzazione richiesta per:</b>	<b>Oggetto coda ( “1” a pagina 121 )</b>	<b>Oggetto processo</b>	<b>Oggetto gestore code</b>
MQPMO_PASS_CONTESTO_IDENTITÀ	MQZAO_PASS_IDENTITY_CONTEXT ( “11” a pagina 122 )	Non applicabile	Nessun controllo
MQPMO_PASS_ALL_CONTESTO	MQZAO_PASS_ALL_CONTEXT ( “11” a pagina 122 )	Non applicabile	Nessun controllo
MQPMO_SET_CONTESTO_IDENTITÀ	MQZAO_SET_CONTESTO_IDENTITÀ ( “11” a pagina 122 )	Non applicabile	MQZAO_SET_CONTESTO_IDENTITÀ ( “6” a pagina 121 )
MQPMO_SET_TUTTO_CONTESTO	MQZAO_SET_ALL_CONTEXT ( “11” a pagina 122 )	Non applicabile	MQZAO_SET_ALL_CONTEXT ( “6” a pagina 121 )
(Coda di trasmissione) ( “8” a pagina 122 )	MQZAO_SET_TUTTO_CONTESTO	Non applicabile	MQZAO_SET_ALL_CONTEXT ( “6” a pagina 121 )
AUTORIZZAZIONE_UTENTE_MQPMO_ALTERNATE_	( “12” a pagina 122 )	Non applicabile	MQZAO_ALTERNATE_USER_AUTHORITY ( “10” a pagina 122 )

<i>Tabella 13. Autorizzazione di sicurezza necessaria per le chiamate MQCLOSE</i>			
<b>Autorizzazione richiesta per:</b>	<b>Oggetto coda ( “1” a pagina 121 )</b>	<b>Oggetto processo</b>	<b>Oggetto gestore code</b>
MQCO_DELETE	MQZAO_DELETE ( “13” a pagina 122 )	Non applicabile	Non applicabile
MQCO_DELETE_PURGE	MQZAO_DELETE ( “13” a pagina 122 )	Non applicabile	Non applicabile

**Note per le tabelle:**

- 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.
- È necessario specificare anche MQOO\_INPUT\_\*. È valido per una coda locale, modello o alias.
- Questo controllo viene eseguito per tutti i casi di output, tranne le code di trasmissione (consultare la nota “8” a pagina 122).
- È necessario specificare anche MQOO\_OUTPUT.
- MQOO\_PASS\_IDENTITY\_CONTEXT è implicito anche da questa opzione.
- Questa autorizzazione è richiesta sia per l'oggetto gestore code che per la particolare coda.
- 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 entrambe le seguenti istruzioni sono vere:
  - 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.

### **ULW** *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
Canale connessione client	Non applicabile
Listener	Non applicabile
Servizio	Non applicabile
Informazioni di comunicazione	Non applicabile

**DEFINE oggetto NOREPLACE ( “1” a pagina 127 )**

Oggetto	Autorizzazione richiesta
Coda	<b>MQZAO_CREATE ( “2” a pagina 127 )</b>
Argomento	<b>MQZAO_CREATE ( “2” a pagina 127 )</b>
Processo	<b>MQZAO_CREATE ( “2” a pagina 127 )</b>
Gestore code	Non applicabile
Elenco nomi	<b>MQZAO_CREATE ( “2” a pagina 127 )</b>
Informazioni di autenticazione	<b>MQZAO_CREATE ( “2” a pagina 127 )</b>
Canale	<b>MQZAO_CREATE ( “2” a pagina 127 )</b>
Canale connessione client	<b>MQZAO_CREATE ( “2” a pagina 127 )</b>
Listener	<b>MQZAO_CREATE ( “2” a pagina 127 )</b>
Servizio	<b>MQZAO_CREATE ( “2” a pagina 127 )</b>
Informazioni di comunicazione	<b>MQZAO_CREATE ( “2” a pagina 127 )</b>

**DEFINE oggetto REPLACE ( “1” a pagina 127, “3” a pagina 127 )**

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

Oggetto	Autorizzazione richiesta
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
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

Oggetto	Autorizzazione richiesta
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

### 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_
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_

Comando	Oggetto	Autorizzazione richiesta
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 omissso.
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](#)

**ULW** **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
<u>Coda</u>	MODIFICA_MQZO
<u>Argomento</u>	MODIFICA_MQZO
<u>Processo</u>	MODIFICA_MQZO
<u>Gestore code</u>	MODIFICA_MQZO
<u>Elenco nomi</u>	MODIFICA_MQZO
<u>Informazioni di autenticazione</u>	MODIFICA_MQZO
<u>Canale</u>	MODIFICA_MQZO
<u>Canale connessione client</u>	MODIFICA_MQZO
<u>Listener</u>	MODIFICA_MQZO
<u>Servizio</u>	MODIFICA_MQZO
<u>Informazioni di comunicazione</u>	MODIFICA_MQZO

**Cancella oggetto**

Oggetto	Autorizzazione richiesta
<u>Coda</u>	CLEAR MQZAO_

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
<u>Argomento</u>	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
Listener	Non applicabile
Servizio	Non applicabile
Informazioni di comunicazione	Non applicabile

**Copia oggetto (senza sostituzione) ( 1 )**

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
Coda	MQZAO_CREATE ( <b>2</b> )
<u>Argomento</u>	MQZAO_CREATE ( <b>2</b> )
<u>Processo</u>	MQZAO_CREATE ( <b>2</b> )
Gestore code	Non applicabile
<u>Elenco nomi</u>	MQZAO_CREATE ( <b>2</b> )
<u>Informazioni di autenticazione</u>	MQZAO_CREATE ( <b>2</b> )
<u>Canale</u>	MQZAO_CREATE ( <b>2</b> )
<u>Canale connessione client</u>	MQZAO_CREATE ( <b>2</b> )
<u>Listener</u>	MQZAO_CREATE ( <b>2</b> )
<u>Servizio</u>	MQZAO_CREATE ( <b>2</b> )
<u>Informazioni di comunicazione</u>	<b>MQZAO_CREATE ( “2” a pagina 133 )</b>

**Copia oggetto (con sostituzione) ( 1, 4 )**

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
Coda	MODIFICA_MQZO
<u>Argomento</u>	MODIFICA_MQZO
<u>Processo</u>	MODIFICA_MQZO
Gestore code	Non applicabile
<u>Elenco nomi</u>	MODIFICA_MQZO
<u>Informazioni di autenticazione</u>	MODIFICA_MQZO
<u>Canale</u>	MODIFICA_MQZO
<u>Canale connessione client</u>	MODIFICA_MQZO
<u>Listener</u>	MODIFICA_MQZO



<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
<u>Servizio</u>	MODIFICA_MQZO
<u>Informazioni di comunicazione</u>	MODIFICA_MQZO

**Crea oggetto (senza sostituzione) ( 3 )**

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
<u>Coda</u>	MQZAO_CREATE ( <u>2</u> )
<u>Argomento</u>	MQZAO_CREATE ( <u>2</u> )
<u>Processo</u>	MQZAO_CREATE ( <u>2</u> )
<u>Gestore code</u>	Non applicabile
<u>Elenco nomi</u>	MQZAO_CREATE ( <u>2</u> )
<u>Informazioni di autenticazione</u>	MQZAO_CREATE ( <u>2</u> )
<u>Canale</u>	MQZAO_CREATE ( <u>2</u> )
<u>Canale connessione client</u>	MQZAO_CREATE ( <u>2</u> )
<u>Listener</u>	MQZAO_CREATE ( <u>2</u> )
<u>Servizio</u>	MQZAO_CREATE ( <u>2</u> )
<u>Informazioni di comunicazione</u>	MQZAO_CREATE ( <u>2</u> )

**Crea oggetto (con sostituzione) ( 3, 4 )**

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
<u>Coda</u>	MODIFICA_MQZO
<u>Argomento</u>	MODIFICA_MQZO
<u>Processo</u>	MODIFICA_MQZO
<u>Gestore code</u>	Non applicabile
<u>Elenco nomi</u>	MODIFICA_MQZO
<u>Informazioni di autenticazione</u>	MODIFICA_MQZO
<u>Canale</u>	MODIFICA_MQZO
<u>Canale connessione client</u>	MODIFICA_MQZO
<u>Listener</u>	MODIFICA_MQZO
<u>Servizio</u>	MODIFICA_MQZO
<u>Informazioni di comunicazione</u>	MODIFICA_MQZO

**Elimina oggetto**

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
<u>Coda</u>	MQZAO_DELETE
<u>Argomento</u>	MQZAO_DELETE
<u>Processo</u>	MQZAO_DELETE
<u>Gestore code</u>	Non applicabile

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
<u>Elenco nomi</u>	MQZAO_DELETE
<u>Informazioni di autenticazione</u>	MQZAO_DELETE
<u>Canale</u>	MQZAO_DELETE
<u>Canale connessione client</u>	MQZAO_DELETE
<u>Listener</u>	MQZAO_DELETE
<u>Servizio</u>	MQZAO_DELETE
<u>Informazioni di comunicazione</u>	MQZAO_DELETE

### **Interrogazione *oggetto***

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
<u>Coda</u>	DISPLAY MQZAO_
<u>Argomento</u>	DISPLAY MQZAO_
<u>Processo</u>	DISPLAY MQZAO_
<u>Gestore code</u>	DISPLAY MQZAO_
<u>Elenco nomi</u>	DISPLAY MQZAO_
<u>Informazioni di autenticazione</u>	DISPLAY MQZAO_
<u>Canale</u>	DISPLAY MQZAO_
<u>Canale connessione client</u>	DISPLAY MQZAO_
<u>Listener</u>	DISPLAY MQZAO_
<u>Servizio</u>	DISPLAY MQZAO_
<u>Informazioni di comunicazione</u>	DISPLAY MQZAO_

### **Interroga nomi *oggetto***

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
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 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
<u>Canale</u>	CONTROL MQZAO_
Canale connessione client	Non applicabile
<u>Listener</u>	CONTROL MQZAO_
<u>Servizio</u>	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
<u>Canale</u>	CONTROL MQZAO_
Canale connessione client	Non applicabile
<u>Listener</u>	CONTROL MQZAO_
<u>Servizio</u>	CONTROL MQZAO_
Informazioni di comunicazione	Non applicabile

### Comandi per i canali

Comando	Oggetto	Autorizzazione richiesta
<u>Ping canale</u>	Canale	CONTROL MQZAO_
<u>Reimposta canale</u>	Canale	MQZAO_CONTROL_XX_ENCODING_CASE_ONE uscita
<u>Risolvi canale</u>	Canale	MQZAO_CONTROL_XX_ENCODING_CASE_ONE uscita

### Comandi sottoscrizione

Comando	Oggetto	Autorizzazione richiesta
<u>Modifica sottoscrizione</u>	Argomento	CONTROL MQZAO_

<b>Comando</b>	<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
<u>Crea sottoscrizione</u>	Argomento	CONTROL MQZAO_
<u>Elimina sottoscrizione</u>	Argomento	CONTROL MQZAO_
<u>Interroga sottoscrizione</u>	Argomento	DISPLAY MQZAO_

### Comandi sicurezza

<b>Comando</b>	<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
<u>Imposta record di autorizzazione</u>	Gestore code	MODIFICA_MQZO
<u>Eliminare il record di autorizzazione</u>	Gestore code	MODIFICA_MQZO
<u>Interrogazione record autorizzazione</u>	Gestore code	DISPLAY MQZAO_
<u>Interrogazione servizio autorizzazione</u>	Gestore code	DISPLAY MQZAO_
<u>Richiedi autorità entità</u>	Gestore code	DISPLAY MQZAO_
<u>Imposta record autenticazione canale</u>	Gestore code	MODIFICA_MQZO
<u>Interroga record autenticazione canale</u>	Gestore code	DISPLAY MQZAO_
<u>Aggiornamento sicurezza</u>	Gestore code	MODIFICA_MQZO

### Visualizzazioni stato

<b>Comando</b>	<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
<u>Interrogazione stato canale</u>	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.
<u>Richiedi stato listener canale</u>	Gestore code	DISPLAY MQZAO_
<u>Interroga stato pubblicazione / sottoscrizione</u>	Gestore code	DISPLAY MQZAO_
<u>Richiedi stato sottoscrizione</u>	Gestore code	DISPLAY MQZAO_
<u>Interrogazione stato servizio</u>	Gestore code	DISPLAY MQZAO_
<u>Interroga stato argomento</u>	Gestore code	DISPLAY MQZAO_

### Comandi per i Cluster

<b>Comando</b>	<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
<u>Interrogazione gestore code cluster</u>	Gestore code	DISPLAY MQZAO_

Comando	Oggetto	Autorizzazione richiesta
<u>Aggiornamento cluster</u>	È richiesta l'appartenenza al gruppo 'mqm'	È richiesta l'appartenenza al gruppo 'mqm'
<u>Reimposta cluster</u>	È richiesta l'appartenenza al gruppo 'mqm'	È richiesta l'appartenenza al gruppo 'mqm'
<u>Sospensione cluster gestore code</u>	È richiesta l'appartenenza al gruppo 'mqm'	È richiesta l'appartenenza al gruppo 'mqm'
<u>Ripristino cluster gestore code</u>	È richiesta l'appartenenza al gruppo 'mqm'	È richiesta l'appartenenza al gruppo 'mqm'

### Altri comandi di gestione

Comando	Oggetto	Autorizzazione richiesta
<u>Ping gestore code</u>	Gestore code	DISPLAY MQZAO_
<u>Aggiornamento gestore code</u>	Gestore code	MODIFICA_MQZO
<u>Reimpostazione gestore code</u>	Gestore code	MODIFICA_MQZO
<u>Reimposta statistiche coda</u>	Coda	MQZAO_DISPLAY e MQZAO_CHANGE
<u>Interrogazione connessione</u>	Gestore code	DISPLAY MQZAO_
<u>Arresta connessione</u>	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.

## 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 su AIX

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 su AIX**

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 degli utenti in un gruppo su AIX**

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 su AIX**

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 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 degli utenti 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

Linux

### 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.

Linux

### 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.

Linux

### 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 `groupae` deve diventare anche membro di `groupb`, viene utilizzato il seguente comando: `usermod -G groupa,groupb user-name`, dove `user-name` è il nome utente.

Linux

### 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.

Linux

### Rimozione di un utente da un gruppo su Linux

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.

Solaris

### 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.



## **Solaris** 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.

## **Solaris** 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.

## **Solaris** Visualizzazione degli utenti in un gruppo su Solaris

Visualizzare chi si trova in un gruppo esaminando il file `/etc/group` .

### Procedura

- Per rilevare chi è un membro di un gruppo, esaminare la voce per tale gruppo nel file `/etc/group` .

## **Solaris** 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.

## **Windows** Creazione e gestione di gruppi su 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 Server 2008 e Windows Server 2012

Accedere a questo pannello utilizzando **Pannello di controllo > Sistema e manutenzione > Strumenti di amministrazione > Gestione computer.**

## Windows 7 e Windows 8.1

Accedere a questo pannello utilizzando **Strumenti di amministrazione > Gestione computer**

### **Windows** Creazione di un gruppo su 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. Fare clic su **Chiudi.**

### **Windows** 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.

### **Windows** *Visualizzazione degli utenti 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.

### **Windows** *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.

### **Windows** *Considerazioni speciali per la sicurezza su Windows*

Alcune funzioni di sicurezza si comportano in modo diverso su versioni differenti di Windows.

La sicurezza IBM MQ si basa sulle chiamate all'API del sistema operativo per informazioni sulle autorizzazioni utente e sulle appartenenze ai gruppi. Alcune funzioni non si comportano in modo

identico sui sistemi Windows . Questa raccolta di argomenti include le descrizioni di come tali differenze potrebbero influire sulla sicurezza IBM MQ quando si esegue IBM MQ in un ambiente Windows .

## **Windows Account utente locali e di dominio per il servizio IBM MQ Windows**

Quando IBM MQ è in esecuzione, deve verificare che solo gli utenti autorizzati possano accedere alle code o ai gestori code. Ciò richiede un account utente speciale che IBM MQ può utilizzare per eseguire una query delle informazioni relative a qualsiasi utente che tenta tale accesso.

- [“Configurazione di account utente speciali con Prepare IBM MQ Wizard” a pagina 140](#)
- [“Utilizzo di IBM MQ con Active Directory” a pagina 140](#)
- [“Diritti utente richiesti per un servizio IBM MQ Windows” a pagina 141](#)

### **Configurazione di account utente speciali con Prepare IBM MQ Wizard**

Prepare IBM MQ Wizard crea un account utente speciale in modo che il servizio Windows possa essere condiviso dai processi che devono utilizzarlo (vedi [Configurazione di IBM MQ con la procedura guidata Prepara IBM MQ](#)).

Un servizio Windows viene condiviso tra processi client per un'installazione IBM MQ . Viene creato un servizio per ciascuna installazione. Ogni servizio è denominato `MQ_InstallationName` e ha un nome di visualizzazione IBM MQ (`InstallationName`).

Poiché ogni servizio deve essere condiviso tra sessioni di accesso interattivo e non interattivo, è necessario avviarle con un account utente speciale. È possibile utilizzare un account utente speciale per tutti i servizi oppure creare diversi account utente speciali. Ogni account utente speciale deve avere il diritto utente su `Accedi come servizio`, per ulteriori informazioni consultare [Tabella 14 a pagina 141](#). Se l'ID utente non dispone dell'autorizzazione per eseguire il servizio, il servizio non viene avviato e restituisce un errore nel log eventi del sistema Windows . In genere, è stato eseguito Prepare IBM MQ Wizard e l'ID utente è stato configurato correttamente. Tuttavia, se l'ID utente è stato configurato manualmente, è possibile che si sia verificato un problema che sarà necessario risolvere.

Quando si installa IBM MQ e si esegue il Prepare IBM MQ Wizard per la prima volta, viene creato un account utente locale per il servizio denominato `MUSR_MQADMIN` con le impostazioni e le autorizzazioni richieste, incluso `Accesso come servizio`.

Per installazioni successive, il Prepare IBM MQ Wizard crea un account utente denominato `MUSR_MQADMINx`, dove `x` è il successivo numero disponibile che rappresenta un ID utente che non esiste. La password per `MUSR_MQADMINx` viene generata casualmente quando l'account viene creato e utilizzata per configurare l'ambiente di accesso per il servizio. La password generata non scade.

Questo account IBM MQ non è influenzato dalle politiche di account che sono impostate sul sistema per richiedere che le parole d'ordine dell'account vengano modificate dopo un certo periodo di tempo.

La password non è nota all'esterno di questa elaborazione monouso ed è memorizzata dal sistema operativo Windows in una parte sicura del registro.

### **Utilizzo di IBM MQ con Active Directory**

In alcune configurazioni di rete, in cui gli account utente sono definiti su controller di dominio che utilizzano il servizio di directory Active Directory , l'account utente locale con cui viene eseguito IBM MQ potrebbe non disporre dell'autorizzazione richiesta per interrogare l'appartenenza al gruppo di altri account utente del dominio. Quando si installa IBM MQ, Prepare IBM MQ Wizard identifica se questo è il caso eseguendo test e ponendo domande sulla configurazione di rete.

Se l'account utente locale con cui è in esecuzione IBM MQ non dispone dell'autorità richiesta, Prepare IBM MQ Wizard richiede i dettagli dell'account di un account utente del dominio con particolari diritti utente. Per informazioni su come creare e configurare un account di dominio Windows , consulta [Creazione e configurazione di account di dominio Windows per IBM MQ](#). Per i diritti utente richiesti dall'utente del dominio, consultare [Tabella 14 a pagina 141](#).

Una volta immessi i dettagli account validi per l'account utente del dominio in Prepare IBM MQ Wizard, la procedura guidata configura un servizio IBM MQ Windows da eseguire con il nuovo account. I dettagli dell'account sono conservati nella parte protetta del registro e non possono essere letti dagli utenti.

Quando il servizio è in esecuzione, viene avviato un servizio IBM MQ Windows che rimane in esecuzione per tutto il tempo in cui il servizio è in esecuzione. Un amministratore IBM MQ che accede al server dopo l'avvio del servizio Windows può utilizzare IBM MQ Explorer per gestire i gestori code sul server. Ciò connette IBM MQ Explorer al processo del servizio Windows esistente. Queste due azioni hanno bisogno di diversi livelli di autorizzazione prima di poter funzionare:

- Il processo di avvio richiede un'autorizzazione di avvio.
- L'amministratore IBM MQ richiede l'autorizzazione di accesso.

## Diritti utente richiesti per un servizio IBM MQ Windows

La seguente tabella elenca i diritti utente richiesti per gli account utente locali e di dominio con cui viene eseguito il servizio Windows per un'installazione IBM MQ .

<i>Tabella 14. Diritti utente richiesti per un servizio IBM MQ Windows</i>	
<b>Autorizzazione</b>	<b>Descrizione</b>
Accedi come lavoro batch	Abilita un servizio IBM MQ Windows da eseguire con questo account utente.
Accedi come servizio	Consente agli utenti di impostare il servizio IBM MQ Windows per accedere utilizzando l'account configurato.
Arresta il sistema	Consente al servizio IBM MQ Windows di riavviare il server se configurato per farlo quando il ripristino di un servizio ha esito negativo.
Incremento quote	Richiesta per la chiamata <code>CreateProcessAsUser</code> del sistema operativo.
Azione come parte del sistema operativo	Richiesto per la chiamata del sistema operativo <code>LogonUser</code> .
Ignora controllo trasversale	Richiesto per la chiamata del sistema operativo <code>LogonUser</code> .
Sostituzione di un token livello elaborazione	Richiesto per la chiamata del sistema operativo <code>LogonUser</code> .

**Nota:** I diritti dei programmi di debug potrebbero essere richiesti in ambienti su cui sono in esecuzione applicazioni ASP e IIS.

L'account utente del dominio deve avere questi diritti utente Windows impostati come diritti utente effettivi, come elencato nell'applicazione Criteri di sicurezza locali. In caso contrario, impostarli utilizzando l'applicazione Local Security Policy localmente sul server o utilizzando l'applicazione Domain Security Application a livello di dominio.

### **Windows** **Autorizzazioni di sicurezza Windows Server**

L'installazione di IBM MQ si comporta in modo diverso sul server Windows , a seconda che un utente locale o un utente di dominio esegua l'installazione.

Se un utente *locale* installa IBM MQ, Prepare IBM MQ Wizard rileva che l'utente locale creato per il servizio IBM MQ Windows può richiamare le informazioni di appartenenza al gruppo dell'utente di installazione. Prepare IBM MQ Wizard pone domande all'utente sulla configurazione di rete per determinare se ci sono altri account utente definiti sui controller di dominio in esecuzione su Windows 2000 o versioni successive. In caso affermativo, il servizio IBM MQ Windows deve essere eseguito con un

account utente del dominio con impostazioni e autorizzazioni particolari. Prepare IBM MQ Wizard richiede all'utente i dettagli dell'account di questo utente, come descritto in [Configurazione di IBM MQ con la procedura guidata Prepara IBM MQ](#).

Se un utente del *dominio* installa IBM MQ, Prepare IBM MQ Wizard rileva che l'utente locale creato per il servizio IBM MQ Windows non può richiamare le informazioni di appartenenza al gruppo dell'utente di installazione. In questo caso, Prepare IBM MQ Wizard richiede sempre all'utente i dettagli dell'account dell'account utente del dominio per il servizio IBM MQ Windows da utilizzare.

Quando il servizio IBM MQ Windows deve utilizzare un account utente del dominio, IBM MQ non può funzionare correttamente fino a quando non è stato configurato utilizzando Prepare IBM MQ Wizard. Il Prepare IBM MQ Wizard non permette all'utente di continuare con altre attività, fino a quando il servizio Windows non è stato configurato con un account adatto.

Per ulteriori informazioni, vedi [Creazione e configurazione di account di dominio per IBM MQ](#).

### **Windows** *Modifica del nome dell'utente associato con il servizio IBM MQ*

Puoi modificare il nome utente associato al servizio IBM MQ creando un nuovo account e immettendone i dettagli utilizzando Prepare IBM MQ Wizard.

### **Informazioni su questa attività**

Quando si installa IBM MQ e si esegue Prepare IBM MQ Wizard per la prima volta, viene creato un account utente locale per il servizio denominato MUSR\_MQADMIN. Per installazioni successive, il Prepare IBM MQ Wizard crea un account utente denominato MUSR\_MQADMINx, dove x è il successivo numero disponibile che rappresenta un ID utente che non esiste.

Potrebbe essere necessario modificare il nome utente associato al servizio IBM MQ da MUSR\_MQADMIN o MUSR\_MQADMINx a qualcos' altro. Ad esempio, potrebbe essere necessario farlo se il gestore code è associato a Db2, che non accetta nomi utente con più di 8 caratteri.

### **Procedura**

1. Creare un nuovo account utente (ad esempio **NEW\_NAME** )
2. Utilizzare Prepare IBM MQ Wizard per immettere i dettagli del nuovo account utente.

### **Informazioni correlate**

[Configurazione di IBM MQ con la procedura guidata Prepara IBM MQ](#)

### **Windows** *Modifica della password dell'account utente locale del servizio IBM MQ*

#### **Windows**

È possibile modificare la password dell'account utente locale del servizio IBM MQ Windows utilizzando il pannello Gestione computer.

### **Informazioni su questa attività**

Per modificare la password dell'account utente locale del servizio IBM MQ Windows , effettuare le seguenti operazioni:

### **Procedura**

1. Identifica l'utente con cui è in esecuzione il servizio.
2. Arrestare il servizio IBM MQ dal pannello Gestione computer.
3. Modificare la password richiesta nello stesso modo in cui si modificherebbe la password di un individuo.
4. Andare alle proprietà per il servizio IBM MQ dal pannello Gestione computer.
5. Selezionare la pagina **Accesso** .

6. Confermare che il nome account specificato corrisponda all'utente per cui è stata modificata la password.
7. Immettere la password nei campi **Password** e **Conferma password** e fare clic su **OK**.

### **Windows** *Modifica della parola d'ordine per un servizio IBM MQ Windows per un'installazione eseguita con un account utente del dominio*

Come alternativa all'utilizzo di Prepare IBM MQ Wizard per immettere i dettagli dell'account utente del dominio, è possibile utilizzare il pannello Gestione computer per modificare i dettagli **Accesso** per il Servizio IBM MQ specifico dell'installazione.

#### **Informazioni su questa attività**

Se il servizio IBM MQ Windows per un'installazione è in esecuzione in un account utente del dominio, è possibile modificare la password per l'account nel modo seguente:

#### **Procedura**

1. Modificare la password per l'account di dominio sul controller di dominio. Potrebbe essere necessario chiedere all'amministratore del dominio di eseguire questa operazione.
2. Completa la seguente procedura per modificare la pagina di **accesso** per il servizio IBM MQ .
  - a) Identificare l'utente con cui è in esecuzione il servizio.
  - b) Arrestare il servizio IBM MQ dal pannello Gestione computer.
  - c) Modificare la password richiesta nello stesso modo in cui si modificherebbe la password di un individuo.
  - d) Andare alle proprietà per il servizio IBM MQ dal pannello Gestione computer.
  - e) Selezionare la pagina **Accesso** .
  - f) Confermare che il nome account specificato corrisponda all'utente per cui è stata modificata la password.
  - g) Immettere la password nei campi **Password** e **Conferma password** e fare clic su **OK**.

L'account utente con cui viene eseguito il servizio IBM MQ Windows esegue tutti i comandi MQSC emessi dalle applicazioni dell'interfaccia utente o eseguiti automaticamente all'avvio del sistema, all'arresto o al ripristino del servizio. Questo account utente deve quindi disporre dei diritti di amministrazione IBM MQ . Per impostazione predefinita, viene aggiunta al gruppo mq locale sul server. Se questa appartenenza viene rimossa, il servizio IBM MQ Windows non funziona. Per ulteriori informazioni sui diritti utente, consultare [“Diritti utente richiesti per un servizio IBM MQ Windows” a pagina 141](#).

Se si verifica un problema di sicurezza con l'account utente con cui viene eseguito il servizio IBM MQ Windows , i messaggi di errore e le descrizioni vengono visualizzati nel log eventi del sistema.

#### **Informazioni correlate**

[Configurazione di IBM MQ con la procedura guidata Prepara IBM MQ](#)

### **Windows** *Considerazioni sulla promozione dei server Windows ai controller di dominio*

Quando si promuove un server Windows a un controller di dominio, è necessario considerare se l'impostazione di sicurezza relativa alle autorizzazioni utente e gruppo è appropriata. Quando si modifica lo stato di una macchina Windows tra il server e il controller di dominio, è necessario considerare che ciò può influire sul funzionamento di IBM MQ poiché IBM MQ utilizza un gruppo mqm definito localmente.

## Impostazioni di sicurezza relative alle autorizzazioni di utenti e gruppi di domini

IBM MQ si basa sulle informazioni di appartenenza al gruppo per implementare la propria politica di sicurezza, il che significa che è importante che l'ID utente che sta eseguendo operazioni IBM MQ possa determinare le appartenenze al gruppo di altri utenti.

Quando si promuove un server Windows ad un controller di dominio, viene visualizzata un'opzione per l'impostazione di sicurezza relativa alle autorizzazioni utente e gruppo. Questa opzione controlla se gli utenti arbitrari sono in grado di richiamare le appartenenze ai gruppi da Active Directory. Se un controller di dominio è configurato in modo che gli account locali dispongano dell'autorizzazione per interrogare l'appartenenza al gruppo degli account utente del dominio, l'ID utente predefinito creato da IBM MQ durante il processo di installazione può ottenere l'appartenenza al gruppo per altri utenti come richiesto. Tuttavia, se un controller di dominio è configurato in modo che gli account locali non dispongano dell'autorizzazione per interrogare l'appartenenza al gruppo degli account utente del dominio, ciò impedisce a IBM MQ di completare la verifica che gli utenti definiti sul dominio siano autorizzati ad accedere ai gestori code o alle code e che l'accesso non riesca. Se si utilizza Windows su un controller di dominio impostato in questo modo, è necessario utilizzare un account utente di dominio speciale con le autorizzazioni richieste.

In questo caso, è necessario conoscere:

- Come si comportano le autorizzazioni di protezione per la tua versione di Windows .
- Come consentire ai membri del gruppo mqm del dominio di leggere l'appartenenza al gruppo.
- Come configurare un servizio IBM MQ Windows da eseguire in un utente di dominio.

Per ulteriori informazioni, consultare [Configurazione degli account utente per IBM MQ](#).

## Accesso IBM MQ al gruppo mqm locale

Quando i server Windows vengono promossi o retrocessi da controller di dominio, IBM MQ perde l'accesso al gruppo mqm locale.

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. Il sintomo è un errore che indica la mancanza di un gruppo mqm locale, ad esempio:

```
>critmqm qm0
AMQ8066:Local mqm group not found.
```

Per risolvere questo problema, creare nuovamente il gruppo mq locale utilizzando gli strumenti di gestione Windows standard. Poiché tutte le informazioni di appartenenza al gruppo vengono perse, è necessario ripristinare gli utenti IBM MQ con privilegi 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 IBM MQ del dominio privilegiato il livello di autorizzazione richiesto.

## Restrizioni sui gruppi nidificati su Windows

Esistono restrizioni sull'utilizzo dei gruppi nidificati. Questi risultati derivano in parte dal livello funzionale del dominio e in parte dalle limitazioni IBM 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 2008 e Windows Server 2012 seguono il modello di dominio 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 , sono imposte ulteriori limitazioni agli ID utilizzati da IBM MQ. Le API di rete utilizzate da IBM MQ non supportano tutte le configurazioni supportate dal livello funzionale del dominio. Di conseguenza, IBM MQ non è in grado di interrogare le appartenenze ai gruppi di ID dominio presenti in un gruppo Dominio locale 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.

## **Windows** Autorizzazione degli utenti ad utilizzare IBM MQ in remoto

Se è necessario creare e avviare i gestori code quando si è connessi a IBM MQ in remoto, è necessario disporre dell'accesso utente Crea oggetti globali .

### Informazioni su questa attività

**Nota:** Gli amministratori dispongono dell'accesso utente Crea oggetti globali per impostazione predefinita, quindi se si è un amministratore è possibile creare e avviare i gestori code quando si è connessi in remoto senza modificare i diritti utente.

Se ci si sta collegando a una macchina Windows utilizzando Servizi terminal o una connessione desktop remoto e si hanno problemi durante la creazione, l'avvio o l'eliminazione di un gestore code, è possibile che non si disponga dell'accesso utente Crea oggetti globali .

L'accesso utente Crea oggetti globali limita gli utenti autorizzati a creare oggetti nello spazio dei nomi globale. Affinché un'applicazione crei un oggetto globale, deve essere in esecuzione nello spazio dei nomi globale oppure l'utente con cui è in esecuzione l'applicazione deve disporre dell'accesso utente Crea oggetti globali ad esso applicato.

Quando ci si collega in remoto a una macchina Windows utilizzando Servizi terminal o Connessione desktop remoto, le applicazioni vengono eseguite nel proprio spazio dei nomi locale. Se si tenta di creare o eliminare un gestore code utilizzando IBM MQ Explorer o il comando **crtmqm** o **dltmqm** o per avviare un gestore code utilizzando il comando **strmqm** , si verifica un errore di autorizzazione. Ciò crea un IBM MQ FDC con ID sonda XY132002.

L'avvio di un gestore code utilizzando IBM MQ Explorer o il comando **amqmdain qmgr start** funziona correttamente perché questi comandi non avviano direttamente il gestore code. I comandi inviano invece la richiesta di avviare il gestore code a un processo separato in esecuzione nello spazio dei nomi globale.

Se i vari metodi di gestione di IBM MQ non funzionano quando si utilizzano i servizi terminale, provare a impostare il diritto utente Crea oggetti globali .

### Procedura

1. Aprire il pannello Strumenti di amministrazione:

#### **Windows Server 2008 e Windows Server 2012**

Accedere a questo pannello utilizzando **Pannello di controllo > Sistema e manutenzione > Strumenti di amministrazione**.

#### **Windows 8.1**

Accedere a questo pannello utilizzando **Strumenti di amministrazione > Gestione computer**

2. Fare doppio clic su **Criteri di protezione locale**.
3. Espandere **Politiche locali**.
4. Fare clic su **Assegnazione diritti utente**.
5. Aggiungere il nuovo utente o gruppo alla politica Crea oggetti globali .

## **Windows** Il programma di uscita del canale SSPI su Windows

IBM MQ for 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 l'interfaccia SSPI (Security Support Provider Interface), che fornisce le funzioni di sicurezza 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 IBM WebSphere MQ 7.0.

Questo servizio viene generalmente utilizzato su un canale MQI per consentire a un gestore code del server di autenticare un'applicazione IBM MQ MQI client . 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 IBM WebSphere MQ 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 IBM MQ MQI client ad autenticarsi reciprocamente. 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 su sistemi Windows](#).

### **Windows Applicazione dei file del modello di protezione su Windows**

L'applicazione di un modello potrebbe influire sulle impostazioni di sicurezza applicate ai file e alle directory IBM MQ . Se si utilizza il modello altamente sicuro, applicarlo prima di installare IBM 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 Configurazione sicurezza e analisi MMC. In particolare, Windows fornisce diversi template che includono una serie di impostazioni di protezione con l'obiettivo di fornire livelli specifici di protezione. Questi modelli includono Compatibile, Sicuro e Altamente Sicuro.

l'applicazione di uno di questi modelli potrebbe influire sulle impostazioni di sicurezza applicate ai file e alle directory IBM MQ . Se si desidera utilizzare il modello Highly Secure, configurare la macchina prima di installare IBM MQ.

Se si applica il modello altamente sicuro ad una macchina su cui è già installato IBM MQ , tutte le autorizzazioni impostate sui file e directory IBM MQ vengono rimosse. Poiché queste autorizzazioni vengono rimosse, si perde l'accesso *Amministratore*, *mqme*, quando applicabile, il gruppo *Tutti* dalle directory degli errori.

## **Windows** Configurazione dell'autorizzazione supplementare per applicazioni Windows che si collegano a IBM MQ

L'account con cui vengono eseguiti i processi IBM MQ potrebbe richiedere un'ulteriore autorizzazione prima di poter concedere l'accesso SYNCHRONIZE ai processi dell'applicazione.

### Informazioni su questa attività

Potrebbero verificarsi problemi se si dispone di applicazioni Windows , ad esempio pagine ASP, che si collegano a IBM MQ configurate per essere eseguite a un livello di sicurezza superiore al solito.

IBM MQ richiede l'accesso SYNCHRONIZE ai processi dell'applicazione per coordinare alcune azioni. Quando un'applicazione del server tenta per la prima volta di connettersi a un gestore code IBM MQ modifica il processo per concedere l'autorizzazione SYNCHRONIZE agli amministratori IBM MQ . Tuttavia, l'account con cui vengono eseguiti i processi IBM MQ potrebbe richiedere un'ulteriore autorizzazione prima di poter concedere l'accesso richiesto.

Per configurare l'autorizzazione aggiuntiva all'ID utente con cui sono in esecuzione i processi IBM MQ , completare la seguente procedura:

### Procedura

1. Avviare lo strumento Criteri di protezione locali, fare clic su **Impostazioni di protezione->Criteri locali->Assegnazioni diritti utente**, quindi fare clic su **Debug programmi**.
2. Fare doppio clic su **Programmi di debug**, quindi aggiungere l'ID utente IBM MQ all'elenco

Se il sistema si trova in un dominio Windows e l'impostazione della politica effettiva non viene ancora impostata, anche se l'impostazione della politica locale è impostata, l'ID utente deve essere autorizzato allo stesso modo a livello di dominio, utilizzando lo strumento della politica di sicurezza del dominio.

## **IBM i** Impostazione della sicurezza su IBM i

La sicurezza su IBM i viene implementata utilizzando la sicurezza a livello di oggetto IBM MQ OAM (Object Authority Manager) e IBM i .

Considerazioni sulla sicurezza da effettuare quando si determina l'autorizzazione di accesso agli oggetti IBM MQ .

È necessario considerare i seguenti punti quando si impostano le autorità per gli utenti dell'azienda:

1. Concedere e revocare autorizzazioni ai comandi IBM MQ for IBM i utilizzando i comandi IBM i GRTOBJAUT e RVKOBJAUT .

Nella libreria QMQM , alcuni oggetti non di comando (\* cmd) sono impostati per avere l'autorizzazione **\*PUBLIC** per **\*USE**. Non modificare le autorizzazioni di questi oggetti o utilizzare un elenco di autorizzazioni per fornire l'autorizzazione. Qualsiasi autorizzazione non corretta potrebbe compromettere la funzione IBM MQ .

2. Durante l'installazione di IBM MQ for IBM i, vengono creati i seguenti profili utente speciali:

## QMQM

Viene utilizzato principalmente per funzioni interne di soli prodotti. Tuttavia, può essere utilizzato per eseguire applicazioni attendibili utilizzando MQCNO\_FASTPATH\_BINDINGS. Consultare [Connessione a un gestore code mediante la chiamata MQCONNX](#).

## QMADM

Viene utilizzato come profilo di gruppo per gli amministratori di IBM MQ. Il profilo di gruppo fornisce l'accesso ai comandi CL e alle risorse IBM MQ .

Quando si utilizza SBMJOB per inoltrare programmi che richiamano comandi IBM MQ , USER non deve essere impostato esplicitamente su QMADM. Invece, impostare USER su QM o un altro profilo utente che abbia QMADM specificato come gruppo.

3. Se si stanno inviando i comandi del canale ai gestori code remoti, assicurarsi che il profilo utente sia membro del gruppo QMADM sul sistema di destinazione. Per un elenco dei comandi del canale PCF e MQSC, consultare [IBM MQ for IBM i Comandi CL](#).
4. La serie di gruppi associata ad un utente viene memorizzata in cache quando le autorizzazioni del gruppo vengono calcolate da OAM.

**Le modifiche apportate alle appartenenze al gruppo di un utente dopo che la serie di gruppi è stata memorizzata nella cache non vengono riconosciute finché non si riavvia il gestore code o non si esegue RFRMQAUT per aggiornare la sicurezza.**

5. Limitare il numero di utenti che dispongono dell'autorizzazione per gestire i comandi particolarmente sensibili. Questi comandi includono:
  - Crea gestore code messaggi ( CRTMQM )
  - Elimina gestore code messaggi ( DLTMQM )
  - Avvio del gestore code messaggi ( STRMQM )
  - Fine gestore code messaggi ( ENDMQM )
  - Avvia server dei comandi ( STRMQMCSVR )
  - Termina server dei comandi ( ENDMQMCSVR )
6. Le definizioni di canale contengono una specifica del programma di uscita di sicurezza. La creazione e la modifica del canale richiedono considerazioni speciali. I dettagli delle uscite di sicurezza sono forniti in ["Panoramica sull'uscita di sicurezza"](#) a pagina 97.
7. I programmi di uscita del canale e di controllo trigger possono essere sostituiti. La sicurezza di tali sostituzioni è responsabilità del programmatore.

**IBM i**

## Gestore autorizzazioni oggetto su IBM i

L'OAM (object authority manager) gestisce le autorizzazioni degli utenti per manipolare gli oggetti IBM MQ , incluse le definizioni di code e processi. Fornisce inoltre un'interfaccia comandi attraverso la quale è possibile concedere o revocare l'autorizzazione di accesso a un oggetto per un gruppo specifico di utenti. La decisione di consentire l'accesso a una risorsa viene presa da OAM e il gestore code segue tale decisione. Se OAM non è in grado di prendere una decisione, il gestore code impedisce l'accesso a tale risorsa.

Attraverso l'OAM è possibile controllare:

- Accesso agli oggetti IBM MQ tramite MQI. Quando un programma applicativo tenta di accedere ad un oggetto, l'OAM controlla che il profilo utente che effettua la richiesta disponga dell'autorizzazione per l'operazione richiesta.

In particolare, ciò significa che le code e i messaggi sulle code, possono essere protetti da accessi non autorizzati.

- Autorizzazione per utilizzare comandi PCF e MQSC.

Diversi gruppi di utenti possono disporre di autorizzazioni di accesso differenti per lo stesso oggetto. Ad esempio, per una coda specifica, un gruppo può eseguire sia operazioni di inserimento che operazioni di acquisizione; a un altro gruppo potrebbe essere consentito solo sfogliare la coda (MQGET con l'opzione di

esplorazione). Allo stesso modo, alcuni gruppi potrebbero avere l'autorizzazione get e put per una coda, ma non possono modificare o eliminare la coda.

Comandi IBM MQ for IBM i ed esecuzione di operazioni su oggetti IBM MQ for IBM i

## IBM i **Autorizzazioni IBM MQ su IBM i**

Per accedere agli oggetti IBM MQ , è necessaria l'autorità per immettere il comando e per accedere all'oggetto a cui si fa riferimento. Gli amministratori hanno accesso a tutte le risorse IBM MQ .

L'accesso agli oggetti IBM MQ è controllato dalle autorizzazioni per:

1. Immettere il comando IBM MQ
2. Accedere agli oggetti IBM MQ a cui fa riferimento il comando

Tutti i comandi CL IBM MQ for IBM i vengono forniti con un proprietario di QMQM e il profilo di amministrazione (QMADM) dispone dei diritti \*USE con l'accesso \*PUBLIC impostato su \*EXCLUDE.

**Nota:** Il programma QSRDUPER viene utilizzato dal programma di installazione del programma su licenza IBM MQ per IBM i per duplicare gli oggetti Comando (\*CMD) in QSYS. In IBM i V5R4 e versioni successive, il programma QSRDUPER è stato cambiato in modo che il funzionamento predefinito sia di creare un comando proxy piuttosto che un duplicato del comando originale. Un comando proxy reindirizza l'esecuzione del comando ad un altro comando e ha un attributo PRX. Se un comando proxy con lo stesso nome del comando da copiare esiste nella libreria QSYS, le autorizzazioni private per il comando proxy non vengono concesse al comando nella libreria del prodotto. Tenta di richiedere o eseguire il comando proxy in QSYS per controllare l'autorizzazione del comando di destinazione nella libreria del prodotto. Qualsiasi modifica nell'autorizzazione per gli oggetti \*CMD deve essere effettuata nella libreria del prodotto (QMADM) e quelle in QSYS non devono essere modificate. Ad esempio:

```
GRTOBJAUT OBJ(QMQM/DSPMQM) OBJTYPE(*CMD) USER(MQUSER) AUT(*USE)
```

Le modifiche alla struttura delle autorizzazioni di alcuni comandi CL del prodotto consentono l'utilizzo pubblico di tali comandi, se si dispone dell'autorizzazione OAM richiesta per gli oggetti IBM MQ per apportare tali modifiche.

Per essere un amministratore IBM MQ su IBM i, è necessario essere un membro del *Gruppo QMADM*. Questo gruppo ha proprietà come le proprietà del gruppo mqm sui sistemi UNIX, Linux e Windows . In particolare, il gruppo QMADM viene creato quando si installa IBM MQ for IBM i e i membri del gruppo QMADM hanno accesso alle risorse IBM MQ sul sistema. Inoltre, si ha accesso a tutte le risorse IBM MQ se si dispone dell'autorizzazione \*ALLOBJ.

Gli amministratori possono utilizzare i comandi CL per gestire IBM MQ. Uno di questi comandi è GRTMQMAUT, utilizzato per concedere le autorizzazioni ad altri utenti. Un altro comando, STRMQMMQSC, consente a un amministratore di emettere comandi MQSC per un gestore code locale.

### Concetti correlati

[“Autorizzazione per gestire IBM MQ su IBM i” a pagina 76](#)

## IBM i **Autorizzazioni di accesso per oggetti IBM MQ su IBM i**

Autorizzazioni di accesso richieste per l'esecuzione di comandi CL IBM MQ .

IBM MQ for IBM i categorizza i comandi CL del prodotto in due gruppi:

### Gruppo 1

Gli utenti devono essere nel gruppo utenti QMADM o disporre dell'autorizzazione \*ALLOBJ per elaborare questi comandi. Gli utenti che dispongono di una di queste autorizzazioni possono elaborare tutti i comandi in tutte le categorie senza richiedere alcuna autorizzazione aggiuntiva.

**Nota:** Queste autorizzazioni sovrascrivono tutte le autorizzazioni OAM.

Questi comandi possono essere raggruppati come segue:

- Comandi del Server dei comandi

- ENDMQMCSVR, Chiusura server dei comandi IBM MQ
- STRMQMCSVR, Avvio server dei comandi IBM MQ
- Comando gestore code di messaggi non instradabili
  - STRMQMDLQ, Avvio gestore code di messaggi non instradabili IBM MQ
- Comando listener
  - ENDMQMLSR, Termina listener IBM MQ
  - STRMQMLSR, Avvio listener non oggetto
- Comandi ripristino supporti
  - RCDMQMIMG, Registra immagine oggetto IBM MQ
  - RCRMQMOBJ, Nuova creazione oggetto IBM MQ
  - WRKMQMTRN, Gestione transazioni IBM MQ Q
- Comandi Gestore code
  - CRTMQM, Crea gestore code messaggi
  - DLTMQM, Elimina gestore code messaggi
  - ENDMQM, Termina gestore code messaggi
  - STRMQM, Avvio gestore code messaggi
- Comandi sicurezza
  - GRTMQMAUT, Concedi autorizzazione oggetto IBM MQ
  - RVKMQMAUT, Revoca autorizzazione oggetto IBM MQ
- Comando traccia
  - TRCMQM, Traccia lavoro IBM MQ
- Comandi transazione
  - RSVMQMTRN, Risolvi transazione IBM MQ
- Comandi di controllo trigger
  - STRMQMTRM, Avvio controllo trigger
- Comandi IBM MQSC
  - RUNMQSC, Esegui comandi IBM MQSC
  - STRMQMMQSC, Avvio comandi IBM MQSC

## Gruppo 2

Il resto dei comandi, per i quali sono richiesti due livelli di autorizzazione:

1. Autorizzazione IBM i per eseguire il comando. Un amministratore IBM MQ imposta questa impostazione utilizzando il comando **GRTOBJAUT** per sovrascrivere la limitazione \*PUBLIC (\*EXCLUDE) per un utente o un gruppo di utenti.

Ad esempio:

```
GRTOBJAUT OBJ(QMQM/DSPMQMQ) OBJTYPE(*CMD) USER(MQUSER) AUT(*USE)
```

2. L'autorizzazione IBM MQ per manipolare gli oggetti IBM MQ associati al comando o ai comandi, data l'autorizzazione IBM i corretta nel passo 1.

Questa autorizzazione è controllata dall'utente che dispone dell'autorizzazione OAM appropriata per l'azione richiesta, impostata da un amministratore IBM MQ utilizzando il comando

**GRTMQMAUT**

Ad esempio:

```
GRTMQMAUT *connect authority to the queue manager + *admchg authority to
the queue
```

I comandi possono essere raggruppati come segue:

- Comandi per i canali

- CHGMQMCHL, Modifica canale IBM MQ

Ciò richiede \* l'autorizzazione di connessione al gestore code e \* l'autorizzazione admchg al canale.

- CPYMQMCHL, Copia canale IBM MQ

Ciò richiede le autorizzazioni \* connect e \* admcrt per il gestore code, \* admdsp per il tipo di canale predefinito da copiare e \* admcrt per la classe di oggetti del canale.

Ad esempio, la copia di un canale mittente richiede l'autorizzazione \* admdsp a SYSTEM.DEF.SENDER

- CRTMQMCHL, Creazione canale IBM MQ

Ciò richiede le autorizzazioni \* connect e \* admcrt per il gestore code, \* admdsp per il tipo di canale predefinito da creare e \* admcrt per la classe di oggetti del canale.

Ad esempio, creando un canale mittente, è necessaria l'autorizzazione \* admdsp per SYSTEM.DEF.SENDER

- DLTMQMCHL, Elimina canale IBM MQ

Ciò richiede \* l'autorità di connessione al gestore code e \* l'autorità admdlt al canale.

- RSVMQMCHL, Risolvi canale IBM MQ

Ciò richiede \* l'autorizzazione di connessione al gestore code e \* l'autorizzazione ctrlx al canale.

- Visualizza comandi

Per elaborare i comandi DSP, è necessario concedere l'autorizzazione utente \*connect e \*admdsp al gestore code, insieme a qualsiasi opzione specifica elencata:

- DSPMQM, Visualizza gestore code messaggi

- DSPMQMAUT, Visualizzazione autorizzazione oggetto IBM MQ

- DSPMQMAUTI, Visualizzare IBM MQ Informazioni di autenticazione - \*admdsp per l'oggetto delle informazioni di autenticazione

- DSPMQMCHL, Visualizza IBM MQ Canale \*admdsp sul canale

- DSPMQMCSVR, Visualizzazione server dei comandi IBM MQ

- DSPMQMNL, Visualizza IBM MQ Elenco nomi - \*admdsp nell'elenco nomi

- DSPMQMOBJN, Visualizzazione nomi oggetto IBM MQ

- DSPMQMPRC, Visualizza IBM MQ Processo - \*admdsp al processo

- DSPMQMQ, Visualizza coda IBM MQ - \*admdsp nella coda

- DSPMQMTOP, Visualizza IBM MQ Argomento - \*admdsp all'argomento

- Gestione comandi

Per elaborare i comandi WRK e visualizzare il pannello delle opzioni, è necessario concedere l'autorizzazione utente \*connect e \*admdsp al gestore code, insieme a qualsiasi opzione specifica elencata:

- WRKMQM, Gestione gestori code messaggi

- WRKMQMAUT, Gestione autorizzazione oggetto IBM MQ

- WRKMQMAUTD, Gestione dati autorizzazione oggetto IBM MQ

- WRKMQMAUTI, Gestione informazioni di autenticazione IBM MQ

- \*admchg per il comando Modifica IBM MQ Authentication Information Object.
  - \*admcr̄t per il comando Crea e copia oggetto informazioni di autenticazione IBM MQ .
  - \*admd̄t per il comando Elimina IBM MQ Authentication Information Object.
  - \*admdsp per il comando Visualizzazione IBM MQ Authentication Information Object.
- WRKMQMCHL, Gestione canale IBM MQ
- Ciò richiede le seguenti autorizzazioni:
- \*admchg per il comando Modifica canale IBM MQ .
  - \*admc̄r per il comando Clear IBM MQ Channel.
  - \*admcr̄t per il comando Crea e copia canale IBM MQ .
  - \*admd̄t per il comando Elimina canale IBM MQ .
  - \*admdsp per il comando Visualizza canale IBM MQ .
  - \*ctr̄l per il comando Avvia canale IBM MQ .
  - \*ctr̄l per il comando Fine canale IBM MQ .
  - \*ctr̄l per il comando Ping IBM MQ Channel.
  - \*ctr̄lx per il comando Reimposta canale IBM MQ .
  - \*ctr̄lx per il comando Resolve IBM MQ Channel.
- WRKMQMCHST, Gestione stato canale IBM MQ
- Ciò richiede l'autorizzazione \*admdsp per il canale.
- WRKMQMCL, Gestione cluster IBM MQ
- WRKMQMCLQ, Gestione code cluster IBM MQ
- WRKMQMCLQM, Gestione gestore code cluster IBM MQ
- WRKMQMLSR, Gestione listener IBM MQ
- WRKMQMMSG, Gestione messaggi IBM MQ
- Ciò richiede l'autorizzazione \*browse per la coda
- WRKMQMNL, Gestione elenchi nomi IBM MQ
- Ciò richiede le seguenti autorizzazioni:
- \*admchg per il comando Modifica elenco nomi IBM MQ .
  - \*admcr̄t per il comando Crea e copia elenco nomi IBM MQ .
  - \*admd̄t per il comando Elimina elenco nomi IBM MQ .
  - \*admdsp per il comando Visualizza elenco nomi IBM MQ .
- WRKMQMPCR, Gestione processi IBM MQ
- Ciò richiede le seguenti autorizzazioni:
- \*admchg per il comando Change IBM MQ Process.
  - \*admcr̄t per il comando Crea e copia processo IBM MQ .
  - \*admd̄t per il comando Elimina processo IBM MQ .
  - \*admdsp per il comando Visualizza processo IBM MQ .
- WRKMQMQ, Gestione code IBM MQ
- Ciò richiede le seguenti autorizzazioni:
- \*admchg per il comando Modifica coda IBM MQ .
  - \*admc̄r per il comando Cancella coda IBM MQ .
  - \*admcr̄t per il comando Create and Copy IBM MQ Queue.
  - \*admd̄t per il comando Elimina coda IBM MQ .



- \*admdsp per il comando Visualizza coda IBM MQ .
- WRKMQMSTS, Gestione stato coda IBM MQ
- WRKMQMTOP, Gestione argomenti IBM MQ
- Ciò richiede le seguenti autorizzazioni
  - \*admchg per il comando Modifica argomento IBM MQ .
  - \*admctrl per il comando Crea e copia argomento IBM MQ .
  - \*admdlt per il comando Elimina argomento IBM MQ .
  - \*admdsp per il comando Visualizza argomento IBM MQ .
- WRKMQMSUB, Gestione sottoscrizioni IBM MQ
- Altri comandi del canale
- Per elaborare i comandi del canale, è necessario concedere all'utente le autorizzazioni specifiche elencate:
  - ENDMQMCHL, Arresta canale IBM MQ
  - Ciò richiede l'autorizzazione \*connect al gestore code e l'autorizzazione \*allmqi alla coda di trasmissione associata al canale.
  - ENDMQMLSR, Fine listener IBM MQ
  - Ciò richiede l'autorizzazione \*connect per il gestore code e l'autorizzazione \*ctrl per l'oggetto listener indicato.
  - PNGMQMCHL, Ping IBM MQ Channel
  - Ciò richiede l'autorizzazione \*connect e \*inq per il gestore code e l'autorizzazione \*ctrl per l'oggetto canale.
  - RSTMQMCHL, Reimposta canale IBM MQ
  - Ciò richiede l'autorità \*connect per il gestore code.
  - STRMQMCHL, Avvio canale IBM MQ
  - Ciò richiede l'autorizzazione \*connect per il gestore code e l'autorizzazione \*ctrl per l'oggetto canale.
  - STRMQMCHLI, Avvio iniziatore canale IBM MQ
  - Ciò richiede l'autorità \*connect e \*inq per il gestore code e l'autorizzazione \*allmqi per la coda di avvio associata alla coda di trasmissione del canale.
  - STRMQMLSR, Avvio listener IBM MQ
  - Ciò richiede \* l'autorità di connessione al gestore code e \* l'autorità ctrl all'oggetto listener denominato.
- Altri comandi:
  - Per elaborare i comandi riportati di seguito, è necessario concedere all'utente le autorizzazioni specifiche elencate:
    - CCTMQM, Connessione al gestore code messaggi
    - Non è necessaria alcuna autorizzazione sull'oggetto IBM MQ .
    - CHGMQM, Modifica gestore code messaggi
    - Ciò richiede l'autorità \*connect e \*admchg per il gestore code.
    - CHGMQMAUTI, Modifica informazioni di autenticazione IBM MQ
    - Ciò richiede l'autorizzazione \*connect per il gestore code e l'autorizzazione \*admchg e \*admdsp per l'oggetto delle informazioni di autenticazione.
    - CHGMQMNL, Modifica elenco nomi IBM MQ

- Ciò richiede l'autorizzazione \*connect per il gestore code e l'autorizzazione \*admchg per l'elenco nomi.
- CHGMQMPRC, Modifica processo IBM MQ
 

Ciò richiede l'autorizzazione \*connect per il gestore code e l'autorizzazione \*admchg per il processo.
  - CHGMQMQ, Modifica coda IBM MQ
 

Ciò richiede l'autorizzazione \*connect per il gestore code e l'autorizzazione \*admchg per la coda.
  - CLRMQMQ, Cancella coda IBM MQ
 

Ciò richiede l'autorizzazione \*connect per il gestore code e l'autorizzazione \*admc1r per la coda.
  - CPYMQMAUTI, Copia informazioni di autenticazione IBM MQ
 

Ciò richiede l'autorizzazione \*connect per il gestore code e l'autorizzazione \*admdsp per l'oggetto delle informazioni di autenticazione e l'autorizzazione \*admcr1 per la classe oggetto delle informazioni di autenticazione.
  - CPYMQMNL, Copia elenco nomi IBM MQ
 

Ciò richiede l'autorità \*connect e \*admcr1 per il gestore code.
  - CPYMQMPRC, Copia processo IBM MQ
 

Ciò richiede l'autorità \*connect e \*admcr1 per il gestore code.
  - CPYMQMQ, Copia coda IBM MQ
 

Ciò richiede l'autorità \*connect e \*admcr1 per il gestore code.
  - CRTMQMAUTI, Crea informazioni di autenticazione IBM MQ
 

Ciò richiede l'autorizzazione \*connect per il gestore code e l'autorizzazione \*admdsp per l'oggetto delle informazioni di autenticazione e l'autorizzazione \*admcr1 per la classe oggetto delle informazioni di autenticazione.
  - CRTMQMNL, Crea elenco nomi IBM MQ
 

Ciò richiede l'autorità \*connect e \*admcr1 per il gestore code e l'autorizzazione \*admdsp per l'elenco nomi predefinito.
  - CRTMQMPRC, Crea processo IBM MQ
 

Ciò richiede l'autorizzazione \*connect e \*admcr1 al gestore code e l'autorizzazione \*admdsp per il processo predefinito.
  - CRTMQMQ, Crea coda IBM MQ
 

Ciò richiede l'autorizzazione \*connect e \*admcr1 per il gestore code e l'autorizzazione \*admdsp per la coda predefinita.
  - CVTMQMDTA, Comando di conversione del tipo di dati IBM MQ
 

Non è necessaria alcuna autorizzazione sull'oggetto IBM MQ .
  - DLTMQMAUTI, Elimina informazioni di autenticazione IBM MQ
 

Ciò richiede l'autorità \*connect per il gestore code e l'autorizzazione \*ctrlx per l'oggetto delle informazioni di autenticazione.
  - DLTMQMNL, Elimina elenco nomi IBM MQ
 

Ciò richiede l'autorizzazione \*connect per il gestore code e l'autorizzazione \*admdl1 per l'elenco nomi.
  - DLTMQMPRC, Elimina processo IBM MQ
 

Ciò richiede l'autorizzazione \*connect per il gestore code e l'autorizzazione \*admdl1 per il processo.

- DLTMQMQ, Elimina coda IBM MQ  
Ciò richiede l'autorizzazione \*connect per il gestore code e l'autorizzazione \*admdl1t per la coda.
- DSCMQM, Disconnetti dal gestore code messaggi  
Non è necessaria alcuna autorizzazione sull'oggetto IBM MQ .
- RFRMQMAUT, Aggiorna sicurezza  
Ciò richiede l'autorità \*connect per il gestore code.
- RFRMQMCL, Aggiorna cluster  
Ciò richiede l'autorità \*connect per il gestore code.
- RSMMQMCLQM, Riprendi gestore code cluster  
Ciò richiede l'autorità \*connect per il gestore code.
- RSTMQMCL, Reimposta cluster  
Ciò richiede l'autorità \*connect per il gestore code.
- SPDMQMCLQM, Gestore code cluster di sospensione  
Ciò richiede l'autorità \*connect per il gestore code.

### **IBM i** **Autorizzazioni di accesso su IBM i**

Utilizzare queste informazioni per comprendere i comandi di autorizzazione di accesso.

Le autorizzazioni definite dalla parola chiave AUT nei comandi GRTMQMAUT e RVKMQMAUT possono essere categorizzate come segue:

- Autorizzazioni correlate alle chiamate MQI
- Comandi di gestione correlati all'autorizzazione
- Autorizzazioni contesto
- Autorizzazioni generali, ossia, per le chiamate MQI, per i comandi o per entrambi

Le tabelle riportate di seguito elencano le diverse autorizzazioni, utilizzando il parametro AUT per le chiamate MQI, le chiamate di contesto, i comandi MQSC e PCF e le operazioni generico.

<i>Tabella 15. Autorizzazioni per le chiamate MQI</i>	
<b>AUT</b>	<b>Descrizione</b>
*ALTUSR	Consentire l'utilizzo dell'autorizzazione di un altro utente per chiamate MQOPEN e MQPUT1 .
*SFOGLIA	Richiamare un messaggio da una coda emettendo una chiamata MQGET con l'opzione BROWSE.
*CONNECT	Collegare l'applicazione al gestore code specificato emettendo una chiamata MQCONN.
*GET	Richiamare un messaggio da una coda emettendo una chiamata MQGET.
*INQ	Eseguire una richiesta su una coda specifica emettendo una chiamata MQINQ.
*PUB	Aprire un topic per pubblicare un messaggio utilizzando una chiamata MQPUT.
*PUT	Inserire un messaggio su una coda specifica emettendo una chiamata MQPUT.
*RESUME	Riprendere una sottoscrizione utilizzando una chiamata MQSUB.
*SET	Impostare gli attributi su una coda dalla MQI emettendo una chiamata MQSET. Se si apre una coda per più opzioni, è necessario essere autorizzati per ognuna di esse.

Tabella 15. Autorizzazioni per le chiamate MQI (Continua)

AUT	Descrizione
*SUB	Creare, modificare o riprendere una sottoscrizione a un argomento utilizzando una chiamata MQSUB.

Tabella 16. Autorizzazioni per chiamate di contesto

AUT	Descrizione
*PASSALL	Passare tutto il contesto sulla coda specificata. Tutti i campi di contesto vengono copiati dalla richiesta originale.
*PASSID	Passare il contesto di identità sulla coda specificata. Il contesto di identità è uguale a quello della richiesta.
*SETALL	Imposta tutto il contesto sulla coda specificata. Viene utilizzato da programmi di utilità di sistema speciali.
*SETID	Impostare il contesto di identità sulla coda specificata. Viene utilizzato da programmi di utilità di sistema speciali.

Tabella 17. Autorizzazioni per chiamate MQSC e PCF

AUT	Descrizione
*ADMCHG	Modificare gli attributi dell'oggetto specificato.
*ADMCLR	Eliminare l'oggetto specificato (solo comando PCF Clear object).
*ADMCRT	Creare oggetti del tipo specificato.
*ADMDLT	Eliminare l'oggetto specificato.
*ADMDSP	Visualizza gli attributi dell'oggetto specificato.

Tabella 18. Autorizzazioni per operazioni generiche

AUT	Descrizione
*ALL	Utilizzare tutte le operazioni applicabili all'oggetto. L'autorizzazione all è equivalente all'unione delle autorizzazioni alladm, allmqie system appropriate al tipo di oggetto.
*ALLADM	Eseguire tutte le operazioni di amministrazione applicabili all'oggetto.
*ALLMQI	Utilizzare tutte le chiamate MQI applicabili all'oggetto.
*CTRL	Controllare l'avvio e l'arresto di canali, listener e servizi.
*CTRLX	Reimpostare il numero di sequenza e risolvere i canali in dubbio.



### Utilizzo dei comandi di autorizzazione di accesso su IBM i

Utilizzare queste informazioni per informazioni sui comandi di autorizzazione di accesso e utilizzare gli esempi di comando.

#### Utilizzo del comando GRMQMAUT

Se si dispone dell'autorizzazione richiesta, è possibile utilizzare il comando GRMQMAUT per concedere l'autorizzazione di un profilo utente o di un gruppo di utenti per accedere a un particolare oggetto. I seguenti esempi illustrano il modo in cui viene utilizzato il comando GRMQMAUT :

1. 

```
GRTMQMAUT OBJ(RED.LOCAL.QUEUE) OBJTYPE(*LCLQ) USER(GROUPA) +
AUT(*BROWSE *PUT) MQMNAME('saturn.queue.manager')
```

In questo esempio:

- RED.LOCAL.QUEUE è il nome oggetto.
  - \*LCLQ (coda locale) è il tipo di oggetto.
  - GROUPA è il nome di un profilo utente sul sistema per cui devono essere modificate le autorizzazioni. Questo profilo può essere utilizzato come profilo di gruppo per altri utenti.
  - \*BROWSE e \*PUT sono le autorizzazioni concesse alla coda specificata.
    - \*BROWSE aggiunge l'autorizzazione per sfogliare i messaggi sulla coda (per emettere MQGET con l'opzione di ricerca).
    - \*PUT aggiunge l'autorizzazione per inserire i messaggi (MQPUT) nella coda.
  - saturn.queue.manager è il nome del gestore code.
2. Il seguente comando concede agli utenti JACK e JILL tutte le autorizzazioni applicabili, a tutte le definizioni di processo, per il gestore code predefinito.

```
GRTMQMAUT OBJ(*ALL) OBJTYPE(*PRC) USER(JACK JILL) AUT(*ALL)
```

3. Il seguente comando concede all'utente GEORGE l'autorità per inserire un messaggio sulla coda ORDERS, nel gestore code TRENT.

```
GRTMQMAUT OBJ(TRENT) OBJTYPE(*MQM) USER(GEORGE) AUT(*CONNECT) MQMNAME (TRENT)
GRTMQMAUT OBJ(ORDERS) OBJTYPE(*Q) USER(GEORGE) AUT(*PUT) MQMNAME (TRENT)
```

## Utilizzo del comando RVKMQMAUT

Se si dispone dell'autorizzazione richiesta, è possibile utilizzare il comando RVKMQMAUT per rimuovere l'autorizzazione precedentemente concessa di un profilo utente o di un gruppo di utenti per accedere a un oggetto particolare. I seguenti esempi illustrano come viene utilizzato il comando RVKMQMAUT :

1. 

```
RVKMQMAUT OBJ(RED.LOCAL.QUEUE) OBJTYPE(*LCLQ) USER(GROUPA) +
AUT(*PUT) MQMNAME('saturn.queue.manager')
```

L'autorizzazione per inserire messaggi nella coda specificata, concessa nell'esempio precedente, viene rimossa per GROUPA.

2. 

```
RVKMQMAUT OBJ(PAY*) OBJTYPE(*Q) USER(*PUBLIC) AUT(*GET) +
MQMNAME(PAYROLLQM)
```

L'autorizzazione per richiamare i messaggi da qualsiasi coda con un nome che inizia con i caratteri PAY, di proprietà del gestore code PAYROLLQM, viene rimossa da tutti gli utenti del sistema a meno che essi, o un gruppo a cui appartengono, non siano stati autorizzati separatamente.

## Utilizzo del comando DSPMQMAUT

L'autorizzazione MQM di visualizzazione ( DSPMQMAUT ) per l'oggetto e l'utente specificati, l'elenco delle autorizzazioni che l'utente ha per l'oggetto. Il seguente esempio illustra come viene utilizzato il comando:

```
DSPMQMAUT OBJ(ADMINNL) OBJTYPE(*NMLIST) USER(JOE) OUTPUT(*PRINT) +
MQMNAME (ADMINQM)
```

## Utilizzo del comando RFRMQMAUT

La sicurezza MQM di aggiornamento ( RFRMQMAUT ) consente di aggiornare immediatamente le informazioni sul gruppo di autorizzazione di OAM, riflettendo le modifiche apportate al livello di sistema operativo, senza dover arrestare e riavviare il gestore code. Il seguente esempio illustra come viene utilizzato il comando:

```
RFRMQMAUT MQMNAME (ADMINQM)
```

IBM i

## Tablette di specifiche di autorizzazione su IBM i

Utilizzare queste informazioni per determinare quale autorizzazione è richiesta per utilizzare particolari chiamate API e particolari opzioni di tali chiamate, sugli oggetti coda, sugli oggetti processo e sugli oggetti gestore code.

Le tabelle delle specifiche di autorizzazione che iniziano in [Tabella 19 a pagina 159](#) definiscono con precisione il funzionamento delle autorizzazioni e le limitazioni che si applicano. 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 i seguenti dati:

### Azione da eseguire

Opzione MQI, comando MQSC o comando PCF.

### Oggetto controllo accessi

Coda, definizione del processo, gestore code, elenco nomi, canale, canale di connessione client, listener, servizio o oggetto delle informazioni di autenticazione.

### Autorizzazione richiesta

Espresso come costante MQZAO\_.

Nelle tabelle, le costanti con prefisso MQZAO\_ corrispondono alle parole chiave nell'elenco di autorizzazioni per i comandi **GRTMQMAUT** e **RVKMQMAUT** per la particolare entità. Ad esempio, MQZAO\_BROWSE corrisponde alla parola chiave \*BROWSE ; in modo simile, la parola chiave MQZAO\_SET\_ALL\_CONTEXT corrisponde alla parola chiave \*SETALL. Le così via. Queste costanti sono definite nel file di intestazione cmqzc.h, fornito con il prodotto.

## Autorizzazioni MQI

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 richiedono controlli 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 risultanti dalla risoluzione di un nome. Ad esempio, ad una applicazione può essere concessa l'autorizzazione ad aprire una coda alias senza avere l'autorizzazione ad aprire la coda base in cui si risolve l'alias. La regola è che il controllo viene eseguito sulla prima definizione rilevata durante il processo di risoluzione del 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 il particolare oggetto che viene aperto; in alcuni casi è richiesta un'ulteriore autorizzazione indipendente dalla coda, ottenuta tramite un'autorizzazione per l'oggetto gestore code.

[Tabella 19 a pagina 159](#), [Tabella 20 a pagina 159](#), [Tabella 21 a pagina 160](#) e [Tabella 22 a pagina 160](#) riepilogano le autorizzazioni necessarie per ogni chiamata.

**Nota:** Queste tabelle non menzionano 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.

<i>Tabella 19. Autorizzazione di sicurezza necessaria per chiamate MQCONN</i>			
<b>Autorizzazione richiesta per:</b>	<b>Oggetto coda ( “1” a pagina 160 )</b>	<b>Oggetto processo</b>	<b>Oggetto gestore code</b>
opzione MQCONN	Non applicabile	Non applicabile	CONNECT MQZAO_

<i>Tabella 20. Autorizzazione di sicurezza necessaria per chiamate MQOPEN</i>			
<b>Autorizzazione richiesta per:</b>	<b>Oggetto coda ( “1” a pagina 160 )</b>	<b>Oggetto processo</b>	<b>Oggetto gestore code</b>
MQOO_INQUIRE	MQZAO_INQUIRE ( “2” a pagina 160 )	MQZAO_INQUIRE ( “2” a pagina 160 )	MQZAO_INQUIRE ( “2” a pagina 160 )
MQOO_SFOGLIA	MQZAO_BROWSE	Non applicabile	Nessun controllo
MQOO_INPUT_*	INPUT MQZAO_	Non applicabile	Nessun controllo
MQOO_SAVE_ALL_CONTEXT ( “3” a pagina 160 )	INPUT MQZAO_	Non applicabile	Non applicabile
MQOO_OUTPUT (Coda normale) ( “4” a pagina 160 )	OUTPUT MQZAO_	Non applicabile	Non applicabile
MQOO_PASS_IDENTITY_CONTEXT ( “5” a pagina 160 )	MQZAO_PASS_CONTESTO_IDENTITÀX_ENCODE_CASE_CAPS_LOCK_OFF	Non applicabile	Nessun controllo
MQOO_PASS_ALL_CONTEXT ( “5” a pagina 160, “6” a pagina 160 )	MQZAO_PASS_TUTTO_CONTESTO	Non applicabile	Nessun controllo
MQOO_SET_IDENTITY_CONTEXT ( “5” a pagina 160, “6” a pagina 160 )	MQZAO_SET_CONTESTO_IDENTITÀ	Non applicabile	MQZAO_SET_CONTESTO_IDENTITÀ ( “7” a pagina 160 )
MQOO_SET_ALL_CONTEXT ( “5” a pagina 160, “8” a pagina 161 )	MQZAO_SET_TUTTO_CONTESTO	Non applicabile	MQZAO_SET_ALL_CONTEXT ( “7” a pagina 160 )
MQOO_OUTPUT (Coda di trasmissione) ( “9” a pagina 161 )	MQZAO_SET_TUTTO_CONTESTO	Non applicabile	MQZAO_SET_ALL_CONTEXT ( “7” a pagina 160 )
SET MQOO	MQZAO_SET	Non applicabile	Nessun controllo
MQOO_ALTERNATE_AUTORITÀ_UTENTE	( “10” a pagina 161 )	( “10” a pagina 161 )	MQZAO_ALTERNATE_USER_AUTHORITY ( “10” a pagina 161, “11” a pagina 161 )

<i>Tabella 21. Autorizzazione di sicurezza necessaria per le chiamate MQPUT1</i>			
<b>Autorizzazione richiesta per:</b>	<b>Oggetto coda ( “1” a pagina 160 )</b>	<b>Oggetto processo</b>	<b>Oggetto gestore code</b>
MQPMO_PASS_CONTESTO_IDENTITÀ	MQZAO_PASS_IDENTITY_CONTEXT ( “12” a pagina 161 )	Non applicabile	Nessun controllo
MQPMO_PASS_ALL_CONTESTO	MQZAO_PASS_ALL_CONTEXT ( “12” a pagina 161 )	Non applicabile	Nessun controllo
MQPMO_SET_CONTESTO_IDENTITÀ	MQZAO_SET_CONTESTO_IDENTITÀ ( “12” a pagina 161 )	Non applicabile	MQZAO_SET_CONTESTO_IDENTITÀ ( “7” a pagina 160 )
MQPMO_SET_TUTTO_CONTESTO	MQZAO_SET_ALL_CONTEXT ( “12” a pagina 161 )	Non applicabile	MQZAO_SET_ALL_CONTEXT ( “7” a pagina 160 )
(Coda di trasmissione) ( “9” a pagina 161 )	MQZAO_SET_TUTTO_CONTESTO	Non applicabile	MQZAO_SET_ALL_CONTEXT ( “7” a pagina 160 )
AUTORIZZAZIONE_UTENTE_MQPMO_ALTERNATE_	( “13” a pagina 161 )	Non applicabile	MQZAO_ALTERNATE_USER_AUTHORITY ( “11” a pagina 161 )

<i>Tabella 22. Autorizzazione di sicurezza necessaria per le chiamate MQCLOSE</i>			
<b>Autorizzazione richiesta per:</b>	<b>Oggetto coda ( “1” a pagina 160 )</b>	<b>Oggetto processo</b>	<b>Oggetto gestore code</b>
MQCO_DELETE	MQZAO_DELETE ( “14” a pagina 161 )	Non applicabile	Non applicabile
MQCO_DELETE_PURGE	MQZAO_DELETE ( “14” a pagina 161 )	Non applicabile	Non applicabile

**Note per le tabelle:**

- Se si sta aprendo 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.
- L'oggetto coda, processo, elenco nomi o gestore code viene controllato, a seconda del tipo di oggetto che si sta aprendo.
- È necessario specificare anche MQOO\_INPUT\_\*. Questa opzione è valida per una coda locale, modello o alias.
- Questo controllo viene eseguito per tutti i casi di output, ad eccezione del caso specificato nella nota “9” a pagina 161.
- È necessario specificare anche MQOO\_OUTPUT.
- MQOO\_PASS\_IDENTITY\_CONTEXT è implicito anche da questa opzione.
- Questa autorizzazione è richiesta sia per l'oggetto gestore code che per la particolare coda.



8. Anche MQOO\_PASS\_IDENTITY\_CONTEXT, MQOO\_PASS\_ALL\_CONTEXT e MQOO\_SET\_IDENTITY\_CONTEXT sono implicite in questa opzione.
9. 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).
10. È necessario specificare anche almeno uno tra MQOO\_INQUIRE (per qualsiasi tipo di oggetto) o (per le code) MQOO\_BROWSE, MQOO\_INPUT\_\*, MQOO\_OUTPUT o MQOO\_SET. 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.
11. Questa autorizzazione consente di specificare qualsiasi *AlternateUserId*.
12. Viene eseguito anche un controllo MQZAO\_OUTPUT se la coda non dispone di un attributo della coda *Utilizzo* di MQUS\_TRANSMISSION.
13. Il controllo eseguito è quello per le altre opzioni specificate, utilizzando l'identificativo utente alternativo fornito per l'autorità della coda denominata e l'autorità dell'applicazione corrente per il controllo MQZAO\_ALTERNATE\_USER\_IDENTIFIER.
14. Il controllo viene eseguito solo se entrambe le seguenti istruzioni sono vere:
  - Una coda dinamica permanente è in fase di chiusura ed eliminazione.
  - La coda non è stata creata da MQOPEN che ha restituito l'handle dell'oggetto utilizzato.
 In caso contrario, non c'è alcun controllo.

#### **Note generali:**

1. L'autorizzazione speciale MQZAO\_ALL\_MQI include tutte le seguenti autorizzazioni relative al tipo di oggetto:
  - CONNECT MQZAO\_
  - INQUIRE MQZAO\_
  - MQZAO\_SET
  - MQZAO\_BROWSE
  - INPUT MQZAO\_
  - OUTPUT MQZAO\_
  - Contesto MQZAO\_PASS\_IDENTITY\_CONTEXT
  - MQZAO\_PASS\_ALL\_CONTEXT
  - MQZAO\_SET\_IDENTITY\_CONTEXT
  - MQZAO\_SET\_ALL\_CONTEXT
  - MQZAO\_ALTERNATE\_USER\_AUTHORITY
2. MQZAO\_DELETE (vedere nota "14" a pagina 161) e MQZAO\_DISPLAY sono classificati come autorizzazioni di amministrazione. Pertanto, non sono inclusi in MQZAO\_ALL\_MQI.
3. *Nessun controllo* significa che non viene effettuato alcun controllo di autorizzazione.
4. *Non applicabile* indica che il controllo dell'autorizzazione non è rilevante per questa operazione. Ad esempio, non è possibile emettere una chiamata MQPUT per un oggetto processo.

#### **IBM i Autorizzazioni per i comandi MQSC nei PCF di escape su IBM i**

Queste autorizzazioni consentono all'utente di emettere comandi di gestione come messaggio PCF di uscita. Questi metodi consentono a un programma di inviare un comando di gestione come messaggio a un gestore code, per l'esecuzione per conto di tale utente.

Questa sezione riassume le autorizzazioni necessarie per ogni comando MQSC contenuto in Escape PCF.

*Non applicabile* indica che il controllo dell'autorizzazione non è rilevante per questa operazione.

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 DISPLAY sul gestore code per eseguire i comandi PCF
- Autorizzazione per emettere i comandi MQSC all'interno del testo del comando Escape PCF

#### **ALTER oggetto**

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
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

#### **CLEAR oggetto**

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
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
Listener	Non applicabile
Servizio	Non applicabile

#### **DEFINE oggetto NOREPLACE ( "1" a pagina 166 )**

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
Coda	<b>MQZAO_CREATE ( "2" a pagina 166 )</b>
Argomento	<b>MQZAO_CREATE ( "2" a pagina 166 )</b>
Processo	<b>MQZAO_CREATE ( "2" a pagina 166 )</b>
Gestore code	Non applicabile
Elenco nomi	<b>MQZAO_CREATE ( "2" a pagina 166 )</b>

Oggetto	Autorizzazione richiesta
Informazioni di autenticazione	<b>MQZAO_CREATE ( “2” a pagina 166 )</b>
Canale	<b>MQZAO_CREATE ( “2” a pagina 166 )</b>
Canale connessione client	<b>MQZAO_CREATE ( “2” a pagina 166 )</b>
Listener	<b>MQZAO_CREATE ( “2” a pagina 166 )</b>
Servizio	<b>MQZAO_CREATE ( “2” a pagina 166 )</b>

**DEFINE oggetto REPLACE ( “1” a pagina 166, “3” a pagina 166 )**

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

**DELETE 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

**VISUALIZZA oggetto**

Oggetto	Autorizzazione richiesta
Coda	DISPLAY MQZAO_
Argomento	DISPLAY MQZAO_
Processo	DISPLAY MQZAO_

Oggetto	Autorizzazione richiesta
Gestore code	DISPLAY MQZAO_
Elenco nomi	DISPLAY MQZAO_
Informazioni di autenticazione	DISPLAY MQZAO_
Canale	DISPLAY MQZAO_
Canale connessione client	DISPLAY MQZAO_
Listener	
Servizio	

### Ping canale

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	Non applicabile
Servizio	Non applicabile

### Reimpostazione canale

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	MQZAO_CONTROL_XX_ENCODE_CASE_ONE uscita
Canale connessione client	Non applicabile
Listener	Non applicabile
Servizio	Non applicabile

**Risoluzione canale**

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
Coda	Non applicabile
Argomento	Non applicabile
Processo	Non applicabile
Gestore code	Non applicabile
Elenco nomi	Non applicabile
Informazioni di autenticazione	Non applicabile
Canale	MQZAO_CONTROL_XX_ENCODE_CASE_ONE uscita
Canale connessione client	Non applicabile
Listener	Non applicabile
Servizio	Non applicabile

**START oggetto**

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
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_

**STOP oggetto**

<b>Oggetto</b>	<b>Autorizzazione richiesta</b>
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_

Oggetto	Autorizzazione richiesta
Servizio	CONTROL MQZAO_

**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 omissso.
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 GRTRMQMAUT .
3. Questa opzione si applica se l'oggetto da sostituire esiste già. In caso contrario, il controllo è quello per DEFINE *oggetto* NOREPLACE.

**IBM i Autorizzazioni per comandi PCF su IBM i**

Queste autorizzazioni consentono all'utente di emettere comandi di gestione come comandi PCF. Questi metodi consentono a un programma di inviare un comando di gestione come messaggio a un gestore code, per l'esecuzione per conto di tale utente.

Questa sezione riepiloga le autorizzazioni necessarie per ogni comando PCF.

*Nessun controllo* significa che non viene eseguito alcun controllo di autorizzazione; *Non applicabile* significa che il controllo di autorizzazione non è rilevante per questa operazione.

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 DISPLAY sul gestore code per eseguire i comandi PCF

L'autorizzazione speciale MQZAO\_ALL\_ADMIN comprende le seguenti autorizzazioni:

- MODIFICA\_MQZO
- CLEAR MQZAO\_
- MQZAO\_DELETE
- DISPLAY MQZAO\_
- CONTROL MQZAO\_
- MQZAO\_CONTROL\_XX\_ENCODE\_CASE\_ONE uscita

MQZAO\_CREATE non è incluso in quanto non è specifico di 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

Oggetto	Autorizzazione richiesta
Servizio	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
Listener	Non applicabile
Servizio	Non applicabile

**Copiare oggetto (senza sostituire) ( “1” a pagina 172 )**

Oggetto	Autorizzazione richiesta
Coda	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Argomento	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Processo	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Gestore code	Non applicabile
Elenco nomi MQZAO_CREATE	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Informazioni di autenticazione	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Canale	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Canale connessione client	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Listener	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Servizio	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>

**Copia oggetto (con sostituzione) ( “1” a pagina 172, “4” a pagina 172 )**

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

Oggetto	Autorizzazione richiesta
Canale connessione client	MODIFICA_MQZO
Listener	MODIFICA_MQZO
Servizio	MODIFICA_MQZO

**Crea oggetto (senza sostituzione) ( “3” a pagina 172 )**

Oggetto	Autorizzazione richiesta
Coda	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Argomento	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Processo	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Gestore code	Non applicabile
Elenco nomi	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Informazioni di autenticazione	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Canale	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Canale connessione client	<b>MQZAO_CREATE ( “2” a pagina 172 )</b>
Listener	MODIFICA_MQZO
Servizio	MODIFICA_MQZO

**Crea oggetto (con sostituzione) ( “3” a pagina 172, “4” a pagina 172 )**

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

**Elimina oggetto**

Oggetto	Autorizzazione richiesta
Coda	MQZAO_DELETE
Argomento	MQZAO_DELETE
Processo	MQZAO_DELETE
Gestore code	MQZAO_DELETE
Elenco nomi	MQZAO_DELETE



Oggetto	Autorizzazione richiesta
Informazioni di autenticazione	MQZAO_DELETE
Canale	MQZAO_DELETE
Canale connessione client	MQZAO_DELETE
Listener	MQZAO_DELETE
Servizio	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_

#### Interroga nomi oggetto

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

#### Ping canale

Oggetto	Autorizzazione richiesta
Coda	Non applicabile
Argomento	Non applicabile
Processo	Non applicabile

Oggetto	Autorizzazione richiesta
Gestore code	Non applicabile
Elenco nomi	Non applicabile
Informazioni di autenticazione	Non applicabile
Canale	CONTROL MQZAO_
Canale connessione client	Non applicabile
Listener	Non applicabile
Servizio	Non applicabile

#### Reimposta canale

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	MQZAO_CONTROL_XX_ENCODE_CASE_ONE uscita
Canale connessione client	Non applicabile
Listener	Non applicabile
Servizio	Non applicabile

#### Reimposta statistiche coda

Oggetto	Autorizzazione richiesta
Coda	MQZAO_DISPLAY e MQZAO_CHANGE
Argomento	Non applicabile
Processo	Non applicabile
Gestore code	Non applicabile
Elenco nomi	Non applicabile
Informazioni di autenticazione	Non applicabile
Canale	Non applicabile
Canale connessione client	Non applicabile
Listener	
Servizio	

### Risolvi canale

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	MQZAO_CONTROL_XX_ENCODE_CASE_ONE uscita
Canale connessione client	Non applicabile
Listener	Non applicabile
Servizio	Non applicabile

### Avvio canale

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	Non applicabile
Servizio	Non applicabile

### Arresta canale

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	Non applicabile

Oggetto	Autorizzazione richiesta
Servizio	Non applicabile

**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 GRMQMAUT .
3. Per i comandi di creazione, è necessaria anche l'autorizzazione MQZAO\_DISPLAY per il SISTEMA SYSTEM.DEFAULT.\* dell'oggetto.
4. Questa opzione si applica se l'oggetto da sostituire esiste già. In caso contrario, il controllo è come per Copia o Crea senza sostituzione.

**IBM i Profili OAM generici su IBM i**

I profili generici OAM (Object authority manager) consentono di impostare l'autorizzazione di un utente su molti oggetti contemporaneamente, piuttosto che dover immettere comandi **GRMQMAUT** separati per ogni singolo oggetto quando viene creato. L'utilizzo di profili generici nel comando **GRMQMAUT** consente di impostare un'autorizzazione generica per tutti gli oggetti futuri creati che si adattino a tale profilo.

Il resto di questa sezione descrive più dettagliatamente l'uso dei profili generici:

- [“Utilizzo dei caratteri jolly” a pagina 172](#)
- [“Priorità del profilo” a pagina 173](#)

**Utilizzo dei caratteri jolly**

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 concessa a quel profilo si applica a tutti gli oggetti creati 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 applica 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 applica 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 la parola chiave OBJTYPE (\*PRC) per identificare i processi, quindi utilizzare \*\* 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.

## 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:

```
GRTMQMAUT OBJ(AB.*) OBJTYPE(*Q) USER(FRED) AUT(*PUT) MQMNAME(MYQMGR)
GRTMQMAUT OBJ(AB.C*) OBJTYPE(*Q) USER(FRED) AUT(*GET) MQMNAME(MYQMGR)
```

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, GRTMQMAUT può essere applicato 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. \*\*

IBM i

## Specifica del servizio di autorizzazione installato su IBM i

È possibile specificare quale componente del servizio di autorizzazione utilizzare.

Il parametro **Service Component name** su **GRTMQMAUT** e **RVKMQMAUT** consente di specificare il nome del componente del servizio di autorizzazione installato.

Selezionando **F24** sul pannello iniziale, seguito da **F9=All parametri** sul pannello successivo di uno dei comandi, è possibile specificare il componente di autorizzazione installato (\*DFT) o il nome del componente del servizio di autorizzazione richiesto specificato nella stanza Service del file qm.ini del gestore code.

**DSPMQMAUT** ha anche questo ulteriore parametro. Questo parametro consente di ricercare tutti i componenti di autorizzazione installati (\*DFT) o il nome del componente del servizio di autorizzazione specificato, per il nome oggetto, il tipo di oggetto e l'utente specificati

IBM i

## Gestione e senza profili di autorizzazione su IBM i

Utilizzare queste informazioni per informazioni su come gestire i profili di autorizzazione e su come lavorare senza tali profili.

È possibile lavorare con i profili di autorizzazione, come spiegato in [“Gestione dei profili di autorizzazione”](#) a pagina 174, o senza di essi, come spiegato di seguito:

Per lavorare senza profili di autorizzazione, utilizzare \*NONE come parametro di autorizzazione su **GRTMQMAUT** per creare profili senza autorizzazione. In questo modo, i profili esistenti non vengono modificati.

Su **RVKMQMAUT**, utilizzare \*REMOVE come parametro di autorizzazione per rimuovere un profilo di autorizzazione esistente.

## Gestione dei profili di autorizzazione

Esistono due comandi associati alla creazione profili di autorizzazione:

- **WRKMQMAUT**
- **WRKMQMAUTD**

È possibile accedere a questi comandi direttamente dalla riga comandi o dal pannello WRKMQM:

1. Immettere il nome del gestore code e premere il tasto **Enter** per accedere al pannello dei risultati di **WRKMQM**.
2. Selezionando **F23=More options** in questo pannello.

L'opzione 24 seleziona il pannello dei risultati per il **WRKMQMAUT** comando e l'opzione 25 seleziona il comando **WRKMQMAUTI**, utilizzato con il livello dei collegamenti SSL.

### WRKMQMAUT

Questo comando permette di gestire i dati di autorizzazione congelati nella coda di autorizzazioni.

**Nota:** Per eseguire questo comando, è necessaria l'autorità **\*connect** e **\*admdsp** per il gestore code. Tuttavia, per creare o eliminare un profilo, è necessaria l'autorizzazione **QMADM**.

Se si emettono le informazioni sul pannello, viene visualizzato un elenco di nomi di profili di autorizzazione, insieme ai relativi tipi. Se si stampa l'emissione, si riceve un elenco dettagliato di tutti i dati di autorizzazione, gli utenti registrati e le relative autorizzazioni.

Immettendo un oggetto o un nome profilo su questo pannello e premendo **INVIO** si accede al pannello dei risultati per **WRKMQMAUT**.

Se si seleziona **4=Delete**, si accede ad un nuovo pannello da cui è possibile confermare che si desidera eliminare tutti i nomi utente registrati per il nome del profilo di autorizzazione generico specificato. Questa opzione esegue **RVKMQMAUT** con l'opzione **\*REMOVE** per tutti gli utenti e applica **solo** ai nomi di profilo generici.

Se si seleziona **12=Work with profile**, si passa al pannello dei risultati del comando **WRKMQMAUTD**, come descritto in [“WRKMQMAUTD”](#) a pagina 174.

### WRKMQMAUTD

Questo comando permette di visualizzare tutti gli utenti registrati con un particolare nome profilo di autorizzazione e tipo di oggetto. Per eseguire questo comando, è necessaria l'autorità **\*connect** e **\*admdsp** per il gestore code. Tuttavia, per concedere, eseguire, creare o eliminare un profilo è necessaria l'autorizzazione **QMADM**.

Selezionando **F24=More keys** dal pannello di input iniziale, seguito dall'opzione **F9=All Parameters** viene visualizzato il nome del componente del servizio come per **GRTMQMAUT** e **RVKMQMAUT**.

**Nota:** Il tasto **F11=Display Object Authorizations** si sposta tra i seguenti tipi di autorizzazioni:

- Autorizzazioni oggetto
- Autorizzazioni contesto
- Autorizzazioni MQI

Le opzioni sullo schermo sono:

#### **2=Grant**

Porta al pannello **GRTMQMAUT** per aggiungere le autorizzazioni correnti.

#### **3=Revoke**

Visualizza il pannello **RVKMQMAUT** per rimuovere alcune definizioni correnti

#### 4=Delete

Porta a un pannello che consente di cancellare i dati di autorizzazione per gli utenti specificati. Viene eseguito **RVKMQMAUT** con l'opzione \*REMOVE.

#### 5=Display

Porta al comando **DSPMQMAUT** esistente

#### F6=Create

Porta al pannello **GRTMQMAUT** che consente di creare un record di autorizzazione profilo.

## Linee guida di Object Authority Manager su IBM i

Ulteriori suggerimenti e suggerimenti per l'utilizzo di OAM (object authority manager)

### Limita l'accesso alle operazioni sensibili

Alcune operazioni sono sensibili; limitarle agli utenti privilegiati. Ad esempio,

- Accesso ad alcune code speciali, come ad esempio le code di trasmissione o la coda comandi `SYSTEM.ADMIN.COMMAND.QUEUE`
- Esecuzione di programmi che utilizzano le opzioni di contesto MQI complete
- Creazione e copia di code di applicazioni

### Directory del gestore code

Le directory e le librerie che contengono le code e altri dati del gestore code sono privati per il prodotto. Non utilizzare i comandi del sistema operativo standard per concedere o revocare le autorizzazioni alle risorse MQI.

### Code

L'autorizzazione a una coda dinamica si basa, ma non è necessariamente uguale a quella della coda modello da cui è derivata.

Per le code alias e remote, l'autorizzazione è quella dell'oggetto stesso, non la coda in cui si risolve l'alias o la coda remota. È possibile autorizzare un profilo utente ad accedere ad una coda alias che si risolve in una coda locale per cui il profilo utente non dispone delle autorizzazioni di accesso.

Limitare l'autorizzazione a creare code per utenti privilegiati. In caso contrario, gli utenti possono ignorare il controllo accessi normale creando un alias.

### Autorizzazione utente alternativo

L'autorizzazione utente alternativo controlla se un profilo utente può utilizzare l'autorizzazione di un altro profilo utente durante l'accesso a un oggetto IBM MQ. Questa tecnica è essenziale quando un server riceve le richieste da un programma e il server desidera assicurarsi che il programma disponga dell'autorizzazione 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:

- Un programma server in esecuzione sotto il profilo utente PAYSERV richiama un messaggio di richiesta da una coda inserita nella coda dal profilo 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 profilo utente (PAYSERV) per autorizzare l'apertura della coda di risposta, il server può specificare un altro profilo utente, in questo caso USER1. In questo esempio, è possibile utilizzare l'autorizzazione utente alternativo per controllare se PAYSERV può specificare USER1 come profilo utente alternativo quando apre la coda di risposta.

Il profilo utente alternativo viene specificato sul campo *AlternateUserId* del descrittore oggetto.

**Nota:** È possibile utilizzare profili utente alternativi su qualsiasi oggetto IBM MQ . L'utilizzo di un profilo utente alternativo non influisce sul profilo utente utilizzato da altri gestori risorse.

## Autorizzazione contesto

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.

Per le descrizioni dei campi del descrittore del messaggio relativi al contesto, consultare [Panoramica MQMD](#).

Per informazioni sulle opzioni di contesto, consultare [Contesto del messaggio](#).

## Considerazioni sulla sicurezza remota

Per la sicurezza remota, considerare:

### Autorizzazioni Put

Per la sicurezza tra i gestori code, è possibile specificare l'autorizzazione di inserimento utilizzata quando un canale riceve un messaggio inviato da un altro gestore code.

Questo parametro è valido solo per tipi di canale RCVR, RQSTR o CLUSRCVR. Specificare l'attributo del canale PUTAUT come segue:

#### DEF

Profilo utente predefinito. Questo è il profilo utente QMQM con cui è in esecuzione l'agent del canale messaggi.

#### CTX

Il profilo utente nel contesto del messaggio.

### Code di trasmissione

I gestori code inserano automaticamente i messaggi remoti su una coda di trasmissione; non è richiesta alcuna autorizzazione speciale. Tuttavia, l'inserimento di un messaggio direttamente su una coda di trasmissione richiede un'autorizzazione speciale.

### Uscite canale

Le uscite canale possono essere utilizzate per una maggiore sicurezza.

### Record di autenticazione di canale

Utilizzare per esercitare un controllo più preciso sull'accesso concesso ai sistemi di connessione a livello di canale.

Per ulteriori informazioni sulla sicurezza remota, consultare ["Autorizzazione canale" a pagina 100](#).

## Protezione dei canali con SSL/TLS

Il protocollo TLS (Transport Layer Security) fornisce la sicurezza del canale, con protezione da intercettazioni, manomissioni e impersonificazione. Il supporto IBM MQ per TLS consente di specificare, nella definizione del canale, che un determinato canale utilizza la sicurezza TLS. È anche possibile specificare i dettagli della protezione desiderata, ad esempio l'algoritmo di codifica che si desidera utilizzare.

Il supporto TLS in IBM MQ utilizza l' *oggetto delle informazioni di autenticazione* del gestore code e diversi comandi CL e MQSC e i parametri del gestore code e del canale che definiscono il supporto TLS richiesto in dettaglio.

I seguenti comandi CL supportano TLS:

### WRKMQMAUTI

Gestire gli attributi di un oggetto delle informazioni di autenticazione.

### CHGMQMAUTI

Modificare gli attributi di un oggetto delle informazioni di autenticazione.



## CRTMQMAUTI

Creare un oggetto delle informazioni di autenticazione.

## CPYMQMAUTI

Creare un oggetto delle informazioni di autenticazione copiandone uno esistente.

## DLTMQMAUTI

Elimina un oggetto delle informazioni di autenticazione.

## DSPMQMAUTI

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

Per una panoramica sulla sicurezza del canale utilizzando TLS, consultare

- [Protezione dei canali con TLS](#)

Per i comandi PCF associati a TLS, consultare

- [Modificare, copiare e creare l'oggetto delle informazioni di autenticazione](#)
- [Elimina oggetto informazioni di autenticazione](#)
- [Richiedi oggetto informazioni di autenticazione](#)

## Impostazione della sicurezza su z/OS

Considerazioni sulla sicurezza specifiche di z/OS.

La sicurezza in IBM MQ for z/OS viene controllata utilizzando RACF o un ESM (external security manager) equivalente.

Se un ID utente dispone di `uid0`, ha accesso all'intero file system, ossia all'accesso Superuser. In RACF, l'unico modo per limitare ciò è utilizzando la funzionalità FSACCESS, che può limitare l'accesso a livello di file system da parte dell'ID utente RACF. Per ulteriori informazioni, consultare [Utilizzo del profilo della classe FSACCESS per limitare l'accesso](#) nella documentazione *z/OS UNIX System Services Planning*.

Le seguenti istruzioni presuppongono che si stia utilizzando RACF.

### Informazioni correlate

[Scenario di sicurezza: due gestori code su z/OS](#)

[Scenario di sicurezza: gruppo di condivisione code su z/OS](#)

## Classi di sicurezza RACF

Le classi RACF vengono utilizzate per conservare i profili richiesti per il controllo di sicurezza IBM MQ. Molte delle classi membro hanno classi di gruppo equivalenti. È necessario attivare le classi e abitarle ad accettare i profili generici

Ciascuna classe RACF contiene uno o più profili utilizzati in un determinato punto della sequenza di controllo, come mostrato in [Tabella 23 a pagina 177](#).

Classe membro	Classe gruppo	Contenuto
MQADMIN	GMQADMIN	Profili: Utilizzato principalmente per mantenere i profili per le funzioni di tipo amministrazione. Ad esempio: <ul style="list-style-type: none"><li>• Profili per switch di sicurezza IBM MQ</li><li>• Profilo di sicurezza RESLEVEL</li><li>• Profili per la sicurezza utente alternativa</li><li>• Il profilo di sicurezza del contesto</li><li>• Profili per la sicurezza della risorsa comando</li></ul>

Tabella 23. Classi RACF utilizzate da IBM MQ (Continua)

Classe membro	Classe gruppo	Contenuto
MXADMIN	GMXADMIN	<p>Profili:</p> <p>Utilizzato principalmente per mantenere i profili per le funzioni di tipo amministrazione. Ad esempio:</p> <ul style="list-style-type: none"> <li>• Profili per switch di sicurezza IBM MQ</li> <li>• Profilo di sicurezza RESLEVEL</li> <li>• Profili per la sicurezza utente alternativa</li> <li>• Il profilo di sicurezza del contesto</li> <li>• Profili per la sicurezza della risorsa comando</li> </ul> <p>Questa classe può contenere sia profili in maiuscolo che in maiuscolo / minuscolo RACF .</p>
MQCONN		Profili utilizzati per la sicurezza della connessione
MQCMD5		Profili utilizzati per la sicurezza dei comandi
MQQUEUE	GMQUEUE	Profili utilizzati nella sicurezza delle risorse della coda
MXQUEUE	CODAXGM	Profili maiuscoli e minuscoli utilizzati nella sicurezza delle risorse della coda
MQPROC	GMQPROC	Profili utilizzati nella sicurezza delle risorse del processo
MXPROC	PROC GMX	Profili maiuscoli e minuscoli utilizzati nella sicurezza delle risorse del processo
MQNLIST	ELENCO GMQ	Profili utilizzati nella sicurezza delle risorse dell'elenco nomi
MXNLIST	ELENCO GMX	Profili maiuscoli e minuscoli utilizzati nella sicurezza delle risorse dell'elenco nomi
MXTOPIC	GMXTOPIC	Profili maiuscoli e minuscoli utilizzati nella sicurezza dell'argomento

Alcune classi hanno una *classe gruppo* correlata che consente di raggruppare gruppi di risorse che hanno requisiti di accesso simili. Per i dettagli sulla differenza tra le classi membro e gruppo e quando utilizzare una classe membro o gruppo, consultare [z/OS Security Server RACF Security Administrator's Guide](#).

Le classi devono essere attivate prima di poter effettuare i controlli di sicurezza. Per attivare tutte le classi IBM MQ , è possibile utilizzare questo comando RACF :

```
SETROPTS CLASSACT(MQADMIN, MXADMIN, MQQUEUE, MXQUEUE, MQPROC, MXPROC,
                  MQNLIST, MXNLIST, MXTOPIC, MQCONN, MQCMD5)
```

È inoltre necessario assicurarsi di impostare le classi in modo che possano accettare profili generici. Eseguire questa operazione anche con il comando RACF SETROPTS, ad esempio:

```
SETROPTS GENERIC(MQADMIN, MXADMIN, MQQUEUE, MXQUEUE, MQPROC, MXPROC,
                 MQNLIST, MXNLIST, MXTOPIC, MQCONN, MQCMD5)
```

Tutti i profili RACF utilizzati da IBM MQ contengono un prefisso, che è il nome del gestore code o il nome del gruppo di condivisione code. Prestare attenzione quando si utilizza il segno percentuale come carattere jolly.

Tutti i profili RACF utilizzati da IBM MQ contengono un prefisso. Per la sicurezza a livello di gruppo di condivisione code, questo è il nome del gruppo di condivisione code. Per la sicurezza a livello di gestore code, il prefisso è il nome del gestore code. Se si utilizza una combinazione di sicurezza a livello di gestore code e di gruppo di condivisione code, si utilizzeranno i profili con entrambi i tipi di prefisso. (Il gruppo di condivisione code e la sicurezza a livello di gestore code vengono descritti in [IBM MQ for z/OS concetti: sicurezza.](#))

Ad esempio, se si desidera proteggere una coda denominata `QUEUE_FOR_SUBSCRIBER_LIST` nel gruppo di condivisione code `QSG1` a livello di gruppo di condivisione code, il profilo appropriato viene definito in RACF come:

```
RDEFINE MQQUEUE QSG1.QUEUE_FOR_SUBSCRIBER_LIST
```

Se si desidera proteggere una coda denominata `QUEUE_FOR_LOST_CARD_LIST`, appartenente al gestore code `STCD` a livello di gestore code, il profilo appropriato verrà definito in RACF come:

```
RDEFINE MQQUEUE STCD.QUEUE_FOR_LOST_CARD_LIST
```

Ciò significa che diversi gestori code e gruppi di condivisione code possono condividere lo stesso database RACF ma hanno diverse opzioni di protezione.

Non utilizzare nomi di gestori code generici nei profili per evitare l'accesso utente non previsto.

IBM MQ consente l'utilizzo del segno percentuale (%) nei nomi oggetto. Tuttavia, RACF utilizza il carattere% come carattere jolly a carattere singolo. Ciò significa che quando si definisce un nome oggetto con un carattere% nel nome, è necessario considerarlo quando si definiscono i profili corrispondenti.

Ad esempio, per la coda `CREDIT_CARD_%_RATE_QUIRY`, sul CRDP del gestore code, il profilo verrà definito in RACF come segue:

```
RDEFINE MQQUEUE CRDP.CREDIT_CARD_%_RATE_INQUIRY
```

Questa coda non può essere protetta da un profilo generico, ad esempio `CRDP.**`.

IBM MQ consente l'utilizzo di caratteri maiuscoli e minuscoli nei nomi oggetto. È possibile proteggere questi oggetti definendo:

1. Profili con maiuscole e minuscole nelle classi RACF con maiuscole e minuscole appropriate oppure
2. Profili generici nelle classi RACF maiuscole appropriate.

Per utilizzare i profili con maiuscole e minuscole e le classi RACF con maiuscole e minuscole, è necessario seguire la procedura descritta in [z/OS Migrazione di un gestore code alla sicurezza con maiuscole e minuscole.](#)

Ci sono alcuni profili, o parti di profili, che rimangono maiuscoli solo quando i valori sono forniti da IBM MQ. Sono:

- Profili switch.
- Tutti i qualificatori di alto livello (HLQ) inclusi gli identificativi del gruppo di condivisione code e del sottosistema.
- Profili per gli oggetti `SYSTEM`.
- Profili per oggetti predefiniti.

- La classe **MQCMDS** , quindi tutti i profili di comando sono solo in maiuscolo.
- La classe **MQCONN** , in modo che tutti i profili di connessione siano solo in maiuscolo.
- Profili **RESLEVEL** .
- La qualifica 'object' nei profili della risorsa comando; ad esempio, hlq.QUEUE.queuname. Il nome risorsa è composto solo da caratteri maiuscoli e minuscoli.
- Profili della coda dinamica hlq.CSQOREXX.\* , hlq.CSQUTIL.\*e CSQXCMD.\*.
- Il 'CONTEXT' parte di hlq.CONTEXT.resourcename.
- La parte 'ALTERNATE.USER' di hlq.ALTERNATE.USER.userid.

Ad esempio, se si dispone di una coda denominata PAYROLL.Dept1 sul gestore code QM01 e si sta utilizzando:

- Profili con casi misti; è possibile definire un profilo nella classe IBM MQ RACF MXQUEUE

```
RDEFINE MXQUEUE QM01.PAYROLL.Dept1
```

- Profili in maiuscolo; è possibile definire un profilo nella IBM MQ classe RACF MQQUEUE

```
RDEFINE MQQUEUE QM01.PAYROLL.*
```

Il primo esempio, utilizzando profili con maiuscole e minuscole, fornisce un controllo più granulare sulla concessione dell'autorità per accedere alla risorsa.

## Profili switch

Per controllare il controllo di sicurezza eseguito da IBM MQ, utilizzare *profili switch*. Un profilo switch è un profilo RACF normale che ha un significato speciale per IBM MQ. L'elenco di accesso nei profili di switch non viene utilizzato da IBM MQ.

IBM MQ mantiene uno switch interno per ciascun tipo di switch mostrato nelle tabelle [Profili switch](#) per la sicurezza a livello del sottosistema, [Profili switch](#) per la sicurezza a livello del gestore code o del gruppo di condivisione codee [Profili switch](#) per il controllo delle risorse. I profili switch possono essere gestiti a livello di gruppo di condivisione code, a livello di gestore code o a una combinazione di entrambi. Utilizzando una singola serie di profili di commutazione di sicurezza del gruppo di condivisione code, è possibile controllare la sicurezza su tutti i gestori code all'interno di un gruppo di condivisione code.

Quando è impostato un interruttore di sicurezza, vengono eseguiti i controlli di sicurezza associati allo switch. Quando uno switch di sicurezza è disattivato, i controlli di sicurezza associati allo switch vengono ignorati. Il valore predefinito è che tutti gli switch di sicurezza sono impostati.

## Switch e classi

Quando si avvia un gestore code o si aggiorna la sicurezza, IBM MQ imposta gli switch in base allo stato delle varie classi RACF .

Quando un gestore code viene avviato (o quando la classe MQADMIN o MXADMIN viene aggiornata dal comando IBM MQ [REFRESH SECURITY](#) ), IBM MQ verifica innanzitutto lo stato di RACF e la classe appropriata:

- La classe MQADMIN se si utilizzano profili in maiuscolo
- La classe MXADMIN se si utilizza un profilo con caratteri maiuscoli e minuscoli.

Imposta l'interruttore di sicurezza del sottosistema se si verifica una delle seguenti condizioni:

- RACF è inattivo o non installato.
- La classe MQADMIN o MXADMIN non è definita (queste classi sono sempre definite per RACF perché sono incluse nella tabella dei descrittori di classe (CDT)).
- La classe MQADMIN o MXADMIN non è stata attivata.

Se sia RACF che la classe MQADMIN o MXADMIN sono attivi, IBM MQ controlla la classe MQADMIN o MXADMIN per verificare se uno dei profili switch è stato definito. Verifica innanzitutto i profili descritti in [“Profili per controllare la sicurezza del sottosistema”](#) a pagina 181. Se la sicurezza del sottosistema non è richiesta, IBM MQ imposta lo spegnimento della sicurezza del sottosistema interno e non esegue ulteriori controlli.

I profili determinano se l'interruttore IBM MQ corrispondente è impostato su on o off.

- Se l'interruttore è spento, quel tipo di sicurezza viene disattivato.
- Se uno switch IBM MQ è impostato su on, IBM MQ controlla lo stato della classe RACF associata al tipo di sicurezza corrispondente allo switch IBM MQ . Se la classe non è installata o non è attiva, lo switch IBM MQ è disattivato. Ad esempio, i controlli di sicurezza del processo non vengono eseguiti se la classe MQPROC o MXPROC non è stata attivata. La classe non attiva equivale a definire NO.PROCESS.CHECKS per ogni gestore code e gruppo di condivisione code che utilizza questo database RACF .

### **Funzionamento degli switch**

Per disattivare un interruttore di sicurezza, definire un NO.\* il profilo di commutazione. È possibile sovrascrivere un NO.\* profilo impostato a livello di gruppo di condivisione code definendo un YES.\* per un gestore code.

Per disattivare un interruttore di sicurezza, è necessario definire un NO.\* il profilo di commutazione. L'esistenza di un NO.\* Il profilo indica che i controlli di protezione **non** vengono eseguiti per quel tipo di risorsa, a meno che non si scelga di sovrascrivere un'impostazione a livello di gruppo di condivisione code su uno specifico gestore code. Ciò è descritto in [“Sovrascrittura delle impostazioni a livello di gruppo di condivisione code”](#) a pagina 181.

Se il gestore code non è un membro di un gruppo di condivisione code, non è necessario definire alcun profilo a livello di gruppo di condivisione code o qualsiasi profilo di sovrascrittura. Tuttavia, è necessario ricordarsi di definire questi profili se il gestore code si unisce a un gruppo di condivisione code in un secondo momento.

Ogni NO.\* commutare il profilo che IBM MQ rileva disattiva la verifica per quel tipo di risorsa. I profili switch vengono attivati durante l'avvio del gestore code. Se si modificano i profili di switch mentre sono in esecuzione tutti i gestori code interessati, è possibile fare in modo che IBM MQ riconosca le modifiche immettendo il comando IBM MQ REFRESH SECURITY.

I profili switch devono essere sempre definiti nella classe MQADMIN o MXADMIN. Non definirli nella classe GMQADMIN o GMXADMIN. Le tabelle [Profili switch per la sicurezza a livello di sottosistema](#) e [Profili switch per il controllo risorse](#) mostrano i profili switch validi e il tipo di sicurezza che controllano.

### **Sovrascrittura delle impostazioni a livello di gruppo di condivisione code**

È possibile sovrascrivere le impostazioni di protezione a livello di gruppo di condivisione code per uno specifico gestore code membro di tale gruppo. Se si desidera eseguire i controlli del gestore code su un singolo gestore code non eseguito su altri gestori code del gruppo, utilizzare il comando (qmgr-name.YES.\* ) profili di commutazione.

Al contrario, se non si desidera eseguire un determinato controllo su un determinato gestore code all'interno di un gruppo di condivisione code, definire un (qmgr-name.NO.\* ) profilo per quel particolare tipo di risorsa sul gestore code e non definire un profilo per il gruppo di condivisione code. ( IBM MQ controlla solo un profilo a livello di gruppo di condivisione code se non trova un profilo a livello di gestore code.)

### **Profili per controllare la sicurezza del sottosistema**

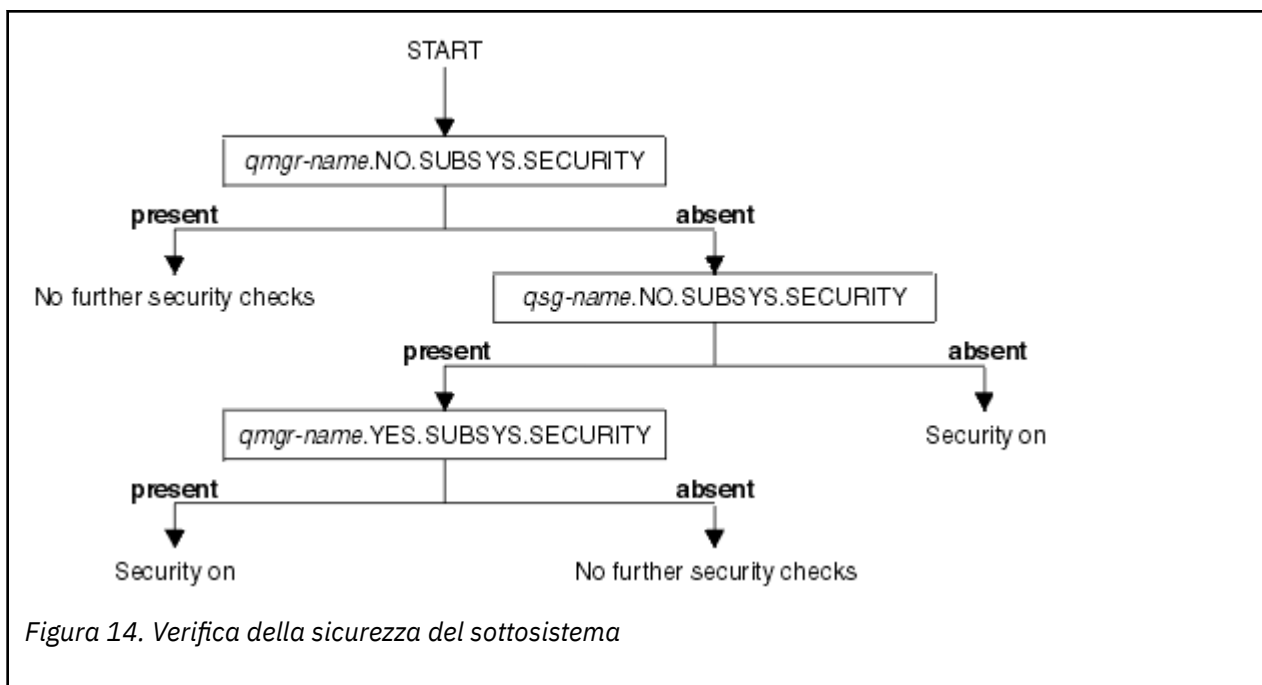
IBM MQ verifica se i controlli di sicurezza del sottosistema sono richiesti per il sistema secondario, per il gestore code e per il gruppo di condivisione code.

Il primo controllo di sicurezza effettuato da IBM MQ viene utilizzato per stabilire se i controlli di sicurezza sono richiesti per l'intero sottosistema IBM MQ . Se si specifica che non si desidera la sicurezza del sistema secondario, non vengono effettuati ulteriori controlli.

I seguenti profili switch vengono controllati per determinare se è richiesta la sicurezza del sottosistema. Figura 14 a pagina 182 mostra l'ordine in cui vengono controllati.

Nome profilo switch	Tipo di risorsa o controllo controllato
qmgr-name.NO.SUBSYS.SECURITY	Sicurezza del sottosistema per questo gestore code
qsg-name.NO.SUBSYS.SECURITY	Sicurezza del sottosistema per questo gruppo di condivisione code
qmgr-name.YES.SUBSYS.SECURITY	Sovrascrittura della sicurezza del sottosistema per questo gestore code

Se il gestore code non è un membro di un gruppo di condivisione code, IBM MQ ricerca solo il profilo di commutazione qmgr-name.NO.SUBSYS.SECURITY .



### z/OS Profili per controllare il gruppo di condivisione code o la sicurezza a livello di gestore code

Se è richiesto il controllo di sicurezza del sistema secondario, IBM MQ verifica se il controllo di protezione è richiesto a livello di gruppo di condivisione code o di gestore code.

Quando IBM MQ ha determinato che è richiesto il controllo di sicurezza, determina se il controllo è richiesto a livello di gruppo di condivisione code, gestore code o entrambi. Questi controlli non vengono eseguiti se il gestore code non è un membro di un gruppo di condivisione code.

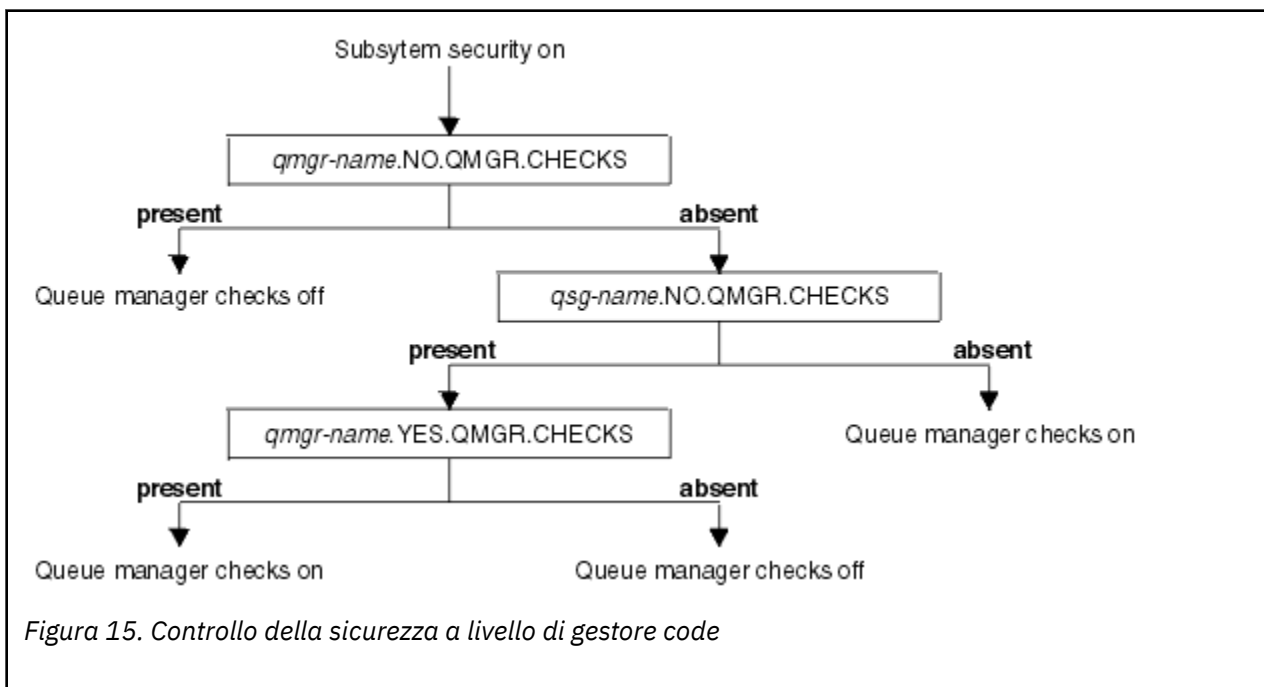
I seguenti profili switch vengono controllati per determinare il livello richiesto. Figura 15 a pagina 183 e Figura 16 a pagina 184 mostrano l'ordine in cui vengono controllati.

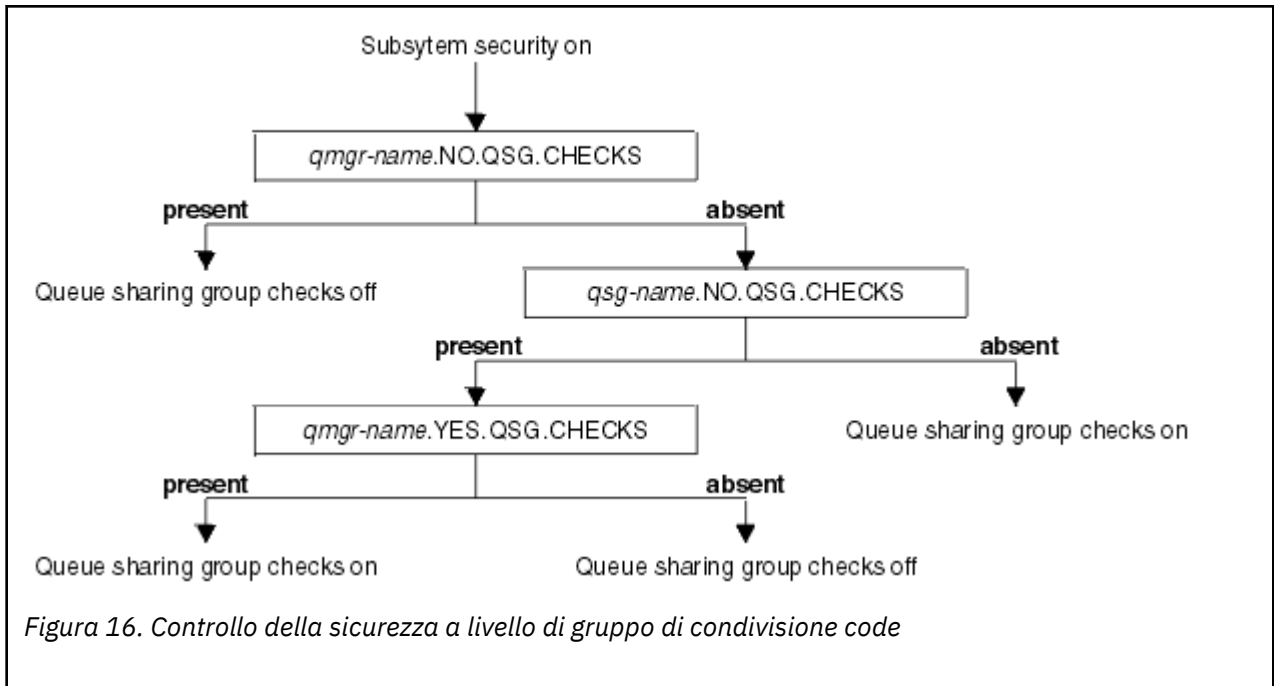
Nome profilo switch	Tipo di risorsa o controllo controllato
qmgr-name.NO.QMGR.CHECKS	Nessun controllo a livello di gestore code per questo gestore code

Tabella 25. Profili di commutazione per il gruppo di condivisione code o la sicurezza a livello di gestore code (Continua)

Nome profilo switch	Tipo di risorsa o controllo controllato
qsg-name.NO.QMGR.CHECKS	Nessun controllo a livello di gestore code per questo gruppo di condivisione code
qmgr-name.YES.QMGR.CHECKS	I controlli a livello di gestore code sovrascrivono questo gestore code
qmgr-name.NO.QSG.CHECKS	Nessun controllo a livello di gruppo di condivisione code per questo gestore code
qsg-name.NO.QSG.CHECKS	Nessun livello di controllo del gruppo di condivisione code per questo gruppo di condivisione code
qmgr-name.YES.QSG.CHECKS	I controlli a livello di gruppo di condivisione code sovrascrivono questo gestore code

Se la sicurezza del sottosistema è attiva, non è possibile disattivare la sicurezza a livello di gestore code e di gruppo di condivisione code. Se si tenta di farlo, IBM MQ imposta il controllo di sicurezza su entrambi i livelli.





**z/OS** *Combinazioni valide di switch di sicurezza*

Sono valide solo alcune combinazioni di switch. Se si utilizza una combinazione di impostazioni switch non valida, viene emesso il messaggio CSQH026I e il controllo di sicurezza è impostato a livello di gruppo di condivisione code e di gestore code.

Tabella 26 a pagina 184, Tabella 27 a pagina 184, Tabella 28 a pagina 185e Tabella 29 a pagina 185 mostrano le serie di combinazioni di impostazioni switch valide per ogni tipo di livello di sicurezza.

*Tabella 26. Combinazioni di switch di sicurezza valide per la sicurezza a livello di gestore code*

<b>Combinazioni</b>
qmgr-name.NO.QSG.CHECKS
qsg-name.NO.QSG.CHECKS
qmgr-name.NO.QSG.CHECKS qsg-name.NO.QMGR.CHECKS qmgr-name.YES.QMGR.CHECKS
qsg-name.NO.QSG.CHECKS qsg-name.NO.QMGR.CHECKS qmgr-name.YES.QMGR.CHECKS

*Tabella 27. Combinazioni di switch di sicurezza valide per la sicurezza a livello di gruppo di condivisione code*

<b>Combinazioni</b>
qmgr-name.NO.QMGR.CHECKS
qsg-name.NO.QMGR.CHECKS
qmgr-name.NO.QMGR.CHECKS qsg-name.NO.QSG.CHECKS qmgr-name.YES.QSG.CHECKS



*Tabella 27. Combinazioni di switch di sicurezza valide per la sicurezza a livello di gruppo di condivisione code (Continua)*

**Combinazioni**

qsg-name.NO.QMGR.CHECKS  
 qsg-name.NO.QSG.CHECKS  
 qmgr-name.YES.QSG.CHECKS

*Tabella 28. Combinazioni di switch di sicurezza valide per la sicurezza a livello del gestore code e del gruppo di condivisione code*

**Combinazioni**

qsg-name.NO.QMGR.CHECKS  
 qmgr-name.YES.QMGR.CHECKS  
 Nessun QSG.\* profili definiti

Nessun QMGR.\* profili definiti  
 qsg-name.NO.QSG.CHECKS  
 qmgr-name.YES.QSG.CHECKS

qsg-name.NO.QMGR.CHECKS  
 qmgr-name.YES.QMGR.CHECKS  
 qsg-name.NO.QSG.CHECKS  
 qmgr-name.YES.QSG.CHECKS

Non è stato definito alcun profilo per lo switch

*Tabella 29. Altre combinazioni di switch di sicurezza valide che commutano entrambi i livelli di controllo su.*

**Combinazioni**

qmgr-name.NO.QMGR.CHECKS  
 qmgr-name.NO.QSG.CHECKS

qsg-name.NO.QMGR.CHECKS  
 qsg-name.NO.QSG.CHECKS

qmgr-name.NO.QMGR.CHECKS  
 qsg-name.NO.QSG.CHECKS

qsg-name.NO.QMGR.CHECKS  
 qmgr-name.NO.QSG.CHECKS

 **Controlli a livello di risorsa**

Un certo numero di profili switch vengono utilizzati per controllare l'accesso alle risorse. Alcuni smettono di eseguire il controllo su un gestore code o su un gruppo di condivisione code. Questi possono essere sovrascritti da profili che consentono il controllo di specifici gestori code.

La [Tabella 30 a pagina 186](#) mostra i profili switch utilizzati per controllare l'accesso alle risorse IBM MQ .

Se il gestore code fa parte di un gruppo di condivisione code e si dispone della sicurezza del gestore code e del gruppo di condivisione code attivi, è possibile utilizzare un comando YES.\* commutare il profilo per

sovrascrivere i profili a livello di gruppo di condivisione code e attivare specificamente la sicurezza per un determinato gestore code.

Alcuni profili si applicano sia ai gestori code che ai gruppi di condivisione code. Questi sono preceduti dalla stringa *hlq* ed è necessario sostituire il nome del gruppo di condivisione code o del gestore code, come applicabile. I nomi profilo visualizzati con prefisso *qmgr - name* sono profili di sovrascrittura del gestore code; è necessario sostituire il nome del gestore code.

*Tabella 30. Profili di commutazione per il controllo delle risorse*

<b>Tipo di controllo delle risorse controllato</b>	<b>Nome profilo switch</b>	<b>Sovrascrivi profilo per un particolare gestore code</b>
Sicurezza connessione	hlq.NO.CONNECT.CHECKS	qmgr-name.YES.CONNECT.CHECKS
Sicurezza coda	hlq.NO.QUEUE.CHECKS	qmgr-name.YES.QUEUE.CHECKS
Sicurezza del processo	hlq.NO.PROCESS.CHECKS	qmgr-name.YES.PROCESS.CHECKS
Sicurezza elenco nomi	hlq.NO.NLIST.CHECKS	qmgr-name.YES.NLIST.CHECKS
sicurezza del contesto	hlq.NO.CONTEXT.CHECKS	qmgr-name.YES.CONTEXT.CHECKS
sicurezza utente alternativa	hlq.NO.ALTERNATE.USER.CHECKS	qmgr-name.YES.ALTERNATE.USER.CHECKS
Sicurezza comando	hlq.NO.CMD.CHECKS	qmgr-name.YES.CMD.CHECKS
Sicurezza della risorsa comando	hlq.NO.CMD.RESC.CHECKS	qmgr-name.YES.CMD.RESC.CHECKS
sicurezza dell'argomento	hlq.NO.TOPIC.CHECKS	qmgr-name.YES.TOPIC.CHECKS

**Nota:** Profili switch generici come hlq.NO. \* \* vengono ignorati da IBM MQ

Ad esempio, se si desidera eseguire i controlli di sicurezza del processo sul gestore code QM01, che è membro del gruppo di condivisione code QSG3 ma non si desidera eseguire i controlli di protezione del processo su nessuno degli altri gestori code del gruppo, definire i seguenti profili switch:

```
QSG3.NO.PROCESS.CHECKS
QM01.YES.PROCESS.CHECKS
```

Se si desidera che i controlli di sicurezza della coda vengano eseguiti su tutti i gestori code nel gruppo di condivisione code, tranne QM02, definire il seguente profilo switch:

```
QM02.NO.QUEUE.CHECKS
```

Non è necessario definire un profilo per il gruppo di condivisione code poiché i controlli vengono abilitati automaticamente se non è definito alcun profilo.

### **Un esempio di definizione degli switch**

Sottosistemi IBM MQ differenti hanno requisiti di sicurezza differenti, che possono essere implementati utilizzando profili di switch differenti.

Sono stati definiti quattro sottosistemi IBM MQ :

- MQP1 (un sistema di produzione)
- MQP2 (un sistema di produzione)
- MQD1 (un sistema di sviluppo)
- MQT1 (un sistema di verifica)

Tutti e quattro i gestori code sono membri del gruppo di condivisione code QS01. Tutte le classi IBM MQ RACF sono state definite e attivate.

Questi sottosistemi hanno diversi requisiti di sicurezza:

- I sistemi di produzione richiedono il controllo di sicurezza IBM MQ completo per essere attivi a livello di gruppo di condivisione code su entrambi i sistemi.

Questa operazione viene eseguita specificando il seguente profilo:

```
RDEFINE MQADMIN QS01.NO.QMGR.CHECKS
```

Imposta il controllo a livello di gruppo di condivisione code per tutti i gestori code nel gruppo di condivisione code. Non è necessario definire altri profili di switch per i gestori code di produzione perché si desidera controllare tutti i sistemi.

- Il gestore code di verifica MQT1 richiede anche un controllo di sicurezza completo. Tuttavia, poiché è possibile modificarlo in seguito, la sicurezza può essere definita a livello di gestore code in modo che sia possibile modificare le impostazioni di sicurezza per questo gestore code senza influire sugli altri membri del gruppo di condivisione code.

Questa operazione viene eseguita definendo NO.QSG.CHECKS per MQT1 come segue:

```
RDEFINE MQADMIN MQT1.NO.QSG.CHECKS
```

- Il gestore code di sviluppo MQD1 ha requisiti di sicurezza diversi dal resto del gruppo di condivisione code. Richiede che solo la connessione e la sicurezza della coda siano attive.

Questa operazione viene eseguita definendo un profilo MQD1.YES.QMGR.CHECKS per questo gestore code e quindi definendo i seguenti profili per disattivare il controllo di sicurezza per le risorse che non devono essere controllate:

```
RDEFINE MQADMIN MQD1.NO.CMD.CHECKS
RDEFINE MQADMIN MQD1.NO.CMD.RESC.CHECKS
RDEFINE MQADMIN MQD1.NO.PROCESS.CHECKS
RDEFINE MQADMIN MQD1.NO.NLIST.CHECKS
RDEFINE MQADMIN MQD1.NO.CONTEXT.CHECKS
RDEFINE MQADMIN MQD1.NO.ALTERNATE.USER.CHECKS
```

Quando il gestore code è attivo, è possibile visualizzare le impostazioni di sicurezza correnti immettendo il comando MQSC DISPLAY SECURITY.

Inoltre, è possibile modificare le impostazioni dello switch quando il gestore code è in esecuzione definendo o eliminando il profilo di switch appropriato nella classe MQADMIN. Per apportare le modifiche alle impostazioni dello switch attive, è necessario immettere il comando REFRESH SECURITY per la classe MQADMIN.

Consultare [“Aggiornamento della sicurezza del gestore code su z/OS” a pagina 242](#) per ulteriori dettagli sull'utilizzo dei comandi DISPLAY SECURITY e REFRESH SECURITY.

## Profili utilizzati per controllare l'accesso alle risorse IBM MQ

È necessario definire profili RACF per controllare l'accesso alle risorse IBM MQ, oltre ai profili switch che potrebbero essere stati definiti. Questa raccolta di argomenti contiene informazioni sui profili RACF per diversi tipi di risorsa IBM MQ.

Se non si dispone di un profilo di risorsa definito per un particolare controllo di sicurezza e un utente emette una richiesta che implicherebbe l'esecuzione di tale controllo, IBM MQ nega l'accesso. Non è necessario definire i profili per i tipi di sicurezza relativi agli switch di sicurezza disattivati.

## **Profili per la sicurezza della connessione**

Se la sicurezza della connessione è attiva, è necessario definire i profili nella classe MQCONN e consentire ai gruppi o agli ID utente necessari l'accesso a tali profili, in modo che possano connettersi a IBM MQ.

Per abilitare una connessione da effettuare, è necessario concedere agli utenti RACF l'accesso READ al profilo appropriato. (Se non esiste alcun profilo a livello di gestore code e il gestore code è un membro di un gruppo di condivisione code, è possibile che vengano eseguiti dei controlli rispetto ai profili a livello di gruppo di condivisione code, se la sicurezza è impostata a tale scopo.)

Un profilo di connessione qualificato con un nome gestore code controlla l'accesso a un determinato gestore code e gli utenti a cui è stato fornito l'accesso a questo profilo possono connettersi a tale gestore code. Un profilo di connessione qualificato con il nome del gruppo di condivisione code controlla l'accesso a tutti i gestori code all'interno del gruppo di condivisione code per quel tipo di connessione. Ad esempio, un utente con accesso a QS01.BATCH può utilizzare una connessione batch a qualsiasi gestore code nel gruppo di condivisione code QS01 per cui non è stato definito un profilo a livello di gestore code.

### **Nota:**

1. Per informazioni sugli ID utente controllati per richieste di sicurezza differenti, consultare [“ID utente per il controllo di sicurezza su z/OS”](#) a pagina 231.
2. I controlli di sicurezza a livello di risorsa (RESLEVEL) vengono eseguiti anche in fase di connessione. Per dettagli, consultare [“Profilo di sicurezza RESLEVEL”](#) a pagina 225.

La sicurezza IBM MQ riconosce i seguenti diversi tipi di connessione:

- Connessioni batch (e di tipo batch), tra cui:
  - z/OS Lavori batch
  - Applicazioni TSO
  - Accessi USS
  - Db2Procedure memorizzate
- CICS connessioni
- Connessioni IMS dalle aree di elaborazione delle applicazioni e di controllo
- L'iniziatore di canali IBM MQ

## **Profili di sicurezza della connessione per connessioni batch**

I profili per il controllo delle connessioni di tipo batch sono costituiti dal nome del gestore code o del gruppo di condivisione code seguito dalla parola *BATCH*. Fornire all'ID utente associato allo spazio di indirizzo di connessione l'accesso READ al profilo di connessione.

I profili per la verifica delle connessioni batch e di tipo batch hanno il formato:

```
h1q.BATCH
```

dove h1q può essere qmgr - name (nome gestore code) o qsg - name (nome gruppo di condivisione code). Se si utilizza la sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ ricerca un profilo con prefisso corrispondente al nome del gestore code. Se non ne trova uno, cerca un profilo preceduto dal nome del gruppo di condivisione code. Se non riesce a trovare uno dei profili, la richiesta di connessione ha esito negativo.

Per le richieste di connessione batch o di tipo batch, è necessario consentire all'ID utente associato allo spazio indirizzi di connessione di accedere al profilo di connessione. Ad esempio, il seguente comando

RACF consente agli utenti del gruppo CONNTQM1 di connettersi al gestore code TQM1; questi ID utente potranno utilizzare qualsiasi connessione batch o di tipo batch.

```
RDEFINE MQCONN TQM1.BATCH UACC(NONE)
PERMIT TQM1.BATCH CLASS(MQCONN) ID(CONNTQM1) ACCESS(READ)
```

## **z/OS** Utilizzo di **CHKLOCL** su applicazioni associate localmente

**CHKLOCL** si applica solo alle connessioni effettuate tramite connessioni BATCH e non alle connessioni effettuate da CICS o IMS. Le connessioni effettuate tramite l'iniziatore di canali sono controllate da **CHKCLNT**.

## Panoramica

Se si desidera configurare il gestore code z/OS per richiedere il controllo dell'ID utente e della password per alcune, ma non per tutte, le applicazioni associate localmente, è necessario eseguire ulteriori operazioni di configurazione.

Il motivo è che una volta configurato **CHKLOCL** (*REQUIRED*), le applicazioni batch legacy che utilizzano la chiamata API MQCONN non possono più connettersi al gestore code.

Solo per z/OS, è possibile utilizzare un meccanismo più granulare basato sulla sicurezza di connessione di uno spazio di indirizzo per eseguire il downgrade della configurazione **CHKLOCL** (*REQUIRED*) globale a **CHKLOCL** (*OPTIONAL*) per ID utente definiti in modo specifico. Il meccanismo utilizzato, è descritto nel seguente testo, insieme a un esempio.

Per consentire una maggiore granularità su **CHKLOCL** (*REQUIRED*) rispetto a *EVERYONE*, modificare **CHKLOCL** nello stesso modo in cui si modifica il livello di accesso dell'ID utente associato allo spazio di indirizzi di connessione ai profili di connessione h1q.batch nella classe MQCONN.

Se l'ID utente dello spazio di indirizzo ha solo l'accesso *READ*, che è il minimo richiesto per potersi connettere, la configurazione **CHKLOCL** si applica come scritto.

Se l'ID utente dello spazio di indirizzo ha accesso *UPDATE* (o superiore), la configurazione **CHKLOCL** opera in modalità *FACOLTATIVA*. Ciò significa che non è necessario fornire un ID utente e una password, ma se lo si fa, l'ID utente e la parola d'ordine devono essere una coppia valida.

## La sicurezza della connessione è stata già configurata per il tuo gestore code z/OS

Se si dispone della sicurezza della connessione configurata per il gestore code z/OS e si desidera che **CHKLOCL** (*REQUIRED*) si applichi alle applicazioni WAS associate localmente e non ad altre, effettuare le seguenti operazioni:

1. Iniziare con **CHKLOCL** (*FACOLTATIVO*) come configurazione. Ciò significa che tutti gli ID utente e le password forniti vengono controllati per la validità, ma non sono obbligatori.
2. Elencare tutti gli utenti che hanno accesso ai profili di sicurezza della connessione immettendo il seguente comando:

```
RLIST MQCONN MQ23.BATCH AUTHUSER
```

Questo comando visualizza, ad esempio:

CLASS	NAME		
----	----		
MQCONN	MQ23.BATCH		
USER	ACCESS	ACCESS	COUNT
----	-----	-----	-----
JOHNDOE	READ	000009	
JDOE1	READ	000003	
WASUSER	READ	000000	

3. Per ogni ID utente elencato come avente accesso *READ*, modificare l'accesso in

```
UPDATE:- PERMIT MQ23.BATCH CLASS(MQCONN) ID(JOHNDOE) ACCESS(UPDATE)
```

4. Aggiorna la configurazione IBM MQ in **CHCKLOCL** (*REQUIRED*).

La combinazione dell'accesso UPDATE a MQ23 . BATCH e dell'impostazione corrente indica che si sta utilizzando **CHCKLOCL** (*FACOLTATIVO*).

5. Ora, applicare il comportamento **CHCKLOCL** (*REQUIRED*) a un ID utente specifico, ad esempio WASUSER, in modo che tutte le connessioni provenienti da tale regione debbano fornire un ID utente e una password.

Eeguire questa operazione annullando la modifica apportata in precedenza, immettendo il seguente comando:

```
PERMIT MQ23.BATCH CLASS(MQCONN) ID(WASUSER) ACCESS(READ)
```

### La sicurezza della connessione non è configurata per il gestore code z/OS

In questa situazione, è necessario:

1. Creare i profili di connessione per h1q . BATCH nella classe MQCONN, immettendo il comando:

```
RDEFINE MQCONN MQ23.BATCH UACC(NONE)
```

2. Autorizzare tutti gli ID utente che creano connessioni batch al gestore code in modo che abbiano accesso UPDATE a questo profilo. Questa operazione ignora il requisito **CHCKLOCL** (*REQUIRED*) per l'ID utente e la password al momento della connessione.

Eeguire questa operazione immettendo il seguente comando:

```
PERMIT MQ23.BATCH CLASS(MQCONN) ID(JOHNDOE) ACCESS(UPDATE)
```

Questi includono gli ID utente:

- a. Utilizzato per CSQUTIL, pannelli ISPF e altri strumenti collegati localmente.
  - b. Associato al batch come le connessioni al gestore code. Considerare, ad esempio, Advanced Message Security, IBM Integration Bus, Db2 procedure memorizzate, utenti USS e TSO e applicazioni Java
3. Eliminare il profilo switch per il gestore code immettendo il comando:

```
h1q.NO.CONNECT.CHECKS
```

4. Ora, applicare il comportamento **CHCKLOCL** (*REQUIRED*) a un ID utente specifico, ad esempio WASUSER, in modo che tutte le connessioni provenienti da tale regione debbano fornire un ID utente e una password.

Eeguire questa operazione annullando la modifica apportata in precedenza, immettendo il seguente comando:

```
PERMIT MQ23.BATCH CLASS(MQCONN) ID(WASUSER) ACCESS(READ)
```

### Profili di protezione della connessione per le connessioni CICS .

I profili per la verifica delle connessioni CICS sono composti dal nome del gestore code o del gruppo di condivisione code seguito dalla parola CICS . Fornire all'ID utente associato allo spazio di indirizzo CICS l'accesso READ al profilo di connessione.

I profili per il controllo delle connessioni da CICS hanno il seguente formato:

```
hlq.CICS
```

dove hlq può essere qmgr-name (nome gestore code) o qsg-name (nome gruppo di condivisione code). Se si utilizza la sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ ricerca un profilo con prefisso corrispondente al nome del gestore code. Se non ne trova uno, cerca un profilo preceduto dal nome del gruppo di condivisione code. Se non riesce a trovare uno dei profili, la richiesta di connessione ha esito negativo

Per le richieste di connessione da CICS, è necessario consentire solo l'accesso dell'ID utente dello spazio di indirizzi CICS al profilo di connessione.

Ad esempio, i seguenti comandi RACF consentono all'ID utente dello spazio di indirizzo CICS KCBCICS di collegarsi al gestore code TQM1:

```
RDEFINE MQCONN TQM1.CICS UACC(NONE)
PERMIT TQM1.CICS CLASS(MQCONN) ID(KCBCICS) ACCESS(READ)
```

#### Profili di protezione della connessione per le connessioni IMS .

I profili per la verifica delle connessioni IMS sono composti dal nome del gestore code o del gruppo di condivisione code seguito dalla parola *IMS* . Fornire al controllo IMS e agli ID utente della regione dipendente l'accesso READ al profilo connessione.

I profili per il controllo delle connessioni da IMS hanno il seguente formato:

```
hlq.IMS
```

dove hlq può essere qmgr-name (nome gestore code) o qsg-name (nome gruppo di condivisione code). Se si utilizza la sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ ricerca un profilo con prefisso corrispondente al nome del gestore code. Se non ne trova uno, cerca un profilo preceduto dal nome del gruppo di condivisione code. Se non riesce a trovare uno dei profili, la richiesta di connessione ha esito negativo

Per le richieste di connessione da IMS, consentire l'accesso al profilo di connessione per il controllo IMS e gli ID utente della regione dipendente.

Ad esempio, i seguenti comandi RACF consentono:

- L'ID utente della regione IMS , IMSREG, per connettersi al gestore code TQM1.
- Utenti nel gruppo BMPGRP per inoltrare lavori BMP.

```
RDEFINE MQCONN TQM1.IMS UACC(NONE)
PERMIT TQM1.IMS CLASS(MQCONN) ID(IMSREG,BMPGRP) ACCESS(READ)
```

#### Profili di sicurezza di connessione per l'iniziatore di canali

I profili per il controllo delle connessioni dall'iniziatore di canali sono composti dal nome del gestore code o del gruppo di condivisione code seguito dalla parola *CHIN*. Fornire all'ID utente utilizzato dall'iniziatore del canale l'accesso READ allo spazio di indirizzo del task avviato al profilo di connessione.

I profili per il controllo delle connessioni dall'iniziatore di canali hanno il modulo:

```
hlq.CHIN
```

dove hlq può essere qmgr-name (nome gestore code) o qsg-name (nome gruppo di condivisione code). Se si utilizza la sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ ricerca un profilo con prefisso corrispondente al nome del gestore code. Se non ne trova uno, cerca un profilo preceduto dal nome del gruppo di condivisione code. Se non riesce a trovare uno dei profili, la richiesta di connessione ha esito negativo

Per le richieste di connessione da parte dell'iniziatore di canali, definire l'accesso al profilo di connessione per l'ID utente utilizzato dallo spazio di indirizzo dell'attività avviata dell'iniziatore di canali.

Ad esempio, i seguenti comandi RACF consentono allo spazio indirizzi dell'iniziatore di canali in esecuzione con ID utente DQCTRL di connettersi al gestore code TQM1:

```
RDEFINE MQCONN TQM1.CHIN UACC(NONE)  
PERMIT TQM1.CHIN CLASS(MQCONN) ID(DQCTRL) ACCESS(READ)
```

### Profili per la sicurezza della coda

Se la sicurezza della coda è attiva, è necessario definire i profili nelle classi appropriate e consentire ai gruppi necessari o agli ID utente di accedere a questi profili. I profili di sicurezza della coda sono denominati in base al gestore code o al gruppo di condivisione code e alla coda da aprire.

Se la sicurezza della coda è attiva, è necessario:

- Definire i profili nelle classi **MQQUEUE** o **GMQUEUE** se si utilizzano i profili in maiuscolo.
- Definire i profili nelle classi **MXQUEUE** o **GMXQUEUE** se si utilizzano profili con caratteri maiuscoli e minuscoli.
- Consentire ai gruppi o ID utente necessari l'accesso a questi profili, in modo che possano emettere richieste API IBM MQ che utilizzano le code.

I profili per la sicurezza della coda hanno il formato:


```
hlq.queueaname
```

dove hlq può essere qmgr-name (nome gestore code) o qsg-name (nome del gruppo di condivisione code) e queueaname è il nome della coda da aprire, come specificato nel descrittore dell'oggetto nella chiamata MQOPEN o MQPUT1 .

Un profilo con prefisso del nome gestore code controlla l'accesso a una singola coda su tale gestore code. Un profilo preceduto dal nome del gruppo di condivisione code controlla l'accesso ad una o più code con tale nome su tutti i gestori code all'interno del gruppo di condivisione code oppure l'accesso ad una coda condivisa da qualsiasi gestore code all'interno del gruppo. Questo accesso può essere sovrascritto su un singolo gestore code definendo un profilo a livello di gestore code per tale coda su tale gestore code.

Se il gestore code è un membro di un gruppo di condivisione code e si sta utilizzando la sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ verifica prima un profilo con il prefisso del nome del gestore code. Se non ne trova uno, cerca un profilo preceduto dal nome del gruppo di condivisione code.

Se si utilizzano code condivise, si consiglia di utilizzare la sicurezza a livello di gruppo di condivisione code.

Per i dettagli su come funziona la sicurezza della coda quando il nome della coda è quello di un alias o di una coda modello , consultare “Considerazioni per le code alias” a pagina 194 e “Considerazioni per le code modello” a pagina 195 .



L'accesso RACF richiesto per aprire una coda dipende dalle opzioni MQOPEN o MQPUT1 specificate. Se è codificata più di una delle opzioni MQOO\_\* e MQPMO\_\*, il controllo di sicurezza della coda viene eseguito per l'autorizzazione RACF più elevata richiesta.

*Tabella 31. Livelli di accesso per la sicurezza della coda utilizzando le chiamate MQOPEN o MQPUT1*

<b>opzione MQOPEN o MQPUT1</b>	<b>RACF livello di accesso richiesto per accedere a hlq.queueName</b>
MQOO_SFOGLIA	LETTURA
MQOO_INQUIRE	LETTURA
BIND_ MQOO *	AGGIORNA
MQOO_INPUT_ *	AGGIORNA
MQOO_OUTPUT o MQPUT1	AGGIORNA
MQOO_PASS_ALL_CONTEXT MQPMO_PASS_ALL_CONTEXT	AGGIORNA
MQOO_PASS_IDENTITY_CONTEXT MQPMO_PASS_IDENTITY_CONTEXT	AGGIORNA
MQOO_SAVE_ALL_CONTEXT	AGGIORNA
MQOO_SET_IDENTITY_CONTEXT MQPMO_SET_IDENTITY_CONTEXT	AGGIORNA
MQOO_SET_ALL_CONTEXT MQPMO_SET_ALL_CONTEXT	AGGIORNA
SET MQOO	MODIFICA

Ad esempio, sul gestore code IBM MQ QM77, a tutti gli ID utente nel gruppo RACF PAYGRP deve essere fornito l'accesso per richiamare i messaggi da o inserire i messaggi in tutte le code con nomi che iniziano con 'PAY.'. È possibile eseguire questa operazione utilizzando i comandi RACF riportati di seguito:

```
RDEFINE MQQUEUE QM77.PAY.** UACC(NONE)
PERMIT QM77.PAY.** CLASS(MQQUEUE) ID(PAYGRP) ACCESS(UPDATE)
```

Inoltre, tutti gli ID utente nel gruppo PAYGRP devono avere accesso per inserire messaggi nelle code che non seguono la convenzione di denominazione PAY. Ad esempio:


```
REQUEST_QUEUE_FOR_PAYROLL
SALARY.INCREASE.SERVER
REPLIES.FROM.SALARY.MODEL
```

È possibile eseguire questa operazione definendo i profili per queste code nella classe GMQQUEUE e fornendo l'accesso a tale classe nel modo seguente:

```
RDEFINE GMQQUEUE PAYROLL.EXTRAS UACC(NONE)
ADDMEM(QM77.REQUEST_QUEUE_FOR_PAYROLL,
        QM77.SALARY.INCREASE.SERVER,
        QM77.REPLIES.FROM.SALARY.MODEL)
PERMIT PAYROLL.EXTRAS CLASS(GMQQUEUE) ID(PAYGRP) ACCESS(UPDATE)
```

**Nota:**

1. Se il livello di accesso RACF di un'applicazione a un profilo di sicurezza della coda viene modificato, le modifiche diventano effettive solo per tutti i nuovi handle di oggetto ottenuti (ossia, i nuovi MQOPEN ) per tale coda. Tali handle già esistenti al momento della modifica conservano l'accesso esistente alla coda. Se un'applicazione deve utilizzare il suo livello di accesso modificato per la coda piuttosto che il suo livello di accesso esistente, deve chiudere e riaprire la coda per ogni handle di oggetto che richiede la modifica.
2. Nell'esempio, il nome gestore code QM77 potrebbe essere anche il nome di un gruppo di condivisione code.

Altri tipi di controlli di sicurezza potrebbero verificarsi anche quando la coda viene aperta, in base alle opzioni di apertura specificate e ai tipi di sicurezza attivi.  Vedere anche “Profili per la sicurezza del contesto” a pagina 209 e “Profili per la sicurezza utente alternativa” a pagina 207. Per una tabella di riepilogo che mostra le opzioni aperte e l'autorizzazione di sicurezza necessaria quando la coda, il contesto e la sicurezza utente alternativa sono tutti attivi, consultare [Tabella 36 a pagina 200](#).

Se si utilizza la pubblicazione / sottoscrizione, è necessario considerare quanto segue. Quando una richiesta MQSUB viene elaborata, viene eseguito un controllo di sicurezza per garantire che l'ID utente che effettua la richiesta disponga dell'accesso richiesto per inserire i messaggi sulla coda IBM MQ di destinazione e l'accesso richiesto per sottoscrivere l'argomento IBM MQ .

<i>Tabella 32. Livelli di accesso per la sicurezza della coda utilizzando la chiamata MQSUB</i>	
<b>opzione MQSUB</b>	<b>RACF livello di accesso richiesto per accedere a hlq.queueName</b>
MQSO ALTER, MQSO CREATE e MQSO RESUME	AGGIORNA


**Nota:**

1. hlq.queueName è una coda di destinazione per pubblicazioni. Quando si tratta di una coda gestita, è necessario accedere alla coda modello appropriata da utilizzare per la coda gestita e la coda dinamica creata.
2. È possibile utilizzare una tecnica come questa per la coda di destinazione fornita su una chiamata API MQSUB se si desidera distinguere tra gli utenti che effettuano le sottoscrizioni e gli utenti che richiamano le pubblicazioni dalla coda di destinazione.

 *Considerazioni per le code alias*

Quando si emette una chiamata MQOPEN o MQPUT1 per una coda alias, IBM MQ effettua una verifica della risorsa rispetto al nome della coda specificato nel MQOD (object descriptor) sulla chiamata. Non controlla se all'utente è consentito l'accesso al nome della coda di destinazione.

Ad esempio, una coda alias denominata PAYROLL.REQUEST si risolve in una coda di destinazione di PAY.REQUEST. Se la sicurezza della coda è attiva, è necessario essere autorizzati solo per accedere alla coda PAYROLL.REQUEST. Non viene effettuato alcun controllo per verificare se si è autorizzati ad accedere alla coda PAY.REQUEST.

 *Utilizzo di code alias per distinguere le richieste MQGET e MQPUT*

L'intervallo di chiamate MQI disponibili in un livello di accesso può causare un problema se si desidera limitare l'accesso a una coda per consentire solo la chiamata **MQPUT** o solo la chiamata **MQGET** . Una coda può essere protetta definendo due alias che si risolvono in quella coda: uno che consente alle applicazioni di richiamare i messaggi dalla coda e uno che consente alle applicazioni di inserire messaggi nella coda.

Il seguente testo fornisce un esempio di come è possibile definire le code in IBM MQ:

```
DEFINE QLOCAL(MUST_USE_ALIAS_TO_ACCESS) GET(ENABLED)  
PUT(ENABLED)
```

```

DEFINE QALIAS(USE_THIS_ONE_FOR_GETS) GET(ENABLED)
PUT(DISABLED) TARGQ(MUST_USE_ALIAS_TO_ACCESS)

DEFINE QALIAS(USE_THIS_ONE_FOR_PUTS) GET(DISABLED)
PUT(ENABLED) TARGQ(MUST_USE_ALIAS_TO_ACCESS)

```

È inoltre necessario creare le seguenti definizioni RACF :

```

RDEFINE MQQUEUE hlq.MUST_USE_ALIAS_TO_ACCESS UACC(NONE)
RDEFINE MQQUEUE hlq.USE_THIS_ONE_FOR_GETS UACC(NONE)
RDEFINE MQQUEUE hlq.USE_THIS_ONE_FOR_PUTS UACC(NONE)

```

Quindi, verificare che nessun utente abbia accesso alla coda hlq.MUST\_USE\_ALIAS\_TO\_ACCESS e fornire agli utenti o ai gruppi appropriati l'accesso all'alias. È possibile eseguire questa operazione utilizzando i seguenti comandi RACF :

```

PERMIT hlq.USE_THIS_ONE_FOR_GETS CLASS(MQQUEUE)
ID(GETUSER,GETGRP) ACCESS(UPDATE)
PERMIT hlq.USE_THIS_ONE_FOR_PUTS CLASS(MQQUEUE)
ID(PUTUSER,PUTGRP) ACCESS(UPDATE)

```

Ciò significa che all'ID utente GETUSER e agli ID utente nel gruppo GETGRP è consentito solo ottenere messaggi su MUST\_USE\_ALIAS\_TO\_ACCESS tramite la coda alias USE\_THIS\_ONE\_FOR\_GETS; e all'ID utente PUTUSER e ID utente nel gruppo PUTGRP è consentito solo inserire messaggi nella coda alias USE\_THIS\_ONE\_FOR\_PUTS.

#### Nota:

1. Se si desidera utilizzare una tecnica come questa, è necessario informare gli sviluppatori dell'applicazione, in modo che possano progettare i loro programmi in modo appropriato.
2. È possibile utilizzare una tecnica come questa per la coda di destinazione fornita e la richiesta API MQSUB se si desidera distinguere tra gli utenti che effettuano le sottoscrizioni e gli utenti che 'richiamano' le pubblicazioni dalla coda di destinazione.

#### Considerazioni per le code modello

Per aprire una coda modello, è necessario essere in grado di aprire sia la coda modello stessa che la coda dinamica in cui viene risolta. Definire profili RACF generici per le code dinamiche, incluse le code dinamiche utilizzate dai programmi di utilità IBM MQ .

Quando si apre una coda modello, la sicurezza IBM MQ effettua due controlli di sicurezza della coda:

1. Si è autorizzati ad accedere alla coda modello?
2. Si è autorizzati ad accedere alla coda dinamica in cui si risolve la coda modello?

Se il nome della coda dinamica contiene un carattere asterisco finale (\*), questo \* viene sostituito da una stringa di caratteri generata da IBM MQ, per creare una coda dinamica con un nome univoco. Tuttavia, poiché l'intero nome, inclusa questa stringa generata, viene utilizzato per controllare l'autorizzazione, è necessario definire profili generici per queste code.

Ad esempio, una chiamata MQOPEN utilizza un nome coda modello di CREDIT.CHECK.REPLY.MODEL e un nome coda dinamica di CREDIT.REPLY.\* sul gestore code (o sul gruppo di condivisione code) MQSP.

Per eseguire questa operazione, è necessario immettere i comandi RACF riportati di seguito per definire i profili coda necessari:

```

RDEFINE MQQUEUE MQSP.CREDIT.CHECK.REPLY.MODEL
RDEFINE MQQUEUE MQSP.CREDIT.REPLY.**

```

È inoltre necessario immettere i comandi PERMIT RACF corrispondenti per consentire all'utente di accedere a questi profili.

Un tipico nome di coda dinamica creato da MQOPEN è simile a CREDIT.REPLY.A346EF00367849A0. Il valore preciso dell'ultimo qualificatore è imprevedibile; questo è il motivo per cui è necessario utilizzare profili generici per tali nomi di coda.

Un numero di programmi di utilità IBM MQ inserisce i messaggi nelle code dinamiche. È necessario definire i profili per i seguenti nomi di code dinamiche e fornire l'accesso UPDATE RACF agli ID utente rilevanti (consultare [“ID utente per il controllo di sicurezza su z/OS”](#) a pagina 231 per gli ID utente corretti):

```
SYSTEM.CSQUIL.* (used by CSQUIL)
SYSTEM.CSQOREXX.* (used by the operations and control panels)
SYSTEM.CSQXCMD.* (used by the channel initiator when processing CSQINPX)
CSQ4SAMP.* (used by the IBM MQ supplied samples)
```

Si potrebbe anche considerare di definire un profilo per controllare l'utilizzo del nome della coda dinamica utilizzato per impostazione predefinita nei membri della copia di programmazione dell'applicazione. I copybook forniti da IBM MQ contengono un *DynamicQName* predefinito, che è CSQ.\*. Ciò consente di stabilire un profilo RACF appropriato.

**Nota:** Non consentire ai programmatori di applicazioni di specificare un singolo \* per il nome della coda dinamica. In tal caso, è necessario definire un hlq. \*\* nella classe MQQUEUE e si dovrebbe fornire un accesso ad ampio raggio. Ciò significa che questo profilo potrebbe essere utilizzato anche per altre code non dinamiche che non hanno un più specifico profilo RACF. Pertanto, gli utenti possono accedere alle code a cui non si desidera accedere.

#### z/OS Opzioni di chiusura su code dinamiche permanenti

Se un'applicazione apre una coda dinamica permanente che è stata creata da un'altra applicazione e poi tenta di eliminare tale coda con un'opzione MQCLOSE, vengono applicati alcuni controlli di sicurezza aggiuntivi quando viene effettuato il tentativo.

Tabella 33. Livelli di accesso per le opzioni di chiusura sulle code dinamiche permanenti	
opzione MQCLOSE	RACF livello di accesso richiesto per hlq.queueName
MQCO_DELETE	MODIFICA
MQCO_DELETE_PURGE	MODIFICA

#### z/OS Code remote e di sicurezza

Quando un messaggio viene inserito su una coda remota, la sicurezza della coda implementata dal gestore code locale dipende da come viene specificata la coda remota quando viene aperta.

Vengono applicate le regole seguenti:

1. Se la coda remota è stata definita sul gestore code locale tramite il comando IBM MQ DEFINE QREMOTE, la coda controllata è il nome della coda remota. Ad esempio, se una coda remota è definita sul gestore code MQS1 come segue:

```
DEFINE QREMOTE(BANK7.CREDIT.REFERENCE)
RNAME(CREDIT.SCORING.REQUEST)
RQMNAME(BNK7)
XMITQ(BANK1.TO.BANK7)
```

In questo caso, un profilo per BANK7.CREDIT.REFERENCE deve essere definita nella classe MQQUEUE.

2. Se il *ObjectQMgrName* per la richiesta non si risolve nel gestore code locale, viene eseguito un controllo di sicurezza rispetto al nome del gestore code (remoto) risolto, tranne nel caso di una coda cluster in cui viene eseguito il controllo rispetto al nome della coda cluster.

Ad esempio, la coda di trasmissione BANK1.TO.BANK7 è definito nel gestore code MQS1.

Una richiesta MQPUT1 viene quindi emessa su MQS1 specificando *ObjectName* come

BANK1.INTERBANK.TRANSFERS e un *ObjectQMGrNome* di BANK1.TO.BANK7. In questo caso, l'utente che esegue la richiesta deve avere accesso a BANK1.TO.BANK7.

3. Se si effettua una richiesta MQPUT a una coda e si specifica *ObjectQMGrName* come nome di un alias del gestore code locale, viene controllato solo il nome della coda per la sicurezza, non quello del gestore code.

Quando il messaggio viene inviato al gestore code remoto, potrebbe essere soggetto a ulteriori elaborazioni di sicurezza. Per ulteriori informazioni, consultare [“Sicurezza per la messaggistica remota” a pagina 86.](#)

### *Sicurezza della coda di messaggi non recapitabili*

Considerazioni speciali si applicano alla coda di messaggi non recapitabili, poiché molti utenti devono essere in grado di inserirvi messaggi, ma l'accesso per richiamare i messaggi deve essere strettamente limitato. È possibile ottenere questo risultato applicando diverse autorizzazioni RACF alla coda di messaggi non recapitabili e a una coda alias.

I messaggi non recapitati possono essere inseriti in una coda speciale denominata coda di messaggi non recapitabili. Se si dispone di dati sensibili che potrebbero finire su questa coda, è necessario considerare le implicazioni di sicurezza di questo perché non si desidera che utenti non autorizzati richiamino questi dati.

È necessario consentire a ciascuno dei seguenti elementi di inserire i messaggi nella coda di messaggi non recapitabili:

- Programmi applicativi.
- Lo spazio di indirizzo dell'iniziatore di canali e qualsiasi ID utente MCA. (Se il profilo RESLEVEL non è presente o è definito in modo che gli ID utente del canale vengano controllati, l'ID utente del canale ha anche bisogno dell'autorizzazione per inserire i messaggi nella coda dei messaggi non instradabili).
- CKTI, l'iniziatore dell'attività CICSfornito CICS .
- CSQQTRMN, il controllo trigger IBM MQfornito IMS .

L'unica applicazione che può richiamare i messaggi dalla coda di messaggi non recapitabili deve essere un'applicazione 'speciale' che elabora tali messaggi. Tuttavia, si verifica un problema se si fornisce alle applicazioni l'autorizzazione RACF UPDATE alla coda di messaggi non recapitabili per MQPUT perché possono richiamare automaticamente i messaggi dalla coda utilizzando le chiamate MQGET . Non è possibile disabilitare la coda di messaggi non recapitabili per le operazioni get perché, in tal caso, nemmeno le applicazioni 'speciali' potrebbero richiamare i messaggi.

Una soluzione a questo problema consiste nell'impostare un accesso a due livelli alla coda di messaggi non recapitabili. CKTI, le transazioni dell'agente del canale dei messaggi o lo spazio di indirizzo dell'iniziatore del canale e le applicazioni 'speciali' hanno accesso diretto; altre applicazioni possono accedere alla coda di messaggi non recapitabili solo tramite una coda alias. Questo alias è definito per consentire alle applicazioni di inserire messaggi nella coda di messaggi non recapitabili, ma di non ricevere messaggi da essa.

Questo è il modo in cui potrebbe funzionare:

1. Definire la coda di messaggi non instradati reale con gli attributi PUT (ENABLED) e GET (ENABLED), come mostrato nell'esempio thlqual.SCSQPROC(CSQ4INYG).
2. Concedere l'autorità RACF UPDATE per la coda di messaggi non recapitabili ai seguenti ID utente:
  - Gli ID utente con cui vengono eseguiti CKTI e gli MCA o lo spazio di indirizzo dell'iniziatore di canali.
  - Gli ID utente associati all'applicazione di elaborazione della coda di messaggi non instradabili 'speciale'.
3. Definire una coda alias che si risolva nella coda di messaggi non instradabili reale, ma fornire alla coda alias questi attributi: PUT (ENABLED) e GET (DISABLED). Assegnare alla coda alias un nome con la stessa radice del nome della coda di messaggi non recapitabili, ma accodare i caratteri ". PUT" a questa radice. Ad esempio, se il nome della coda di messaggi non recapitabili è hlq.DEAD.QUEUE, il nome della coda alias è hlq.DEAD.QUEUE.PUT.

4. Per inserire un messaggio nella coda di messaggi non instradabili, un'applicazione utilizza la coda alias. Questo è ciò che l'applicazione deve fare:
- Richiamare il nome della coda di messaggi non recapitabili reale. A tale scopo, apre l'oggetto del gestore code utilizzando MQOPEN ed emette un MQINQ per ottenere il nome della coda di messaggi non instradabili.
  - Creare il nome della coda alias aggiungendo i caratteri '.PUT' a questo nome, in questo caso, hlq.DEAD.QUEUE.PUT.
  - Aprire la coda alias, hlq.DEAD.QUEUE.PUT.
  - Inserire il messaggio nella coda di messaggi non instradabili reale emettendo un comando MQPUT rispetto alla coda alias.
5. Fornire l'ID utente associato all'autorizzazione RACF UPDATE dell'applicazione all'alias, ma nessun accesso (autorizzazione NONE) alla coda di messaggi non recapitabili reale. Questo vuol dire che:
- L'applicazione può inserire messaggi nella coda di messaggi non recapitabili utilizzando la coda alias.
  - L'applicazione non può richiamare i messaggi dalla coda di messaggi non recapitabili utilizzando la coda alias perché la coda alias è disabilitata per operazioni di acquisizione.
- L'applicazione non può richiamare alcun messaggio dalla coda di messaggi non instradabili reale perché dispone dell'autorità RACF corretta.

Tabella 34 a pagina 198 riassume l'autorizzazione RACF richiesta per i vari partecipanti a questa soluzione.

<i>Tabella 34. Autorizzazione RACF alla coda di messaggi non instradabili e al relativo alias</i>		
<b>ID utente associato</b>	<b>Coda di messaggi non instradabili reale (hlq.DEAD.QUEUE)</b>	<b>Coda di messaggi non recapitabili alias (hlq.DEAD.QUEUE.PUT)</b>
Spazio di indirizzo dell'iniziatore di canali o MCA e CKTI	AGGIORNA	NESSUNO
Applicazione 'Speciale' (per l'elaborazione della coda di messaggi non recapitabili)	AGGIORNA	NESSUNO
ID utente applicazione scritti dall'utente	NESSUNO	AGGIORNA

Se si utilizza questo metodo, l'applicazione non può determinare la lunghezza massima del messaggio (MAXMSGL) della coda di messaggi non recapitabili. Ciò si verifica perché l'attributo MAXMSGL non può essere richiamato da una coda alias. Pertanto, l'applicazione deve presumere che la lunghezza massima del messaggio sia 100 MB, la dimensione massima supportata da IBM MQ for z/OS. La coda di messaggi non recapitabili reale deve essere definita anche con un attributo MAXMSGL di 100 MB.

**Nota:** I programmi di applicazione scritti dall'utente normalmente non utilizzano l'autorizzazione utente alternativa per inserire i messaggi nella coda di messaggi non recapitabili. Ciò riduce il numero di ID utente che hanno accesso alla coda di messaggi non recapitabili.

#### *Sicurezza della coda di sistema*

È necessario configurare l'accesso RACF per consentire a determinati ID utente di accedere a particolari code di sistema.

Molte delle code di sistema sono accessibili dalle parti ausiliarie di IBM MQ:

- Programma di utilità CSQUTIL
- Le operazioni e i pannelli di controllo
- Lo spazio di indirizzo dell'iniziatore di canali (incluso il daemon Pub / Sot accodato)

- **V 9.0.1** Il server mqweb Liberty , utilizzato da MQ Console e REST API (servermqweb) .

**V 9.0.1**

Gli ID utente con cui queste esecuzioni devono disporre dell'accesso RACF a tali code, come mostrato in Tabella 35 a pagina 199.

*Tabella 35. Accesso richiesto alle code SYSTEM da IBM MQ*

<b>Coda SYSTEM</b>	<b>CSQUTIL</b>	<b>CSQOUTIL</b>	<b>WLP per server MQ</b>	<b>Operazioni e pannelli di controllo</b>	<b>Iniziatore di canali per l'accodamento o distribuito</b>
SYSTEM.ADMIN.CHANNEL.EVENT	-	-	-	-	AGGIORNA
SYSTEM.ADMIN.COMMAND.QUEUE	-	-	AGGIORNA	-	-
SYSTEM.BROKER.ADMIN.STREAM	-	-	-	-	MODIFICA
SYSTEM.BROKER.CONTROL.QUEUE	-	-	-	-	MODIFICA
SYSTEM.BROKER.DEFAULT.STREAM	-	-	-	-	MODIFICA
SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS	-	-	-	-	AGGIORNA
SYSTEM.CHANNEL.INITQ	-	-	-	-	AGGIORNA
SYSTEM.CHANNEL.SYNCQ	-	-	-	-	AGGIORNA
SYSTEM.CLUSTER.COMMAND.QUEUE	-	-	-	-	MODIFICA
SYSTEM.CLUSTER.REPOSITORY.QUEUE	-	-	-	-	AGGIORNA
SYSTEM.CLUSTER.TRANSMIT.QUEUE	-	-	-	-	MODIFICA
SYSTEM.COMMAND.INPUT	AGGIORNA	-	-	AGGIORNA	AGGIORNA
SYSTEM.COMMAND.REPLY.*	-	-	-	-	AGGIORNA
SYSTEM.COMMAND.REPLY.MODEL	AGGIORNA	-	-	AGGIORNA	AGGIORNA
SYSTEM.CSQOREXX.*	-	-	-	AGGIORNA	-
SYSTEM.CSQUTIL.*	AGGIORNA	-	-	-	-
SYSTEM.CSQXCMD.*	-	-	-	-	AGGIORNA
SYSTEM.HIERARCHY.STATE	-	-	-	-	AGGIORNA
SYSTEM.INTER.QMGR.CONTROL	-	-	-	-	AGGIORNA
SYSTEM.INTER.QMGR.PUBS	-	-	-	-	AGGIORNA
SYSTEM.INTER.QMGR.FANREQ	-	-	-	-	AGGIORNA
SYSTEM.PROTECTION.ERROR.QUEUE	-	-	-	-	AGGIORNA
SYSTEM.PROTECTION.POLICY.QUEUE	-	UPDATE (1)	-	-	LETTURA
SYSTEM.QSG.CHANNEL.SYNCQ	-	-	-	-	AGGIORNA
SYSTEM.QSG.TRANSMIT.QUEUE	-	-	-	-	AGGIORNA

Tabella 35. Accesso richiesto alle code SYSTEM da IBM MQ (Continua)

Coda SYSTEM	CSQUTIL	CSQOUTIL	WLP per server MQ	Operazioni e pannelli di controllo	Inziatore di canali per l'accodamento o distribuito
SYSTEM.REST.REPLY.QUEUE	-	-	AGGIORNA	-	-



Guida di riferimento rapido per l'accesso alla sicurezza delle risorse API

Un riepilogo delle opzioni MQOPEN, MQPUT1, MQSUB e MQCLOSE e l'accesso richiesto dai diversi tipi di sicurezza delle risorse.

Tabella 36. Le opzioni MQOPEN, MQPUT1, MQSUB e MQCLOSE e l'autorizzazione di sicurezza richiesta. I callout mostrate come questa (1) fanno riferimento alle note che seguono questa tabella.

RACF Classe:	Livello minimo di accesso RACF richiesto			
	MXTOPIC	MQQUEUE o MXQUEUE (1)	MQADMIN o MXADMIN	MQADMIN o MXADMIN
	RACF Profilo: (15 o 16)	(2)	(3)	(4)
opzione MQOPEN				
MQOO_INQUIRE		READ (5)	Nessun controllo	Nessun controllo
MQOO_SFOGLIA		LETTURA	Nessun controllo	Nessun controllo
MQOO_INPUT_*		AGGIORNA	Nessun controllo	Nessun controllo
MQOO_SAVE_ALL_CONTEXT (6)		AGGIORNA	Nessun controllo	Nessun controllo
MQOO_OUTPUT (USAGE = NORMAL) (7)		AGGIORNA	Nessun controllo	Nessun controllo
MQOO_PASS_IDENTITY_CONTEXT (8)		AGGIORNA	LETTURA	Nessun controllo
MQOO_PASS_ALL_CONTEXT (8) (9)		AGGIORNA	LETTURA	Nessun controllo
MQOO_SET_IDENTITY_CONTEXT (8) (9)		AGGIORNA	AGGIORNA	Nessun controllo
MQOO_SET_ALL_CONTEXT (8) (10)		AGGIORNA	CONTROL	Nessun controllo
MQOO_OUTPUT (UTILIZZO (XMITQ) (11)		AGGIORNA	CONTROL	Nessun controllo
MQOO_OUTPUT (oggetto argomento)	UPDATE (16)			
MQOO_OUTPUT (coda alias per oggetto argomento)	UPDATE (16)	AGGIORNA		
SET MQOO		MODIFICA	Nessun controllo	Nessun controllo
MQOO_ALTERNATE_USER_AUTHORITY		(12)	(12)	AGGIORNA



Tabella 36. Le opzioni MQOPEN, MQPUT1, MQSUB e MQCLOSE e l'autorizzazione di sicurezza richiesta. I callout mostrate come questa (1) fanno riferimento alle note che seguono questa tabella. (Continua)

Livello minimo di accesso RACF richiesto				
RACF Classe:	MXTOPIC	MQQUEUE o MXQUEUE ( 1 )	MQADMIN o MXADMIN	MQADMIN o MXADMIN
RACF Profilo:	( 15 o 16 )	( 2 )	( 3 )	( 4 )
opzioneMQPUT1				
Inserimento in una coda normale ( 7 )		AGGIORNA	Nessun controllo	Nessun controllo
MQPMO_PASS_IDENTITY_CONTEXT		AGGIORNA	LETTURA	Nessun controllo
MQPMO_PASS_ALL_CONTEXT		AGGIORNA	LETTURA	Nessun controllo
MQPMO_SET_IDENTITY_CONTEXT		AGGIORNA	AGGIORNA	Nessun controllo
MQPMO_SET_ALL_CONTEXT		AGGIORNA	CONTROL	Nessun controllo
OUTPUT MQOO		AGGIORNA	CONTROL	Nessun controllo
Inserimento in una coda di trasmissione ( 11 )		AGGIORNA	CONTROL	Nessun controllo
MQOO_OUTPUT (oggetto argomento)	UPDATE ( 16 )			
MQOO_OUTPUT (coda alias per oggetto argomento)	UPDATE ( 16 )	AGGIORNA		
MQPMO_ALTERNATE_USER_AUTHORITY		( 13 )	( 13 )	AGGIORNA
Opzione MQCLOSE				
MQCO_DELETE ( 14 )		MODIFICA	Nessun controllo	Nessun controllo
MQCO_DELETE_PURGE ( 14 )		MODIFICA	Nessun controllo	Nessun controllo
MQCO_REMOVE_SUB	ALTER ( 15 )			
opzione MQSUB				
CREA MQSO	ALTER ( 15 )	( 17 )	( 18 )	
ALTER MQSO	ALTER ( 15 )	( 17 )	( 18 )	
RESUME MQSO	READ ( 15 )	( 17 )	Nessun controllo	
MQSO_ALTERNATE_USER_AUTHORITY				AGGIORNA
MQSO_SET_IDENTITY_CONTEXT			( 18 )	

**Nota:**

1. Questa opzione non è limitata alle code. Utilizzare la classe MQNLIST o MXNLIST per gli elenchi nomi e la classe MQPROC o MXPROC per i processi.

2. Utilizzare il profilo RACF : hlq.resourcename
3. Utilizzare il profilo RACF : hlq.CONTEXT.queuename
4. Utilizzare il profilo RACF : hlq.ALTERNATE.USER.alternateuserid  
 alternateuserid è l'identificativo utente specificato nel campo *AlternateUserId* del descrittore oggetto. Notare che per questo controllo vengono utilizzati fino a 12 caratteri del campo *AlternateUserId*, a differenza di altri controlli in cui vengono utilizzati solo i primi 8 caratteri di un identificativo utente.
5. Non viene effettuato alcun controllo quando si apre il gestore code per le richieste.
6. Anche MQOO\_INPUT\_\* deve essere specificato. È valido per una coda locale, modello o alias.
7. Questo controllo viene eseguito per una coda locale o modello che ha un attributo della coda **Usage** di MQUS\_NORMAL e anche per un alias o una coda remota (definita per il gestore code connesso). Se la coda è una coda remota aperta specificando un *ObjectQMgrName* (non il nome del gestore code connesso) esplicitamente, il controllo viene eseguito sulla coda con lo stesso nome di *ObjectQMgrName* (che deve essere una coda locale con un attributo della coda **Usage** MQUS\_TRANSMISSION).
8. È necessario specificare anche MQOO\_OUTPUT.
9. MQOO\_PASS\_IDENTITY\_CONTEXT è implicito anche da questa opzione.
10. MQOO\_PASS\_IDENTITY\_CONTEXT, MQOO\_PASS\_ALL\_CONTEXT e MQOO\_SET\_IDENTITY\_CONTEXT sono impliciti anche da questa opzione.
11. Questo controllo viene eseguito per una coda locale o modello che ha un attributo di coda **Usage** MQUS\_TRANSMISSION e viene aperto direttamente per l'output. Non si applica se una coda remota è in fase di apertura.
12. È necessario specificare anche almeno uno tra MQOO\_INQUIRE, MQOO\_BROWSE, MQOO\_INPUT\_\*, MQOO\_OUTPUT o MQOO\_SET. Il controllo effettuato è lo stesso di quello per le altre opzioni specificate.
13. Il controllo effettuato è lo stesso di quello per le altre opzioni specificate.
14. Ciò si applica solo per le code dinamiche permanenti che sono state aperte direttamente, ossia non aperte tramite una coda modello. Non è richiesta alcuna sicurezza per eliminare una coda dinamica temporanea.
15. Utilizzare RACF profile hlq.SUBSCRIBE.topicname.
16. Utilizzare il RACF profilo hlq.PUBLISH.topicname.
17. Se nella richiesta MQSUB è stata specificata una coda di destinazione per le pubblicazioni a cui inviare le pubblicazioni, viene eseguito un controllo di sicurezza su tale coda per garantire che si disponga dell'autorizzazione all'immissione per tale coda.
18. Se nella richiesta MQSUB, con le opzioni MQSO\_CREATE o MQSO\_ALTER specificate, si desidera impostare uno qualsiasi dei campi del contesto di identità nella struttura MQSD, è necessario specificare anche l'opzione MQSO\_SET\_IDENTITY\_CONTEXT e l'autorizzazione appropriata per il profilo di contesto per la coda di destinazione.

## Profili per la sicurezza degli argomenti

Se la sicurezza dell'argomento è attiva, è necessario definire i profili nelle classi appropriate e consentire ai gruppi o ID utente necessari l'accesso a tali profili.

Il concetto di sicurezza dell'argomento all'interno di una struttura di argomenti è descritto in [Sicurezza di pubblicazione / sottoscrizione](#).

Se la sicurezza dell'argomento è attiva, è necessario eseguire le seguenti operazioni:

- Definire i profili nelle classi **MXTOPIC** o **GMXTOPIC**.
- Consentire ai gruppi o ID utente necessari l'accesso a questi profili, in modo che possano emettere richieste API IBM MQ che utilizzano argomenti.

I profili per la sicurezza degli argomenti hanno il formato:

hlq.SUBSCRIBE.topicname  
hlq.PUBLISH.topicname

dove

- hlq è qmgr-name (nome gestore code) o qsg-name (nome gruppo di condivisione code).
- topicname è il nome del nodo di gestione argomenti nella struttura ad albero degli argomenti, associato all'argomento che viene sottoscritto tramite una chiamata MQSUB o pubblicato tramite una chiamata MQOPEN.

Un profilo preceduto dal nome gestore code controlla l'accesso a un singolo argomento su tale gestore code. Un profilo preceduto dal nome del gruppo di condivisione code controlla l'accesso a uno o più argomenti con tale nome su tutti i gestori code all'interno del gruppo di condivisione code. Questo accesso può essere sovrascritto su un singolo gestore code definendo un profilo a livello di gestore code per tale argomento su tale gestore code.

Se il gestore code è un membro di un gruppo di condivisione code e si sta utilizzando la sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ verifica prima un profilo con il prefisso del nome del gestore code. Se non ne trova uno, cerca un profilo preceduto dal nome del gruppo di condivisione code.

## Sottoscrivere

Per sottoscrivere un argomento, è necessario accedere sia all'argomento che si sta tentando di sottoscrivere, sia alla coda di destinazione per le pubblicazioni.

Quando si emette una richiesta MQSUB, si verificano le seguenti verifiche di sicurezza:

- Indica se si dispone del livello di accesso appropriato per la sottoscrizione a tale argomento e se la coda di destinazione (se specificata) è aperta per l'output
- Indica se si dispone del livello di accesso appropriato a quella coda di destinazione.

Azione	Livello di accesso richiesto
Da MQSUB a un argomento	Accesso RACF richiesto al profilo <i>hlq.SUBSCRIBE.topicname</i> nella classe MXTOPIC
MQSO_CREATE e MQSO_ALTER	MODIFICA
RESUME MQSO	LETTURA
MQSUB - autorizzazione aggiuntiva alle code di destinazione non gestite.	Accesso RACF richiesto al profilo <i>hlq.CONTEXT.queueName</i> nella classe MQADMIN o MXADMIN
MQSO_CREATE, MQSO_ALTER e MQSO_RESUME	AGGIORNA
	Accesso RACF richiesto al profilo <i>hlq.queueName</i> nella classe MQQUEUE o MXQUEUE
MQSO_CREATE e MQSO_ALTER	AGGIORNA
	Accesso RACF richiesto al profilo <i>hlq.ALTERNATE.USER.alternateuserid</i> nella classe MQADMIN o MXADMIN
MQSO_ALTERNATE_USER_AUTHORITY	AGGIORNA

## Considerazioni per le code gestite per le sottoscrizioni

Viene eseguito un controllo di sicurezza per verificare se si è autorizzati a sottoscrivere l'argomento. Tuttavia, non viene eseguito alcun controllo di sicurezza quando viene creata la coda gestita o per determinare se si dispone dell'accesso per inserire i messaggi in questa coda di destinazione.

Non è possibile chiudere l'eliminazione di una coda gestita.

Le code modello utilizzate sono: SYSTEM . DURABLE . MODEL . QUEUE e SYSTEM . NDURABLE . MODEL . QUEUE.

Le code gestite create da queste code modello sono nel formato SYSTEM . MANAGED . DURABLE . A346EF00367849A0 e SYSTEM . MANAGED . NDURABLE . A346EF0036785EA0 dove l'ultimo qualificatore è imprevedibile.

Non fornire alcun accesso utente a queste code. Le code possono essere protette utilizzando profili generici nel formato SYSTEM . MANAGED . DURABLE . \* e SYSTEM . MANAGED . NDURABLE . \* senza autorizzazioni concesse.

I messaggi possono essere recuperati da queste code utilizzando l'handle restituito sulla richiesta MQSUB.

Se si emette esplicitamente una chiamata MQCLOSE per una sottoscrizione con l'opzione MQCO\_REMOVE\_SUB specificata e non si crea la sottoscrizione che si sta chiudendo con questo handle, viene eseguito un controllo di sicurezza al momento della chiusura per assicurarsi di disporre dell'autorità corretta per eseguire l'operazione.

<i>Tabella 38. Livello di accesso richiesto ai profili per la sicurezza dell'argomento per la chiusura di un'operazione di sottoscrizione</i>	
<b>Azione</b>	<b>Livello di accesso richiesto</b>
MQCLOSE (di una sottoscrizione)	Accesso RACF richiesto al profilo <i>hlq.SUBSCRIBE.topicname</i> nella classe MXTOPIC
MQCO_REMOVE_SUB	MODIFICA

## Pubblica

Per pubblicare su un argomento è necessario accedere all'argomento e, se si utilizzano le code alias, anche alla coda alias.

<i>Tabella 39. Livello di accesso richiesto ai profili per la sicurezza dell'argomento per un'operazione di pubblicazione</i>	
<b>Azione</b>	<b>Livello di accesso richiesto</b>
MQOPEN (di un argomento)	Accesso RACF richiesto al profilo <i>hlq.PUBLISH.topicname</i> nella classe MXTOPIC
MQOO_OUTPUT o MQPUT1	AGGIORNA
MQOPEN (coda alias per argomento)	Accesso RACF richiesto al profilo <i>hlq.queueName</i> nella classe MQQUEUE o MXQUEUE per la coda alias
MQOO_OUTPUT o MQPUT1	AGGIORNA

Per i dettagli su come funziona la sicurezza dell'argomento quando una coda alias che si risolve in un nome argomento viene aperta per la pubblicazione, vedere [“Considerazioni per le code alias che si risolvono in argomenti per un'operazione di pubblicazione”](#) a pagina 205.

Quando si considerano le code alias utilizzate per le code di destinazione per le limitazioni PUT o GET, consultare [“Considerazioni per le code alias”](#) a pagina 194.

Se il livello di accesso RACF di un'applicazione a un profilo di sicurezza dell'argomento viene modificato, le modifiche diventano effettive solo per i nuovi handle di oggetto ottenuti (ossia, un nuovo MQSUB o MQOPEN) per tale argomento. Tali handle già esistenti al momento della modifica conservano l'accesso esistente all'argomento. Inoltre, i sottoscrittori esistenti conservano l'accesso a tutte le sottoscrizioni che hanno già effettuato.

## Considerazioni per le code alias che si risolvono in argomenti per un'operazione di pubblicazione

Quando si emette una chiamata MQOPEN o MQPUT1 per una coda alias che si risolve in un argomento, IBM MQ effettua due controlli delle risorse:

- Il primo rispetto al nome della coda alias specificato nel descrittore dell'oggetto (MQOD) sulla chiamata MQOPEN o MQPUT1 .
- Il secondo rispetto all'argomento in cui si risolve la coda alias

È necessario essere consapevoli che questo comportamento è diverso da quello che si ottiene quando le code alias si risolvono in altre code. È necessario l'accesso corretto a entrambi i profili per poter procedere con l'azione di pubblicazione.

## Sicurezza argomenti di sistema

I seguenti argomenti di sistema sono acceduti dallo spazio di indirizzo dell'iniziatore di canali.

Gli ID utente con cui viene eseguita questa esecuzione devono disporre dell'accesso RACF a queste code, come mostrato in [Tabella 40 a pagina 205](#).

Tabella 40. Accesso richiesto agli argomenti SYSTEM		
Argomento SYSTEM	Profilo	Iniziatore di canali per accodamento distribuito
SYSTEM.BROKER.ADMIN.STREAM	PUBLISH.topicname	AGGIORNA
SYSTEM.BROKER.ADMIN.STREAM	SUBSCRIBE.topicname	MODIFICA

## Profili per processi

Se la sicurezza del processo è attiva, è necessario definire i profili nelle classi appropriate e consentire ai gruppi o ID utente necessari l'accesso a tali profili.

Se la sicurezza del processo è attiva, è necessario:

- Definire i profili nelle classi **MQPROC** o **GMQPROC** se si utilizzano i profili in maiuscolo.
- Definire i profili nelle classi **MXPROC** o **GMXPROC** se si utilizzano profili con caratteri maiuscoli e minuscoli.
- Consentire ai gruppi o ID utente necessari l'accesso a questi profili, in modo che possano emettere richieste API IBM MQ che utilizzano processi.

I profili per i processi hanno il formato:

hlq.processname
-----------------

dove hlq può essere qmgr - name (nome gestore code) o qsg - name (nome gruppo di condivisione code) e processname è il nome del processo che si sta aprendo.

Un profilo preceduto dal nome gestore code controlla l'accesso a una singola definizione di processo su tale gestore code. Un profilo preceduto dal nome del gruppo di condivisione code controlla l'accesso a

una o più definizioni di processo con tale nome su tutti i gestori code all'interno del gruppo di condivisione code. Questo accesso può essere sovrascritto su un singolo gestore code definendo un profilo a livello di gestore code per tale definizione di processo su tale gestore code.

Se il gestore code è un membro di un gruppo di condivisione code e si sta utilizzando la sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ verifica prima un profilo con il prefisso del nome del gestore code. Se non ne trova uno, cerca un profilo preceduto dal nome del gruppo di condivisione code.

La seguente tabella mostra l'accesso richiesto per aprire un processo.

Tabella 41. Livelli di accesso per la sicurezza del processo	
opzione MQOPEN	RACF livello di accesso richiesto per hlq.processname
MQOO_INQUIRE	LETTURA

Ad esempio, sul gestore code MQS9, il gruppo RACF INQVPRC deve essere in grado di eseguire un'interrogazione (MQINQ) su tutti i processi che iniziano con la lettera V. Le definizioni di RACF sono:

```
RDEFINE MQPROC MQS9.V* UACC(NONE)
PERMIT MQS9.V* CLASS(MQPROC) ID(INQVPRC) ACCESS(READ)
```

Anche la sicurezza dell'utente alternativo potrebbe essere attiva, a seconda delle opzioni di apertura specificate quando viene aperto un oggetto definizione processo.

## Profili per elenchi nomi

Se la sicurezza dell'elenco nomi è attiva, si definiscono i profili nelle classi appropriate e si fornisce ai gruppi o ID utente necessari l'accesso a tali profili.

Se la sicurezza dell'elenco nomi è attiva, è necessario:

- Definire i profili nelle classi **MQNLIST** o **GMQNLIST** se si utilizzano i profili in maiuscolo.
- Definire i profili nelle classi **MXNLIST** o **GMXNLIST** se si utilizzano profili con caratteri maiuscoli e minuscoli.
- Consentire ai gruppi o ID utente necessari l'accesso a questi profili.

I profili per gli elenchi nomi assumono il formato:

```
hlq.namelistname
```

dove hlq può essere qmgr-name (nome gestore code) o qsg-name (nome gruppo di condivisione code) e namelistname è il nome dell'elenco nomi che si sta aprendo.

Un profilo preceduto dal nome gestore code controlla l'accesso a un singolo elenco nomi su tale gestore code. Un profilo preceduto dal nome del gruppo di condivisione code controlla l'accesso ad uno o più elenchi nomi con tale nome su tutti i gestori code all'interno del gruppo di condivisione code. Questo accesso può essere sovrascritto su un singolo gestore code definendo un profilo a livello di gestore code per tale elenco nomi su tale gestore code.

Se il gestore code è un membro di un gruppo di condivisione code e si sta utilizzando la sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ verifica prima un profilo con il prefisso del nome del gestore code. Se non ne trova uno, cerca un profilo preceduto dal nome del gruppo di condivisione code.

La seguente tabella mostra l'accesso richiesto per l'apertura di un elenco nomi.

Tabella 42. Livelli di accesso per la sicurezza dell'elenco nomi	
opzione MQOPEN	RACF livello di accesso richiesto per hlq.namelistname
MQOO_INQUIRE	LETTURA

Ad esempio, sul gestore code (o sul gruppo di condivisione code) PQM3, il gruppo RACF DEPT571 deve essere in grado di analizzare (MQINQ) su questi elenchi nomi:

- Tutti gli elenchi nomi che iniziano con "DEPT571".
- PRINTER/DESTINATIONS/DEPT571
- AGENZIA/RICHIESTA/CODE
- WAREHOUSE.BROADCAST

Le definizioni RACF per eseguire questa operazione sono:

```
RDEFINE MQNLIST PQM3.DEPT571.** UACC(NONE)
PERMIT PQM3.DEPT571.** CLASS(MQNLIST) ID(DEPT571) ACCESS(READ)

RDEFINE GMQNLIST NLISTS.FOR.DEPT571 UACC(NONE)
  ADDMEM(PQM3.PRINTER/DESTINATIONS/DEPT571,
          PQM3.AGENCY/REQUEST/QUEUES,
          PQM3.WAREHOUSE.BROADCAST)
PERMIT NLISTS.FOR.DEPT571 CLASS(GMQNLIST) ID(DEPT571) ACCESS(READ)
```

La sicurezza dell'utente alternativo potrebbe essere attiva, in base alle opzioni specificate quando viene aperto un oggetto elenco nomi.

## Sicurezza elenco nomi di sistema

Molte delle liste dei nomi di sistema sono accessibili dalle parti ausiliarie di IBM MQ:

- Programma di utilità CSQUTIL
- Le operazioni e i pannelli di controllo
- Lo spazio di indirizzo dell'iniziatore di canali (incluso il daemon di pubblicazione / sottoscrizione accodata)

Gli ID utente con cui queste esecuzioni devono disporre dell'accesso RACF a questi elenchi nomi, come mostrato in [Tabella 43 a pagina 207](#).

Tabella 43. Accesso richiesto agli elenchi nomi SYSTEM da parte di IBM MQ			
Elenco nomi SISTEMA	CSQUTIL	Operazioni e pannelli di controllo	Iniziatore di canali per l'accodamento distribuito
SYSTEM.QPUBSUB.QUEUE.NAMELIST	-	-	LETTURA
SYSTEM.QPUBSUB.SUBPOINT.NAMELIST	-	-	LETTURA

## Profili per la sicurezza utente alternativa

Se è attiva la sicurezza utente alternativa, è necessario definire i profili nelle classi appropriate e consentire ai gruppi o ID utente necessari l'accesso a tali profili.

Per ulteriori informazioni su *AlternateUserId*, consultare [AlternateUserID \(MQCHAR12\)](#).

Se è attiva la sicurezza dell'utente alternativo, è necessario:

- Definire i profili nelle classi MQADMIN o GMQADMIN se si utilizzano i profili in maiuscolo.

- Definire i profili nelle classi MXADMIN o GMXADMIN se si utilizzano profili con caratteri maiuscoli e minuscoli.

Consentire ai gruppi o agli ID utente necessari l'accesso a questi profili, in modo che possano utilizzare le opzioni ALTERNATE\_USER\_AUTHORITY quando l'oggetto viene aperto.

I profili per la sicurezza dell'utente alternativo possono essere specificati a livello di sottosistema o a livello di gruppo di condivisione della coda e possono avere il formato seguente:

```
hlq.ALTERNATE.USER.alternateuserid
```

Dove hlq può essere qmgr-name (nome gestore code) o qsg-name (nome gruppo di condivisione code) e alternateuserid è il valore del campo *AlternateUserId* nel descrittore oggetto.

Un profilo con prefisso del nome gestore code controlla l'utilizzo di un ID utente alternativo su tale gestore code. Un profilo con il prefisso del nome del gruppo di condivisione code controlla l'utilizzo di un ID utente alternativo su tutti i gestori code all'interno del gruppo di condivisione code. Questo ID utente alternativo può essere utilizzato su qualsiasi gestore code all'interno del gruppo di condivisione code da un utente che dispone dell'accesso corretto. Questo accesso può essere sovrascritto su un singolo gestore code definendo un profilo a livello di gestore code per l'ID utente alternativo su tale gestore code.

Se il gestore code è un membro di un gruppo di condivisione code e si sta utilizzando la sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ verifica prima un profilo con il prefisso del nome del gestore code. Se non ne trova uno, cerca un profilo preceduto dal nome del gruppo di condivisione code.

La seguente tabella mostra l'accesso quando si specifica un'opzione utente alternativa.

<i>Tabella 44. Livelli di accesso per la sicurezza di un utente alternativo</i>	
<b>opzione MQOPEN, MQSUB o MQPUT1</b>	<b>Livello di accesso RACF richiesto</b>
MQOO_ALTERNATE_USER_AUTHORITY MQSO_ALTERNATE_USER_AUTHORITY MQPMO_ALTERNATE_USER_AUTHORITY	AGGIORNA

Oltre ai controlli di sicurezza degli utenti alternativi, possono essere eseguiti anche altri controlli di sicurezza per la coda, il processo, l'elenco nomi e la sicurezza del contesto. L'ID utente alternativo, se fornito, viene utilizzato solo per i controlli di sicurezza sulle risorse della coda, della definizione del processo o dell'elenco nomi. Per i controlli di sicurezza del contesto e dell'utente alternativi, l'ID utente che richiede l'utilizzo del controllo. Per dettagli su come vengono gestiti gli ID utente, consultare [“ID utente per il controllo di sicurezza su z/OS” a pagina 231](#). Per una tabella di riepilogo che mostra le opzioni di apertura e i controlli di sicurezza richiesti quando la coda, il contesto e la sicurezza dell'utente alternativo sono tutti attivi, consultare [Tabella 36 a pagina 200](#).

Un profilo utente alternativo fornisce all'ID utente richiedente l'accesso alle risorse associate all'ID utente specificato nell'ID utente alternativo. Ad esempio, il server payroll in esecuzione con l'ID utente PAYSERV sul gestore code QMPY elabora le richieste dagli ID utente del personale, che iniziano tutte con PS. Per fare in modo che il lavoro eseguito dal server di retribuzione venga eseguito con l'ID utente dell'utente richiedente, viene utilizzata l'autorizzazione utente alternativa. Il server payroll conosce l'ID utente da specificare come ID utente alternativo poiché i programmi richiedenti generano messaggi utilizzando l'opzione di inserimento del messaggio MQPMO\_DEFAULT\_CONTEXT. Consultare [“ID utente per il controllo di sicurezza su z/OS” a pagina 231](#) per ulteriori dettagli su come ottenere gli ID utente alternativi.

Le seguenti definizioni RACF di esempio consentono al programma server di specificare ID utente alternativi che iniziano con i caratteri PS:

```
RDEFINE MQADMIN QMPY.ALTERNATE.USER.PS* UACC(NONE)
PERMIT QMPY.ALTERNATE.USER.PS* CLASS(MQADMIN) ID(PAYSERV) ACCESS(UPDATE)
```



## Nota:

1. I campi *AlternateUserId* nel descrittore dell'oggetto e nel descrittore della sottoscrizione sono lunghi 12 byte. Tutti i 12 byte vengono utilizzati nelle verifiche del profilo, ma solo i primi 8 byte vengono utilizzati come ID utente da IBM MQ. Se questo troncamento dell'ID utente non è desiderabile, i programmi applicativi che effettuano la richiesta devono convertire qualsiasi ID utente alternativo superiore a 8 byte in qualcosa di più appropriato.
2. Se si specifica MQOO\_ALTERNATE\_USER\_AUTHORITY, MQSO\_ALTERNATE\_USER\_AUTHORITY, o MQPMO\_ALTERNATE\_USER\_AUTHORITY e non si specifica un campo *AlternateUserId* nel descrittore dell'oggetto, viene utilizzato un ID utente di spazi vuoti. Ai fini del controllo di sicurezza dell'utente alternativo, l'ID utente utilizzato per il qualificativo *AlternateUserId* è -BLANK-. Ad esempio RDEF MQADMIN hlq.ALTERNATE.USER.-BLANK-.

Se l'utente è autorizzato ad accedere a questo profilo, tutte le ulteriori verifiche vengono effettuate con un ID utente di spazi vuoti. Per i dettagli relativi agli ID utente vuoti, consultare [“ID utente vuoti e livelli UACC”](#) a pagina 239.

La gestione degli ID utente alternativi è più semplice se si dispone di una convenzione di denominazione per gli ID utente che consente di utilizzare profili utente alternativi generici. In caso contrario, è possibile utilizzare la funzione RACF RACVARS. Per i dettagli sull'utilizzo di RACVARS, consultare il manuale *z/OS SecureWay Security Server RACF Security Administrator's Guide*.

Quando un messaggio viene inserito in una coda che è stata aperta con autorizzazione utente alternativa e il contesto del messaggio è stato generato dal gestore code, il campo MQMD\_USER\_IDENTIFIER è impostato sull'ID utente alternativo.

## **Profili per la sicurezza del contesto**

IBM MQ utilizza profili per controllare l'accesso alle informazioni di contesto specifiche di uno specifico messaggio. Il contesto è contenuto nel descrittore del messaggio (MQMD).

### Utilizzo dei profili per la sicurezza del contesto

Se la sicurezza del contesto è attiva, è necessario:

- Definire un profilo nella classe **MQADMIN** se si utilizzano profili in maiuscolo.
- Definire il profilo nella classe **MXADMIN** se si utilizzano profili con caratteri maiuscoli e minuscoli.

Il profilo è denominato hlq.CONTEXT.queuname, dove:

#### hlq

Può essere qmgx-name (nome gestore code) o qsg-name (nome gruppo di condivisione code).

#### nomecoda

Può essere il nome completo della coda per cui si desidera definire il profilo di contesto o un profilo generico.

Un profilo con il prefisso del nome del gestore code e con \*\* specificato come nome della coda, consente il controllo della sicurezza del contesto su tutte le code appartenenti a tale gestore code. Questo valore può essere sovrascritto su una singola coda definendo un profilo di livello coda per il contesto su quella coda.

Un profilo con il prefisso del nome del gruppo di condivisione code e con \*\* specificato come nome della coda, consente il controllo per il contesto su tutte le code che appartengono ai gestori code all'interno del gruppo di condivisione code. Questo può essere sovrascritto su un singolo gestore code definendo un profilo a livello di gestore code per il contesto su tale gestore code, specificando un profilo con il prefisso del nome del gestore code. Può anche essere sovrascritto su una singola coda specificando un profilo con il suffisso del nome coda.

Se il gestore code è un membro di un gruppo di condivisione code e si sta utilizzando la sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ verifica prima un profilo con il prefisso del nome del gestore code. Se non ne trova uno, cerca un profilo preceduto dal nome del gruppo di condivisione code.

È necessario fornire ai gruppi o ID utente necessari l'accesso a questo profilo. La seguente tabella mostra il livello di accesso richiesto, in base alla specifica delle opzioni di contesto quando la coda viene aperta.

<i>Tabella 45. Livelli di accesso per la sicurezza del contesto</i>	
<b>opzione MQOPEN o MQPUT1</b>	<b>RACF livello di accesso richiesto per hlq.CONTEXT.queueename</b>
MQPMO_NO_CONTEXT	Nessun controllo di sicurezza del contesto
MQPMO_DEFAULT_CONTEXT	Nessun controllo di sicurezza del contesto
MQOO_SAVE_ALL_CONTEXT	Nessun controllo di sicurezza del contesto
MQOO_PASS_IDENTITY_CONTEXT MQPMO_PASS_IDENTITY_CONTEXT	LETTURA
MQOO_PASS_ALL_CONTEXT MQPMO_PASS_ALL_CONTEXT	LETTURA
MQOO_SET_IDENTITY_CONTEXT MQPMO_SET_IDENTITY_CONTEXT	AGGIORNA
MQOO_SET_ALL_CONTEXT MQPMO_SET_ALL_CONTEXT	CONTROL
MQOO_OUTPUT o MQPUT1(USO (XMITQ))	CONTROL
<b>Opzione MQSUB</b>	
MQSO_SET_IDENTITY_CONTEXT ( <b>Nota 2</b> )	AGGIORNA

**Nota:**

1. Gli ID utente utilizzati per l'accodamento distribuito richiedono l'accesso CONTROL a hlq.CONTEXT.queueename per inserire i messaggi nella coda di destinazione. Consultare [“ID utente utilizzati dall'iniziatore di canali”](#) a pagina 234 per informazioni sugli ID utente utilizzati.
2. Se sulla richiesta MQSUB, con le opzioni MQSO\_CREATE o MQSO\_ALTER specificate, si desidera impostare uno qualsiasi dei campi del contesto di identità nella struttura MQSD, è necessario specificare l'opzione MQSO\_SET\_IDENTITY\_CONTEXT. È inoltre necessaria l'autorità appropriata per il profilo di contesto per la coda di destinazione.

Se si immettono i comandi nella coda di input del comando di sistema, utilizzare l'opzione del messaggio di inserimento del contesto predefinito per associare l'ID utente corretto al comando.

Ad esempio, il programma di utilità fornito da IBM MQCSQUTIL può essere utilizzato per scaricare e ricaricare i messaggi nelle code. Quando i messaggi con offload vengono ripristinati in una coda, il programma di utilità CSQUTIL utilizza l'opzione MQOO\_SET\_ALL\_CONTEXT per riportare i messaggi allo stato originale. Oltre alla sicurezza della coda richiesta da questa opzione di apertura, è richiesta anche l'autorizzazione del contesto. Ad esempio, se questa autorizzazione è necessaria per il gruppo BACKGRP sul gestore code MQS1, questa verrà definita come segue:

```
RDEFINE MQADMIN MQS1.CONTEXT.** UACC(NONE)
PERMIT MQS1.CONTEXT.** CLASS(MQADMIN) ID(BACKGRP) ACCESS(CONTROL)
```

A seconda delle opzioni specificate e dei tipi di sicurezza eseguiti, potrebbero verificarsi anche altri tipi di controlli di sicurezza quando la coda viene aperta. Questi includono la sicurezza della coda (consultare [“Profili per la sicurezza della coda”](#) a pagina 192) e la sicurezza dell'utente alternativo (consultare [“Profili per la sicurezza utente alternativa”](#) a pagina 207). Per una tabella di riepilogo che mostra le opzioni di apertura e i controlli di sicurezza richiesti quando la coda, il contesto e la sicurezza dell'utente alternativo sono tutti attivi, consultare [Tabella 36](#) a pagina 200.

## Sicurezza del contesto della coda di sistema

A molte delle code di sistema si accede dalle parti ausiliarie di IBM MQ, ad esempio lo spazio di indirizzo dell'inziatore di canali **V9.0.1** e il profilo IBM WebSphere Application Server Liberty per il server IBM MQ (WLP per il server MQ) utilizzato da IBM MQ Console e amministrative REST API.

Gli ID utente con cui vengono eseguiti devono avere l'accesso RACF a queste code, come mostrato in Tabella 46 a pagina 211.

**V9.0.1**

Coda SYSTEM	Inziatore di canali per accodamento distribuito	WLP per server MQ
SYSTEM.ADMIN.COMMAND.QUEUE	-	CONTROL
SYSTEM.BROKER.CONTROL.QUEUE	CONTROL	-
SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS	CONTROL	-
SYSTEM.CHANNEL.SYNCQ	CONTROL	-
SYSTEM.CLUSTER.COMMAND.QUEUE	CONTROL	-
SYSTEM.CLUSTER.TRANSMIT.QUEUE	CONTROL	-

**Z/OS**

### Profili per la sicurezza dei comandi

Per abilitare il controllo di sicurezza per i comandi, aggiungere profili alla classe MQCMDS. I nomi profilo si basano sui comandi MQSC ma controllano sia i comandi MQSC che PCF. I profili possono essere applicati a un gestore code o a un gruppo di condivisione code.

Se si desidera il controllo di sicurezza per i comandi (in modo che non sia stato definito il profilo di commutazione di sicurezza del comando hlq.NO.CMD.CHECKS) è necessario aggiungere profili alla classe MQCMDS.

Gli stessi profili di sicurezza controllano i comandi MQSC e PCF. I nomi dei profili RACF per il controllo di sicurezza dei comandi si basano sui nomi dei comandi MQSC stessi. Questi profili hanno il formato:

```
hlq.verb.pkw
```

Dove hlq può essere qmgr-name (nome gestore code) o qsg-name (nome gruppo di condivisione code), verb è la parte verbo del nome del comando, ad esempio ALTER, e pkw è il tipo di oggetto, ad esempio QLOCAL per una coda locale.

Quindi, il nome del profilo per il comando ALTER QLOCAL nel sottosistema CSQ1 è:

```
CSQ1.ALTER.QLOCAL
```

È possibile utilizzare profili generici per proteggere serie di comandi in modo da avere meno profili da gestire e, quindi, meno elenchi di accesso. Creare un profilo generico che si applichi a tutti i comandi non protetti da un profilo più specifico. Definire questo profilo con UACC (NONE) e concedere l'accesso ALTER solo ai gruppi RACF che contengono gli amministratori. È possibile, quindi, creare un profilo generico applicabile a tutti i comandi DISPLAY e concedere un accesso generalizzato ad esso. Tra questi estremi, è possibile identificare i gruppi di utenti che hanno bisogno di accedere a determinate serie di comandi, nel qual caso è possibile creare profili per tali serie e concedere l'accesso ai gruppi RACF che rappresentano tali classi di utenti. Evitare di concedere agli utenti l'accesso ai comandi che non richiedono: applicare il principio del privilegio minimo, in modo che gli utenti abbiano accesso solo ai comandi richiesti per i loro lavori.

Un profilo preceduto dal nome del gestore code controlla l'utilizzo del comando su tale gestore code. Un profilo preceduto dal nome del gruppo di condivisione code controlla l'utilizzo del comando su tutti i gestori code all'interno del gruppo di condivisione code. Questo accesso può essere sovrascritto su un singolo gestore code definendo un profilo a livello di gestore code per tale comando su tale gestore code.

Se il gestore code è un membro di un gruppo di condivisione code e si sta utilizzando la sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ ricerca un profilo con prefisso corrispondente al nome del gestore code. Se non ne trova uno, cerca un profilo preceduto dal nome del gruppo di condivisione code.

Impostando i profili di comando a livello di gestore code, è possibile limitare l'immissione dei comandi su un determinato gestore code. In alternativa, è possibile definire un profilo per un gruppo di condivisione code per ciascun comando e tutti i controlli di protezione vengono eseguiti su tale profilo invece che su singoli gestori code.

Se sia la sicurezza del sottosistema che la sicurezza del gruppo di condivisione code sono attive e non viene trovato un profilo locale, viene eseguito un controllo di sicurezza del comando per vedere se l'utente ha accesso a un profilo del gruppo di condivisione code.

Se si utilizza l'attributo CMDSCOPE per instradare un comando ad altri gestori code in un gruppo di condivisione code, la sicurezza viene controllata su ciascun gestore code in cui viene eseguito il comando, ma non necessariamente sul gestore code in cui viene immesso il comando.

Tabella 47 a pagina 212 mostra, per ogni comando MQSC IBM MQ, i profili richiesti per il controllo della sicurezza dei comandi da eseguire e il livello di accesso corrispondente per ogni profilo nella classe MQCMDS.

Tabella 48 a pagina 218 mostra, per ogni comando PCF IBM MQ, i profili richiesti per il controllo di sicurezza del comando da eseguire e il livello di accesso corrispondente per ogni profilo nella classe MQCMDS.

<i>Tabella 47. Comandi MQSC, profili e relativi livelli di accesso</i>				
<b>Comando</b>	<b>Profilo di comando per MQCMDS</b>	<b>Livello di accesso per MQCMDS</b>	<b>Profilo della risorsa del comando per MQADMIN o MXADMIN</b>	<b>Livello di accesso per MQADMIN o MXADMIN</b>
MODIFICA AUTHINFO	hlq.ALTER.AUTHINFO	MODIFICA	hlq.AUTHINFO.resourcenam e	MODIFICA
MODIFICA BUFFERPOOL	hlq.ALTER.BUFFPOOL	MODIFICA	Nessun controllo	-
MODIFICA CFSTRUCT	hlq.ALTER.CFSTRUCT	MODIFICA	Nessun controllo	-
MODIFICA CANALE	hlq.ALTER.CHANNEL	MODIFICA	hlq.CHANNEL.channel	MODIFICA
ELENCO NOMI ALTER	hlq.ALTER.NAMELIST	MODIFICA	hlq.NAMELIST.namelist	MODIFICA
MODIFICA PROCESSO	hlq.ALTER.PROCESS	MODIFICA	hlq.PROCESS.process	MODIFICA
PSID ALTER	hlq.ALTER.PSID	MODIFICA	Nessun controllo	-
MODIFICA QALIAS	hlq.ALTER.QALIAS	MODIFICA	hlq.QUEUE.queue	MODIFICA
MODIFICA QLOCAL	hlq.ALTER.QLOCAL	MODIFICA	hlq.QUEUE.queue	MODIFICA
Gestore code ALTER	hlq.ALTER.QMGR	MODIFICA	Nessun controllo	-
MODIFICA QMODEL	hlq.ALTER.QMODEL	MODIFICA	hlq.QUEUE.queue	MODIFICA
MODIFICA QREMOTE	hlq.ALTER.QREMOTE	MODIFICA	hlq.QUEUE.queue	MODIFICA
MODIFICA SICUREZZA	hlq.ALTER.SECURITY	MODIFICA	Nessun controllo	-

Tabella 47. Comandi MQSC, profili e relativi livelli di accesso (Continua)

Comando	Profilo di comando per MQCMDS	Livello di accesso per MQCMDS	Profilo della risorsa del comando per MQADMIN o MXADMIN	Livello di accesso per MQADMIN o MXADMIN
SMDS ALTER	hlq.ALTER.SMDS	MODIFICA	Nessun controllo	-
ALTER STGCLASS	hlq.ALTER.STGCLASS	MODIFICA	Nessun controllo	-
MODIFICA SUB	hlq.ALTER.SUB	MODIFICA	Nessun controllo	-
ALTER TOPIC	hlq.ALTER.TOPIC	MODIFICA	hlq.TOPIC.topic	MODIFICA
MODIFICA TRACCIA	hlq.ALTER.TRACE	MODIFICA	Nessun controllo	-
LOG DI ARCHIVIO	hlq.ARCHIVE.LOG	CONTROL	Nessun controllo	-
CFSTRUCT DI BACKUP	hlq.BACKUP.CFSTRUCT	CONTROL	Nessun controllo	-
CANCELLA QLOCAL	hlq.CLEAR.QLOCAL	MODIFICA	hlq.QUEUE.queue	MODIFICA
CANCELLA STR ARGOMENTO "3" a pagina 217	hlq.CLEAR.TOPICSTR	MODIFICA	hlq.TOPIC.topic	MODIFICA
DEFINE AUTINFO	hlq.DEFINE.AUTHINFO	MODIFICA	hlq.AUTHINFO.resourcenam e	MODIFICA
DEFINISCI BUFFPOOL	hlq.DEFINE.BUFFPOOL	MODIFICA	Nessun controllo	-
DEFINE CFSTRUCT	hlq.DEFINE.CFSTRUCT	MODIFICA	Nessun controllo	-
Definire il canale	hlq.DEFINE.CHANNEL	MODIFICA	hlq.CHANNEL.channel	MODIFICA
DEFINE LOG	hlq.DEFINE.LOG	MODIFICA	Nessun controllo	-
DEFINE MAXSMSGS	hlq.DEFINE.MAXSMSGS	MODIFICA	Nessun controllo	-
DEFINE ELENCO_NOMI	hlq.DEFINE.NAMELIST	MODIFICA	hlq.NAMELIST.namelist	MODIFICA
DEFINE PROCESS	hlq.DEFINE.PROCESS	MODIFICA	hlq.PROCESS.process	MODIFICA
DEFINE PSID	hlq.DEFINE.PSID	MODIFICA	Nessun controllo	-
DEFINE QALIAS	hlq.DEFINE.QALIAS	MODIFICA	hlq.QUEUE.queue	MODIFICA
DEFINE QLOCAL	hlq.DEFINE.QLOCAL	MODIFICA	hlq.QUEUE.queue	MODIFICA
DEFINE QMODEL	hlq.DEFINE.QMODEL	MODIFICA	hlq.QUEUE.queue	MODIFICA
DEFINIRE QREMOTE	hlq.DEFINE.QREMOTE	MODIFICA	hlq.QUEUE.queue	MODIFICA
DEFINE STGCLASS	hlq.DEFINE.STGCLASS	MODIFICA	Nessun controllo	-
DEFINE SUB	hlq.DEFINE.SUB	MODIFICA	Nessun controllo	-
DEFINISCI ARGOMENTO	hlq.DEFINE.TOPIC	MODIFICA	hlq.TOPIC.topic	MODIFICA
DELETE AUTINFO	hlq.DELETE.AUTHINFO	MODIFICA	hlq.AUTHINFO.resourcenam e	MODIFICA
ELIMINA BUFFPOOL	hlq.DELETE.BUFFPOOL	MODIFICA	Nessun controllo	-
DELETE CFSTRUCT	hlq.DELETE.CFSTRUCT	MODIFICA	Nessun controllo	-

Tabella 47. Comandi MQSC, profili e relativi livelli di accesso (Continua)

Comando	Profilo di comando per MQCMDS	Livello di accesso per MQCMDS	Profilo della risorsa del comando per MQADMIN o MXADMIN	Livello di accesso per MQADMIN o MXADMIN
Elimina canale	hlq.DELETE.CHANNEL	MODIFICA	hlq.CHANNEL.channel	MODIFICA
Eliminazione elenco nomi	hlq.DELETE.NAMELIST	MODIFICA	hlq.NAMELIST.namelist	MODIFICA
Eliminazione processo	hlq.DELETE.PROCESS	MODIFICA	hlq.PROCESS.process	MODIFICA
ELIMINA PSID	hlq.DELETE.PSID	MODIFICA	Nessun controllo	-
DELETE QALIAS	hlq.DELETE.QALIAS	MODIFICA	hlq.QUEUE.queue	MODIFICA
ELIMINA QLOCALE	hlq.DELETE.QLOCAL	MODIFICA	hlq.QUEUE.queue	MODIFICA
DELETE QMODEL	hlq.DELETE.QMODEL	MODIFICA	hlq.QUEUE.queue	MODIFICA
ELIMINA QREMOTE	hlq.DELETE.QREMOTE	MODIFICA	hlq.QUEUE.queue	MODIFICA
DELETE STGCLASS	hlq.DELETE.STGCLASS	MODIFICA	Nessun controllo	-
ELIMINA SUB	hlq.DELETE.SUB	MODIFICA	Nessun controllo	-
Elimina argomento	hlq.DELETE.TOPIC	MODIFICA	hlq.TOPIC.topic	MODIFICA
VISUALIZZA ARCHIVIO "1" a pagina 217	hlq.DISPLAY.ARCHIVE	LETTURA	Nessun controllo	-
VISUALIZZA AUTHINFO	hlq.DISPLAY.AUTHINFO	LETTURA	Nessun controllo	-
VISUALIZZARE CFSTATUS	hlq.DISPLAY.CFSTATUS	LETTURA	Nessun controllo	-
VISUALIZZA CFSTRUCT	hlq.DISPLAY.CFSTRUCT	LETTURA	Nessun controllo	-
VISUALIZZA CANALE	hlq.DISPLAY.CHANNEL	LETTURA	Nessun controllo	-
VISUALIZZA CHINIT	hlq.DISPLAY.CHINIT	LETTURA	Nessun controllo	-
VISUALIZZA CHLAUTH	hlq.DISPLAY.CHLAUTH	LETTURA	Nessun controllo	-
VISUALIZZA CHSTATUS	hlq.DISPLAY.CHSTATUS	LETTURA	Nessun controllo	-
VISUALIZZA CLUSQMGR	hlq.DISPLAY.CLUSQMGR	LETTURA	Nessun controllo	-
VISUALIZZA CMDSERV	hlq.DISPLAY.CMDSERV	LETTURA	Nessun controllo	-
VISUALIZZA CONN "1" a pagina 217	hlq.DISPLAY.CONN	LETTURA	Nessun controllo	-
Visualizza gruppo	hlq.DISPLAY.GROUP	LETTURA	Nessun controllo	-
VISUALIZZA LOG "1" a pagina 217	hlq.DISPLAY.LOG	LETTURA	Nessun controllo	-

Tabella 47. Comandi MQSC, profili e relativi livelli di accesso (Continua)

Comando	Profilo di comando per MQCMDS	Livello di accesso per MQCMDS	Profilo della risorsa del comando per MQADMIN o MXADMIN	Livello di accesso per MQADMIN o MXADMIN
VISUALIZZA MAXSMMSG	hlq.DISPLAY.MAXSMMSG	LETTURA	Nessun controllo	-
VISUALIZZA ELENCO NOMI	hlq.DISPLAY.NAMELIST	LETTURA	Nessun controllo	-
VISUALIZZA PROCESSO	hlq.DISPLAY.PROCESS	LETTURA	Nessun controllo	-
VISUALIZZA PUBSUB	hlq.DISPLAY.PUBSUB	LETTURA	Nessun controllo	-
VISUALIZZA QALIAS	hlq.DISPLAY.QALIAS	LETTURA	Nessun controllo	-
VISUALIZZAZIONE QCLUSTER	hlq.DISPLAY.QCLUSTER	LETTURA	Nessun controllo	-
VISUALIZZA QLOCAL	hlq.DISPLAY.QLOCAL	LETTURA	Nessun controllo	-
VISUALIZZAZIONE QMGR	hlq.DISPLAY.QMGR	LETTURA	Nessun controllo	-
VISUALIZZA QMODEL	hlq.DISPLAY.QMODEL	LETTURA	Nessun controllo	-
VISUALIZZAZIONE QREMOTE	hlq.DISPLAY.QREMOTE	LETTURA	Nessun controllo	-
VISUALIZZAZIONE QSTATUS	hlq.DISPLAY.QSTATUS	LETTURA	Nessun controllo	-
VISUALIZZAZIONE CODA	hlq.DISPLAY.QUEUE	LETTURA	Nessun controllo	-
VISUALIZZAZIONE STATO SB	hlq.DISPLAY.SBSTATUS	LETTURA	Nessun controllo	-
Visualizzazione SMDS	hlq.DISPLAY.SMDS	LETTURA	Nessun controllo	-
VISUALIZZA SMDSCONN	hlq.DISPLAY.SMDSCONN	LETTURA	Nessun controllo	-
VISUALIZZA SECONDARIO	hlq.DISPLAY.SUB	LETTURA	Nessun controllo	-
VISUALIZZAZIONE SICUREZZA	hlq.DISPLAY.SECURITY	LETTURA	Nessun controllo	-
VISUALIZZA CLASSE DI STG	hlq.DISPLAY.STGCLASS	LETTURA	Nessun controllo	-
VISUALIZZA SISTEMA "1" a pagina 217	hlq.DISPLAY.SYSTEM	LETTURA	Nessun controllo	-
VISUALIZZA THREAD	hlq.DISPLAY.THREAD	LETTURA	Nessun controllo	-
VISUALIZZA TPSTATUS	hlq.DISPLAY.TPSTATUS	LETTURA	Nessun controllo	-
VISUALIZZA ARGOMENTO	hlq.DISPLAY.TOPIC	LETTURA	Nessun controllo	-

Tabella 47. Comandi MQSC, profili e relativi livelli di accesso (Continua)

Comando	Profilo di comando per MQCMDS	Livello di accesso per MQCMDS	Profilo della risorsa del comando per MQADMIN o MXADMIN	Livello di accesso per MQADMIN o MXADMIN
VISUALIZZA TPSTATUS	hlq.DISPLAY.TPSTATUS	LETTURA	Nessun controllo	-
VISUALIZZA TRACCIA	hlq.DISPLAY.TRACE	LETTURA	Nessun controllo	-
Visualizza utilizzo“1” a pagina 217	hlq.DISPLAY.USAGE	LETTURA	Nessun controllo	-
SPOSTA QLOCAL	hlq.MOVE.QLOCAL	MODIFICA	hlq.QUEUE.from-queue hlq.QUEUE.to-queue	MODIFICA
Ping canale	hlq.PING.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Recupero BSDS	hlq.RECOVER.BSDS	CONTROL	Nessun controllo	-
RECUPERO CFSTRUCT	hlq.RECOVER.CFSTRUCT	CONTROL	Nessun controllo	-
Aggiornamento cluster	hlq.REFRESH.CLUSTER	MODIFICA	Nessun controllo	-
AGGIORNA QMGR	hlq.REFRESH.QMGR	MODIFICA	Nessun controllo	-
Aggiorna sicurezza	hlq.REFRESH.SECURITY	MODIFICA	Nessun controllo	-
REIMPOSTA CFSTRUCT	hlq.RESET.CFSTRUCT	CONTROL	Nessun controllo	-
Reimpostazione canale	hlq.RESET.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Reimposta cluster	hlq.RESET.CLUSTER	CONTROL	Nessun controllo	-
RESET QMGR	hlq.RESET.QMGR	CONTROL	Nessun controllo	-
QSTATS RESET	hlq.RESET.QSTATS	CONTROL	hlq.QUEUE.queue	CONTROL
Reimposta SMDS	hlq.RESET.SMDS	CONTROL	Nessun controllo	-
Reimpostazione tpipe	hlq.RESET.TPIPE	CONTROL	Nessun controllo	-
Risoluzione canale	hlq.RESOLVE.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Risoluzione in dubbio	hlq.RESOLVE.INDOUBT	CONTROL	Nessun controllo	-
RESUME QMGR	hlq.RESUME.QMGR	CONTROL	Nessun controllo	-
SICUREZZA RVERIFY	hlq.RVERIFY.SECURITY	MODIFICA	Nessun controllo	-
Impostazione archivio	hlq.SET.ARCHIVE	CONTROL	Nessun controllo	-
SET CHLAUTH	hlq.SET.CHLAUTH	CONTROL	Nessun controllo	-
Impostazione log	hlq.SET.LOG	CONTROL	Nessun controllo	-
Impostazione sistema	hlq.SET.SYSTEM	CONTROL	Nessun controllo	-
Avvio canale	hlq.START.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
INIZIO CHINIT “4” a pagina 217	hlq.START.CHINIT	CONTROL	Nessun controllo	-
AVVIA CMDSERV	hlq.START.CMDSERV	CONTROL	Nessun controllo	-



Tabella 47. Comandi MQSC, profili e relativi livelli di accesso (Continua)

Comando	Profilo di comando per MQCMDS	Livello di accesso per MQCMDS	Profilo della risorsa del comando per MQADMIN o MXADMIN	Livello di accesso per MQADMIN o MXADMIN
Avvia listener	hlq.START.LISTENER	CONTROL	Nessun controllo	-
INIZIO GESTORE CODE	Nessuno <a href="#">“2” a pagina 217</a>	-	-	-
AVVIO SMDSCONN	hlq.START.SMDSCONN	CONTROL	Nessun controllo	-
Avvio traccia	hlq.START.TRACE	CONTROL	Nessun controllo	-
Arresto canale	hlq.STOP.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
ARRESTARE CHINIT	hlq.STOP.CHINIT	CONTROL	Nessun controllo	-
STOP CMDSERV	hlq.STOP.CMDSERV	CONTROL	Nessun controllo	-
Arresto del listener	hlq.STOP.LISTENER	CONTROL	Nessun controllo	-
STOP QMGR	hlq.STOP.QMGR	CONTROL	Nessun controllo	-
STOP SMDSCONN	hlq.STOP.SMDSCONN	CONTROL	Nessun controllo	-
Arresta traccia	hlq.STOP.TRACE	CONTROL	Nessun controllo	-
Gestore code in stato SUSPEND	hlq.SUSPEND.QMGR	CONTROL	Nessun controllo	-

**Note:**

1. Questi comandi potrebbero essere emessi internamente dal gestore code; in questi casi non viene controllata alcuna autorizzazione.
2. IBM MQ non controlla l'autorizzazione dell'utente che immette il comando START QMGR. Tuttavia, è possibile utilizzare RACFo le funzioni di sicurezza alternative per controllare l'accesso al comando START xxxxMSTR emesso come risultato del comando START QMGR. Questa operazione viene effettuata controllando l'accesso al profilo MVS.START.STC.xxxxMSTR nella classe OPERCMDS ( RACF operator commands). Per i dettagli di questa procedura, consultare *z/OS SecureWay Security Server RACF Security Administrator's Guide*. Se si utilizza questa tecnica, e un utente non autorizzato tenta di avviare il gestore code, questo termina con un codice motivo 00F30216.
3. La risorsa **hlq.TOPIC.topic** fa riferimento all'oggetto Argomento derivato da TOPICSTR. Per ulteriori dettagli, consultare [“Sicurezza di pubblicazione/sottoscrizione” a pagina 438](#)
4. Nelle release precedenti a IBM MQ for z/OS V6, il controllo di sicurezza era per MVS.START.STC.CSQ1CHIN. In IBM MQ for z/OS V6 e versioni successive, al nome della risorsa è aggiunto un ulteriore qualificatore JOBNAME. Ciò può causare problemi all'avvio dell'iniziatore di canali.

Per risolvere il problema sostituire MVS.START.STC. ssid CHIN con un profilo per una risorsa denominata MVS.START.STC. ssid CHIN.\* o MVS.START.STC. ssid CHIN. ssid CHIN dove ssid è l'ID del sottosistema per il gestore code. Ciò richiede l'autorizzazione RACF UPDATE. Per ulteriori dettagli, consultare [Documentazione del prodotto z/OS per Pianificazione delle operazioni, Comandi MVS, RACF Autorizzazioni di accesso e Nomi delle risorse](#).

Il parametro START per ssid MSTR non include JOBNAME=. Per coerenza, è possibile aggiornare il profilo per MVS.START.STC.ssidMSTR in MVS.START.STC.ssidMSTR.\*.

Tabella 48. Comandi PCF, profili e relativi livelli di accesso

Comando	Profilo di comando per MQCMDS	Livello di accesso per MQCMDS	Profilo della risorsa del comando per MQADMIN o MXADMIN	Livello di accesso per MQADMIN o MXADMIN
Backup struttura CF	hlq.BACKUP.CFSTRUCT	CONTROL	Nessun controllo	-
Modifica oggetto informazioni di autenticazione	hlq.ALTER.AUTHINFO	MODIFICA	hlq.AUTHINFO.resourcename	MODIFICA
Modifica struttura CF	hlq.ALTER.CFSTRUCT	MODIFICA	Nessun controllo	-
Modifica canale	hlq.ALTER.CHANNEL	MODIFICA	hlq.CHANNEL.channel	MODIFICA
Modifica elenco nomi	hlq.ALTER.NAMELIST	MODIFICA	hlq.NAMELIST.namelist	MODIFICA
Modifica processo	hlq.ALTER.PROCESS	MODIFICA	hlq.PROCESS.process	MODIFICA
Modifica coda	hlq.ALTER.QUEUE	MODIFICA	hlq.QUEUE.queue	MODIFICA
Modifica gestore code	hlq.ALTER.QMGR	MODIFICA	Nessun controllo	-
Sicurezza modifica	hlq.ALTER.SECURITY	MODIFICA	Nessun controllo	-
Modifica SMDS	hlq.ALTER.SMDS	MODIFICA	Nessun controllo	-
Modifica classe di memoria	hlq.ALTER.STGCLASS	MODIFICA	Nessun controllo	-
Modifica sottoscrizione	hlq.ALTER.SUB	MODIFICA	Nessun controllo	-
Modifica argomento	hlq.ALTER.TOPIC	MODIFICA	hlq.TOPIC.topic	MODIFICA
Cancellazione coda	hlq.CLEAR.QLOCAL	MODIFICA	hlq.QUEUE.queue	MODIFICA
Cancella stringa argomento "1" a pagina 222	hlq.CLEAR.TOPICSTR	MODIFICA	hlq.TOPIC.topic	MODIFICA
Copia oggetto informazioni di autenticazione	hlq.DEFINE.AUTHINFO	MODIFICA	hlq.AUTHINFO.resourcename	MODIFICA
Copia struttura CF	hlq.DEFINE.CFSTRUCT	MODIFICA	Nessun controllo	-
Copia canale	hlq.DEFINE.CHANNEL	MODIFICA	hlq.CHANNEL.channel	MODIFICA
Copia elenco nomi	hlq.DEFINE.NAMELIST	MODIFICA	hlq.NAMELIST.namelist	MODIFICA
Copia processo	hlq.DEFINE.PROCESS	MODIFICA	hlq.PROCESS.process	MODIFICA
Copia coda	hlq.DEFINE.QUEUE	MODIFICA	hlq.QUEUE.queue	MODIFICA
Copia sottoscrizione	hlq.DEFINE.SUB	MODIFICA	Nessun controllo	-
Copia classe di memoria	hlq.DEFINE.STGCLASS	MODIFICA	Nessun controllo	-
Copia argomento	hlq.DEFINE.TOPIC	MODIFICA	hlq.TOPIC.topic	MODIFICA
Crea oggetto informazioni di autenticazione	hlq.DEFINE.AUTHINFO	MODIFICA	hlq.AUTHINFO.resourcename	MODIFICA
Creazione struttura CF	hlq.DEFINE.CFSTRUCT	MODIFICA	Nessun controllo	-
Creazione canale	hlq.DEFINE.CHANNEL	MODIFICA	hlq.CHANNEL.channel	MODIFICA
Creazione elenco nomi	hlq.DEFINE.NAMELIST	MODIFICA	hlq.NAMELIST.namelist	MODIFICA
Creazione processo	hlq.DEFINE.PROCESS	MODIFICA	hlq.PROCESS.process	MODIFICA

Tabella 48. Comandi PCF, profili e relativi livelli di accesso (Continua)

<b>Comando</b>	<b>Profilo di comando per MQCMDS</b>	<b>Livello di accesso per MQCMDS</b>	<b>Profilo della risorsa del comando per MQADMIN o MXADMIN</b>	<b>Livello di accesso per MQADMIN o MXADMIN</b>
Creazione coda	hlq.DEFINE.QUEUE	MODIFICA	hlq.QUEUE.queue	MODIFICA
Creazione classe di memoria	hlq.DEFINE.STGCLASS	MODIFICA	Nessun controllo	-
Crea sottoscrizione	hlq.DEFINE.SUB	MODIFICA	Nessun controllo	-
Crea argomento	hlq.DEFINE.TOPIC	MODIFICA	hlq.TOPIC.topic	MODIFICA
Elimina oggetto informazioni di autenticazione	hlq.DELETE.AUTHINFO	MODIFICA	hlq.AUTHINFO.resourcename	MODIFICA
Eliminazione struttura CF	hlq.DELETE.CFSTRUCT	MODIFICA	Nessun controllo	-
Eliminazione canale	hlq.DELETE.CHANNEL	MODIFICA	hlq.CHANNEL.channel	MODIFICA
Eliminazione elenco nomi	hlq.DELETE.NAMELIST	MODIFICA	hlq.NAMELIST.namelist	MODIFICA
Eliminazione processo	hlq.DELETE.PROCESS	MODIFICA	hlq.PROCESS.process	MODIFICA
Eliminazione coda	hlq.DELETE.QUEUE	MODIFICA	hlq.QUEUE.queue	MODIFICA
Eliminazione classe di memoria	hlq.DELETE.STGCLASS	MODIFICA	Nessun controllo	-
Elimina sottoscrizione	hlq.DELETE.SUB	MODIFICA	Nessun controllo	-
Elimina argomento	hlq.DELETE.TOPIC	MODIFICA	hlq.TOPIC.topic	MODIFICA
Interrogazione archivio	hlq.DISPLAY.ARCHIVE	LETTURA	Nessun controllo	-
Interroga oggetto informazioni di autenticazione	hlq.DISPLAY.AUTHINFO	LETTURA	Nessun controllo	-
Interroga nomi oggetto informazioni di autenticazione	hlq.DISPLAY.AUTHINFO	LETTURA	Nessun controllo	-
Interrogazione struttura CF	hlq.DISPLAY.CFSTRUCT	LETTURA	Nessun controllo	-
Interrogazione nomi strutture CF	hlq.DISPLAY.CFSTRUCT	LETTURA	Nessun controllo	-
Interrogazione stato struttura CF	hlq.DISPLAY.CFSTATUS	LETTURA	Nessun controllo	-
Interrogazione canale	hlq.DISPLAY.CHANNEL	LETTURA	Nessun controllo	-
Interroga record autenticazione canale	hlq.DISPLAY.CHLAUTH	LETTURA	Nessun controllo	-
Interroga iniziatore canale	hlq.DISPLAY.CHINIT	LETTURA	Nessun controllo	-
Interrogazione nomi canali	hlq.DISPLAY.CHANNEL	LETTURA	Nessun controllo	-
Interrogazione stato canale	hlq.DISPLAY.CHSTATUS	LETTURA	Nessun controllo	-

Tabella 48. Comandi PCF, profili e relativi livelli di accesso (Continua)

<b>Comando</b>	<b>Profilo di comando per MQCMDS</b>	<b>Livello di accesso per MQCMDS</b>	<b>Profilo della risorsa del comando per MQADMIN o MXADMIN</b>	<b>Livello di accesso per MQADMIN o MXADMIN</b>
Interrogazione gestore code cluster	hlq.DISPLAY.CLUSQMGR	LETTURA	Nessun controllo	-
Interrogazione connessione	hlq.DISPLAY.CONNPCF	LETTURA	Nessun controllo	-
Interroga gruppo	hlq.DISPLAY.GROUP	LETTURA	Nessun controllo	-
Interrogazione log	hlq.DISPLAY.LOG	LETTURA	Nessun controllo	-
Interrogazione elenco nomi	hlq.DISPLAY.NAMELIST	LETTURA	Nessun controllo	-
Interrogazione nomi elenco nomi	hlq.DISPLAY.NAMELIST	LETTURA	Nessun controllo	-
Interrogazione processo	hlq.DISPLAY.PROCESS	LETTURA	Nessun controllo	-
Interrogazione nomi processi	hlq.DISPLAY.PROCESS	LETTURA	Nessun controllo	-
Interroga stato pubblicazione / sottoscrizione	hlq.DISPLAY.PUBSUB	LETTURA	Nessun controllo	-
Interrogazione coda	hlq.DISPLAY.QUEUE	LETTURA	Nessun controllo	-
Interrogazione gestore code	hlq.DISPLAY.QMGR	LETTURA	Nessun controllo	-
Interrogazione nomi code	hlq.DISPLAY.QUEUE	LETTURA	Nessun controllo	-
Interrogazione stato coda	hlq.DISPLAY.QSTATUS	LETTURA	Nessun controllo	-
Interrogazione sicurezza	hlq.DISPLAY.SECURITY	LETTURA	Nessun controllo	-
Interroga SMDS	hlq.DISPLAY.SMDS	LETTURA	Nessun controllo	-
Interroga SMDSCONN	hlq.DISPLAY.SMDSCONN	LETTURA	Nessun controllo	-
Interrogazione classe di memoria	hlq.DISPLAY.STGCLASS	LETTURA	Nessun controllo	-
Interrogazione nomi classi di memoria	hlq.DISPLAY.STGCLASS	LETTURA	Nessun controllo	-
Interroga sottoscrizione	hlq.INQUIRE.SUB	LETTURA	Nessun controllo	-
Interroga stato sottoscrizione	hlq.INQUIRE.SBSTATUS	LETTURA	Nessun controllo	-
Interrogazione sistema	hlq.DISPLAY.SYSTEM	LETTURA	Nessun controllo	-
Interroga argomento	hlq.DISPLAY.TOPIC	LETTURA	Nessun controllo	-
Interroga nomi argomento	hlq.DISPLAY.TOPIC	LETTURA	Nessun controllo	-
Interroga stato argomento	hlq.DISPLAY.TPSTATUS	LETTURA	Nessun controllo	-
Interrogazione utilizzo	hlq.DISPLAY.USAGE	LETTURA	Nessun controllo	-

Tabella 48. Comandi PCF, profili e relativi livelli di accesso (Continua)

Comando	Profilo di comando per MQCMDS	Livello di accesso per MQCMDS	Profilo della risorsa del comando per MQADMIN o MXADMIN	Livello di accesso per MQADMIN o MXADMIN
Spostamento coda	hlq.MOVE.QLOCAL	MODIFICA	hlq.QUEUE.from-queue hlq.QUEUE.to-queue	MODIFICA
Ping canale	hlq.PING.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Recupero struttura CF	hlq.RECOVER.CFSTRUCT	CONTROL	Nessun controllo	-
Aggiornamento cluster	hlq.REFRESH.CLUSTER	MODIFICA	Nessun controllo	-
Aggiornamento gestore code	hlq.REFRESH.QMGR	MODIFICA	Nessun controllo	-
Aggiornamento sicurezza	hlq.REFRESH.SECURITY	MODIFICA	Nessun controllo	-
Reimposta struttura CF	hlq.RESET.CFSTRUCT	CONTROL	Nessun controllo	-
Reimposta canale	hlq.RESET.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Reimposta cluster	hlq.RESET.CLUSTER	CONTROL	Nessun controllo	-
Reimpostazione gestore code	hlq.RESET.QMGR	CONTROL	Nessun controllo	-
Reimposta statistiche coda	hlq.RESET.QSTATS	CONTROL	hlq.QUEUE.queue	CONTROL
Reimposta SMDS	hlq.RESET.SMDS	CONTROL	Nessun controllo	-
Risolvi canale	hlq.RESOLVE.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Ripristino gestore code	hlq.RESUME.QMGR	CONTROL	Nessun controllo	-
Ripristino cluster gestore code	hlq.RESUME.QMGR	CONTROL	Nessun controllo	-
Nuova verifica sicurezza	hlq.RVERIFY.SECURITY	MODIFICA	Nessun controllo	-
Impostazione archivio	hlq.SET.ARCHIVE	CONTROL	Nessun controllo	-
Imposta record autenticazione canale	hlq.SET.CHLAUTH	CONTROL	Nessun controllo	-
Impostazione log	hlq.SET.LOG	CONTROL	Nessun controllo	-
Impostazione sistema	hlq.SET.SYSTEM	CONTROL	Nessun controllo	-
Avvio canale	hlq.START.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Avvio iniziatore canale	hlq.START.CHINIT	CONTROL	Nessun controllo	-
Avvio listener canale	hlq.START.LISTENER	CONTROL	Nessun controllo	-
Avvia connessione SMDS	hlq.START.SMDSCONN	CONTROL	Nessun controllo	-
Arresta canale	hlq.STOP.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Arresta iniziatore canale	hlq.STOP.CHINIT	CONTROL	Nessun controllo	-
Arresto listener canale	hlq.STOP.LISTENER	CONTROL	Nessun controllo	-
Arresta connessione SMDS	hlq.STOP.SMDSCONN	CONTROL	Nessun controllo	-
Sospensione gestore code	hlq.SUSPEND.QMGR	CONTROL	Nessun controllo	-

Tabella 48. Comandi PCF, profili e relativi livelli di accesso (Continua)

Comando	Profilo di comando per MQCMDS	Livello di accesso per MQCMDS	Profilo della risorsa del comando per MQADMIN o MXADMIN	Livello di accesso per MQADMIN o MXADMIN
Sospensione cluster gestore code	hlq.SUSPEND.QMGR	CONTROL	Nessun controllo	-

**Note:**

1. La risorsa **hlq.TOPIC.topic** fa riferimento all'oggetto Argomento derivato da TOPICSTR. Per ulteriori dettagli, consultare ["Sicurezza di pubblicazione/sottoscrizione"](#) a pagina 438

**V 9.0.1** Consultare ["IBM MQ Console - profili di sicurezza del comando richiesti"](#) a pagina 222 per i dettagli dei profili PCF IBM MQ richiesti, quando si utilizza IBM MQ Console.

**Z/OS V 9.0.1** *IBM MQ Console - profili di sicurezza del comando richiesti*

Se si desidera utilizzare IBM MQ Console amministrative REST API, l'ID utente dello spazio di indirizzo del server dei profili WebSphere Application Server Liberty necessita dell'autorizzazione per immettere determinati comandi PCF

Tabella 49 a pagina 222 mostra, per ogni comando PCF IBM MQ , i profili richiesti per eseguire il controllo di sicurezza dei comandi e il corrispondente livello di accesso per ogni profilo nella classe MQCMDS quando si utilizza IBM MQ Console.

Tabella 49. IBM MQ Console Comandi PCF, profili e relativi livelli di accesso

Comando	Profilo di comando per MQCMDS	Livello di accesso per MQCMDS	Profilo della risorsa del comando per MQADMIN o MXADMIN	Livello di accesso per MQADMIN o MXADMIN
Modifica oggetto informazioni di autenticazione	hlq.ALTER.AUTHINFO	MODIFICA	hlq.AUTHINFO.resourcename	MODIFICA
Modifica canale	hlq.ALTER.CHANNEL	MODIFICA	hlq.CHANNEL.channel	MODIFICA
Modifica coda	hlq.ALTER.QUEUE	MODIFICA	hlq.QUEUE.queue	MODIFICA
Modifica gestore code	hlq.ALTER.QMGR	MODIFICA	Nessun controllo	-
Modifica argomento	hlq.ALTER.TOPIC	MODIFICA	hlq.TOPIC.topic	MODIFICA
Cancellazione coda	hlq.CLEAR.QLOCAL	MODIFICA	hlq.QUEUE.queue	MODIFICA
Crea oggetto informazioni di autenticazione	hlq.DEFINE.AUTHINFO	MODIFICA	hlq.AUTHINFO.resourcename	MODIFICA
Creazione canale	hlq.DEFINE.CHANNEL	MODIFICA	hlq.CHANNEL.channel	MODIFICA
Creazione coda	hlq.DEFINE.QUEUE	MODIFICA	hlq.QUEUE.queue	MODIFICA
Crea sottoscrizione	hlq.DEFINE.SUB	MODIFICA	Nessun controllo	-
Crea argomento	hlq.DEFINE.TOPIC	MODIFICA	hlq.TOPIC.topic	MODIFICA
Elimina oggetto informazioni di autenticazione	hlq.DELETE.AUTHINFO	MODIFICA	hlq.AUTHINFO.resourcename	MODIFICA
Eliminazione canale	hlq.DELETE.CHANNEL	MODIFICA	hlq.CHANNEL.channel	MODIFICA
Eliminazione coda	hlq.DELETE.QUEUE	MODIFICA	hlq.QUEUE.queue	MODIFICA
Elimina sottoscrizione	hlq.DELETE.SUB	MODIFICA	Nessun controllo	-

Tabella 49. IBM MQ Console Comandi PCF, profili e relativi livelli di accesso (Continua)

Comando	Profilo di comando per MQCMDS	Livello di accesso per MQCMDS	Profilo della risorsa del comando per MQADMIN o MXADMIN	Livello di accesso per MQADMIN o MXADMIN
Elimina argomento	hlq.DELETE.TOPIC	MODIFICA	hlq.TOPIC.topic	MODIFICA
Interroga oggetto informazioni di autenticazione	hlq.DISPLAY.AUTHINFO	LETTURA	Nessun controllo	-
Interroga nomi oggetto informazioni di autenticazione	hlq.DISPLAY.AUTHINFO	LETTURA	Nessun controllo	-
Interrogazione canale	hlq.DISPLAY.CHANNEL	LETTURA	Nessun controllo	-
Interroga record autenticazione canale	hlq.DISPLAY.CHLAUTH	LETTURA	Nessun controllo	-
Interroga iniziatore canale	hlq.DISPLAY.CHINIT	LETTURA	Nessun controllo	-
Interrogazione nomi canali	hlq.DISPLAY.CHANNEL	LETTURA	Nessun controllo	-
Interrogazione stato canale	hlq.DISPLAY.CHSTATUS	LETTURA	Nessun controllo	-
Interrogazione coda	hlq.DISPLAY.QUEUE	LETTURA	Nessun controllo	-
Interrogazione gestore code	hlq.DISPLAY.QMGR	LETTURA	Nessun controllo	-
Interrogazione nomi code	hlq.DISPLAY.QUEUE	LETTURA	Nessun controllo	-
Interrogazione stato coda	hlq.DISPLAY.QSTATUS	LETTURA	Nessun controllo	-
Interroga sottoscrizione	hlq.INQUIRE.SUB	LETTURA	Nessun controllo	-
Interroga stato sottoscrizione	hlq.INQUIRE.SBSTATUS	LETTURA	Nessun controllo	-
Interroga argomento	hlq.DISPLAY.TOPIC	LETTURA	Nessun controllo	-
Interroga nomi argomento	hlq.DISPLAY.TOPIC	LETTURA	Nessun controllo	-
Interroga stato argomento	hlq.DISPLAY.TPSTATUS	LETTURA	Nessun controllo	-
Ping canale	hlq.PING.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Aggiornamento cluster	hlq.REFRESH.CLUSTER	MODIFICA	Nessun controllo	-
Aggiornamento sicurezza	hlq.REFRESH.SECURITY	MODIFICA	Nessun controllo	-
Reimposta canale	hlq.RESET.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Risolvi canale	hlq.RESOLVE.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Imposta record autenticazione canale	hlq.SET.CHLAUTH	CONTROL	Nessun controllo	-
Avvio canale	hlq.START.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Arresta canale	hlq.STOP.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL

## **Profili per la sicurezza della risorsa comando**

Se non è stato definito il profilo di commutazione della sicurezza della risorsa del comando, poiché si desidera il controllo di sicurezza per le risorse associate ai comandi, è necessario aggiungere i profili di risorsa per ogni risorsa alla classe appropriata. Gli stessi profili di sicurezza controllano i comandi MQSC e PCF.

Se non è stato definito il profilo di commutazione della sicurezza della risorsa del comando, hlq.NO.COMD.RESC.CHECKS, poiché si desidera il controllo della sicurezza per le risorse associate ai comandi, è necessario:

- Aggiungere un profilo di risorsa nella classe **MQADMIN**, se si utilizzano profili in maiuscolo, per ogni risorsa.
- Aggiungere un profilo di risorsa nella classe **MXADMIN**, se si utilizzano profili con maiuscole e minuscole, per ogni risorsa.

Gli stessi profili di sicurezza controllano i comandi MQSC e PCF.

I profili per il controllo di sicurezza della risorsa comando hanno il seguente formato:

```
hlq.type.resourcenam
```

dove hlq può essere qmgr-name (nome gestore code) o qsg-name (nome gruppo di condivisione code).

Un profilo preceduto dal nome del gestore code controlla l'accesso alle risorse associate ai comandi su tale gestore code. Un profilo preceduto dal nome del gruppo di condivisione code controlla l'accesso alle risorse associate ai comandi su tutti i gestori code all'interno del gruppo di condivisione code. Questo accesso può essere sovrascritto su un singolo gestore code definendo un profilo a livello di gestore code per tale risorsa di comandi su tale gestore code.


Se il gestore code è un membro di un gruppo di condivisione code e si sta utilizzando la sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ verifica prima un profilo con il prefisso del nome del gestore code. Se non ne trova uno, cerca un profilo preceduto dal nome del gruppo di condivisione code.


Ad esempio, il nome profilo RACF per la verifica della sicurezza della risorsa del comando rispetto alla coda modello CREDIT.WORTHY nel sottosistema CSQ1 è:

```
CSQ1.QUEUE.CREDIT.WORTHY
```

Dal momento che i profili per tutti i tipi di risorse di comando sono contenuti nella classe MQADMIN, la parte "tipo" del nome profilo è necessaria nel profilo per distinguere tra risorse di tipi differenti che hanno lo stesso nome. La parte "tipo" del nome profilo può essere CHANNEL, QUEUE, TOPIC, PROCESS o NAMELIST. Ad esempio, un utente potrebbe essere autorizzato a definire hlq.QUEUE.PAYROLL.ONE, ma non si è autorizzati a definire hlq.PROCESS.PAYROLL.ONE

Se il tipo di risorsa è una coda e il profilo è un profilo a livello di gruppo di condivisione code, controlla l'accesso a una o più code locali all'interno del gruppo di condivisione code o a una singola coda condivisa da qualsiasi gestore code nel gruppo di condivisione code.

 **I comandi MQSC, i profili e i relativi livelli di accesso mostrano, per ogni comando MQSC IBM MQ, i profili richiesti per il controllo della sicurezza dei comandi da eseguire e il livello di accesso corrispondente per ogni profilo della classe MQCMD5.**

 **Comandi PCF, profili e relativi livelli di accesso mostra, per ogni comando PCF IBM MQ, i profili richiesti per il controllo della sicurezza dei comandi da eseguire e il livello di accesso corrispondente per ogni profilo nella classe MQCMD5.**

 **Controllo di sicurezza della risorsa comando per le code alias e remote**

Le code alias e remote forniscono entrambe la via indiretta ad un'altra coda. Ulteriori punti si applicano quando si considera il controllo di sicurezza per queste code.



## Code alias

Quando si definisce una coda alias, i controlli di sicurezza delle risorse del comando vengono eseguiti solo rispetto al nome della coda alias, non rispetto al nome della coda di destinazione in cui l'alias viene risolto.

Le code alias possono risolversi sia in code locali che remote. Se non si desidera consentire agli utenti l'accesso a determinate code locali o remote, è necessario effettuare entrambe le seguenti operazioni:

1. Non consentire agli utenti l'accesso a queste code locali e remote.
2. Impedire agli utenti di definire alias per queste code. In altre parole, impedire loro di emettere i comandi DEFINE QALIAS e ALTER QALIAS.

## Code remote

Quando si definisce una coda remota, i controlli di sicurezza delle risorse dei comandi vengono eseguiti solo rispetto al nome della coda remota. Non vengono eseguiti controlli sui nomi delle code specificate negli attributi RNAME o XMITQ nella definizione dell'oggetto coda remota.

## Profilo di sicurezza RESLEVEL

È possibile definire un profilo speciale nella classe MQADMIN o MXADMIN per controllare il numero di ID utente controllati per la sicurezza delle risorse API. Questo profilo è denominato profilo RESLEVEL. Il modo in cui questo profilo influenza la sicurezza delle risorse API dipende dal modo in cui si accede a IBM MQ.

Quando un'applicazione tenta di connettersi a IBM MQ, IBM MQ verifica l'accesso che l'ID utente associato alla connessione ha a un profilo nella classe MQADMIN o MXADMIN denominato:

```
hlq.RESLEVEL
```

Dove hlq può essere ssid (ID sottosistema) o qsg (ID gruppo di condivisione code).

Gli ID utente associati a ciascun tipo di collegamento sono:

- L'ID utente dell'attività di connessione per connessioni batch
- L'ID utente dello spazio di indirizzo CICS per le connessioni CICS
- L'ID utente dello spazio di indirizzi della regione IMS per le connessioni IMS
- L'ID utente dello spazio di indirizzo dell'iniziatore di canali per le connessioni dell'iniziatore di canali



**Attenzione:** RESLEVEL è un'opzione molto potente; può causare l'aggiornamento di tutti i controlli di sicurezza delle risorse per una connessione particolare.

Se non si dispone di un profilo RESLEVEL definito, è necessario fare attenzione che nessun altro profilo nella classe MQADMIN corrisponda a hlq.RESLEVEL. Ad esempio, se si dispone di un profilo in MQADMIN denominato hlq. \* \* e nessun profilo hlq.RESLEVEL, fare attenzione alle conseguenze dell'hlq. \* \* poiché viene utilizzato per il controllo RESLEVEL.

Definire un profilo hlq.RESLEVEL e impostare UACC su NONE, invece di non avere alcun profilo RESLEVEL. Avere il minor numero possibile di utenti o gruppi nell'elenco di accesso. Per i dettagli su come controllare l'accesso RESLEVEL, consultare [“Considerazioni sul controllo su z/OS” a pagina 251.](#)

Se si utilizza solo la sicurezza a livello di gestore code, IBM MQ esegue i controlli RESLEVEL sul profilo qmgr-name . RESLEVEL . Se si utilizza solo la sicurezza a livello di gruppo di condivisione code, IBM MQ esegue i controlli RESLEVEL sul profilo qsg-name . RESLEVEL . Se si utilizza una combinazione di sicurezza a livello di gestore code e di gruppo di condivisione code, IBM MQ verifica innanzitutto l'esistenza di un profilo RESLEVEL a livello di gestore code. Se non ne trova uno, verifica la presenza di un profilo RESLEVEL a livello di gruppo di condivisione code.

Se non riesce a trovare un profilo RESLEVEL, IBM MQ abilita il controllo dell'ID lavoro e attività (o utente alternativo) per una connessione CICS o IMS . Per una connessione batch, IBM MQ abilita il controllo dell'ID utente del lavoro (o alternativo). Per l'inziatore di canali, IBM MQ abilita il controllo dell'ID utente del canale e dell'ID utente MCA (o alternativo).

Se è presente un profilo RESLEVEL, il livello di controllo dipende dall'ambiente e dal livello di accesso per il profilo.

Tenere presente che se il gestore code è membro di un gruppo di condivisione code e non si definisce questo profilo a livello di gestore code, potrebbe essere definito a livello di gruppo di condivisione code che influenzerà il livello di checking. To attivare il controllo di due ID utente, definire un profilo RESLEVEL (con il prefisso del nome del gestore code del nome del gruppo di condivisione code) con un UACC (NONE) e verificare che gli utenti pertinenti non abbiano l'accesso concesso per questo profilo.

Quando si considera l'accesso che l'ID utente dell'inziatore di canali ha per RESLEVEL, tenere presente che la connessione stabilita dall'inziatore di canali è anche la connessione utilizzata dai canali. Un'impostazione che causa l'aggiramento di tutti i controlli di sicurezza delle risorse per l'ID utente dell'inziatore di canali ignora effettivamente i controlli di sicurezza per tutti i canali. Se l'accesso dell'ID utente dell'inziatore di canali a RESLEVEL è diverso da NONE, solo un ID utente (per un livello di accesso READ o UPDATE) o nessun ID utente (per un livello di accesso CONTROL o ALTER) viene controllato per l'accesso. Se si concede all'ID utente dell'inziatore di canali un livello di accesso diverso da NONE a RESLEVEL, accertarsi di aver compreso l'effetto di questa impostazione sui controlli di sicurezza eseguiti per i canali.

L'uso del profilo RESLEVEL significa che non vengono utilizzati i normali record di controllo della sicurezza. Ad esempio, se si immette UAUDIT su un utente, l'accesso al profilo hlq.RESLEVEL in MQADMIN non viene controllato.

Se si utilizza l'opzione RACF WARNING sul profilo hlq.RESLEVEL , non viene prodotto alcun RACF messaggio di avviso per i profili nella classe RESLEVEL.

Il controllo di sicurezza per i messaggi di report come i COD è controllato dal profilo RESLEVEL associato all'applicazione di origine. Ad esempio, se l'id utente di un lavoro batch dispone dell'autorizzazione CONTROL o ALTER per un profilo RESLEVEL, tutte le verifiche delle risorse eseguite dal lavoro batch vengono ignorate, incluso il controllo di sicurezza dei messaggi di report.

Se si modifica il profilo RESLEVEL, gli utenti devono scollegarsi e collegarsi di nuovo prima che si verifichi la modifica. (Ciò include l'arresto e il riavvio dell'inziatore di canali se l'accesso dell'ID utente dello spazio di indirizzo di accodamento distribuito al profilo RESLEVEL viene modificato.)

Per disattivare il controllo RESLEVEL, utilizzare il parametro di sistema RESAUDIT.

## **Connessioni RESLEVEL e batch**

Per impostazione predefinita, quando si accede a una risorsa IBM MQ tramite connessioni batch e di tipo batch, l'utente deve essere autorizzato ad accedere a tale risorsa per la specifica operazione. È possibile ignorare il controllo di sicurezza impostando una definizione RESLEVEL appropriata.

Se l'utente viene selezionato o meno si basa sull'ID utente utilizzato al momento della connessione, lo stesso ID utente utilizzato per il controllo della connessione.

Ad esempio, è possibile impostare RESLEVEL in modo che quando un utente attendibile accede a determinate risorse tramite una connessione batch, non vengono eseguiti controlli di sicurezza delle risorse API; ma quando un utente non attendibile tenta di accedere alle stesse risorse, i controlli di sicurezza vengono eseguiti normalmente. È necessario impostare il controllo RESLEVEL per ignorare i controlli di sicurezza delle risorse API solo quando l'utente e i programmi eseguiti da tale utente sono sufficientemente affidabili.

La seguente tabella mostra i controlli effettuati per le connessioni batch.

Tabella 50. Verifiche effettuate a diversi livelli di accesso RACF per connessioni batch

RACF Livello di accesso	Livello di controllo
NESSUNO	Controlli delle risorse eseguiti
LETTURA	Controlli delle risorse eseguiti
AGGIORNA	Controlli delle risorse eseguiti
CONTROL	Nessun controllo.
MODIFICA	Nessun controllo.

### **z/OS Funzioni di sistema e RESLEVEL**

L'applicazione di RESLEVEL ai pannelli di funzionamento e di controllo e a CSQUTIL.

I pannelli di controllo e di operazione e il programma di utilità CSQUTIL sono applicazioni di tipo batch che effettuano richieste al server dei comandi del gestore code, pertanto sono soggette alle considerazioni descritte in “Connessioni RESLEVEL e batch” a pagina 226. È possibile utilizzare RESLEVEL per ignorare il controllo di protezione per SYSTEM.COMMAND.INPUT e SYSTEM.COMMAND.REPLY.MODEL code che utilizzano, ma non per le code dinamiche SYSTEM.CSQXCMD. \*, SYSTEM.CSQOREXX.\*, e SYSTEM.CSQUTIL. \*.

Il server dei comandi è parte integrante del gestore code e pertanto non dispone di una connessione o di un controllo RESLEVEL ad esso associati. Per mantenere la sicurezza, quindi, il server dei comandi deve confermare che l'ID utente dell'applicazione richiedente dispone dell'autorità per aprire la coda utilizzata per le repliche. Per le operazioni e i pannelli di controllo, questo è SYSTEM.CSQOREXX. \*. Per CSQUTIL, è SYSTEM.CSQUTIL. \*. Gli utenti devono essere autorizzati a utilizzare queste code, come descritto in “Sicurezza della coda di sistema” a pagina 198, in aggiunta a qualsiasi autorizzazione RESLEVEL loro fornita.

Per altre applicazioni che utilizzano il server dei comandi, è la coda che chiamano come coda di risposta. Tali altre applicazioni potrebbero ingannare il server dei comandi nel posizionare i messaggi su code non autorizzate passando (nel contesto del messaggio) un ID utente più attendibile rispetto al proprio al server dei comandi. Per evitare ciò, utilizzare un profilo CONTEXT per proteggere il contesto di identità dei messaggi collocati su SYSTEM.COMMAND.INPUT.

### **z/OS RESLEVEL e connessioni CICS**

Per impostazione predefinita, quando viene eseguito un controllo di sicurezza delle risorse API su un collegamento CICS, vengono controllati due ID utente. È possibile modificare gli ID utente controllati impostando un profilo RESLEVEL.

Il primo ID utente controllato è quello dello spazio di indirizzo CICS. Si tratta dell'ID utente sulla scheda di lavoro del lavoro CICS o dell'ID utente assegnato all'attività avviata CICS dalla classe STARTED z/OS o dalla tabella delle procedure avviate. (Non è CICS DFLTUSER.)

Il secondo ID utente selezionato è l'ID utente associato alla transazione CICS.

Se uno di questi ID utente non ha accesso alla risorsa, la richiesta ha esito negativo con un codice di completamento MQRN\_NOT\_AUTHORIZED. L'ID utente dello spazio di indirizzi CICS e l'ID utente della persona che esegue la transazione CICS devono avere accesso alla risorsa al livello corretto.

### **In che modo RESLEVEL può influire sulle verifiche effettuate**

A seconda di come si imposta il profilo RESLEVEL, è possibile modificare quali ID utente vengono controllati quando viene richiesto l'accesso ad una risorsa. Per ulteriori informazioni, consultare [Tabella 51 a pagina 228](#).

Gli ID utente selezionati dipendono dall'ID utente utilizzato al momento della connessione, ossia l'ID utente dello spazio di indirizzi CICS. Questo controllo consente di ignorare il controllo di sicurezza delle

risorse API per le richieste IBM MQ provenienti da un sistema (ad esempio, un sistema di test, TESTCICS) ma di implementarle per un'altro (ad esempio, un sistema di produzione, PRODCICS).

**Nota:** Se si configura l'ID utente dello spazio di indirizzo CICS con l'attributo "trusted" nella classe STARTED o nella RACF tabella delle procedure avviate ICHRIN03, questo sovrascrive qualsiasi controllo ID utente per lo spazio di indirizzo CICS stabilito dal profilo RESLEVEL per il gestore code (ossia, il gestore code non esegue i controlli di sicurezza per lo spazio di indirizzo CICS ). Per ulteriori informazioni, consultare il manuale *CICS Transaction Server for z/OS V3.2 RACF Security Guide*.

La seguente tabella mostra le verifiche effettuate per le connessioni CICS .

<i>Tabella 51. Verifiche effettuate a diversi livelli di accesso RACF per connessioni CICS</i>	
<b>RACF Livello di accesso</b>	<b>Livello di controllo</b>
NESSUNO	IBM MQ controlla l'ID utente dello spazio di indirizzo CICS e l'ID utente di transazione.
LETTURA	IBM MQ controlla solo l'ID utente dello spazio di indirizzo CICS .
AGGIORNA	Se la transazione è definita in CICS con RESSEC (YES), IBM MQ controlla l'ID utente dello spazio di indirizzi CICS e l'ID utente della transazione.
AGGIORNA	Se la transazione è definita su CICS con RESSEC (NO), IBM MQ controlla solo l'ID utente dello spazio di indirizzi CICS .
CONTROL o ALTER	IBM MQ non controlla alcun ID utente.

## **RESLEVEL e connessioni IMS**

Per impostazione predefinita, quando viene eseguito un controllo di sicurezza delle risorse API per una connessione IMS , vengono controllati due ID utente. È possibile modificare gli ID utente controllati impostando un profilo RESLEVEL.

Per impostazione predefinita, quando viene eseguito un controllo di sicurezza della risorsa API per una connessione IMS , vengono controllati due ID utente per vedere se è consentito l'accesso alla risorsa.

Il primo ID utente selezionato è quello dello spazio di indirizzo della regione IMS . Questo viene preso dal campo USER dalla scheda di lavoro o dall'ID utente assegnato alla regione dalla classe z/OS STARTED o dalla tabella delle procedure avviate (SPT).

Il secondo ID utente controllato è associato al lavoro che viene eseguito nella regione dipendente. Viene determinato in base al tipo di regione dipendente come mostrato in [Come viene determinato il secondo ID utente per la connessione IMS\(tm\)](#).

Se il primo o il secondo ID utente IMS non ha accesso alla risorsa, la richiesta ha esito negativo con un codice di completamento di MQRC\_NOT\_AUTHORIZED.

L'impostazione dei profili IBM MQ RESLEVEL non può modificare l'ID utente con cui sono pianificate le transazioni IMS dal programma di controllo trigger IBMfornito MQ-IMS CSQQTRMN. Questo ID utente è il PSBNAME di tale controllo trigger, che per impostazione predefinita è CSQQTRMN.

## **In che modo RESLEVEL può influire sulle verifiche effettuate**

A seconda di come si imposta il profilo RESLEVEL, è possibile modificare quali ID utente vengono controllati quando viene richiesto l'accesso ad una risorsa. I controlli possibili sono:

- Controllare l'ID utente dello spazio di indirizzo della regione IMS e il secondo ID utente o ID utente alternativo.
- Controllare solo l'ID utente dello spazio di indirizzi della region IMS .
- Non controllare alcun ID utente.

La seguente tabella mostra le verifiche effettuate per le connessioni IMS .

Tabella 52. Verifiche effettuate a diversi livelli di accesso RACF per connessioni IMS

RACF Livello di accesso	Livello di controllo
NESSUNO	Controllare l'ID utente dello spazio di indirizzo IMS e il secondo ID utente IMS o l'ID utente alternativo.
LETTURA	Controllare l'ID utente dello spazio di indirizzi IMS .
AGGIORNA	Controllare l'ID utente dello spazio di indirizzi IMS .
CONTROL	Nessun controllo.
MODIFICA	Nessun controllo.

### **RESLEVEL e la connessione dell'iniziatore di canali**

Per impostazione predefinita, quando un controllo di sicurezza delle risorse API viene eseguito dall'iniziatore del canale, vengono controllati due ID utente. È possibile modificare gli ID utente controllati impostando un profilo RESLEVEL.

Per impostazione predefinita, quando un controllo di sicurezza della risorsa API viene eseguito dall'iniziatore del canale, vengono controllati due ID utente per vedere se è consentito l'accesso alla risorsa.

Gli ID utente controllati possono essere quelli specificati dall'attributo del canale MCAUSER, ricevuti dalla rete, dallo spazio di indirizzo dell'iniziatore del canale o dall'ID utente alternativo per il descrittore del messaggio. Gli ID utente che vengono controllati dipendono dal protocollo di comunicazione che si sta utilizzando e dall'impostazione dell'attributo del canale PUTAUT. Per ulteriori informazioni, fare riferimento a [“ID utente utilizzati dall'iniziatore di canali”](#) a pagina 234.

Se uno di questi ID utente non ha accesso alla risorsa, la richiesta ha esito negativo con un codice di completamento MQRC\_NOT\_AUTHORIZED.

### **In che modo RESLEVEL può influire sulle verifiche effettuate**

A seconda di come si imposta il profilo RESLEVEL, è possibile modificare quali ID utente vengono controllati quando viene richiesto l'accesso a una risorsa e quanti vengono controllati.

La seguente tabella mostra i controlli effettuati per la connessione dell'iniziatore di canali e per tutti i canali poiché utilizzano questa connessione.

Tabella 53. Verifiche effettuate a diversi livelli di accesso RACF per le connessioni dell'iniziatore di canali

RACF Livello di accesso	Livello di controllo
NESSUNO	Controllare due ID utente.
LETTURA	Verificare un ID utente.
AGGIORNA	Verificare un ID utente.
CONTROL	Nessun controllo.
MODIFICA	Nessun controllo.

**Nota:** Consultare [“ID utente utilizzati dall'iniziatore di canali”](#) a pagina 234 per una definizione degli ID utente selezionati

### **RESLEVEL e accodamento all'interno del gruppo**

Per impostazione predefinita, quando un controllo di sicurezza delle risorse API viene eseguito dall'agent di accodamento all'interno del gruppo, vengono controllati due ID utente per vedere se è consentito l'accesso alla risorsa. È possibile modificare gli ID utente controllati impostando un profilo RESLEVEL.

Gli ID utente controllati possono essere l'ID utente determinato dall'attributo IGQUSER del gestore code di ricezione, l'ID utente del gestore code all'interno del gruppo di condivisione code che ha inserito il messaggio nel sistema SYSTEM.QSG.TRANSMIT.QUEUE, o l'ID utente alternativo specificato nel campo *UserIdentifier* del descrittore del messaggio. Per ulteriori informazioni, fare riferimento a [“ID utente utilizzati dall'agent di accodamento all'interno del gruppo”](#) a pagina 238.

Poiché l'agent di accodamento all'interno del gruppo è un'attività interna del gestore code, non emette una richiesta di connessione esplicita e viene eseguito con l'ID utente del gestore code. L'agent di accodamento all'interno del gruppo viene avviato all'inizializzazione del gestore code. Durante l'inizializzazione dell'agent di accodamento all'interno del gruppo, IBM MQ controlla l'accesso che l'ID utente associato al gestore code ha a un profilo della classe MQADMIN denominato:

```
hlq.RESLEVEL
```

Questo controllo viene sempre eseguito a meno che non sia stato impostato l'interruttore `hlq.NO.SUBSYS.SECURITY`.

Se non esiste alcun profilo RESLEVEL, IBM MQ abilita la verifica di due ID utente. Se è presente un profilo RESLEVEL, il livello di controllo dipende dal livello di accesso concesso all'ID utente del gestore code per il profilo. [I controlli effettuati a diversi livelli di accesso RACF\(r\) per l'agent di accodamento all'interno del gruppo](#) mostrano i controlli effettuati per l'agent di accodamento all'interno del gruppo.

*Tabella 54. Controlli effettuati a diversi livelli di accesso RACF per l'agente di accodamento all'interno del gruppo*

<b>RACF Livello di accesso</b>	<b>Livello di controllo</b>
NESSUNO	Controllare due ID utente.
LETTURA	Verificare un ID utente.
AGGIORNA	Verificare un ID utente.
CONTROL	Nessun controllo.
MODIFICA	Nessun controllo.

**Nota:** Consultare [“ID utente utilizzati dall'agent di accodamento all'interno del gruppo”](#) a pagina 238 per una definizione degli ID utente selezionati

Se le autorizzazioni concesse al profilo RESLEVEL per l'ID utente del gestore code vengono modificate, l'agent di accodamento all'interno del gruppo deve essere arrestato e riavviato per acquisire le nuove autorizzazioni. Poiché non esiste un modo per arrestare e riavviare in maniera indipendente l'agent di accodamento all'interno del gruppo, il gestore code deve essere arrestato e riavviato per ottenere tale risultato.

## **RESLEVEL e gli ID utente controllati**

Esempio di impostazione di un profilo RESLEVEL e concessione dell'accesso ad esso.

Il controllo ID utente rispetto al nome profilo per le connessioni batch tramite ID utente rispetto al nome profilo per LU 6.2 e canali di connessione server TCP/IP mostra in che modo RESLEVEL influenza gli ID utente controllati per richieste MQI differenti.

Ad esempio, si ha un gestore code denominato QM66 con i seguenti requisiti:

- L'utente WS21B deve essere esente dalla protezione delle risorse.
- L'attività avviata CICS WXNCICS in esecuzione nell'ID utente dello spazio di indirizzo CICSWXN deve eseguire il controllo completo delle risorse solo per transazioni definite con RESSEC (YES).

Per definire il profilo RESLEVEL appropriato, immettere il comando RACF riportato di seguito:

```
RDEFINE MQADMIN QM66.RESLEVEL UACC(NONE)
```

Quindi, fornire agli utenti l'accesso a questo profilo, utilizzando i seguenti comandi:

```
PERMIT QM66.RESLEVEL CLASS(MQADMIN) ID(WS21B) ACCESS(CONTROL)  
PERMIT QM66.RESLEVEL CLASS(MQADMIN) ID(CICSWXN) ACCESS(UPDATE)
```

Se si apportano queste modifiche mentre gli ID utente sono connessi al gestore code QM66, gli utenti devono disconnettersi e connettersi di nuovo prima che la modifica abbia luogo.

Se la sicurezza del sottosistema non è attiva quando un utente si connette ma, mentre questo utente è ancora connesso, la sicurezza del sottosistema diventa attiva, il controllo completo della sicurezza delle risorse viene applicato all'utente. L'utente deve riconnettersi per ottenere l'elaborazione RESLEVEL corretta.

## **ID utente per il controllo di sicurezza su z/OS**

IBM MQ avvia i controlli di sicurezza in base agli ID utente associati a utenti, terminali, applicazioni e altre risorse. Questa raccolta di argomenti elenca gli ID utente utilizzati per ciascun tipo di controllo di sicurezza.

## **ID utente per la sicurezza della connessione**

L'ID utente utilizzato per la sicurezza della connessione dipende dal tipo di collegamento.

Tipo di connessione	Contenuto ID utente
Connessione batch	L'ID utente dell'attività di connessione. Ad esempio: <ul style="list-style-type: none"><li>• L'ID utente TSO</li><li>• L'ID utente assegnato a un lavoro batch dal parametro USER JCL</li><li>• L'ID utente assegnato a un'attività avviata dalla classe STARTED o dalla tabella delle procedure avviate</li></ul>
CICS connessione	L'ID utente dello spazio di indirizzi CICS .
IMS connessione	L'ID utente dello spazio di indirizzo della regione IMS .
Connessione iniziatore di canali	L'ID utente dello spazio di indirizzo dell'iniziatore di canali.

## **ID utente per la sicurezza delle risorse di comandi e comandi**

L'ID utente utilizzato per la sicurezza del comando o per la sicurezza della risorsa del comando dipende da dove viene emesso il comando.

Emesso da ...	Contenuto ID utente
CSQINP1, CSQINP2o CSQINPT	Non viene effettuato alcun controllo.
Coda di immissione comandi di sistema	L'ID utente trovato nel <i>UserIdentifier</i> del descrittore del messaggio che contiene il comando. Se il messaggio non contiene un <i>UserIdentifier</i> , un ID utente vuoto viene passato al gestore della sicurezza.

Emesso da ...	Contenuto ID utente
Console	L'ID utente si è collegato alla console. Se la console non è collegata, l'ID utente predefinito impostato dal parametro di sistema CMDUSER in CSQ6SYSP.  Per immettere comandi da una console, la console deve avere l'attributo z/OS SYS AUTHORITY.
Console SDSF/TSO	TSO o ID utente lavoro.
Operazioni e pannelli di controllo	ID utente TSO.  Se si intende utilizzare le operazioni e i pannelli di controllo, è necessario disporre dell'autorità appropriata per immettere i comandi corrispondenti alle azioni scelte. Inoltre, è necessario disporre dell'accesso READ a tutti i hlq.DISPLAY. <i>oggetto</i> profili nella classe MQCMDS poiché i pannelli utilizzano i vari comandi DISPLAY per raccogliere le relative informazioni.
MGCRE	Se MGCRE viene utilizzato con UTOKEN, l'ID utente in UTOKEN.  Se MGCRE viene emesso senza UTOKEN, viene utilizzato il TSO o l'ID utente del lavoro.
CSQOUTIL	ID utente lavoro.
CSQUTIL	ID utente lavoro.
CSQINPX	ID utente dello spazio di indirizzo dell'iniziatore di canali.

### ► z/OS ID utente per la sicurezza della risorsa (MQOPEN, MQSUB e MQPUT1)

Queste informazioni mostrano il contenuto degli ID utente per gli ID utente normali e alternativi per ogni tipo di connessione. Il numero di controlli è definito dal profilo RESLEVEL. L'ID utente selezionato è quello utilizzato per le chiamate **MQOPEN**, **MQSUB** o **MQPUT1**.

**Nota:** Tutti i campi ID utente vengono controllati esattamente come vengono ricevuti. Non avviene alcuna conversione e, ad esempio, tre campi ID utente contenenti "Bob", "BOB" e "bob" non sono equivalenti.

### ► z/OS ID utente controllati per connessioni batch

L'ID utente selezionato per una connessione batch dipende dal modo in cui viene eseguita l'attività e se è stato specificato un ID utente alternativo.

ID utente alternativo specificato all'apertura?	hlq.ALTERNATE.USER.userid USER.userid	profilo hlq.CONTEXT.queue name	Profilo hlq.resourcename
<b>No</b>	-	JOB	JOB
<b>Sì</b>	JOB	JOB	Alt

Chiave:

**Alt**

ID utente alternativo.

**JOB**

- L'ID utente di un collegamento TSO o USS.
- L'ID utente assegnato a un lavoro batch.



- L'ID utente assegnato a un'attività avviata dalla classe STARTED o dalla tabella delle procedure avviate.
- L'ID utente associato alla procedura memorizzata Db2 in esecuzione

Un lavoro batch sta eseguendo un MQPUT1 su una coda denominata Q1 con RESLEVEL impostato su READ e la verifica dell'ID utente alternativo disattivata.

Le verifiche effettuate a diversi livelli di accesso RACF(r) per connessioni batch e Verifica ID utente rispetto al nome profilo per connessioni batch mostrano che l'ID utente lavoro viene controllato rispetto al profilo hlq.Q1.

### ID utente controllati per connessioni CICS

Gli ID utente controllati per le connessioni CICS dipendono dal fatto che sia necessario eseguire uno o due controlli e se è stato specificato un ID utente alternativo.

*Tabella 56. Verifica ID utente rispetto al nome profilo per gli ID utente di tipo CICS*

ID utente alternativo specificato all'apertura?	hlq.ALTERNATE.USER.userid USER.userid	profilo hlq.CONTEXT.queueename	Profilo hlq.resourcename
<b>No, 1 verifica</b>	-	ADS	ADS
<b>No, 2 verifiche</b>	-	ADS + TXN	ADS + TXN
<b>Si, 1 verifica</b>	ADS	ADS	ADS
<b>Si, 2 verifiche</b>	ADS + TXN	ADS + TXN	ADS + ALT

Chiave:

#### **Alt**

ID utente alternativo

#### **ADS**

L'ID utente associato al lavoro batch CICS o, se CICS è in esecuzione come attività avviata, tramite la classe STARTED o la tabella delle procedure avviate.

#### **TXN**

L'ID utente associato alla transazione CICS . Di solito è l'ID utente dell'utente del terminale che ha avviato la transazione. Può essere CICS DFLTUSER, un terminale di sicurezza PRESET o un utente collegato manualmente.

Determinare gli ID utente controllati per le seguenti condizioni:

- Il livello di accesso RACF al profilo RESLEVEL, per l'ID utente dello spazio di indirizzo CICS , è impostato su NONE.
- Viene effettuata una chiamata MQOPEN su una coda con MQOO\_OUTPUT e MQOO\_PASS\_IDENTITY\_CONTEXT.

Innanzitutto, verificare il numero di ID utente CICS controllati in base all'accesso dell'ID utente dello spazio di indirizzo CICS al profilo RESLEVEL. Da [Tabella 51 a pagina 228](#) nell'argomento “RESLEVEL e connessioni CICS” a [pagina 227](#), vengono controllati due ID utente se il profilo RESLEVEL è impostato su NONE. Quindi, da [Tabella 56 a pagina 233](#) in poi, vengono eseguiti questi controlli:

- hlq.ALTERNATE.USER.userid non è selezionato.
- Il profilo hlq.CONTEXT.queueename viene controllato con l'ID dell'utente dello spazio di indirizzi CICS e l'ID utente della transazione CICS .
- Il profilo hlq.resourcename viene controllato con l'ID utente dello spazio di indirizzi CICS e l'ID utente della transazione CICS .

Ciò significa che vengono eseguiti quattro controlli di sicurezza per questa chiamata MQOPEN .

**z/OS** ID utente controllati per connessioni IMS

Gli ID utente controllati per le connessioni IMS dipendono dal fatto che sia necessario eseguire uno o due controlli e che sia specificato un ID utente alternativo. Se viene selezionato un secondo ID utente, questo dipende dal tipo di regione dipendente e da quali ID utente sono disponibili.

*Tabella 57. Verifica ID utente rispetto al nome profilo per gli ID utente di tipo IMS*

ID utente alternativo specificato all'apertura?	hlq.ALTERNATE.USER.userid USER.userid	profilo hlq.CONTEXT.queuenam e	Profilo hlq.resourcename
<b>No, 1 verifica</b>	-	REG	REG
<b>No, 2 verifiche</b>	-	REG + SEC	REG + SEC
<b>Sì, 1 verifica</b>	REG	REG	REG
<b>Sì, 2 verifiche</b>	REG + SEC	REG + SEC	REG + ALT

Chiave:

**Alt**

ID utente alternativo.

**REG**

L'ID utente viene normalmente impostato tramite la classe STARTED o la tabella delle procedure avviate oppure, se IMS è in esecuzione, da un lavoro inoltrato, tramite il parametro USER JCL.

**SEC**

Il secondo ID utente è associato al lavoro che viene eseguito in una regione dipendente. Viene determinato in base a [Tabella 58 a pagina 234](#).

*Tabella 58. Modalità di determinazione del secondo ID utente per la connessione IMS*

Tipi di regione dipendente	Gerarchia per determinare il secondo ID utente
<ul style="list-style-type: none"> <li>• Messaggio BMP guidato e GET UNIQUE eseguito correttamente.</li> <li>• IFP e GET UNIQUE emessi.</li> <li>• MPP.</li> </ul>	<p>ID utente associato alla transazione IMS se l'utente è collegato.</p> <p>Nome LTERM, se disponibile.</p> <p>PSBNAME.</p>
<ul style="list-style-type: none"> <li>• Messaggio BMP guidato e GET UNIQUE riuscito non emesso.</li> <li>• BMP non basato sui messaggi.</li> <li>• IFP e GET UNIQUE non emessi.</li> </ul>	<p>ID utente associato allo spazio di indirizzo della regione dipendente IMS se non si tratta di tutti spazi vuoti o di tutti zeri.</p> <p>PSBNAME.</p>

**z/OS** ID utente utilizzati dall'iniziatore di canali

Questa raccolta di argomenti descrive gli ID utente utilizzati e controllati per la ricezione dei canali e per le richieste MQI client emesse sui canali di connessione server. Le informazioni vengono fornite per TCP/IP e per LU6.2

È possibile utilizzare il parametro PUTAUT della definizione del canale ricevente per stabilire il tipo di controllo di sicurezza utilizzato. Per ottenere un controllo di sicurezza coerente in tutta la rete IBM MQ, è possibile utilizzare le opzioni ONLYMCA e ALTMCA.

È possibile utilizzare il comando DISPLAY CHSTATUS per stabilire l'ID utente utilizzato da MCA.

Gli ID utente selezionati dipendono dall'opzione PUTAUT del canale e dall'eventuale esecuzione di uno o due controlli.

Tabella 59. ID utente controllati rispetto al nome profilo per i canali TCP/IP

Opzione PUTAUT specificata sul canale ricevente o richiedente	hlq.ALTERNATE.USER.userid	profilo hlq.CONTEXT.queue name	Profilo hlq.resourcename
<b>DEF, 1 verifica</b>	-	CHL	CHL
<b>DEF, 2 verifiche</b>	-	CHL + MCA	CHL + MCA
<b>CTX, 1 verifica</b>	CHL	CHL	CHL
<b>CTX, 2 controlli</b>	CHL + MCA	CHL + MCA	CHL + ALT
<b>ONLYMCA, 1 controllo</b>	-	MCA	MCA
<b>ONLYMCA, 2 verifiche</b>	-	MCA	MCA
<b>ALTMCA, 1 controllo</b>	MCA	MCA	MCA
<b>ALTMCA, 2 controlli</b>	MCA	MCA	MCA + ALT

Chiave:

**MCA (ID utente MCA)**

L'ID utente specificato per l'attributo del canale MCAUSER sul ricevitore; se vuoto, viene utilizzato l'ID utente dello spazio di indirizzo dell'iniziatore del canale del lato del destinatario o del richiedente.

**CHL (ID utente canale)**

Su TCP/IP, la protezione non è supportata dal sistema di comunicazioni per il canale. Se viene utilizzato TLS (Transport Layer Security) e un certificato digitale è stato emesso dal partner, viene utilizzato l'ID utente associato a questo certificato (se installato) o l'ID utente associato a un filtro corrispondente trovato utilizzando CNF ( RACF Certificate Name Filtering). Se non viene trovato alcun ID utente associato o se TLS non viene utilizzato, l'ID utente dello spazio di indirizzo dell'iniziatore del canale del destinatario o dell'estremità del richiedente viene utilizzato come ID utente del canale sui canali definiti con il parametro PUTAUT impostato su DEF o CTX.

**Nota:** L'utilizzo di RACF CNF (Certificate Name Filtering) consente di assegnare lo stesso ID utente RACF a più utenti remoti, ad esempio tutti gli utenti nella stessa unità organizzativa, che avrebbero naturalmente la stessa autorità di protezione. Ciò significa che il server non deve avere una copia del certificato di ogni possibile utente remoto in tutto il mondo, e semplifica notevolmente la gestione e la distribuzione dei certificati.

Se il parametro PUTAUT è impostato su ONLYMCA o ALTMCA per il canale, l'ID utente del canale viene ignorato e viene utilizzato l'ID utente MCA del destinatario o del richiedente. Ciò si applica anche ai canali TCP/IP che utilizzano TLS.

**ALT (ID utente alternativo)**

L'ID utente dalle informazioni di contesto (ovvero, il campo *UserIdentifier*) all'interno del descrittore del messaggio. Questo ID utente viene spostato nel campo *AlternateUserID* nel descrittore oggetto prima che venga emessa una chiamata **MQOPEN** o **MQPUT1** per la coda di destinazione.

Gli ID utente selezionati dipendono dall'opzione PUTAUT del canale e dall'eventuale esecuzione di uno o due controlli.

Tabella 60. ID utente controllati rispetto al nome profilo per canali LU 6.2

Opzione PUTAUT specificata sul canale ricevente o richiedente	hlq.ALTERNATE.USER.userid USER.userid	profilo hlq.CONTEXT.queue name	Profilo hlq.resourcename
<b>DEF, 1 verifica</b>	-	CHL	CHL
<b>DEF, 2 verifiche</b>	-	CHL + MCA	CHL + MCA
<b>CTX, 1 verifica</b>	CHL	CHL	CHL
<b>CTX, 2 controlli</b>	CHL + MCA	CHL + MCA	CHL + ALT
<b>ONLYMCA, 1 controllo</b>	-	MCA	MCA
<b>ONLYMCA, 2 verifiche</b>	-	MCA	MCA
<b>ALTMCA, 1 controllo</b>	MCA	MCA	MCA
<b>ALTMCA, 2 controlli</b>	MCA	MCA	MCA + ALT

Chiave:

**MCA (ID utente MCA)**

L'ID utente specificato per l'attributo del canale MCAUSER sul ricevitore; se vuoto, viene utilizzato l'ID utente dello spazio di indirizzo dell'iniziatore del canale del lato del destinatario o del richiedente.

**CHL (ID utente canale)**

**Canali richiedente-server**

Se il canale viene avviato dal richiedente, non è possibile ricevere un ID utente di rete (l'ID utente del canale).

Se il parametro PUTAUT è impostato su DEF o CTX sul canale richiedente, l'ID utente del canale è quello dello spazio di indirizzo dell'iniziatore del canale del richiedente perché non viene ricevuto alcun ID utente dalla rete.

Se il parametro PUTAUT è impostato su ONLYMCA o ALTMCA, l'ID utente del canale viene ignorato e viene utilizzato l'ID utente MCA del richiedente.

**Altri tipi di canale**

Se il parametro PUTAUT è impostato su DEF o CTX sul canale ricevente o richiedente, l'ID utente del canale è l'ID utente ricevuto dal sistema di comunicazioni quando il canale viene avviato.

- Se il canale di invio è su z/OS, l'ID utente del canale ricevuto è l'ID utente dello spazio di indirizzo dell'iniziatore del canale del mittente.
- Se il canale di invio si trova su una piattaforma differente (ad esempio, AIX o HP-UX), l'ID utente del canale ricevuto viene generalmente fornito dal parametro USERID della definizione del canale.

Se l'ID utente ricevuto è vuoto o non viene ricevuto alcun ID utente, viene utilizzato un ID utente del canale vuoto.

**ALT (ID utente alternativo)**

L'ID utente dalle informazioni di contesto (ovvero, il campo *UserIdentifier*) all'interno del descrittore del messaggio. Questo ID utente viene spostato nel campo *AlternateUserID* nel descrizione oggetto prima che venga emessa una chiamata MQOPEN o MQPUT1 per la coda di destinazione.

È possibile utilizzare diversi ID utente, a seconda di quali ID utente e variabili di ambiente sono stati impostati. Questi ID utente vengono controllati rispetto a vari profili, in base all'opzione PUTAUT utilizzata e se viene specificato un ID utente alternativo.

Questa sezione descrive gli ID utente controllati per le richieste MQI client emesse sui canali di connessione server per TCP/IP e LU 6.2. L'ID utente MCA e l'ID utente del canale sono quelli per i canali TCP/IP e LU 6.2 descritti nelle sezioni precedenti.

Per i canali di connessione server, l'ID utente ricevuto dal client viene utilizzato se l'attributo MCAUSER è vuoto.

Per ulteriori informazioni, fare riferimento a “Controllo accessi per client” a pagina 88.

Per le richieste client **MQOPEN, MQSUBe MQPUT1**, utilizzare le regole riportate di seguito per determinare il profilo selezionato:

- Se la richiesta specifica l'autorizzazione di utente alternativo, viene eseguito un controllo rispetto a *hlq.ALTERNATE.USER*. Profilo *userid*.
- Se la richiesta specifica l'autorizzazione di contesto, viene eseguito un controllo rispetto a *hlq.CONTESTO*. Profilo *queuename*.
- Per tutte le richieste **MQOPEN, MQSUBe MQPUT1**, viene effettuato un controllo sul profilo *hlq.resourcename*.

Una volta determinati i profili, utilizzare la seguente tabella per determinare quali ID utente vengono controllati rispetto a questi profili.

*Tabella 61. ID utente controllati rispetto al nome profilo per LU 6.2 e canali di connessione server TCP/IP*

<b>Opzione PUTAUT specificata sul canale di connessione e server</b>	<b>ID utente alternativo specificato all'apertura ?</b>	<b>hlq.ALTERNATE.USER.userid</b>	<b>profilo hlq.CONTEXT.queuename</b>	<b>Profilo hlq.resourcename</b>
<b>DEF, 1 verifica</b>	No	-	CHL	CHL
<b>DEF, 1 verifica</b>	Sì	CHL	CHL	CHL
<b>DEF, 2 verifiche</b>	No	-	CHL + MCA	CHL + MCA
<b>DEF, 2 verifiche</b>	Sì	CHL + MCA	CHL + MCA	CHL + ALT
<b>ONLYMCA, 1 controllo</b>	No	-	MCA	MCA
<b>ONLYMCA, 1 controllo</b>	Sì	MCA	MCA	MCA
<b>ONLYMCA, 2 verifiche</b>	No	-	MCA	MCA
<b>ONLYMCA, 2 verifiche</b>	Sì	MCA	MCA	MCA + ALT

Chiave:

### **MCA (ID utente MCA)**

L'ID utente specificato per l'attributo del canale MCAUSER sulla connessione server; se vuoto, viene utilizzato l'ID utente dello spazio di indirizzo dell'iniziatore di canali.

### **CHL (ID utente canale)**

Su TCP/IP, la protezione non è supportata dal sistema di comunicazioni per il canale. Se viene utilizzato TLS (Transport Layer Security) e un certificato digitale è stato emesso dal partner, viene utilizzato l'ID utente associato a questo certificato (se installato) o l'ID utente associato a un filtro corrispondente trovato utilizzando CNF ( RACF Certificate Name Filtering). Se non viene trovato alcun ID utente associato o se TLS non viene utilizzato, l'ID utente dello spazio di indirizzo dell'iniziatore del canale viene utilizzato come ID utente del canale sui canali definiti con il parametro PUTAUT impostato su DEF o CTX.

**Nota:** L'utilizzo di RACF CNF (Certificate Name Filtering) consente di assegnare lo stesso ID utente RACF a più utenti remoti, ad esempio tutti gli utenti nella stessa unità organizzativa, che avrebbero naturalmente la stessa autorità di protezione. Ciò significa che il server non deve avere una copia del certificato di ogni possibile utente remoto in tutto il mondo, e semplifica notevolmente la gestione e la distribuzione dei certificati.

Se il parametro PUTAUT è impostato su ONLYMCA o ALTMCA per il canale, l'ID utente del canale viene ignorato e viene utilizzato l'ID utente MCA del canale di connessione server. Ciò si applica anche ai canali TCP/IP che utilizzano TLS.

### **ALT (ID utente alternativo)**

L'ID utente dalle informazioni di contesto (ovvero, il campo *UserIdentifier*) all'interno del descrittore del messaggio. Questo ID utente viene spostato nel campo *AlternateUserID* nel descrittore dell'oggetto o della sottoscrizione prima che venga emessa una chiamata **MQOPEN**, **MQSUB** o **MQPUT1** per conto dell'applicazione client.

#### *Esempio di iniziatore di canali*

Un esempio di come gli ID utente vengono controllati rispetto ai profili RACF .

Un utente esegue un'operazione **MQPUT1** su una coda sul gestore code QM01 che si risolve in una coda denominata QB sul gestore code QM02. Il messaggio viene inviato su un canale TCP/IP denominato QM01.TO.QM02. RESLEVEL è impostato su NONE e l'apertura viene eseguita con l'ID utente alternativo e il controllo del contesto. La definizione del canale ricevente ha PUTAUT (CTX) e l'ID utente MCA è impostato. Quali ID utente vengono utilizzati sul canale ricevente per inserire il messaggio nella coda QB?

**Risposta:** Tabella 53 a pagina 229 mostra che due ID utente vengono controllati perché RESLEVEL è impostata su NONE.

Tabella 59 a pagina 235 mostra che, con PUTAUT impostato su CTX e 2 controlli, vengono controllati i seguenti ID utente:

- L'ID utente dell'iniziatore di canali e l'ID utente MCAUSER vengono controllati rispetto a hlq.ALTERNATE.USER.userid .
- L'ID utente dell'iniziatore di canali e l'ID utente MCAUSER vengono controllati rispetto al profilo hlq.CONTEXT.queueName .
- L'ID utente dell'iniziatore del canale e l'ID utente alternativo specificato nel descrittore del messaggio (MQMD) vengono verificati rispetto al profilo hlq.Q2 .

#### *ID utente utilizzati dall'agent di accodamento all'interno del gruppo*

Gli ID utente che vengono controllati quando l'agent di accodamento all'interno del gruppo apre le code di destinazione sono determinati dai valori degli attributi del gestore code IGQAUT e IGQUSER.

Gli ID utente possibili sono:

### **ID utente accodamento all'interno del gruppo (IGQ)**

L'ID utente determinato dall'attributo IGQUSER del gestore code di ricezione. Se questo valore è impostato su spazi vuoti, viene utilizzato l'ID utente del gestore code di ricezione. Tuttavia, poiché

il gestore code di ricezione dispone dell'autorità per accedere a tutte le code definite, i controlli di sicurezza non vengono eseguiti per l'ID utente del gestore code di ricezione. In questo caso:

- Se deve essere selezionato un solo ID utente e l'ID utente è quello del gestore code di ricezione, non viene eseguito alcun controllo di sicurezza. Ciò può verificarsi quando IGQAUT è impostato su ONLYIGQ o ALTIGQ.
- Se due ID utente devono essere controllati e uno degli ID utente è quello del gestore code di ricezione, i controlli di sicurezza vengono eseguiti solo per l'altro ID utente. Ciò può verificarsi quando IGQAUT è impostato su DEF, CTX o ALTIGQ.
- Se è necessario controllare due ID utente ed entrambi gli ID utente sono quelli del gestore code di ricezione, non viene eseguito alcun controllo di sicurezza. Ciò può verificarsi quando IGQAUT è impostato su ONLYIGQ.

#### **ID utente gestore code di invio (SND)**

L'ID utente del gestore code all'interno del gruppo di condivisione code che ha inserito il messaggio nel SISTEMA SYSTEM.QSG.TRANSMIT.QUEUE.

#### **ID utente alternativo (ALT)**

L'ID utente specificato nel campo *UserIdentifier* nel descrittore del messaggio.

*Tabella 62. ID utente controllati rispetto al nome profilo per l'accodamento all'interno del gruppo*

<b>Opzione IGQAUT specificata sul gestore code di ricezione</b>	<b>hlq.ALTERNATE.USER.userid USER.userid</b>	<b>profilo hlq.CONTEXT.queuename</b>	<b>Profilo hlq.resourcename</b>
<b>DEF, 1 verifica</b>	-	SDN	SDN
<b>DEF, 2 verifiche</b>	-	SND + IGQ	SND + IGQ
<b>CTX, 1 verifica</b>	SDN	SDN	SDN
<b>CTX, 2 controlli</b>	SND + IGQ	SND + IGQ	SND + ALT
<b>ONLYIGQ, 1 verifica</b>	-	IGQ	IGQ
<b>ONLYIGQ, 2 verifiche</b>	-	IGQ	IGQ
<b>ALTIGQ, 1 controllo</b>	-	IGQ	IGQ
<b>ALTIGQ, 2 controlli</b>	IGQ	IGQ	IGQ + ALT

Chiave:

#### **Alt**

ID utente alternativo.

#### **IGQ**

ID utente IGQ.

#### **SDN**

ID utente gestore code di invio.

### **ID utente vuoti e livelli UACC**

Se si verifica un ID utente vuoto, un utente non definito RACF viene collegato. Non concedere un accesso ad ampio raggio all'utente non definito.

Gli ID utente vuoti possono esistere quando un utente sta manipolando i messaggi utilizzando il contesto o la sicurezza di un utente alternativo oppure quando a IBM MQ viene passato un ID utente vuoto. Ad esempio, viene utilizzato un ID utente vuoto quando un messaggio viene scritto nella coda di input del comando di sistema senza contesto.

**Nota:** Un ID utente di " \* " (ovvero, un carattere asterisco seguito da sette spazi) viene considerato come un ID utente non definito.

IBM MQ passa l'ID utente vuoto a RACF e un RACF utente non definito è collegato. Tutti i controlli di sicurezza utilizzano quindi l'accesso universale (UACC) per il profilo pertinente. A seconda di come sono stati impostati i livelli di accesso, l'UACC potrebbe fornire all'utente non definito un accesso ad ampio raggio.

Ad esempio, se si immette questo comando RACF da TSO:

```
RDEFINE MQQUEUE Q.AVAILABLE.TO.EVERYONE UACC(UPDATE)
```

si definisce un profilo che abilita sia gli ID utente definiti da z/OS (che non sono stati inseriti nell'elenco di accesso) che l'ID utente non definito RACF a inserire e richiamare messaggi da tale coda.

Per proteggersi dagli ID utente vuoti, è necessario pianificare attentamente i livelli di accesso e limitare il numero di persone che possono utilizzare il contesto e la sicurezza dell'utente alternativo. È necessario impedire alle persone che utilizzano l'ID utente non definito RACF di ottenere l'accesso alle risorse a cui non devono accedere. Tuttavia, allo stesso tempo, è necessario consentire l'accesso a persone con ID definiti. Per fare ciò, è possibile specificare un ID utente di asterisco (\*) in un comando RACF PERMIT, fornendo l'accesso alle risorse per tutti gli ID definiti. Pertanto, tutti gli ID utente non definiti (come " \* ") è stato negato l'accesso. Ad esempio, questi comandi RACF impediscono all'ID utente non definito RACF di ottenere l'accesso alla coda per inserire o richiamare i messaggi:

```
RDEFINE MQQUEUE Q.AVAILABLE.TO.RACF.DEFINED.USERS.ONLY UACC(NONE)  
PERMIT Q.AVAILABLE.TO.RACF.DEFINED.USERS.ONLY CLASS(MQQUEUE) ACCESS(UPDATE) ID(*)
```

## ID utente z/OS e MFA (Multi - Factor Authentication)

IBM L'autenticazione a più fattori per z/OS consente agli amministratori della sicurezza di z/OS di migliorare l'autenticazione SAF, richiedendo agli utenti identificati di utilizzare più fattori di autenticazione (ad esempio, sia una password che un token crittografico) per collegarsi a un sistema z/OS . IBM MFA fornisce anche il supporto per le tecnologie di generazione della password one time basate sul tempo come RSA SecureId.

Per la maggior parte, IBM MQ non è a conoscenza del modo in cui gli utenti hanno "eseguito l'accesso" al lavoro CICS o ai sistemi batch che stanno guidando IBM MQ , la credenziale ID utente collegato è associata all'attività z/OS o allo spazio di indirizzo e IBM MQ la utilizza per controllare l'autorizzazione alle risorse. Gli ID utente abilitati per MFA possono essere utilizzati per l'autorizzazione alle risorse IBM MQ e l'autenticazione tramite i ticket di passaggio utilizzati con i bridge CICS e IMS .

**Importante:** Tuttavia, considerazioni speciali si applicano quando si utilizzano applicazioni, come ad esempio IBM MQ Explorer, che passano un ID utente e le credenziali password su una chiamata API MQCONNX con l'opzione *MQCSP\_AUTH\_USER\_ID\_AND\_PWD* . IBM MQ non ha alcuna funzione per passare una credenziale aggiuntiva su questa richiesta API.

Le limitazioni e le potenziali soluzioni temporanee sono descritte nel seguente testo.

### IBM MQ Explorer

IBM MQ Explorer non può essere utilizzato per accedere a un sistema z/OS con un id utente per cui è abilitata la MFA perché non esiste alcuna funzione per passare un secondo fattore di autenticazione da IBM MQ Explorer a z/OS.

Inoltre, esistono due diversi meccanismi utilizzati da IBM MQ Explorer per riutilizzare una credenziale ID utente e password, che richiedono un'attenzione particolare quando sono attive le password di utilizzo una sola volta:

1. IBM MQ Explorer ha la possibilità di memorizzare le parole d'ordine in un formato offuscato sulla macchina locale per l'accesso in un secondo momento. Questa capacità deve essere disabilitata



facendo in modo che explorer richieda una password ogni volta che viene effettuata una connessione al gestore code z/OS .

Per eseguire questa operazione, utilizzare la seguente procedura:

- a. Selezionare **Gestori code**.
- b. Dall'elenco visualizzato, scegliere il gestore code richiesto e fare clic con il pulsante destro del mouse su tale gestore code.
- c. Selezionare **Dettagli connessione** dall'elenco di menu visualizzato.
- d. Selezionare **Proprietà** dall'elenco di menu successivo e scegliere la scheda **ID utente** .

Accertarsi di selezionare il pulsante di opzione **Richiedi password** .

2. Diverse operazioni in IBM MQ Explorer, come l'esplorazione dei messaggi sulle code, la verifica delle sottoscrizioni e così via, avviano un nuovo thread che si autentica su IBM MQ utilizzando la credenziale utilizzata per la prima volta al momento dell'accesso. Poiché le credenziali della password non possono essere riutilizzate, non è possibile utilizzare queste operazioni.

Esistono due possibili soluzioni temporanee a livello di configurazione MFA per questi problemi:

- Utilizzare l'esclusione ID applicazione di MFA per escludere completamente le attività IBM MQ dall'elaborazione MFA.

A tale scopo, immettere i seguenti comandi:

```
1. RDEFINE MFADEF MFABYPASS.USERID.chinuser
```

dove *chinuser* è l'ID utente a livello di spazio di indirizzo dell'inziatore di canale (associato all'inziatore di canale tramite la classe STC)

```
2. PERMIT MFABYPASS.USERID.chinuser CLASS MFADEF ACCESS(READ) ID(explorer user)
```

Per ulteriori informazioni su questo approccio, vedi [Bypassare IBM MFA for applications](#).

- Utilizza il supporto fuori banda su MFA, che è stato introdotto con IBM MFA 1.2. Con questo approccio, presegui l'autenticazione sul server web MFA IBM e, oltre al tuo ID utente e alla tua password, specifica un'altra autenticazione come determinato dalla politica. Il server IBM MFA genera una credenziale token cache che specifichi nella finestra di dialogo di autenticazione IBM MQ Explorer . L'amministratore della sicurezza può consentire la ripetizione di questa credenziale per un periodo di tempo ragionevole, consentendo quindi il normale utilizzo di IBM MQ Explorer .

Per ulteriori informazioni su questo approccio, vedi [Introduzione a IBM MFA](#).

## **IBM MQ for z/OS Gestione della sicurezza**

IBM MQ utilizza una tabella in memoria per conservare le informazioni relative a ogni utente e le richieste di accesso effettuate da ogni utente. Per gestire questa tabella in modo efficiente e ridurre il numero di richieste effettuate da IBM MQ al gestore di sicurezza esterno (ESM), sono disponibili diversi controlli.

Questi controlli sono disponibili tramite le operazioni e i pannelli di controllo e i comandi IBM MQ .

### **Nuova verifica ID utente**

Se la definizione RACF di un utente che utilizza le risorse IBM MQ è stata modificata, ad esempio collegando l'utente a un nuovo gruppo, è possibile indicare al gestore code di collegarsi nuovamente a questo utente la volta successiva che tenta di accedere a una risorsa IBM MQ . È possibile eseguire questa operazione utilizzando il comando IBM MQ RVERIFY SECURITY.

- L'utente HX0804 sta ricevendo e inserendo messaggi nelle code PAYROLL sul gestore code PRD1. Tuttavia HX0804 ora richiede l'accesso ad alcune code PENSION sullo stesso gestore code (PRD1).
- L'amministratore della protezione dei dati collega l'utente HX0804 al gruppo RACF che consente l'accesso alle code PENSION.

- In modo che HX0804 possa accedere immediatamente alle code PENSION (vale a dire, senza arrestare il gestore code PRD1 o attendere il timeout di HX0804 ), è necessario utilizzare il comando IBM MQ :

```
RVERIFY SECURITY(HX0804)
```

**Nota:** Se si disattiva il timeout dell'ID utente per lunghi periodi di tempo (giorni o anche settimane) mentre il gestore code è in esecuzione, è necessario ricordare di eseguire il comando RVERIFY SECURITY per tutti gli utenti che sono stati revocati o eliminati in quel momento.

### **Timeout ID utente**

È possibile fare in modo che IBM MQ scollegi un utente da un gestore code dopo un periodo di inattività.

Quando un utente accede a una risorsa IBM MQ , il gestore code tenta di collegarsi al gestore code (se la sicurezza del sottosistema è attiva). Ciò significa che l'utente è autenticato su ESM. Questo utente rimane collegato a IBM MQ fino a quando il gestore code non viene arrestato o fino a quando l'ID utente non viene *scaduto* (l'autenticazione scade) o ripristinato (riautenticato).

Quando un utente va in timeout, l'ID utente è *scollegato* all'interno del gestore code e tutte le informazioni relative alla sicurezza conservate per questo utente vengono eliminate. Il collegamento e lo scollegamento dell'utente nel gestore code non sono evidenti al programma applicativo o all'utente.

Gli utenti sono idonei per il timeout quando non hanno utilizzato alcuna risorsa IBM MQ per un periodo di tempo predeterminato. Questo periodo di tempo è impostato dal comando MQSC ALTER SECURITY.

È possibile specificare due valori nel comando ALTER SECURITY:

#### **TIMEOUT**

Il periodo di tempo in minuti in cui un ID utente non utilizzato e le relative risorse associate possono rimanere all'interno del gestore code IBM MQ .

#### **INTERVAL**

Il periodo di tempo in minuti tra i controlli per gli ID utente e le relative risorse associate, per determinare se il *TIMEOUT* è scaduto.

Ad esempio, se il valore *TIMEOUT* è 30 e il valore *INTERVAL* è 10, ogni 10 minuti IBM MQ controlla gli ID utente e le risorse associate per determinare se alcuni non sono stati utilizzati per 30 minuti. Se viene rilevato un ID utente scaduto, allora tale ID utente verrà scollegato dal gestore code. Se vengono trovate informazioni sulle risorse in timeout associate a ID utente non in timeout, tali informazioni sulle risorse vengono eliminate. Se non si desidera eseguire il timeout degli ID utente, impostare il valore *INTERVAL* su zero. Tuttavia, se il valore *INTERVAL* è zero, la memoria occupata dagli ID utente e dalle relative risorse associate non viene liberata fino a quando non si immette un comando **REFRESH SECURITY** o **RVERIFY SECURITY** .

L'ottimizzazione di questo valore può essere importante se si dispone di molti utenti una tantum. Se si impostano valori di intervallo e timeout ridotti, le risorse non più necessarie vengono liberate.

**Nota:** Se si utilizzano valori per *INTERVAL* o *TIMEOUT* diversi da quelli predefiniti, è necessario reimmettere il comando ad ogni avvio del gestore code. È possibile eseguire questa operazione automaticamente inserendo il comando **ALTER SECURITY** nel dataset CSQINP1 per tale gestore code.

### **Aggiornamento della sicurezza del gestore code su z/OS**

IBM MQ for z/OS memorizza nella cache RACF i dati per migliorare le prestazioni. Quando si modificano alcune classi di protezione, è necessario aggiornare queste informazioni memorizzate nella cache. Aggiornare la sicurezza raramente, per motivi di prestazioni. È anche possibile scegliere di aggiornare solo le informazioni di sicurezza TLS.

Quando una coda viene aperta per la prima volta (o per la prima volta dopo un aggiornamento della sicurezza), IBM MQ esegue un controllo RACF per ottenere i diritti di accesso dell'utente e inserisce queste informazioni nella cache. I dati memorizzati nella cache includono gli ID utente e le risorse su cui è stato eseguito il controllo di sicurezza. Se la coda viene aperta di nuovo dallo stesso utente, la presenza dei dati memorizzati nella cache significa che IBM MQ non deve emettere i controlli RACF , il che migliora le prestazioni. L'azione di un aggiornamento della sicurezza è quella di eliminare tutte le informazioni di

sicurezza memorizzate nella cache e quindi forzare IBM MQ a effettuare un nuovo controllo rispetto a RACF. Ogni volta che si aggiunge, si modifica o si elimina un profilo di risorsa RACF contenuto nella classe MQADMIN, MXADMIN, MQPROC, MXPROC, MQQUEUE, MXQUEUE, MQNLIST, MXNLIST o MXTOPIC, è necessario indicare ai gestori code che utilizzano questa classe di aggiornare le informazioni di sicurezza che essi contengono. A tale scopo, immettere i seguenti comandi:

- Il comando RACF SETROPTS RACLIST (classname) REFRESH per aggiornare al livello RACF .
- Il comando IBM MQ REFRESH SECURITY per aggiornare le informazioni di sicurezza del gestore code. Questo comando deve essere immesso da ciascun gestore code che accede ai profili modificati. Se si dispone di un gruppo di condivisione code, è possibile utilizzare l'attributo dell'ambito del comando per indirizzare il comando a tutti i gestori code nel gruppo.

**Nota:** Se è stato connesso un nuovo utente ad un gruppo esistente, è necessario eseguire il comando IBM MQ RVERIFY SECURITY(userid). Il comando REFRESH SECURITY (\*) non consente al gestore code di accedere nuovamente a questo utente, la volta successiva che tenta di accedere a una risorsa IBM MQ .

Se si utilizzano profili generici in una delle classi IBM MQ , è necessario immettere anche i normali comandi di aggiornamento RACF se si modificano, si aggiungono o si eliminano profili generici. Ad esempio, SETROPTS GENERIC (nome classe) AGGIORNA.

Tuttavia, se un profilo risorsa RACF viene aggiunto, modificato o eliminato e la risorsa a cui si applica non è ancora stata acceduta (quindi nessuna informazione viene memorizzata nella cache), IBM MQ utilizza le nuove informazioni RACF senza che venga emesso un comando REFRESH SECURITY.

Se il controllo RACF è attivato, (ad esempio, utilizzando il comando RACF RALTER AUDIT (access - tentativi (audit\_access\_level))), non viene eseguita alcuna memorizzazione nella cache e quindi IBM MQ fa riferimento direttamente allo spazio dati RACF per ogni controllo. Le modifiche vengono quindi rilevate immediatamente e REFRESH SECURITY non è necessario per accedere alle modifiche. È possibile confermare se il controllo RACF è attivo utilizzando il comando RACF RLIST. Ad esempio, è possibile immettere il comando

```
RLIST MQQUEUE (qmgx.SYSTEM.COMMAND.INPUT) GEN
```

e ricevere i risultati

```
CLASS      NAME
-----
MQQUEUE    QP*.SYSTEM.COMMAND.*.* (G)
  AUDITING
  -----
  FAILURES(READ)
```

Ciò indica che il controllo è attivo. Per ulteriori informazioni, fare riferimento al manuale *z/OS Security Server RACF Auditor's Guide* e al manuale *z/OS Security Server RACF Command Language Reference*.

Figura 17 a pagina 244 riepiloga le situazioni in cui le informazioni di sicurezza vengono memorizzate nella cache e in cui vengono utilizzate.

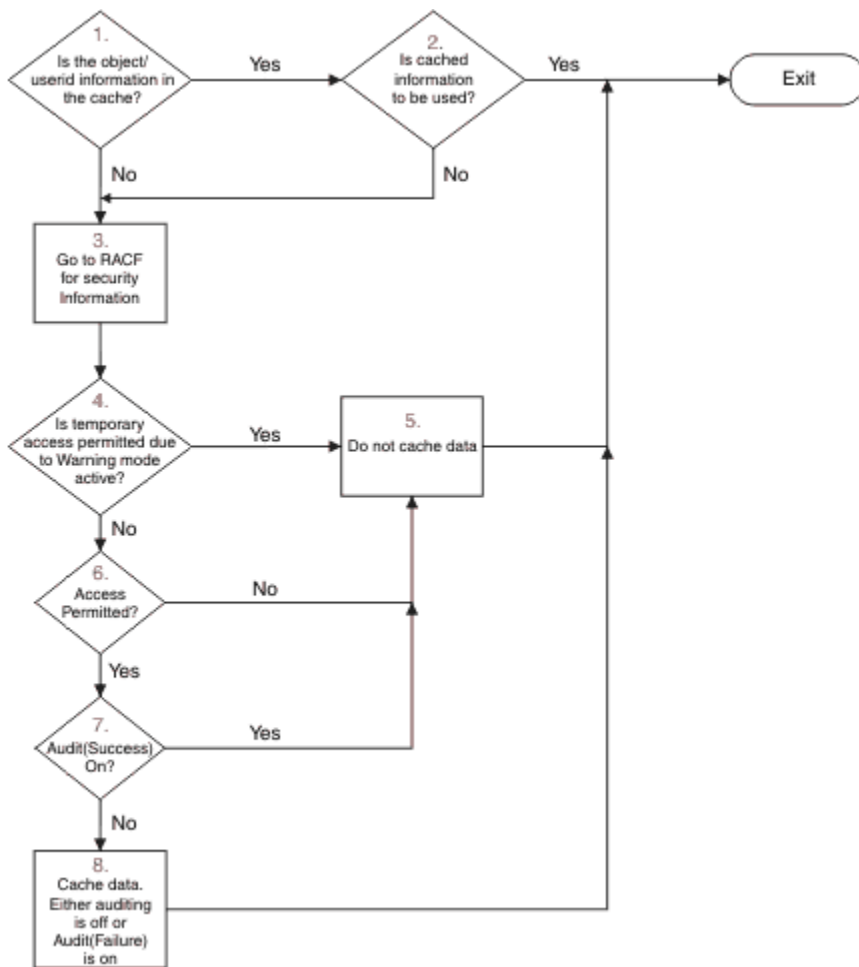


Figura 17. Flusso logico per la memorizzazione nella cache della sicurezza IBM MQ

Se si modificano le impostazioni di sicurezza aggiungendo o eliminando i profili switch nelle classi MQADMIN o MXADMIN, utilizzare uno di questi comandi per applicare dinamicamente queste modifiche:

```

AGGIORNA SICUREZZA (*)
AGGIORNA SICUREZZA (MQADMIN)
AGGIORNA SICUREZZA (MXADMIN)
  
```

Ciò significa che è possibile attivare nuovi tipi di sicurezza o disattivarli senza dover riavviare il gestore code.

Per motivi di prestazioni, queste sono le uniche classi interessate dal comando REFRESH SECURITY. Non è necessario utilizzare REFRESH SECURITY se si modifica un profilo nelle classi MQCONN o MQCMD5.

**Nota:** Un aggiornamento della classe MQADMIN o MXADMIN non è richiesto se si modifica un profilo di sicurezza RESLEVEL.

Per motivi di prestazioni, utilizzare REFRESH SECURITY il più raramente possibile, idealmente in orari non di picco. È possibile ridurre al minimo il numero di aggiornamenti della sicurezza collegando gli utenti ai gruppi RACF che sono già nell'elenco di accesso per profili IBM MQ, piuttosto che inserire singoli utenti negli elenchi di accesso. In questo modo, si modifica l'utente piuttosto che il profilo risorsa. È anche possibile RVERIFY SECURITY l'utente appropriato invece di aggiornare la sicurezza.

Come esempio di REFRESH SECURITY, si supponga di definire i nuovi profili per proteggere l'accesso alle code che iniziano con INSURANCE.LIFE sul gestore code PRMQ. Utilizzare questi comandi RACF:

```

RDEFINE MQQUEUE PRMQ.INSURANCE.LIFE.** UACC(NONE)
PERMIT PRMQ.INSURANCE.LIFE.** ID(LIFEGRP) ACCESS(UPDATE)
  
```

È necessario immettere il seguente comando per indicare a RACF di aggiornare le informazioni di sicurezza che contiene, ad esempio:

```
SETROPTS RACLIST(MQQUEUE) REFRESH
```

Poiché tali profili sono generici, è necessario indicare a RACF di aggiornare i profili generici per MQQUEUE. Ad esempio:

```
SETROPTS GENERIC(MQQUEUE) REFRESH
```

Quindi, è necessario utilizzare questo comando per indicare al gestore code PRMQ che i profili coda sono cambiati:

```
REFRESH SECURITY(MQQUEUE)
```

## Aggiornamento della sicurezza SSL/TLS

Per aggiornare la vista memorizzata nella cache del repository delle chiavi TLS, immettere il comando REFRESH SECURITY con l'opzione TYPE (SSL). Ciò ti consente di aggiornare alcune delle tue impostazioni TLS senza dover riavviare il tuo iniziatore di canale.

## Visualizzazione dello stato di sicurezza

Per visualizzare lo stato degli switch di sicurezza e altri controlli di sicurezza, immettere il comando MQSC DISPLAY SECURITY.

La seguente figura mostra l'output tipico del comando DISPLAY SECURITY ALL.

```
CSQH015I +CSQ1 Security timeout = 54 MINUTES
CSQH016I +CSQ1 Security interval = 12 MINUTES
CSQH030I +CSQ1 Security switches ...
CSQH034I +CSQ1 SUBSYSTEM: ON, 'SQ05.NO.SUBSYS.SECURITY' not found
CSQH032I +CSQ1 QMGR: ON, 'CSQ1.YES.QMGR.CHECKS' found
CSQH031I +CSQ1 QSG: OFF, 'SQ05.NO.QSG.CHECKS' found
CSQH031I +CSQ1 CONNECTION: OFF, 'CSQ1.NO.CONNECT.CHECKS' found
CSQH034I +CSQ1 COMMAND: ON, 'CSQ1.NO.COMMAND.CHECKS' not found
CSQH031I +CSQ1 CONTEXT: OFF, 'CSQ1.NO.CONTEXT.CHECKS' found
CSQH034I +CSQ1 ALTERNATE USER: ON, 'CSQ1.NO.ALTERNATE.USER.CHECKS' not found
CSQH034I +CSQ1 PROCESS: ON, 'CSQ1.NO.PROCESS.CHECKS' not found
CSQH034I +CSQ1 NAMELIST: ON, 'CSQ1.NO.NLIST.CHECKS' not found
CSQH034I +CSQ1 QUEUE: ON, 'CSQ1.NO.QUEUE.CHECKS' not found
CSQH034I +CSQ1 TOPIC: ON, 'CSQ1.NO.TOPIC.CHECKS' not found
CSQH031I +CSQ1 COMMAND RESOURCES: OFF, 'CSQ1.NO.CMD.RESC.CHECKS' found
CSQ9022I +CSQ1 CSQHPDTC ' DISPLAY SECURITY' NORMAL COMPLETION
```

Figura 18. Output tipico del comando DISPLAY SECURITY

L'esempio mostra che il gestore code che ha risposto al comando ha il sottosistema, il comando, l'utente alternativo, il processo, l'elenco nomi e la sicurezza della coda attivi a livello di gestore code ma non a livello di gruppo di condivisione code. La connessione, la risorsa comando e la sicurezza del contesto non sono attive. Mostra inoltre che i timeout degli ID utente sono attivi e che ogni 12 minuti il gestore code controlla gli ID utente che non sono stati utilizzati in questo gestore code per 54 minuti e li rimuove.

**Nota:** Questo comando mostra lo stato di sicurezza corrente. Non riflette necessariamente lo stato corrente dei profili switch definiti in RACF o lo stato delle classi RACF. Ad esempio, i profili switch potrebbero essere stati modificati dall'ultimo riavvio di questo gestore code o dal comando REFRESH SECURITY.

## Attività di installazione di sicurezza per z/OS

Dopo l'installazione e la personalizzazione di IBM MQ, autorizzare le procedure di attività avviate per RACF, autorizzare l'accesso a varie risorse e impostare definizioni RACF . Facoltativamente, configurare il sistema per TLS.

Quando IBM MQ viene installato e personalizzato per la prima volta, è necessario eseguire queste attività relative alla sicurezza:

1. Impostare la sicurezza del sistema e del set di dati IBM MQ :
  - Autorizzazione della procedura dell'attività avviata del gestore code xxxxMSTR e della procedura dell'attività avviata dell'accodamento distribuita xxxxCHIN da eseguire in RACF.
  - Autorizzazione dell'accesso ai dataset del gestore code.
  - Autorizzazione dell'accesso alle risorse per gli ID utente che utilizzeranno il programma di utilità e il gestore code.
  - Autorizzazione dell'accesso per i gestori code che utilizzeranno le strutture dell'elenco CF (Coupling Facility).
  - Autorizzazione dell'accesso per i gestori code che utilizzeranno Db2.
2. Impostare le definizioni RACF per la sicurezza IBM MQ .
3. Se si desidera utilizzare TLS (Transport Layer Security), preparare il sistema all'utilizzo di certificati e chiavi.

## Impostazione della sicurezza del dataset IBM MQ for z/OS

Esistono molti tipi di utenti IBM MQ . Utilizzare RACF per controllare l'accesso ai dataset di sistema.

I possibili utenti dei dataset IBM MQ includono le seguenti entità:

- Il gestore code stesso.
- Il programma di avvio dei canali
- Gli amministratori IBM MQ , che devono creare dataset IBM MQ , eseguire programmi di utilità e attività simili.
- Programmatori di applicazioni che devono utilizzare i copybook forniti da IBM MQ , includono dataset, macro e risorse simili.
- Applicazioni che coinvolgono una o più di:
  - Lavori batch
  - Utenti TSO
  - Regioni CICS
  - Regioni IMS
- Dataset CSQOUTX e CSQSNAP
- Code dinamiche SYSTEM.CSQXCMD.\*

Per tutti questi potenziali utenti, proteggere i dataset IBM MQ con RACF.

È inoltre necessario controllare l'accesso a tutti i dataset 'CSQINP'.

## Autorizzazione RACF di procedure di attività avviate

Alcuni dataset IBM MQ sono per l'uso esclusivo del gestore code. Se si proteggono i data set IBM MQ utilizzando RACF, è necessario autorizzare anche la procedura dell'attività avviata del gestore code xxxxMSTR e la procedura dell'attività avviata dell'accodamento distribuita xxxxCHIN, utilizzando RACF. Per fare ciò, utilizzare la classe STARTED. In alternativa, è possibile utilizzare la tabella delle procedure avviate (ICHRIN03), ma è necessario eseguire un IPL del proprio sistema z/OS prima che le modifiche diventino effettive.

Per ulteriori informazioni, consultare il manuale *z/OS Security Server RACF System Programmer's Guide*.

L'ID utente RACF identificato deve avere l'accesso richiesto ai dataset nella procedura dell'attività avviata. Ad esempio, se si associa una procedura dell'attività avviata del gestore code denominata CSQ1MSTR con RACF ID utente QMGRCSQ1, l'ID utente QMGRCSQ1 deve avere accesso alle risorse z/OS a cui accede il gestore code CSQ1 .

Inoltre, il contenuto del campo GROUP nell'ID utente del gestore code deve essere uguale al contenuto del campo GROUP nel profilo STARTED per tale gestore code. Se il contenuto in ogni campo GROUP non corrisponde, viene impedito all'ID utente appropriato di entrare nel sistema. Questa situazione fa sì che IBM MQ venga eseguito con un ID utente non definito e di conseguenza chiuso a causa di una violazione della sicurezza.

Gli ID utente RACF associati alle procedure dell'attività avviata del gestore code e dell'iniziatore di canale non devono avere impostato l'attributo TRUSTED.

### **Autorizzazione dell'accesso ai dataset**

I dataset IBM MQ devono essere protetti in modo che nessun utente non autorizzato possa eseguire un'istanza del gestore code o ottenere l'accesso ai dati del gestore code. A tale scopo, utilizzare la normale protezione del set di dati z/OS RACF .

Tabella 63 a pagina 247 riassume l' RACF accesso che la procedura dell'attività avviata dal gestore code deve avere ai diversi dataset.

RACFaccesso	I data set
LETTURA	<ul style="list-style-type: none"> <li>• thlqual.SCSQAUTH e thlqual.SCSQANLx (dove x è la lettera della lingua per la propria lingua nazionale).</li> <li>• I dataset a cui fanno riferimento CSQINP1, CSQINP2 e CSQXLIB nella procedura dell'attività avviata del gestore code.</li> <li>• Dataset SMDS di proprietà di altri gestori code del gruppo.</li> <li>• Dataset di log, BSDS e log di archivio per altri gestori code nel gruppo.</li> </ul>
AGGIORNA	<ul style="list-style-type: none"> <li>• Tutte le serie di pagine e le serie di dati di log e BSDS.</li> <li>• Data set SMDS di proprietà di un gestore code</li> </ul>
MODIFICA	<ul style="list-style-type: none"> <li>• Tutti i dataset del log di archiviazione.</li> </ul>

Tabella 64 a pagina 247 riassume l'accesso RACF che la procedura di attività avviata per l'accodamento distribuito deve avere per i diversi dataset.

RACFaccesso	I data set
LETTURA	<ul style="list-style-type: none"> <li>• thlqual.SCSQAUTH, thlqual.SCSQANLx (dove x è la lettera della lingua per la propria lingua nazionale) e thlqual.SCSQMVR1.</li> <li>• Dataset della libreria LE.</li> <li>• I dataset a cui fanno riferimento CSQXLIB e CSQINPX nella procedura dell'attività avviata dall'iniziatore di canali.</li> </ul>
AGGIORNA	<ul style="list-style-type: none"> <li>• Dataset CSQOUTX e CSQSNAP</li> </ul>

Per ulteriori informazioni, consultare [z/OS Security Server RACF Security Administrator's Guide](#).

### **Impostazione della sicurezza delle risorse IBM MQ for z/OS**

Esistono molti tipi di utenti IBM MQ . Utilizzare RACF per controllare l'accesso alle risorse IBM MQ .

I possibili utenti delle risorse IBM MQ , come code e canali, includono le seguenti entità:

- Il gestore code stesso.
- Il programma di avvio dei canali
- Amministratori IBM MQ , che devono creare dataset IBM MQ , eseguire programmi di utilità e attività simili
- Programmatori di applicazioni che devono utilizzare i copybook forniti da IBM MQ, includono dataset, macro e risorse simili.
- Applicazioni che coinvolgono una o più di:
  - Lavori batch
  - Utenti TSO
  - Regioni CICS
  - Regioni IMS
- Dataset CSQOUTX e CSQSNAP
- Code dinamiche SYSTEM.CSQXCMD.\*

Per tutti questi potenziali utenti, proteggere le risorse IBM MQ con RACF. In particolare, tenere presente che l'iniziatore di canali deve accedere a varie risorse, come descritto in [“Considerazioni sulla protezione per l'iniziatore di canali su z/OS” a pagina 254](#), quindi l'ID utente con cui viene eseguito deve essere autorizzato ad accedere a queste risorse.

Se si sta utilizzando un gruppo di condivisione code, il gestore code potrebbe emettere diversi comandi internamente, quindi l'ID utente utilizzato deve essere autorizzato a immettere tali comandi. I comandi sono:

- DEFINE, ALTER e DELETE per ogni oggetto che ha QSGDISP (GROUP)
- START e STOP CHANNEL per ogni canale utilizzato con CHLDISP (SHARED)

### **Configurazione del sistema z/OS per utilizzare TLS**

Utilizzare questo argomento come esempio per configurare IBM MQ for z/OS con TLS (Transport Layer Security) utilizzando i comandi RACF .

Se si desidera utilizzare TLS per la sicurezza del canale, è necessario eseguire una serie di attività sul sistema. (Per i dettagli sull'uso dei comandi RACF per i certificati e i repository delle chiavi (key ring), consultare [Utilizzo di TLS su z/OS](#) .)

1. Creare un keyring in RACF per contenere tutte le chiavi e i certificati per il sistema, utilizzando il comando RACDCERT RACF . Ad esempio:

```
RACDCERT ID(CHINUSER) ADDRING(QM1RING)
```

L'ID deve essere l'ID utente dello spazio di indirizzo dell'iniziatore del canale o l'ID utente che si desidera possedere il keyring se deve essere un keyring condiviso.

2. Creare un certificato digitale per ciascun gestore code utilizzando il comando RACF RACDCERT.

L'etichetta del certificato deve essere il valore dell'attributo IBM MQ **CERTLABL** , se impostato, o il valore predefinito `ibmWebSphereMQ` con il nome del gestore code o il gruppo di condivisione code accodato. Per i dettagli, consultare [Etichetta certificato digitale](#) . In questo esempio è `ibmWebSphereMQQM1`.

Ad esempio:



```
RACDCERT ID(USERID) GENCERT
SUBJECTSDN(CN('username') O('IBM') OU('departmentname') C('England'))
WITHLABEL('ibmWebSphereMQM1')
```

3. Collegare il certificato in RACF al key ring, utilizzando il comando RACDCERT RACF . Ad esempio:

```
RACDCERT CONNECT(ID(USERID) LABEL('ibmWebSphereMQM1') RING(QM1RING))
CONNECT ID(CHINUSER)
```

È inoltre necessario collegare tutti i certificati del firmatario rilevanti (da un'autorità di certificazione) al key ring. Vale a dire, tutte le autorità di certificazione per il certificato TLS di questo gestore code e tutte le autorità di certificazione per tutti i certificati TLS con cui comunica questo gestore code. Ad esempio:

```
RACDCERT ID(CHINUSER)
CONNECT(CERTAUTH LABEL('My CA') RING(QM1RING) USAGE(CERTAUTH))
```

4. Su ciascuno dei gestori code, utilizzare il comando IBM MQ ALTER QMGR per specificare il repository delle chiavi a cui il gestore code deve puntare. Ad esempio, se il keyring è di proprietà dello spazio di indirizzo dell'iniziatore di canali:

```
ALTER QMGR SSLKEYR(QM1RING)
```

o se si sta utilizzando un keyring condiviso:

```
ALTER QMGR SSLKEYR(userid/QM1RING)
```

dove *userid* è l'ID utente che possiede il keyring condiviso.

5. I CRL (Certificate Revocation List) consentono alle autorità di certificazione di revocare i certificati che non possono più essere considerati attendibili. I CRL sono memorizzati nei server LDAP. Per accedere a questo elenco sul server LDAP, è necessario prima creare un oggetto AUTHINFO di AUTHTYPE CRLLDAP utilizzando il comando IBM MQ DEFINE AUTHINFO. Ad esempio:

```
DEFINE AUTHINFO(LDAP1)
AUTHTYPE(CRLLDAP)
CONNNAME(ldap.server(389))
LDAPUSER('')
LDAPPWD('')
```

In questo esempio, il CRL (Certificate Revocation List) viene memorizzato in un'area pubblica del server LDAP, quindi i campi LDAPUSER e LDAPPWD non sono necessari.

Successivamente, inserire l'oggetto AUTHINFO in un elenco nomi, utilizzando il comando IBM MQ DEFINE NAMELIST. Ad esempio:

```
DEFINE NAMELIST(LDAPNL) NAMES(LDAP1)
```

Infine, associare l'elenco nomi a ciascun gestore code, utilizzando il comando IBM MQ ALTER QMGR. Ad esempio:

```
ALTER QMGR SSLCRLNL(LDAPNL)
```

6. Impostare il gestore code per eseguire chiamate TLS, utilizzando il comando IBM MQ ALTER QMGR. Ciò definisce le attività secondarie del server che gestiscono solo chiamate SSL, il che lascia i normali dispatcher per continuare l'elaborazione come normale senza essere influenzati da alcuna chiamata SSL. È necessario disporre di almeno due di queste attività secondarie. Ad esempio:

```
ALTER QMGR SSLTASKS(8)
```

Questa modifica diventa effettiva solo quando l'iniziatore di canali viene riavviato.

7. Specificare la specifica di cifratura da utilizzare per ciascun canale, utilizzando il comando IBM MQ DEFINE CHANNEL o ALTER CHANNEL. Ad esempio:

```
ALTER CHANNEL(LDAPCHL)  
CHLTYPE(SDR)  
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA)
```

Entrambe le estremità del canale devono specificare la stessa specifica di codifica.

## Gestione dei record di autenticazione di canale in un QSG

I record di autenticazione di canale si applicano al gestore code su cui vengono creati, non vengono condivisi all'interno del gruppo di condivisione code (QSG). Pertanto, se tutti i gestori code nel gruppo di condivisione code devono avere le stesse regole, è necessario eseguire una gestione per mantenere tutte le regole congruenti.

1. Aggiungere sempre l'opzione CMDSCOPE(\*) a tutti i comandi SET CHLAUTH. Il comando verrà inviato a tutti i gestori code in esecuzione nel gruppo di condivisione code
2. Utilizzare il comando DISPLAY CHLAUTH con l'opzione CMDSCOPE(\*) e quindi analizzare le risposte per vedere se i record sono gli stessi da tutti i gestori code. Quando viene rilevata un'incongruenza, è possibile immettere un comando SET CHLAUTH contenente la stessa regola con CMDSCOPE(\*) o CMDSCOPE(*qmgr-name*).
3. Aggiungere un membro alla concatenazione CSQINP2 del gestore code (per i dettagli, consultare [Comandi di inizializzazione](#)) che dispone della serie completa di regole. Questi verranno letti come parte del processo di inizializzazione del gestore code. Se il comando SET CHLAUTH utilizza ACTION(ADD), la regola verrà aggiunta solo se non esiste. L'utilizzo di ACTION(REPLACE) sostituirà una regola esistente se esiste già oppure la aggiungerà se non esiste. Lo stesso membro potrebbe quindi essere inserito nella concatenazione CSQINP2 di tutti i gestori code nel gruppo di condivisione code.
4. Utilizzare il programma di utilità CSQUTIL (consultare [Emissione di comandi in IBM MQ \(COMMAND\)](#) per i dettagli) per estrarre le regole da un gestore code utilizzando l'opzione MAKEDEF o MAKEREP. Quindi ripetere l'output utilizzando CSQUTIL nel gestore code di destinazione.

### Concetti correlati

#### [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.

## z/OS Considerazioni sul controllo su z/OS

I normali controlli di controllo RACF sono disponibili per eseguire un controllo di sicurezza di un gestore code. IBM MQ non raccoglie alcuna statistica di sicurezza. Le uniche statistiche sono quelle che possono essere create tramite il controllo.

Il controllo RACF può essere basato su:

- ID utente
- Classi di risorse
- Profili

Per ulteriori dettagli, consultare il manuale *z/OS Security Server RACF Auditor's Guide*.

**Nota:** Il controllo riduce le prestazioni; più il controllo viene implementato, più le prestazioni vengono degradate. Questa è anche una considerazione per l'utilizzo dell'opzione RACF WARNING.

## z/OS Controllo RESLEVEL

Utilizzare il parametro di sistema RESAUDIT per controllare la produzione dei record di controllo RESLEVEL. RACF vengono prodotti record di controllo GENERALE.

Produrre i record di controllo RESLEVEL impostando il parametro di sistema RESAUDIT su YES. Se il parametro RESAUDIT è impostato su NO, i record di controllo non vengono prodotti. Per ulteriori dettagli sull'impostazione di questo parametro, consultare [Utilizzo di CSQ6SYSP](#).

Se RESAUDIT è impostato su YES, non viene eseguito alcun record di controllo RACF normale quando viene eseguito il controllo RESLEVEL per visualizzare l'accesso di un ID utente dello spazio di indirizzi al profilo hlq.RESLEVEL. Invece, IBM MQ richiede che RACF crei un record di controllo GENERALE (evento numero 27). Questi controlli vengono eseguiti solo al momento della connessione, quindi il costo delle prestazioni è minimo.

È possibile riportare i record di controllo generali IBM MQ utilizzando il writer di report RACF (RACFRW). È possibile utilizzare i seguenti comandi RACFRW per notificare l'accesso RESLEVEL:

```
RACFRW
SELECT PROCESS
EVENT GENERAL
LIST
END
```

Un report di esempio da RACFRW, esclusi i campi *Date*, *Time* e *SYSID*, viene mostrato in [Figura 19](#) a pagina 251.

```

RACF REPORT - LISTING OF PROCESS RECORDS                                PAGE 4
E
V Q
E U
N A
*JOB/USER *STEP/  --TERMINAL--  N A
NAME      GROUP   ID      LVL  T  L
WS21B    MQMGRP IGJZM000  0   27 0  JOBID=(WS21B 05.111 09:44:57),USERDATA=(
TRUSTED  USER                                AUTH=(NONE),REASON=(NONE)
SESSION=TSOLOGON,TERMINAL=IGJZM000,
LOGSTR='CSQH RESLEVEL CHECK PERFORMED AGAINST
PROFILE(QM66.RESLEVEL),
CLASS(MQADMIN), ACCESS EQUATES TO
(CONTROL)',RESULT=SUCCESS,MQADMIN
```

Figura 19. Output di esempio da RACFRW che mostra i record di controllo generali RESLEVEL

Controllando i dati LOGSTR in questo output di esempio, è possibile vedere che l'utente TSO WS21B dispone dell'accesso CONTROL a QM66.RESLEVEL. Ciò significa che tutti i controlli di sicurezza delle risorse vengono ignorati quando l'utente WS21B accede alle risorse QM66 .

Per ulteriori informazioni sull'utilizzo di RACFRW, consultare il manuale *z/OS Security Server RACF Auditor's Guide*.

## Personalizzazione della sicurezza

Se si desidera modificare il modo in cui opera la sicurezza IBM MQ , è necessario farlo tramite l'uscita SAF (ICHRFR00) o tramite le uscite nel gestore della sicurezza esterno.

Per ulteriori informazioni sulle uscite RACF , consultare il manuale *z/OS Security Server RACROUTE Macro Reference* .

**Nota:** Poiché IBM MQ ottimizza le chiamate a ESM, è possibile che le richieste RACROUTE non vengano effettuate, ad esempio, su ogni apertura per una particolare coda da parte di un determinato utente.

## Messaggi di violazione della sicurezza su z/OS

Una violazione della sicurezza viene indicata dal codice di ritorno MQRN\_NOT\_AUTHORIZED in un programma di applicazione o da un messaggio nel log del lavoro.

Un codice di ritorno di MQRN\_NOT\_AUTHORIZED può essere restituito a un'applicazione per i motivi riportati di seguito:

- Un utente non può connettersi al gestore code. In questo caso, si riceve un messaggio ICH408I nel log dei lavori Batch / TSO, CICS o IMS .
- Un collegamento dell'utente al gestore code non è riuscito perché, ad esempio, l'ID utente del lavoro non è valido o appropriato oppure l'ID utente dell'attività o l'ID utente alternativo non sono validi. Uno o più di questi ID utente potrebbero non essere validi perché sono stati revocati o eliminati. In questo caso, si riceve un messaggio ICHxxxx e possibilmente un messaggio IRRxxxx nel log del lavoro del gestore code che indica il motivo dell'errore di collegamento. Ad esempio:

```
ICH408I USER(NOTDFND ) GROUP(          ) NAME(???)
LOGON/JOB INITIATION - USER AT TERMINAL          NOT RACF-DEFINED
IRR012I VERIFICATION FAILED. USER PROFILE NOT FOUND
```

- È stato richiesto un utente alternativo, ma l'ID utente del lavoro o dell'attività non dispone dell'accesso all'ID utente alternativo. Per questo errore, si riceve un messaggio di violazione nel log del lavoro del gestore code pertinente.
- Un'opzione di contesto è stata utilizzata o è implicita aprendo una coda di trasmissione per l'emissione, ma l'ID utente del lavoro o, se applicabile, l'ID utente alternativo o l'attività non ha accesso all'opzione di contesto. In questo caso, viene inserito un messaggio di violazione nel registro lavori del gestore code pertinente.
- Un utente non autorizzato ha tentato di accedere a un oggetto gestore code protetto, ad esempio una coda. In questo caso, un messaggio ICH408I per la violazione viene inserito nel log lavori del gestore code pertinente. Questa violazione potrebbe essere dovuta al lavoro o, quando applicabile, all'attività o all'ID utente alternativo.

I messaggi di violazione per la sicurezza dei comandi e per la sicurezza delle risorse dei comandi possono essere trovati anche nel log lavori del gestore code.

Se il messaggio di violazione ICH408I mostra il nome lavoro del gestore code piuttosto che un ID utente, questo è di solito il risultato di un ID utente alternativo vuoto specificato. Ad esempio:

```
ICH408I JOB(MQS1MSTR) STEP(MQS1MSTR)
      MQS1.PAYROLL.REQUEST CL(MQQUEUE)
      INSUFFICIENT ACCESS AUTHORITY
      ACCESS INTENT(UPDATE ) ACCESS ALLOWED(NONE )
```

È possibile scoprire a chi è consentito utilizzare ID utente alternativi vuoti controllando l'elenco di accessi del profilo MQADMIN hlq.ALTERNATE.USER.-BLANK-.

Un messaggio di violazione ICH408I può essere generato anche da:

- Un comando inviato alla coda di input del sistema - comando senza contesto. I programmi scritti dall'utente che scrivono nella coda di input del comando di sistema devono sempre utilizzare un'opzione di contesto. Per ulteriori informazioni, consultare [“Profili per la sicurezza del contesto”](#) a pagina 209.
- Quando il lavoro che accede alla risorsa IBM MQ non dispone di un ID utente associato o quando un adattatore IBM MQ non può estrarre l'ID utente dall'ambiente dell'adattatore.

I messaggi di violazione potrebbero essere emessi anche se si sta utilizzando sia il gruppo di condivisione code che la sicurezza a livello di gestore code. È possibile che vengano visualizzati messaggi che indicano che non è stato trovato alcun profilo a livello di gestore code, ma a cui è ancora concesso l'accesso a causa di un profilo a livello di gruppo di condivisione code.

```
ICH408I JOB(MQS1MSTR) STEP(MQS1MSTR)
      MQS1.PAYROLL.REQUEST CL(MQQUEUE)
      PROFILE NOT FOUND - REQUIRED FOR AUTHORITY CHECKING
      ACCESS INTENT(UPDATE ) ACCESS ALLOWED(NONE )
```

## **Cosa fare se l'accesso è consentito o non consentito in modo non corretto**

Oltre ai passi descritti in *z/OS Security Server RACF Security Administrator's Guide*, utilizzare questo elenco di controllo se l'accesso a una risorsa sembra essere controllato in modo non corretto.

- I profili switch sono impostati correttamente?
  - RACF è attivo?
  - Le classi IBM MQ RACF sono installate e attive?  
Utilizzare il comando RACF , SETROPTS LIST, per verificarlo.
  - Utilizzare il comando IBM MQ DISPLAY SECURITY per visualizzare lo stato dello switch corrente dal gestore code.
  - Controllare i profili switch nella classe MQADMIN.  
Utilizzare i comandi RACF , SEARCH e RLIST, per questo.
  - Ricontrollare i profili di switch RACF immettendo il comando IBM MQ REFRESH SECURITY (MQADMIN).
- Il profilo della risorsa RACF è stato modificato? Ad esempio, l'accesso universale sul profilo è stato modificato o l'elenco di accesso del profilo è stato modificato?
  - Il profilo è generico?  
In caso affermativo, immettere il comando RACF , SETROPTS GENERIC (nomeclasse) AGGIORNA.
  - È stata aggiornata la sicurezza su questo gestore code?  
Se necessario, immettere il comando RACF SETROPTS RACLIST (classname) AGGIORNA.

- Se necessario, immettere il comando IBM MQ REFRESH SECURITY (\*).
- La definizione RACF dell'utente è stata modificata? Ad esempio, l'utente è stato connesso a un nuovo gruppo o l'autorità di accesso utente è stata revocata?
    - L'utente è stato ripristinato immettendo il comando IBM MQ RVERIFY SECURITY (userid)?
  - I controlli di sicurezza vengono ignorati a causa di RESLEVEL?
    - Controllare l'accesso dell'ID utente di connessione al profilo RESLEVEL. Utilizzare i record di verifica RACF per stabilire su cosa è impostato RESLEVEL.
    - Per i canali, tenere presente che il livello di accesso che l'ID utente dell'iniziatore di canali ha per RESLEVEL è ereditato da tutti i canali, quindi un livello di accesso, come ALTER, che fa sì che tutti i controlli vengano ignorati, fa sì che i controlli di sicurezza vengano ignorati per tutti i canali.
    - Se si è in esecuzione da CICS, verificare l'impostazione RESSEC della transazione.
    - Se RESLEVEL è stato modificato mentre un utente è connesso, è necessario disconnettersi e riconnettersi prima che la nuova impostazione RESLEVEL diventi effettiva.
  - Si stanno utilizzando i gruppi di condivisione code?
    - Se si sta utilizzando sia il gruppo di condivisione code che la sicurezza a livello di gestore code, verificare di aver definito tutti i profili corretti. Se il profilo del gestore code non è definito, viene inviato un messaggio al log che indica che il profilo non è stato trovato.
    - È stata utilizzata una combinazione di impostazioni switch che non è valida in modo che sia stato impostato il controllo di sicurezza completo?
    - È necessario definire gli switch di sicurezza per sovrascrivere alcune delle impostazioni del gruppo di condivisione code per il gestore code?
    - Un profilo a livello di gestore code ha la precedenza su un profilo a livello di gruppo di condivisione code?

## Considerazioni sulla protezione per l'iniziatore di canali su z/OS

Se si utilizza la sicurezza delle risorse in un ambiente di accodamento distribuito, lo spazio di indirizzo dell'iniziatore di canali necessita dell'accesso appropriato a varie risorse IBM MQ . È possibile utilizzare ICSF (Integrated Cryptographic Support Facility) per eseguire il seeding dell'algoritmo di protezione della password.

### Utilizzo della sicurezza delle risorse

Se si sta utilizzando la sicurezza delle risorse, considerare i seguenti punti se si sta utilizzando l'accodamento distribuito:

#### code di sistema

Lo spazio di indirizzo dell'iniziatore di canali richiede l'accesso RACF UPDATE alle code di sistema elencate in [“Sicurezza della coda di sistema” a pagina 198e](#) a tutte le code di destinazione utente e alla coda di messaggi non instradabili (ma consultare [“Sicurezza della coda di messaggi non recapitabili” a pagina 197](#) ).

#### Code di trasmissione

Lo spazio di indirizzo dell'iniziatore di canali richiede l'accesso ALTER a tutte le code di trasmissione utente.

#### sicurezza del contesto

L'ID utente del canale (e l'ID utente MCA, se ne è stato specificato uno) necessita dell'accesso RACF CONTROL ai profili hlq.CONTEXT.queueName nella classe MQADMIN. A seconda del profilo RESLEVEL, l'ID utente del canale potrebbe richiedere l'accesso CONTROL a questi profili.

Tutti i canali richiedono l'accesso CONTROL a MQADMIN hlq.CONTEXT. profilo DLQ (dead - letter - queue). Tutti i canali (sia in fase di avvio che di risposta) possono generare report e, di conseguenza, necessitano dell'accesso CONTROL al profilo hlq.CONTEXT.reply-q .

I canali SENDER, CLUSSDR e SERVER necessitano dell'accesso CONTROL ai profili hlq.CONTEXT.xmit-queue-name poiché i messaggi possono essere inseriti nella coda di trasmissione per attivare il canale in modo da terminare correttamente.

**Nota:** Se l'ID utente del canale, o un gruppo RACF a cui è collegato l'ID utente del canale, ha accesso CONTROL o ALTER a hlq.RESLEVEL, non ci sono controlli delle risorse per l'iniziatore del canale o per uno dei suoi canali.

Per ulteriori informazioni, consultare [“Profili per la sicurezza del contesto” a pagina 209](#) [“RESLEVEL e la connessione dell'iniziatore di canali” a pagina 229](#) e [“ID utente per il controllo di sicurezza su z/OS” a pagina 231](#).

## CSQINPX

Se si sta utilizzando il dataset di input CSQINPX, l'iniziatore di canali ha anche bisogno dell'accesso READ a CSQINPX e dell'accesso UPDATE al dataset CSQOUTX e alle code dinamiche SYSTEM.CSQXCMD. \*.

## Sicurezza connessione

Le richieste di connessione dello spazio di indirizzo dell'iniziatore di canali utilizzano un tipo di connessione CHIN, per cui è necessario impostare la sicurezza di accesso appropriata, consultare [“Profili di sicurezza di connessione per l'iniziatore di canali” a pagina 191](#).

## I data set

Lo spazio di indirizzo dell'iniziatore di canali necessita di un accesso appropriato ai dataset del gestore code, consultare [“Autorizzazione dell'accesso ai dataset” a pagina 247](#).

## Comandi

I comandi di accodamento distribuiti (ad esempio, DEFINE CHANNEL, START CHINIT, START LISTENER e altri comandi del canale) devono avere la serie di sicurezza del comando appropriata, consultare [Tabella 47 a pagina 212](#).

Se si sta utilizzando un gruppo di condivisione code, l'iniziatore di canali potrebbe emettere diversi comandi internamente, quindi l'ID utente utilizzato deve essere autorizzato a immettere tali comandi. Questi comandi sono START e STOP CHANNEL per ogni canale utilizzato con CHLDISP (SHARED).

Se il PSMODE del gestore code non è DISABLED, l'iniziatore del canale deve disporre dell'accesso READ al comando DISPLAY PUBSUB.

## Sicurezza canale

I canali, in particolare i ricevitori e le connessioni server, devono essere configurati con la sicurezza appropriata; per ulteriori informazioni, consultare [“ID utente per il controllo di sicurezza su z/OS” a pagina 231](#).

Puoi anche utilizzare il protocollo TLS (Transport Layer Security) per fornire la sicurezza sui canali. Consultare [“Protocolli di sicurezza TLS in IBM MQ” a pagina 24](#) per ulteriori informazioni sull'utilizzo di TLS con IBM MQ.

Consultare anche [“Controllo accessi per client” a pagina 88](#) per informazioni sulla sicurezza della connessione server.

## ID utente

Gli ID utente descritti in [“ID utente utilizzati dall'iniziatore di canali” a pagina 234](#) e [“ID utente utilizzati dall'agent di accodamento all'interno del gruppo” a pagina 238](#) hanno bisogno del seguente accesso:

- RACF Accesso UPDATE alle code di destinazione appropriate e alla coda di messaggi non instradabili
- Accesso RACF CONTROL al profilo hlq.CONTEXT.queueName se la verifica del contesto viene eseguita sul destinatario
- Accesso appropriato a hlq.ALTERNATE.USER.userid profili che potrebbe essere necessario utilizzare.
- Per i client, l'accesso RACF appropriato alle risorse da utilizzare.

## Sicurezza APPC

Impostare la sicurezza APPC appropriata se si sta utilizzando il protocollo di trasmissione LU 6.2 . Ad esempio, utilizzare la classe RACF APPCLU. Per informazioni sull'impostazione della sicurezza per APPC, consultare i seguenti manuali:

- *z/OS V1R2.0 MVS Planning: APPC Management*
- *Multiplatform APPC Configuration Guide*, una pubblicazione IBM Redbooks

Le trasmissioni in uscita utilizzano l'opzione APPC "SECURITY (SAME)". Come risultato, l'ID utente dello spazio di indirizzo dell'iniziatore di canali e il relativo profilo predefinito ( RACF GROUP) vengono trasmessi attraverso la rete al ricevitore con un indicatore che l'ID utente è già stato verificato (ALREADYV).

Se anche il lato ricevente è z/OS, l'ID utente e profilo vengono verificati da APPC e l'ID utente viene presentato al canale ricevente e utilizzato come ID utente del canale.

In un ambiente in cui il gestore code utilizza APPC per comunicare con un altro gestore code sullo stesso o su un altro sistema z/OS , è necessario verificare che:

- La definizione VTAM per la LU di comunicazione specifica SETACPT (ALREADYV)
- Esiste un profilo RACF APPCLU per la connessione tra le LU che specifica CONVSEC (ALREADYV)

## Modifica delle impostazioni di sicurezza

Se viene modificato il livello di accesso RACF che l'ID utente del canale o l'ID utente MCA ha per una coda di destinazione, questa modifica diventa effettiva solo per i nuovi handle di oggetto (ossia, nuovi MQOPEN ) per la coda di destinazione. I tempi in cui gli MCA aprono e chiudono le code sono variabili; se un canale è già in esecuzione quando viene apportata tale modifica di accesso, l'MCA può continuare a inserire i messaggi nella coda di destinazione utilizzando l'accesso di sicurezza esistente degli ID utente piuttosto che l'accesso di sicurezza aggiornato. L'arresto e il riavvio dei canali per applicare il livello di accesso aggiornato evita questo scenario.

## Riavvio automatico

Se si sta utilizzando z/OS ARM (Automatic Restart Manager) per riavviare l'iniziatore del canale, l'ID utente associato allo spazio di indirizzo XCFAS deve essere autorizzato a immettere il comando IBM MQ START CHINIT.

## Utilizzo di ICSF (Integrated Cryptographic Service Facility)

L'iniziatore di canali può utilizzare ICSF per generare un numero casuale quando si esegue il seeding dell'algoritmo di protezione della password per nascondere le password che fluiscono sui canali client se TLS non viene utilizzato. Il processo di generazione di un numero casuale è denominato *entropia*.

Se la funzione z/OS è installata ma non è stato avviato ICSF, viene visualizzato il messaggio [CSQX213E](#) e l'iniziatore del canale utilizza STCK per l'entropia.

Il messaggio CSQX213E avvisa che l'algoritmo di protezione della password non è sicuro come potrebbe essere. Tuttavia, è possibile continuare il processo; non vi è altro impatto sul runtime.

Se la funzione z/OS non è installata, l'iniziatore di canali utilizza automaticamente STCK.

### Note:

1. L'utilizzo di ICSF per l'entropia genera più sequenze casuali rispetto all'utilizzo di STCK.
2. Se si avvia ICSF, è necessario riavviare l'iniziatore di canali.
3. ICSF è richiesto per alcuni CipherSpecs. Se si tenta di utilizzare uno di questi CipherSpecs e non si dispone di ICSF installato, si riceve il messaggio [CSQX629E](#).

## Sicurezza nei cluster del gestore code su z/OS

Le considerazioni sulla protezione per i cluster sono le stesse per i gestori code e i canali non raggruppati in cluster. L'iniziatore di canali ha bisogno di accedere ad alcune code di sistema aggiuntive e alcuni comandi aggiuntivi necessitano di una serie di sicurezza appropriata.



È possibile utilizzare l'ID utente MCA, i record di autenticazione di canale, TLS e le uscite di sicurezza per autenticare i canali cluster (come con i canali convenzionali). I record di autenticazione del canale o l'uscita di sicurezza relativi al canale ricevente del cluster devono verificare che al gestore code remoto sia consentito l'accesso alle code del cluster del gestore code del server. È possibile iniziare a utilizzare il supporto cluster IBM MQ senza modificare la sicurezza dell'accesso alla coda esistente. Tuttavia, è necessario consentire ad altri gestori code nel cluster di scrivere nel SISTEMA SYSTEM.CLUSTER.COMMAND.QUEUE se devono unirsi al cluster.

Il supporto cluster IBM MQ non fornisce un meccanismo per limitare un membro di un cluster solo al ruolo client. Di conseguenza, è necessario assicurarsi di considerare attendibili tutti i gestori code consentiti nel cluster. Se un gestore code nel cluster crea una coda con un nome particolare, può ricevere messaggi per tale coda, indipendentemente dal fatto che l'applicazione che inserisce i messaggi in tale coda lo intenda o meno.

Per limitare l'appartenenza di un cluster, eseguire la stessa azione che si dovrebbe intraprendere per impedire ai gestori code di connettersi ai canali riceventi. Limitare l'appartenenza di un cluster utilizzando i record di autenticazione di canale o scrivendo un programma di uscita di sicurezza sul canale ricevente. È anche possibile scrivere un programma di uscita per impedire ai gestori code non autorizzati di scrivere sul sistema SYSTEM.CLUSTER.COMMAND.QUEUE.

**Nota:** Non è consigliabile consentire alle applicazioni di aprire SYSTEM.CLUSTER.TRANSMIT.QUEUE direttamente. Inoltre, non è consigliabile consentire a un'applicazione di aprire direttamente qualsiasi altra coda di trasmissione.

Se si utilizza la sicurezza delle risorse, considerare i seguenti punti in aggiunta alle considerazioni contenute in [“Considerazioni sulla protezione per l'iniziatore di canali su z/OS” a pagina 254:](#)

#### code di sistema

L'iniziatore di canali ha bisogno dell'accesso RACF ALTER alle code di sistema seguenti:

- SYSTEM.CLUSTER.COMMAND
- SYSTEM.CLUSTER.TRANSMIT.QUEUE.

e l'accesso UPDATE a SYSTEM.CLUSTER.REPOSITORY.QUEUE

È inoltre necessario l'accesso READ a tutti gli elenchi nomi utilizzati per il clustering.

#### Comandi

Impostare la sicurezza del comando appropriata (come descritto in [Tabella 47 a pagina 212](#)) per i comandi di supporto cluster (REFRESH e RESET CLUSTER, SUSPEND e RESUME QMGR).

## **Considerazioni sulla sicurezza per l'uso di IBM MQ con CICS**

Tutte le versioni CICS supportate da IBM MQ 9.0.0e successive, utilizzano la versione fornita da CICS dell'adattatore e del bridge.

Per i dettagli sulle considerazioni di sicurezza, consultare:

- [Sicurezza per l'adattatore CICS -IBM MQ.](#)
- [Sicurezza per il bridge CICS -IBM MQ.](#)

## **Considerazioni sulla sicurezza per l'uso di IBM MQ con IMS**

Utilizzare questo argomento per pianificare i requisiti di sicurezza quando si utilizza IBM MQ con IMS.

### **Utilizzo della classe OPERCMDS**

Se si utilizza RACF per proteggere le risorse nella classe OPERCMDS, assicurarsi che l'id utente associato allo spazio di indirizzo del gestore code IBM MQ disponga dell'autorizzazione per emettere il comando MODIFY su qualsiasi sistema IMS a cui può connettersi.

## Considerazioni sulla sicurezza per il bridge di IMS

Esistono quattro aspetti da considerare quando si decidono i requisiti di sicurezza per il bridge IMS :

- Quale autorizzazione di protezione è necessaria per collegare IBM MQ a IMS
- La quantità di controllo di sicurezza eseguita sulle applicazioni che utilizzano il bridge per accedere IMS
- Quali risorse IMS queste applicazioni possono utilizzare
- Quale autorizzazione deve essere utilizzata per i messaggi immessi e ricevuti dal bridge

Quando si definiscono i requisiti di sicurezza per il bridge IMS , è necessario considerare quanto segue:

- I messaggi che passano attraverso il bridge potrebbero essere stati originati da applicazioni su piattaforme che non offrono funzioni di sicurezza avanzate
- I messaggi che passano attraverso il bridge potrebbero essere stati originati da applicazioni che non sono controllate dalla stessa azienda o organizzazione

### **Considerazioni sulla sicurezza per la connessione a IMS**

Concedere all'ID utente dello spazio di indirizzo del gestore code IBM MQ l'accesso al gruppo OTMA.

Il bridge IMS è un client OTMA. La connessione a IMS opera con l'ID utente dello spazio di indirizzo del gestore code IBM MQ . Questo è normalmente definito come membro del gruppo di attività avviato. Questo ID utente deve avere accesso al gruppo OTMA (a meno che l'impostazione /SECURE OTMA non sia NONE).

A tale scopo, definire il profilo seguente nella classe FACILITY:

```
IMSXCF.xcfgrname.mqxcfmname
```

Dove xcfgrname è il nome del gruppo XCF e mqxcfmname è il nome del membro XCF di IBM MQ.

È necessario fornire all'ID utente del gestore code IBM MQ l'accesso in lettura a questo profilo.

#### **Nota:**

1. Se si modificano le autorizzazioni nella classe FACILITY, è necessario immettere il comando RACF SETROPTS RACLIST (FACILITY) REFRESH per attivare le modifiche.
2. Se il profilo hlq.NO.SUBSYS.SECURITY esiste nella classe MQADMIN, non viene passato alcun ID utente a IMS e la connessione ha esito negativo a meno che l'impostazione /SECURE OTMA non sia NONE.

### **Controllo accessi applicazione per il bridge IMS**

Definire un profilo RACF nella classe FACILITY per ogni sistema IMS . Concedere un appropriato livello di accesso all'ID utente del gestore code IBM MQ .

Per ogni IMS sistema a cui si connette il bridge IMS , è possibile definire il seguente profilo RACF nella classe FACILITY per determinare la quantità di controllo di sicurezza eseguita per ogni messaggio passato al sistema IMS .

```
IMSXCF.xcfgrname.imsxcfmname
```

Dove xcfgrname è il nome del gruppo XCF e imsxcfmname è il nome del membro XCF per IMS. (È necessario definire un profilo separato per ogni sistema IMS .)

Il livello di accesso consentito per l'ID utente del gestore code IBM MQ in questo profilo viene restituito a IBM MQ quando il bridge IMS si connette a IMS e indica il livello di sicurezza richiesto nelle transazioni successive. Per transazioni successive, IBM MQ richiede i servizi appropriati da RACF e, dove l'ID utente è autorizzato, trasmette il messaggio a IMS.

OTMA non supporta il comando IMS /SIGN; tuttavia, IBM MQ consente di impostare il controllo dell'accesso per ogni messaggio per abilitare l'implementazione del livello di controllo necessario.

Possono essere restituite le seguenti informazioni sul livello di accesso:

#### **NONE o NESSUN PROFILO TROVATO**

Questi valori indicano che è richiesta la massima sicurezza, ovvero, l'autenticazione è richiesta per ogni transazione. Viene eseguito un controllo per verificare che l'ID utente specificato nel campo *UserIdentifier* della struttura MQMD e la password o PassTicket nel campo *Authenticator* della struttura MQIIH siano noti a RACFe siano una combinazione valida. Un UTOKEN viene creato con una parola d'ordine o PassTickete passato a IMS ; UTOKEN non è memorizzato nella cache.

**Nota:** Se il profilo hlq.NO.SUBSYS.SECURITY esiste nella classe MQADMIN, questo livello di sicurezza sovrascrive qualsiasi elemento definito nel profilo.

#### **LETTURA**

Questo valore indica che la stessa autenticazione deve essere eseguita come per NONE nelle seguenti circostanze:

- La prima volta che viene rilevato un ID utente specifico
- Quando l'ID utente è stato rilevato in precedenza ma il UTOKEN memorizzato nella cache non è stato creato con una password o con PassTicket

IBM MQ richiede un UTOKEN se richiesto e lo passa a IMS.

**Nota:** Se è stata eseguita una richiesta per verificare nuovamente la sicurezza, tutte le informazioni memorizzate nella cache vengono perse e viene richiesto un UTOKEN la prima volta che viene rilevato un ID utente in un secondo momento.

#### **AGGIORNA**

Viene eseguito un controllo che l'ID utente nel campo *UserIdentifier* della struttura MQMD sia noto a RACF.

Un UTOKEN viene creato e passato a IMS ; UTOKEN è memorizzato nella cache.

#### **CONTROL/ALTER**

Questi valori indicano che non è necessario fornire UTOKEN di sicurezza per gli ID utente per questo sistema IMS . È possibile utilizzare questa opzione solo per sistemi di sviluppo e test.



**Attenzione:** Notare che l'ID utente contenuto nel campo *UserIdentifier* della struttura MQMD viene ancora passato per **CONTROL/ALTER**.

#### **Nota:**

1. Questo accesso viene definito quando IBM MQ si connette a IMSe dura per la durata della connessione. Per modificare il livello di sicurezza, è necessario modificare l'accesso al profilo di sicurezza e quindi arrestare e riavviare il bridge (ad esempio, arrestando e riavviando OTMA).
2. Se si modificano le autorizzazioni nella classe FACILITY, è necessario immettere il comando RACF SETROPTS RACLIST (FACILITY) REFRESH per attivare le modifiche.
3. È possibile utilizzare una password o un PassTicket, ma è necessario ricordare che il bridge IMS non codifica i dati. Per informazioni sull'utilizzo di PassTickets, consultare [“Utilizzo di RACF PassTickets nell'intestazione IMS”](#) a pagina 261.
4. Alcuni di questi risultati potrebbero essere influenzati dalle impostazioni di sicurezza in IMS, utilizzando il comando /SECURE OTMA.
5. Le informazioni UTOKEN memorizzate nella cache vengono conservate per la durata definita dai parametri INTERVAL e TIMEOUT del comando IBM MQ ALTER SECURITY.
6. L'opzione RACF WARNING non ha effetto sul profilo IMSXCF.xcfname.imsxcfname . Il suo utilizzo non influisce sul livello di accesso concesso e non viene prodotto alcun messaggio RACF WARNING.

### **z/OS Controllo di sicurezza su IMS**

I messaggi che passano attraverso il ponte contengono informazioni di sicurezza. I controlli di sicurezza effettuati dipendono dall'impostazione del comando IMS /SECURE OTMA.

Ogni messaggio IBM MQ che passa attraverso il bridge contiene le seguenti informazioni di sicurezza:

- Un ID utente contenuto nel campo *UserIdentifier* della struttura MQMD
- L'ambito di sicurezza contenuto nel campo *SecurityScope* della struttura MQIIH (se la struttura MQIIH è presente)
- Un UTOKEN (a meno che il sottosistema IBM MQ non abbia accesso CONTROL o ALTER al profilo IMSXCF.xcfcname.imsxcfmname pertinente)

I controlli di sicurezza effettuati dipendono dall'impostazione del comando IMS /SECURE OTMA, come segue:

#### **/PROTEZIONE OTMA NESSUNO**

Non vengono effettuati controlli di sicurezza per la transazione.

#### **/CONTROLLO OTMA SICURO**

Il campo *UserIdentifier* della struttura MQMD viene passato a IMS per il controllo dell'autorità di transazione o comando.

Un ACEE (Accessor Environment Element) viene creato nella control region IMS .

#### **/PROTEZIONE OTMA PIENA**

Il campo *UserIdentifier* della struttura MQMD viene passato a IMS per il controllo dell'autorità di transazione o comando.

Un ACEE viene creato nella region dipendente IMS e nella region di controllo IMS .

#### **/PROFILO OTMA SICURO**

Il campo *UserIdentifier* della struttura MQMD viene inoltrato a IMS per il controllo dell'autorizzazione della transazione o del comando

Il campo *SecurityScope* nella struttura MQIIH viene utilizzato per stabilire se creare un ACEE nella regione dipendente IMS e nella regione di controllo.

#### **Nota:**

1. Se si modificano le autorizzazioni nella classe TIMS o CIMS o nelle classi di gruppo GIMS o DIMSassociate, è necessario immettere il seguente comando IMS per attivare le modifiche:
  - /MODIFY PREPARARE RACF
  - /MODIFICA COMMIT
2. Se non si utilizza /SECURE OTMA PROFILE, qualsiasi valore specificato nel campo *SecurityScope* della struttura MQIIH viene ignorato.

### **Controllo di sicurezza eseguito dal bridge IMS**

Vengono utilizzate autorizzazioni differenti a seconda dell'azione eseguita.

Quando il bridge inserisce o riceve un messaggio, vengono utilizzate le seguenti autorizzazioni:

#### **Richiamo di un messaggio dalla coda bridge**

Non viene eseguita alcuna verifica di sicurezza.

#### **Inserimento di un'eccezione o di un messaggio di report COA**

Utilizza l'autorizzazione dell'ID utente nel campo *UserIdentifier* della struttura MQMD.

#### **Inserimento di un messaggio di risposta**

Utilizza l'autorità dell'ID utente nel campo *UserIdentifier* della struttura MQMD del messaggio originale

#### **Inserimento di un messaggio nella coda di messaggi non recapitabili**

Non viene eseguita alcuna verifica di sicurezza.

#### **Nota:**

1. Se si modificano i profili della classe IBM MQ , è necessario immettere il comando IBM MQ REFRESH SECURITY (\*) per attivare le modifiche.

2. Se si modifica l'autorizzazione di un utente, è necessario immettere il comando MQSC RVERIFY SECURITY per attivare la modifica.

## **z/OS** **Utilizzo di RACF PassTickets nell'intestazione IMS**

Puoi utilizzare un PassTicket al posto di una password nell'intestazione IMS .

Se si desidera utilizzare un PassTicket anziché una password nell'intestazione IMS (MQIIH), specificare il nome dell'applicazione rispetto al quale il PassTicket viene convalidato nell'attributo PASSTKTA della definizione STGCLASS della coda bridge IMS a cui deve essere instradato il messaggio.

Se il valore PASSTKTA viene lasciato vuoto, è necessario creare un PassTicket . Il nome dell'applicazione in questo caso deve essere nel formato MVSxxxx, dove xxxx è l'SMFID del sistema z/OS su cui viene eseguito il gestore code di destinazione.

Un PassTicket viene creato da un ID utente, il nome dell'applicazione di destinazione e una chiave segreta. È un valore a 8 byte contenente caratteri alfabetici e numerici maiuscoli. Può essere utilizzato una sola volta ed è valido per un periodo di 20 minuti. Se un PassTicket viene generato da un RACF sistema locale, RACF controlla solo che il profilo esista e non che l'utente disponga dell'autorizzazione per il profilo. Se il PassTicket è stato generato su un sistema remoto, RACF convalida l'accesso dell>ID utente al profilo. Per informazioni complete su PassTickets, consultare *z/OS SecureWay Security Server RACF Security Administrator's Guide*.

I PassTickets nelle intestazioni IMS vengono forniti a RACF da IBM MQ, non IMS.

## **Impostazione della sicurezza IBM MQ MQI client**

È necessario considerare la sicurezza IBM MQ MQI client , 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.

Da IBM MQ 8.0, l'autenticazione è fornita da due opzioni:

- La funzione di autenticazione della connessione.

Per ulteriori informazioni sull'autenticazione della connessione, consultare [“Autenticazione connessione”](#) a pagina 59.

- Utilizzo dell'autenticazione reciproca in TLS.

Per ulteriori informazioni su TLS, consultare [“Utilizzo di SSL/TLS”](#) a pagina 266.

- 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 la verifica dei diritti di controllo accessi.

Per ulteriori informazioni sul controllo accessi, consultare [“Autorizzazione dell'accesso agli oggetti”](#) a pagina 340.

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 client a un ID utente MCAUSER”](#) a pagina 376.

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 47.

## **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 CipherSpecs certificati 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 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 TLS attive e un valore FipsRequired è specificato in modo valido su un MQCONNX SSL, tutte le connessioni TLS successive associate a questo processo devono utilizzare solo i CipherSpecs associati a questo valore. Ciò si applica fino a quando questa e tutte le altre connessioni TLS non vengono arrestate, a questo punto un successivo MQCONNX può fornire un nuovo valore per FipsRequired.

Se l'hardware crittografico è presente, i moduli crittografici utilizzati da IBM MQ possono essere configurati in modo da essere quei moduli forniti dal prodotto hardware e potrebbero essere certificati FIPS a un determinato 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. IBM MQ non garantisce di rifiutare tutte queste connessioni ed è responsabilità dell'utente determinare se la configurazione di IBM MQ è conforme a FIPS.

### **Concetti correlati**

[“FIPS \(Federal Information Processing Standards\) per UNIX, Linux, and Windows”](#) a pagina 33

Quando la crittografia è richiesta su un canale SSL/TLS su sistemi Windows, UNIX and Linux, IBM MQ utilizza un pacchetto di crittografia denominato ICC (IBM Crypto for C). Sulle piattaforme Windows, UNIX and Linux, il software ICC ha superato il programma di convalida crittografica FIPS (Federal Information Processing Standards) del National Institute of Standards and Technology degli Stati Uniti, al livello 140-2.

### **Informazioni correlate**

[FipsRequired \(MQLONG\)](#)

[FIPS MQSSL](#)

[Stanza SSL del file di configurazione client](#)

## 8.0 su AIX

Le applicazioni client TLS su AIX potrebbero riscontrare MQRC\_CHANNEL\_CONFIG\_ERROR ed errore AMQ6175 durante l'esecuzione su sistemi AIX con più installazioni GSKit 8.0 .

Quando si eseguono applicazioni client su un sistema AIX con più installazioni GSKit 8.0 , le chiamate di connessione client possono restituire MQRC\_CHANNEL\_CONFIG\_ERROR quando si utilizza 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__9GSKASN0ID (number 16) is not
exported from dependent module /db2data/db2inst1/sqllib/lib64/libgsk8cms_64.so.
Symbol VALUE_EC_NamedCurve_secp384r1__9GSKASN0ID (number 17) is not exported from
dependent module /db2data/db2inst1/sqllib/lib64/libgsk8cms_64.so.
Symbol VALUE_EC_NamedCurve_secp521r1__9GSKASN0ID (number 18) is not exported from
dependent module /db2data/db2inst1/sqllib/lib64/libgsk8cms_64.so.
Symbol VALUE_EC_ecPublicKey__9GSKASN0ID (number 19) is not exported from dependent
module /db2data/db2inst1/sqllib/lib64/libgsk8cms_64.so.
Symbol VALUE_EC_ecdsa_with_SHA1__9GSKASN0ID (number 20) is not exported from
dependent module /db2data/db2inst1/sqllib/lib64/libgsk8cms_64.so.
Symbol VALUE_EC_ecdsa__9GSKASN0ID (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.

```
----- amqcgkska.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 MQ di una serie mista di librerie da due diverse installazioni di GSKit 8.0 . L'esecuzione di un'applicazione client IBM MQ in un ambiente Db2 può causare questo errore.

Per evitare questo errore, includere le directory della libreria IBM MQ all'inizio del percorso della libreria in modo che le librerie IBM 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 \(set IBM MQ environment\)](#)

## Impostazione delle comunicazioni per SSL o TLS su IBM i

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 alcuni sistemi operativi, è possibile eseguire i test con certificati autofirmati. Tuttavia, su IBM i, è necessario utilizzare i certificati personali firmati da una CA locale.

Per informazioni complete sulla creazione e la gestione dei certificati, consultare [“Utilizzo di SSL/TLS in IBM i”](#) a pagina 266.

Questa raccolta di argomenti introduce alcune delle attività coinvolte nella configurazione delle comunicazioni SSL o TLS e fornisce una guida dettagliata per il completamento di tali attività

Potresti anche voler verificare l'autenticazione client SSL o TLS, che sono parti opzionali dei protocolli SSL e TLS. Durante l'handshake SSL o TLS, il client SSL o TLS ottiene e convalida sempre un certificato digitale dal server. Con l'implementazione IBM MQ, il server SSL o TLS richiede sempre un certificato dal client.

Su IBM i, il client SSL o TLS invia un certificato solo se ha un'etichetta nel formato IBM MQ corretto:

- Per un gestore code, `ibmwebspheremq` seguito dal nome del gestore code modificato in minuscolo. Ad esempio, per QM1, `ibmwebspheremqqm1`.
- Per un IBM MQ client C per IBM i, `ibmwebspheremq` seguito dall'ID utente di collegamento modificato in minuscolo, ad esempio `ibmwebspheremqmyuserid`.

IBM 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 SSL o TLS non invia un certificato, l'autenticazione ha esito negativo solo se la fine del canale che funge da server SSL o TLS è definita con il parametro `SSLCAUTH` impostato su `REQUIRED` o un valore del parametro `SSLPEER` impostato. Per ulteriori informazioni, fare riferimento a [Connessione di due gestori code utilizzando SSL o TLS](#).

## Impostazione delle comunicazioni per SSL o TLS su UNIX, Linux o

### 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.



**Attenzione:** Non è possibile utilizzare una combinazione di certificati firmati Elliptic Curve e certificati firmati RSA sui gestori code che si desidera unire utilizzando i canali abilitati TLS.

I gestori code che utilizzano i canali abilitati TLS devono utilizzare tutti i certificati firmati RSA o tutti i certificati firmati EC, non una combinazione di entrambi.

Per ulteriori informazioni, fare riferimento a [“Certificati digitali e compatibilità CipherSpec in IBM MQ”](#) a pagina 42.

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/TLS in UNIX, Linux, and Windows”](#) a pagina 278.

Questa raccolta di argomenti introduce alcune delle attività coinvolte nella configurazione delle comunicazioni SSL e fornisce istruzioni dettagliate sul completamento di queste attività.

Potresti 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 MQ, il server SSL o TLS richiede sempre un certificato dal client.

Su UNIX, Linux, and Windows, il client SSL o TLS invia un certificato solo se ha un'etichetta nel formato IBM 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 MQ, `ibmwebspheremq` seguito dall'ID utente di collegamento modificato in minuscolo, ad esempio `ibmwebspheremquserid`.

IBM 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, fare riferimento a [Connessione di due gestori code utilizzando SSL o TLS](#).

**z/OS**

## **Impostazione delle comunicazioni per SSL o TLS su z/OS**

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 z/OS è possibile eseguire i test con i certificati autofirmati o con i certificati personali firmati da una CA (Certificate Authority) locale.

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/TLS in z/OS”](#) a pagina 309.

Questa raccolta di argomenti introduce alcune delle attività coinvolte nell'impostazione delle comunicazioni SSL o TLS e fornisce istruzioni dettagliate 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 IBM MQ, il server SSL o TLS richiede sempre un certificato dal client.

Su z/OS, il client SSL o TLS invia un certificato solo se dispone di uno dei seguenti certificati:

- Solo per un canale condiviso, un certificato con un'etichetta nel formato `ibmWebSphereMQ` seguito dal nome del proprio gruppo di condivisione code, ad esempio `ibmWebSphereMQSG1`
- Un certificato con un'etichetta nel formato `ibmWebSphereMQ` seguito dal nome del tuo gestore code, ad esempio `ibmWebSphereMQQM1`
- Un certificato predefinito (che potrebbe essere il certificato `ibmWebSphereMQ`).

Se il canale è condiviso, tenta prima di trovare un certificato per il gruppo di condivisione code. Se non trova un certificato per un gruppo di condivisione code, prova a trovare un certificato per il gestore code.

In z/OS, IBM MQ utilizza il prefisso `ibmWebSphereMQ` su un'etichetta per evitare confusione con certificati per altri prodotti.

Il server SSL o TLS convalida sempre il certificato client, se ne viene inviato uno. Se il client SSL o TLS non invia un certificato, l'autenticazione ha esito negativo solo se la fine del canale che funge da server SSL o TLS è definita con il parametro `SSLCAUTH` impostato su `REQUIRED` o un valore del parametro `SSLPEER` impostato. Per ulteriori informazioni, fare riferimento a [Connessione di due gestori code utilizzando SSL o TLS](#).

## Utilizzo di SSL/TLS

Questi argomenti forniscono istruzioni per eseguire singole attività relative all'utilizzo di TLS con IBM MQ. Molti di essi vengono utilizzati come passi nelle attività di livello superiore descritte nelle seguenti sezioni:

- [“Identificazione e autenticazione degli utenti”](#) a pagina 321
- [“Autorizzazione dell'accesso agli oggetti”](#) a pagina 340
- [“Riservatezza dei messaggi”](#) a pagina 407
- [“Integrità dei dati dei messaggi”](#) a pagina 430
- [“Proteggere i cluster”](#) a pagina 432

### IBM i Utilizzo di SSL/TLS in IBM i

Questa raccolta di argomenti fornisce istruzioni per le singole attività che utilizzano TLS (Transport Layer Security) in IBM MQ for IBM i.

Per IBM i il supporto TLS è parte integrante del sistema operativo. Assicurarsi di aver installato i prerequisiti elencati in [Requisiti hardware e software su IBM i](#).

Su IBM i, si gestiscono le chiavi e certificati digitali con lo strumento DCM (Digital Certificate Manager).

#### **Accesso a DCM**

Seguire queste istruzioni per accedere all'interfaccia DCM.

#### **Informazioni su questa attività**

Effettuare le seguenti operazioni in un browser Web che supporti i frame.

#### **Procedura**

1. Andare a `http://machine.domain:2001` o `https://machine.domain:2010`, dove *macchina* è il nome del computer.
2. Immettere un profilo utente e una parola d'ordine validi quando richiesto.  
Verificare che il proprio profilo utente disponga delle autorizzazioni speciali `*ALLOBJ` e `*SECADM` per consentire la creazione di nuove memorizzazioni certificato. Se non si dispone delle autorizzazioni speciali, è possibile gestire solo i certificati personali o visualizzare le firme oggetto per gli oggetti per i quali si è autorizzati. Se si è autorizzati ad utilizzare un'applicazione di firma oggetto, è anche possibile firmare gli oggetti da DCM.
3. Nella pagina Configurazioni Internet, fare clic su **Gestore certificati Certificate Manager**.  
Viene visualizzata la pagina Certificate Manager digitale.

#### **Assegnazione di un certificato a un gestore code su IBM i**

Utilizzare DCM per assegnare un certificato a un gestore code.

Utilizzare la gestione certificati digitali IBM i tradizionale per assegnare un certificato a un gestore code. Ciò significa che è possibile specificare che un gestore code utilizza l'archivio certificati di sistema e che il gestore code è registrato per essere utilizzato come applicazione con Certificate Manager digitali. A tale scopo, modificare il valore dell'attributo **SSLKEYR** del gestore code in `*SYSTEM`.

Quando il parametro **SSLKEYR** viene modificato in \*SYSTEM, IBM MQ registra il gestore code come applicazione server con un'etichetta di applicazione univoca QIBM\_WEBSPPHERE\_MQ\_QMGRNAME e un'etichetta con una descrizione di Qmgrname (WMQ). Notare che gli attributi del canale **CERTLABL** non vengono utilizzati se si utilizza la memorizzazione certificato \*SYSTEM. Il gestore code viene quindi visualizzato come applicazione server in Digital Certificate Manager ed è possibile assegnare a questa applicazione qualsiasi certificato server o client nell'archivio di sistema.

Poiché il gestore code è registrato come un'applicazione, è possibile eseguire funzioni avanzate di DCM come la definizione di elenchi di attendibilità CA.

Se il parametro **SSLKEYR** viene modificato in un valore diverso da \*SYSTEM, IBM MQ annulla la registrazione del gestore code come applicazione con Digital Certificate Manager. Se un gestore code viene eliminato, viene annullata anche la registrazione da DCM. Un utente con sufficiente autorizzazione \*SECADM può anche aggiungere o rimuovere manualmente le applicazioni da DCM.

### **Configurazione di un repository delle chiavi su IBM i**

È necessario impostare un repository delle chiavi ad entrambe le estremità della connessione. È possibile utilizzare gli archivi certificati predefiniti oppure è possibile crearne di propri.

Una connessione TLS richiede un *repository chiavi* ad ogni estremità della connessione. Ogni gestore code e IBM MQ MQI client deve avere accesso a un repository delle chiavi. Se si desidera accedere al repository delle chiavi utilizzando un nome file e una parola d'ordine (ovvero, non utilizzando l'opzione \*SYSTEM) assicurarsi che il profilo utente QMQM disponga delle seguenti autorizzazioni:

- Autorizzazione di esecuzione per la directory contenente il repository delle chiavi
- Autorizzazione di lettura per il file contenente il repository chiavi

Per ulteriori informazioni, fare riferimento a [“Il repository delle chiavi SSL/TLS” a pagina 24](#). Si noti che gli attributi del canale **CERTLABL** non vengono utilizzati se si utilizza la memorizzazione certificato \*SYSTEM.

Su IBM i, i certificati digitali vengono memorizzati in un archivio certificati gestito con DCM. Questi certificati digitali hanno etichette che associano un certificato a un gestore code o a un IBM MQ MQI client. TLS utilizza i certificati per scopi di autenticazione.

L'etichetta è il valore dell'attributo **CERTLABL**, se è impostato, oppure il valore predefinito `ibmwebsphereMQ` con il nome del gestore code o l'ID di accesso dell'utente IBM MQ MQI client accodato, tutto in minuscolo. Per i dettagli, consultare [Etichetta certificato digitale](#).

Il nome del gestore code o dell'archivio certificati IBM MQ MQI client comprende un percorso e un nome di origine. Il percorso predefinito è `/QIBM/UserData/ICSS/Cert/Server/` e il nome della radice predefinito è `Default`. In IBM i, l'archivio certificati predefinito, `/QIBM/UserData/ICSS/Cert/Server/Default.kdb`, è noto anche come \*SYSTEM. Facoltativamente, è possibile definire il proprio percorso e il nome della radice.

Se si definisce il proprio percorso o nome file, impostare le autorizzazioni sul file per controllare strettamente l'accesso ad esso.

[“Modifica dell'ubicazione del repository delle chiavi per un gestore code su IBM i” a pagina 269](#) indica di specificare il nome dell'archivio certificati. È possibile specificare il nome dell'archivio certificati prima o dopo la creazione dell'archivio certificati.

**Nota:** Le operazioni che è possibile eseguire con DCM potrebbero essere limitate dall'autorizzazione del proprio profilo utente. Ad esempio, è necessario disporre delle autorizzazioni \*ALLOBJ e \*SECADM per creare un certificato CA.

#### *Creazione di un archivio certificati su IBM i*

Se non si desidera utilizzare la memorizzazione certificato predefinita, seguire questa procedura per crearne una propria.

### **Informazioni su questa attività**

Creare un nuovo archivio certificati solo se non si desidera utilizzare l'archivio certificati predefinito IBM i.

Per specificare che deve essere utilizzata la memorizzazione certificato del sistema IBM i , modificare il valore dell'attributo SSLKEYR del gestore code in \*SYSTEM. Questo valore indica che il gestore code utilizza l'archivio certificati di sistema e che il gestore code è registrato per essere utilizzato come applicazione con DCM (Digital Certificate Manager ).

## Procedura

1. Accedere all'interfaccia DCM, come descritto in [“Accesso a DCM” a pagina 266](#)
2. Nel pannello di navigazione, fare clic su **Crea nuovo archivio certificati**.  
La pagina Crea nuovo archivio certificati viene visualizzata nel frame delle attività.
3. Nel frame di attività, selezionare **Altro archivio certificati di sistema** e fare clic su **Continua**.  
La pagina Crea un certificato nella nuova memorizzazione certificato viene visualizzata nel frame di attività.
4. Selezionare **No - Non creare un certificato nella memorizzazione certificato** e fare clic su **Continua**.  
La pagina Nome archivio certificati e password viene visualizzata nel frame delle attività.
5. Nel campo **Nome file e percorso archivio certificati** , immettere un percorso IFS e un nome file, ad esempio /QIBM/UserData/mqm/qmgrs/qm1/key.kdb
6. Immettere una password nel campo **Password** e immetterla nuovamente nel campo **Conferma password** . Fare clic su **Continua**.  
Prendere nota della password (che è sensibile al maiuscolo / minuscolo) perché è necessaria quando si esegue lo stash della chiave del repository.
7. Per uscire da DCM, chiudere la finestra del browser.

## Operazioni successive

Una volta creato l'archivio certificati utilizzando DCM, accertarsi di memorizzare la password, come descritto in [“Stash della parola d'ordine dell'archivio certificati sui sistemi IBM i” a pagina 268](#)

### Attività correlate

[“Importazione di un certificato in un repository delle chiavi su IBM i” a pagina 274](#)

Seguire questa procedura per importare un certificato.

*Stash della parola d'ordine dell'archivio certificati sui sistemi IBM i*

Memorizzare la parola d'ordine della memorizzazione certificato utilizzando i comandi CL.

Le seguenti istruzioni si applicano allo stash della password dell'archivio certificati su IBM i per un gestore code. In alternativa, per un IBM MQ MQI client, se non si utilizza la memorizzazione certificato \*SYSTEM (ossia, l'ambiente MQSSLKEYR è impostato su un valore diverso da \*SYSTEM), seguire la procedura descritta nella sezione [“Stash della password dell'archivio certificati” a pagina 277](#) di [“Programma di utilità IBM MQ SSL Client \(amqrssl\) per IBM i” a pagina 276](#).

Se è stato specificato che la memorizzazione certificato \*SYSTEM deve essere utilizzata (modificando il valore dell'attributo SSLKEYR del gestore code in \*SYSTEM) non è necessario seguire questa procedura.

Una volta creata la memorizzazione certificato utilizzando DCM, utilizzare i comandi riportati di seguito per memorizzare la password:

```
STRMQM MQMNAME('queue_manager_name')  
CHGMQM MQMNAME('queue_manager_name') SSLKEYRPWD('password')
```

La password è sensibile al maiuscolo/minuscolo. Deve essere immesso tra virgolette singole esattamente come è stato immesso al passo 6 di [“Creazione di un archivio certificati su IBM i” a pagina 267](#).

**Nota:** Se non si utilizza l'archivio certificati di sistema predefinito e non si conserva la parola d'ordine, i tentativi di avviare i canali TLS non riescono perché non possono ottenere la password richiesta per accedere all'archivio certificati.

## Individuazione del repository delle chiavi per un gestore code su IBM i

Utilizzare questa procedura per ottenere l'ubicazione dell'archivio certificati del gestore code.

## Procedura

1. Visualizzare gli attributi del gestore code utilizzando il seguente comando:

```
DSPMQM MQMNAME('queue manager name')
```

2. Esaminare l'output del comando per il percorso e il nome della radice dell'archivio certificati.  
Ad esempio: /QIBM/UserData/ICSS/Cert/Server/Default, dove /QIBM/UserData/ICSS/Cert/Server è il percorso e Default è il nome della radice.

### **Modifica dell'ubicazione del repository delle chiavi per un gestore code su IBM i**

Modificare l'ubicazione della memorizzazione certificato del gestore code utilizzando CHGMQM o ALTER QMGR.

## Procedura

Utilizzare il comando CHGMQM o il comando ALTER QMGR MQSC per impostare l'attributo del repository delle chiavi del gestore code.

- a) Utilizzo di CHGMQM: CHGMQM MQMNAME('qm1') SSLKEYR('/QIBM/UserData/ICSS/Cert/Server/MyKey')
- b) Utilizzo di ALTER QMGR: ALTER QMGR SSLKEYR('/QIBM/UserData/ICSS/Cert/Server/MyKey')

In entrambi i casi, l'archivio certificati ha il nome file completo: /QIBM/UserData/ICSS/Cert/Server/MyKey.kdb

## Operazioni successive

Quando si modifica l'ubicazione di un archivio certificati del gestore code, i certificati non vengono trasferiti dalla precedente ubicazione. Se i certificati CA preinstallati quando si crea l'archivio di certificati non sono sufficienti, è necessario popolare il nuovo archivio di certificati con i certificati, come descritto in [“Importazione di un certificato in un repository delle chiavi su IBM i”](#) a pagina 274. È inoltre necessario inserire la password per la nuova ubicazione, come descritto in [“Stash della parola d'ordine dell'archivio certificati sui sistemi IBM i”](#) a pagina 268.

### **Creazione di un'autorità di certificazione e di un certificato per il test su IBM i**

Utilizzare questa procedura per creare un certificato CA locale per firmare le richieste di certificato e per creare e installare il certificato CA.

## Prima di iniziare

Le istruzioni contenute in questo argomento presuppongono che non esista una CA (Certificate Authority) locale. Se una CA locale non esiste, andare a [“Richiesta di un certificato server su IBM i”](#) a pagina 270.

## Informazioni su questa attività

I certificati CA forniti quando si installa TLS sono firmati dalla CA emittente. Su IBM i, è possibile creare un'autorità di certificazione locale che può firmare i certificati server per verificare le comunicazioni TLS sul sistema. Attenersi alla seguente procedura in un browser Web per creare un certificato CA locale:

## Procedura

1. Accedere all'interfaccia DCM, come descritto in [“Accesso a DCM”](#) a pagina 266.
2. Nel pannello di navigazione, fare clic su **Crea una CA (Certificate Authority)**.  
La pagina Crea una CA (Certificate Authority) viene visualizzata nel frame delle attività.
3. Immettere una password nel campo **Password memorizzazione certificato** e immetterla nuovamente nel campo **Conferma password**.

4. Immettere un nome nel campo **Nome CA (Certificate Authority)** , ad esempio TLS Test Certificate Authority.
5. Immettere i valori appropriati nei campi **Nome comune** e **Organizzazione** e selezionare un paese. Per i restanti campi facoltativi, immettere i valori richiesti.
6. Immettere un periodo di validità per la AC locale nel campo **Periodo di validità** .  
Il valore predefinito è 1095 giorni.
7. Fare clic su **Continua**.  
La CA viene creata e DCM crea un archivio certificati e un certificato CA per l'AC locale.
8. Fare clic su **Installa certificato**.  
Viene visualizzata la finestra di dialogo Gestore download.
9. Immettere il nome percorso completo per il file temporaneo in cui si desidera memorizzare il certificato CA e fare clic su **Salva**.
10. Una volta completato il download, fare clic su **Apri**.  
Viene visualizzata la finestra Certificato.
11. Fare clic su **Installa certificato**.  
Viene visualizzata la procedura guidata Importazione certificato.
12. Fare clic su **Avanti**.
13. Selezionare **Seleziona automaticamente l'archivio certificati in base al tipo di certificato** e fare clic su **Avanti**.
14. Fare clic su **Fine**.  
Viene visualizzata una finestra di conferma.
15. Fare clic su **OK**.
16. Nella finestra Certificati, fare clic su **OK**.
17. Fare clic su **Continua**.  
La pagina Politica dell'autorità di certificazione viene visualizzata nel frame di attività.
18. Nel campo **Consenti creazione di certificati utente** , selezionare **Sì**.
19. Nel campo **Periodo di validità** , immettere il periodo di validità dei certificati emessi dalla CA locale.  
Il valore predefinito è 365 giorni.
20. Fare clic su **Continua**.  
La pagina Crea un certificato nella nuova memorizzazione certificato viene visualizzata nel frame di attività.
21. Verificare che nessuna delle applicazioni sia selezionata.
22. Fare clic su **Continua** per completare la configurazione della CA locale.

### ***Richiesta di un certificato server su IBM i***

I certificati digitali proteggono dall'impersonificazione, certificando che una chiave pubblica appartiene a un'entità specificata. Un nuovo certificato server può essere richiesto da un'autorità di certificazione utilizzando DCM (Digital Certificate Manager ).

### **Informazioni su questa attività**

Effettuare le seguenti operazioni in un browser Web:

#### **Procedura**

1. Accedere all'interfaccia DCM, come descritto in [“Accesso a DCM”](#) a pagina 266.
2. Nel pannello di navigazione, fare clic su **Seleziona un archivio certificati**.  
La pagina Seleziona una memorizzazione certificato viene visualizzata nel frame delle attività.
3. Selezionare la memorizzazione certificato che si desidera utilizzare e fare clic su **Continua**.

4. Opzionale: Se si seleziona **\*SYSTEM** nel passo 3, immettere la parola d'ordine dell'archivio di sistema e fare clic su **Continua**.
5. Opzionale: Se è stato selezionato **Altro archivio certificati di sistema** nel passo 3, nel campo **Percorso archivio certificati e nome file** , immettere il percorso IFS e il nome file impostati quando è stato creato l'archivio certificati. Immettere anche una password nel campo **Password archivio certificati** . Quindi fare clic su **Continua**
6. Nel pannello di navigazione, fare clic su **Crea certificato**.
7. Nel frame di attività, selezionare il pulsante di opzione **Certificato server o client** e fare clic su **Continua**.  
La pagina Seleziona una CA (Certificate Authority) viene visualizzata nel frame di attività.
8. Se si dispone di una CA locale sulla stazione di lavoro, scegliere la CA locale o una CA commerciale per firmare il certificato. Selezionare il pulsante di opzione per la CA desiderata e fare clic su **Continua**.  
La pagina Crea un certificato viene visualizzata nel frame di attività.
9. Opzionale: Per un gestore code, immettere l'etichetta del certificato nel campo **Etichetta certificato** .  
L'etichetta è il valore dell'attributo **CERTLABL** , se impostato, oppure il valore predefinito `ibmwebsphoremq` con il nome del gestore code accodato, tutto in minuscolo. Per i dettagli, consultare [Etichetta certificato digitale](#) .  
Ad esempio, per il gestore code QM1, immettere `ibmwebsphoremqmqm1` per utilizzare il valore predefinito.
10. Opzionale: Per un IBM MQ MQI client, nel campo **Etichetta del certificato** , immettere `ibmwebsphoremq` seguito dall'ID utente di collegamento ridotto in minuscolo.  
Ad esempio, digitare `ibmwebsphoremqmyuserid`
11. Immettere i valori appropriati nei campi **Nome comune** e **Organizzazione** e selezionare un paese.  
Per i restanti campi facoltativi, immettere i valori richiesti.

## Risultati

Se è stata selezionata una CA commerciale per firmare il certificato, DCM crea una richiesta di certificato in formato PEM (Privacy - Enhanced Mail). Inoltrare la richiesta alla CA scelta.

Se è stata selezionata la CA locale per firmare il certificato, DCM informa l'utente che il certificato è stato creato nella memorizzazione certificato e può essere utilizzato.

### ***Richiesta di un certificato server per IBM Key Manager su IBM i***

Attenersi a questa procedura per creare un certificato firmato dalla propria autorità di certificazione (CA) locale o per richiedere un certificato server firmato da una CA commerciale per l'importazione nel programma di utilità IBM Key Management (iKeyman).

## Informazioni su questa attività

Un certificato utente deve essere utilizzato quando DCM (Digital Certificate Manager ) funge da gestore certificati per IBM MQ su più piattaforme. Per certificati personali distribuiti ad altre piattaforme e per l'importazione nel programma di utilità iKeyman , effettuare le seguenti operazioni in un browser Web:

## Procedura

1. Accedere all'interfaccia DCM, come descritto in [“Accesso a DCM”](#) a pagina 266.
2. Nel riquadro di **navigazione** , fare clic su **Crea certificato**.  
La pagina **Crea certificato** viene visualizzata nel frame di attività.
3. Sul pannello **Crea certificato** , selezionare il pulsante d'opzione **Certificato utente** e fare clic su **Continua**.  
Viene visualizzata la pagina **Crea certificato utente** .

4. Nel riquadro **Crea certificato utente** , completare i campi richiesti in Informazioni sul certificato per **Nome organizzazione, Stato o provincia, Paese o regione**. Facoltativamente, inserire i valori nei campi **Unità organizzativa e Località o città** . Fare clic su **Continua**.  
Il **Nome comune** viene impostato automaticamente sull'ID utente con cui si è collegati al sistema iSeries .
5. Nel pannello successivo **Crea certificato utente** , fare clic su **Installa certificato** e fare clic su **Continua**.  
Viene visualizzato un messaggio che indica il certificato personale è stato installato. È necessario conservare una copia di riserva di questo certificato.
6. Fare clic su **OK**.
7. A seconda del browser internet utilizzato per accedere a DCM, effettuare le seguenti operazioni:
  - a) Per Internet Explorer scegliere: **Strumenti> Opzioni Internet> scheda Contenuto> pulsante Certificati> Scheda personale>**. Selezionare il certificato e fare clic su **Esporta**.
  - b) Per Mozilla Firefox scegliere: **Tools> Options> Advanced> Encryption tab> View Certificates button> Your Certificates tab>**. Selezionare il certificato e fare clic su **Backup**. Selezionare il nome file e il percorso e fare clic su **OK**.
8. Trasferire il certificato esportato al sistema remoto utilizzando FTP in formato binario.
9. Aggiungere il certificato esportato dal passo 7 al programma di utilità iKeyman nel database delle chiavi.
  - a) Se il certificato è stato salvato utilizzando Internet Explorer, utilizzare le istruzioni descritte in [Importazione da un file Microsoft . pfx](#) .
  - b) Se il certificato è stato salvato utilizzando Mozilla FireFox, utilizzare le istruzioni descritte in [Importazione di un certificato personale in un repository delle chiavi](#).

Durante l'importazione, assicurarsi che il nome etichetta del certificato personale e del certificato del firmatario siano modificati in quello previsto da IBM MQ . L'etichetta deve essere il valore dell'attributo IBM MQ **CERTLABL** , se impostato, oppure il valore predefinito `ibmwebspheremq` con il nome del gestore code accodato, tutto in minuscolo. Per i dettagli, consultare [Etichetta certificato digitale](#) .

### **Aggiunta di certificati server a un repository delle chiavi su IBM i**

Seguire questa procedura per aggiungere un certificato richiesto al repository delle chiavi.

### **Informazioni su questa attività**

Dopo che la CA ha inviato un nuovo certificato server, lo si aggiunge all'archivio certificati da cui è stata generata la richiesta. Se la CA invia il certificato come parte di un messaggio email, copiare il certificato in un file separato.

#### **Nota:**

- Non è necessario eseguire questa procedura se il certificato del server è firmato dalla CA locale.
- Prima di importare un certificato server in formato PKCS #12 in DCM, è necessario importare il certificato CA corrispondente.

Utilizzare la seguente procedura per ricevere un certificato del server nell'archivio certificati del gestore code:

### **Procedura**

1. Accedere all'interfaccia DCM, come descritto in [“Accesso a DCM”](#) a pagina 266.
2. Nella categoria di attività **Gestisci certificati** nel pannello di navigazione, fare clic su **Importa certificato**.  
La pagina Importa certificato viene visualizzata nel frame di attività.
3. Selezionare il pulsante di opzione per il proprio tipo di certificato e fare clic su **Continua**.



La pagina Importa certificato server o client o la pagina Importa certificato CA (Certificate Authority) viene visualizzata nel frame delle attività.

4. Nel campo **Importa file** , immettere il nome file del certificato che si desidera importare e fare clic su **Continua**.

DCM determina automaticamente il formato del file.

5. Se il certificato è un certificato **Server o client** , immettere la parola d'ordine nel frame delle attività e fare clic su **Continua**.

DCM informa che il certificato è stato importato.

### ***Esportazione di un certificato da un repository delle chiavi su IBM i***

L'esportazione di un certificato esporta sia la chiave pubblica che quella privata. Questa azione deve essere eseguita con estrema cautela, poiché la trasmissione di una chiave privata comprometterebbe completamente la tua sicurezza.

### **Prima di iniziare**

Quando condividi un certificato utente con un altro utente, scambia le chiavi pubbliche. Questo processo è descritto in **Attività 5. Condivisione dei certificati** nella [Guida rapida per AMS su UNIX](#). Quando si esporta un certificato come descritto qui, si esporta sia la chiave pubblica che la chiave privata. Questa azione deve essere eseguita con estrema cautela, poiché la trasmissione di una chiave privata comprometterebbe completamente la tua sicurezza.

### **Informazioni su questa attività**

Eeguire le seguenti operazioni sul computer da cui si desidera esportare il certificato:

### **Procedura**

1. Accedere all'interfaccia DCM, come descritto in "[Accesso a DCM](#)" a pagina 266.
2. Nel pannello di navigazione, fare clic su **Seleziona un archivio certificati**.  
La pagina Seleziona una memorizzazione certificato viene visualizzata nel frame delle attività.
3. Selezionare la memorizzazione certificato che si desidera utilizzare e fare clic su **Continua**.
4. Opzionale: Se si seleziona **\*SYSTEM** nel passo 3, immettere la parola d'ordine dell'archivio di sistema e fare clic su **Continua**.
5. Opzionale: Se è stato selezionato **Altro archivio certificati di sistema** al passo 3, nel campo **Percorso archivio certificati e nome file** , immettere il percorso IFS e il nome file impostati quando è stato creato l'archivio certificati e immettere una password nel campo **Password archivio certificati** . Quindi fare clic su **Continua**
6. Nella categoria di attività **Gestisci certificati** nel pannello di navigazione, fare clic su **Esporta certificato**.  
La pagina Esporta un certificato viene visualizzata nel frame di attività.
7. Selezionare il pulsante di opzione per il proprio tipo di certificato e fare clic su **Continua**.  
Nel frame di attività viene visualizzata la pagina Esporta server o Certificato client o la pagina Esporta certificato CA (Certificate Authority).
8. Selezionare il certificato che si desidera esportare.
9. Selezionare il pulsante di opzione per specificare se si desidera esportare il certificato in un file o direttamente in un altro archivio di certificati.
10. Se si è scelto di esportare un certificato server o client in un file, fornire le seguenti informazioni:
  - Il percorso e il nome file dell'ubicazione in cui si desidera memorizzare il certificato esportato.
  - Per un certificato personale, la password utilizzata per codificare il certificato esportato e la release di destinazione. Per i certificati AC, non è necessario specificare la parola d'ordine.
11. Se si è scelto di esportare un certificato direttamente in un altro archivio di certificati, specificare l'archivio di certificati di destinazione e la relativa password.

12. Fare clic su **Continua**.

### ***Importazione di un certificato in un repository delle chiavi su IBM i***

Seguire questa procedura per importare un certificato.

#### **Prima di iniziare**

Prima di importare un certificato personale in formato PKCS #12 in DCM, è necessario prima importare il certificato CA corrispondente.

#### **Informazioni su questa attività**

Eeguire questi passi sulla macchina in cui si desidera importare il certificato.

#### **Procedura**

1. Accedere all'interfaccia DCM, come descritto in [“Accesso a DCM” a pagina 266](#).
2. Nel pannello di navigazione, fare clic su **Seleziona un archivio certificati**.  
La pagina Seleziona una memorizzazione certificato viene visualizzata nel frame delle attività.
3. Selezionare la memorizzazione certificato che si desidera utilizzare e fare clic su **Continua**.
4. Opzionale: Se si seleziona **\*SYSTEM** nel passo 3, immettere la parola d'ordine dell'archivio di sistema e fare clic su **Continua**.
5. Opzionale: Se è stato selezionato **Altro archivio certificati di sistema** al passo 3, nel campo **Percorso archivio certificati e nome file**, immettere il percorso IFS e il nome file impostati quando è stato creato l'archivio certificati e immettere una password nel campo **Password archivio certificati**. Quindi fare clic su **Continua**.
6. Nella categoria di attività **Gestisci certificati** nel pannello di navigazione, fare clic su **Importa certificato**.  
La pagina Importa certificato viene visualizzata nel frame di attività.
7. Selezionare il pulsante di opzione per il proprio tipo di certificato e fare clic su **Continua**.  
Nel frame di attività viene visualizzata la pagina Importa certificato server o client o la pagina Importa certificato CA (Certificate Authority).
8. Nel campo **Importa file**, immettere il nome file del certificato che si desidera importare e fare clic su **Continua**.  
DCM determina automaticamente il formato del file.
9. Se il certificato è un certificato **Server o client**, immettere la parola d'ordine nel frame delle attività e fare clic su **Continua**. DCM informa che il certificato è stato importato.

### ***Rimozione dei certificati in IBM i***

Utilizzare questa procedura per rimuovere certificati personali.

#### **Procedura**

1. Accedere all'interfaccia DCM, come descritto in [“Accesso a DCM” a pagina 266](#).
2. Nel pannello di navigazione, fare clic su **Seleziona un archivio certificati**.  
La pagina Seleziona una memorizzazione certificato viene visualizzata nel frame delle attività.
3. Selezionare la casella di spunta **Altra memorizzazione certificato di sistema** e fare clic su **Continua**.  
Viene visualizzata la pagina Archivio certificati e password.
4. Nel campo **Percorso archivio certificati e nome file**, immettere il percorso IFS e il nome file impostati quando è stata creata l'archivio certificati.
5. Immettere una password nel campo **Password archivio certificati**. Fare clic su **Continua**.  
La pagina Archivio certificati corrente viene visualizzata nel frame di attività.
6. Nella categoria di attività **Gestisci certificati** nel pannello di navigazione, fare clic su **Elimina certificato**.

La pagina Conferma eliminazione certificato viene visualizzata nel frame di attività.

7. Selezionare il certificato che si desidera eliminare. Fare clic su **Elimina**.
8. Fare clic su **Sì** per confermare che si desidera eliminare il certificato. Altrimenti, fare clic su **No**. DCM informa l'utente se ha eliminato il certificato.

## **Utilizzo della memorizzazione certificato \*SYSTEM per l'autenticazione unidirezionale su IBM i**

Seguire queste istruzioni per configurare l'autenticazione unidirezionale.

### **Prima di iniziare**

- Creare un gestore code, canali e code di trasmissione.
- Creare un certificato server o client sul gestore code del server.
- Trasferire il certificato CA al gestore code client e importarlo nel contenitore chiavi.
- Avviare un listener sui gestori code server e client.

### **Informazioni su questa attività**

Per utilizzare l'autenticazione unidirezionale, utilizzando un computer che esegue IBM i come server TLS, impostare il parametro SSLKEYR (SSL Key Repository) su \*SYSTEM. Questa impostazione registra il gestore code IBM MQ come un'applicazione. È quindi possibile assegnare un certificato al gestore code per abilitare l'autenticazione unidirezionale.

È anche possibile utilizzare i keystore privati per implementare l'autenticazione unidirezionale creando un certificato fittizio per il gestore code client nel repository delle chiavi.

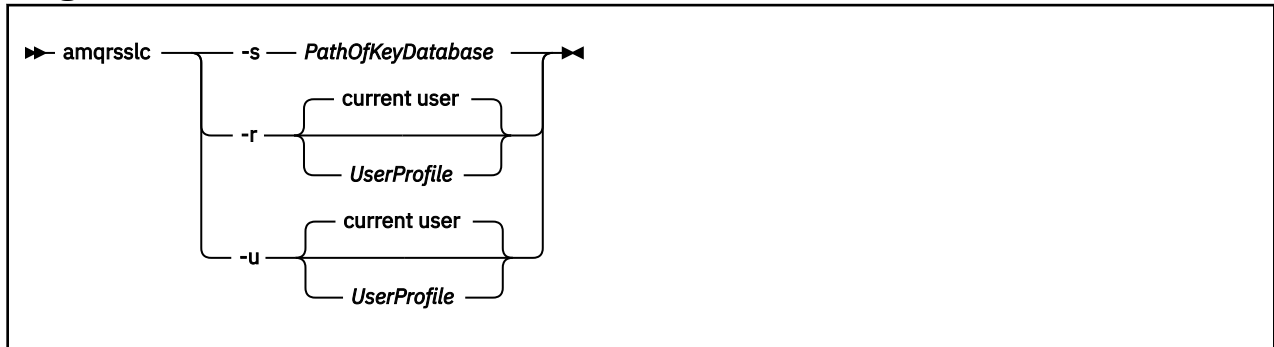
### **Procedura**

1. Effettuare le seguenti operazioni sui gestori code del client e del server:
  - a) Modificare il gestore code in modo da impostare il parametro SSLKEYR immettendo il comando `CHGMQM MQMNAME(SSL) SSLKEYR(*SYSTEM)`.
  - b) Eseguire il stash della password per il repository delle chiavi predefinito immettendo il comando `CHGMQM MQMNAME(SSL) SSLKEYRPWD('xxxxxxx')`.  
La password deve essere racchiusa tra virgolette singole.
  - c) Modificare i canali in modo che abbiano la CipherSpec corretta nel parametro SSLCIPHER.
  - d) Aggiorna la sicurezza TLS immettendo il comando `RFRMQMAUT QMNAME(QMGRNAME) TYPE(*SSL)`.
2. Assegnare il certificato al gestore code del server utilizzando DCM, come segue:
  - a) Accedere all'interfaccia DCM, come descritto in [“Accesso a DCM”](#) a pagina 266.
  - b) Nel pannello di navigazione, fare clic su **Seleziona un archivio certificati**.  
La pagina Seleziona una memorizzazione certificato viene visualizzata nel frame delle attività.
  - c) Selezionare la memorizzazione certificato \*SYSTEM e fare clic su **Continua**.
  - d) Nel pannello di sinistra, espandere **Gestisci applicazioni**.
  - e) Selezionare la definizione **Visualizza applicazione** per verificare che il gestore code sia stato registrato come un'applicazione.  
SSL (WMQ) è elencato nella tabella.
  - f) Selezionare **Aggiorna assegnazione certificato**.
  - g) Selezionare **Server** e fare clic su **Continua**.
  - h) Selezionare QMGRNAME (WMQ) e fare clic su **Aggiorna assegnazione certificato**.
  - i) Selezionare il certificato e fare clic su **Assegna nuovo certificato**. Si apre una finestra che indica che il certificato è stato assegnato all'applicazione.

## Programma di utilità IBM MQ SSL Client (amqrssl) per IBM i

Il programma di utilità IBM MQ SSL Client (amqrssl) per IBM i viene utilizzato da IBM MQ MQI client su sistemi IBM i per registrare o annullare la registrazione del profilo utente client o per memorizzare la password dell'archivio certificati. Il programma di utilità può essere eseguito solo da un utente con un profilo con l'autorizzazione speciale \*ALLOBJ o un membro di QMQMADM che dispone di opzioni per creare o eliminare le registrazioni dell'applicazione in DCM (Digital Certificate Manager).

### Diagramma della sintassi



### Registra profilo utente del client

Se IBM MQ MQI client utilizza la memorizzazione certificato \*SYSTEM, è necessario registrare il profilo utente del client (utente di collegamento) da utilizzare come applicazione con DCM (Digital Certificate Manager).

Se si desidera registrare il profilo utente del client, eseguire il programma **amqrssl** con l'opzione -r con *UserProfile*. Il profilo utente utilizzato quando si richiama **amqrssl** deve avere l'autorizzazione \*USE. Fornendo *UserProfile* con l'opzione -r si registra *UserProfile* come un'applicazione server con un'etichetta di applicazione univoca QIBM\_WEBSPHERE\_MQ\_*UserProfile* e un'etichetta con una descrizione *UserProfile* (WMQ). Questa applicazione server viene quindi visualizzata nel DCM ed è possibile assegnare a questa applicazione qualsiasi certificato server o client nell'archivio di sistema.

**Nota:** Se un profilo utente non è specificato con l'opzione -r, il profilo utente dell'utente che esegue lo strumento **amqrssl** viene registrato.

Il seguente codice utilizza **amqrssl** per registrare un profilo utente. Nel primo esempio, il profilo utente specificato è registrato; nel secondo è il profilo dell'utente collegato:

```
CALL PGM(QMQM/AMQRSSL) PARM('-r' UserProfile)
CALL PGM(QMQM/AMQRSSL) PARM('-r')
```

### Annulla la registrazione del profilo utente client

Per annullare la registrazione del profilo client, eseguire il programma **amqrssl** con l'opzione -u con *UserProfile*. Il profilo utente utilizzato quando si richiama **amqrssl** deve avere l'autorizzazione \*USE. Fornendo *UserProfile* con l'opzione -u si annulla la registrazione di *UserProfile* con l'etichetta QIBM\_WEBSPHERE\_MQ\_*UserProfile* dal DCM.

**Nota:** Se un profilo utente non viene specificato con l'opzione -u, la registrazione del profilo utente dell'utente che esegue lo strumento **amqrssl** viene annullata.

Il seguente codice utilizza **amqrssl** per annullare la registrazione di un profilo utente. Nel primo esempio, la registrazione del profilo utente specificato è annullata; nel secondo è il profilo dell'utente collegato:

```
CALL PGM(QMQM/AMQRSSL) PARM('-u' UserProfile)
CALL PGM(QMQM/AMQRSSL) PARM('-u')
```

## Stash della password dell'archivio certificati

Se IBM MQ MQI client non utilizza la memorizzazione certificato \*SYSTEM e utilizza un'altra memorizzazione certificato (cioè, MQSSLKEYR è impostato su un valore diverso da \*SYSTEM), la password del database delle chiavi deve essere nascosta. Utilizzare l'opzione -s per eseguire lo stash della password del database delle chiavi.

Nel codice seguente, il nome file completo dell'archivio certificati è /Path/Of/KeyDatabase/MyKey.kdb:

```
CALL PGM(QMQM/AMQRSSLC) PARM('-s' '/Path/Of/KeyDatabase/MyKey')
```

L'esecuzione di questo codice comporta una richiesta per la parola d'ordine di questo database di chiavi. Questa password viene nascosta in un file con lo stesso nome del database delle chiavi con estensione .sth. Questo file viene memorizzato nello stesso percorso del database delle chiavi. L'esempio di codice genera un file stash di /Path/Of/KeyDatabase/MyKey.sth. QMQM è il proprietario dell'utente e QMQMADM è il proprietario del gruppo per questo file. QMQM e QMQMADM hanno autorizzazioni di lettura, scrittura e altri profili hanno solo autorizzazioni di lettura.

## **Quando le modifiche ai certificati o all'archivio di certificati diventano effettive su IBM i**

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 nell'archivio certificati e all'attributo del repository chiavi diventano effettive nelle seguenti situazioni:

- Quando un nuovo processo di canale singolo in uscita esegue per la prima volta un canale TLS.
- Quando un nuovo processo di canale singolo TCP/IP in entrata riceve per la prima volta una richiesta di avvio di un canale TLS.
- Quando viene emesso il comando MQSC REFRESH SECURITY TYPE (SSL) per aggiornare l'ambiente TLS IBM MQ .
- Per i processi dell'applicazione client, quando viene chiusa l'ultima connessione TLS nel processo. La connessione TLS successiva acquisirà 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 TLS. Se il processo di pooling del processo ha già eseguito un canale TLS 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 ed esegue per la prima volta un canale TLS. Se il processo dell'iniziatore di canali ha già eseguito un canale TLS 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 avvio di un canale TLS. Se il listener ha già eseguito un canale TLS e si desidera che la modifica diventi immediatamente effettiva, eseguire il comando MQSC REFRESH SECURITY TYPE (SSL).

## **Configurazione dell'hardware di crittografia su IBM i**

Utilizzare questa procedura per configurare il coprocessore crittografico su IBM i

### **Prima di iniziare**

Assicurarsi che il proprio profilo utente disponga delle autorizzazioni speciali \*ALLOBJ e \*SECADM per consentire la configurazione dell'hardware del coprocessore.

## Procedura

1. Andare a `http://machine.domain:2001` o `https://machine.domain:2010`, dove *macchina* è il nome del computer.  
Viene visualizzata una casella di dialogo che richiede un nome utente e una password.
2. Immettere un profilo utente e una password IBM i validi.
3. Andare a [Crittografia](#) e seguire i link appropriati per ulteriori informazioni.

## Operazioni successive

Per ulteriori informazioni sulla configurazione del 4767 Cryptographic Coprocessor, consultare [4767 Cryptographic Coprocessor](#).

### Utilizzo di SSL/TLS in UNIX, Linux, and Windows

Sui sistemi UNIX, Linux, and Windows , il supporto TLS (Transport Layer Security) è installato con IBM MQ. Per informazioni più dettagliate sulle politiche di convalida dei certificati, consultare [Convalida dei certificati e progettazione delle politiche di attendibilità](#).

### Utilizzo di `runmqckm`, `runmqakm` e `strmqikm` per gestire i certificati digitali

Su sistemi UNIX, Linux, and Windows , gestire chiavi e certificati digitali con `strmqikm` (iKeyman) Dalla GUI o dalla riga comandi utilizzando `runmqckm` (iKeycmd) o `runmqakm` (GSKCapiCmd).

#### V 9.0.2



**Attenzione:** Entrambi i comandi `runmqckm` e `strmqikm` si basano su JRE (Java Runtime Environment) IBM MQ . A partire da IBM MQ 9.0.2, se il JRE non è installato, si riceve il messaggio AMQ9183.

- Per sistemi **UNIX and Linux** :
  - Utilizzare il comando `strmqikm` (iKeyman) per avviare la GUI iKeyman .
  - Utilizzare il comando `runmqckm` (iKeycmd) per eseguire attività con l'interfaccia della riga comandi iKeycmd .
  - Utilizzare il comando `runmqakm` (GSKCapiCmd) per eseguire attività con l'interfaccia della riga comandi `runmqakm`. La sintassi del comando per `runmqakm` è uguale a quella per `runmqckm`.  
Se devi gestire i certificati TLS in un modo conforme a FIPS, utilizza 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.

Per ulteriori informazioni, consultare [GSKit: PKCS#11 e IBM MQ Modalità di indirizzamento JRE](#) .

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 devi gestire i certificati TLS in un modo conforme a FIPS, utilizza il comando **runmqakm** invece dei comandi **runmqckm** o **strmqikm** .

- Utilizzare il comando **runmqakm -keydb** con l'opzione *stashpw* o *stash* .

Quando si usa il comando **runmqakm -keydb** in questo modo, ad esempio:

```
runmqakm -keydb -create -db key.kdb -pw secretpwd -stash
```

il file `.sth` risultante non dispone dell'autorizzazione di lettura abilitata per il gruppo di mqm .

Solo il creatore può leggere il file. Dopo aver creato un file di stash utilizzando il comando **runmqakm** , controllare le autorizzazioni del file e concedere l'autorizzazione all'account di servizio che esegue il gestore code o a un gruppo come mqmlocale.

Per richiedere la traccia TLS su sistemi UNIX, Linux o Windows , consultare [strmqtrc](#).

### Riferimenti correlati

comandi [runmqckm](#) e [runmqakm](#)

Questa sezione descrive i comandi [runmqckm](#) e [runmqakm](#) in base all'oggetto del comando.

## ULW

### Configurazione di un repository delle chiavi su UNIX, Linux, and Windows

È possibile configurare un repository delle chiavi utilizzando **strmqikm** (iKeyman) GUI o dalla riga comandi utilizzando i comandi **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd).

### Informazioni su questa attività

Una connessione TLS richiede un *repository chiavi* ad ogni estremità della connessione. Ogni gestore code IBM MQ e IBM MQ MQI client devono avere accesso a un repository delle chiavi. Per ulteriori informazioni, consultare [“Il repository delle chiavi SSL/TLS” a pagina 24](#).

Sui sistemi UNIX, Linux, and Windows , i certificati digitali vengono memorizzati in un file di database delle chiavi gestito utilizzando l'interfaccia utente **strmqikm** o utilizzando i comandi **runmqckm** o **runmqakm** . Questi certificati digitali hanno etichette. Un'etichetta specifica associa un certificato personale a un gestore code o a IBM MQ MQI client. TLS utilizza tale certificato per scopi di autenticazione. Sui sistemi UNIX, Linux, and Windows , IBM MQ utilizza il valore dell'attributo **CERTLABL** , se impostato, oppure il valore predefinito `ibmwebspheremq` con il nome del gestore code o l'ID di accesso utente IBM MQ MQI client accodato, tutti in minuscolo. Per i dettagli, consultare [Etichetta certificato digitale](#) .

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`.

Sui 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 MQ . Ad esempio, `C:\Programmi\IBM\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 IBM 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 un file system che non supporta i 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 UNIX, Linux, and Windows”](#) a pagina 284 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 **strmqikm** o **runmqckm** 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 **strmqikm** o **runmqckm** deve essere un membro del gruppo `mqm`. Per un IBM MQ MQI client, se si esegue **strmqikm** o **runmqckm** da un ID utente diverso da quello con cui viene eseguito il client, è necessario modificare le autorizzazioni del file per consentire a IBM 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 282 o [“Accesso e protezione dei file del database di chiavi su sistemi UNIX and Linux”](#) a pagina 282.

In **strmqikm** o **runmqckm** in IBM WebSphere MQ 7.0, i nuovi database di chiavi vengono popolati automaticamente con una serie di certificati CA predefiniti. In **strmqikm** o **runmqckm** in IBM MQ 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 file del database di chiavi.

**Nota:** A causa di questa modifica del comportamento per GSKit 8.0 che determina l'non aggiunta automatica dei certificati CA al repository, devi aggiungere manualmente i tuoi certificati CA preferiti. Questa modifica di comportamento fornisce un controllo più granulare sui certificati CA utilizzati. Consultare [“Aggiunta di certificati CA predefiniti in un archivio di chiavi vuoto su UNIX, Linux, and Windows con GSKit 8.0”](#) a pagina 282.

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:

- Utilizzo di **runmqckm**:

```
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 MQ, deve essere cms.)



**-stash**

Salva la password del database delle chiavi in un file.

**-fips**

specifica che il comando viene eseguito 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 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/ss1 (su sistemi UNIX and Linux ) o C:\ProgramData\IBM\MQ\qmgrs\QM1\ss1 (su sistemi Windows ).

Il percorso deve corrispondere al valore dell'attributo **SSLKeyRepository** del gestore code.

- Per un client IBM MQ : /var/mqm/ss1 (su sistemi UNIX and Linux ) o C:\mqm\ss1 (su sistemi Windows ).

8. Fare clic su **OK**.

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 password, i tentativi di avviare i canali TLS hanno esito negativo perché non possono ottenere la password richiesta per accedere al file 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 282](#) o [“Accesso e protezione dei file del database di chiavi su sistemi UNIX and Linux” a pagina 282](#).

## Windows 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.

## Linux > UNIX

### Accesso e protezione dei file del database di chiavi su 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 -----`).

## ULW

### Aggiunta di certificati CA predefiniti in un archivio di chiavi vuoto su UNIX, Linux, and Windows con GSKit 8.0

Seguire questa procedura per aggiungere uno o più certificati CA predefiniti a un repository chiavi vuoto con GSKit versione 8.

In GSKit 7.0, il funzionamento 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 8.0, questo comportamento è stato modificato in modo che i certificati CA non vengano aggiunti automaticamente al repository. L'utente deve ora aggiungere manualmente i certificati CA nel repository delle chiavi.

## Utilizzo di `strmqikm`

Sulla macchina su cui si desidera aggiungere il certificato della CA, effettuare le seguenti operazioni:

1. Avviare la GUI utilizzando il comando **`strmqikm`** (su UNIX, Linux, and 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 considerare attendibili 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 **runmqckm**:

- 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 a un repository di chiavi determina l'attendibilità di IBM MQ di 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 del repository delle chiavi per un gestore code su 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 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 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. su 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 livello superiore in cui è installato IBM MQ.

### **ULW** *Modifica dell'ubicazione del repository delle chiavi per un gestore code su UNIX, Linux, and 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, su 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:\Programmi\IBM\MQ\Qmgrs\QM1\ssl\Mykey')
```

Il file di database delle chiavi ha il nome file completo:  
`C:\Programmi\IBM\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 inoltre possibile modificare gli attributi del gestore code utilizzando i comandi IBM 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 delle chiavi su UNIX, Linux, and Windows”](#) a pagina 299.

### **ULW** *Ubicazione del repository delle chiavi per un IBM MQ MQI client su UNIX, Linux, and Windows*

L'ubicazione del repository delle chiavi viene fornita dalla variabile `MQSSLKEYR` o specificata nella chiamata `MQCONN`.

Esaminare la variabile di ambiente `MQSSLKEYR` per trovare l'ubicazione del file del database di chiavi per IBM MQ MQI client. Ad esempio:

```
echo $MQSSLKEYR
```

Controllare anche l'applicazione, poiché il nome del file database delle chiavi può essere impostato anche in una chiamata `MQCONN`, come descritto in [“Specifica dell'ubicazione del repository delle chiavi per un IBM MQ MQI client su UNIX, Linux, and Windows”](#) a pagina 284. Il valore impostato in una chiamata `MQCONN` sostituisce il valore di `MQSSLKEYR`.

### **ULW** *Specifica dell'ubicazione del repository delle chiavi per un IBM MQ MQI client su UNIX, Linux, and Windows*

Non esiste alcun repository delle chiavi predefinito per un IBM MQ MQI client. 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 del database delle chiavi per IBM MQ MQI client in due modi:

- Impostazione della variabile di ambiente MQSSLKEYR. Ad esempio, su 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:\Programmi\IBM\MQ\ssl\key
```

Il file di database delle chiavi ha il nome file completo:

```
C:\Programmi\IBM\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](#).

### **ULW** *Quando le modifiche ai certificati o all'archivio di certificati diventano effettive su UNIX, Linux, and 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 canale singolo in uscita esegue per la prima volta un canale TLS.
- Quando un nuovo processo di canale singolo TCP/IP in entrata riceve per la prima volta una richiesta di avvio di un canale TLS.
- Quando viene emesso il comando MQSC REFRESH SECURITY TYPE (SSL) per aggiornare l'ambiente TLS.
- Per i processi dell'applicazione client, quando viene chiusa l'ultima connessione TLS nel processo. La successiva connessione TLS raccoglierà 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 TLS. Se il processo di pooling del processo ha già eseguito un canale TLS e si desidera che la modifica diventi immediatamente effettiva, eseguire il comando MQSC REFRESH SECURITY TYPE (SSL).
- Per i canali eseguiti come thread dell'inziatore di canali, quando l'inziatore di canali viene avviato o riavviato ed esegue per la prima volta un canale TLS. Se il processo dell'inziatore di canali ha già eseguito un canale TLS 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 avvio di un canale TLS. Se il listener ha già eseguito un canale TLS e si desidera che la modifica diventi immediatamente effettiva, eseguire il comando MQSC REFRESH SECURITY TYPE (SSL).

È anche possibile aggiornare l'ambiente IBM MQ TLS utilizzando i comandi IBM MQ Explorer o PCF.

### **ULW** *Creazione di un certificato personale autofirmato su UNIX, Linux, and Windows*

È possibile creare un certificato autofirmato utilizzando **strmqikm** (iKeyman) Dalla GUI o dalla riga comandi utilizzando **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd).

**Nota:** IBM 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 potresti voler utilizzare i certificati autofirmati, vedi [Utilizzo dei certificati autofirmati per l'autenticazione reciproca di due gestori code](#).

Non tutti i certificati digitali possono essere utilizzati con tutti i CipherSpecs. Assicurati di creare un certificato compatibile con i CipherSpecs che devi utilizzare. IBM MQ supporta tre diversi tipi di CipherSpec. Per i dettagli, consultare [“Interoperabilità di Elliptic Curve e RSA CipherSpecs” a pagina 44](#) nell'argomento [“Certificati digitali e compatibilità CipherSpec in IBM MQ” a pagina 42](#) .

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.

Se si utilizza:

- GUI, consultare [“Utilizzo dell'interfaccia utente strmqikm” a pagina 286](#)
- Riga comandi, consultare [“Utilizzo della riga di comando” a pagina 287](#)

 *Utilizzo dell'interfaccia utente **strmqikm***

È possibile creare un certificato personale utilizzando **strmqikm** (iKeyman) GUI.

## Informazioni su questa attività

**strmqikm** non fornisce una opzione conforme a FIPS. Se è necessario gestire i certificati TLS in modo compatibile con FIPS, utilizzare il comando **runmqakm** .

## Procedura

Completa la seguente procedura per creare un certificato personale per il tuo gestore code o IBM MQ MQI client utilizzando la GUI (graphical user interface):

1. Avviare la GUI 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 **Sfoggia** 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 **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 delle 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 l'etichetta del certificato.  
L'etichetta è il valore dell'attributo **CERTLABL** , se impostato, oppure il valore predefinito **ibmwebspheremq** con il nome del gestore code o l'ID utente di accesso IBM MQ MQI client accodato, tutto in minuscolo. Per i dettagli, consultare [Etichetta certificato digitale](#) .
10. Immettere o selezionare un valore per qualsiasi campo nel campo **DN (Distinguished Name)** o uno dei campi **Nome alternativo oggetto** .
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.

## Operazioni successive

Inoltare una richiesta di certificato a una CA. Consultare [“Ricezione di certificati personali in un repository di chiavi su UNIX, Linux, and Windows”](#) a pagina 293 per ulteriori informazioni.

### **ULW** Utilizzo della riga di comando

È possibile creare un certificato personale dalla riga comandi utilizzando i comandi **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd). Se è necessario gestire i certificati SSL o TLS in modo compatibile con FIPS, utilizzare il comando **runmqakm**.

## Informazioni su questa attività

### Procedura

Creare un certificato personale autofirmato utilizzando il comando **runmqckm** o **runmqakm** (GSKCapiCmd).

- Utilizzando **runmqckm** su UNIX, Linux, and Windows:

```
runmqckm -cert -create -db filename -pw  
password -label label  
          -dn distinguished_name -size key_size  
          -x509version version -expire days  
          -sig_alg algorithm
```

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
```

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. L'etichetta è il valore dell'attributo **CERTLABL**, se impostato, oppure il valore predefinito `ibmwebspheremq` con il nome del gestore code o l'ID utente di collegamento IBM MQ MQI client accodato, tutto in minuscolo. Vedi [“Etichette dei certificati digitali, comprensione dei requisiti”](#) a pagina 26 per i dettagli.

#### **-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.

**Nota:** Gli strumenti **runmqckm** e **runmqakm** si riferiscono all'attributo codice postale come `POSTALCODE`, non come `PC`. Specificare sempre `POSTALCODE` nel parametro **-dn** quando si utilizzano questi comandi di gestione certificati per richiedere certificati con un codice postale.

**-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, 1024 o 2048.

**x509version versione**

La versione del certificato X.509 da creare. Il valore può essere 1, 2 o 3. Il valore predefinito è 3.

**-file nomefile**

Specifica il nome file per la richiesta di certificato.

**-expire giorni**

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. Viene utilizzato solo il componente FIPS ICC e questo 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** ha esito negativo.

**-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, SHA1WithECDSA, SHA1WithRSA, SHA2/ECDSA, SHA224WithECDSA, SHA256\_WITH\_RSA, SHA256WithECDSA, SHA256WithRSA, SHA2WithECDSA, SHA3/ECDSA, SHA384\_WITH\_RSA, SHA384WithECDSA, SHA384WithRSA, SHA3WithECDSA, SHA5/ECDSA, SHA512\_WITH\_RSA, SHA512WithECDSA, SHA512WithRSA, SHA5WithECDSA, SHA\_WITH\_DSA, SHA\_WITH\_RSA, SHAWithDSA, SHAWithRSA. Il valore di default 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.

**Operazioni successive**

Inoltare una richiesta di certificato a una CA. Consultare [“Ricezione di certificati personali in un repository di chiavi su UNIX, Linux, and Windows”](#) a pagina 293 per ulteriori informazioni.

**ULW Richiesta di un certificato personale su UNIX, Linux, and Windows**

È possibile richiedere un certificato personale utilizzando la **strmqikm** (iKeyman) GUI o dalla riga comandi utilizzando i comandi **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd). Se è necessario gestire i certificati SSL o TLS in modo compatibile con FIPS, utilizzare il comando **runmqakm**.



## Informazioni su questa attività

È possibile richiedere un certificato personale utilizzando la GUI di **strmqikm** o dalla riga comandi, in base alle seguenti considerazioni:

- IBM 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. IBM MQ supporta tre diversi tipi di CipherSpec. Per i dettagli, consultare [“Interoperabilità di Elliptic Curve e RSA CipherSpecs”](#) a pagina 44 nell'argomento [“Certificati digitali e compatibilità CipherSpec in IBM MQ”](#) a pagina 42 .
- 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 307.

Se si utilizza:

- GUI, consultare [“Utilizzo dell'interfaccia utente strmqikm”](#) a pagina 289
- Riga comandi, consultare [“Utilizzo della riga di comando”](#) a pagina 290

### Utilizzo dell'interfaccia utente **strmqikm**

È possibile richiedere un certificato personale utilizzando la **strmqikm** (iKeyman) GUI o dalla riga comandi utilizzando i comandi **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd). Se è necessario gestire i certificati SSL o TLS in modo compatibile con FIPS, utilizzare il comando **runmqakm** .

## Informazioni su questa attività

**strmqikm** non fornisce una opzione conforme a FIPS. Se è necessario gestire i certificati 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 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 **Sfoggia** 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 l'etichetta del certificato.

L'etichetta è il valore dell'attributo **CERTLABL** , se impostato, oppure il valore predefinito `ibmwebsphermq` con il nome del gestore code o l'ID utente di accesso IBM MQ MQI client accodato, tutto in minuscolo. Per i dettagli, consultare [Etichetta certificato digitale](#) .

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.arm` oppure 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 certificati viene memorizzata nel file scelto nel passo [“11”](#) a pagina 290.

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.

### **ULW** Utilizzo della riga di comando

È possibile richiedere un certificato personale dalla riga comandi utilizzando i comandi **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd). Se è necessario gestire i certificati SSL o TLS in modo compatibile con FIPS, utilizzare il comando **runmqakm** .

## Procedura

Richiedere un certificato personale utilizzando il comando **runmqckm** o **runmqakm** (GSKCapiCmd).

- 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. L'etichetta è il valore dell'attributo **CERTLABL** , se impostato, oppure il valore predefinito `ibmwebsphermq` con il nome del gestore code o l'ID utente di collegamento IBM MQ MQI client accodato, tutto in minuscolo. Vedi [“Etichette dei certificati digitali, comprensione dei requisiti”](#) a pagina 26 per i dettagli.

**-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.

**Nota:** Gli strumenti **runmqckm** e **runmqakm** si riferiscono all'attributo codice postale come POSTALCODE, non come PC. Specificare sempre POSTALCODE nel parametro **-dn** quando si utilizzano questi comandi di gestione certificati per richiedere certificati con un codice postale.

**-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, 1024 o 2048.

**-file nomefile**

Specifica il nome file per la richiesta di certificato.

**-fips**

specifica che il comando viene eseguito 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, SHA1WithECDSA, SHA1WithRSA, SHA2/ECDSA, SHA224WithECDSA, SHA256\_WITH\_RSA, SHA256WithECDSA, SHA256WithRSA, SHA2WithECDSA, SHA3/ECDSA, SHA384\_WITH\_RSA, SHA384WithECDSA, SHA384WithRSA, SHA3WithECDSA, SHA5/ECDSA, SHA512\_WITH\_RSA, SHA512WithECDSA, SHA512WithRSA, SHA5WithECDSA, SHA\_WITH\_DSA, SHA\_WITH\_RSA, SHAWithDSA, SHAWithRSA. Il valore di default 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.

## Operazioni successive

Inoltare una richiesta di certificato a una CA. Consultare [“Ricezione di certificati personali in un repository di chiavi su UNIX, Linux, and Windows”](#) a pagina 293 per ulteriori informazioni.

### **Rinnovo di un certificato personale esistente su UNIX, Linux, and Windows**

È possibile rinnovare un certificato personale utilizzando **strmqikm** (iKeyman) GUI o dalla riga comandi utilizzando i comandi **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd).

## Informazioni su questa attività

Se hai il requisito di utilizzare dimensioni di chiavi più grandi per i tuoi certificati personali, non puoi rinnovare un certificato esistente. È necessario sostituire la chiave esistente attenendosi alla procedura descritta in [“Richiesta di un certificato personale su UNIX, Linux, and Windows”](#) a pagina 288 per creare una nuova richiesta di certificato che utilizzi le dimensioni chiave richieste.

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 **strmqikm**

## Informazioni su questa attività

**strmqikm** non fornisce una opzione conforme a FIPS. Se è necessario gestire i certificati TLS in modo compatibile con FIPS, utilizzare il comando **runmqakm**.

## Procedura

Completare la seguente procedura per applicare un certificato personale, utilizzando l'interfaccia utente **strmqikm**:

1. Avviare l'interfaccia utente utilizzando il comando **strmqikm** su 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 **Sfoggia** 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 certificato viene memorizzata nel file selezionato nel passo [“9”](#) a pagina 292.
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 **runmqckm** o **runmqakm**:

- Utilizzo di **runmqckm** 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 repository di chiavi su UNIX, Linux, and Windows"](#) a pagina 293.

### **Ricezione di certificati personali in un repository di chiavi su UNIX, Linux, and 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 **strmqikm**

Se è necessario gestire i certificati TLS in modo conforme a FIPS, utilizzare il comando **runmqakm . strmqikm** non fornisce una opzione conforme a FIPS.

Assicurarsi 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 IBM MQ MQI client per ricevere un certificato personale nel file del database delle chiavi:

1. Avviare la GUI 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 **Sfoggia** 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 **Sfoggia** 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

Per aggiungere un certificato personale a un file di database delle chiavi, utilizzare uno dei seguenti comandi:

- Utilizzo di **runmqckm**:

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

- Utilizzo di **runmqakm**:

```
runmqakm -cert -receive -file filename -db filename -pw password -fips
```

dove:

### **-file nomefile**

Specifica il nome file completo del certificato personale.

### **-db nomefile**

Specifica il nome file completo di un database di chiavi CMS.

### **-pw password**

Specifica la password per il database di chiavi CMS.

### **-format ascii**

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. 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.

Se si utilizza l'hardware crittografico, fare riferimento a [“Ricezione di un certificato personale nell'hardware PKCS #11”](#) a pagina 308.

## Estrazione di un certificato CA da un repository delle chiavi su UNIX, Linux, and Windows

Seguire questa procedura per estrarre un certificato CA.

## Utilizzo di **strmqikm**

Se è necessario gestire i certificati TLS in modo conforme a FIPS, utilizzare il comando **runmqakm**. **strmqikm** (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 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 **Sfoggia** 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 **Sfoggia** 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 **runmqckm**:

- Su UNIX, Linux, and Windows:

```
runmqckm -cert -extract -db filename -pw password -label label -target filename
        -format ascii
```

dove:

<code>-db filename</code>	è il nome percorso completo di un database di chiavi CMS.
<code>-pw password</code>	è la password per il database di chiavi CMS.
<code>-label label</code>	è l'etichetta allegata al certificato.
<code>-target filename</code>	è il nome del file di destinazione.
<code>-format ascii</code>	è il formato del certificato. Il valore può essere <code>ascii</code> per ASCII con codifica Base64 oppure <code>binary</code> per i dati binari DER. Il valore predefinito è <code>ascii</code> .
<code>-fips</code>	specifica che il comando viene eseguito 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 <b>runmqckm</b> non riesce.

## Estrazione della parte pubblica di un'autocertificazione da un repository delle chiavi su UNIX, Linux, and Windows

Seguire questa procedura per estrarre la parte pubblica di un certificato autofirmato.

### Utilizzo di **strmqikm**

Se è necessario gestire i certificati TLS in modo conforme a FIPS, utilizzare il comando **runmqckm**. **strmqikm** (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 utilizzando il comando **strmqikm** (su UNIX, Linux, and 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 **Sfoggia** 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 **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**.
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 **Sfoggia** 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 **runmqckm** o **runmqakm**:

- Su UNIX, Linux, and 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 <i>filename</i>	è il nome percorso completo di un database di chiavi CMS.
-pw <i>password</i>	è la password per il database di chiavi CMS.
-label <i>label</i>	è l'etichetta allegata al certificato.
-target <i>filename</i>	è il nome del file di destinazione.
-format <i>ascii</i>	è il formato del certificato. Il valore può essere <i>ascii</i> per ASCII con codifica Base64 oppure <i>binary</i> per i dati binari DER. Il valore predefinito è <i>ascii</i> .
-fips	specifica che il comando viene eseguito 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 <b>runmqakm</b> non riesce.

## **Aggiunta di un certificato CA, o della parte pubblica di un certificato autofirmato, in un repository delle chiavi su 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:** È necessario assicurarsi che il certificato sia in codifica ASCII (UTF-8) o binaria (DER), poiché IBM Global Secure Toolkit (GSKit) non supporta certificati con altri tipi di codifica.



## Utilizzo di **strmqikm**

Se è necessario gestire i certificati TLS in modo conforme a FIPS, utilizzare il comando **runmqakm**. **strmqikm** non fornisce una opzione conforme a FIPS.

Sulla macchina su cui si desidera aggiungere il certificato della CA, effettuare le seguenti operazioni:

1. Avviare la GUI utilizzando il comando **strmqikm** (su 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 **Sfoggia** 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 **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 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 **Sfoggia** 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

Per aggiungere un certificato CA ad un database delle chiavi, utilizzare uno dei seguenti comandi:

- Utilizzo di **runmqckm**:

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

- Utilizzo di **runmqakm**:

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

dove:

**-db nomefile**

Specifica il nome file completo del database di chiavi CMS.

**-pw password**

Specifica la password per il database di chiavi CMS.

**-label label**

Specifica l'etichetta allegata al certificato.

**-file nomefile**

Specifica il nome del file contenente il certificato.

**-format ascii**

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. 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.

## Esportazione di un certificato personale da un repository delle chiavi su UNIX, Linux, and Windows

Seguire questa procedura per esportare un certificato personale.

### Utilizzo di `strmqikm`

Se è necessario gestire i certificati TLS in modo conforme a FIPS, utilizzare il comando `runmqckm`. `strmqikm` (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 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 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 `runmqckm`:

- Su UNIX, Linux, and 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.
- fips specifica che il comando viene eseguito 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 `runmqckm` non riesce.
- 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.

## **ULW** **Importazione di un certificato personale in un repository delle chiavi su 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 emessi al file di database delle chiavi (consultare [“Aggiunta di un certificato CA, o della parte pubblica di un certificato autofirmato, in un repository delle chiavi su UNIX, Linux, and Windows”](#) a pagina 296 ).

I file PKCS #12 devono essere considerati temporanei ed eliminati dopo l'utilizzo.

### **Utilizzo di `strmqikm`**

Se è necessario gestire i certificati TLS in modo compatibile con FIPS, utilizzare il comando `runmqakm`. `strmqikm` non fornisce una opzione conforme a FIPS.

Eeguire le seguenti operazioni sulla macchina su cui si desidera importare il certificato personale:

1. Avviare la GUI 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. È possibile modificare le etichette dei certificati importati se, ad esempio, un certificato con la stessa etichetta già esiste nel database di chiavi di destinazione. La modifica delle etichette certificato non ha alcun effetto sulla convalida della catena di certificati. Per associare il certificato a un determinato gestore code o IBM MQ MQI client, IBM MQ utilizza il valore dell'attributo **CERTLABL**, se impostato, oppure il valore predefinito `ibmwebspheremq` con il nome del gestore code o l'ID di accesso utente IBM MQ MQI client accodato, tutto in minuscolo. Per i dettagli, consultare [Etichetta certificato digitale](#).

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 Gestione chiavi originale viene visualizzata nuovamente con i campi **Certificati personali** e **Certificato 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 **runmqckm**, utilizzare il seguente comando:

- Su UNIX, Linux, and 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 .
- fips specifica che il comando viene eseguito 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 **runmqckm** non riesce.

**runmqckm** 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 di un certificato personale da un file Microsoft.pfx**

Seguire questa procedura per importare da un file Microsoft.pfx su UNIX, Linux, and Windows.

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 utilizzando il comando **strmqikm** . 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 **Si**. 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 il valore dell'attributo IBM MQ **CERTLABL** , se è impostato, oppure il valore

predefinito `ibmwebspheremq` con il nome del gestore code o l'ID di accesso dell'utente IBM MQ MQI client accodato, tutti in minuscolo. Per i dettagli, consultare [Etichetta certificato digitale](#).

21. Fare clic su **OK**. La finestra Modifica etichette viene ora rimossa e la finestra IBM Key Management originale viene nuovamente visualizzata con i pannelli Certificati personali e Certificati del 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 di certificato utilizzando **runmqckm** o **runmqakm**.

## Utilizzo della riga di comando

Per importare un certificato personale utilizzando **runmqckm** su UNIX, Linux, and Windows, utilizzare il seguente comando:

```
runmqckm -cert -import -file filename -pw password -type pkcs12 -target filename  
-target_pw password -target_type cms -label label -pfx
```

Per importare un certificato personale utilizzando **runmqakm**, utilizzare il seguente comando:

```
runmqakm -cert -import -file filename -pw password -type pkcs12 -target filename  
-target_pw password -target_type cms -label label -fips -pfx
```

dove:

<code>-file filename</code>	è il nome file completo del file contenente il certificato PKCS #12 .
<code>-pw password</code>	è la parola d'ordine per il certificato #12 PKCS.
<code>-type pkcs12</code>	è il tipo di file.
<code>-target filename</code>	è il nome del database di chiavi CMS di destinazione.
<code>-target_pw password</code>	è la password per il database di chiavi CMS.
<code>-target_type cms</code>	è il tipo di database specificato da <code>-target</code>
<code>-label label</code>	è l'etichetta del certificato da importare dal database delle chiavi origine.
<code>-new_label label</code>	è l'etichetta che il certificato verrà assegnato nel database di destinazione. Se si omette l'opzione <code>-new_label</code> , il valore predefinito è utilizzare lo stesso dell'opzione <code>-label</code> .
<code>-fips</code>	specifica che il comando viene eseguito 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 <b>runmqakm</b> non riesce.
<code>-pfx</code>	indica il formato file PFX.

**runmqckm** 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 di un certificato personale da un file PKCS #7

Gli strumenti **strmqikm** (iKeyman) e **runmqckm** (iKeycmd) non supportano PKCS #7 ( .p7b ) file. Utilizzare lo strumento **runmqckm** per importare i certificati da un file PKCS #7 su UNIX, Linux, and Windows.

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 <i>filename</i>	è il nome file completo del database delle chiavi CMS.
-pw <i>password</i>	è la password per il database delle chiavi.
-type <i>cms</i>	è il tipo di database delle chiavi.
-file <i>filename</i>	è il nome del file PKCS #7 .
-label <i>label</i>	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 <i>filename</i>	è il nome file completo del file contenente il certificato PKCS #7 .
-pw <i>password</i>	è la parola d'ordine per il certificato #7 PKCS.
-type <i>pkcs7</i>	è il tipo di file.
-target <i>filename</i>	è il nome del database delle chiavi di destinazione.
-target_pw <i>password</i>	è la password per il database delle chiavi di destinazione.
-target_type <i>cms</i>	è il tipo di database specificato da -target
-label <i>label</i>	è l'etichetta del certificato da importare.
-new_label <i>label</i>	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 UNIX, Linux, and Windows

Utilizzare questa procedura per rimuovere i certificati personali o CA.

### Utilizzo di **strmqikm**

Se è necessario gestire i certificati TLS in modo conforme a FIPS, utilizzare il comando **runmqakm** . **strmqikm** (iKeyman) non fornisce un'opzione conforme a FIPS.

1. Avviare la GUI utilizzando il comando **strmqikm** (su UNIX, Linux, and 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 su UNIX, Linux, and Windows”](#) a pagina 298).
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 **runmqckm**:

- Su UNIX, Linux, and Windows:

```
runmqckm -cert -delete -db filename -pw password -label label
```

dove:

-db <i>filename</i>	è il nome file completo di un database di chiavi CMS.
-pw <i>password</i>	è la password per il database di chiavi CMS.
-label <i>label</i>	è l'etichetta allegata al certificato personale.
-fips	specifica che il comando viene eseguito 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 <b>runmqakm</b> non riesce.

## **ULW** Generazione di password complesse per la protezione del repository delle chiavi su UNIX, Linux, and Windows

È possibile generare password complesse per la protezione del repository delle chiavi utilizzando il comando **runmqakm** (GSKCapiCmd).

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 a una richiesta da **runmqckm**, **runmqakm** o dalla GUI di **strmqikm**, non è necessario racchiudere tra virgolette o ignorare la password. Non è necessario perché la shell del sistema operativo non influisce sull'immissione dei dati in questi casi.

## **ULW** Configurazione dell'hardware crittografico su 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 il gestore code su UNIX, Linux, and Windows utilizzando uno dei metodi riportati di seguito:

- Utilizzare il comando ALTER QMGR MQSC con il parametro SSLCRYP, come descritto in [ALTER QMGR](#).



- Utilizzare IBM MQ Explorer per configurare l'hardware crittografico 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 IBM MQ su UNIX, Linux, and Windows utilizzando uno dei metodi riportati di seguito:

- 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\_PKCS11 del parametro SSLCRYP, l'etichetta del token PKCS #11 deve corrispondere all'etichetta con cui è stato configurato l'hardware.

- 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 305.

### *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 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:

- Utilizzo di **runmqckm**:

```
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 MQ, deve essere cms.)

**-stash**

Salva la password del database delle chiavi in un file.

### -fips

specifica che il comando viene eseguito 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. Aprire il file delle proprietà di sicurezza Java, `java.security`.

- Su sistemi UNIX and Linux , il file delle proprietà di sicurezza Java si trova nella sottodirectory `java/jre64/jre/lib/security` della directory di installazione IBM MQ .
- Su sistemi Windows , il file delle proprietà di sicurezza Java si trova nella sottodirectory `java\jre\lib\security` della directory di installazione IBM MQ .

Se non è già presente nel file, aggiungere il provider di sicurezza `IBMPKCS11Impl` . Ad esempio, aggiungendo la seguente riga:

```
security.provider.12=com.ibm.crypto.pkcs11impl.provider.IBMPKCS11Impl
```

4. Avviare l'interfaccia utente eseguendo il comando **strmqikm** .

5. Fare clic su **File database di chiavi > Apri**.

6. Fare clic su **Tipo di database delle chiavi** e selezionare **PKCS11Direct**.

7. 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, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

8. Nel campo **Ubicazione** , immettere il percorso:

- Su sistemi UNIX and Linux , potrebbe essere `/usr/lib/pkcs11`, 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.

9. Selezionare l'etichetta del token dell'unità crittografica che si desidera utilizzare per memorizzare i certificati.

10. Nel campo **Password token crittografico** , immettere la parola d'ordine impostata durante la configurazione dell'hardware crittografico.

11. 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 [“15” a pagina 307](#).

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**.

12. 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.

13. Nel campo **Ubicazione**, immettere il percorso, ad esempio:

- Per un gestore code: /var/mqm/qmgrs/QM1/ssl
- Per un IBM MQ MQI client: /var/mqm/ssl

Fare clic su **OK**. Viene visualizzata la finestra Richiesta password.

14. Immetti una password.

Se è stato selezionato **Apri file database di chiavi secondario esistente** nel passo [“11” a pagina 307](#), immettere una password nel campo **Password**.

Se si seleziona **Crea nuovo file di database delle chiavi secondario** nel passo [“11” a pagina 307](#), 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**. Notare che se non si esegue lo stash della password, i tentativi di avviare i canali TLS 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).

15. 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 IBM MQ MQI client per richiedere un certificato personale per l'hardware di crittografia.

## Informazioni su questa attività

Questa attività descrive come utilizzare l'interfaccia utente **strmqikm** per richiedere un certificato personale. Se si utilizza la CLI (command line interface), consultare [“Utilizzo della riga di comando” a pagina 290](#).

**Nota:** IBM 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 **strmqikm** (iKeyman), completare la seguente procedura:

1. Completare la procedura per gestire l'hardware crittografico. Consultare [“Gestione dei certificati sull'hardware PKCS #11” a pagina 305](#).
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 l'etichetta del certificato.

L'etichetta è il valore dell'attributo **CERTLABL** , se impostato, oppure il valore predefinito `ibmwebspheremq` con il nome del gestore code o l'ID utente di accesso IBM MQ MQI client accodato, tutto in minuscolo. Per i dettagli, consultare [Etichetta certificato digitale](#) .

4. Selezionare la **Dimensione chiave** e l' **Algoritmo di firma** richiesti.
5. 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](#).
6. 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.
7. Fare clic su **OK**.  
Viene aperta una finestra di conferma.
8. Fare clic su **OK**.  
L'elenco **Richieste di certificati personali** mostra l'etichetta della nuova richiesta di certificato personale creata. La richiesta di certificati viene memorizzata nel file scelto nel passo [“6” a pagina 308](#).
9. 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.

#### *Ricezione di un certificato personale nell'hardware PKCS #11*

Utilizzare questa procedura per un gestore code o un IBM MQ MQI client per ricevere un certificato personale per l'hardware di crittografia.

### Prima di iniziare

Aggiungere il certificato CA della CA che ha firmato il certificato personale. Aggiungerlo all'hardware crittografico o al database di chiavi CMS secondario. Eseguire questa operazione prima di ricevere il certificato firmato nell'hardware crittografico. Per aggiungere un certificato CA a un keyring, seguire la procedura in [“Aggiunta di un certificato CA, o della parte pubblica di un certificato autofirmato, in un repository delle chiavi su UNIX, Linux, and Windows” a pagina 296](#).

### Procedura

- Per ricevere un certificato personale utilizzando l'interfaccia utente di **strmqikm** (iKeyman), completare la seguente procedura:
  - a) Completare la procedura per gestire l'hardware crittografico. Consultare [“Gestione dei certificati sull'hardware PKCS #11” a pagina 305](#).
  - b) Fare clic su **Ricevi**. Viene visualizzata la finestra Ricevi certificato da file.
  - c) Immettere il nome file del certificato e l'ubicazione per il nuovo certificato personale oppure fare clic su **Sfogli**a per selezionare il nome e l'ubicazione.
  - d) 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.
  - e) Fare clic su **Si** o **No**. Viene visualizzata la finestra Immettere un'etichetta.
  - f) 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.
- Per ricevere un certificato personale utilizzando il comando **runmqakm** (GSKCapiCmd), completare la seguente procedura:
  - a) Aprire una finestra comandi configurata per il proprio ambiente.
  - b) Ricevere il certificato personale utilizzando il comando **runmqakm** (GSKCapiCmd):

```
runmqakm -cert -receive -file filename -crypto module_name
        -tokenlabel hardware_token -pw hardware_password
        -format cert_format -fips
        -secondaryDB filename -secondaryDBpw password
```

dove:

**-file nomefile**

Specifica il nome file completo del file contenente il certificato personale.

**-crypto nome\_modulo**

Specifica il nome completo della libreria PKCS #11 fornita con l'hardware crittografico.

**-tokenlabel token hardware**

Specifica l'etichetta del token dell'unità crittografica PKCS #11 .

**-pw password\_hardware**

Specifica la parola d'ordine per l'accesso all'hardware crittografico.

**-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. 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.

**-secondaryDB nome file**

Specifica il nome file completo del database di chiavi CMS.

**-secondaryDBpw password**

Specifica la password per il database di chiavi CMS.

## Utilizzo di SSL/TLS in IBM MQ Appliance

IBM MQ Appliance ha il supporto TLS (Transport Layer Security).

IBM MQ Appliance dispone di comandi distinti per la gestione dei certificati. Per informazioni dettagliate sulla gestione dei certificati, consultare la documentazione IBM MQ Appliance , [Gestione dei certificati TLS](#)

## Utilizzo di SSL/TLS in z/OS

Queste informazioni descrivono come configurare e utilizzare TLS (Transport Layer Security) su z/OS.

Ogni argomento include esempi di esecuzione di ciascuna attività utilizzando RACF. È possibile eseguire attività simili utilizzando altri gestori di sicurezza esterni.

Su z/OS, è necessario impostare anche il numero di attività secondarie del server che ciascun gestore code utilizza per elaborare le chiamate TLS, come descritto in [“Impostazione del parametro SSLTASKS su z/OS”](#) a pagina 310.

z/OS Il supporto TLS è parte integrante del sistema operativo ed è noto come *SSL di sistema*. System SSL fa parte dell'elemento Cryptographic Services Base di z/OS. I membri Cryptographic Services Base sono installati in *pdsname*. PDS (partitioned data set) SIEALNKE. Quando si installa SSL di sistema, assicurarsi di scegliere le opzioni appropriate per fornire i CipherSpecs richiesti.

## Ulteriori requisiti ID utente per TLS su z/OS

Queste informazioni descrivono i requisiti aggiuntivi necessari all'ID utente per configurare e gestire TLS su z/OS.

Assicurarsi di disporre di tutti gli aggiornamenti HIPER (High Impact o Pervasive) appropriati sul sistema.

Assicurarsi di aver configurato i seguenti prerequisiti:

- L'ID utente *ssidCHIN* è definito correttamente in RACFe l'ID utente *ssidCHIN* ha accesso READ ai seguenti profili:
  - IRR.DIGTCERT.LIST
  - IRR.DIGTCERT.LISTRING

Queste variabili sono definite in RACF FACILITY Class.

- L'ID utente *ssidCHIN* è il proprietario del keyring.
- Il certificato personale del gestore code, se creato dal comando RACDCERT, viene creato con un ID utente del tipo di certificato che è uguale all'ID utente *ssidCHIN*.
- L'iniziatore del canale viene riciclato oppure viene emesso il comando **REFRESH SECURITY TYPE(SSL)** per rilevare eventuali modifiche apportate al key ring.
- La procedura IBM MQ Channel Initiator ha accesso alla libreria di runtime SSL del sistema *pdsname.SIEALNKE* tramite l'elenco di link, LPA o un'istruzione STEPLIB DD. Questa libreria deve essere autorizzata APF.
- L'ID utente con la cui autorizzazione l'iniziatore di canali è in esecuzione è configurato per utilizzare USS (UNIX System Services), come descritto nella documentazione di z/OS UNIX System Services Planning.

Gli utenti che non desiderano che l'iniziatore di canali richiami UNIX System Services utilizzando l'UID predefinito e il segmento OMVS, devono solo modellare un nuovo segmento OMVS basato sul segmento predefinito poiché l'iniziatore di canali non richiede autorizzazioni speciali e non viene eseguito all'interno di UNIX come superutente.

### **Impostazione del parametro SSLTASKS su z/OS**

Utilizzare il comando ALTER QMGR per impostare il numero di attività secondarie del server per l'elaborazione delle chiamate TLS

Per utilizzare i canali TLS, assicurarsi che vi siano almeno due attività secondarie del server impostando il parametro SSLTASKS, utilizzando il comando ALTER QMGR. Ad esempio:

```
ALTER QMGR SSLTASKS(5)
```

Per evitare problemi con l'assegnazione della memoria, non impostare l'attributo SSLTASKS su un valore maggiore di otto in un ambiente in cui non esiste alcun controllo CRL (Certificate Revocation List).

Se viene utilizzato il controllo CRL, un SSLTASK viene trattenuto dal canale interessato per la durata di tale controllo. Ciò potrebbe verificarsi per un tempo trascorso significativo mentre viene contattato il server LDAP pertinente, poiché ogni SSLTASK è un blocco di controllo attività z/OS.

È necessario riavviare l'iniziatore del canale se si modifica il valore dell'attributo SSLTASKS.

### **Configurazione di un repository delle chiavi su z/OS**

Impostare un repository delle chiavi ad entrambe le estremità della connessione. Associare ciascun repository delle chiavi al relativo gestore code.

Una connessione TLS richiede un *repository chiavi* ad ogni estremità della connessione. Ogni gestore code deve avere accesso a un repository delle chiavi. Utilizzare il parametro SSLKEYR nel comando ALTER QMGR per associare un contenitore chiavi ad un gestore code. Per ulteriori informazioni, fare riferimento a [“Il repository delle chiavi SSL/TLS” a pagina 24](#).

Su z/OS, i certificati digitali vengono memorizzati in *key ring* gestito da ESM (External Security Manager). Questi certificati digitali hanno etichette che associano il certificato a un gestore code. TLS utilizza questi certificati per scopi di autenticazione. Tutti gli esempi che seguono utilizzano i comandi RACF. Esistono comandi equivalenti per altri programmi ESM.

In z/OS, IBM MQ utilizza il valore dell'attributo **CERTLABL**, se è impostato, oppure il valore predefinito *ibmWebSphereMQ* con il nome del gestore code aggiunto. Per i dettagli, consultare [Etichetta certificato digitale](#).

Il nome del repository delle chiavi per un gestore code è il nome di un file di chiavi nel database RACF . È possibile specificare il nome del file di chiavi prima o dopo la creazione del file di chiavi.

Utilizzare la seguente procedura per creare un nuovo keyring per un gestore code:

1. Accertarsi di disporre dell'autorità appropriata per emettere il comando RACDCERT (per ulteriori dettagli, consultare il manuale *SecureWay Security Server RACF Command Language Reference* ).
2. Emetti il seguente comando:

```
RACDCERT ID( userid1 ) ADDRING( ring-name )
```

dove:

- *userid1* è l'ID utente dello spazio di indirizzo dell'iniziatore di canali o l'ID utente che sarà proprietario del file di chiavi (se il file di chiavi è condiviso).
- *ring - name* è il nome che si desidera dare al proprio key ring. La lunghezza di questo nome può essere fino a 237 caratteri. Il nome è sensibile al maiuscolo/minuscolo. Specificare *ring - name* in caratteri maiuscoli per evitare problemi.

### **Come rendere i certificati CA disponibili per un gestore code su z/OS**

Una volta creato il keyring, connettere tutti i certificati CA pertinenti.

Se si dispone del certificato CA in un dataset, è necessario prima aggiungere il certificato nel database RACF utilizzando il seguente comando:

```
RACDCERT ID( userid1 ) ADD( input-data-set-name ) WITHLABEL( 'My CA' )
```

Quindi, per collegare un certificato CA per My CA al tuo keyring, utilizza il seguente comando:

```
RACDCERT ID(userid1)  
CONNECT(CERTAUTH LABEL( 'My CA' ) RING(ring-name) USAGE(CERTAUTH))
```

dove *userid1* è l'ID utente iniziatore del canale o il proprietario di un keyring condiviso.

Per ulteriori informazioni sui certificati CA, fare riferimento a [“Certificati digitali”](#) a pagina 9.

### **Individuazione del repository delle chiavi per un gestore code su z/OS**

Utilizzare questa procedura per ottenere l'ubicazione del keyring del gestore code.

1. Visualizzare gli attributi del gestore code utilizzando uno dei seguenti comandi MQSC:

```
DISPLAY QMGR ALL  
DISPLAY QMGR SSLKEYR
```

2. Esaminare l'emissione del comando per individuare l'ubicazione del keyring.

### **Specifica dell'ubicazione del repository delle chiavi per un gestore code su z/OS**

Per specificare l'ubicazione del file di chiavi del gestore code, utilizzare il comando ALTER QMGR MQSC per impostare l'attributo del repository delle chiavi del gestore code.

Ad esempio:

```
ALTER QMGR SSLKEYR(CSQ1RING)
```

se il keyring è di proprietà dello spazio di indirizzo dell'iniziatore di canali oppure:

```
ALTER QMGR SSLKEYR(userid1/CSQ1RING)
```

se si tratta di un file keyring condiviso, dove *userid1* è l'ID utente proprietario del file keyring.

## **z/OS** **Concessione all'inziatore di canali dei diritti di accesso corretti su z/OS**

L'inziatore di canali (CHINIT) ha bisogno dell'accesso al repository delle chiavi e ad alcuni profili di sicurezza.

### **Concessione dell'accesso CHINIT per leggere il repository chiavi**

Se il repository delle chiavi è di proprietà dell'ID utente CHINIT, questo ID utente necessita dell'accesso in lettura all'IRR.IRR.DIGTCERT.LISTRING nella classe FACILITY, altrimenti aggiornare l'accesso. Concedere l'accesso utilizzando il comando PERMIT con ACCESS (UPDATE) o ACCESS (READ) come appropriato:

```
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ID( userid ) ACCESS(UPDATE)
```

dove *userid* è l'ID utente dello spazio di indirizzo dell'inziatore di canali.

### **Concessione dell'accesso in lettura CHINIT ai profili CSF\* appropriati**

Per il supporto hardware fornito tramite ICSF (Integrated Cryptographic Service Facility) da utilizzare, assicurarsi che l'ID utente CHINIT disponga dell'accesso in lettura ai profili CSF\* appropriati nella classe CSFSERV utilizzando il seguente comando:

```
PERMIT csf-resource CLASS(CSFSERV) ID( userid ) ACCESS(READ)
```

dove *csf-resource* è il nome del profilo CSF\* e *userid* è l'ID utente dello spazio di indirizzo dell'inziatore di canali.

Ripetere questo comando per ciascuno dei seguenti profili CSF\*:

- CSFDSG
- CSFDSV
- CFPKD
- CFPKE
- CFPKI

L'ID utente CHINIT potrebbe anche richiedere l'accesso in lettura ad altri profili CSF\*. Ad esempio, se si sta utilizzando la specifica di cifratura ECDHE\_RSA\_AES\_256\_GCM\_SHA384, l'ID utente CHINIT ha anche bisogno dell'accesso in lettura ai seguenti profili CSF\*:

- CSF1DVK
- CSF1GAV
- CSF1GKP
- CSF1SKE
- CSF1TRC
- CSF1TRD

Per ulteriori informazioni, consultare [RACF CSFSERV resource requirements](#).

Se le chiavi del certificato sono memorizzate in ICSF e l'installazione ha stabilito il controllo dell'accesso sulle chiavi memorizzate in ICSF, assicurarsi che l'ID utente CHINIT disponga dell'accesso in lettura al profilo nella classe CSFKEYS utilizzando il seguente comando:

```
PERMIT IRR.DIGTCERT. userid.* CLASS(CSFKEYS) ID( userid ) ACCESS(READ)
```

dove *userid* è l'ID utente dello spazio di indirizzo dell'inziatore di canali.



## Utilizzo di ICSF (Integrated Cryptographic Service Facility)

L'iniziatore di canali può utilizzare ICSF per generare un numero casuale quando si esegue il seeding dell'algoritmo di protezione della password per nascondere le password che fluiscono sui canali client se TLS non viene utilizzato.

Per ulteriori informazioni, consultare [“Utilizzo di ICSF \(Integrated Cryptographic Service Facility\)”](#) a pagina 256

### **Quando le modifiche ai certificati o al repository delle chiavi diventano effettive su z/OS**

Le modifiche diventano effettive all'avvio dell'iniziatore di canali o all'aggiornamento del repository.

In particolare, le modifiche ai certificati nel key ring e all'attributo del repository delle chiavi diventano effettive in una delle seguenti occasioni:

- Quando l'iniziatore di canali viene avviato o riavviato.
- Quando viene immesso il comando REFRESH SECURITY TYPE (SSL) per aggiornare il contenuto del repository delle chiavi.

### **Creazione di un certificato personale autofirmato su z/OS**

Utilizzare questa procedura per creare un certificato personale autofirmato.

1. Genera un certificato e una coppia di chiave pubblica e privata utilizzando il seguente comando:

```
RACDCERT ID(userid2) GENCERT
SUBJECTSDN(CN('common-name')
            T('title')
            OU('organizational-unit')
            O('organization')
            L('locality')
            SP('state-or-province')
            C('country'))
WITHLABEL('label-name')
```

2. Collegare il certificato al proprio keyring utilizzando il seguente comando:

```
RACDCERT ID(userid1)
CONNECT(ID(userid2) LABEL('label-name') RING(ring-name) USAGE(PERSONAL))
```

dove:

- *userid1* è l'ID utente dello spazio di indirizzo dell'iniziatore di canali o il proprietario del keyring condiviso.
- *userid2* è l'ID utente associato al certificato e deve essere l'ID utente dello spazio di indirizzo dell'iniziatore di canali.  
*userid1* e *userid2* possono essere lo stesso ID.
- *ring - name* è il nome assegnato al keyring in [“Configurazione di un repository delle chiavi su z/OS”](#) a pagina 310.
- *nome - etichetta* deve essere il valore dell'attributo IBM MQ **CERTLABL** , se impostato, oppure il valore predefinito `ibmWebSphereMQ` con il nome del gestore code accodato. Per i dettagli, consultare [Etichetta certificato digitale](#) .

### **Richiesta di un certificato personale su z/OS**

Richiedere un certificato personale utilizzando RACF.

Per richiedere un certificato personale, utilizzare RACF nel modo seguente:

1. Creare un certificato personale autofirmato, come in [“Creazione di un certificato personale autofirmato su z/OS”](#) a pagina 313. Questo certificato fornisce alla richiesta i valori di attributo per il DN (Distinguished Name).
2. Crea una richiesta di certificato PKCS #10 Base64-encoded scritta in un dataset, utilizzando il seguente comando:

```
RACDCERT ID(userid2) GENREQ(LABEL(' label_name ')) DSN(' output_data_set_name ')
```

dove

- *userid2* è l'ID utente associato al certificato e deve essere l'ID utente dello spazio di indirizzo dell'iniziatore di canali
- *nome\_etichetta* è l'etichetta utilizzata durante la creazione del certificato autofirmato

Vedi [“Etichette dei certificati digitali, comprensione dei requisiti”](#) a pagina 26 per i dettagli.

3. Inviare il dataset a una CA (Certificate Authority) per richiedere un nuovo certificato personale.
4. Quando il certificato firmato viene restituito dall'autorità di certificazione, aggiungere nuovamente il certificato nel database RACF, utilizzando l'etichetta originale, come descritto in [“Aggiunta di certificati personali a un repository delle chiavi su z/OS”](#) a pagina 315.

### **Creazione di un certificato personale firmato RACF**

RACF può funzionare come un'autorità di certificazione ed emettere il proprio certificato CA.

Questa sezione utilizza il termine *certificato del firmatario* per indicare un certificato CA emesso da RACF.

La chiave privata per il certificato firmatario deve essere nel database RACF prima di eseguire la seguente procedura:

1. Utilizzare il seguente comando per generare un certificato personale firmato da RACF, utilizzando il certificato del firmatario contenuto nel database RACF :

```
RACDCERT ID(userid2) GENCERT
SUBJECTSDN(CN('common-name')
            T('title')
            OU('organizational-unit')
            O('organization')
            L('locality')
            SP('state-or-province')
            C('country'))
WITHLABEL('label-name')
SIGNWITH(CERTAUTH LABEL('signer-label'))
```

2. Collegare il certificato al proprio keyring utilizzando il seguente comando:

```
RACDCERT ID(userid1)
CONNECT(ID(userid2) LABEL('label-name') RING(ring-name) USAGE(PERSONAL))
```

dove:

- *userid1* è l'ID utente dello spazio di indirizzo dell'iniziatore di canali o il proprietario del keyring condiviso.
- *userid2* è l'ID utente associato al certificato e deve essere l'ID utente dello spazio di indirizzo dell'iniziatore di canali.  
*userid1* e *userid2* possono essere lo stesso ID.
- *ring - name* è il nome assegnato al keyring in [“Configurazione di un repository delle chiavi su z/OS”](#) a pagina 310.
- *nome - etichetta* deve essere il valore dell'attributo IBM MQ **CERTLABL**, se impostato, o il valore predefinito **ibmWebSphereMQ** con il nome del gestore code o del gruppo di condivisione code accordato. Per i dettagli, consultare [Etichetta certificato digitale](#).

- *signer - label* è l'etichetta del tuo certificato del firmatario.

## **Aggiunta di certificati personali a un repository delle chiavi su z/OS**

Utilizzare questa procedura per aggiungere o importare un certificato personale in un file di chiavi.

Dopo che l'autorità di certificazione ha inviato un nuovo certificato personale, aggiungerlo al keyring utilizzando la seguente procedura:

1. Aggiungere il certificato al database RACF utilizzando il seguente comando:

```
RACDCERT ID( userid2 ) ADD( input-data-set-name ) WITHLABEL( ' label-name ' )
```

2. Collegare il certificato al proprio keyring utilizzando il seguente comando:

```
RACDCERT ID( userid1 )  
CONNECT(ID( userid2 ) LABEL( ' label-name ' ) RING( ring-name ) USAGE(PERSONAL))
```

dove:

- *userid1* è l'ID utente dello spazio di indirizzo dell'iniziatore di canali o il proprietario del keyring condiviso.
- *userid2* è l'ID utente associato al certificato e deve essere l'ID utente dello spazio di indirizzo dell'iniziatore di canali.
- *ring - name* è il nome assegnato al keyring in [“Configurazione di un repository delle chiavi su z/OS”](#) a pagina 310.
- *input - data - set - name* è il nome del dataset contenente il certificato firmato CA. Il dataset deve essere catalogato e non deve essere un PDS o un membro di un PDS. Il formato record (RECFM) previsto da RACDCERT è VB. RACDCERT assegna e apre dinamicamente il dataset e legge il certificato da esso come dati binari.
- *nome - etichetta* è il nome dell'etichetta utilizzato quando hai creato la richiesta originale. Deve essere il valore dell'attributo IBM MQ **CERTLABEL** , se impostato, oppure il valore predefinito `ibmWebSphereMQ` con il nome del gestore code o del gruppo di condivisione code accodato. Per i dettagli, consultare [Etichetta certificato digitale](#) .

## **Esportazione di un certificato personale da un repository delle chiavi su z/OS**

Esportare il certificato utilizzando il comando RACDCERT.

Sul sistema da cui si desidera esportare il certificato, utilizzare il comando seguente:

```
RACDCERT ID(userid2) EXPORT(LABEL(' label-name '))  
DSN(output-data-set-name) FORMAT(CERTB64)
```

dove:

- *userid2* è l'ID utente con cui è stato aggiunto il certificato al keyring.
- *nome - etichetta* è l'etichetta del certificato che si desidera estrarre.
- *output - data - set - name* è il dataset in cui viene inserito il certificato.
- CERTB64 è un certificato X.509 codificato DER in formato Base64 . È possibile scegliere un formato alternativo, ad esempio:

### **CERTDER**

Certificato codificato DER X.509 in formato binario

### **PKCS12B64**

Certificato PKCS #12 in formato Base64

### **PKCS12DER**

Certificato PKCS #12 in formato binario

## **Eliminazione di un certificato personale da un repository delle chiavi su**

### **z/OS**

Eliminare un certificato personale utilizzando il comando RACDCERT.

Prima di eliminare un certificato personale, è possibile salvarne una copia. Per copiare il certificato personale in un dataset prima di eliminarlo, seguire la procedura in [“Esportazione di un certificato personale da un repository delle chiavi su z/OS”](#) a pagina 315. Utilizzare quindi il seguente comando per eliminare il certificato personale:

```
RACDCERT ID( userid2 ) DELETE(LABEL(' label-name '))
```

dove:

- *userid2* è l'ID utente con cui è stato aggiunto il certificato al keyring.
- *nome - etichetta* è il nome del certificato che si desidera eliminare.

## **Ridenominazione di un certificato personale in un repository delle chiavi su**

### **z/OS**

Ridenominare un certificato utilizzando il comando RACDCERT.

Se non si desidera che venga trovato un certificato con un'etichetta specifica, ma non si desidera eliminarlo, è possibile ridenominarlo temporaneamente utilizzando il seguente comando:

```
RACDCERT ID( userid2 ) LABEL(' label-name ') NEWLABEL(' new-label-name ')
```

dove:

- *userid2* è l'ID utente con cui è stato aggiunto il certificato al keyring.
- *nome - etichetta* è il nome del certificato che si desidera ridenominare.
- *nuovo nome - etichetta* è il nuovo nome del certificato.

Ciò può essere utile quando si verifica l'autenticazione client TLS.

## **Associazione di un ID utente con un certificato digitale su z/OS**

IBM MQ può utilizzare un ID utente associato a un certificato RACF come ID utente del canale. Associare un ID utente a un certificato installandolo con tale ID utente o utilizzando un filtro nome certificato.

Il metodo descritto in questo argomento è un'alternativa al metodo indipendente dalla piattaforma per associare un ID utente ad un certificato digitale, che utilizza i record di autenticazione di canale. Per ulteriori informazioni sui record di autenticazione di canale, consultare [“Record di autenticazione di canale”](#) a pagina 47.

Quando un'entità a un'estremità di un canale TLS riceve un certificato da una connessione remota, l'entità chiede a RACF se è presente un ID utente associato a tale certificato. L'entità utilizza tale ID utente come ID utente del canale. Se non vi è alcun ID utente associato al certificato, l'entità utilizza l'ID utente con cui è in esecuzione l'iniziatore di canali.

Associare un ID utente a un certificato in uno dei seguenti modi:

- Installare il certificato nel database di RACF con l'ID utente a cui si desidera associarlo, come descritto in [“Aggiunta di certificati personali a un repository delle chiavi su z/OS”](#) a pagina 315.
- Utilizzare un CNF (Certificate Name Filter) per associare il DN (Distinguished Name) dell'oggetto o dell'emittente del certificato all'ID utente, come descritto in [“Impostazione di un filtro nome certificato su z/OS”](#) a pagina 316.

## **Impostazione di un filtro nome certificato su z/OS**

Utilizzare il comando RACDCERT per definire un CNF (Certificate Name Filter), che associa un DN a un ID utente.

Effettuare le seguenti operazioni per impostare un CNF.

1. Abilitare le funzioni CNF utilizzando il seguente comando. Per eseguire questa operazione, è necessaria l'autorizzazione di aggiornamento sulla classe DIGTNMAP.

```
SETOPTS CLASSACT(DIGTNMAP) RACLIST(DIGTNMAP)
```

2. Definire il CNF. Ad esempio:

```
RACDCERT ID(USER1) MAP WITHLABEL('filter1') TRUST  
SDNFILTER('O=IBM.C=UK') IDNFILTER('O=ExampleCA.L=Internet')
```

dove USER1 è l'ID utente da utilizzare quando:

- Il DN del soggetto ha un'organizzazione di IBM e un paese di UK.
- Il DN dell'emittente ha un'organizzazione ExampleCA e una località Internet.

3. Aggiornare le associazioni CNF:

```
SETOPTS RACLIST(DIGTNMAP) REFRESH
```

#### Nota:

1. Se il certificato effettivo è memorizzato nel database RACF, l'ID utente con cui è installato viene utilizzato in luogo dell'ID utente associato a qualsiasi CNF. Se il certificato non è memorizzato nel database RACF, viene utilizzato l'ID utente associato al CNF corrispondente più specifico. Le corrispondenze del DN soggetto vengono considerate più specifiche delle corrispondenze del DN emittente.
2. Le modifiche ai CNFs non vengono applicate fino a quando non si aggiornano le associazioni CNF.
3. Un DN corrisponde al filtro DN in un CNF solo se il filtro DN è identico alla *parte meno significativa* del DN. La parte meno significativa del DN comprende gli attributi che di solito sono elencati all'estremità destra del DN, ma che appaiono all'inizio del certificato.

Ad esempio, si consideri SDNFILTER 'O=IBM.C=UK'. Un DN oggetto di 'CN=QM1.O=IBM.C=UK' corrisponde a tale filtro, ma un DN oggetto di 'CN=QM1.O=IBM.L=Hursley.C=UK' non corrisponde al filtro.

La parte meno significativa di alcuni certificati può contenere campi che non corrispondono al filtro DN. Considerare l'esclusione di tali certificati specificando un modello DN nel pattern SSLPEER nel comando DEFINE CHANNEL.

4. Se il CNF corrispondente più specifico è definito in RACF come NOTRUST, l'entità ... utilizza l'ID utente con cui è in esecuzione l'iniziatore di canali.
5. RACF utilizza il carattere ' .' come separatore. IBM MQ utilizza una virgola o un punto e virgola.

È possibile definire i CNFs per garantire che l'entità non imposti mai l'ID utente del canale sul valore predefinito, ovvero l'ID utente con cui è in esecuzione l'iniziatore del canale. Per ogni certificato CA nel key ring associato all'entità, definire un CNF con un IDNFILTER che corrisponda esattamente al DN oggetto di tale certificato CA. Ciò garantisce che tutti i certificati che l'entità potrebbe utilizzare corrispondano ad almeno uno di questi CNFs. Questo perché tutti questi certificati devono essere collegati al key ring associato all'entità o devono essere emessi da una CA per cui un certificato è collegato al key ring associato all'entità.

Fare riferimento al manuale *SecureWay Security Server RACF Security Administrator's Guide* per ulteriori informazioni sui comandi utilizzati per manipolare i CNFs.

## **Definizione di un canale mittente e di una coda di trasmissione su QMA su z/OS**

Utilizzare i comandi **DEFINE CHANNEL** e **DEFINE QLOCAL** per impostare gli oggetti richiesti.

## Procedura

Su QMA, immettere comandi come il seguente esempio:

```
DEFINE CHANNEL(TO.QMB) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QMB.MACH.COM) XMITQ(QMB)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA) DESCR('Sender channel using TLS from QMA to QMB')

DEFINE QLOCAL(QMB) USAGE(XMITQ)
```

## Risultati

Un canale mittente, TO.QMB, è una coda di trasmissione, QMB.

### **Definizione di un canale ricevente su QMB su z/OS**

Utilizzare il comando **DEFINE CHANNEL** per impostare l'oggetto richiesto.

## Procedura

Su QMB, immettere un comando simile al seguente:

```
DEFINE CHANNEL(TO.QMB) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using TLS to QMB')
```

## Risultati

Un canale ricevente, TO.QMB, viene creato.

### **Avvio del canale mittente su QMA su z/OS**

Se necessario, avviare un programma listener e aggiornare la sicurezza. Quindi avviare il canale utilizzando il comando **START CHANNEL**.

## Procedura

1. Opzionale: Se non è stato ancora fatto, avviare un programma listener su QMB.  
Il programma listener ascolta le richieste di rete in entrata e avvia il canale ricevente quando è necessario. Per informazioni su come avviare un listener, consultare [Avvio del listener del canale](#).
2. Opzionale: Se in precedenza sono stati eseguiti canali SSL/TLS, immettere il comando **REFRESH SECURITY TYPE(SSL)**.  
Ciò garantisce che tutte le modifiche apportate al repository delle chiavi siano disponibili.
3. Avviare il canale su QMA, utilizzando il comando **START CHANNEL(TO.QMB)**.

## Risultati

Il canale mittente è avviato.

### **Scambio di certificati autofirmati su z/OS**

Scambiare i certificati precedentemente estratti. Se si utilizza FTP, utilizzare il formato corretto.

## Procedura

Trasferire la parte CA del certificato QM1 al sistema QM2 e viceversa, ad esempio tramite FTP.

Se si trasferiscono i certificati utilizzando FTP, è necessario farlo nel formato corretto.

Trasferire i tipi di certificato seguenti in formato *binario* :

- Binario codificato DER X.509
- PKCS #7 (certificati CA)

- PKCS #12 (certificati personali)

Trasferire i tipi di certificato seguenti in formato ASCII:

- PEM (privacy - posta avanzata)
- Base64 codificato X.509

## **Definizione di un canale mittente e di una coda di trasmissione su QM1 su z/OS**

Utilizzare i comandi **DEFINE CHANNEL** e **DEFINE QLOCAL** per impostare gli oggetti richiesti.

### Procedura

In QM1, immettere comandi simili al seguente esempio:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QM1.MACH.COM) XMITQ(QM2)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA) DESCR('Sender channel using TLS from QM1 to QM2')

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

Le CipherSpecs ad ogni estremità del canale devono essere uguali.

Solo il parametro SSLCIPH è obbligatorio se vuoi che il tuo canale utilizzi TLS. Consultare [“CipherSpecs e CipherSuites in IBM MQ”](#) a pagina 38 per informazioni sui valori consentiti per il parametro SSLCIPH.

### Risultati

Un canale mittente, QM1.TO.QM2e una coda di trasmissione, QM2.

## **Definizione di un canale ricevente su QM2 su z/OS**

Utilizzare il comando **DEFINE CHANNEL** per impostare l'oggetto richiesto.

### Procedura

Su QM2, immettere un comando simile al seguente:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using TLS from QM1 to QM2')
```

Il canale deve avere lo stesso nome del canale mittente definito in [“Definizione di un canale mittente e di una coda di trasmissione su QM1 su z/OS”](#) a pagina 319e utilizzare lo stesso CipherSpec.

## **Avvio del canale mittente su QM1 su z/OS**

Se necessario, avviare un programma listener e aggiornare la sicurezza. Quindi avviare il canale utilizzando il comando **START CHANNEL**.

### Procedura

1. Opzionale: Se non è stato ancora fatto, avviare un programma listener su QM2.  
Il programma listener ascolta le richieste di rete in entrata e avvia il canale ricevente quando è necessario. Per informazioni su come avviare un listener, consultare [Avvio di un listener del canale](#)
2. Opzionale: Se i canali SSL/TLS sono stati eseguiti in precedenza, immettere il comando REFRESH SECURITY TYPE (SSL).  
Ciò garantisce che tutte le modifiche apportate al repository delle chiavi siano disponibili.
3. Su QM1, avviare il canale utilizzando il comando **START CHANNEL (QM1 . TO . QM2)**.

## Risultati

Il canale mittente è avviato.

### **Aggiornamento dell'ambiente SSL o TLS su z/OS**

Aggiornare l'ambiente TLS sul gestore code QMA utilizzando il comando **REFRESH SECURITY** .

## Procedura

Su QMA, immettere il seguente comando:

```
REFRESH SECURITY TYPE(SSL)
```

Ciò garantisce che tutte le modifiche apportate al repository delle chiavi siano disponibili.

### **Consentire connessioni anonime su un canale ricevente su z/OS**

Utilizzare il comando **ALTER CHANNEL** per rendere l'autenticazione client SSL o TLS facoltativa.

## Procedura

Su QMB, immettere il seguente comando:

```
ALTER CHANNEL(TO.QMB) CHLTYPE(RCVR) SSLCAUTH(OPTIONAL)
```

### **Avvio del canale mittente su QM1 su z/OS**

Se necessario, avviare l'iniziatore di canali, avviare un programma listener e aggiornare la sicurezza. Quindi avviare il canale utilizzando il comando **START CHANNEL** .

## Procedura

1. Opzionale: se non è stato ancora fatto, avviare l'iniziatore del canale.
2. Opzionale: Se non è stato ancora fatto, avviare un programma listener su QM2.  
Il programma listener ascolta le richieste di rete in entrata e avvia il canale ricevente quando è necessario. Per informazioni su come avviare un listener, consultare [Avvio di un listener del canale](#)
3. Opzionale: Se l'iniziatore di canali era già in esecuzione o qualsiasi canale SSL/TLS è stato eseguito precedentemente, immettere il comando **REFRESH SECURITY TYPE (SSL)**.  
Ciò garantisce che tutte le modifiche apportate al repository delle chiavi siano disponibili.
4. Su QM1, avviare il canale utilizzando il comando **START CHANNEL (QM1 . TO . QM2)** .

## Risultati

Il canale mittente è avviato.

### **Avvio del canale mittente su QMA su z/OS**

Se necessario, avviare l'iniziatore di canali, avviare un programma listener e aggiornare la sicurezza. Quindi avviare il canale utilizzando il comando **START CHANNEL** .

## Procedura

1. Opzionale: Se non è stato ancora fatto, avviare l'iniziatore di canali.
2. Opzionale: Se non è stato ancora fatto, avviare un programma listener su QMB.  
Il programma listener ascolta le richieste di rete in entrata e avvia il canale ricevente quando è necessario. Per informazioni su come avviare un listener, consultare [Avvio del listener del canale](#).
3. Opzionale: Se l'iniziatore di canali era già in esecuzione o se qualche canale SSL/TLS è stato eseguito in precedenza, immettere il comando **REFRESH SECURITY TYPE(SSL)** .



Ciò garantisce che tutte le modifiche apportate al repository delle chiavi siano disponibili.  
4. Avviare il canale su QMA, utilizzando il comando `START CHANNEL (TO . QMB)`.

## Risultati

Il canale mittente è avviato.

## Identificazione e autenticazione degli utenti

---

È possibile identificare e autenticare gli utenti utilizzando i certificati X.509, la struttura MQCSP o diversi tipi di programmi di uscita utente.

### Utilizzo di certificati X.509

Puoi identificare e autenticare gli utenti utilizzando i certificati x.509 con il comando **CHLAUTH** e il parametro **SSLPEER**. Il parametro **SSLPEER** specifica un filtro da utilizzare per il confronto con il DN (Distinguished Name) dell'oggetto del certificato dal gestore code peer o dal client sull'altra estremità del canale.

Per ulteriori informazioni sull'utilizzo del comando **CHLAUTH** e del parametro **SSLPEER**, consultare [SET CHLAUTH](#).

### Utilizzo della struttura MQCSP

Specificare la struttura dei parametri di sicurezza della connessione MQCSP su una chiamata MQCONN; 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.

**Avviso:** In alcuni casi, la password in una struttura MQCSP per un'applicazione client verrà inviata attraverso una rete in testo semplice. Per assicurarsi che le password dell'applicazione del client siano protette in modo appropriato, consultare ["Protezione password MQCSP"](#) a pagina 30.

## 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 di solito agisce per conto dell'utente dell'applicazione client IBM MQ. In questa situazione, l'autenticazione reciproca avviene effettivamente tra due gestori code o tra un gestore code e l'utente di una applicazione IBM MQ MQI client.

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 su Windows"](#) a pagina 145.

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.

TLS (Transport Layer Security) utilizza tecniche PKI come quelle appena descritte. Per ulteriori informazioni su come TLS esegue l'autenticazione, consultare [“Concetti di TLS \(Transport Layer Security\)” a pagina 15.](#)

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 429.](#) 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 108.](#)

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 327.](#)

## **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.

## PAM (Pluggable Authentication Method)

Linux → UNIX

PAM è ora comune nelle piattaforme UNIX and Linux e fornisce un meccanismo generale che nasconde i dettagli dell'autenticazione utente ai servizi.

Regole di autenticazione differenti possono essere utilizzate per servizi differenti, configurando le regole, senza alcuna modifica necessaria ai servizi stessi.

Consultare [“Utilizzo di PAM \(Pluggable Authentication Method\)”](#) a pagina 340 per ulteriori informazioni.


## Utenti privilegiati

Un utente privilegiato è un utente che dispone di autorizzazioni amministrative complete per IBM MQ.

Oltre agli utenti elencati nella seguente tabella, ci sono alcuni oggetti e autorizzazioni per i quali è necessario prestare particolare attenzione quando si concede l'accesso, per garantire l'integrità e sicurezza del gestore code. Il controllo supplementare deve essere applicato in caso di rilascio di una delle seguenti autorizzazioni:

- Tutte le autorizzazioni per gli oggetti SYSTEM
- Autorizzazioni di amministrazione come `+crt`, `+chg` e `+dlr`
- L'autorizzazione di amministrazione `+clr` per cancellare le code
- Le autorizzazioni di gestione `+ctrl` e `+ctrlx` consentono alle applicazioni di arrestare i canali, eseguire il backout o il commit dei messaggi
- L'autorizzazione MQI `+altusr` consente alle applicazioni di eseguire l'escalation dei privilegi per i controlli di autorizzazione
- Le autorizzazioni di contesto come `+setall` e `+setid` consentono alle applicazioni di modificare il contesto di sicurezza dei messaggi

Come principio generale, alle applicazioni di messaggistica devono essere concesse solo le autorizzazioni MQI di base per le code o gli argomenti necessari. I canali MCA che vengono eseguiti sotto un MCAUSER non privilegiato e alcuni altri tipi speciali di applicazioni, come i gestori di code di messaggi non recapitabili, possono richiedere ulteriori autorizzazioni non normalmente concesse alle applicazioni per funzionare correttamente.

Piattaforma	Utenti privilegiati
Sistemi Windows	<ul style="list-style-type: none"><li>• SYSTEM</li><li>• Membri del gruppo mqm</li><li>• Membri del gruppo Amministratori</li></ul>
Sistemi UNIX and Linux	<ul style="list-style-type: none"><li>• Membri del gruppo mqm</li></ul>
 Sistemi IBM i	<ul style="list-style-type: none"><li>• Profili qmqm e qmqmadm</li><li>• Tutti i membri del gruppo qmqmadm</li><li>• Qualsiasi utente definito con l'impostazione *ALLOBJ</li></ul>
z/OS	L'ID utente con cui vengono eseguiti l'iniziatore di canali, il gestore code e gli spazi di indirizzo di sicurezza dei messaggi avanzati.

## 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.

È possibile modificare MQCSP in un'uscita di sicurezza.

**Avviso:** In alcuni casi, la password in una struttura MQCSP per un'applicazione client verrà inviata attraverso una rete in testo semplice. Per assicurarsi che le password dell'applicazione del client siano protette in modo appropriato, consultare [“Protezione password MQCSP”](#) a pagina 30.

### Relazione tra MQCSP ed impostazioni AdoptCTX

IBM MQ autentica sempre le credenziali passate attraverso la struttura di MQCSP a meno che la funzione di autenticazione della connessione non sia abilitata. Una volta autenticate correttamente le credenziali, IBM MQ tenta di adottare l'ID utente per i controlli di autorizzazione futuri, a meno che ADOPTCTX non sia abilitato.

IBM MQ ha un limite sulla lunghezza degli ID utente che può utilizzare per i controlli di autorizzazione. Questi limiti sono descritti in dettaglio in [“ID utente”](#) a pagina 74. Quando si adotta un ID utente passato attraverso la struttura MQCSP, IBM MQ si comporta in modo diverso, a seconda di altre opzioni di configurazione:

- Quando si utilizza l'autenticazione della connessione LDAP, IBM MQ richiama il valore del campo impostato in SHORTUSR dal record LDAP dell'utente di tale utente e adotta tale ID utente.

Ad esempio, se SHORTUSR è impostato su 'CN' e un record LDAP elenca un utente come 'CN=Test,SN=MQ,O=IBM,C=UK', viene utilizzato l'ID utente Test.

- Quando si utilizza l'autenticazione della connessione SO o l'autenticazione PAM, se ADOPTCTX è YES, l'ID utente passato attraverso la struttura MQCSP viene troncato per soddisfare il limite di ID utente di 12 caratteri di IBM MQ quando viene adottato come contesto di connessione.

Se **Ch1AuthEarlyAdopt** è abilitato, il troncamento avviene dopo che le credenziali utente sono state autenticate.

Se **Ch1AuthEarlyAdopt** non è abilitato, il troncamento avviene prima dell'adozione. In Windows, se l'utente viene fornito nel formato user@domain, ciò significa che il troncamento può risultare in una specifica di dominio non valida quando l'utente ha meno di 12 caratteri.

Ad esempio, se un utente `ibmmq@windowsdomain` viene fornito tramite MQCSP, viene troncato a `ibmmq@window` in questo scenario. Ciò causa il seguente errore:

```
AMQ8074W: Autorizzazione non riuscita poiché SID 'SID' non corrisponde all'entità 'ibmmq@window'
```

Su questa base, se si passa un ID utente più lungo di 12 caratteri, come ad esempio un ID utente del dominio Windows nel modulo user@domain, tramite MQCSP è necessario configurare **Ch1AuthEarlyAdopt=Y** nel file qm.ini per evitare questo errore.

In alternativa, utilizzare ADOPTCTX (NO) sulla configurazione CONNAUTH AUTHINFO e utilizzare un approccio alternativo come una regola CHLAUTH USERMAP, un'exit di sicurezza o l'impostazione MCAUSER dell'oggetto canale per impostare l'ID utente per il canale.

## 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 di solito agisce per conto dell'utente dell'applicazione IBM MQ MQI client. In

questa situazione, l'autenticazione reciproca avviene effettivamente tra due gestori code o tra un gestore code e l'utente di una applicazione IBM MQ MQI client .

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 su Windows” a pagina 145.](#)

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.

TLS (Transport Layer Security) utilizza tecniche PKI come quelle appena descritte. Per ulteriori informazioni su come SSL (Secure Sockets Layer) esegue l'autenticazione, consultare [“Concetti di TLS \(Transport Layer Security\)” a pagina 15.](#)

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 429.](#) 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 108.](#)

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 327.](#)

## 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 TLS, 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).

Sulle seguenti piattaforme, il supporto SSL IBM MQ verifica la presenza di certificati revocati utilizzando OCSP (Online Certificate Status Protocol) o CRL e ARL sui server LDAP (Lightweight Directory Access Protocol). OCSP è il metodo preferito.

-  Linux
-  UNIX
-  Windows

IBM MQ classes for Java e IBM MQ classes for JMS non possono utilizzare le informazioni OCSP contenute in un file della tabella di definizione di canale client. Tuttavia, è possibile configurare OCSP come descritto in [Utilizzo di Online Certificate Protocol](#).

Sulle seguenti piattaforme e il supporto IBM MQ SSL verifica la presenza di certificati revocati utilizzando solo CRL e ARL sui server LDAP:

-  IBM i
-  z/OS

Per ulteriori informazioni sulle autorità di certificazione, consultare [“Certificati digitali” a pagina 9](#).

## controllo OCSP/CRL

Il controllo OCSP (Online Certificate Status Protocol) /CRL (Certificate Revocation List) viene eseguito sui certificati in entrata remoti. Il processo controlla l'intera catena coinvolta dal certificato personale del sistema remoto fino al suo certificato root.



## Utilizzo di openSSL per verificare la convalida OCSP

Se l'azienda utilizza openSSL per convalidare OCSP e poi si tenta di utilizzare una connessione TLS GSKit, si riceve un'avvertenza di stato SCNONSCIUTO.

Ciò è dovuto al fatto che tutti i certificati nella catena, tranne la root, vengono controllati da GSKit per lo stato di revoca. L'operazione GSKit è conforme alla RFC 5280 e ciò è descritto nella politica di sicurezza GSKit. L'algoritmo GSKit tenta tutte le origini disponibili per le informazioni di revoca, come descritto in RFC 5280 e nella politica di attendibilità GSKit.

## Come funziona il controllo OCSP/CRL in IBM MQ?

IBM MQ supporta due meccanismi per controllare il comportamento quando si controllano i certificati rispetto agli endpoint OCSP o CRL denominati, nell'estensione del certificato o, come definito negli oggetti AUTHINFO:

- Gli attributi **OCSPCheckExtensions**, **CDPCheckExtensionse** **OCSPAthentication** della sezione SSL del file `qm.inie`
- Utilizzo del parametro `SSLCRLNL` del gestore code e delle configurazioni AUTHINFO OCSP e CRLLDAP. Per ulteriori informazioni, consultare [ALTER AUTHINFO](#) e [ALTER QMGR](#).



### Attenzione:

Il comando ALTER AUTHINFO con **AUTHTYPE (OCSP)** non si applica per l'utilizzo su IBM i o z/OS gestori code. Tuttavia, può essere specificato su tali piattaforme per essere copiato nella CCDT (client channel definition table) per l'utilizzo da parte del client.

Gli attributi della stanza **OCSPCheckExtensions** e **CDPCheckExtensions** SSL controllano se IBM MQ verificherà un certificato rispetto al server OCSP o CRL dettagliato all'interno dell'estensione AIA del certificato.

Se non è abilitato, il server OCSP o CRL nell'estensione del certificato non viene contattato.

Se i server OCSP o CRL sono dettagliati attraverso gli oggetti AUTHINFO e a cui si fa riferimento utilizzando l'attributo `SSLCRLNL QMGR`, durante l'elaborazione della revoca dei certificati, IBM MQ tenta di contattare questi server.

**Importante:** Solo un oggetto OCSP AUTHINFO può essere definito nell'elenco nomi `SSLCRLNL`.

If:

**OCSPCheckExtensions= NO** e **CDPCheckExtensions=NO** sono impostati e  
Nessun server OCSP o CRL definito negli oggetti AUTHINFO

non viene eseguito alcun controllo di revoca del certificato.

Quando si verifica un certificato per il suo stato di revoca, IBM MQ contatta i server OCSP o CRL indicati nel seguente ordine, se abilitato:

1. Il server OCSP dettagliato in un oggetto **AUTHTYPE (OCSP)** e a cui si fa riferimento nell'attributo `SSLCRLNL QMGR`.
2. I server OCSP dettagliati nell'estensione AIA dei certificati, se **OCSPCheckExtensions=YES**.
3. Server CRL dettagliati nell'estensione **CRLDistributionPoints** dei certificati, se **CDPCheckExtensions =YES**.
4. Qualsiasi server CRL descritto in dettaglio negli oggetti **AUTHINFO (CRLLDAP)** e a cui si fa riferimento nell'attributo `SSLCRLNL QMGR`.

Durante la verifica di un certificato, se un passo risulta nel server OCSP o CRL che restituisce una risposta REVOKED o VALID definitiva a una query per il certificato, non vengono eseguiti ulteriori controlli e lo stato del certificato come presentato viene utilizzato per stabilire se considerarlo attendibile o meno.

Se un server OCSP o un server CRL restituisce un risultato di UNKNOWN, l'elaborazione continua fino a quando un server OCSP o CRL restituisce un risultato definitivo o tutte le opzioni sono esaurite.

Il comportamento che indica se un certificato viene considerato revocato, se non è possibile determinarne lo stato, è diverso per i server OCSP e CRL:

- Per i server CRL, se non è possibile ottenere alcun CRL, il certificato viene considerato NOT\_REVOKED
- Per i server OCSP, se non è possibile ottenere uno stato di revoca da un server OCSP denominato, il comportamento viene controllato tramite l'attributo **OCSPAuthentication** nella stanza SSL del file `qm.ini`.

È possibile configurare questo attributo per bloccare una connessione, consentire una connessione o consentire una connessione con un messaggio di avvertenza.

È possibile utilizzare l'attributo **SSLHTTPProxyName=string** nella stanza SSL dei file `qm.ini` e `mqclient.ini` per le verifiche OCSP, se necessario. La stringa è il nome host o l'indirizzo di rete del server proxy HTTP che deve essere utilizzato da GSKit per le verifiche OCSP.

## **OCSP e certificati revocati**

IBM 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 IBM MQ su sistemi UNIX, Linux, and Windows .

Per verificare lo stato di revoca di un certificato digitale utilizzando OCSP, IBM 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 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 MQ non è in grado di accedere al responder OCSP.
- Il responder OCSP ha inviato una risposta, ma IBM MQ non può verificare la firma digitale della risposta.
- Il responder OCSP ha inviato una risposta che indica che non dispone di dati di revoca per il certificato.

Se IBM MQ riceve un esito OCSP di Sconosciuto, il suo comportamento dipende dall'impostazione dell'attributo `OCSPAuthentication`. Per i gestori code, questo attributo è contenuto in una delle seguenti posizioni:

-   Nella stanza SSL del file `qm.ini` su UNIX and Linux.
-  Nel registro Windows .

Questo attributo può essere impostato utilizzando IBM MQ Explorer. Per i client, l'attributo è contenuto nella stanza SSL del file di configurazione client.

Se viene ricevuto un esito di `Sconosciuto` e `OCSPAuthentication` è impostato su `REQUIRED` (il valore predefinito), IBM MQ rifiuta la connessione e genera 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 `MQRQ_CHANNEL_SSL_ERROR` con `ReasonQualifier` impostato su `MQRQ_SSL_HANDSHAKE_ERROR`.

Se viene ricevuto un esito di `Sconosciuto` e `OCSPAuthentication` è impostato su `OPTIONAL`, IBM MQ consente l'avvio del canale e non vengono generati messaggi di evento SSL né vengono generate avvertenze.

Se viene ricevuto un esito di `Sconosciuto` e `OCSPAuthentication` è impostato su `WARN`, viene avviato il canale SSL ma IBM 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 `MQRQ_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 alcun certificato aggiuntivo; i passi già intrapresi per stabilire la connettività TLS 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 lo ha firmato (e tale certificato è firmato da una CA già attendibile per la connettività TLS), non è richiesta alcuna ulteriore configurazione del 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 risponditore OCSP al database delle chiavi del client o del gestore code che esegue il controllo OCSP . 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, IBM 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 MQ, a seconda del valore di `OCSPAuthentication`.

## OCSP (Online Certificate Status Protocol) nelle applicazioni Java e JMS client

A causa di una limitazione dell'API Java , IBM MQ può utilizzare il controllo della revoca dei certificati OCSP (Online Certificate Status Protocol) per i socket protetti TLS solo quando OCSP è abilitato per il 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 `java.security.Security.setProperty()` API, soggetta a qualsiasi politica Java Security Manager in vigore.

Come minimo, è necessario specificare uno dei valori `ocsp.enable` e `ocsp.responderURL`.

Nome proprietà	Descrizione
<code>ocsp.enable</code>	Il valore di questa proprietà è <code>true</code> o <code>false</code> . Se è <code>true</code> , la verifica OCSP viene abilitata quando si esegue il controllo sulla revoca dei certificati; se è <code>false</code> o non è impostato, la verifica OCSP è disabilitata.
<code>ocsp.responderURL</code>	Il valore di questa proprietà è un URL che identifica l'ubicazione del responder OCSP. Di seguito è riportato un esempio: <code>ocsp.responderURL=http://ocsp.example.net:80</code> . 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.
<code>ocsp.responderCertSubjectName</code>	Il valore di questa proprietà è il nome oggetto del certificato del responder OCSP. Di seguito è riportato un esempio: <code>ocsp.responderCertSubjectName="CN=OCSP Responder, O=XYZ Corp"</code> . 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 utilizzare entrambe le proprietà <code>ocsp.responderCertIssuerName</code> e <code>ocsp.responderCertSerialNumber</code> . Quando questa proprietà è impostata, le proprietà <code>ocsp.responderCertIssuerName</code> e <code>ocsp.responderCertSerialNumber</code> vengono ignorate.
<code>ocsp.responderCertIssuerName</code>	Il valore di questa proprietà è il nome emittente del certificato del responder OCSP. Di seguito è riportato un esempio: <code>ocsp.responderCertIssuerName="CN=Enterprise CA, O=XYZ Corp"</code> . 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 <code>ocsp.responderCertSerialNumber</code> . Questa proprietà viene ignorata quando è impostata la proprietà <code>ocsp.responderCertSubjectName</code> .
<code>ocsp.responderCertSerialNumber</code>	Il valore di questa proprietà è il numero di serie del certificato del responder OCSP. Di seguito è riportato un esempio: <code>ocsp.responderCertSerialNumber=2A:FF:00</code> . 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

Nome proprietà	Descrizione
	proprietà è impostata, è necessario impostare anche la proprietà <code>ocsp.responderCertIssuerName</code> . Questa proprietà viene ignorata quando è impostata la proprietà <code>ocsp.responderCertSubjectName</code> .

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 potrebbe avere effetti collaterali indesiderati quando la JVM viene condivisa con un altro codice dell'applicazione che utilizza socket protetti 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 non 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'AMS Java Interceptor descritto in [Abilitazione del controllo OCSP negli intercettatori nativi](#), evitare di utilizzare una configurazione OCSP `java.security` in conflitto 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 IBM MQ per CRL e ARL varia in base alla 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.

IBM MQ gestisce una cache di CRL e ARL a cui è stato eseguito l'accesso nelle 12 ore precedenti.

Quando un gestore code o IBM MQ MQI client riceve un certificato, controlla il CRL per confermare che il certificato è ancora valido. IBM MQ esegue prima il check-in della cache, se è presente una cache. Se il CRL non si trova nella cache, IBM 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 `SSLCRLNL`, finché IBM 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.

### 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 `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'utilizzo con IBM MQ TLS, 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 20 a pagina 334 mostra un file LDIF di esempio che è possibile creare come input per il 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 all'interno di IBM.

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

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

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

Figura 20. File LDIF di esempio per una CA (Certificate Authority). Ciò può variare da implementazione a implementazione.

Figura 21 a pagina 334 mostra la struttura DIT creata dal server LDAP quando si carica il file LDIF di esempio mostrato in Figura 20 a pagina 334 insieme a un file simile per CA2, una CA immaginaria impostata dall'organizzazione PKI, anche all'interno di IBM.

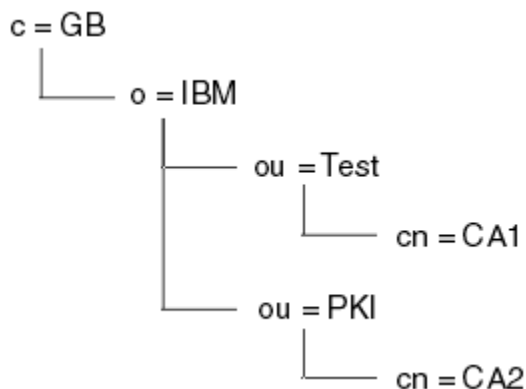


Figura 21. Esempio di una struttura ad albero di informazioni della directory LDAP

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 334.

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. **IBM i** IBM MQ su IBM i si comporta in modo diverso da altre piattaforme.

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 `SSLCRLNL`.

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`. **IBM i** Su IBM i, è anche possibile utilizzare il comando `CL CRTMQMAUTI`

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.

**z/OS** Inoltre, solo su z/OS, è necessario accedere a tutti i server LDAP utilizzando gli stessi ID utente e password. L'ID utente e la password utilizzati sono quelli specificati nel primo oggetto `AUTHINFO` nell'elenco nomi.

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. **z/OS** Su z/OS, assicurarsi che l'attributo dell'elenco nomi `NLTYPE` sia impostato su `AUTHINFO`.

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 `SSLCRLNL`. Il valore iniziale del gestore code per questo attributo è vuoto.

**IBM i** Su IBM i, è possibile specificare gli oggetti delle informazioni di autenticazione, ma il gestore code non utilizza né gli oggetti delle informazioni di autenticazione né un elenco nomi degli oggetti delle informazioni di autenticazione. Solo i IBM MQ client che utilizzano una tabella di connessioni client generata da un gestore code IBM i utilizzano le informazioni di autenticazione specificate per tale gestore code IBM i. L'attributo del gestore code `SSLCRLNL` su IBM i determina le informazioni di autenticazione

utilizzate da tali client. Consultare [“Accesso a CRL e ARL su IBM i”](#) a pagina 336 per informazioni su come indicare a un gestore code IBM i come accedere ai CRL.

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. Notare che i server LDAP devono contenere informazioni identiche.

#### **IBM i** Accesso a CRL e ARL su IBM i

Utilizzare questa procedura per accedere ai CRL o agli ARL su IBM i.

Si noti che in questa sezione, le informazioni sui CRL (Certificate Revocation Lists) si applicano anche agli ARL (Authority Revocation Lists).

Seguire questa procedura per impostare un'ubicazione CRL per un certificato specifico su IBM i:

1. Accedere all'interfaccia DCM, come descritto in [“Accesso a DCM”](#) a pagina 266.
2. Nella categoria attività **Gestisci ubicazioni CRL** del pannello di navigazione, fare clic su **Aggiungi ubicazione CRL**. La pagina Gestisci ubicazioni CRL viene visualizzata nel frame di attività.
3. Nel campo **Nome ubicazione CRL**, immettere un nome ubicazione CRL, ad esempio LDAP Server #1
4. Nel campo **Server LDAP**, immettere il nome server LDAP.
5. Nel campo **Utilizza SSL (Secure Sockets Layer)**, selezionare **Sì** se si desidera connettersi al server LDAP utilizzando TLS. Altrimenti, selezionare **No**.
6. Nel campo **Numero porta**, immettere un numero di porta per il server LDAP, ad esempio 389.
7. Se il server LDAP non consente agli utenti anonimi di interrogare la directory, immettere un DN (distinguished name) di login per il server nel campo **DN (distinguished name) di login**.
8. Fare clic su **OK**. DCM informa che è stata creata l'ubicazione CRL.
9. Nel pannello di navigazione, fare clic su **Seleziona un archivio certificati**. La pagina Seleziona una memorizzazione certificato viene visualizzata nel frame delle attività.
10. Selezionare la casella di spunta **Altra memorizzazione certificato di sistema** e fare clic su **Continua**. Viene visualizzata la pagina Archivio certificati e password.
11. Nel campo **Percorso archivio certificati e nome file**, immettere il nome file e il percorso IFS impostati quando [“Creazione di un archivio certificati su IBM i”](#) a pagina 267.
12. Immettere una password nel campo **Password archivio certificati**. Fare clic su **Continua**. La pagina Archivio certificati corrente viene visualizzata nel frame di attività.
13. Nella categoria attività **Gestisci certificati** nel pannello di navigazione, fare clic su **Aggiorna assegnazione ubicazione CRL**. La pagina Assegnazione collocazione CRL viene visualizzata nella cornice dell'attività.
14. Selezionare il pulsante di opzione per il certificato CA a cui si desidera assegnare l'ubicazione CRL. Cliccare su **Aggiorna assegnazione collocazione CRL**. La pagina Aggiorna assegnazione ubicazione CRL viene visualizzata nel frame delle attività.
15. Selezionare il pulsante di opzione per l'ubicazione CRL che si desidera assegnare al certificato. Fare clic su **Aggiorna assegnazione**. DCM informa che ha aggiornato l'assegnazione.

Tenere presente che DCM consente di assegnare un server LDAP differente mediante l'autorità di certificazione.

#### *Accesso a CRL e ARL utilizzando IBM MQ Explorer*

È possibile utilizzare IBM 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 pulsante destro del mouse sulla cartella Elenchi nomi e fare clic su **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 337](#)) 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 337](#)) nel campo **Elenco nomi CRL** .

### **Accesso a CRL e ARL con un IBM MQ MQI client**

Sono disponibili tre opzioni per specificare i server LDAP che contengono i CRL per il controllo da parte di un IBM MQ MQI client.

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. Notare che i server LDAP devono contenere informazioni identiche.

Non è possibile accedere ai CRL LDAP da un canale IBM MQ MQI client in esecuzione su Linux (piattaforma zSeries).

#### *Ubicazione di un responder OCSP e dei server LDAP che contengono i CRL*

Su un sistema IBM MQ MQI client, 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.

 (Per IBM i, vedere [Accesso ai CRL e agli ARL su IBM i](#).)

### **Quando un'applicazione IBM MQ MQI client 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 **MQCONN**, 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 da IBM MQ MQI client 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 IBM MQ MQI client 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 su 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 IBM MQ MQI client 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:

### Su piattaforme server AIX, HP-UX, Linux, IBM i, 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, **SSLCRLNL**, 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 IBM MQ MQI client 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, **SSLCRLNL**, 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 IBM MQ MQI client su tale sistema può utilizzare l'ubicazione di revoca del certificato nella CCDT per accedere a un responder OCSP o a 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, **SSLCRLNL**, 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.

4. Sulla macchina client, è possibile utilizzare il comando **runmqsc** con il parametro **-n**.

#### **Sulle piattaforme client AIX, HP-UX, Linux, IBM i, Solarise Windows**

È possibile creare una CCDT sulla macchina del client utilizzando il comando **runmqsc** con il parametro **-n** e gli oggetti **DEFINE AUTHINFO** nel file CCDT. L'ordine in cui gli oggetti vengono definiti è l'ordine in cui vengono utilizzati nel file. Qualsiasi nome che è possibile utilizzare in un oggetto **DEFINE AUTHINFO** non viene conservato nel file. Vengono utilizzati solo numeri posizionali quando si **DISPLAY** gli **AUTHINFO** oggetti in un file CCDT.

**Nota:** Se si specifica il parametro **-n**, non è necessario specificare alcun altro parametro.

#### **Utilizzo di Active Directory su Windows**

Sui 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 IBM MQ classes for Java e IBM MQ classes for JMS**

IBM MQ classes for Java e IBM MQ classes for JMS accedono ai CRL in modo diverso rispetto ad altre piattaforme.

Per informazioni sull'utilizzo di CRL e ARL con IBM MQ classes for Java, consultare [Utilizzo degli elenchi di revoca dei certificati](#)

Per informazioni sull'utilizzo di CRL e ARL con IBM MQ classes for JMS, consultare [Proprietà dell'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 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 possibile utilizzare anche IBM MQ Explorer.

## Utilizzo di PAM (Pluggable Authentication Method)

È possibile utilizzare PAM solo su piattaforme UNIX and Linux . Un tipico sistema UNIX dispone di moduli PAM che implementano il meccanismo di autenticazione tradizionale; tuttavia, potrebbe essere disponibile di più. Oltre all'attività di base di convalida delle password, è possibile richiamare i moduli PAM per eseguire ulteriori regole.

I file di configurazione definiscono il metodo di autenticazione da utilizzare per ciascuna applicazione. Le applicazioni di esempio includono il login terminale standard, ftp e telnet.

Il vantaggio di PAM è che l'applicazione non ha bisogno di conoscere o preoccuparsi di come l'ID utente viene autenticato. Fino a quando l'applicazione può fornire una forma corretta di dati di autenticazione a PAM, il meccanismo dietro di esso è trasparente.

Il formato dei dati di autenticazione dipende dal sistema utilizzato. Ad esempio, IBM MQ ottiene una parola d'ordine tramite parametri, come la struttura `MQCSP` utilizzata nella chiamata API `MQCONN` .

**Importante:** Non è possibile impostare l'attributo `AUTHENMD` fino a quando non si installa IBM MQ 8.0.0 Fix Pack 3e quindi riavviare il gestore code, utilizzando un livello `-e CMDLEVEL=` di `802` (sul comando `strmqm` ) per impostare il livello di comando richiesto.

### Configurazione del sistema per l'utilizzo di PAM

Il nome servizio utilizzato da IBM MQ, quando si richiama PAM, è `ibmmq`.

Tenere presente che un'installazione di IBM MQ tenta di mantenere una configurazione PAM predefinita, che consente le connessioni dagli utenti del sistema operativo, in base ai valori predefiniti noti per i diversi sistemi operativi.

Tuttavia, l'amministratore di sistema deve verificare che le regole definite nei file `/etc/pam.conf` `/etc/pam.d/ibmq` siano ancora appropriate.

## 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.

### ULW

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.

Questa sezione contiene anche un elenco di controllo che è possibile utilizzare per determinare quali attività eseguire per applicare la sicurezza al proprio sistema su tutte le piattaforme e considerazioni per concedere agli utenti l'autorità di gestire IBM MQ e gestire gli oggetti IBM MQ .

Se i meccanismi di sicurezza forniti non soddisfano le proprie esigenze, è possibile sviluppare i propri programmi di uscita canale.

### ULW

## Controllo dell'accesso agli oggetti utilizzando OAM su UNIX, Linux, and Windows

OAM (object authority manager) fornisce un'interfaccia di comando per concedere e revocare l'autorizzazione agli oggetti IBM MQ .

È necessario essere autorizzati a utilizzare questi comandi, come descritto in [“Autorizzazione per gestire IBM MQ su UNIX, Linux, and Windows”](#) a pagina 391. Gli ID utente che sono autorizzati a gestire IBM 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.

Da IBM MQ 8.0, su sistemi UNIX and Linux, OAM (Object Authority Manager) può utilizzare l'autorizzazione basata sull'utente e l'autorizzazione basata sul gruppo.

Prima di IBM MQ 8.0, gli ACL (access control list) su UNIX and Linux si basano solo sui gruppi. Da IBM MQ 8.0, gli ACL si basano sia sugli ID utente che sui gruppi ed è possibile utilizzare il modello basato sull'utente o il modello basato sul gruppo per l'autorizzazione impostando l'attributo **SecurityPolicy** sul valore appropriato come descritto in [Configurazione dei servizi installabili](#) e [Configurazione delle stanze del servizio di autorizzazione su UNIX e Linux](#).

## Modifiche nel comportamento per IBM MQ 8.0 e versioni successive

Da IBM MQ 8.0, quando si esegue con la politica basata sull'utente, alcuni comandi restituiscono informazioni differenti dalle precedenti versioni del prodotto:

- I comandi **dmpmqaut** e **dmpmqcfig** mostrano i record basati sull'utente, come le operazioni PCF equivalenti.
- Il plug-in OAM per IBM MQ Explorer mostra i record basati sull'utente e consente le modifiche basate sull'utente.
- La funzione OAM **Inquire** restituisce risultati che mostrano che è compatibile con l'utente.

Utilizzando l'attributo **-p** nel comando **setmqaut** non si concede l'accesso a tutti gli utenti dello stesso gruppo primario, quando le autorizzazioni basate sugli utenti sono abilitate nel file `qm.ini` come descritto nel documento [Formato della stanza del servizio](#).

Se si inizia a utilizzare l'autorizzazione basata sull'utente e si dispone di molti utenti, probabilmente ci saranno più record memorizzati nella coda AUTH che con il modello basato sul gruppo e il processo di autorizzazione potrebbe richiedere un po' più di tempo rispetto al passato poiché ci sono più record da verificare. Questo aumento non dovrebbe essere significativo. Se necessario, è possibile utilizzare una combinazione di autorizzazioni utente e gruppo.

## Considerazioni sulla migrazione

Se si modifica il modello da gruppo a utente per un gestore code esistente, non vi è alcun effetto immediato. Le autorizzazioni che sono già state effettuate continuano ad essere applicate. Qualsiasi utente che si connette al gestore code riceve gli stessi privilegi di prima: la combinazione di tutti i gruppi a cui appartiene il proprio ID. Quando vengono emessi nuovi comandi **setmqaut** per gli ID utente, essi hanno effetto immediato.

Se si crea un nuovo gestore code con la politica utente, questo gestore code dispone delle autorizzazioni solo per l'utente che lo crea (che normalmente, ma non necessariamente, è l'ID utente `mqm`). Esistono anche autorizzazioni che vengono concesse automaticamente al gruppo `mqm`. Tuttavia, se non si dispone di `mqm` come gruppo primario, il gruppo `mqm` non viene incluso nella serie iniziale di autorizzazioni.

Se si passa da un criterio utente a un criterio di gruppo, le autorizzazioni basate sull'utente non vengono eliminate automaticamente. Tuttavia, non vengono più utilizzati durante il controllo delle autorizzazioni. Prima di ripristinare la politica, salvare la configurazione corrente, modificare la politica, riavviare il gestore code e ripetere lo script. Poiché ora è un gestore code basato su un gruppo, l'effetto è che le regole ID utente vengono memorizzate in base al gruppo primario.

### Informazioni correlate

[object authority manager \(OAM\)](#)

[Principal e gruppi su UNIX, Linux e Windows](#)

[File di configurazione del gestore code: formato stanza del servizio](#)  
comando **crtmqm** (create queue manager)

Utilizzare il comando di controllo **setmqaut**, il comando **SET AUTHREC** MQSC o il comando PCF **MQCMD\_SET\_AUTH\_REC** per concedere agli utenti e ai gruppi di utenti l'accesso agli oggetti IBM MQ. Notare che in IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC**.

Per una definizione completa del comando di controllo **setmqaut** e della sua sintassi, consultare [setmqaut](#).

Per una definizione completa del comando MQSC **SET AUTHREC** e della relativa sintassi, consultare [SET AUTHREC](#).

Per una definizione completa del comando PCF **MQCMD\_SET\_AUTH\_REC** e della sua sintassi, consultare [Impostazione record di autorizzazione](#).

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*; 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 UNIX, Linux, and Windows"](#) a pagina 343.
- 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 UNIX and Linux , questo comando revoca anche l'autorizzazione put 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 setmqaut 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.

### Note sull'utilizzo di SET AUTHREC

L'elenco di autorizzazioni da aggiungere e l'elenco di autorizzazioni da rimuovere non devono sovrapporsi. Ad esempio, non è possibile aggiungere l'autorizzazione di visualizzazione e rimuovere l'autorizzazione di visualizzazione con lo stesso comando. Questa regola si applica anche se le autorizzazioni vengono espresse utilizzando opzioni diverse. Ad esempio, il seguente comando non riesce perché l'autorizzazione DSP si sovrappone all'autorizzazione ALLADM:

```
SET AUTHREC PROFILE(*) OBJTYPE(Queue) PRINCIPAL(PRINC01) AUTHADD(DSP) AUTHRMV(ALLADM)
```

L'eccezione a questo comportamento di sovrapposizione avviene con l'autorizzazione ALL. Il seguente comando aggiunge prima le autorizzazioni ALL e quindi rimuove l'autorizzazione SETID:

```
SET AUTHREC PROFILE(*) OBJTYPE(Queue) PRINCIPAL(PRINC01) AUTHADD(ALL) AUTHRMV(SETID)
```

Il seguente comando rimuove prima le autorizzazioni ALL e quindi aggiunge l'autorizzazione DSP:

```
SET AUTHREC PROFILE(*) OBJTYPE(Queue) PRINCIPAL(PRINC01) AUTHADD(DSP) AUTHRMV(ALL)
```

Indipendentemente dall'ordine in cui vengono fornite nel comando, le autorizzazioni ALL vengono elaborate per prime.

## Utilizzo di profili generici OAM su UNIX, Linux, and Windows

Utilizzare profili generici OAM per impostare, in una singola operazione, i privilegi di un utente per molti oggetti; invece di dover immettere comandi **setmqaut** separati o comandi **SET AUTHREC**, per ogni singolo oggetto quando viene creato. Notare che in IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC**.

L'utilizzo di profili generici nei comandi **setmqaut** o **SET AUTHREC** 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 . 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 applicano 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 caratteri jolly sui sistemi UNIX e Linux , **è necessario** racchiudere il nome 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. In questo esempio, 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 la relativa sintassi, consultare [dmpmqaut](#).

Per una definizione completa del comando MQSC **DISPLAY AUTHREC** e della relativa sintassi, consultare [DISPLAY AUTHREC](#).

Per una definizione completa del comando PCF **MQCMD\_INQUIRE\_AUTH\_RECS** e la relativa sintassi, consultare [Richiedi record di autorizzazione](#).

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:** Sebbene gli utenti su UNIX e Linux possano utilizzare l'opzione -p per il comando **dmpmqaut**, devono utilizzare -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:** Solo per IBM MQ for Windows , tutti i principal visualizzati includono informazioni sul dominio, ad esempio:

```
profile:      a.b.*
object type:  queue
entity:       user1@domain1
type:         principal
authority:    get, browse, put, inq
```

**ULW**

### **Utilizzo di caratteri jolly nei profili OAM su UNIX, Linux, and Windows**

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 . JKL e ABC . GHI . JKL. (Si noti che **non** si applicano 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 caratteri jolly sui sistemi UNIX and Linux , è **necessario** racchiudere il nome profilo tra virgolette singole.

## **Priorità del profilo su UNIX, Linux, and Windows**

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. In questo esempio, 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. \*\*

Consultare [SET AUTHREC](#) per le informazioni equivalenti quando si utilizza questo comando MQSC.

**Dump delle impostazioni del profilo su UNIX, Linux, and Windows**

Utilizzare il comando di controllo **dmpmqaut**, il comando MQSC **DISPLAY AUTHREC** o il comando PCF **MQCMD\_INQUIRE\_AUTH\_RECS** per eseguire il dump delle autorizzazioni correnti associate ad un profilo specificato. Notare che in IBM MQ Appliance è possibile utilizzare solo il comando **DISPLAY AUTHREC**.

Per una definizione completa del comando di controllo **dmpmqaut** e la relativa sintassi, consultare [dmpmqaut](#).

Per una definizione completa del comando MQSC **DISPLAY AUTHREC** e della relativa sintassi, consultare [DISPLAY AUTHREC](#).

Per una definizione completa del comando PCF **MQCMD\_INQUIRE\_AUTH\_RECS** e la relativa sintassi, consultare [Richiedi record di autorizzazione](#).

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:** Solo per IBM MQ for Windows , tutti i principal visualizzati includono 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 su UNIX, Linux, and Windows

Utilizzare il comando di controllo **dspmqaut** , il comando MQSC **DISPLAY AUTHREC** o il comando PCF **MQCMD\_INQUIRE\_ENTITY\_AUTH** per visualizzare le autorizzazioni di un determinato principal o gruppo per un determinato oggetto. Notare che in IBM MQ Appliance è possibile utilizzare solo il comando **DISPLAY AUTHREC** .

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 [dmpmqaut](#).

Per una definizione completa del comando MQSC **DISPLAY AUTHREC** e della relativa sintassi, consultare [DISPLAY AUTHREC](#).

Per una definizione completa del comando PCF **MQCMD\_INQUIRE\_AUTH\_RECS** e la relativa sintassi, consultare [Richiedi record di autorizzazione](#).

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
```

## **ULW** Modifica e revoca dell'accesso a un oggetto IBM MQ su UNIX, Linux, and Windows

Per modificare il livello di accesso di un utente o di un gruppo a un oggetto, utilizzare il comando di controllo **setmqaut**, il comando MQSC **DELETE AUTHREC** o il comando PCF **MQCMD\_DELETE\_AUTH\_REC**. Notare che in IBM MQ Appliance è possibile utilizzare solo il comando **DELETE AUTHREC**.

Il processo di rimozione dell'utente da un gruppo è descritto in:

- [“Creazione e gestione di gruppi su Windows” a pagina 137](#)
- [“Creazione e gestione di gruppi su HP-UX” a pagina 134](#)
- [“Creazione e gestione di gruppi su AIX” a pagina 133](#)
- [“Creazione e gestione di gruppi su Solaris” a pagina 136](#)
- [“Creazione e gestione di gruppi su Linux” a pagina 136](#)

All'ID utente che crea un oggetto IBM 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 sui 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 relativa sintassi, consultare [setmqaut](#).

Per una definizione completa del comando MQSC **DELETE AUTHREC** e della sua sintassi, consultare [DELETE AUTHREC](#).

Per una definizione completa del comando PCF **MQCMD\_DELETE\_AUTH\_REC** e della sua sintassi, consultare [Cancellazione record di autorizzazioni](#).

Su Windows, da IBM MQ 8.0, è possibile eliminare le voci OAM corrispondenti ad un particolare account utente Windows in qualsiasi momento utilizzando il parametro **-u SID** di **setmqaut**.

Prima di IBM MQ 8.0, era necessario eliminare le voci OAM corrispondenti ad uno specifico account utente Windows prima di eliminare il profilo utente. Non è possibile rimuovere le voci OAM dopo aver rimosso l'account utente.

## **ULW** Come impedire i test di accesso di sicurezza sui sistemi UNIX, Linux, and Windows

Per disattivare tutti i controlli di sicurezza, è possibile disabilitare OAM (object authority manager). 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. Consultare [Variabili di ambiente](#) per informazioni sulle implicazioni dell'impostazione della variabile MQSNOAUT e su come impostare MQSNOAUT su Windows e UNIX.
- Modificare il file di configurazione del gestore code per rimuovere il servizio.

Se si utilizza il comando **setmqaut**, o **dspmqaut** mentre OAM è disabilitato, tenere presente quanto segue:

- 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.

Quando un OAM viene rimosso, non può essere reinserito in un gestore code esistente. Questo perché l'OAM deve essere presente al momento della creazione dell'oggetto. Per utilizzare nuovamente IBM MQ OAM dopo che è stato rimosso, rigenerare il gestore code.

### Informazioni correlate

[Servizi e componenti installabili per Unix, Linux e Windows](#)

[Configurazione dei servizi installabili](#)

[Informazioni di riferimento sui servizi installabili](#)

## Concessione dell'accesso richiesto alle risorse

Utilizzare questo argomento per determinare quali attività eseguire per applicare la sicurezza al proprio sistema IBM MQ su UNIX, Linux, Windows, IBM e z/OS.

### Informazioni su questa attività

Durante questa attività, si decide quali azioni sono necessarie per applicare il livello appropriato di sicurezza agli elementi dell'installazione di IBM 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 352.
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 361.
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 368.

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 370.
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 371
  - b) Sì: consultare [“Come consentire alle applicazioni utente di collegarsi al gestore code”](#) a pagina 372.

## Multi z/OS **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.

<i>Tabella 66. Concessione dell'accesso di gestione parziale a un sottoinsieme di risorse del gestore code</i>	
<b>Gli utenti devono gestire gli oggetti di questo tipo</b>	<b>Esegui questa azione</b>
Code	Concedere l'accesso di gestione parziale alle code richieste, come descritto in <a href="#">“Concessione di un accesso amministrativo limitato ad alcune code”</a> a pagina 352
Argomenti	Concedere l'accesso di gestione parziale agli argomenti richiesti, come descritto in <a href="#">“Concessione di un accesso amministrativo limitato ad alcuni argomenti”</a> a pagina 354
Canali	Concedere l'accesso amministrativo parziale ai canali richiesti, come descritto in <a href="#">“Concessione di un accesso amministrativo limitato ad alcuni canali”</a> a pagina 355
Il gestore code	Concedere l'accesso di gestione parziale al gestore code, come descritto in <a href="#">“Concessione di un accesso di gestione limitato a un gestore code”</a> a pagina 356
Processo padre	Concedere l'accesso amministrativo parziale ai processi richiesti, come descritto in <a href="#">“Concessione di un accesso amministrativo limitato ad alcuni processi”</a> a pagina 358
Elenchi nomi	Concedere l'accesso amministrativo parziale agli elenchi nomi richiesti, come descritto in <a href="#">“Concessione di un accesso amministrativo limitato ad alcuni elenchi nomi”</a> a pagina 359
Servizi	Concedere l'accesso amministrativo parziale ai servizi richiesti, come descritto in <a href="#">“Concessione di un accesso amministrativo limitato ad alcuni servizi”</a> a pagina 360

### **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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

- ▶ **IBM i** IBM i
- ▶ **Linux** Linux
- ▶ **UNIX** UNIX
- ▶ **IBM i** Windows

**Nota:** ▶ **MQ Appliance** Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC**.

## Procedura

- ▶ **ULW**

Per sistemi UNIX, Linux, and Windows , immettere il seguente comando:

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

- ▶ **IBM i**

Per IBM i, immettere il seguente comando:

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

- ▶ **z/OS** Per z/OS, immettere i seguenti comandi per concedere l'accesso a una coda specificata:

```
RDEFINE MQADMIN QMgrName.QUEUE. ObjectProfile UACC(NONE)  
PERMIT QMgrName.QUEUE. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Per specificare i comandi MQSC che l'utente può eseguire sulla coda, immettere i seguenti comandi per ciascun comando MQSC:

```
RDEFINE MQCMDS QMgrName. ReqdAction. QType UACC(NONE)  
PERMIT QMgrName. ReqdAction. QType CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```

Per consentire all'utente di utilizzare il comando DISPLAY QUEUE, immettere i seguenti comandi:

```
RDEFINE MQCMDS QMgrName.DISPLAY. QType UACC(NONE)  
PERMIT QMgrName.DISPLAY. QType CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

I nomi delle variabili hanno i seguenti significati:

### QMGrName

Il nome del gestore code.

▶ **z/OS** Su z/OS, questo valore può essere anche 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:

- **ULW** Su sistemi UNIX, Linux, and Windows , qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + dlt, + dsp. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.
- **IBM i** Su IBM i, qualsiasi combinazione delle seguenti autorizzazioni: \*ADMCHG, \*ADMCLR, \*ADMDLT, \*ADMDSP. L'autorizzazione \*ALLADM è equivalente a tutte queste autorizzazioni individuali.
- **z/OS** Su z/OS, uno dei valori ALTER, CLEAR, DELETE o MOVE.

**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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

- **IBM i** IBM i
- **Linux** Linux
- **UNIX** UNIX
- **IBM i** Windows

**Nota:** **MQ Appliance** Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

## Procedura

- **ULW**  
Per sistemi UNIX, Linux, and Windows , immettere il seguente comando:

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

- **IBM i**  
Per IBM i, immettere il seguente comando:

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

- **z/OS** Per z/OS, immettere i seguenti comandi:

```
RDEFINE MQADMIN QMgrName.TOPIC. ObjectProfile UACC(NONE)
PERMIT QMgrName.TOPIC. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Questi comandi concedono l'accesso all'argomento specificato. Per stabilire quali comandi MQSC l'utente può eseguire sull'argomento, immettere i seguenti comandi per ogni comando MQSC:

```
RDEFINE MQCMDS QMgrName.ReqAction.TOPIC UACC(NONE)
PERMIT QMgrName.ReqAction.TOPIC CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```

Per consentire all'utente di utilizzare il comando DISPLAY TOPIC, immettere i seguenti comandi:

```
RDEFINE MQCMDS QMgrName.DISPLAY.TOPIC UACC(NONE)
PERMIT QMgrName.DISPLAY.TOPIC CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

I nomi delle variabili hanno i seguenti significati:

### QMgrName

Il nome del gestore code.

► **z/OS** Su z/OS, questo valore può essere anche 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:

- ► **ULW** Su sistemi UNIX, Linux, and Windows , qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + crt, + dlt, + dsp. + ctrl. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.
- ► **IBM i** Su IBM i, qualsiasi combinazione delle seguenti autorizzazioni: \*ADMCHG, \*ADMCLR, \*ADMCRRT, \*ADMDLT, \*ADM DSP, \*CTRL. L'autorizzazione \*ALLADM è equivalente a tutte queste autorizzazioni individuali.
- ► **z/OS** Su z/OS, uno dei valori ALTER, CLEAR, DEFINE, DELETE o MOVE.

## 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

- ► **IBM i** IBM i
- ► **Linux** Linux
- ► **UNIX** UNIX
- ► **IBM i** Windows

**Nota:** ► **MQ Appliance** Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

## Procedura

- ► **ULW**  
Su UNIX, Linux, and Windows:

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

#### IBM i

Su IBM i:

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

#### z/OS

Su z/OS:

```
RDEFINE MQADMIN QMgrName.CHANNEL.ObjectProfile UACC(NONE)  
PERMIT QMgrName.CHANNEL.ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Questi comandi concedono l'accesso al canale specificato. Per determinare quali comandi MQSC l'utente può eseguire sul canale, immettere i seguenti comandi per ogni comando MQSC:

```
RDEFINE MQCMD5 QMgrName.ReqdAction.CHANNEL UACC(NONE)  
PERMIT QMgrName.ReqdAction.CHANNEL CLASS(MQCMD5) ID(GroupName) ACCESS(ALTER)
```

Per consentire all'utente di utilizzare il comando DISPLAY CHANNEL, immettere i seguenti comandi:

```
RDEFINE MQCMD5 QMgrName.DISPLAY.CHANNEL UACC(NONE)  
PERMIT QMgrName.DISPLAY.CHANNEL CLASS(MQCMD5) ID(GroupName) ACCESS(READ)
```

I nomi delle variabili hanno i seguenti significati:

#### QMgrName

Il nome del gestore code.

#### z/OS

Su z/OS, questo valore può essere anche 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:

- **ULW** Su UNIX, Linux, and Windows, qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + crt, + dlt, + dsp, + ctrl, + ctrlx. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.
- **IBM i** Su IBM i, qualsiasi combinazione delle seguenti autorizzazioni: \*ADMCHG, \*ADMCLR, \*ADMCRRT, \*ADMDLT, \*ADM DSP, \*CTRL, \*CTRLx. L'autorizzazione \*ALLADM è equivalente a tutte queste autorizzazioni individuali.
- **z/OS** Su z/OS, uno dei valori ALTER, CLEAR, DEFINE, DELETE o MOVE.

### 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC**.

## Procedura

- 

Su UNIX, Linux, and Windows:

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

- 

Su IBM i:

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

- 

Su z/OS:

Per determinare quali comandi MQSC è possibile eseguire sul gestore code, immettere i seguenti comandi per ciascun 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 UNIX, Linux, and 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.

- **IBM i** Su IBM i, qualsiasi combinazione delle seguenti autorizzazioni: \*ADMCHG, \*ADMCLR, \*ADMCRRT, \*ADMDLT, \*ADMDSP. L'autorizzazione \*ALLADM è equivalente a tutte queste autorizzazioni individuali.

### **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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

- **IBM i** IBM i
- **Linux** Linux
- **UNIX** UNIX
- **IBM i** Windows

**Nota:** **MQ Appliance** Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

### **Procedura**

- **ULW**  
Su UNIX, Linux, and Windows:

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

- **IBM i**  
Su IBM i:

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

- **z/OS** Su z/OS:

```
RDEFINE MQADMIN QMgrName.PROCESS. ObjectProfile UACC(NONE)  
PERMIT QMgrName.PROCESS. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Questi comandi concedono l'accesso al canale specificato. Per determinare quali comandi MQSC l'utente può eseguire sul canale, immettere i seguenti comandi per ogni comando MQSC:

```
RDEFINE MQCMDS QMgrName. ReqdAction.PROCESS UACC(NONE)  
PERMIT QMgrName. ReqdAction.PROCESS CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```


Per consentire all'utente di utilizzare il comando DISPLAY PROCESS, immettere i seguenti comandi:

```
RDEFINE MQCMDS QMgrName.DISPLAY.PROCESS UACC(NONE)  
PERMIT QMgrName.DISPLAY.PROCESS 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 essere anche 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 UNIX, Linux, and Windows, qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + crt, + dlt, + dsp. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.
-  Su IBM i, qualsiasi combinazione delle seguenti autorizzazioni: \*ADMCHG, \*ADMCLR, \*ADMCRRT, \*ADMDLT, \*ADM DSP. L'autorizzazione \*ALLADM è equivalente a tutte queste autorizzazioni individuali.
-  Su z/OS, uno dei valori ALTER, CLEAR, DEFINE, DELETE o MOVE.


### 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :


-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

### Procedura

-  Su UNIX, Linux, and Windows:

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

-  Su IBM i:

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

-  Su z/OS:

```
RDEFINE MQADMIN QMgrName.NAMELIST. ObjectProfile UACC(NONE)  
PERMIT QMgrName.NAMELIST. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Questi comandi concedono l'accesso all'elenco nomi specificato. Per determinare quali comandi MQSC l'utente può eseguire sull'elenco nomi, immettere i seguenti comandi per ogni comando MQSC:

```
RDEFINE MQCMD5 QMgrName.ReqdAction.NAMELIST UACC(NONE)
PERMIT QMgrName.ReqdAction.NAMELIST CLASS(MQCMD5) ID(GroupName) ACCESS(ALTER)
```

Per consentire all'utente di utilizzare il comando DISPLAY NAMELIST, immettere i seguenti comandi:

```
RDEFINE MQCMD5 QMgrName.DISPLAY.NAMELIST UACC(NONE)
PERMIT QMgrName.DISPLAY.NAMELIST CLASS(MQCMD5) ID(GroupName) ACCESS(READ)
```

I nomi delle variabili hanno i seguenti significati:

### QMgrName

Il nome del gestore code.

► **z/OS** Su z/OS, questo valore può essere anche 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:

- ► **ULW** Su UNIX, Linux, and Windows, qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + crt, + dlt, + ctrl, + ctrlx, + dsp. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.
- ► **IBM i** Su IBM i, qualsiasi combinazione delle seguenti autorizzazioni: \*ADMCHG, \*ADMCLR, \*ADMCRRT, \*ADMDLT, \*ADM DSP, \*CTRL, \*CTRLX. L'autorizzazione \*ALLADM è equivalente a tutte queste autorizzazioni individuali.
- ► **z/OS** Su z/OS, uno dei valori ALTER, CLEAR, DEFINE, DELETE o MOVE.

## 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. ► **z/OS** Si noti che gli oggetti di servizio non esistono su z/OS.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

- ► **IBM i** IBM i
- ► **Linux** Linux
- ► **UNIX** UNIX
- ► **IBM i** Windows

**Nota:** ► **MQ Appliance** Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

## Procedura

- ► **ULW**



Su UNIX, Linux, and Windows:

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

- Su IBM i:

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

-  Su z/OS:

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, and Windows , qualsiasi combinazione delle seguenti autorizzazioni: + chg, + clr, + crt, + dlt, + ctrl, + ctrlx, + dsp. L'autorizzazione + alladm è equivalente a + chg + clr + dlt + dsp.
-  Su IBM i, qualsiasi combinazione delle seguenti autorizzazioni: \*ADMCHG, \*ADMCLR, \*ADMCRRT, \*ADMDLT, \*ADMDSP, \*CTRL, \*CTRLX. L'autorizzazione \*ALLADM è equivalente a tutte queste autorizzazioni individuali.

## **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.

<b>Gli utenti devono gestire gli oggetti di questo tipo</b>	<b>Esegui questa azione</b>
Code	Concedere l'accesso amministrativo completo alle code richieste, come descritto in <a href="#">“Concessione di un accesso amministrativo completo ad alcune code”</a> a pagina 362

Tabella 67. Concessione dell'accesso di gestione completo 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 amministrativo completo agli argomenti richiesti, come descritto in <a href="#">“Concessione dell'accesso amministrativo completo ad alcuni argomenti”</a> a pagina 363
Canali	Concedere l'accesso di gestione completo ai canali richiesti, come descritto in <a href="#">“Concessione di un accesso amministrativo completo ad alcuni canali”</a> a pagina 364
Il gestore code	Concedere l'autorizzazione di gestione completa al gestore code, come descritto in <a href="#">“Concessione di un accesso di gestione completo a un gestore code”</a> a pagina 365
Processo padre	Concedere l'accesso amministrativo completo ai processi richiesti, come descritto in <a href="#">“Concessione dell'accesso amministrativo completo ad alcuni processi”</a> a pagina 366
Elenchi nomi	Concedere l'accesso amministrativo completo agli elenchi nomi richiesti, come descritto in <a href="#">“Concessione di accesso amministrativo completo ad alcuni elenchi nomi”</a> a pagina 367
Servizi	Concedere l'accesso amministrativo completo ai servizi richiesti, come descritto in <a href="#">“Concessione di un accesso amministrativo completo ad alcuni servizi”</a> a pagina 367





### 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

### Procedura

-  Su UNIX, Linux, and Windows:

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

- ▶ **IBM i**

Su IBM i:

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

- ▶ **z/OS**

Su z/OS:

```
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.

▶ **z/OS**

Su z/OS, questo valore può essere anche 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

- ▶ **IBM i** IBM i

- ▶ **Linux** Linux

- ▶ **UNIX** UNIX

- ▶ **IBM i** Windows

**Nota:** ▶ **MQ Appliance** Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

### **Procedura**

- ▶ **ULW**

Su UNIX, Linux, and Windows:

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

- ▶ **IBM i**

Su IBM i:

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

- ▶ **z/OS**

Su z/OS:

```
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.

- ▶ **z/OS**

Su z/OS, questo valore può essere anche 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

- ▶ **IBM i** IBM i
- ▶ **Linux** Linux
- ▶ **UNIX** UNIX
- ▶ **IBM i** Windows

**Nota:** [MQ Appliance](#) Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

### **Procedura**

- ▶ **ULW**

Su UNIX, Linux, and Windows:

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

- ▶ **IBM i**

Su IBM i:

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

- ▶ **z/OS**


Su z/OS:

```
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 essere anche 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :


-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

## Procedura

-  Su UNIX, Linux, and Windows:

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

-  Su IBM i:

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


-  Su z/OS:

```
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 essere anche 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :


-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

## Procedura

-  Su UNIX, Linux, and Windows:

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

-  Su IBM i:

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


-  Su z/OS:

```
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 essere anche 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :


-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

### Procedura

-  Su UNIX, Linux, and Windows:

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

-  Su IBM i:

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


-  Su z/OS:

```
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 essere anche 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

- ▶ **IBM i** IBM i
- ▶ **Linux** Linux
- ▶ **UNIX** UNIX
- ▶ **IBM i** Windows

**Nota:** [MQ Appliance](#) Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

## Procedura

- ▶ **ULW**

Su UNIX, Linux, and Windows:

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

- ▶ **IBM i**

Su IBM i:

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

- ▶ **z/OS**

Su z/OS:

```
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.

▶ **z/OS** Su z/OS, questo valore può essere anche 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

- ▶ **IBM i** IBM i



-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC**.

Una volta modificati i dettagli di autorizzazione, eseguire un aggiornamento della sicurezza utilizzando il comando [REFRESH SECURITY](#).

## Procedura

- Utilizzando la procedura guidata:
  - a) Nel riquadro IBM 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 i sistemi UNIX e Windows, immettere i seguenti comandi:

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

Le autorizzazioni specifiche per SYSTEM.ADMIN.COMMAND.QUEUE e SYSTEM.MQEXPLORER.REPLY.MODEL sono necessari solo se si desidera utilizzare IBM MQ Explorer.

- 

Per IBM i, immettere i seguenti comandi:

```
GRTMQMAUT OBJ(*ALL) OBJTYPE(*Q) USER('GroupName') AUT(*ADM DSP *BROWSE) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*TOPIC) USER('GroupName') AUT(*ADM DSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*CHL) USER('GroupName') AUT(*ADM DSP *INQ) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*CLTCN) USER('GroupName') AUT(*ADM DSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*AUTHINFO) USER('GroupName') AUT(*ADM DSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*LSR) USER('GroupName') AUT(*ADM DSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*NMLIST) USER('GroupName') AUT(*ADM DSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*PRC) USER('GroupName') AUT(*ADM DSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*SVC) USER('GroupName') AUT(*ADM DSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ('object-name') OBJTYPE(*MQM) USER('GroupName') AUT(*ADM DSP *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 MXTOPIC QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MXTOPIC) ID(GroupName) ACCESS(READ)
RDEFINE MQPROC QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQPROC) ID(GroupName) ACCESS(READ)
RDEFINE MQNLIST QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQNLIST) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
PERMIT QMgrName.BATCH CLASS(MQCONN) ID(GroupName) ACCESS(READ)
```

```
RDEFINE MQCONN QMgrName.CICS UACC(NONE)
PERMIT QMgrName.CICS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.IMS UACC(NONE)
PERMIT QMgrName.IMS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.CHIN UACC(NONE)
PERMIT QMgrName.CHIN CLASS(MQCONN) ID(GroupName) ACCESS(READ)
```

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

È possibile utilizzare la procedura guidata Aggiungi autorizzazioni basate sui ruoli o i comandi appropriati per il proprio sistema operativo.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

- IBM i
- Linux
- UNIX
- Windows

**Nota:** Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

**Note:**

1. Se si utilizza **runmqsc** per gestire il gestore code invece di IBM MQ Explorer, è necessario concedere l'autorizzazione per interrogare, richiamare e sfogliare il SISTEMA SYSTEM.MQSC.REPLY.QUEUEe non è necessario concedere alcuna autorizzazione su SYSTEM.MQEXPLORER.REPLY.MODEL .
2. Quando si concede a un utente l'accesso a tutte le risorse su un gestore code, l'utente non può eseguire alcuni comandi, a meno che non disponga dell'accesso in lettura al file `qm.ini` . Ciò è dovuto alle limitazioni per gli utenti non mqm che possono leggere il file `qm.ini` .

L'utente non può immettere i seguenti comandi a meno che non sia stato concesso l'accesso in lettura al file `qm.ini` :

- Definizione di un canale configurato per utilizzare TLS
- Definizione di un canale utilizzando le variabili di inserimento della configurazione automatica definite in `qm.ini`

## Procedura

- Se si sta utilizzando la procedura guidata, nel riquadro IBM MQ Explorer Navigator , fare clic con il tasto destro del mouse sul gestore code e fare clic su **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.MODEL -t queue -g GroupName +dsp +inq +get +put
setmqaut -m QMgrName -n '**' -t topic -g GroupName +alladm
setmqaut -m QMgrName -n @class -t topic -g GroupName +crt
setmqaut -m QMgrName -n '**' -t channel -g GroupName +alladm
setmqaut -m QMgrName -n @class -t channel -g GroupName +crt
setmqaut -m QMgrName -n '**' -t clntconn -g GroupName +alladm
setmqaut -m QMgrName -n @class -t clntconn -g GroupName +crt
setmqaut -m QMgrName -n '**' -t authinfo -g GroupName +alladm
setmqaut -m QMgrName -n @class -t authinfo -g GroupName +crt
setmqaut -m QMgrName -n '**' -t listener -g GroupName +alladm
setmqaut -m QMgrName -n @class -t listener -g GroupName +crt
setmqaut -m QMgrName -n '**' -t namelist -g GroupName +alladm
setmqaut -m QMgrName -n @class -t namelist -g GroupName +crt
setmqaut -m QMgrName -n '**' -t process -g GroupName +alladm
setmqaut -m QMgrName -n @class -t process -g GroupName +crt
setmqaut -m QMgrName -n '**' -t service -g GroupName +alladm
setmqaut -m QMgrName -n @class -t service -g GroupName +crt
setmqaut -m QMgrName -t qmgr -g GroupName +alladm +connect
```

#### Windows

Per sistemi Windows , immettere gli stessi comandi dei sistemi UNIX and Linux , ma utilizzando il nome profilo @CLASS invece di @class.

#### IBM i

Per IBM i, immettere il seguente comando:

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

#### z/OS

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.

#### z/OS

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.

Su sistemi UNIX, Linux, Windows e IBM i, è anche possibile utilizzare il comando [DELETE AUTHREC](#) .

**Nota:** In IBM MQ Appliance è possibile utilizzare solo il comando **DELETE AUTHREC** .

## Procedura

#### ULW

Per sistemi UNIX, Linux, and Windows , immettere il seguente comando:

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

### IBM i

Per IBM i, immettere il seguente comando:

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

### z/OS

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.

#### z/OS

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 380.

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 373.

In entrambi i casi, impostare la sicurezza della connessione come descritto in [“Impostazione della sicurezza della connessione”](#) a pagina 380

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 <a href="#">“Controllo dell'accesso utente alle code”</a> a pagina 381
Si dispone di applicazioni che utilizzano argomenti	Consultare <a href="#">“Controllo dell'accesso utente agli argomenti”</a> a pagina 387.
Sono presenti applicazioni che interrogano l'oggetto gestore code	Consultare <a href="#">“Concessione dell'autorità per richiedere informazioni su un gestore code”</a> a pagina 389.
Si dispone di applicazioni che utilizzano oggetti processo	Vedi <a href="#">“Concessione dell'autorità per accedere ai processi”</a> a pagina 390

Istruzione	Esegui questa azione
Si dispone di applicazioni che utilizzano elenchi nomi	Vedi <a href="#">“Concessione dell'autorità per accedere agli elenchi nomi”</a> a pagina 390

### **Protezione della connettività remota al gestore code**

È possibile proteggere la connettività remota al gestore code utilizzando 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 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 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 sicurezza per assegnare un valore MCAUSER in base al valore del DN TLS che riceve nei campi SSLPeerNamePtr e SSLPeerNameLength passati all'uscita nella struttura MQCD.
3. Utilizzo di 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 canali su canali che non utilizzano 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

## Informazioni correlate

### 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 373](#)

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.

## Informazioni correlate

[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 324. Per ulteriori informazioni su \*MQADMIN, consultare [SET CHLAUTH](#).

## Informazioni correlate

### [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 seguenti, 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](#).

## Informazioni correlate

### [SET CHLAUTH](#)

*Associazione di un ID utente client a 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 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 associato alla connessione client, il valore potrebbe essere asserito dall'applicazione client, modificato dall'autenticazione della connessione utilizzando l'adozione anticipata o impostato tramite un'exit del canale.

*utente* è l'ID utente da utilizzare al posto del nome utente client.

## Informazioni correlate

[SET CHLAUTH](#)

[Attributi della stanza dei canali \(ChlauthEarlyAdopt\)](#)

*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) SSLCERTI(generic-issuer-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 MQ per i valori SSLPEER. Consultare [IBM MQ regole per i valori SSLPE](#).

*user* è l'ID utente da utilizzare per tutte le connessioni che utilizzano il DN specificato.

*nome - emittente - generico* fa riferimento al DN dell'emittente del certificato per la corrispondenza. Questo parametro è facoltativo, ma è necessario utilizzarlo, per evitare la corrispondenza errata del certificato, se sono in uso più autorità di certificazione.

## Informazioni correlate

### [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 seguente comando, 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 ')\nUSERSRC(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.

## Informazioni correlate

### [SET CHLAUTH](#)

*Blocco dell'accesso per un ID utente client*

È possibile utilizzare un record di autenticazione di canale per evitare che un ID utente client stabilisca una connessione di canale.

## 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 ')\nUSERSRC(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 associato alla connessione client, il valore potrebbe essere asserito dall'applicazione client, modificato dall'autenticazione della connessione utilizzando l'adozione anticipata o impostato tramite un'exit del canale.

### Informazioni correlate

#### SET CHLAUTH

*Blocco dell'accesso per un DN (Distinguished Name) SSL o TLS*

È possibile utilizzare un record di autenticazione di canale per evitare che un DN (Distinguished Name) TLS 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') SSLCERTI(generic-issuer-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 MQ per i valori SSLPEER. Consultare [IBM MQ regole per i valori SSLPE](#).

*nome - emittente - generico* fa riferimento al DN dell'emittente del certificato per la corrispondenza. Questo parametro è facoltativo, ma è necessario utilizzarlo, per evitare la corrispondenza errata del certificato, se sono in uso più autorità di certificazione.

### Informazioni correlate

#### 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.

## Informazioni correlate

[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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :


-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

### Procedura

-  Su UNIX, Linux, and Windows:

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

-  Su IBM i:

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

-  Su z/OS:

```
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)  
PERMIT QMgrName.BATCH CLASS(MQCONN) ID(GroupName) ACCESS(READ)  
RDEFINE MQCONN QMgrName.CICS UACC(NONE)
```

```

PERMIT QMgrName.CICS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName. IMS UACC(NONE)
PERMIT QMgrName. IMS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName. CHIN UACC(NONE)
PERMIT QMgrName. CHIN CLASS(MQCONN) ID(GroupName) ACCESS(READ)

```

Questi comandi forniscono l'autorizzazione a collegarsi 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.

**Informazioni correlate**

Profili di sicurezza di connessione per l'iniziatore di canali

**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 <a href="#">“Concessione dell'autorità per richiamare i messaggi dalle code”</a> a pagina 381
L'applicazione imposta il contesto	Vedi <a href="#">“Concessione dell'autorità per impostare il contesto”</a> a pagina 382
L'applicazione passa il contesto	Vedi <a href="#">“Concessione dell'autorizzazione per passare il contesto”</a> a pagina 383
L'applicazione inserisce i messaggi in una coda cluster	Vedi <a href="#">“Autorizzazione all'inserimento di messaggi nelle code del cluster remoto”</a> a pagina 433
L'applicazione inserisce i messaggi su una coda locale	Vedi <a href="#">“Concessione dell'autorizzazione per inserire i messaggi in una coda locale”</a> a pagina 384
L'applicazione inserisce i messaggi in una coda modello	Vedi <a href="#">“Concessione dell'autorizzazione per inserire i messaggi in una coda modello”</a> a pagina 385
L'applicazione inserisce i messaggi su una coda remota	Vedi <a href="#">“Concessione dell'autorità per inserire i messaggi in una coda cluster remota”</a> a pagina 386

*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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

- ▶ **IBM i** IBM i
- ▶ **Linux** Linux

-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

## Procedura

- Per sistemi UNIX, Linux, and 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(MQADMIN) 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

## Procedura

- Per sistemi UNIX, Linux, and Windows , immettere uno dei seguenti comandi:
  - 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
```

**Nota:** Per utilizzare l'autorizzazione `setid` o `setall`, le autorizzazioni devono essere concesse sia sull'oggetto coda appropriato che sull'oggetto `gestore code`.

- 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC**.

## Procedura

- Per sistemi UNIX, Linux, and Windows , immettere uno dei seguenti comandi:
  - 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, immetti 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .



## Procedura

- Per sistemi UNIX, Linux, and 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

## Procedura

- Per sistemi UNIX, Linux, and 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 sta utilizzando 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 384. 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC**.

### **Procedura**

- Per sistemi UNIX, Linux, and 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 QMGrName.ObjectProfile UACC(NONE)
PERMIT QMGrName.QUEUE.ObjectProfile CLASS(MQQUEUE)
ID(GroupName) ACCESS(UPDATE)
```

Si noti 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.

<i>Tabella 68. Controllo dell'accesso utente agli argomenti</i>	
<b>Istruzione</b>	<b>Azione</b>
L'applicazione pubblica i messaggi in un argomento	Vedi <a href="#">“Concessione dell'autorizzazione a pubblicare messaggi in un argomento”</a> a pagina 387
L'applicazione si sottoscrive a un argomento	Vedi <a href="#">“Concessione dell'autorizzazione alla sottoscrizione di argomenti”</a> a pagina 388





*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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

## Procedura

- Per sistemi UNIX, Linux, and 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

## Procedura

- Per sistemi UNIX, Linux, and 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

- ▶ **IBM i** IBM i
- ▶ **Linux** Linux
- ▶ **UNIX** UNIX
- ▶ **IBM i** Windows

**Nota:** [MQ Appliance](#) Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

### **Procedura**

- Per sistemi UNIX, Linux, and 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando [SET AUTHREC](#) :

-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando **SET AUTHREC** .

## **Procedura**

- Per sistemi UNIX, Linux, and 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.

Sulle seguenti piattaforme, è anche possibile utilizzare il comando `SET AUTHREC` :

-  IBM i
-  Linux
-  UNIX
-  Windows

**Nota:**  Su IBM MQ Appliance è possibile utilizzare solo il comando `SET AUTHREC` .

## Procedura

- Per sistemi UNIX, Linux, and 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.


## ULW

## **Autorizzazione per gestire IBM MQ su UNIX, Linux, and Windows**

Gli amministratori IBM MQ possono utilizzare tutti i comandi IBM 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 MQ hanno l'autorità di utilizzare tutti i comandi IBM MQ (inclusi i comandi per concedere le autorizzazioni IBM MQ ad altri utenti).

Per essere un amministratore IBM MQ , è necessario essere membro di un gruppo speciale denominato gruppo **mqm** .

 In alternativa, solo su Windows , è possibile essere membri del gruppo Amministratori sui sistemi Windows .

Il gruppo **mqm** viene creato automaticamente quando IBM MQ è installato. È possibile 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 IBM MQ. Uno di questi comandi di controllo è **setmqaut**, che viene utilizzato per concedere le autorizzazioni ad altri utenti per consentire loro di accedere o controllare le risorse IBM MQ. I comandi PCF per la gestione dei record di autorizzazioni sono disponibili per i non amministratori a cui sono 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 devono disporre delle autorizzazioni richieste per i comandi MQSC che devono essere elaborati dal gestore code remoto. IBM MQ Explorer emette comandi PCF per eseguire attività di gestione. Gli amministratori non richiedono ulteriori autorizzazioni per utilizzare IBM MQ Explorer per gestire un gestore code sul sistema locale. Quando IBM 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.



**Attenzione:** Da IBM MQ 8.0, non è necessario essere un amministratore per utilizzare il comando di controllo **runmqsc**, che emette comandi MQSC (IBM MQ Script).

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.

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 [Autorità per gestire gli oggetti IBM MQ](#). 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](#)
-  Per i comandi MQSC elaborati dal server dei comandi su IBM MQ for z/OS, fare riferimento a [Sicurezza dei comandi e sicurezza delle risorse dei comandi su z/OS](#).

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

Sulle piattaforme UNIX and Linux, viene creato anche un ID utente speciale **mqm**, che può essere utilizzato solo dal prodotto. Non deve essere mai disponibile per utenti non privilegiati. Tutti gli oggetti IBM 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 inoltre possibile creare un gruppo di dominio **mqm** sul controller di dominio che contiene tutti gli ID utente con privilegi attivi all'interno del dominio e aggiungerlo al gruppo **mqm** locale. Alcuni comandi, ad esempio **crtmqm**, manipolano le autorizzazioni sugli oggetti IBM MQ e quindi necessitano dell'autorizzazione per gestire tali oggetti (come descritto nelle seguenti sezioni). I membri del gruppo **mqm** dispongono dell'autorizzazione per gestire tutti gli oggetti, ma potrebbero verificarsi delle circostanze sui sistemi Windows quando 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 su UNIX, Linux, and Windows”](#) a pagina 396.

Le versioni di Windows con una funzione UAC (User Account Control) limitano le azioni che gli utenti possono eseguire su determinate funzionalità del sistema operativo, anche se sono membri del gruppo Amministratori. Se il proprio ID utente si trova nel gruppo Administrators ma non nel gruppo **mqm**, è necessario utilizzare un prompt dei comandi elevato per immettere i comandi di amministrazione IBM 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 pulsante 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 un membro del gruppo **mqm** per effettuare le azioni riportate di seguito:



- 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'inziatore di canali” a pagina 102](#)).
- Emettere chiamate MQI da un programma applicativo (a meno che non si desideri utilizzare i collegamenti rapidi sulla chiamata MQCONN).
- Utilizzare il comando `cimqcvx` per creare un frammento di codice che esegue la conversione dei dati sulle strutture dei tipi di dati.
- Utilizzare il comando `dspmqr` per visualizzare i gestori code.
- Utilizzare il comando `dspmqrtrc` per visualizzare l'emissione della traccia formattata IBM 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 5.3 ha aumentato questo limite ma IBM MQ continua a osservare una limitazione di 12 caratteri su tutte le piattaforme UNIX and Linux. Se si utilizza un ID utente superiore a 12 caratteri, IBM MQ lo sostituisce con il valore UNKNOWN. Non definire un ID utente con valore UNKNOWN.

## ULW Gestione del gruppo mqm su UNIX, Linux, and Windows

Agli utenti nel gruppo mqm vengono concessi privilegi amministrativi completi su IBM 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 IBM MQ.

Queste attività sono descritte in:

- **Windows** [Creazione e gestione di gruppi su Windows](#)
- **HP-UX** [Creazione e gestione di gruppi su HP-UX](#)
- **AIX** [Creazione e gestione di gruppi su AIX](#)
- **Solaris** [Creazione e gestione di gruppi su Solaris](#)
- **Linux** [Creazione e gestione di gruppi su Linux](#)

**Windows** Se il tuo controller di dominio viene eseguito su Windows 2000 o Windows 2003, il tuo amministratore di dominio potrebbe dover configurare un account speciale per l'utilizzo da parte di IBM MQ. Per ulteriori informazioni, vedi [Configurazione di IBM MQ con Prepare IBM MQ Wizard](#) e [Creazione e impostazione di account di dominio Windows per IBM MQ](#).

## ULW Autorizzazione per gestire gli oggetti IBM MQ su UNIX, Linux, and Windows

Tutti gli oggetti sono protetti da IBM MQ e 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 IBM MQ e alle applicazioni deve essere concessa l'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, a un gruppo potrebbe essere consentito eseguire sia operazioni di inserimento che di acquisizione; a un altro gruppo potrebbe essere consentito solo 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 a un oggetto viene concessa automaticamente all'ID utente che ha creato l'oggetto e a tutti i membri del gruppo mqm (e ai membri del gruppo Amministratori locale sui sistemi Windows).

### Concetti correlati

[“Autorizzazione per gestire IBM MQ su UNIX, Linux, and Windows” a pagina 391](#)

Gli amministratori IBM MQ possono utilizzare tutti i comandi IBM 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.

## **ULW** Quando i controlli di sicurezza vengono eseguiti su 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. IBM MQ, quindi, verifica che l'ID utente sia autorizzato a connettersi al gestore code e conserva l'ID utente per i controlli futuri.

Gli utenti non devono collegarsi a IBM MQ; IBM MQ presuppone che gli utenti abbiano eseguito l'accesso al sistema operativo sottostante e che siano stati autenticati da tale sistema.

### Apertura dell'oggetto (chiamate MQOPEN o MQPUT1)

È possibile accedere agli oggetti IBM MQ aprendo l'oggetto ed emettendo 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).

IBM 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 121](#).

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 nel punto o al di sopra del punto 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 accessi da parte di IBM MQ su UNIX, Linux, and Windows

IBM MQ utilizza i servizi di sicurezza forniti dal sistema operativo sottostante, utilizzando il gestore autorizzazioni oggetto. IBM 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 IBM MQ. IBM MQ fornisce un'implementazione di un Access Control Manager (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 350](#)). 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 IBM MQ solo se dispongono dell'autorizzazione corretta. [“Controllo dell'accesso agli oggetti utilizzando OAM su UNIX, Linux, and Windows” a pagina 340](#) 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, in aggiunta, su Windows, agli utenti del gruppo Amministratori e agli utenti collegati con l'ID SISTEMA. L'accesso utente alla coda non può essere modificato.

IBM MQ fornisce comandi per la creazione e la gestione degli elenchi di controllo accessi. Per ulteriori informazioni su questi comandi, consultare [“Controllo dell'accesso agli oggetti utilizzando OAM su UNIX, Linux, and Windows” a pagina 340](#).

IBM 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. IBM MQ segue la decisione di OAM; se OAM non può prendere una decisione, IBM MQ non consente l'accesso.

## Identificazione dell'ID utente su 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 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 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 UNIX, Linux, and Windows”](#) a pagina 398. Le applicazioni non possono modificare queste informazioni a meno che non siano state autorizzate a modificare le informazioni di contesto.

## **ULW** *Principal e gruppi su UNIX, Linux, and Windows*

I principal possono appartenere a gruppi. Concedendo l'accesso alle risorse ai gruppi piuttosto che agli individui, è possibile ridurre la quantità di amministrazione richiesta. Gli ACL (Access Control List) si basano su gruppi e ID utente.

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 di definizione e gestione dei gruppi è descritto per particolari piattaforme:

- **Windows** [Creazione e gestione di gruppi su Windows](#)
- **HP-UX** [Creazione e gestione di gruppi su HP-UX](#)
- **AIX** [Creazione e gestione di gruppi su AIX](#)
- **Solaris** [Creazione e gestione di gruppi su Solaris](#)
- **Linux** [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. Queste autorizzazioni vengono memorizzate nella cache, quindi tutte le modifiche apportate all'appartenenza al gruppo del principal non vengono riconosciute fino al riavvio del gestore code, a meno che non si immette il comando MQSC **REFRESH SECURITY** (o il suo equivalente PCF).

## **Linux** **UNIX** *Sistemi UNIX and Linux*

Da IBM MQ 8.0, gli ACL (access control list) si basano su ID utente e gruppi ed è possibile utilizzarli per l'autorizzazione impostando l'attributo **SecurityPolicy** sul valore appropriato come descritto in [Configurazione dei servizi installabili](#) e [Configurazione delle stanze del servizio di autorizzazione in UNIX e Linux](#).

Da IBM MQ 8.0, è possibile utilizzare il modello basato sull'utente per l'autorizzazione e ciò consente di utilizzare sia utenti che gruppi. Tuttavia, quando si specifica un utente nel comando `setmqaut`, le nuove autorizzazioni si applicano solo a tale utente e non ai gruppi a cui appartiene tale utente. Per ulteriori informazioni, consultare [OAM user - based permissions on UNIX and Linux systems](#).

Quando si utilizza il modello *basato sui gruppi*, per l'autorizzazione, il gruppo primario a cui appartiene l'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. Puoi modificare

l'autorizzazione nel gruppo nobody per concedere l'accesso alle risorse IBM 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 gli ID utente arbitrari utilizzano le autorizzazioni di accesso di UNKNOWN.

Gli ID utente possono contenere fino a 12 caratteri e i nomi gruppo fino a 12 caratteri.

## Windows Sistemi Windows

Gli ACL si basano su ID utente e gruppi. Le verifiche sono le stesse di UNIX. È possibile avere utenti differenti su domini differenti con lo stesso ID utente. IBM 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_name\group_name
```

I gruppi globali vengono controllati da OAM solo in due casi:

1. La stanza di protezione del gestore code include l'impostazione: `GroupModel=GlobalGroups`. Consultare [Protezione](#).
2. Il gestore code sta utilizzando un gruppo di accesso di protezione alternativo. Vedere [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 sugli oggetti IBM 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 si trova nel gruppo mqm locale.

Per ulteriori informazioni sull'impostazione dell'attributo **SecurityPolicy** su Windows, consultare [Servizi installabili](#) e [Configurazione delle stanze del servizio di autorizzazione su Windows](#).

## Windows Identificativi di sicurezza (SID) Windows

IBM MQ su Windows utilizza il SID dove è disponibile. Se un SID Windows non viene fornito con una richiesta di autorizzazione, IBM MQ identifica l'utente in base al solo nome utente, ma ciò potrebbe comportare la concessione di un'autorizzazione errata.

Sui sistemi Windows, il SID (security identifier) viene utilizzato per integrare l'ID utente. Il SID contiene informazioni che identificano i dettagli dell'account utente completo sul database SAM (security account manager) Windows in cui è definito l'utente. Quando viene creato un messaggio su IBM MQ for Windows, IBM MQ memorizza il SID nel descrittore del messaggio. Quando IBM MQ on Windows esegue i controlli di autorizzazione, utilizza il SID per interrogare le informazioni complete 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 non viene fornito un SID Windows con una richiesta di autorizzazione, IBM MQ identifica l'utente in base al solo nome utente. Eseguire 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, potrebbe essere concessa l'autorizzazione IBM MQ non corretta. Per evitare questo problema, includere un SID in ogni richiesta di autorizzazione; il SID viene utilizzato da IBM MQ per stabilire le credenziali utente.

Per indicare che tutte le richieste di autorizzazione devono includere un SID, utilizzare **regedit**. Impostare SecurityPolicy su NTSIDsRequired.

## **ULW** Autorizzazione utente alternativo su UNIX, Linux, and Windows

È possibile specificare che un utente ID può utilizzare l'autorizzazione di un altro utente quando accede a un oggetto IBM MQ. Questa è denominata *autorizzazione utente alternativo* ed è possibile utilizzarla su qualsiasi oggetto IBM 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.

## **ULW** Autorizzazione contesto su UNIX, Linux, and 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 `AppIdentityData`.

### **Sezione Origine**

Da dove proviene il messaggio e quando è stato inserito nella coda. È costituito dai campi `PutAppType`, `PutAppName`, `PutDate`, `PutTime` e `AppOriginData`.

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 del canale creata da un responsabile IBM 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.

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.

**Multi** Su Multipiattaforme, a meno che il valore *MCAUserIdentifier* non sia vuoto, il gestore code utilizza il valore *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.

**z/OS** 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'inziatore 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'inziatore 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 IBM MQ passa dall'MCA della connessione client all'MCA della 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 del client del canale può inviare l'ID utente associato all'applicazione IBM MQ MQI client. 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 viene utilizzato TLS sul canale, il DN (Distinguished Name) del partner viene passato all'uscita nel campo `Ptr SSLPeerNamedi MQCD` e il DN dell'emittente di tale certificato viene passato all'uscita nel campo `Ptr SSLRemCertIssNamedi 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 del rilascio di IBM WebSphere MQ 7.1 potrebbero richiedere l'aggiornamento. Per ulteriori informazioni, consultare [Programmi di uscita di sicurezza del canale](#).

## Autenticazione utente di IBM MQ object authority manager

Su connessioni IBM MQ MQI client, le uscite di sicurezza possono essere utilizzate 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 un'autorizzazione utente alternativa, chiedere al gestore code di utilizzare questo ID utente per i controlli di autorizzazione quando accede a risorse IBM 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 una 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 326](#).

## Implementazione del controllo accessi nell'uscita API e nell'uscita incrociata API

Un'API o un'uscita incrociata API può fornire controlli di accesso per integrare quelli forniti da IBM 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.

## Autorizzazione LDAP

È possibile utilizzare l'autorizzazione LDAP per rimuovere la necessità di un ID utente locale.

### Disponibilità dell'autorizzazione LDAP sulle piattaforme supportate

L'autorizzazione LDAP è disponibile sulle piattaforme seguenti:

-  UNIX
-  IBM i
-  Windows



#### Attenzione:

Dalla disponibilità generale IBM MQ 9.0, questa funzionalità è disponibile su tutti i gestori code, sia nuovi che migrati da una release precedente.

### Panoramica dell'autorizzazione LDAP

Con l'autorizzazione LDAP, i comandi che gestiscono la configurazione dell'autorizzazione, come [setmqaut](#) e [DISPLAY AUTHREC](#), possono elaborare i DN (Distinguished Name). In precedenza, gli utenti venivano autenticati confrontando le credenziali con il numero massimo di caratteri disponibili per utenti e gruppi sul sistema operativo locale.



**Attenzione:** Se è stato eseguito il comando `DEFINE AUTHINFO`, è necessario riavviare il gestore code. Se non si riavvia il gestore code, il comando [setmqaut](#) non restituisce il risultato corretto.

Se un utente fornisce un ID utente, piuttosto che un DN (Distinguished Name), l'ID utente viene elaborato. Ad esempio, quando è presente un messaggio in entrata su un canale con `PUTAUT (CTX)`, i caratteri nell'ID utente vengono associati a un DN (Distinguished Name) LDAP e vengono eseguiti i controlli di autorizzazione appropriati.

Altri comandi, come [DISPLAY CONN](#), continuano a gestire e a mostrare il valore effettivo per l'ID utente, anche se tale ID utente potrebbe non esistere sul sistema operativo locale.

Quando l'autorizzazione LDAP è attiva, il gestore code utilizza sempre il modello utente di sicurezza sulle piattaforme UNIX, indipendentemente dall'attributo **SecurityPolicy** nel file `qm.ini`. Quindi, l'impostazione delle autorizzazioni per un singolo utente influisce solo su tale utente e non su chiunque altro appartenga a uno dei gruppi di tale utente.

Come per il modello di sistema operativo, un utente ha ancora l'autorizzazione combinata che è stata assegnata sia all'individuo che a tutti i gruppi (se presenti) a cui appartiene l'utente.

Ad esempio, si supponga che i seguenti record siano stati definiti in un repository LDAP.

- Nella classe **inetOrgPerson** :

```
dn="cn=JohnDoe, ou=users, o=yourcompany, c=yourcountry"
  email=JohnDoe1@yourcompany.com [longer than 12 characters]
  shortu=jodoe
  Phone=1234567
```

- Nella classe **groupOfNames** :

```
dn="cn=Application Group A, ou=groups, o=yourcompany, c=yourcountry"
  longname=ApplicationGroupA [longer than 12 characters]
  members="cn=JaneDoe, ou=users, o=yourcompany, c=yourcountry",
          "cn=JohnDoe, ou=users, o=yourcompany, c=yourcountry"
```

Per motivi di autenticazione, un gestore code che utilizza questo server LDAP deve essere stato definito in modo che il suo valore CONNAUTH punti a un oggetto AUTHINFO di tipo IDPWLDAP e i cui attributi di risoluzione del nome rilevanti siano probabilmente impostati come segue:

```
USRFIELD(email) SHORTUSR(shortu)
BASEDNU(ou=users,o=yourcompany,c=yourcountry) CLASSUSR(inetOrgPerson)
```

Data questa configurazione per l'autenticazione, un'applicazione può completare il campo CSPUserID, utilizzato nella chiamata MQCNO, con una delle seguenti serie di valori:

```
" cn=JohnDoe ", " JohnDoe1@yourcompany.com ", " email=JohnDoe1@yourcompany.com "
```

o

```
" cn=JohnDoe, ou=users, o=ibm, c=uk ", " shortu=jdoe "
```

In entrambi i casi, il sistema può utilizzare i valori forniti per autenticare il contesto SO di " jdoe".

## Impostazione delle autorizzazioni

Come utilizzare il nome breve o **USRFIELD** per impostare le autorizzazioni.

L'approccio dell'utilizzo di più formati, descritto in "Autorizzazione LDAP" a pagina 401, continua nei comandi di autorizzazione, con un'ulteriore estensione che consente di utilizzare `shortname` o **USRFIELD** in modo non adornato.

La stringa di caratteri specifica un particolare attributo nel record LDAP quando si denominano utenti (principal) per l'autorizzazione.

**Importante:** La stringa di caratteri non deve contenere il carattere = , poiché non può essere utilizzato in un ID utente del sistema operativo.

Se si passa un nome principal all'OAM per l'autorizzazione che è potenzialmente un `shortname`, la stringa di caratteri deve contenere 12 caratteri. L'algoritmo di associazione tenta di risolverlo in un DN utilizzando l'attributo **SHORTUSR** nella relativa query LDAP.

Se l'operazione ha esito negativo con un errore **UNKNOWN\_ENTITY** o se la stringa fornita non può essere un `shortname`, viene effettuato un ulteriore tentativo utilizzando l'attributo **USRFIELD** per creare la query LDAP.



**Attenzione:** Se è stato eseguito il comando **DEFINE AUTHINFO**, è necessario riavviare il gestore code. Se non si riavvia il gestore code, il comando `setmqaut` non restituisce il risultato corretto.

Per l'elaborazione delle autorizzazioni utente, le seguenti impostazioni del comando `setmqaut` sono tutte equivalenti.

Tabella 69. Impostazioni di autorizzazione utente	
Comando	Nota
<code>setmqaut -m QM -t qmgr -p jdoe +connect</code>	Questo è un nome semplice, non qualificato, risolto tramite <b>SHORTUSR</b> .
<code>setmqaut -m QM -t qmgr -pJohnDoe1@yourcompany.com +connect</code>	Anche un nome semplice, non qualificato, che si risolve tramite <b>USRFIELD</b> nella stessa entità.

Tabella 69. Impostazioni di autorizzazione utente (Continua)

Comando	Nota
setmqaut -m QM -t qmgr -p email=JohnDoe1@yourcompany.com +connect	Utilizzo di un attributo denominato.
setmqaut -m QM -t qmgr -p "phone=1234567" +connect	Utilizzo di un altro attributo denominato che non deve essere uno di quelli configurati sull'oggetto AUTHINFO.

È possibile utilizzare il comando MQSC [SET AUTHREC](#) come alternativa al comando **setmqaut** :

```
SET AUTHREC OBJTYPE(QMGR) PRINCIPAL('JohnDoe1@yourcompany.com') AUTHADD(connect)
```

oppure il comando PCF [Set Authority Record \(MQCMD\\_SET\\_AUTH\\_REC\)](#) con l'elemento MQCACF\_principe PAL\_ENTITY\_NAMES contenente la stringa:

```
"cn=JohnDoe,ou=users,o=yourcompany,c=yourcountry"
```

Quando si elaborano i gruppi, non vi è alcuna ambiguità sull'elaborazione di shortname , poiché non vi è alcun requisito per adattare qualsiasi forma di un nome gruppo in 12 caratteri. Pertanto, non esiste un equivalente dell'attributo SHORTUSR per i gruppi.

Ciò significa che gli esempi di sintassi descritti in [Tabella 70 a pagina 403](#) sono validi, supponendo che l'oggetto AUTHINFO sia stato configurato con gli attributi estesi e impostato su:

```
GRPFIELD(longname)  
BASEDNG(ou=groups,o=yourcompany,c=yourcountry ) CLASSGRP(groupOfNames)
```

Tabella 70. Impostazioni di autorizzazione gruppo

Comando	Nota
setmqaut -m QM -t qmgr -g ApplicationGroupA +connect	Utilizzo di GRPFIELD per risolvere
setmqaut -m QM -t qmgr -g longname=ApplicationGroupA +connect	Denominazione di un singolo attributo
setmqaut -m QM -t qmgr -g "cn=Application Group A,ou=groups,o=yourcompany,c=yourcountry" +connect	Utilizzo del DN completo

È possibile utilizzare il comando MQSC [SET AUTHREC](#) come alternativa al comando **setmqaut** precedente:

```
SET AUTHREC OBJTYPE(QMGR) GROUP('ApplicationGroupA')  
AUTHADD(connect)
```

o il comando PCF [Imposta record di autorizzazione \(MQCMD\\_SET\\_AUTH\\_REC\)](#) con l'elemento MQCACF\_GROUP\_ENTITY\_NAMES contenente la stringa:

```
"ApplicationGroupA"
```

**Importante:**

Qualunque sia il formato utilizzato per fare riferimento ad un nome, sia per l'utente che per il gruppo, deve essere possibile derivare un DN univoco.

Quindi, ad esempio, non è necessario avere due record distinti che hanno entrambi "shortu=jodoe".

Se non è possibile determinare un singolo DN univoco, OAM restituisce MQRC\_UNKNOWN\_ENTITY.

## Visualizzazione delle autorizzazioni

Vari metodi di visualizzazione dell'autorizzazione di utenti o gruppi.

### comando dspmqaut

Il metodo più semplice per visualizzare le autorizzazioni disponibili per un utente o un gruppo è quello di utilizzare il comando `dspmqaut`.

È possibile utilizzare una query su qualsiasi variazione di sintassi per identificare un utente o un gruppo. Si noti che l'output del comando ripete l'identità nel formato fornito sulla riga comandi. L'output non riporta il DN risolto completo.

Ad esempio:

```
dspmqaut -m QM -t qmgr -p johndoe
Entity johndoe has the following authorizations for object QM:
  connect
```

o

```
dspmqaut -m QM -t qmgr -p email=JohnDoe1@yourcompany.com
Entity email=JohnDoe1@yourcompany.com has the following authorizations for object QM:
  connect
```

### Comandi dmpmqaut e dmpmqcfg

Il comando `dmpmqaut` e i relativi equivalenti MQSC o PCF possono specificare il principal o il gruppo in uno qualsiasi dei formati supportati, come le tabelle `setmqaut` descritte in “Impostazione delle autorizzazioni” a [pagina 402](#). Tuttavia, a differenza di `dspmqaut`, il comando `dmpmqaut` riporta sempre il DN completo.

```
dmpmqaut -m QM -t qmgr -p jodoe
-----
profile: self
object type:qmgr
entity:cn=JohnDoe, ou=users, o=yourcompany, c=yourcountry
entity type: principal
authority: connect
```

Analogamente, il comando `dmpmqcfg`, che non ha alcun filtro sui record selezionati, mostra sempre il DN completo in un formato che può essere riprodotto in un secondo momento.

```
dmpmqcfg -m QM -x authrec
-----
SET AUTHREC PROFILE(SELF) +
  PRINCIPAL('cn=JohnDoe, ou=users, o=yourcompany, c=yourcountry') +
  OBJTYPE(QMGR)
  AUTHADD(CONNECT)
```

## Altre considerazioni sull'utilizzo dell'autorizzazione LDAP

Una breve descrizione delle modifiche a MQI (Message Queue Interface) e ad altri comandi MQSC e PCF di cui è necessario essere consapevoli quando si utilizza l'autorizzazione LDAP da IBM MQ 9.0.0.

## ADOPTCTX

Non è richiesto che le applicazioni forniscano informazioni di autenticazione o che l'attributo [ADOPTCTX](#) sia impostato su YES.

Se un'applicazione non esegue l'autenticazione esplicita o se **ADOPTCTX** è impostato su NO per l'oggetto CONNAUTH attivo, il contesto di identità associato all'applicazione viene preso dall>ID utente del sistema operativo.

Quando è necessario applicare le autorizzazioni, tale contesto viene associato a un'identità LDAP utilizzando le stesse regole dei comandi [setmqaut](#) .

## Parametri di input per chiamate MQI

[MQOPEN](#), [MQPUT1](#)e [MQSUB](#) hanno strutture che consentono di specificare un ID utente alternativo.

Se vengono utilizzati tali campi, l>ID utente di 12 caratteri viene associato ad un DN utilizzando le stesse regole dei comandi **setmqaut**, **dmpmqaute** **dspmqaut** .

MQPUT e MQPUT1 consentono inoltre ai programmi autorizzati di impostare il campo MQMD `UserIdentifier` . Il valore di questo campo non è regolato durante il processo PUT e può essere impostato su qualsiasi valore.

Come di consueto, tuttavia, è possibile utilizzare il valore **UserIdentifier** per l'autorizzazione nelle fasi successive dell'elaborazione del messaggio, ad esempio quando PUTAUT (CTX) è definito su un canale ricevente.

A quel punto, l'identificativo verrà controllato per l'autorizzazione utilizzando la configurazione del gestore code di ricezione - che può essere basato su LDAP o su SO.

## Parametri di output per chiamate MQI

Ogni volta che un ID utente viene fornito a un programma in una struttura MQI, è la versione del nome breve di 12 caratteri associata alla connessione.

Ad esempio, il valore **MQAXC.UserId** per Uscite API è il nome breve restituito dall'associazione LDAP.

## Altri comandi MQSC e PCF di gestione

I comandi che mostrano le informazioni utente nello stato dell'oggetto come `DISPLAY CONN USERID` restituiscono il nome breve di 12 caratteri associato al contesto. Il DN completo non viene visualizzato.

I comandi che consentono l'asserzione delle identità, come le regole di associazione `CHLAUTH` o i valori `MCAUSER` per i canali, possono assumere valori fino alla lunghezza massima definita per tali attributi (attualmente 64 caratteri).

La sintassi non è stata modificata. Quando l'autorizzazione è richiesta per tale identità, viene associata internamente a un DN utilizzando le stesse regole dei comandi **setmqaut**, **dmpmqaute** **dspmqaut** .

Ciò significa che il valore `MCAUSER` su una definizione di canale potrebbe non essere visualizzato come la stessa stringa di `DISPLAY CHSTATUS` ma fanno riferimento alla stessa identità.

Ad esempio:

```
DEFINE CHL(SV1) CHLTYPE(SVRCONN) MCAUSER('cn=JohnDoe')
DEFINE CHL(SV2) CHLTYPE(SVRCONN) MCAUSER('jodoe')
DEFINE CHL(SV3) CHLTYPE(SVRCONN) MCAUSER('JohnDoe1@yourcompany.com')
```

Quindi `DISPLAY CHSTATUS (*) ALL` mostra il valore `SHORTUSR`, `MCAUSER(jodoe)` per tutte le connessioni.

## Passaggio tra i modelli di autorizzazione SO e LDAP

Come passare da un metodo di autorizzazione all'altro su piattaforme differenti.

L'attributo `CONNAUTH` del gestore code punta a un oggetto `AUTHINFO`. Quando l'oggetto è di tipo `IDPWLDAP`, per l'autenticazione viene utilizzato un repository LDAP.

È ora possibile applicare un metodo di autorizzazione allo stesso oggetto, che consente di continuare con l'autorizzazione basata sul sistema operativo o di utilizzare l'autorizzazione LDAP

## Piattaforme UNIX e IBM i



Il gestore code può essere commutato in qualsiasi momento tra i modelli SO e LDAP. È possibile modificare la configurazione e renderla attiva utilizzando il comando `REFRESH SECURITY TYPE (CONNAUTH)`.

Ad esempio, se questo oggetto è già stato configurato con le informazioni di connessione per l'autenticazione:

```
ALTER AUTHINFO(MYLDAP) AUTHTYPE(IDPWLDAP) +
    AUTHORMD(SEARCHGRP) +
    BASEDNG('ou=groups,o=ibm,c=uk') +
    <other attributes>
ALTER QMGR CONNAUTH(MYLDAP)
REFRESH SECURITY
```

## Windows



Se una modifica della configurazione dell'autorizzazione implica il passaggio da un modello SO a un modello LDAP, il gestore code deve essere riavviato per rendere effettiva la modifica. Altrimenti, è possibile rendere attiva la modifica utilizzando il comando `REFRESH SECURITY TYPE (CONNAUTH)`.

## Regole di elaborazione

Quando si passa dall'autorizzazione SO all'autorizzazione LDAP, tutte le regole di autorizzazione SO esistenti che sono state impostate, diventano inattive e invisibili.

Comandi come `dmpmqaut` non visualizzano tali regole del sistema operativo. Allo stesso modo, quando si torna da LDAP al sistema operativo, tutte le autorizzazioni LDAP definite diventano inattive e invisibili, ripristinando le regole del sistema operativo originali.

Se si desidera eseguire il backup delle definizioni di un gestore code per un qualsiasi motivo, utilizzando il comando `dmpmqcfig`, tale backup conterrà solo le regole definite per il metodo di autorizzazione in vigore al momento del backup.

## Gestione LDAP

Una panoramica su come ciascuna piattaforma amministra LDAP.

Quando si utilizza l'autorizzazione LDAP, l'appartenenza del gruppo `mqm` (o equivalente) nel sistema operativo non è così importante. Essere un membro di tale gruppo controlla solo se è possibile elaborare determinati comandi della riga comandi.

In particolare, è necessario essere in tale gruppo per immettere i comandi `strmqm` e `endmqm`.

Una volta che il gestore code è in esecuzione, ci sono ora dei limiti sull'account con privilegi completi. Oltre all'ID utente della persona che emette il comando `strmqm`, gli altri utenti appartenenti al gruppo SO `mqm` (o equivalente) non ottengono privilegi speciali.

Le autorizzazioni di altri utenti si basano sui gruppi LDAP a cui appartengono. Un utilizzo non qualificato del nome gruppo `mqm` in comandi come `setmqaut` non è consentito per l'associazione ad alcun gruppo LDAP.

## UNIX piattaforme



Una volta che il gestore code è in esecuzione, l'unico account con privilegi completi è l'utente reale che ha avviato il gestore code.

L'ID mqm esiste ancora e viene utilizzato come proprietario delle risorse del sistema operativo, ad esempio i file, perché mqm è l'ID effettivo con cui è in esecuzione il gestore code. Tuttavia, l'utente mqm non sarà automaticamente in grado di eseguire attività amministrative controllate da OAM.

## IBM i



Su IBM i, gli account con privilegi automatici sono quelli che avviano il gestore code e l'ID QMQM.

Sono necessari entrambi gli ID, poiché l'ID utente che avvia il gestore code è richiesto solo per avviare il sistema. Una volta in esecuzione, i processi del gestore code dispongono solo dell'autorizzazione QMQM.

## Windows piattaforme



Su Windows, gli account con privilegi completi automaticamente sono l'utente del sistema operativo che ha avviato il gestore code e anche l'utente che esegue i processi del gestore code principale, come MUSR\_MQADMIN se il gestore code è stato avviato come un servizio Windows .

Durante l'esecuzione in modalità di autorizzazione LDAP, Windows si comporta in modo molto simile alle piattaforme UNIX . Si occupa di nomi brevi di 12 caratteri e DN completi.

## Script di esempio

Poiché è utile avere un gruppo in grado di eseguire l'amministrazione completa su un gestore code, uno script di esempio viene fornito su piattaforme UNIX come:

```
MQ_INSTALLATION_PATH/samp/bin/amqauthg.sh
```

Questo esempio prende due parametri:

- Un nome gestore code
- Un nome gruppo LDAP

L'esempio elabora i comandi `setmqaut` , concedendo l'autorizzazione completa per tutti gli oggetti. Questo è lo stesso script generato dalla procedura guidata OAM IBM MQ Explorer per i ruoli di amministrazione. Ad esempio, il codice inizia:

```
setmqaut -t q -m qmgr -n "*" +alladm +allmqi -g  
groupname
```

## Riservatezza dei messaggi

Per mantenere la riservatezza, crittografare i messaggi. Esistono vari metodi di crittografia dei messaggi in IBM MQ in base alle proprie esigenze.

La scelta di CipherSpec determina il livello di riservatezza di cui si dispone.

Se hai bisogno di protezione dei dati end-to-end a livello di applicazione per la tua infrastruttura di messaggistica point - to - point, puoi utilizzare Advanced Message Security per crittografare i messaggi o scrivere la tua uscita API o l'uscita API - crossing.

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 TLS oppure scrivere le proprie uscite di sicurezza, uscite di messaggi o programmi di uscita di invio e ricezione.

Per ulteriori informazioni su Advanced Message Security, consultare [“pianificazione per Advanced Message Security”](#) a pagina 95. L'utilizzo di TLS con IBM MQ è descritto in [“Protocolli di sicurezza TLS in IBM MQ”](#) a pagina 24. L'utilizzo dei programmi di uscita nella crittografia dei messaggi è descritto in [“Implementazione della riservatezza nei programmi di uscita utente”](#) a pagina 429.

### Informazioni correlate

[Connessione di due gestori code mediante TLS](#)

[Connessione sicura di un client a un gestore code](#)

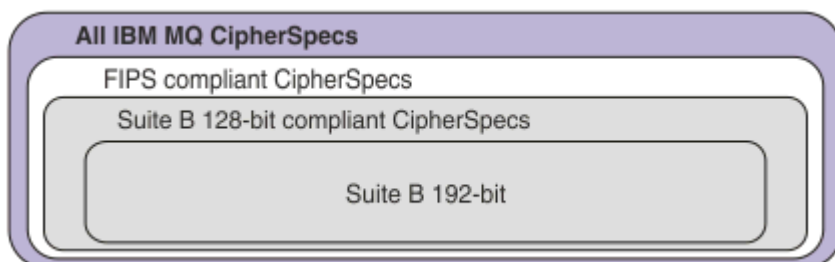
## Abilitazione di CipherSpecs

Abilitare 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 MQ sono compatibili con FIPS. Alcuni CipherSpecs compatibili con FIPS sono compatibili anche con Suite B, mentre altri, come TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA, 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:






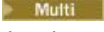





Da IBM MQ 8.0.0 Fix Pack 3 il numero di CipherSpecs supportati è stato ridotto.





Per informazioni sull'abilitazione dei CipherSpecs obsoleti, consultare [“Abilitazione di CipherSpecs obsoleti su Multiplatforms”](#) a pagina 411 o [“Abilitazione di CipherSpecs obsoleti su z/OS”](#) a pagina 412. Per un elenco di CipherSpecs che è possibile riabilitare per l'utilizzo con IBM MQ, vedere [“CipherSpecs obsoleto”](#) a pagina 412.

Le specifiche di cifratura che è possibile utilizzare con il gestore code IBM MQ vengono 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 TLS è la dimensione memorizzata nel certificato a meno che non sia determinata da CipherSpec, come indicato nella tabella.

Supporto piattaforma <a href="#">“1”</a> a pagina 410	Nome CipherSpec	Protocollo utilizzato	Integrità dati	Algoritmo di codifica	Bit di codifica	FIPS <a href="#">“2”</a> a pagina 410	Suite B
<div style="background-color: #e06666; padding: 2px;">z/OS</div> <div style="background-color: #336699; padding: 2px;">ULW</div>	TLS_RSA_WITH_AES_128_CBC_SHA	TLS 1.0	SHA-1	AES	128	Sì	No





Supporto piattaforma "1" a pagina 410	Nome CipherSpec	Protocollo utilizzato	Integrità dati	Algoritmo di codifica	Bit di codifica	FIPS "2" a pagina 410	Suite B
	TLS_RSA_WITH_AES_256_CBC_SHA "3" a pagina 410	TLS 1.0	SHA-1	AES	256	Sì	No
Tutto	ECDHE_ECDSA_AES_128_CBC_SHA256	TLS 1.2	SHA-256	AES	128	Sì	No
Tutto	ECDHE_ECDSA_AES_256_CBC_SHA384 "3" a pagina 410	TLS 1.2	SHA-384	AES	256	Sì	No
	ECDHE_ECDSA_AES_128_GCM_SHA256 "4" a pagina 410	TLS 1.2	AEAD AES-128 GCM	AES	128	Sì	128 bit
	ECDHE_ECDSA_AES_256_GCM_SHA384 "3" a pagina 410 "4" a pagina 410	TLS 1.2	AEAD AES-128 GCM	AES	256	Sì	192 bit
Tutto	ECDHE_RSA_AES_128_CBC_SHA256	TLS 1.2	SHA-256	AES	128	Sì	No
Tutto	ECDHE_RSA_AES_256_CBC_SHA384 "3" a pagina 410	TLS 1.2	SHA-384	AES	256	Sì	No
 (LTS) Tutti (V9.0.5 e successive)	ECDHE_RSA_AES_128_GCM_SHA256 "4" a pagina 410	TLS 1.2	AEAD AES-128 GCM	AES	128	Sì	No
 (LTS) Tutti (V9.0.5 e successive)	ECDHE_RSA_AES_256_GCM_SHA384 "3" a pagina 410 "4" a pagina 410	TLS 1.2	AEAD AES-128 GCM	AES	SHA384	Sì	No
 "5" a pagina 410	ECDHE_ECDSA_RC4_128_SHA256	TLS 1.2	AEAD AES-128 GCM	AES	SHA256	Sì	No
	ECDHE_ECDSA_3DES_EDE_CBC_SHA256	TLS 1.2	AEAD AES-128 GCM	3DES	SHA256	Sì	No
	ECDHE_ECDSA_NULL_SHA256	TLS 1.2	AEAD AES-128 GCM	ECDSA	SHA256	Sì	No
	ECDHE_ECDSA_AES_256_GCM_SHA384 "3" a pagina 410 "4" a pagina 410	TLS 1.2	AEAD AES-128 GCM	AES	SHA384	Sì	No

Supporto piattaforma "1" a pagina 410	Nome CipherSpec	Protocollo utilizzato	Integrità dati	Algoritmo di codifica	Bit di codifica	FIPS "2" a pagina 410	Suite B
 z/OS  ULW	TLS_RSA_WITH_AES_128_CBC_SHA256	TLS 1.2	SHA-256	AES	128	Sì	No
 z/OS  ULW	TLS_RSA_WITH_AES_256_CBC_SHA256 "3" a pagina 410	TLS 1.2	SHA-256	AES	256	Sì	No
Tutti (V9.0.5 e versioni successive e 9.0 LTS)	TLS_RSA_WITH_AES_128_GCM_SHA256 "4" a pagina 410	TLS 1.2	AEAD AES-128 GCM	AES	128	Sì	No
Tutti (V9.0.5 e versioni successive e 9.0 LTS)	TLS_RSA_WITH_AES_256_GCM_SHA384 "3" a pagina 410 "4" a pagina 410	TLS 1.2	AEAD AES-128 GCM	AES	256	Sì	No

**Note:**

1. Se non si nota una piattaforma specifica, il CipherSpec è disponibile su tutte le piattaforme. Per un elenco delle piattaforme a cui è attinente ciascuna icona di piattaforma, consultare [Release and platform icons in the product documentation](#).
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. Questa CipherSpec non può essere utilizzata per proteggere una connessione da IBM MQ Explorer a un gestore code a meno che non vengano applicati i file di politiche senza restrizioni appropriati al JRE utilizzato dall'Explorer.
4. Seguendo un suggerimento di GSKit, GCM CipherSpecs ha una restrizione che indica che dopo l'invio di 224.5 record TLS, utilizzando la stessa chiave di sessione, la connessione viene terminata con il messaggio AMQ9288.

 Windows  Linux Per evitare che accada questo errore, non utilizzare cifrature GCM, abilitare la reimpostazione della chiave segreta o avviare il gestore code o client IBM MQ con la variabile di ambiente GSK\_ENFORCE\_GCM\_RESTRICTION=GSK\_FALSE impostata.

**Note:**

- È necessario impostare questa variabile di ambiente su entrambi i lati della connessione e si applica sia al client per le connessioni del gestore code che al gestore code per le connessioni del gestore code.
- Questa istruzione si applica solo alle librerie GSKit, quindi interessa anche i client .NET non gestiti ma non i client Java o .NET gestiti.

Questa restrizione non si applica a IBM MQ for z/OS.

**Importante:** La restrizione GCM è attiva, indipendentemente dalla modalità FIPS utilizzata.

5.  IBM i I CipherSpec indicati come supportati su IBM i, si applicano alle versioni 7.2 e 7.3 di IBM i.

## Abilitazione di CipherSpecs obsoleti su Multiplatforms

Multi

Per impostazione predefinita, non è consentito specificare una CipherSpec obsoleta su una definizione di canale. Se si tenta di specificare un CipherSpec obsoleto su Multiplatforms, si riceve il messaggio AMQ8242: la definizione SSLCIPH non è corretta e PCF restituisce MQRCCF\_SSL\_CIPHER\_SPEC\_ERROR.

Non è possibile avviare un canale con un CipherSpec obsoleto. Se si tenta di eseguire tale operazione con un CipherSpec obsoleto, il sistema restituisce *MQCC\_FAILED (2)*, insieme a un **Reason** di *MQRC\_SSL\_INITIALIZATION\_ERROR (2393)* al client.

È possibile riattivare uno o più CipherSpecs obsoleti per definire i canali, al runtime sul server, impostando la variabile di ambiente *AMQ\_SSL\_WEAK\_CIPHER\_ENABLE*.

La variabile di ambiente *AMQ\_SSL\_WEAK\_CIPHER\_ENABLE* accetta:

- Un singolo nome CipherSpec oppure
- Un elenco separato da virgole di nomi IBM MQ CipherSpec da riabilitare o
- Il valore speciale di *ALL*, che rappresenta tutti i CipherSpecs.

Ad esempio, se si desidera riabilitare *ECDHE\_RSA\_RC4\_128\_SHA256*, impostare la seguente variabile di ambiente:

```
AMQ_SSL_WEAK_CIPHER_ENABLE=ECDHE_RSA_RC4_128_SHA256
```

oppure, in alternativa, modificare la sezione SSL nel file *qm.ini*, impostando:

```
SSL
AllowWeakCipherSpec=ECDHE_RSA_RC4_128_SHA256
```

### Abilitazione di CipherSpec CipherSpecs

Oltre a emettere *AMQ\_SSL\_WEAK\_CIPHER\_ENABLE* o *AllowWeakCipherSpec*, come descritto nel testo precedente, è necessario impostare la variabile di ambiente *AMQ\_SSL\_V3\_ENABLE=1* o impostare *AllowSSLV3=Y*, per continuare a utilizzare i CipherSpec SSLv3 CipherSpecs, come descritto in [Deprecazione: protocollo SSLv3](#).

Ad esempio, se si desidera riabilitare *RC4\_MD5\_US*, impostare le seguenti variabili di ambiente:

```
AMQ_SSL_V3_ENABLE=1
AMQ_SSL_WEAK_CIPHER_ENABLE=RC4_MD5_US
```

o, in alternativa, modificare la stanza SSL nel file *qm.ini*, impostando:

```
SSL
AllowSSLV3=Y
AllowWeakCipherSpec=RC4_MD5_US
```



**Attenzione:** Le seguenti informazioni relative a *TLS\_V1* si applicano solo da IBM MQ 9.0.0 Fix Pack 3 o IBM MQ 9.0.5.

Oltre a emettere *AMQ\_TLS\_WEAK\_CIPHER\_ENABLE* o *AllowWeakCipherSpec*, è necessario impostare la variabile di ambiente *AMQ\_TLS\_V1\_ENABLE=1* o impostare *AllowTLV1=Y* per continuare a utilizzare *TLSv1* CipherSpecs obsoleti.

Ad esempio, se si desidera riabilitare *TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA*, impostare le seguenti variabili di ambiente:

```
AMQ_TLS_V1_ENABLE=1
AMQ_TLS_WEAK_CIPHER_ENABLE=TLS_RSA_WITH_AES_128_CBC_SHA
```

o, in alternativa, modificare la stanza SSL nel file `qm.ini` , impostando:

```
SSL
AllowTLSV1=Y
AllowWeakCipherSpec=TLS_RSA_WITH_AES_128_CBC_SHA
```

## Abilitazione di CipherSpecs obsoleti su z/OS



Per impostazione predefinita, non è consentito specificare una CipherSpec obsoleta su una definizione di canale. Se si tenta di specificare un CipherSpec obsoleto su z/OS, si riceve il messaggio [CSQM102E](#) o il messaggio [CSQX674E](#).

Per abilitare le specifiche di cifratura deboli (obsolete), è necessario definire la seguente istruzione DD nel JCL CHINIT:

```
JCL: //CSQXWEAK DD DUMMY
```

Per abilitare il protocollo SSLv3 obsoleto, è necessario definire anche la seguente istruzione DD nel JCL CHINIT:

```
JCL: //CSQXSSL3 DD DUMMY
```

Per disattivare TLS 1.0 , utilizzare la seguente istruzione:

```
JCL: //TLS10OFF DD DUMMY
```

Se non si desidera negoziare con il listener utilizzando specifiche di cifratura deboli o interrotte, è necessario definire la seguente istruzione DD nel JCL CHINIT:

```
JCL: //WCIPSOFF DD DUMMY
```

Se si desidera negoziare solo con il listener utilizzando le specifiche di cifratura elencate nell'elenco di specifiche di cifratura predefinite di **System SSL** , è necessario definire la seguente istruzione DD nel JCL CHINIT:

```
JCL: //GSKDCIPS DD DUMMY
```

### Concetti correlati

[“Certificati digitali e compatibilità CipherSpec in IBM MQ”](#) a pagina 42

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 MQ.

### Informazioni correlate

[Definire il canale](#)


















[MODIFICA CANALE](#)

## CipherSpecs obsoleto

Un elenco di CipherSpecs obsoleti che è possibile utilizzare con IBM MQ , se necessario.

Per informazioni sull'abilitazione dei CipherSpecs obsoleti, consultare [“Abilitazione di CipherSpecs obsoleti su Multiplatforms”](#) a pagina 411 o [“Abilitazione di CipherSpecs obsoleti su z/OS”](#) a pagina 412.

I CipherSpecs obsoleti che puoi utilizzare con il supporto TLS IBM MQ sono elencati nella seguente tabella.

Supporto piattaforma "1" a pagina 415	Nome CipherSpec	Protocollo utilizzato	Integrità dati	Algoritmo di codifica	Bit di codifica	FIPS "2" a pagina 415	Suite B	Aggiorna quando obsoleto
	AES_SHA_US	SSL 3.0	SHA-1	AES	128	No	No	9.0.0.0
Tutto	DES_SHA_EXPORT "3" a pagina 415 "8" a pagina 415	SSL 3.0	SHA-1	DES	56	No	No	9.0.0.0
  	DES_SHA_EXPORT1024 "4" a pagina 415	SSL 3.0	SHA-1	DES	56	No	No	9.0.0.0
  	FIPS_WITH_DES_CBC_SHA	SSL 3.0	SHA-1	DES	56	No "6" a pagina 415	No	9.0.0.0
  	FIPS_WITH_3DES_EDE_CBC_SHA	SSL 3.0	SHA-1	3DES	168	No "7" a pagina 415	No	9.0.0.1 e 9.0.1
Tutto	NULL_MD5	SSL 3.0	MD5	Nessuno	0	No	No	9.0.0.1
Tutto	NULL_SHA	SSL 3.0	SHA-1	Nessuno	0	No	No	9.0.0.1
Tutto	RC2_MD5_EXPORT "3" a pagina 415 "8" a pagina 415	SSL 3.0	MD5	RC2	40	No	No	9.0.0.0
Tutto	RC4_MD5_EXPORT "3" a pagina 415	SSL 3.0	MD5	RC4	40	No	No	9.0.0.0
Tutto	RC4_MD5_US	SSL 3.0	MD5	RC4	128	No	No	9.0.0.0
Tutto	RC4_SHA_US "8" a pagina 415	SSL 3.0	SHA-1	RC4	128	No	No	9.0.0.0
  	RC4_56_SHA_EXPORT1024 "4" a pagina 415	SSL 3.0	SHA-1	RC4	56	No	No	9.0.0.0
Tutto	TRIPLE_DES_SHA_US "8" a pagina 415	SSL 3.0	SHA-1	3DES	168	No	No	9.0.0.1 e 9.0.1
	TLS_RSA_EXPORT_WITH_RC2_40_MD5	TLS 1.0	MD5	RC2	40	No	No	9.0.0.0
	TLS_RSA_EXPORT_WITH_RC4_40_MD5 "3" a pagina 415	TLS 1.0	MD5	RC4	40	No	No	9.0.0.0
Tutto	TLS_RSA_WITH_DES_CBC_SHA	TLS 1.0	SHA-1	DES	56	No "5" a pagina 415	No	9.0.0.0
	TLS_RSA_WITH_NULL_MD5	TLS 1.0	MD5	Nessuno	0	No	No	9.0.0.1
	TLS_RSA_WITH_NULL_SHA	TLS 1.0	SHA-1	Nessuno	0	No	No	9.0.0.1

Supporto piattaforma "1" a pagina 415	Nome CipherSpec	Protocollo utilizzato	Integrità dati	Algoritmo di codifica	Bit di codifica	FIPS "2" a pagina 415	Suite B	Aggiornato quando obsoleto
IBM i	TLS_RSA_WITH_RC4_128_MD5	TLS 1.0	MD5	RC4	128	No	No	9.0.0.0
Windows UNIX Linux	ECDHE_ECDSA_NULL_SHA256	TLS 1.2	SHA-1	Nessuno	0	No	No	9.0.0.1
Windows UNIX Linux	ECDHE_ECDSA_RC4_128_SHA256	TLS 1.2	SHA-1	RC4	128	No	No	9.0.0.0
Windows UNIX IBM i Linux	ECDHE_RSA_NULL_SHA256	TLS 1.2	SHA-1	Nessuno	0	No	No	9.0.0.1
Windows UNIX IBM i Linux	ECDHE_RSA_RC4_128_SHA256	TLS 1.2	SHA-1	RC4	128	No	No	9.0.0.0
Windows UNIX Linux	TLS_RSA_WITH_NULL_NULL	TLS 1.2	Nessuno	Nessuno	0	No	No	9.0.0.1
Tutto	TLS_RSA_WITH_NULL_SHA256	TLS 1.2	SHA-256	Nessuno	0	No	No	9.0.0.1
Windows UNIX Linux	TLS_RSA_WITH_RC4_128_SHA256	TLS 1.2	SHA-1	RC4	128	No	No	9.0.0.0
Tutto	TLS_RSA_WITH_3DES_EDE_CBC_SHA <sup>9</sup> a pagina 415	TLS 1.0	SHA-1	3DES	168	Sì	No	9.0.0.1 e 9.0.1
Windows UNIX Linux	ECDHE_ECDSA_3DES_EDE_CBC_SHA256 <sup>9</sup> a pagina 415	TLS 1.2	SHA-1	3DES	168	Sì	No	9.0.0.1 e 9.0.1
Windows UNIX IBM i Linux	ECDHE_RSA_3DES_EDE_CBC_SHA256 <sup>9</sup> a pagina 415	TLS 1.2	SHA-1	3DES	168	Sì	No	9.0.0.1 e 9.0.1

Supporto piattaforma "1" a pagina 415	Nome CipherSpec	Protocollo utilizzato	Integrità dati	Algoritmo di codifica	Bit di codifica	FIPS "2" a pagina 415	Suite B	Aggiorna quando obsoleto
---------------------------------------	-----------------	-----------------------	----------------	-----------------------	-----------------	-----------------------	---------	--------------------------

**Note:**

1. Se non si nota una piattaforma 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. Queste CipherSpec non sono più supportate da IBM MQ classes for Java o IBM MQ classes for JMS. Per ulteriori informazioni, consultare [CipherSpec e CipherSuite SSL/TLS in IBM MQ classes for Java](#) o [CipherSpec e CipherSuite SSL/TLS in IBM MQ classes for JMS](#).
9. 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.

**Concetti correlati**

[“Certificati digitali e compatibilità CipherSpec in IBM MQ” a pagina 42](#)

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 MQ.

**Informazioni correlate**

[Definire il canale](#)

[MODIFICA CANALE](#)

**Acquisizione di informazioni su CipherSpecs utilizzando IBM MQ Explorer**

È possibile utilizzare IBM MQ Explorer per visualizzare le descrizioni di CipherSpecs.

Utilizzare la seguente procedura per ottenere informazioni su CipherSpecs in [“Abilitazione di CipherSpecs” a pagina 408](#):

1. Aprire IBM 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 TLS, il proprio 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 IBM MQ , quindi i nuovi CipherSpecs non diventano disponibili dopo la spedizione.

Per le piattaforme in cui il sistema operativo fornisce il supporto TLS, il tuo sistema potrebbe supportare nuovi CipherSpecs non inclusi in [“Abilitazione di CipherSpecs”](#) a pagina 408. 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 TLS che sia valido e supportato dalla versione di TLS che il tuo sistema sta eseguendo.

### IBM i

Una stringa di due caratteri che rappresenta un valore esadecimale.

Per ulteriori informazioni sui valori consentiti, vedere il punto tre nella sezione [Note sull'uso di Impostazione delle informazioni sui caratteri per una sessione sicura](#).



**Attenzione:** Non si dovrebbero specificare valori di cifratura esadecimale in SSLCIPH, poiché dal valore non è chiaro quale cifratura verrà utilizzata e la scelta del protocollo da utilizzare è indeterminata. L'uso di valori di cifratura esadecimale può causare errori di mancata corrispondenza di CipherSpec .

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 quattro caratteri che rappresenta un valore esadecimale. I codici esadecimale corrispondono ai valori definiti nel protocollo TLS.

Per ulteriori informazioni, fare riferimento a [Definizioni suite di cifratura](#) dove è presente un elenco di tutte le specifiche di cifratura TLS 1.0, TLS 1.2e TLS 1.3 nel formato di codici esadecimale a 4 cifre.

## Considerazioni per i cluster IBM MQ

Con i cluster IBM MQ è più sicuro utilizzare i nomi CipherSpec in [“Abilitazione di CipherSpecs”](#) a pagina 408. 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/TLS e cluster”](#) a pagina 436.

## Specifica di un CipherSpec per un IBM MQ MQI client

Si dispone di tre opzioni per specificare un CipherSpec per un IBM MQ MQI client.

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 MQCONN.
- Utilizzo di Active Directory (su sistemi Windows con supporto Active Directory)

## Specifica di una CipherSuite con IBM MQ classes for Java e IBM MQ classes for JMS

IBM MQ classes for Java e IBM MQ classes for JMS specificano CipherSuites in modo diverso rispetto ad altre piattaforme.



Per informazioni su come specificare una CipherSuite con IBM MQ classes for Java, consultare il supporto [TLS \(Transport Layer Security\) per Java](#)

Per informazioni su come specificare una CipherSuite con IBM MQ classes for JMS, consultare [Using Transport Layer Security \(TLS\) with IBM MQ classes for JMS](#)

## Specifica di un CipherSpec per IBM MQ.NET

Per IBM MQ.NET è possibile specificare CipherSpec utilizzando la classe MQEnvironment oppure utilizzando MQC.SSL\_CIPHER\_SPEC\_PROPERTY nella tabella hash delle proprietà di connessione.

Per informazioni sulla specifica di CipherSpec per il client non gestito .NET , consultare [Abilitazione di TLS per il client .NET non gestito](#)

Per informazioni sulla specifica di un CipherSpec per il client gestito .NET , consultare il supporto [CipherSpec per il client .NET gestito](#)

## **z/OS** Utilizzo di AT - TLS con IBM MQ for z/OS

AT-TLS (Application Transparent Transport Layer Security) fornisce il supporto TLS per applicazioni z/OS senza che tali applicazioni debbano implementare il supporto TLS o anche essere consapevoli del fatto che TLS viene utilizzato. AT - TLS è disponibile solo su z/OS.

AT - TLS può essere utilizzato con tutte le versioni di IBM MQ for z/OS.

Prima di utilizzare AT - TLS con IBM MQ for z/OS, assicurati di comprendere il [“Limitazioni”](#) a pagina 419 coinvolto.

Per utilizzare [Application Transparent Transport Layer Security](#) , definire le istruzioni della politica contenenti una serie di regole utilizzate da z/OS Communications Server per decidere quali connessioni TCP/IP hanno TLS abilitato in modo trasparente.

IBM MQ for z/OS ha la propria implementazione TLS, che richiede che i canali abbiano il parametro SSLCIPH configurato con un CipherSpecsupportato.

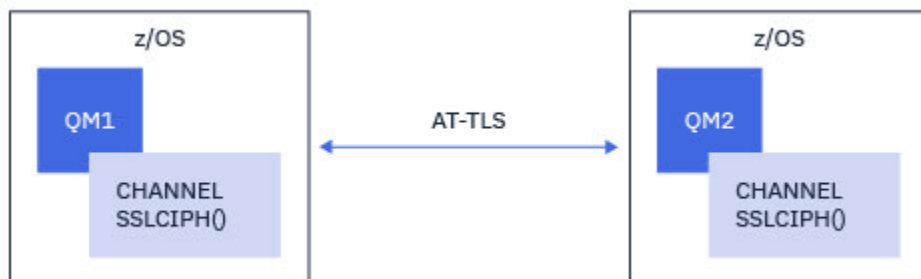
Quando si decide di abilitare TLS su un canale, l'amministratore di IBM MQ può decidere di utilizzare AT - TLS o IBM MQ TLS. La decisione viene spesso presa in base all'utilizzo di AT - TLS per altri middleware o a causa delle implicazioni delle prestazioni. Per un confronto di base delle prestazioni di AT - TLS e IBM MQ TLS, consultare [MP16: Capacity Planning and Tuning for IBM MQ for z/OS](#).

## Scenari

L'utilizzo di AT - TLS con IBM MQ è supportato nei seguenti scenari:

### Scenario 1

Tra due gestori code IBM MQ for z/OS in cui entrambi i lati del canale utilizzano AT - TLS. Ovvero, nessuno dei due canali specifica l'attributo SSLCIPH. Questo approccio può essere utilizzato con qualsiasi canale di messaggi.

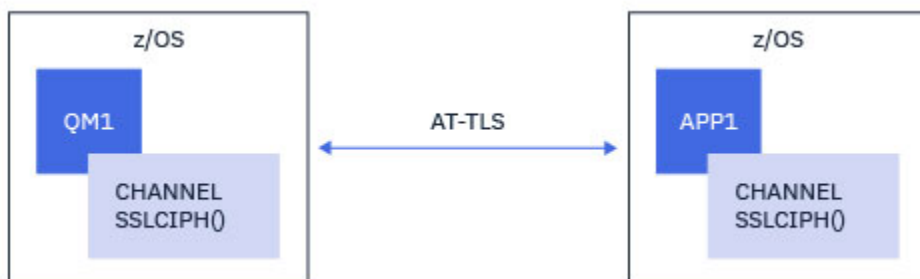


L'implementazione di questo scenario consiste nella definizione di due politiche AT - TLS, una per ogni lato del canale. Queste politiche sono uguali a quelle utilizzate con lo [Scenario 3](#).

Ad esempio, se il canale è stato modificato dall'utilizzo di un singolo CipherSpec all'utilizzo di AT - TLS, il canale in entrata utilizzerà la politica da ["Configurazione di AT - TLS su un canale in uscita in un gestore code IBM MQ for Multiplatforms utilizzando un singolo, denominato CipherSpec"](#) a pagina 420 e il canale in entrata utilizzerà la politica da ["Configurazione di AT - TLS su un canale in entrata da un gestore code IBM MQ for Multiplatforms utilizzando un singolo, denominato CipherSpec"](#) a pagina 423.

## Scenario 2

Tra un gestore code IBM MQ for z/OS e un'applicazione client IBM MQ Java in esecuzione su z/OS dove entrambi i lati del canale utilizzano AT - TLS. In altre parole, né il canale di connessione server, né il canale di connessione client specificano l'attributo SSLCIPH.

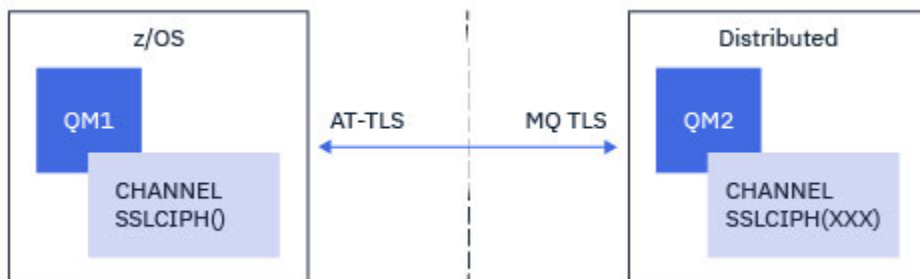


L'implementazione di questo scenario consiste nella definizione di due politiche AT - TLS, una per ogni lato del canale. Queste politiche sono uguali a quelle utilizzate con lo [Scenario 3](#).

Ad esempio, se il canale è stato modificato dall'utilizzo di un singolo CipherSpec all'utilizzo di AT - TLS, il canale di connessione client utilizzerà la politica da ["Configurazione di AT - TLS su un canale in uscita in un gestore code IBM MQ for Multiplatforms utilizzando un singolo, denominato CipherSpec"](#) a pagina 420 e il canale di connessione server utilizzerà la politica da ["Configurazione di AT - TLS su un canale in entrata da un gestore code IBM MQ for Multiplatforms utilizzando un singolo, denominato CipherSpec"](#) a pagina 423.

## Scenario 3

Tra un gestore code IBM MQ for z/OS e un gestore code in esecuzione su IBM MQ for Multiplatforms, dove il gestore code IBM MQ for z/OS utilizza AT - TLS e il gestore code IBM MQ for Multiplatforms utilizza IBM MQ TLS. Ciò si applica a tutti i tipi di canale di messaggi diversi da mittente e destinatario del cluster.

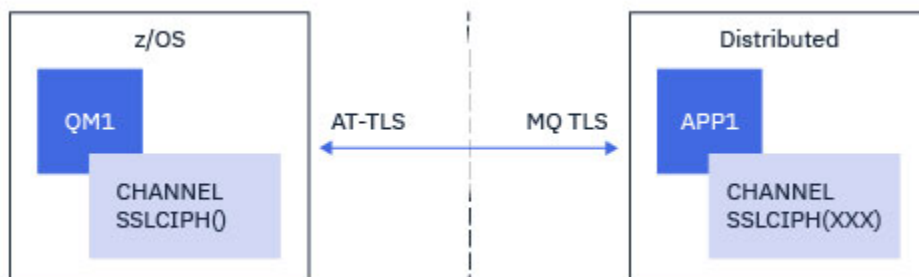


Consultare ["Configurazione di AT - TLS su un canale in uscita in un gestore code IBM MQ for Multiplatforms utilizzando un singolo, denominato CipherSpec"](#) a pagina 420 per una configurazione AT - TLS di esempio per i canali in uscita dal gestore code IBM MQ for z/OS al gestore code IBM MQ for Multiplatforms e ["Configurazione di AT - TLS su un canale in entrata da un gestore code IBM MQ for Multiplatforms utilizzando un singolo, denominato CipherSpec"](#) a pagina 423 per una configurazione AT - TLS di esempio per i canali in entrata dal gestore code IBM MQ for Multiplatforms al gestore code IBM MQ for z/OS .

La stessa configurazione AT - TLS può essere utilizzata quando entrambi i gestori code si trova su z/OS, ma il gestore code sul lato destro non è stato configurato per utilizzare AT - TLS.

#### Scenario 4

Tra un gestore code IBM MQ for z/OS e un'applicazione client in esecuzione su IBM MQ for Multiplatforms, dove il gestore code IBM MQ for z/OS utilizza AT - TLS e l'applicazione client utilizza IBM MQ TLS specificando l'attributo SSLCIPH con un singolo, denominato CipherSpec.



Questo scenario richiede una singola politica AT - TLS che soddisfi gli stessi requisiti di quelli utilizzati da un canale di messaggi in entrata; consultare [“Configurazione di AT - TLS su un canale in entrata da un gestore code IBM MQ for Multiplatforms utilizzando un singolo, denominato CipherSpec”](#) a pagina 423.

La stessa configurazione AT - TLS può essere utilizzata quando l'applicazione client è un'applicazione Java ed è anche in esecuzione su z/OS, ma non è stata configurata per utilizzare AT - TLS.

#### Limitazioni

IBM MQ for z/OS non è consapevole di AT - TLS, quindi ci sono diverse limitazioni che si applicano agli scenari precedenti:

- AT - TLS in combinazione con IBM MQ TLS non funziona con i canali mittente cluster e ricevente cluster.
- I gestori code IBM MQ for z/OS non sono consapevoli del fatto che stanno utilizzando AT - TLS e non ricevono alcuna informazione sul certificato dal loro gestore code o client partner. Pertanto, i seguenti attributi non hanno alcun effetto sul lato z/OS di un canale che utilizza AT - TLS:
  - Attributi SSLCAUTH e canale SSLPEER
  - attributo Gestore code SSLRKEYC
  - Attributi SSLPEERMAP delle regole CHLAUTH
- L'utilizzo della rinegoziazione della chiave segreta TLS richiede che entrambi i lati del canale utilizzino IBM MQ TLS. Pertanto, un gestore code o un client IBM MQ for Multiplatforms non deve avere la rinegoziazione della chiave segreta TLS abilitata se si connette a un gestore code IBM MQ for z/OS utilizzando AT - TLS.

Per disabilitare la rinegoziazione della chiave segreta TLS per un gestore code, impostare il parametro SSLRKEYC del gestore code su 0. Per un client, impostare il parametro pertinente su 0 in base al tipo client. Per dettagli su questa procedura, consultare [“Reimpostazione delle chiavi segrete SSL e TLS”](#) a pagina 427.

#### Istruzioni di configurazione AT - TLS

AT - TLS è configurato utilizzando una serie di istruzioni. Quelli utilizzati negli scenari documentati in questo argomento sono:

##### Regola TTL

Specifica una serie di criteri per la corrispondenza di una connessione TCP/IP a una configurazione TLS. Questo a sua volta fa riferimento agli altri tipi di istruzione.

### **TTLSTLSGroupAction**

Specifica se il `TTLSTLSRule` di riferimento è abilitato o meno.

### **TTLSEnvironmentAction**

Specifica la configurazione dettagliata per il riferimento `TTLSTLSRule` e fa riferimento a un numero di altre istruzioni.

### **TTLSTLSKeyringParms**

Fa riferimento al keyring che deve essere utilizzato da AT - TLS.

### **TTLSTLSCipherParms**

Definisce le suite di crittografia da utilizzare.

### **TTLSEnvironmentAdvancedParametri**

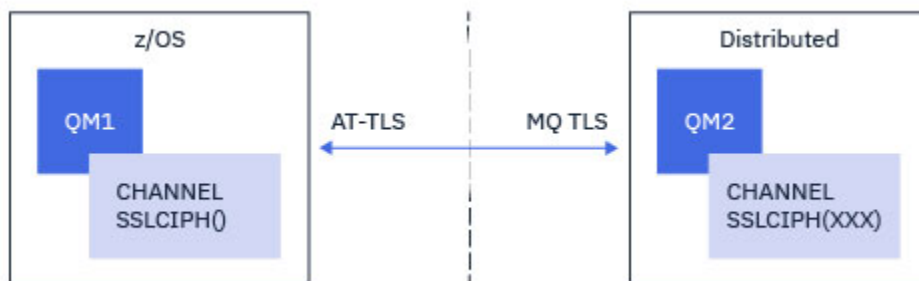
Definisce quali protocolli TLS o SSL sono abilitati.



**Attenzione:** Ci sono altre istruzioni della politica AT - TLS con AT - TLS che non sono documentate qui e potrebbero essere utilizzate con IBM MQ a seconda della necessità. Tuttavia, IBM MQ è stato testato solo con le politiche descritte in questo argomento.

## ***Configurazione di AT - TLS su un canale in uscita in un gestore code IBM MQ for Multiplatforms utilizzando un singolo, denominato CipherSpec***

Come si imposta AT - TLS su un canale in uscita da un gestore code IBM MQ for z/OS a un gestore code IBM MQ for Multiplatforms . In questo caso, il canale sul gestore code z/OS è un canale mittente che non ha l'attributo `SSLCIPH` impostato e il canale sul gestore code nonz/OS è un canale ricevente con l'attributo `SSLCIPH` impostato su un singolo, denominato `CipherSpec`.



In questo esempio, una coppia di canali mittente - destinatario esistente, che utilizza TLS 1.2 `TLS_RSA_WITH_AES_256_GCM_SHA384` CipherSpec verrà regolata in modo che il canale mittente utilizzi AT - TLS invece di IBM MQ TLS.

Altri protocolli TLS e CipherSpecs possono essere utilizzati apportando modifiche minori alla configurazione. È possibile utilizzare altri tipi di canali di messaggi, ad eccezione dei canali mittente del cluster e ricevente del cluster, senza alcuna modifica alla configurazione AT - TLS.

## **Procedura**

### **Passo 1: arresta il canale**

### **Passo 2: crea e applica una politica AT - TLS**

È necessario creare le seguenti istruzioni AT - TLS per questo scenario:

1. Un'istruzione `TTLSTLSRule` per associare le connessioni in uscita dallo spazio di indirizzo dell'iniziatore di canali all'indirizzo IP e al numero di porta del canale del destinatario di destinazione. Questi valori devono corrispondere alle informazioni utilizzate nel `CONNNAME` del canale mittente. In questo caso, è stato incluso un ulteriore filtro per corrispondere a un nome lavoro dell'iniziatore di canali specifico.

```

TTLSSRule                CSQ1-T0-REMOTE
{
  LocalAddr               ALL
  RemoteAddr              123.456.78.9
  RemotePortRange         1414
  Jobname                  CSQ1CHIN
  Direction                OUTBOUND
  TTLSSGroupActionRef     CSQ1-GROUP-ACTION
  TTLSEnvironmentActionRef CSQ1-OUTBOUND-ENVIRONMENT-ACTION
}

```

La regola precedente corrisponde alle connessioni dirette all'indirizzo IP 123.456.78.9 sulla porta 1414 dal lavoro CSQ1CHIN .

Opzioni di filtraggio più avanzate sono descritte in [TTLSSRule](#).

- Un'istruzione [TTLSSGroupAction](#) che abilita la regola. [TTLSSRule](#) fa riferimento al [TTLSSGroupAction](#) utilizzando la proprietà **TTLSSGroupActionRef** .

```

TTLSSGroupAction         CSQ1-GROUP-ACTION
{
  TTLSEnabled             ON
}

```

- Un'istruzione [TTLSEnvironmentAction](#) associata a [TTLSSRule](#) dalla proprietà **TTLSEnvironmentActionRef** . Un [TTLSEnvironmentAction](#) configura l'ambiente TLS e specifica quale keyring utilizzare.

```

TTLSEnvironmentAction    CSQ1-OUTBOUND-ENVIRONMENT-ACTION
{
  HandshakeRole           CLIENT
  TTLSSKeyringParmsRef    CSQ1-KEYRING
  TTLSCipherParmsRef      CSQ1-CIPHERPARM
  TTLSEnvironmentAdvancedParmsRef CSQ1-ENVIRONMENT-ADVANCED
}

```

- Un'istruzione [TTLSSKeyringParms](#) associata alla [TTLSEnvironmentAction](#) dalla proprietà **TTLSSKeyringParmsRef** e definisce il keyring utilizzato da AT - TLS.

Il key ring deve contenere i certificati ritenuti attendibili dal gestore code nonz/OS remoto. Questo keyring può essere definito nello stesso modo di un keyring utilizzato dall'iniziatore del canale; consultare [“Configurazione del sistema z/OS per utilizzare TLS”](#) a pagina 248.

```

TTLSSKeyringParms        CSQ1-KEYRING
{
  Keyring                  MQCHIN/CSQ1RING
}

```

- Un'istruzione [TTLSCipherParms](#) associata a [TTLSEnvironmentAction](#) dalla proprietà **TTLSCipherParmsRef** .

Questa istruzione deve contenere un nome suite di cifratura singolo che deve essere l'equivalente del nome IBM MQ CipherSpec utilizzato nel canale ricevente di destinazione.

**Nota:** I nomi della suite di cifratura AT - TLS non corrispondono necessariamente ai nomi IBM MQ CipherSpec . Tuttavia, è possibile trovare il nome della suite di cifratura AT - TLS che corrisponde a un nome IBM MQ CipherSpec ricercando il nome IBM MQ CipherSpec dalla seguente tabella e facendo riferimento incrociato alla colonna di codice di quattro caratteri con la colonna di caratteri espansi dalla Tabella 2 nell'argomento [TTLSCipherParms](#) .

Tabella 71. Converti da codici di quattro caratteri a nomi CipherSpec

Codice di quattro caratteri	Protocollo	Abilitato per impostazione predefinita.	Nome CipherSpec
0001	SSL 3.0	No	NULL_MD5
0002	SSL 3.0	No	NULL_SHA
0003	SSL 3.0	No	RC4_MD5_EXPORT
0004	SSL 3.0	No	RC4_MD5_US
0005	SSL 3.0	No	RC4_SHA_US
0006	SSL 3.0	No	RC2_MD5_EXPORT
0008	SSL 3.0	No	DES_SHA_EXPORT
0009	TLS 1.0	Sì	TLS_RSA_WITH_DES_CBC_SHA
000A	SSL 3.0	No	TRIPLE_DES_SHA_US
000A	TLS 1.0	Sì	TLS_RSA_WITH_3DES_EDE_CBC_SHA
002F	TLS 1.0	Sì	TLS_RSA_WITH_AES_128_CBC_SHA
0035	TLS 1.0	Sì	TLS_RSA_WITH_AES_256_CBC_SHA
003B	TLS 1.2	Sì	TLS_RSA_WITH_NULL_SHA256
003C	TLS 1.2	Sì	TLS_RSA_WITH_AES_128_CBC_SHA256
003D	TLS 1.2	Sì	TLS_RSA_WITH_AES_256_CBC_SHA256
C023	TLS 1.2	Sì	ECDHE_ECDSA_AES_128_CBC_SHA256
C024	TLS 1.2	Sì	ECDHE_ECDSA_AES_256_CBC_SHA384
C027	TLS 1.2	Sì	ECDHE_RSA_AES_128_CBC_SHA256
C028	TLS 1.2	Sì	ECDHE_RSA_AES_256_CBC_SHA384

```
TTLSCipherParms      CSQ1-CIPHERPARM
{
  V3CipherSuites      TLS_RSA_WITH_AES_256_GCM_SHA384
}
```

6. Un'istruzione `TTLSEnvironmentAdvancedParms` è associata alla `TTLSEnvironmentAction` dalla proprietà **`TTLSEnvironmentAdvancedParmsRef`**.

Questa istruzione può essere utilizzata per specificare quali protocolli SSL e TLS sono abilitati. Con IBM MQ, è necessario abilitare solo il singolo protocollo che corrisponde al nome della suite di cifratura utilizzato nell'istruzione `TTLSCipherParms`.

```
TTLSEnvironmentAdvancedParms CSQ1-ENVIRONMENT-ADVANCED
{
  SSLv3          OFF
  TLSv1          OFF
  TLSv1.1        OFF
  SecondaryMap   OFF
  TLSv1.2        ON
  TLSv1.3        OFF
}
```

La serie completa di istruzioni è la seguente e deve essere applicata all'agent della politica:

```

TTLSSRule                                CSQ1-T0-REMOTE
{
  LocalAddr                               ALL
  RemoteAddr                              123.456.78.9
  RemotePortRange                         1414
  Jobname                                 CSQ1CHIN
  Direction                               OUTBOUND
  TLSGroupActionRef                       CSQ1-GROUP-ACTION
  TTLEnvironmentActionRef                 CSQ1-OUTBOUND-ENVIRONMENT-ACTION
}

TLSGroupAction                            CSQ1-GROUP-ACTION
{
  TTLEnabled                              ON
}

TTLEnvironmentAction                      CSQ1-OUTBOUND-ENVIRONMENT-ACTION
{
  HandshakeRole                           CLIENT
  TLSKeyringParmsRef                      CSQ1-KEYRING
  TLSCipherParmsRef                       CSQ1-CIPHERPARM
  TTLEnvironmentAdvancedParmsRef          CSQ1-ENVIRONMENT-ADVANCED
}

TLSKeyringParms                           CSQ1-KEYRING
{
  Keyring                                 MQCHIN/CSQ1RING
}

TLSCipherParms                            CSQ1-CIPHERPARM
{
  V3CipherSuites                          TLS_RSA_WITH_AES_256_GCM_SHA384
}

TTLEnvironmentAdvancedParms               CSQ1-ENVIRONMENT-ADVANCED
{
  SSLv3                                    OFF
  TLSv1                                    OFF
  TLSv1.1                                  OFF
  SecondaryMap                             OFF
  TLSv1.2                                  ON
  TLSv1.3                                  OFF
}

```

### Passo 3: rimuovi SSLCIPH dal canale di z/OS

Rimuovere la CipherSpec dal canale z/OS utilizzando il seguente comando:

```
ALTER CHANNEL(channel-name) CHLTYPE(SDR) SSLCIPH('')
```

### Passo 4: avvia canale

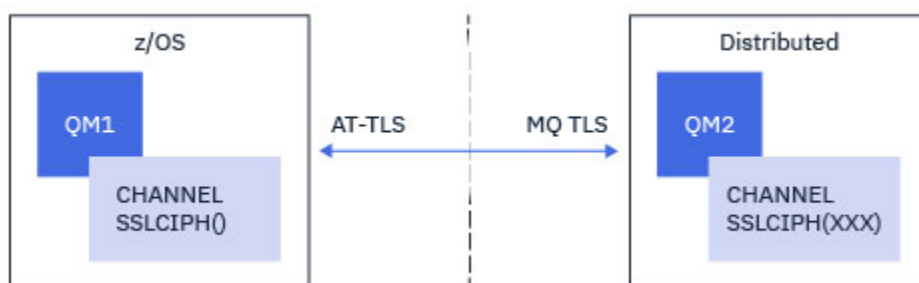
Una volta avviato, il canale utilizzerà una combinazione di AT - TLS e IBM MQ TLS.



**Attenzione:** Le precedenti istruzioni AT - TLS sono solo una configurazione minima. Ci sono altre istruzioni della politica AT - TLS con AT - TLS che non sono documentate qui e potrebbero essere utilizzate con IBM MQ a seconda della necessità. Tuttavia, IBM MQ è stato testato solo con le politiche descritte.

## Configurazione di AT - TLS su un canale in entrata da un gestore code IBM MQ for Multiplatforms utilizzando un singolo, denominato CipherSpec

Come impostare AT - TLS su un canale in ingresso da un gestore code IBM MQ for Multiplatforms a un gestore code IBM MQ for z/OS . In questo caso, il canale sul gestore code z/OS è un canale ricevente che non ha l'attributo SSLCIPH impostato e il canale sul gestore code non -z/OS è un canale mittente con l'attributo SSLCIPH impostato su un singolo, denominato CipherSpec.



In questo esempio, una coppia di canali mittente - ricevitore esistente, che utilizza TLS 1.2 TLS\_RSA\_WITH\_AES\_256\_GCM\_SHA384 CipherSpec verrà regolata in modo che il canale destinatario utilizzi AT - TLS invece di IBM MQ TLS.

Altri protocolli TLS e CipherSpecs possono essere utilizzati apportando modifiche minori alla configurazione. È possibile utilizzare altri tipi di canali di messaggi, ad eccezione dei canali mittente del cluster e ricevente del cluster, senza alcuna modifica alla configurazione AT - TLS.

## Procedura

### Passo 1: arresta il canale

### Passo 2: crea e applica una politica AT - TLS

È necessario creare le seguenti istruzioni AT - TLS per questo scenario:

1. Un'istruzione [TTLSRule](#) per far corrispondere le connessioni in entrata allo spazio di indirizzi dell'iniziatore di canali dall'indirizzo IP del canale mittente. In questo caso, è stato incluso un ulteriore filtro per corrispondere a un nome lavoro dell'iniziatore di canali specifico.

```
TTLSRule REMOTE-T0-CSQ1
{
  LocalAddr ALL
  LocalPortRange 1414
  RemoteAddr 123.456.78.9
  Jobname CSQ1CHIN
  Direction INBOUND
  TTLSGroupActionRef CSQ1-GROUP-ACTION
  TTLSEnvironmentActionRef CSQ1-INBOUND-ENVIRONMENT-ACTION
}
```

La regola precedente corrisponde alle connessioni provenienti dal lavoro CSQ1CHIN sulla porta locale 1414 dall'indirizzo IP remoto 123.456.78.9.

Opzioni di filtraggio più avanzate sono descritte in [TTLSRule](#).

2. Un'istruzione [TTLSGroupAction](#) che abilita la regola. [TTLSRule](#) fa riferimento al [TTLSGroupAction](#) utilizzando la proprietà **TTLSGroupActionRef**.

```
TTLSGroupAction CSQ1-GROUP-ACTION
{
  TTLSEnabled ON
}
```

3. Un'istruzione [TTLSEnvironmentAction](#) è associata a [TTLSRule](#) dalla proprietà **TTLSEnvironmentActionRef**. Un [TTLSEnvironmentAction](#) configura l'ambiente TLS e specifica quale keyring utilizzare.



```

TTLSEnvironmentAction          CSQ1-INBOUND-ENVIRONMENT-ACTION
{
  HandshakeRole                SERVER
  TLSKeyringParmsRef          CSQ1-KEYRING
  TTLSCipherParmsRef          CSQ1-CIPHERPARM
  TTLSEnvironmentAdvancedParmsRef CSQ1-ENVIRONMENT-ADVANCED
}

```

AT - TLS fornisce la possibilità di fornire l'autenticazione reciproca, che è l'equivalente dell'utilizzo dell'attributo del canale SSLCAUTH. Questa operazione viene eseguita con un'istruzione `TTLSEnvironmentAction` con un valore **HandshakeRole** di `ServerWithClientAuth` per l'istruzione `TTLSEnvironmentAction` in entrata.

4. Un'istruzione `TLSKeyringParms` è associata a `TTLSEnvironmentAction` dalla proprietà **TLSKeyringParmsRef** e definisce il keyring utilizzato da AT - TLS.

Il key ring deve contenere i certificati ritenuti attendibili dal gestore code nonz/OS remoto. Questo keyring può essere definito nello stesso modo di un keyring utilizzato dall'iniziatore del canale; consultare [“Configurazione del sistema z/OS per utilizzare TLS”](#) a pagina 248.

```

TLSKeyringParms              CSQ1-KEYRING
{
  Keyring                    MQCHIN/CSQ1RING
}

```

5. Un'istruzione `TTLSCipherParms` associata a `TTLSEnvironmentAction` dalla proprietà **TTLSCipherParmsRef**.

Questa istruzione deve contenere un singolo nome suite di cifratura che deve essere l'equivalente del nome IBM MQ CipherSpec utilizzato sul canale mittente remoto.

**Nota:** I nomi della suite di cifratura AT - TLS non corrispondono necessariamente ai nomi IBM MQ CipherSpec. Tuttavia, è possibile trovare il nome della suite di cifratura AT - TLS che corrisponde a un nome IBM MQ CipherSpec ricercando il nome IBM MQ CipherSpec dalla seguente tabella e facendo riferimento incrociato alla colonna di codice di quattro caratteri con la colonna di caratteri espansi dalla Tabella 2 nell'argomento [TTLSCipherParms](#).

*Tabella 72. Converti da codici di quattro caratteri a nomi CipherSpec*

Codice di quattro caratteri	Protocollo	Abilitato per impostazione predefinita.	Nome CipherSpec
0001	SSL 3.0	No	NULL_MD5
0002	SSL 3.0	No	NULL_SHA
0003	SSL 3.0	No	RC4_MD5_EXPORT
0004	SSL 3.0	No	RC4_MD5_US
0005	SSL 3.0	No	RC4_SHA_US
0006	SSL 3.0	No	RC2_MD5_EXPORT
0008	SSL 3.0	No	DES_SHA_EXPORT
0009	TLS 1.0	Sì	TLS_RSA_WITH_DES_CBC_SHA
000A	SSL 3.0	No	TRIPLE_DES_SHA_US
000A	TLS 1.0	Sì	TLS_RSA_WITH_3DES_EDE_CBC_SHA
002F	TLS 1.0	Sì	TLS_RSA_WITH_AES_128_CBC_SHA
0035	TLS 1.0	Sì	TLS_RSA_WITH_AES_256_CBC_SHA

Tabella 72. Converti da codici di quattro caratteri a nomi CipherSpec (Continua)

Codice di quattro caratteri	Protocollo	Abilitato per impostazione predefinita.	Nome CipherSpec
003B	TLS 1.2	Sì	TLS_RSA_WITH_NULL_SHA256
003C	TLS 1.2	Sì	TLS_RSA_WITH_AES_128_CBC_SHA256
003D	TLS 1.2	Sì	TLS_RSA_WITH_AES_256_CBC_SHA256
C023	TLS 1.2	Sì	ECDHE_ECDSA_AES_128_CBC_SHA256
C024	TLS 1.2	Sì	ECDHE_ECDSA_AES_256_CBC_SHA384
C027	TLS 1.2	Sì	ECDHE_RSA_AES_128_CBC_SHA256
C028	TLS 1.2	Sì	ECDHE_RSA_AES_256_CBC_SHA384

```
TTLSCipherParms CSQ1-CIPHERPARM
{
  V3CipherSuites TLS_RSA_WITH_AES_256_GCM_SHA384
}
```

6. Un'istruzione `TTLSEnvironmentAdvancedParms` è associata alla `TTLSEnvironmentAction` dalla proprietà **`TTLSEnvironmentAdvancedParmsRef`**.

Questa istruzione può essere utilizzata per specificare quali protocolli SSL e TLS sono abilitati. Con IBM MQ, è necessario abilitare solo il singolo protocollo che corrisponde al nome della suite di cifratura utilizzato nell'istruzione `TTLSCipherParms`.

```
TTLSEnvironmentAdvancedParms CSQ1-ENVIRONMENT-ADVANCED
{
  SSLv3 OFF
  TLSv1 OFF
  TLSv1.1 OFF
  SecondaryMap OFF
  TLSv1.2 ON
  TLSv1.3 OFF
}
```

La serie completa di istruzioni è la seguente e deve essere applicata all'agent della politica:

```

TTLSSRule                                REMOTE-T0-CSQ1
{
  LocalAddr                               ALL
  LocalPortRange                          1414
  RemoteAddr                               123.456.78.9
  Jobname                                  CSQ1CHIN
  Direction                                INBOUND
  TLSGroupActionRef                       CSQ1-GROUP-ACTION
  TTLEnvironmentActionRef                 CSQ1-INBOUND-ENVIRONMENT-ACTION
}

TLSGroupAction                            CSQ1-GROUP-ACTION
{
  TTLEnabled                              ON
}

TTLEnvironmentAction                      CSQ1-INBOUND-ENVIRONMENT-ACTION
{
  HandshakeRole                           CLIENT
  TLSKeyringParmsRef                     CSQ1-KEYRING
  TLSCipherParmsRef                     CSQ1-CIPHERPARM
  TTLEnvironmentAdvancedParmsRef        CSQ1-ENVIRONMENT-ADVANCED
}

TLSKeyringParms                          CSQ1-KEYRING
{
  Keyring                                  MQCHIN/CSQ1RING
}

TLSCipherParms                           CSQ1-CIPHERPARM
{
  V3CipherSuites                         TLS_RSA_WITH_AES_256_GCM_SHA384
}

TTLEnvironmentAdvancedParms              CSQ1-ENVIRONMENT-ADVANCED
{
  SSLv3                                   OFF
  TLSv1                                   OFF
  TLSv1.1                                 OFF
  SecondaryMap                            OFF
  TLSv1.2                                 OFF
  TLSv1.3                                 ON
}

```

### Passo 3: rimuovi SSLCIPH dal canale di z/OS

Rimuovere la CipherSpec dal canale z/OS utilizzando il seguente comando:

```
ALTER CHANNEL(channel-name) CHLTYPE(RCVR) SSLCIPH(' ')
```

### Passo 4: avvia canale

Una volta avviato, il canale utilizzerà una combinazione di AT - TLS e IBM MQ TLS.



**Attenzione:** Le precedenti istruzioni AT - TLS sono solo una configurazione minima. Ci sono altre istruzioni della politica AT - TLS con AT - TLS che non sono documentate qui e potrebbero essere utilizzate con IBM MQ a seconda della necessità. Tuttavia, IBM MQ è stato testato solo con le politiche descritte.

## Reimpostazione delle chiavi segrete SSL e TLS

IBM MQ supporta la reimpostazione delle chiavi segrete su gestori code e client.

Le chiavi segrete vengono reimpostate quando un numero specificato di byte di dati codificati è stato trasmesso attraverso il canale. Se gli heartbeat del canale sono abilitati, la chiave segreta viene reimpostata prima che i dati vengano inviati o ricevuti dopo un heartbeat del canale.

Il valore di reimpostazione della chiave è sempre impostato dal lato di inizializzazione del canale IBM 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.

**IBM i**

Su IBM i, utilizzare **CHGMQM** con il parametro **SSLRSTCNT** .

## 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 MQCONN
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 codificati inviati e ricevuti all'interno di una conversazione TLS prima che la chiave segreta TLS venga rinegoziata. Specificare un valore 0 indica che le chiavi segrete TLS non vengono mai rinegoziate. Se si specifica un conteggio di reimpostazione della chiave segreta TLS compreso tra 1 byte e 32 KB, i canali 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 i valori di reimpostazione della chiave segreta 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 TLS nel file di configurazione client, consultare [Stanza SSL del file di configurazione client](#).

## Java

Per IBM 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 sslResetCount la proprietà dell'ambiente MQC.SSL\_RESET\_COUNT\_PROPERTY rappresenta il numero totale di byte inviati e ricevuti dal codice client IBM MQ classes for 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 IBM MQ classes for Java .

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 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 MQ classes for JMS. Ad esempio, per configurare un oggetto ConnectionFactory che può essere utilizzato per creare una connessione su un canale MQI abilitato TLS con una chiave segreta 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 .NET client non gestiti, la proprietà integer SSLKeyResetCount indica il numero di byte non codificati inviati e ricevuti all'interno di una conversazione TLS prima che la chiave segreta venga rinegoziata.

Per informazioni sull'utilizzo delle proprietà oggetto in IBM MQ classes for .NET, consultare [Acquisizione e impostazione dei valori degli attributi](#).

Per i client gestiti .NET, la classe SSLStream non supporta la reimpostazione / rinegoziazione della chiave segreta. Tuttavia, per essere congruente con gli altri client IBM MQ, il client IBM MQ gestito .NET consente alle applicazioni di impostare il conteggio SSLKeyReset. Per ulteriori informazioni, vedi [Reimpostazione o rinegoziazione della chiave segreta](#).

## **XMS .NET**

Per i client non gestiti XMS .NET, consultare [Connessioni sicure a un gestore code IBM MQ](#).

### **Informazioni correlate**

[Gestore code ALTER](#)

[VISUALIZZAZIONE QMGR](#)

[Modifica gestore code messaggi \(CHGMQM\)](#)

[Visualizza gestore code messaggi \(DSPMQM\)](#)

## **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 la chiave simmetrica può essere generata e distribuita, consultare [“Implementazione della riservatezza nei programmi di uscita utente”](#) a pagina 429.

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 TLS, la scelta di CipherSpec determina il livello di integrità dei dati nell'enterprise. Se si utilizza l'AMS (Advanced Message Service) di IBM MQ, 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,

## Ulteriori informazioni

Consultare la sezione su [“Abilitazione di CipherSpecs” a pagina 408](#) per ulteriori informazioni su come garantire l'integrità dei dati.

### Informazioni correlate

[Connessione di due gestori code mediante TLS](#)

[Connessione sicura di un client a un gestore code](#)

## 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 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 durante la configurazione di TLS per 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/OS o OAM su altre piattaforme.

#### Informazioni su questa attività

Utilizzare le funzionalità di sicurezza di una piattaforma e il meccanismo di controllo accessi in IBM 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 della sicurezza esterni su IBM 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, il controllo potrebbe comportare 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. Su z/OS la verifica potrebbe implicare l'utilizzo dell'ID utente ricevuto dalla rete o di quello derivato da MCAUSER o entrambi. Utilizzare questa opzione se il link è attendibile e autenticato.

## ONLYMCA (solo z/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 (solo z/OS)

Come per CTX, ma qualsiasi ID utente ricevuto dalla rete non viene utilizzato.

## Autorizzazione all'inserimento di messaggi nelle code del cluster remoto

Su z/OS impostare l'autorizzazione per l'inserimento in una coda cluster utilizzando RACF. Su altre piattaforme, autorizzare l'accesso per connettersi ai gestori code e per inserire le code su tali gestori 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 z/OS, immettere i seguenti comandi:

```
RDEFINE MQQUEUE QMgrName.QUEUE. QueueName UACC(NONE)
PERMIT QMgrName.QUEUE. QueueName CLASS(MQADMIN) ID(GroupName) ACCESS(UPDATE)
```

- Per sistemi UNIX, Linux, and Windows, immettere i seguenti comandi:

```
setmqaut -m QMgrName -t qmgr -g GroupName +connect
setmqaut -m QMgrName -t queue -n QueueName -g GroupName -all +put
```

- Per IBM i, immettere i seguenti comandi:

```
GRTMQMAUT OBJ(' QMgrName ') OBJTYPE(*MQM) USER(GroupName) AUT(*CONNECT)
GRTMQMAUT OBJ(' QueueName ') OBJTYPE(*Q) USER(GroupName) AUT(*PUT) MQMNAME(' QMgrName ')
```

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. 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.

#### 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 386.

### 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 del 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) 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.

### z/OS

Il modulo di caricamento dell'uscita di sicurezza deve trovarsi nel dataset specificato nell'istruzione CSQXLIB DD della procedura dello spazio di indirizzo dell'iniziatore di canali.

### Sistemi Windows, 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.

**Nota:** Sebbene l'utilizzo del comando RESET CLUSTER rimuova forzatamente un gestore code da un cluster, l'utilizzo di RESET CLUSTER da solo non impedisce al gestore code di ricongiungersi al cluster in un secondo momento. Per assicurarsi che il gestore code non si riunisca al cluster, attenersi alla procedura descritta in [“Impedire ai gestori code di unirsi a un cluster”](#) a pagina 434.

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 definizioni del 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/TLS e cluster

Quando si configura il TLS per i cluster, tenere presente che una definizione di canale CLUSRCVR viene propagata ad altri gestori code come un canale CLUSSDR definito automaticamente. Se un canale CLUSRCVR utilizza TLS, è necessario configurare TLS su tutti i gestori code che comunicano utilizzando il canale.

Per ulteriori informazioni su TLS, consultare [“Protocolli di sicurezza TLS in IBM MQ”](#) a pagina 24. Il consiglio è generalmente applicabile ai canali cluster, ma è possibile considerare in modo particolare quanto segue:

In un cluster IBM MQ una particolare definizione di canale CLUSRCVR viene spesso propagata a molti altri gestori code in cui viene trasformata in un CLUSSDR definito automaticamente. Successivamente, il CLUSSDR definito automaticamente viene utilizzato per avviare un canale per CLUSRCVR. Se CLUSRCVR è configurato per la connessione TLS, si applicano le seguenti considerazioni:

- Tutti i gestori code che desiderano comunicare con questo CLUSRCVR devono avere accesso al supporto TLS. Questo provisioning TLS 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 conatteranno 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 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=IBM 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 del mittente del cluster definiti automaticamente risultanti devono risiedere su piattaforme su cui il supporto TLS sottostante implementa questo CipherSpec e su cui può essere specificato 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 TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA).

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/TLS

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 IBM MQ corrente e non 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.



**Attenzione:** Non è possibile utilizzare una combinazione di certificati firmati Elliptic Curve e certificati firmati RSA sui gestori code che si desidera unire insieme come parte di un cluster.

I gestori code in un cluster devono utilizzare tutti i certificati firmati RSA o tutti i certificati firmati EC, non una combinazione di entrambi.

Per ulteriori informazioni, fare riferimento a [“Certificati digitali e compatibilità CipherSpec in IBM MQ”](#) a pagina 42.

Aggiornare tutti i gestori code nel cluster a IBM MQ V7.5 o versioni successive, se non sono già a questi livelli. Distribuire i certificati e le chiavi in modo che TLS funzioni da ciascuno di essi.

### Informazioni su questa attività

Modificare i canali CLUSRCVR prima dei canali CLUSSDR .

### Procedura

1. Passare i canali CLUSRCVR a TLS in qualsiasi ordine, modificando un CLUSRCVR 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.

2. Opzionale: Passare tutti i canali CLUSSDR manuali a TLS.

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'uso del comando **REFRESH CLUSTER** può essere distruttivo per il cluster mentre è in corso e di nuovo a intervalli di 27 giorni quando gli oggetti cluster inviano automaticamente gli aggiornamenti dello stato a tutti i gestori code interessati. Consultare [Refreshing in a large cluster can affect performance and availability of the cluster](#).

3. Utilizzare il comando `DISPLAY CLUSQMgr` per assicurarsi che la nuova configurazione di sicurezza sia stata propagata in tutto il cluster.
4. Riavviare i canali per utilizzare TLS ed eseguire [REFRESH SECURITY \(SSL\)](#).

## Concetti correlati

“Abilitazione di CipherSpecs” a pagina 408

Abilitare una CipherSpec utilizzando il parametro **SSLCIPH** nel comando MQSC **DEFINE CHANNEL** o nel comando MQSC **ALTER CHANNEL** .

“Certificati digitali e compatibilità CipherSpec in IBM MQ” a pagina 42

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 MQ.

## Informazioni correlate

[Cluster: utilizzo delle procedure consigliate per REFRESH CLUSTER](#)

## Disabilitazione di SSL/TLS su canali e gestori code con cluster

Per disattivare 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 tra virgolette singole

 o \*NONE su IBM i .

È possibile disattivare TLS sui canali riceventi del cluster in qualsiasi ordine desiderato.

Tieni presente che le modifiche fluiscono nella direzione opposta sui canali su cui lasci attivo TLS.

2. Verificare che il nuovo valore si rifletta in tutti i gestori code utilizzando il comando **DISPLAY CLUSQMGR(\*) ALL**.

3. Disattivare TLS su tutti i canali mittenti 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 distruttivo 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 22 a pagina 439](#) e descritte nel seguente esempio.

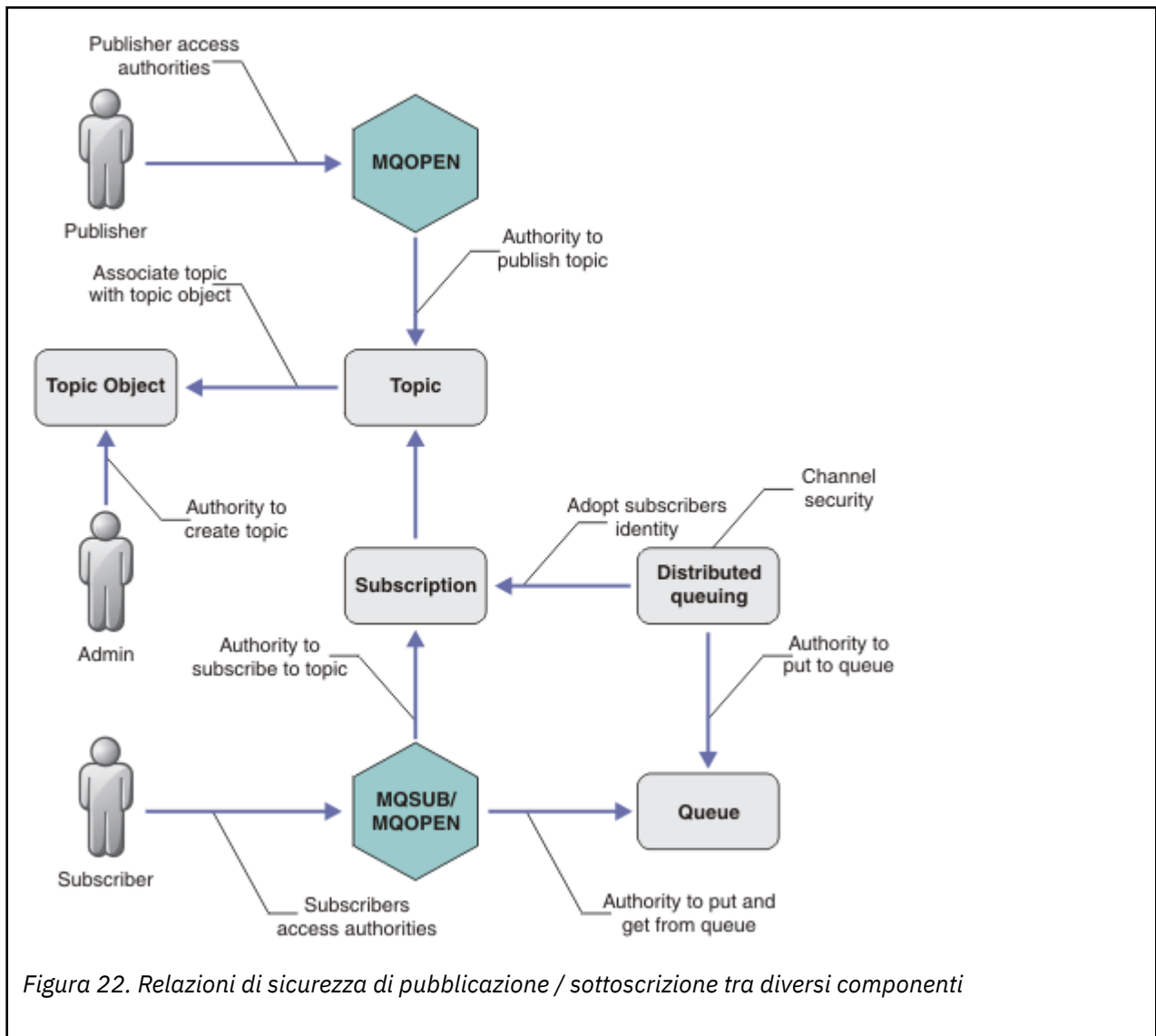
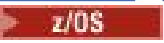


Figura 22. Relazioni di sicurezza di pubblicazione / sottoscrizione tra diversi componenti

### 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 441 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 446 e [“Concedi l'accesso a un utente per la pubblicazione in un argomento”](#) a pagina 454.  Per il controllo dell'accesso agli oggetti argomento su z/OS, vedere [Profili per la sicurezza degli argomenti](#).

### 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 442

### **Sottoscrivi utilizzando un argomento**

[“Sottoscrizione utilizzando una stringa di argomenti in cui il nodo di argomenti non esiste” a pagina 443](#)

### **Sottoscrivi utilizzando un argomento con caratteri jolly**

[“Sottoscrizione utilizzando una stringa di argomenti che contiene caratteri jolly” a pagina 444](#)

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 459](#).

### **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 444](#).

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 i gestori code” a pagina 461**

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 IBM 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.

[“Sicurezza di pubblicazione / sottoscrizione tra i gestori code” a pagina 461](#) descrive come modificare le proprie definizioni di canale 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 [Specifiche 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. IBM MQ Amministratore - 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.

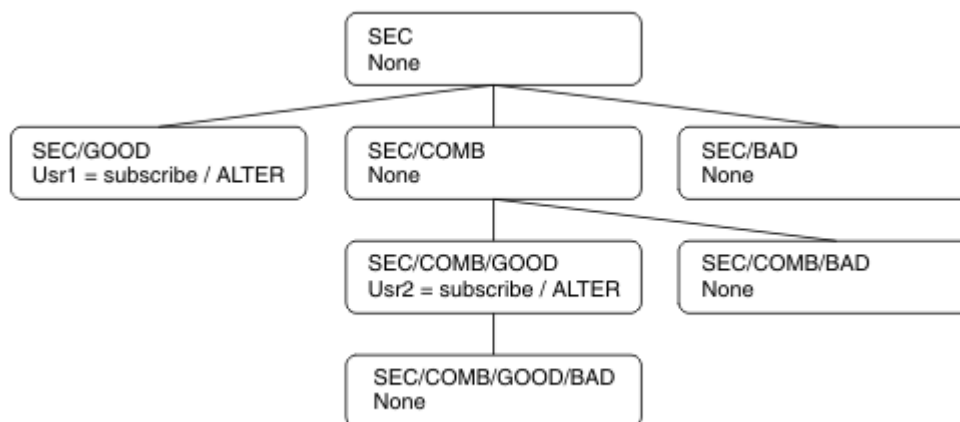
Considerare il seguente esempio in cui gli oggetti argomento sono stati definiti con gli attributi di sicurezza o le autorizzazioni mostrate:

Nome argomento	Stringa argomento	Autorizzazioni - non z/OS	z/OS Autorizzazioni
SECROOT	SEC	Nessuno	Nessuno

Tabella 73. Autorizzazioni oggetto argomento di esempio (Continua)

Nome argomento	Stringa argomento	Autorizzazioni - non z/OS	z/OS Autorizzazioni
SECGOOD	SEC/GOOD	usr1+subscribe	ALTER HLQ.SUBSCRIBE.SECGOOD
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`.

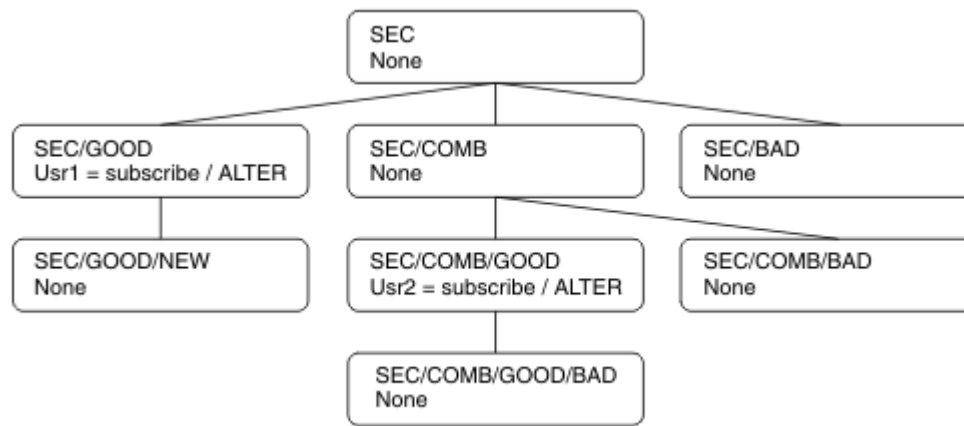
## **Sottoscrizione utilizzando una stringa di argomenti in cui esiste il nodo di argomenti**

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 si risolve 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 viene specificato il parametro **SUBUSER** , ciò influisce sul modo in cui viene eseguito il controllo, come mostrato in [Tabella 74 a pagina 446](#):

<i>Tabella 74. ID utente utilizzati per i controlli di sicurezza per i comandi</i>			
<b>Comando</b>	<b>SUBUSER specificato e vuoto</b>	<b>SUBUSER specificato e completato</b>	<b>SUBUSER non specificato</b>
	Utilizza l'ID amministratore		Utilizzare l'ID utente dalla sottoscrizione e LIKE
	Utilizza l'ID amministratore		Utilizzare.DEFAULT.SU l'ID utenteB - se vuoto, dalutilizzare SISTEMA l'ID SYSTEM amministratore
	Utilizza l'ID amministratore		Utilizza l'ID utente dalla sottoscrizione e 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 ha il controllo accessi impostato 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 attività ... 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.

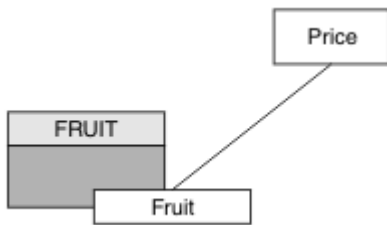


Figura 23. Esempio di accesso all'oggetto argomento

Tabella 75. 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:

- **z/OS** z/OS :

Concedere l'accesso a USER1 per sottoscrivere l'argomento "Price/Fruit" concedendo l'accesso utente al profilo hlq.SUBSCRIBE.FRUIT. Effettuare questa operazione, utilizzando i seguenti comandi RACF :

```
RDEFINE MXTOPIC hlq.SUBSCRIBE.FRUIT UACC(NONE)
PERMIT hlq.SUBSCRIBE.FRUIT CLASS(MXTOPIC) ID(USER1) ACCESS(ALTER)
```

- 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:

**ULW** Sistemi Windows, UNIX and Linux

```
setmqaut -t topic -n FRUIT -p USER1 +sub
```

**IBM i** IBM i

```
GRTRMQAUT OBJ(FRUIT) OBJTYPE(*TOPIC) USER(USER1) AUT(*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:

- **z/OS** 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 ...

```

- **ULW** Su altre piattaforme, il seguente evento di autorizzazione:

```

MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRC_SUB_NOT_AUTHORIZED
UserIdentifier       USER2
AdminTopicNames     FRUIT, SYSTEM.BASE.TOPIC
TopicString          "Price/Fruit"

```

- **IBM i** Su IBMi, il seguente evento di autorizzazione:

```

MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRC_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 446.

### 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.



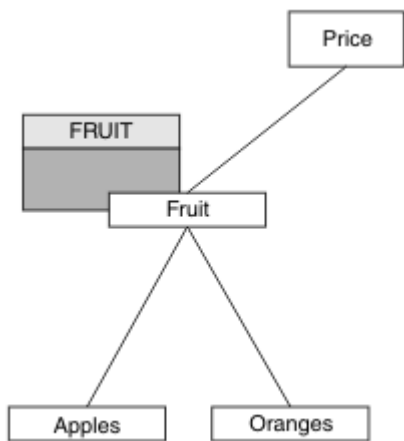


Figura 24. Esempio di concessione dell'accesso a un argomento all'interno di una struttura ad albero degli argomenti

Tabella 76. 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, a USER1 è stato concesso l'accesso per sottoscrivere l'argomento "Price/Fruit" concedendo l'accesso al profilo hlq.SUBSCRIBE.FRUIT su z/OS e l'accesso 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 MQRQ\_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:

```

MQRQ_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 448.

### 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.

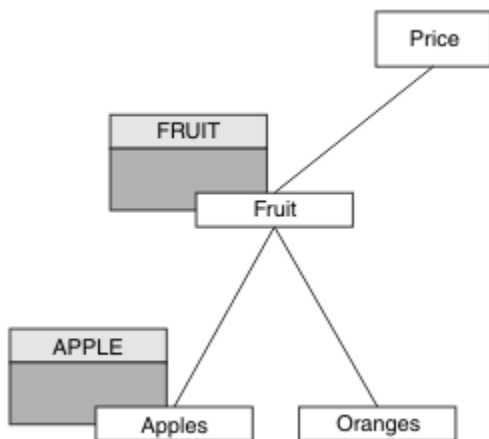


Figura 25. Concessione dell'accesso a specifici argomenti all'interno di una struttura ad albero degli argomenti

Tabella 77. 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
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:

• **z/OS** **z/OS :**

Nell'attività precedente USER1 è stato concesso l'accesso per la sottoscrizione all'argomento "Price/Fruit/Apples" concedendo all'utente l'accesso al profilo h1q.SUBSCRIBE.FRUIT .

Questo singolo profilo ha anche concesso l'accesso USER1 per sottoscrivere "Price/Fruit/Oranges" "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 l'accesso utente al profilo h1q.SUBSCRIBE.APPLE . Effettuare questa operazione, utilizzando i seguenti comandi RACF :

```
RDEFINE MXTOPIC h1q.SUBSCRIBE.APPLE UACC(NONE)
PERMIT h1q.SUBSCRIBE.FRUIT APPLE(MXTOPIC) ID(USER2) ACCESS(ALTER)
```

• 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:

• **ULW** **Sistemi Windows, UNIX and Linux**

```
setmqaut -t topic -n APPLE -p USER2 +sub
```

• **IBM i** **IBM i**

```
GRTMQAUT OBJ(APPLE) OBJTYPE(*TOPIC) USER(USER2) AUT(*SUB)
```

## Risultati

Su z/OS, quando USER1 tenta di sottoscrivere l'argomento "Price/Fruit/Apples" il primo controllo di sicurezza sul profilo h1q.SUBSCRIBE.APPLE ha esito negativo, ma spostando la struttura ad albero il profilo h1q.SUBSCRIBE.FRUIT consente a USER1 di sottoscrivere, quindi la sottoscrizione ha esito positivo e nessun codice di ritorno viene inviato alla chiamata MQSUB. Tuttavia, viene generato un messaggio RACF ICH per la prima verifica:

```
ICH408I USER(USER1 ) ...
h1q.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 MQRN\_NOT\_AUTHORIZED , insieme a:

- **z/OS** 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 ...
```

- **U/LW** Sulle piattaforme Windows, UNIX and 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"
```

- **IBM i** Su IBMi, 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 per sottoscrivere gli 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 450](#) 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.

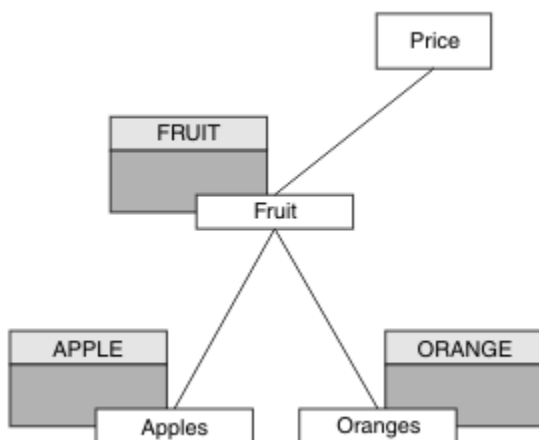


Figura 26. Esempio di concessione del controllo accessi per evitare ulteriori messaggi.

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:

- **z/OS** **z/OS** :

Definire un nuovo profilo e aggiungere l'accesso a tale profilo e ai profili esistenti. Effettuare questa operazione, utilizzando i seguenti comandi RACF :

```
RDEFINE MXTOPIC h1q.SUBSCRIBE.ORANGE UACC(NONE)
PERMIT h1q.SUBSCRIBE.ORANGE CLASS(MXTOPIC) ID(USER1) ACCESS(ALTER)
PERMIT h1q.SUBSCRIBE.APPLE CLASS(MXTOPIC) ID(USER1) ACCESS(ALTER)
```

- Altre piattaforme:

Impostare l'accesso equivalente utilizzando i comandi di autorizzazione per la piattaforma:

- **ULW** **Sistemi Windows, UNIX and Linux**

```
setmqaut -t topic -n ORANGE -p USER1 +sub
setmqaut -t topic -n APPLE -p USER1 +sub
```

- **IBM i** **IBM i**

```
GRTMQAUT OBJ(ORANGE) OBJTYPE(*TOPIC) USER(USER1) AUT(*SUB)
GRTMQAUT OBJ(APPLE) OBJTYPE(*TOPIC) USER(USER1) AUT(*SUB)
```

## Risultati

Su z/OS, quando USER1 tenta di sottoscrivere un argomento "Price/Fruit/Apples" , il primo controllo di sicurezza sul profilo h1q.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:

- **z/OS** 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 ) ...
h1q.SUBSCRIBE.ORANGE ...

ICH408I USER(USER2 ) ...
h1q.SUBSCRIBE.FRUIT ...

ICH408I USER(USER2 ) ...
h1q.SUBSCRIBE.SYSTEM.BASE.TOPIC ...
```

- **ULW** Su altre piattaforme, il seguente evento di autorizzazione:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRC_SUB_NOT_AUTHORIZED
UserIdentifier       USER2
AdminTopicNames     ORANGE, FRUIT, SYSTEM.BASE.TOPIC
TopicString          "Price/Fruit/Oranges"
```

- **IBM i** Su IBMi, il seguente evento di autorizzazione:

```

MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRC_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 attività 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:

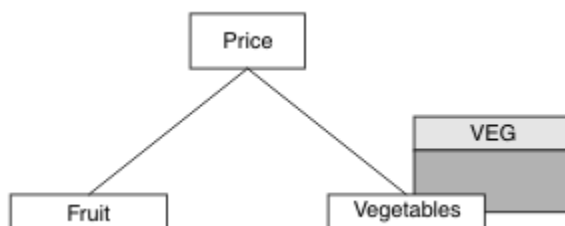


Figura 27. Concessione dell'accesso di pubblicazione a un argomento

Tabella 78. 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:

- **z/OS** **z/OS** :

Concedere l'accesso a USER1 per pubblicare nell'argomento "Price/Vegetables" concedendo all'utente l'accesso al profilo `hlq.PUBLISH.VEG`. Effettuare questa operazione, utilizzando i seguenti comandi RACF :

```

RDEFINE MXTOPIC hlq.PUBLISH.VEG UACC(NONE)
PERMIT hlq.PUBLISH.VEG CLASS(MXTOPIC) ID(USER1) ACCESS(UPDATE)

```

- 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:

#### Sistemi Windows, UNIX and Linux

```
setmqaut -t topic -n VEG -p USER1 +pub
```


#### IBM i

```
GRTRMQUAUT OBJ(VEG) OBJTYPE(*TOPIC) USER(USER1) AUT(*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:

-  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"
```

-  Su IBMi, 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 454.

## 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.

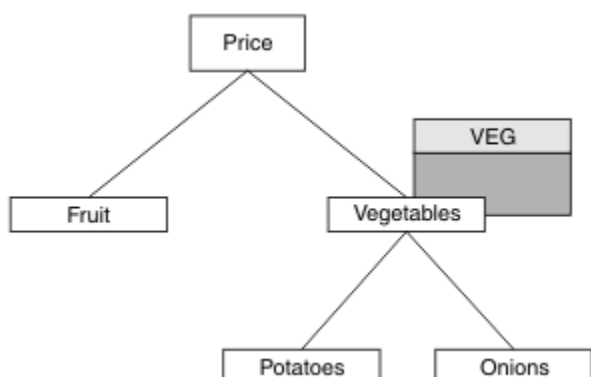


Figura 28. Concessione dell'accesso di pubblicazione a un argomento all'interno di una struttura ad albero degli argomenti

Tabella 79. Requisiti di accesso alla pubblicazione di esempio

Argomento	Accesso di sottoscrizione richiesto	Oggetto sezione
Prezzo	Nessun utente	Nessuno
Prezzo / Verdure	USER1	VEG
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" concedendogli 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      MQRC_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 455.

### 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.

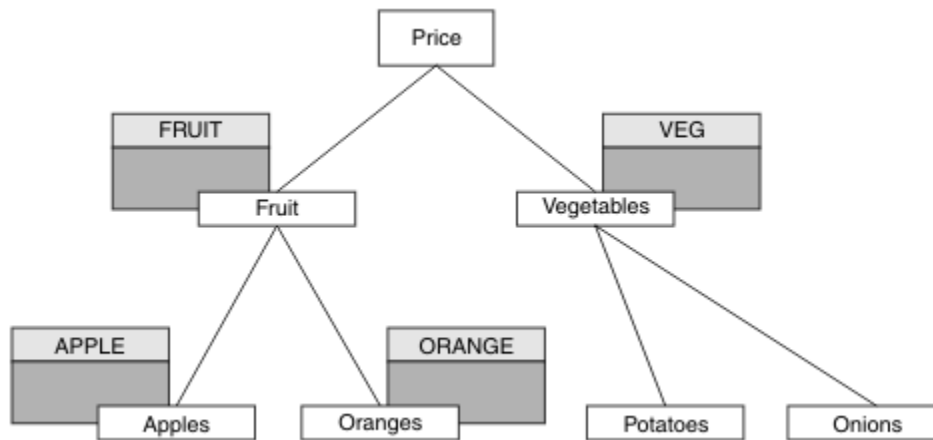


Figura 29. Concessione dell'accesso per la pubblicazione e la sottoscrizione

Tabella 80. 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

Tabella 80. Esempio di pubblicazione e sottoscrizione dei requisiti di accesso (Continua)

Argomento	Accesso di sottoscrizione richiesto	È richiesto l'accesso alla pubblicazione	Oggetto sezione
Prezzo / Frutta / Arance	USER1		ARANCIO

## Procedura

Concedere l'accesso come segue:

- ▶ **z/OS** **z/OS** :

In un'attività precedente, a USER1 era stato concesso l'accesso per sottoscrivere l'argomento "Price/Fruit" concedendo all'utente l'accesso al profilo hlq.SUBSCRIBE.FRUIT.

Per pubblicare nell'argomento "Price/Fruit", concedere l'accesso a USER1 al profilo hlq.PUBLISH.FRUIT. Effettuare questa operazione, utilizzando i seguenti comandi RACF :

```
RDEFINE MXTOPIC hlq.PUBLISH.FRUIT UACC(NONE)
PERMIT hlq.PUBLISH.FRUIT CLASS(MXTOPIC) ID(USER1) ACCESS(ALTER)
```

- 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:

- ▶ **ULW** **Sistemi Windows, UNIX and Linux**

```
setmqaut -t topic -n FRUIT -p USER1 +pub
```

- ▶ **IBM i** **IBM i**

```
GRTMQAUT OBJ(FRUIT) OBJTYPE(*TOPIC) USER(USER1) AUT(*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:

- ▶ **z/OS** 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.FRUIT ...

ICH408I USER(USER2 ) ...
hlq.PUBLISH.SYSTEM.BASE.TOPIC ...
```

- **ULW** Su piattaforme Windows, UNIXe Linux , il seguente evento autorizzazione:

```

MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier       USER2
AdminTopicNames     FRUIT, SYSTEM.BASE.TOPIC
TopicString          "Price/Fruit"

```

- **IBM i** Su IBMi, 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 81. Elenco completo delle autorizzazioni di accesso risultanti da esempi di sicurezza*

<b>Argomento</b>	<b>Accesso di sottoscrizione richiesto</b>	<b>È richiesto l'accesso alla pubblicazione</b>	<b>Oggetto sezione</b>
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 di 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 AlternateUser contiene un identificativo utente da utilizzare per convalidare questa chiamata MQSUB. La chiamata può riuscire solo se questo ID AlternateUser è 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:

<i>Tabella 82. Informazioni sul contesto di pubblicazione predefinito</i>	
<b>Campo in MQMD</b>	<b>Valore utilizzato</b>
<i>UserIdentifier</i>	L'ID utente associato alla sottoscrizione (vedere il campo SUBUSER su DISPLAY SBSTATUS) al momento della pubblicazione.
<i>AccountingToken</i>	Determinato dall'ambiente, se possibile; altrimenti, impostare su MQACT_NONE.
<i>ApplIdentityData</i>	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 AlternateUser al servizio di autorizzazione per consentire l'esecuzione di controlli di autorizzazione appropriati. L'ID AlternateSecurity viene utilizzato solo se viene specificato MQSO\_ALTERNATE\_USER\_AUTHORITY e il campo ID AlternateUser non 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'identità ... del sottoscrittore non è 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.

## **Sicurezza di pubblicazione / sottoscrizione tra i gestori code**

I messaggi interni di pubblicazione / sottoscrizione, come le sottoscrizioni proxy e le pubblicazioni, vengono inseriti nelle code di sistema di pubblicazione / sottoscrizione utilizzando le regole di sicurezza del canale normali. Le informazioni e i diagrammi in questo argomento evidenziano i diversi processi e ID utente coinvolti nella consegna di questi messaggi.

### **Controllo accesso locale**

L'accesso agli argomenti per la pubblicazione e le sottoscrizioni è regolato da regole e definizioni di sicurezza locali descritte in [Sicurezza di pubblicazione / sottoscrizione](#). Su z/OS, non è richiesto alcun oggetto argomento locale per stabilire il controllo accessi. Non è richiesto alcun argomento locale per il controllo accessi su altre piattaforme. Gli amministratori possono scegliere di applicare il controllo accessi agli oggetti argomento in cluster, indipendentemente dal fatto che siano già presenti nel cluster.

Gli amministratori di sistema sono responsabili del controllo degli accessi sul proprio sistema locale. È necessario che gli amministratori di altri membri della gerarchia o dei collettivi del cluster siano responsabili della loro politica di controllo degli accessi. Poiché il controllo di accesso è definito per ogni macchina separata, è probabile che sia gravoso se è necessario un controllo di livello fine. Potrebbe non essere necessario imporre alcun controllo accessi oppure il controllo accessi potrebbe essere definito su oggetti di alto livello nella struttura ad albero degli argomenti. È possibile definire un controllo di accesso di livello fine per ogni suddivisione dello spazio dei nomi argomento.

### **Esecuzione di una sottoscrizione proxy**

L'attendibilità per un'organizzazione di connettere il proprio gestore code al proprio gestore code è confermata dai normali mezzi di autenticazione del canale. Se a tale organizzazione attendibile è consentito eseguire anche la pubblicazione / sottoscrizione distribuita, viene eseguito un controllo dell'autorizzazione. La verifica viene effettuata quando il canale inserisce un messaggio in una coda di pubblicazione / sottoscrizione distribuita. Ad esempio, se un messaggio viene inserito nella coda SYSTEM.INTER.QMGR.CONTROL. L'ID utente per il controllo autorizzazione coda dipende dai valori

PUTAUT del canale ricevente. Ad esempio, l'ID utente del canale, MCAUSER, il contesto del messaggio, a seconda del valore e della piattaforma. Per ulteriori informazioni sulla sicurezza del canale, vedi [Sicurezza del canale](#).

Le sottoscrizioni proxy vengono effettuate con l'ID utente dell'agente di pubblicazione / sottoscrizione distribuito sul gestore code remoto. Ad esempio, QM2 in [Figura 30](#) a pagina 462. All'utente viene quindi facilmente concesso l'accesso ai profili oggetto argomento locale, poiché tale ID utente è definito nel sistema e quindi non vi sono conflitti di dominio.

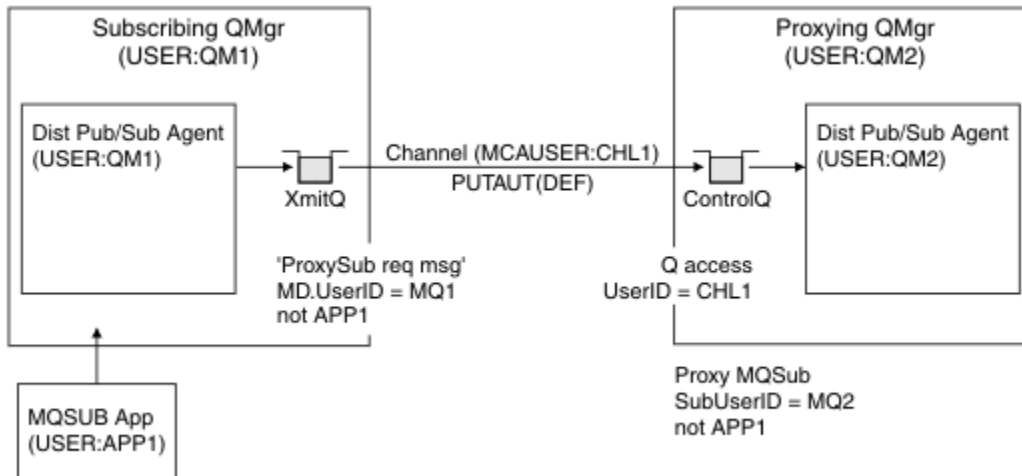


Figura 30. Sicurezza della sottoscrizione proxy, creazione di una sottoscrizione

## Invio di pubblicazioni remote

Quando una pubblicazione viene creata sul gestore code di pubblicazione, viene creata una copia della pubblicazione per qualsiasi sottoscrizione proxy. Il contesto della pubblicazione copiata contiene il contesto dell'ID utente che ha effettuato la sottoscrizione; QM2 in [Figura 31](#) a pagina 463. La sottoscrizione proxy viene creata con una coda di destinazione che è una coda remota, in modo che il messaggio di pubblicazione venga risolto in una coda di trasmissione.

L'attendibilità per un'organizzazione per la connessione del proprio gestore code, QM2, a un altro gestore code, QM1, è confermata dai normali mezzi di autenticazione di canale. Se a tale organizzazione attendibile è quindi consentito eseguire la pubblicazione / sottoscrizione distribuita, viene eseguito un controllo di autorizzazione quando il canale inserisce il messaggio di pubblicazione nella coda di pubblicazione / sottoscrizione distribuita SYSTEM.INTER.QMGR.PUBS. L'ID utente per il controllo autorizzazione coda dipende dal valore PUTAUT del canale ricevente (ad esempio, l'ID utente del canale, MCAUSER, il contesto del messaggio e altri, in base al valore e alla piattaforma). Per ulteriori informazioni sulla sicurezza del canale, vedi [Sicurezza del canale](#).

Quando il messaggio di pubblicazione raggiunge il gestore code di sottoscrizione, viene eseguito un altro MQPUT per l'argomento sotto l'autorità di tale gestore code e il contesto con il messaggio viene sostituito dal contesto di ciascuno dei sottoscrittori locali, poiché a ciascuno viene fornito il messaggio.

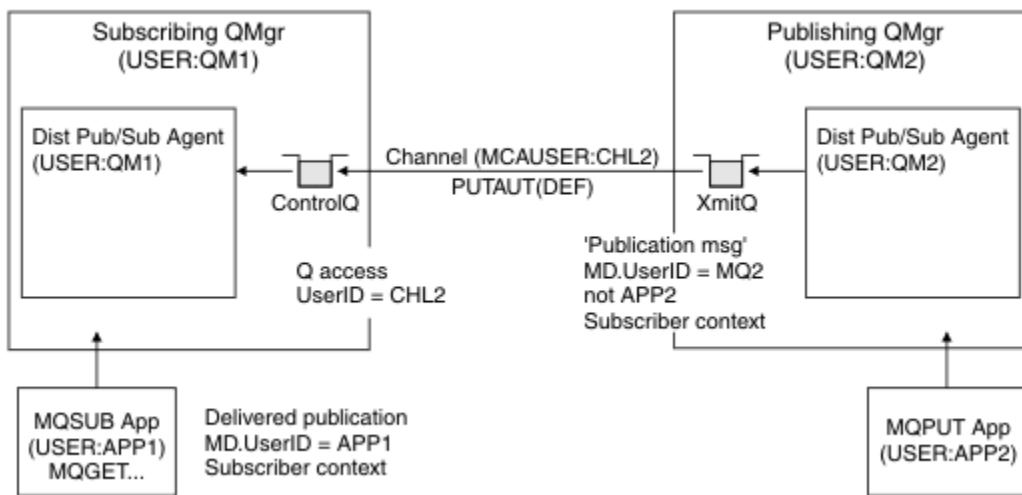


Figura 31. Sicurezza della sottoscrizione proxy, inoltro delle pubblicazioni

Su un sistema in cui è stato considerato poco per quanto riguarda la sicurezza, è probabile che i processi di pubblicazione / sottoscrizione distribuiti siano in esecuzione con un ID utente nel gruppo mqm , il parametro MCAUSER su un canale è vuoto (valore predefinito) e i messaggi vengono consegnati alle diverse code di sistema come richiesto. Il sistema non protetto rende semplice l'impostazione di una prova di concetto per dimostrare la pubblicazione / sottoscrizione distribuita.

Su un sistema in cui la sicurezza è considerata più seriamente, questi messaggi interni sono soggetti agli stessi controlli di sicurezza di qualsiasi messaggio che passa attraverso il canale.

Se il canale è impostato con un valore MCAUSER non vuoto e un valore PUTAUT che specifica che è necessario selezionare MCAUSER , è necessario concedere l'accesso alle code SYSTEM . INTER . QMGR . \* al MCAUSER in questione. Se ci sono più gestori code remoti differenti, con canali in esecuzione con ID MCAUSER differenti, a tutti questi ID utente deve essere concesso l'accesso alle code SYSTEM . INTER . QMGR . \* . I canali in esecuzione con ID MCAUSER differenti potrebbero verificarsi, ad esempio, quando più connessioni gerarchiche sono configurate su un singolo gestore code.

Se il canale è impostato con un valore PUTAUT che specifica che viene utilizzato il contesto del messaggio, l'accesso alle code SYSTEM . INTER . QMGR . \* viene controllato in base all'ID utente all'interno del messaggio interno. Poiché tutti questi messaggi vengono inseriti con l'ID utente dell'agent di pubblicazione / sottoscrizione distribuito dal gestore code che sta inviando il messaggio interno o il messaggio di pubblicazione (vedere Figura 31 a pagina 463 ), non è troppo grande una serie di ID utente per concedere l'accesso alle varie code di sistema (una per gestore code remoto), se si desidera configurare la sicurezza di pubblicazione / sottoscrizione distribuita in questo modo. Ha ancora tutti gli stessi problemi che la sicurezza del contesto del canale ha sempre; quello dei diversi domini ID utente e il fatto che l'ID utente nel messaggio potrebbe non essere definito nel sistema ricevente. Tuttavia, è un modo perfettamente accettabile di funzionare, se necessario.

**z/OS** Sicurezza della coda di sistema fornisce un elenco di code e l'accesso richiesto per configurare in modo sicuro l'ambiente di pubblicazione / sottoscrizione distribuito. Se i messaggi interni o le pubblicazioni non vengono inseriti a causa di violazioni della sicurezza, il canale scrive un messaggio nel log in modo normale e i messaggi possono essere inviati alla coda di messaggi non recapitabili in base alla normale elaborazione degli errori del canale.

Tutta la messaggistica tra gestori code per la pubblicazione / sottoscrizione distribuita viene eseguita utilizzando la normale sicurezza del canale.

Per informazioni sulla limitazione delle pubblicazioni e delle sottoscrizioni proxy a livello di argomento, consultare Sicurezza di pubblicazione / sottoscrizione.

## Utilizzo degli ID utente predefiniti con una gerarchia di gestori code

Se si dispone di una gerarchia di gestori code in esecuzione su piattaforme diverse e si utilizzano ID utente predefiniti, notare che questi ID utente predefiniti differiscono tra le piattaforme e potrebbero non essere noti sulla piattaforma di destinazione. Di conseguenza, un gestore code in esecuzione su una piattaforma rifiuta i messaggi ricevuti dai gestori code su altre piattaforme con il codice motivo MQRC\_NOT\_AUTHORIZED.

Per evitare che i messaggi vengano rifiutati, è necessario aggiungere le seguenti autorizzazioni agli ID utente predefiniti utilizzati su altre piattaforme:

- Autorizzazione \*PUT \*GET su SYSTEM.BROKER. Code
- \*PUB Autorizzazione \*SUB su SYSTEM.BROKER. argomenti
- Autorizzazione \*ADMCR \*ADMCLT \*ADMCHG sul SISTEMA SYSTEM.BROKER.CONTROL.QUEUE .

Gli ID utente predefiniti con una gerarchia di gestori code sono i seguenti:

Piattaforma	ID utente predefinito
Windows	MUSR_MQADMIN
Sistemi UNIX and Linux	mqm
IBM i	QMQM
z/OS	L'ID utente dello spazio di indirizzo dell'iniziatore di canali

Creare e concedere l'accesso all'ID utente 'qmqm' se gerarchicamente collegato a un gestore code su IBM i per gestori code su piattaforme Windows, UNIX, Linux e z/OS .

Creare e concedere l'accesso all'ID utente 'mqm' se gerarchicamente collegato a un gestore code su Windows, UNIX o Linux per i gestori code su piattaforme IBM i e z/OS .

Creare e concedere l'accesso utente all'ID utente dello spazio di indirizzo dell'iniziatore del canale z/OS se è gerarchicamente collegato a un gestore code su z/OS per gestori code su piattaforme Windows, UNIX, Linux e IBM i .

Gli ID utente possono essere sensibili al maiuscolo / minuscolo. Il gestore code di origine (se i sistemi IBM i, Windows, UNIX o Linux ) impone che l'ID utente sia tutto maiuscolo. Il gestore code di ricezione (se i sistemi Windows, UNIX o Linux ) impone che l'ID utente sia tutto in minuscolo. Pertanto, tutti gli ID utente creati su sistemi UNIX and Linux devono essere creati in minuscolo. Se è stata installata un'uscita del messaggio, non viene eseguita la forzatura dell'ID utente in maiuscolo o in minuscolo. È necessario prestare attenzione al modo in cui l'uscita del messaggio elabora l'ID utente.

Per evitare potenziali problemi con la conversione degli ID utente:

- Sui sistemi UNIX, Linux, and Windows , assicurarsi che gli ID utente siano specificati in minuscolo.
- Su IBM i e z/OS, accertarsi che gli ID utente vengano specificati in maiuscolo.

## V 9.0.1 Sicurezza IBM MQ Console e REST API

La protezione per IBM MQ Console e REST API viene configurata modificando la configurazione del server mqweb nel file mqwebuser.xml .

### Informazioni su questa attività

È possibile tenere traccia delle azioni utente e controllare l'utilizzo di IBM MQ Console e REST API esaminando i file di log del server mqweb.

Gli utenti di IBM MQ Console e REST API possono essere autenticati utilizzando:

- Registro di base
- Registro LDAP



- **V 9.0.4** Registro SO locale
- [Interfaccia System Authorization Facility](#) su z/OS
- Qualsiasi altro tipo di registro supportato da WebSphere Application Server Liberty

I ruoli possono essere assegnati agli IBM MQ Console utenti e agli REST API utenti per determinare il livello di accesso concesso agli oggetti IBM MQ .

Una volta assegnato un ruolo a un utente, è possibile utilizzare diversi metodi per autenticare l'utente. Con IBM MQ Console, gli utenti possono accedere con un nome utente e una password oppure possono utilizzare l'autenticazione del certificato client. Con REST API, gli utenti possono utilizzare l'autenticazione HTTP di base, l'autenticazione basata su token o l'autenticazione del certificato client.

## Procedura

1. Definire il registro utenti per autenticare gli utenti e assegnare a ogni utente o gruppo un ruolo per autorizzare gli utenti e i gruppi ad utilizzare IBM MQ Console o REST API. Per ulteriori informazioni, consultare [“Configurazione di utenti e ruoli”](#) a pagina 466
2. Scegliere la modalità di autenticazione degli utenti di IBM MQ Console con il server mqweb. Non è necessario utilizzare lo stesso metodo per tutti gli utenti:
  - Consentire agli utenti di autenticarsi utilizzando l'autenticazione token. In questo caso, un utente immette un ID utente e una parola d'ordine nella schermata di accesso IBM MQ Console . Viene generato un token LTPA che consente all'utente di rimanere collegato e autorizzato per un periodo di tempo impostato. Non è richiesta alcuna ulteriore configurazione per utilizzare questa opzione di autenticazione, ma è possibile configurare facoltativamente la scadenza per il token LTPA. **V 9.0.1** Per ulteriori informazioni, consultare [Configurazione dell'intervallo di scadenza del token LTPA](#).
  - Consentire agli utenti di autenticarsi utilizzando i certificati client. In questo caso, l'utente non utilizza un ID utente o una password per accedere a IBM MQ Console, ma utilizza invece il certificato client. Per ulteriori informazioni, consultare [“Utilizzo dell'autenticazione del certificato client con REST API e IBM MQ Console”](#) a pagina 471.
3. **V 9.0.2** Scegliere la modalità di autenticazione degli utenti di REST API con il server mqweb. Non è necessario utilizzare lo stesso metodo per tutti gli utenti:
  - Consente agli utenti di autenticarsi utilizzando l'autenticazione di base HTTP. In questo caso, un nome utente e una parola d'ordine vengono codificati, ma non codificati e inviati con ogni richiesta REST API per autenticare e autorizzare l'utente per tale richiesta. Per rendere sicura questa autenticazione, è necessario utilizzare una connessione sicura. Ovvero, è necessario utilizzare HTTPS. Per ulteriori informazioni, consultare [“Utilizzo di autenticazione di base HTTP con REST API”](#) a pagina 475.
  - Consentire agli utenti di autenticarsi utilizzando l'autenticazione token. In questo caso, un utente fornisce un ID utente e una password alla risorsa REST API login con il metodo HTTP POST. Viene generato un token LTPA che consente all'utente di rimanere collegato e autorizzato per un periodo di tempo impostato. Per rendere sicura questa autenticazione, è necessario utilizzare una connessione sicura. Ovvero, è necessario utilizzare HTTPS. **V 9.0.1** Per ulteriori informazioni, consultare [Configurazione dell'intervallo di scadenza del token LTPA](#).
  - Consentire agli utenti di autenticarsi utilizzando i certificati client. In questo caso, l'utente non utilizza un ID utente o una password per accedere a REST API, ma utilizza invece il certificato client. Per ulteriori informazioni, consultare [“Utilizzo dell'autenticazione del certificato client con REST API e IBM MQ Console”](#) a pagina 471.
4. **V 9.0.2** Opzionale: Configurare la condivisione di risorse tra origini per REST API.

Per impostazione predefinita, un browser Web non consente agli script, come JavaScript, di richiamare REST API quando lo script non proviene dalla stessa origine di REST API. In altre parole, le richieste tra origini non sono abilitate. È possibile configurare CORS (Cross Origin Resource Sharing) per consentire richieste tra origini da URL specificati. Per ulteriori informazioni, consultare [“Configurazione di CORS per REST API”](#) a pagina 481.

## V 9.0.1 Configurazione di utenti e ruoli

Per utilizzare IBM MQ Console o REST API, gli utenti devono eseguire l'autenticazione rispetto a un registro utente, definito sul server mqweb.

### Informazioni su questa attività

Gli utenti autenticati devono essere membri di un gruppo che autorizza l'accesso alle funzioni di IBM MQ Console e REST API. Per impostazione predefinita, il registro utenti non contiene alcun utente; questi devono essere aggiunti modificando il file `mqwebuser.xml`.

Quando si configurano utenti e gruppi, si configura innanzitutto un registro utenti per autenticare utenti e gruppi. Questo registro utente è condiviso tra IBM MQ Console e REST API. È possibile controllare se gli utenti e i gruppi hanno accesso a IBM MQ Console, REST API o a entrambi quando si configurano i ruoli per gli utenti e i gruppi.

Dopo aver configurato il registro utenti, configurare i ruoli per gli utenti e i gruppi per concedere loro l'autorizzazione. Ci sono tre ruoli disponibili e ogni ruolo garantisce un diverso livello di accesso. Per ulteriori informazioni, consultare [“Ruoli su IBM MQ Console e su REST API”](#) a pagina 469.

Con il server mqweb vengono forniti diversi file XML di esempio per semplificare la configurazione di utenti e gruppi. Questa attività descrive come utilizzare gli esempi e modificarli per il proprio ambiente.

Gli utenti che hanno familiarità con la configurazione della sicurezza in WebSphere Application Server Liberty (WLP) potrebbero preferire di non utilizzare gli esempi. WLP fornisce altre funzionalità di autorizzazione oltre a quelle documentate qui.

**V 9.0.5** Per informazioni sui ruoli MFT e un esempio, consultare [“Configurazione della sicurezza MFTREST API”](#) a pagina 491

### Procedura

1. Assicurati di essere un [utente privilegiato](#).

2. Copiare uno dei file XML di esempio da uno dei seguenti percorsi:

- **ULW** Su UNIX, Linux, and Windows: `MQ_INSTALLATION_PATH /web/mq/samp/configuration`
- **z/OS** Su z/OS: `PathPrefix /web/mq/samp/configuration`  
dove `PathPrefix` è il percorso di installazione dei componenti di IBM MQ Unix System Services.
- **no\_security.xml**  
Questo esempio disabilita la sicurezza, inclusa la capacità di accedere a IBM MQ Console o a REST API, utilizzando HTTPS.
- **basic\_registry.xml**  
Questo esempio definisce un registro di base di utenti e gruppi.  
I nomi utente e le password nel registro vengono utilizzati per autenticare e autorizzare gli utenti di IBM MQ Console e REST API.
- **ULW** **V 9.0.4**  
**local\_os\_registry.xml**  
Questo esempio configura l'utilizzo di utenti e gruppi del sistema operativo locale.

Ai membri del gruppo 'mqm' viene concesso il ruolo MQWebAdmin e a tutti gli utenti autenticati viene concesso il ruolo MQWebUser .

I nomi utente e le parole d'ordine nel registro del sistema operativo vengono utilizzati per autenticare e autorizzare gli utenti di IBM MQ Console e REST API.

- **ldap\_registry.xml**

Questo esempio definisce una connessione a un registro LDAP da cui vengono richiamate le informazioni su utenti e gruppi.

I nomi utente e le password nel registro LDAP vengono utilizzati per autenticare e autorizzare l'utilizzo di IBM MQ Console e REST API.

- **V 9.0.5** **z/OS** **zos\_saf\_registry.xml**

Questo esempio configura l'utilizzo dell'interfaccia SAF (System authorization facility) su z/OS.

I profili RACF, o altri prodotti di sicurezza, vengono utilizzati per concedere agli utenti e ai gruppi l'accesso ai ruoli.

I nomi utente e le password nel database RACF vengono utilizzati per autenticare e autorizzare gli utenti di IBM MQ Console e REST API.

3. Inserire il file di esempio nella directory appropriata:

- **ULW**

Su UNIX, Linux, and Windows: `MQ_DATA_DIRECTORY/web/installations/  
installationName/servers/mqweb`

- **z/OS**

Su z/OS: `WLP_user_directory/servers/mqweb`

dove `WLP_user_directory` è la directory specificata quando è stato eseguito lo script `crtmqweb.sh` per creare la definizione del server mqweb.

4. Opzionale: Se sono state modificate le impostazioni di configurazione in `mqwebuser.xml`, copiarle nel file di esempio.

5. Eliminare il file `mqwebuser.xml` esistente e ridenominarlo come `mqwebuser.xml`.

6. Modificare il nuovo file `mqwebuser.xml` per aggiungere utenti e gruppi come necessario:

- Per le configurazioni di sicurezza basate sull'esempio `basic_registry.xml` , aggiungere utenti e gruppi all'interno delle tag **basicRegistry** .

Tenere presente che qualsiasi utente con il ruolo MQWebUser può eseguire solo le operazioni che l'ID utente può eseguire sul gestore code. Pertanto, l'ID definito nel registro deve avere un ID utente identico sul sistema su cui è installato IBM MQ . Questi ID utente devono essere nello stesso caso oppure l'associazione tra gli ID utente potrebbe non riuscire.

Per ulteriori informazioni sulla configurazione dei registri utente di base, consultare [Configurazione di un registro utente di base per Liberty](#) nella documentazione di WebSphere Application Server Liberty .

- **ULW** **V 9.0.4**

Per le configurazioni di sicurezza basate sull'esempio `local_os_registry.xml` , il registro accede al sistema operativo locale per convalidare le parole d'ordine, identificare gli utenti e calcolare l'appartenenza al gruppo. Questo tipo di registro utenti viene abilitato aggiungendo **<feature>localOSAuthenticationMQ-1.0</feature>** alla sezione **featureManager** del file `mqwebuser.xml` .

Per l'autenticazione del certificato client con la funzione di autenticazione del SO locale, l'identità utente è il CN (common name) dal DN (distinguished name) del certificato client. Se l'identità utente non esiste come utente del sistema operativo, il login del certificato client avrà esito negativo e il fallback all'autenticazione basata sulla password.

- Per le impostazioni di sicurezza basate sull'esempio `ldap_registry.xml`, modificare le impostazioni di registro LDAP all'interno delle tag **ldapRegistry** e **idsLdapFilterProperties**.

Tenere presente che qualsiasi utente con il ruolo `MQWebUser` può eseguire solo le operazioni che l'ID utente può eseguire sul gestore code. Pertanto, l'ID utente definito sul server LDAP deve avere un ID utente identico sul sistema su cui è installato IBM MQ. Questi ID utente devono essere nello stesso caso oppure l'associazione tra gli ID utente potrebbe non riuscire.

Per ulteriori informazioni sulla configurazione dei registri LDAP, consultare [Configurazione dei registri utenti LDAP in Liberty](#) nella documentazione WebSphere Application Server Liberty.

#### 7. Modificare il nuovo file `mqwebuser.xml` per assegnare ruoli a utenti e gruppi.

Sono disponibili tre ruoli che autorizzano utenti e gruppi ad utilizzare IBM MQ Console e REST API. Ogni ruolo concede un diverso livello di accesso. Per ulteriori informazioni, consultare ["Ruoli su IBM MQ Console e su REST API"](#) a pagina 469.

- Per assegnare i ruoli e concedere l'accesso a IBM MQ Console, aggiungere i propri utenti e gruppi tra le tag **security-role** appropriate all'interno delle tag **<enterpriseApplication id="com.ibm.mq.console">**.

#### • **V 9.0.2**

Per assegnare i ruoli e concedere l'accesso a REST API, aggiungere i propri utenti e gruppi tra le tag **security-role** appropriate all'interno delle tag **<enterpriseApplication id="com.ibm.mq.rest">**.

Per informazioni sul formato delle informazioni utente e gruppo all'interno delle tag **security-role**, consultare l'esempio.

8. Se hai fornito le password per gli utenti in `mqwebuser.xml`, devi codificarle, per renderle più sicure, utilizzando il comando **securityUtility encoding** fornito da WebSphere Application Server Liberty. Per ulteriori informazioni, consultare [Liberty:securityUtility command](#) nella documentazione del prodotto WebSphere Application Server Liberty.

### Esempio

Nel seguente esempio, al gruppo `MQWebAdminGroup` viene concesso l'accesso a IBM MQ Console con il ruolo `MQWebAdmin`. All'utente, `reader`, viene concesso l'accesso con il ruolo `MQWebAdminRO` e all'utente `guest` viene concesso l'accesso con il ruolo `MQWebUser`:

```
<enterpriseApplication id="com.ibm.mq.console">
  <application-bnd>
    <security-role name="MQWebAdmin">
      <group name="MQWebAdminGroup" realm="defaultRealm"/>
    </security-role>
    <security-role name="MQWebAdminRO">
      <user name="reader" realm="defaultRealm"/>
    </security-role>
    <security-role name="MQWebUser">
      <user name="guest" realm="defaultRealm"/>
    </security-role>
  </application-bnd>
</enterpriseApplication>
```

Nel seguente esempio, agli utenti `reader` e a `guest` viene concesso l'accesso a IBM MQ Console. All'utente `user` viene concesso l'accesso al REST API e a tutti gli utenti all'interno del gruppo `MQAdmin` viene concesso l'accesso al IBM MQ Console e al REST API:

```
<enterpriseApplication id="com.ibm.mq.console">
  <application-bnd>
    <security-role name="MQWebAdmin">
      <group name="MQAdmin" realm="defaultRealm"/>
    </security-role>
    <security-role name="MQWebAdminRO">
      <user name="reader" realm="defaultRealm"/>
    </security-role>
    <security-role name="MQWebUser">
```

```

        <user name="guest" realm="defaultRealm"/>
    </security-role>
</application-bnd>
</enterpriseApplication>

<enterpriseApplication id="com.ibm.mq.rest">
  <application-bnd>
    <security-role name="MQWebAdmin">
      <group name="MQAdmin" realm="defaultRealm"/>
    </security-role>
    <security-role name="MQWebUser">
      <user name="user" realm="defaultRealm"/>
    </security-role>
  </application-bnd>
</enterpriseApplication>

```

## Operazioni successive

Scegliere il modo in cui gli utenti eseguono l'autenticazione:

### IBM MQ Console Opzioni di autenticazione

- Consentire agli utenti di autenticarsi utilizzando l'autenticazione token. In questo caso, un utente immette un ID utente e una parola d'ordine nella schermata di accesso IBM MQ Console . Viene generato un token LTPA che consente all'utente di rimanere collegato e autorizzato per un periodo di tempo impostato. Non è richiesta alcuna ulteriore configurazione per utilizzare questa opzione di autenticazione, ma è possibile, facoltativamente, configurare l'intervallo di scadenza per il token LTPA. **V 9.0.1** Per ulteriori informazioni, consultare [Configurazione dell'intervallo di scadenza del token LTPA](#).
- Consentire agli utenti di autenticarsi utilizzando i certificati client. In questo caso, l'utente non utilizza un ID utente o una password per accedere a IBM MQ Console, ma utilizza invece il certificato client. Per ulteriori informazioni, consultare ["Utilizzo dell'autenticazione del certificato client con REST API e IBM MQ Console"](#) a pagina 471.

### **V 9.0.2** REST API Opzioni di autenticazione

- Consente agli utenti di autenticarsi utilizzando l'autenticazione di base HTTP. In questo caso, un nome utente e una parola d'ordine vengono codificati, ma non codificati e inviati con ogni richiesta REST API per autenticare e autorizzare l'utente per tale richiesta. Per rendere sicura questa autenticazione, è necessario utilizzare una connessione sicura. Ovvero, è necessario utilizzare HTTPS. Per ulteriori informazioni, consultare ["Utilizzo di autenticazione di base HTTP con REST API"](#) a pagina 475.
- Consentire agli utenti di autenticarsi utilizzando l'autenticazione token. In questo caso, un utente fornisce un ID utente e una password alla risorsa REST API login con il metodo HTTP POST. Viene generato un token LTPA che consente all'utente di rimanere collegato e autorizzato per un periodo di tempo impostato. Per ulteriori informazioni, consultare ["Utilizzo dell'autenticazione basata su token con la API REST per IBM MQ 9.0.4 e versioni precedenti"](#) a pagina 479. È possibile configurare l'intervallo di scadenza per il token LTPA. **V 9.0.1** Per ulteriori informazioni, consultare [Configurazione dell'intervallo di scadenza del token LTPA](#).
- Consentire agli utenti di autenticarsi utilizzando i certificati client. In questo caso, l'utente non utilizza un ID utente o una password per accedere a REST API, ma utilizza invece il certificato client. Per ulteriori informazioni, consultare ["Utilizzo dell'autenticazione del certificato client con REST API e IBM MQ Console"](#) a pagina 471.

### **V 9.0.1** Ruoli su IBM MQ Console e su REST API

Quando si autorizzano utenti e gruppi a utilizzare IBM MQ Console o REST API, è necessario assegnare agli utenti e ai gruppi uno dei tre ruoli: **MQWebAdmin**, **MQWebAdminRO** e **MQWebUser**. Ogni ruolo fornisce diversi livelli di privilegi per accedere a IBM MQ Console e REST API e determina il contesto di sicurezza utilizzato quando viene tentata un'operazione consentita.

**Nota:** Ad eccezione del ruolo **MQWebUser** , l'ID utente non è sensibile al maiuscolo / minuscolo. Consultare [“MQWebUser”](#) a pagina 470 per i requisiti specifici per questo ruolo.

### **MQWebAdmin**

Un utente o un gruppo a cui viene assegnato questo ruolo può eseguire tutte le operazioni e operare nel contesto di sicurezza dell'ID utente del sistema operativo utilizzato per avviare il server mqweb.

### **MQWebAdminRO**

Questo ruolo fornisce l'accesso di sola lettura a IBM MQ Console o REST API. Un utente o un gruppo a cui è assegnato questo ruolo può eseguire le seguenti operazioni:

- Visualizza e interroga operazioni su oggetti IBM MQ come code e canali.
- Sfoglia messaggi sulle code.

Un utente o un gruppo a cui è assegnato questo ruolo opera nel contesto di sicurezza dell'ID utente del sistema operativo utilizzato per avviare il server mqweb.

### **MQWebUser**

Un utente o un gruppo a cui è assegnato questo ruolo può eseguire qualsiasi operazione che l'ID utente può eseguire sul gestore code. Ad esempio:

- Operazioni di avvio e arresto su oggetti IBM MQ come i canali.
- Definire e impostare operazioni su oggetti IBM MQ come code e canali.
- Visualizza e interroga operazioni su oggetti IBM MQ come code e canali.

Un utente o un gruppo a cui è assegnato questo ruolo opera nel contesto di sicurezza del principal e può eseguire solo le operazioni che l'ID utente è autorizzato ad eseguire sul gestore code.

Pertanto, l'utente o il gruppo definito nel registro utente mqweb deve disporre dell'autorizzazione all'interno di IBM MQ prima che l'utente possa eseguire qualsiasi operazione. Utilizzando questo ruolo, puoi controllare finemente quali utenti hanno quale tipo di accesso a specifiche risorse IBM MQ quando utilizzano IBM MQ Console e REST API.

#### **Nota:**

- La lunghezza massima di un ID utente assegnato a questo ruolo è di 12 caratteri.
- Il maiuscolo / minuscolo dell'ID utente deve essere lo stesso nel registro utente mqweb e sul sistema IBM MQ . Se il caso dell'ID utente è diverso, l'utente potrebbe essere autenticato da IBM MQ Console e REST API ma non autorizzato ad utilizzare le risorse IBM MQ .

Per ulteriori informazioni sulla configurazione di utenti e gruppi per l'utilizzo di questi ruoli, consultare [“Configurazione di utenti e ruoli”](#) a pagina 466.

## **Ruoli di sovrapposizione**

A un utente o a un gruppo può essere assegnato più di un ruolo. Quando un utente esegue un'operazione in questa situazione, viene utilizzato il ruolo con privilegi più elevati applicabile all'operazione. Ad esempio, se un utente con i ruoli **MQWebAdminRO** e **MQWebUser** esegue un'operazione di interrogazione della coda, viene utilizzato il ruolo **MQWebAdminRO** e l'operazione viene tentata nel contesto dell'ID utente di sistema che ha avviato il server Web. Se lo stesso utente esegue un'operazione di definizione, viene utilizzato il ruolo **MQWebUser** e l'operazione viene tentata nel contesto del principal.

## **Sicurezza MFT REST API**

V 9.0.5

Per informazioni su utenti, ruoli e sicurezza MFT , consultare [“Configurazione della sicurezza MFTREST API”](#) a pagina 491

## Utilizzo dell'autenticazione del certificato client con REST API e IBM MQ Console

È possibile associare i certificati client ai principal per autenticare utenti IBM MQ Console e REST API .

### Prima di iniziare

- Configurare utenti, gruppi e ruoli per essere autorizzati ad utilizzare IBM MQ Console e REST API. Per ulteriori informazioni, consultare [“Configurazione di utenti e ruoli”](#) a pagina 466.
- Quando si utilizza REST API, è possibile eseguire una query delle credenziali dell'utente corrente utilizzando il metodo HTTP GET sulla risorsa `login` , fornendo il certificato client per autenticare la richiesta. Questa richiesta restituisce informazioni sul metodo di autenticazione, il nome utente e i ruoli assegnati all'utente. Per ulteriori informazioni, vedi [GET /login](#).
- Quando si associano i certificati client ai principal per autenticare gli utenti, il DN (distinguished name) del certificato client viene utilizzato per la corrispondenza con gli utenti nel registro utenti configurato:
  - Per un registro di base, il CN (Common Name) viene confrontato con l'utente. Ad esempio, CN=Fred , O=IBM, C=GB viene confrontato con un nome utente Fred.
  - Per un registro LDAP, per impostazione predefinita il DN (distinguished name) completo viene confrontato con LDAP. È possibile impostare filtri e associazioni per personalizzare la corrispondenza. Per ulteriori informazioni, consultare [Liberty :LDAP certificate map mode](#) nella documentazione WebSphere Application Server Liberty .

### Informazioni su questa attività

Quando un utente esegue l'autenticazione utilizzando un certificato client, il certificato viene utilizzato al posto di un nome utente e di una password. Per REST API, il certificato client viene fornito con ogni richiesta REST per autenticare l'utente. Per IBM MQ Console, quando un utente accede con un certificato, l'utente non può essere scollegato.

La procedura presuppone le seguenti informazioni:

- Il file `mqwebuser.xml` si basa su uno dei seguenti esempi:
  - `basic_registry.xml`
  - `V 9.0.4 local_os_registry.xml`
  - `ldap_registry.xml`
  - `V 9.0.5 zos_saf_registry.xml`
- Che si sta utilizzando un sistema UNIX, Linux o Windows .
- L'utente è un [utente privilegiato](#).

**Nota:** La seguente procedura descrive i passi necessari per utilizzare i certificati del client con IBM MQ Console e REST API. Per comodità dello sviluppatore, la procedura descrive come creare e utilizzare i certificati autofirmati. Tuttavia, per la produzione, utilizzare i certificati ottenuti da un'autorità di certificazione.

### Procedura

1. Avviare il server `mqweb` immettendo il comando `strmqweb` sulla riga comandi.
2. Creare un certificato client:
  - a) Creare un keystore PKCS#12 :
    - i) Aprire lo strumento IBM Key Management immettendo il comando `strmqikm` sulla riga comandi.
    - ii) Dal menu **File database di chiavi** nello strumento IBM Key Management, fare clic su **Nuovo**.

- iii) Selezionare **PKCS12** dall'elenco **Tipo database delle chiavi**.
  - iv) Selezionare un percorso per salvare il keystore e immettere un nome appropriato nel campo **Nome file**. Ad esempio, `user.p12`
  - v) Impostare una password quando richiesto.
- b) Creare il certificato, creando un certificato autofirmato o ottenendo un certificato da un'autorità di certificazione:
- Creare un certificato autofirmato:
    - i) Fare clic su **Nuovo autofirmato**.
    - ii) Immettere `user` nel campo **Etichetta chiave**.
    - iii) Se si sta utilizzando un registro utente di base, immettere il nome di un utente dal proprio registro utente nel campo **Nome comune**. Ad esempio, `mqadmin`. Per un registro utenti LDAP, assicurarsi che il DN (distinguished name) per il certificato corrisponda al DN (distinguished name) nel registro LDAP.
    - iv) Fare clic su **OK**.
  - Ottenere un certificato da un'autorità di certificazione. Il certificato CA deve includere il nome utente appropriato all'interno del nome comune (CN) del campo DN (distinguished name):
    - i) Richiedere un nuovo certificato. Dal menu **Crea**, fare clic su **Nuova richiesta certificato**.
    - ii) Nel campo **Etichetta chiave**, immettere l'etichetta del certificato.
    - iii) Se si sta utilizzando un registro utente di base, nel campo **Nome comune**, immettere il nome utente dell'utente a cui è destinato il certificato.  
  
 Se si utilizza un registro del sistema operativo locale, il campo **Nome comune** deve corrispondere all'ID utente del sistema operativo locale.  
  
 Per un registro utenti LDAP, assicurarsi che il DN (distinguished name) per il certificato corrisponda al DN (distinguished name) nel registro LDAP.
    - iv) Immettere o selezionare i valori per i restanti campi, come applicabile.
    - v) Scegliere dove salvare la richiesta di certificato e il nome file per la richiesta di certificato, quindi fare clic su **OK**.
    - vi) Inviare il file di richiesta certificato a una CA (Certificate Authority).
    - vii) Quando si dispone del certificato dalla CA, aprire lo strumento IBM Key Management immettendo il comando **strmqikm** dalla riga comandi.
    - viii) Dal menu **File database di chiavi** nello strumento IBM Key Management, fare clic su **Apri**.
    - ix) Selezionare il keystore PKCS#12 che contiene il certificato client. Ad esempio, `user.p12`
    - x) Fare clic su **Ricevi**, selezionare il certificato appropriato e fare clic su **OK**.
3. Estrarre la parte pubblica del certificato client:
- a) Aprire lo strumento IBM Key Management immettendo il comando **strmqikm** sulla riga comandi.
  - b) Dal menu **File database di chiavi** nello strumento IBM Key Management, fare clic su **Apri**.
  - c) Selezionare il keystore PKCS#12 che contiene il certificato client. Ad esempio, `user.p12`
  - d) Selezionare il certificato client dall'elenco di certificati nello strumento IBM Key Management.
  - e) Fare clic su **Estrai certificato**.
  - f) Selezionare un'ubicazione in cui salvare il certificato e immettere un nome file appropriato nel campo **Nome file certificato**. Ad esempio, `user.arm`.
4. Importare la parte pubblica del certificato client nel keystore di attendibilità del server mqweb come certificato firmatario in modo che il server possa convalidare il certificato client:
- a) Creare un keystore `trust.jks` per l'utilizzo da parte del server mqweb, se non ne esiste già uno:
    - i) Dal menu **File database di chiavi** nello strumento IBM Key Management, fare clic su **Nuovo**.
    - ii) Selezionare **JKS** dall'elenco **Key database type**.



- iii) Fare clic su **Sfogli** e passare a `MQ_DATA_DIRECTORY/web/installations/  
installationName/servers/mqweb/resources/security`.

Questa directory deve contenere già un file `key.jks`. Se esiste già un file `trust.jks`, aprire quello esistente invece di sovrascriverlo.

- iv) Immettere `trust.jks` nel campo **Nome file**.

- v) Impostare una password quando richiesto.

- b) Dal menu a discesa, selezionare **Certificati firmatario**.
  - c) Fare clic su **Aggiungi**.
  - d) Selezionare il file `arm` appropriato e fare clic su **OK**. Ad esempio, selezionare `user.arm`.
  - e) Immettere un'etichetta per il certificato.
5. Modificare la parola d'ordine del keystore del server `mqweb`:
- a) Dal menu **File del database delle chiavi**, fare clic su **Apri**.
  - b) Selezionare **JKS** dall'elenco **Key database type**.
  - c) Fare clic su **Sfogli** e passare a `MQ_DATA_DIRECTORY/web/installations/  
installationName/servers/mqweb/resources/security`
  - d) Selezionare il keystore `key.jks` e fare clic su **Apri**.
  - e) Immettere la password quando richiesto. La password predefinita è `password`.
  - f) Nel menu **File database di chiavi**, fare clic su **Modifica password**.
  - g) Immettere una nuova password per il keystore.
6. Abilitare l'autenticazione del certificato client nel file `mqwebuser.xml`:

Il file `mqwebuser.xml` si trova nel seguente percorso: `MQ_DATA_DIRECTORY/web/  
installations/installationName/servers/mqweb`

- a) Eliminare il commento dalla sezione del file `mqwebuser.xml` che abilita l'autenticazione del certificato client. La sezione contiene il seguente testo:

```
<keyStore id="defaultKeyStore" location="key.jks" type="JKS" password="password"/>  
<keyStore id="defaultTrustStore" location="trust.jks" type="JKS" password="password"/>  
<ssl id="thisSSLConfig" clientAuthenticationSupported="true"  
keyStoreRef="defaultKeyStore"  
trustStoreRef="defaultTrustStore" sslProtocol="TLSv1.2"  
serverKeyAlias="default"/>  
<sslDefault sslRef="thisSSLConfig"/>
```

- b) Verificare che il valore **serverKeyAlias** corrisponda al nome del certificato server. Se si sta utilizzando il certificato server predefinito, il valore è corretto.
- c) Modificare il valore per **password** per `defaultKeyStore` in una versione codificata della password per il keystore `key.jks`:
  - i) Dalla directory `MQ_INSTALLATION_PATH/web/bin`, immettere il seguente comando sulla riga comandi:

```
securityUtility encode password
```

- ii) Inserire l'output di questo comando nel campo **password** per `defaultKeyStore`.

- d) Modificare il valore per **password** per `defaultTrustStore` in modo che corrisponda alla password per il keystore `trust.jks`:

- i) Dalla directory `MQ_INSTALLATION_PATH/web/bin`, immettere il seguente comando sulla riga comandi:

```
securityUtility encode password
```

- ii) Inserire l'output di questo comando nel campo **password** per `defaultTrustStore`.

- e) Rimuovere o commentare:

```
<sslDefault sslRef="mqDefaultSSLConfig"/>
```

da `mqwebuser.xml`.

7. Arrestare il server mqweb immettendo il comando **endmqweb** sulla riga comandi.
8. Avviare il server mqweb immettendo il comando **strmqweb** sulla riga comandi.
9. Utilizzare il certificato client per autenticare:
  - Per utilizzare il certificato client con IBM MQ Console, installare il certificato client sul browser Web utilizzato per accedere a IBM MQ Console. Ad esempio, installare il certificato client `user.p12` come certificato personale.
  - **V 9.0.2** Per utilizzare il certificato client con REST API, fornire il certificato client con ogni richiesta REST. Quando si utilizzano i metodi HTTP POST, PATCH o DELETE, è necessario fornire un'ulteriore autenticazione con il certificato client per evitare attacchi CSRF (cross - site request forgery). In altre parole, l'autenticazione supplementare viene utilizzata per confermare che le credenziali utilizzate per autenticare la richiesta vengono utilizzate dal proprietario delle credenziali.

Questa autenticazione aggiuntiva viene fornita dall'intestazione HTTP `ibm-mq-rest-csrf-token`. Il contenuto richiesto di questa intestazione varia a seconda della versione di IBM MQ.

Per IBM MQ 9.0.4 e versioni precedenti, calcola il valore dell'intestazione HTTP `ibm-mq-rest-csrf-token`:

- a. Generazione di un cookie token CSRF, inoltrando una richiesta HTTP GET sulla risorsa `login` REST API. Utilizzare il certificato client per autenticare la richiesta.
- b. L'impostazione del valore dell'intestazione `ibm-mq-csrf-token` sul valore del cookie del token CSRF, `csrfToken`, restituito dalla richiesta.

Si noti che non è possibile utilizzare una versione memorizzata nella cache del contenuto del cookie, perché il contenuto del cookie può cambiare. È necessario utilizzare l'ultimo valore del cookie per ogni richiesta.

**V 9.0.5** Per IBM MQ 9.0.5 e versioni successive, impostare il valore dell'intestazione `ibm-mq-csrf-token` su qualsiasi valore incluso il valore vuoto. L'intestazione deve essere impostata nella richiesta, ma il valore non viene controllato.

Quindi, in entrambi i casi, inoltrare la richiesta.

## Esempio

**V 9.0.2**

**Importante:** Nell'esempio, non tutte le implementazioni cURL supportano i certificati autofirmati, quindi devi utilizzare un curl che lo faccia.

Il seguente esempio cURL mostra come creare una nuova coda Q1, sul gestore code QM1, con l'autenticazione del certificato client. L'esatta configurazione di questo comando cURL dipende dalle librerie su cui è stato creato cURL. L'esempio si basa su un sistema Windows, con cURL creato rispetto a OpenSSL:

- **V 9.0.5** Da IBM MQ 9.0.5, è necessario emettere solo una singola richiesta HTTP. Utilizzare il metodo HTTP POST con la risorsa coda, autenticandosi con il certificato client e includendo l'intestazione HTTP `ibm-mq-rest-csrf-token` con un valore arbitrario. Questo valore può essere qualsiasi o vuoto; non viene controllato dal server mqweb.

IBM MQ 9.0.5 e successive:

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X POST -  
-cert-type P12 --cert c:\user.p12:password  
-H "ibm-mq-rest-csrf-token: value"  
-H "Content-Type: application/json" --data '{"name": "Q1"}'
```

- Per IBM MQ 9.0.4 e versioni precedenti sono necessarie due richieste HTTP:

1. La prima richiesta genera il cookie token CSRF.

Utilizzare il metodo HTTP GET con la risorsa `login`, autenticandosi con il certificato client. Il token CSRF restituito viene memorizzato nel file `cookiejar.txt`. L'indicatore `--cert-type1` specifica che il certificato è un PKCS#12. L'indicatore `--cert` specifica l'ubicazione del certificato, seguito da due punti, `:` e quindi la parola d'ordine per il certificato. L'indicatore `-c` specifica l'ubicazione del file in cui memorizzare il token CSRF:

```
curl -k https://localhost:9443/ibmmq/rest/v1/login --cert-type P12 --cert
c:\user.p12:password -c c:\cookiejar.txt
```

2. La seconda richiesta crea la coda.

Utilizzare il metodo HTTP POST con la risorsa della coda, autenticandosi con il certificato client e includendo il contenuto del token CSRF in un'intestazione:

IBM MQ 9.0.3 e precedenti:

```
curl -k https://localhost:9443/ibmmq/rest/v1/qmgr/QM1/queue -X POST --cert-type P12 --cert
c:\user.p12:password
-H "ibm-mq-rest-csrf-token:
A6E85DE02E15EEAD2DBE49C0BCD6F191A7FA9535A161A4B7F019F2DE4625A8A5D24191E2409D2011F8700
8E28786BD4E0ABD202B26C89360E8DA1F77CE737167A08E54AAB7C932CED1CF040E1F83C1D87F49C50BE724803A
1642DB23FDF959CAA27B12
A5BE36259FC2D9F92199B8AFFE886670EE454BD4B14A32C8A24E574F1FA0F251686B3670ED24F301615FC417BD4
0457CFCC92AC15C1A1C567
3BCBB43963598FA9A1A77D91F53861290B0598AADD04591CEFB018D6D55BB157A4F67FDB4D203AD42EB82799AC6
1CA0071417446C9E47A982
099C0331AE4F3CD9C2F1611FC5FB122CE3C0F0A60AAC166EF98DE19234A524265391F9BF5496695EEAFED847"
-H "Content-Type: application/json" --data '{"name":"Q1"}'
```

## V 9.0.2 Utilizzo di autenticazione di base HTTP con REST API

Gli utenti di REST API possono eseguire l'autenticazione fornendo il loro ID utente e la password all'interno di una intestazione HTTP. **V 9.0.5** Per utilizzare questo metodo di autenticazione con i metodi HTTP, ad esempio POST, PATCH e DELETE, è necessario fornire anche l'intestazione HTTP `ibm-mq-rest-csrf-token`, nonché un ID utente e una password.

### Prima di iniziare

- Configurare gli utenti, i gruppi e i ruoli per essere autorizzati all'utilizzo di REST API. Per ulteriori informazioni, consultare ["Configurazione di utenti e ruoli"](#) a pagina 466.
- Accertarsi che l'autenticazione di base HTTP sia abilitata. Verificare che il seguente XML sia presente e non sia commentato nel file `mqwebuser.xml`. Questo XML deve essere all'interno delle tag `<featureManager>`:

```
<feature>basicAuthenticationMQ-1.0</feature>
```

È necessario essere un utente privilegiato per modificare il file `mqwebuser.xml`.

- Assicurarsi di utilizzare una connessione sicura quando si inviano richieste REST. Poiché la combinazione di nome utente e password è codificata, ma non codificata, è necessario utilizzare una connessione sicura (HTTPS) quando si utilizza l'autenticazione di base HTTP con REST API.
- È possibile eseguire una query delle credenziali dell'utente corrente utilizzando il metodo HTTP GET sulla risorsa `login`, fornendo le informazioni di autenticazione di base per autenticare la richiesta. Questa richiesta restituisce informazioni sul metodo di autenticazione, il nome utente e i ruoli assegnati all'utente. Per ulteriori informazioni, vedi [GET /login](#).

### Procedura

1. Concatenare il nome utente con i due punti e la password.

Ad esempio, un nome utente admin e una password admin diventano la seguente stringa:

```
admin:admin
```

2. Codifica questa stringa nome utente e password nella codifica base64 .
3. Includere questo nome utente e password codificati in un'intestazione HTTP Authorization: Basic .

Ad esempio, con un nome utente codificato di admin e una password di admin, viene creata la seguente intestazione:

```
Authorization: Basic YWRtaW46YWRtaW4=
```

#### 4. **V 9.0.5**

Quando si utilizzano i metodi HTTP POST, PATCH o DELETE, è necessario fornire un'ulteriore autenticazione, nonché un nome utente e una password.

Questa autenticazione aggiuntiva viene fornita dall'intestazione HTTP `ibm-mq-rest-csrf-token` . Il contenuto richiesto di questa intestazione varia a seconda della versione di IBM MQ.

Per IBM MQ 9.0.4 e versioni precedenti, il valore dell'intestazione viene preso da un cookie token CSRF. Per ottenere questo cookie è necessario eseguire la seguente procedura:

- a) Generare un cookie token CSRF inoltrando una richiesta HTTP GET sulla risorsa `login REST API` . Utilizzare l'autenticazione di base nome utente e password indicata in questa procedura per autenticare la richiesta.
- b) Inserire il contenuto del token CSRF, `csrfToken`, restituito dalla richiesta in un'intestazione HTTP aggiuntiva come valore dell'intestazione. L'intestazione deve essere denominata `ibm-mq-rest-csrf-token`.

Il contenuto del cookie `csrfToken` viene utilizzato per confermare che le credenziali utilizzate per autenticare la richiesta vengono utilizzate dal proprietario delle credenziali. Ovvero, il token viene utilizzato per evitare attacchi cross - site request forgery.

Non è possibile utilizzare una versione memorizzata nella cache del contenuto del cookie perché il contenuto del cookie può cambiare. È necessario utilizzare l'ultimo valore del cookie per ogni richiesta.

Da IBM MQ 9.0.5, l'intestazione HTTP `ibm-mq-rest-csrf-token` deve essere presente nella richiesta; il suo valore può essere qualsiasi cosa, incluso il vuoto.

5. Inoltra la tua richiesta REST a IBM MQ con le intestazioni appropriate.

### Esempio

Il seguente esempio mostra come creare una nuova coda Q1, sul gestore code QM1, con autenticazione di base, sui sistemi Windows . L'esempio utilizza cURL:

- **V 9.0.5** Da IBM MQ 9.0.5, è necessario emettere solo una singola richiesta HTTP. Utilizzare il metodo HTTP POST con la risorsa della coda, autenticandosi con l'autenticazione di base e includendo l'intestazione HTTP `ibm-mq-rest-csrf-token` con un valore arbitrario. Questo valore può essere qualsiasi o vuoto; non viene controllato dal server mqweb.

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X POST
-u mqadmin:mqadmin
-H "ibm-mq-rest-csrf-token: value"
-H "Content-Type: application/json" --data "{\"name\":\"Q1\"}"
```

- Per IBM MQ 9.0.4 e versioni precedenti, sono richieste due richieste HTTP:
  1. La prima richiesta genera il cookie token CSRF.

Utilizzare il metodo HTTP GET con la risorsa di login, autenticandosi con l'autenticazione di base. Il token CSRF restituito viene memorizzato nel file `cookiejar.txt`. L'indicatore `-u` specifica il nome utente e la password. L'indicatore `-c` specifica l'ubicazione del file in cui memorizzare il token:

```
curl -k https://localhost:9443/ibmmq/rest/v1/login -u admin:admin -c c:\cookiejar.txt
```

2. La seconda richiesta crea la coda.

Utilizzare il metodo HTTP POST con la risorsa coda, autenticandosi con l'autenticazione di base e includendo il contenuto del token CSRF in un'intestazione:

IBM MQ 9.0.3 e precedenti:

```
curl -k https://localhost:9443/ibmmq/rest/v1/qmgr/QM1/queue -X POST -u admin:admin
-H "ibm-mq-rest-csrf-token:
83F1817976A6E6F1E980F9F09D7E8A161DC9D9867A634497CE03667B2AF5532B5
E6F9314E40B4FB31E00A2885E07150C0FD06FFB07B46FD4A5D2DA4C239C2D82C3C87588C8850B975892E1AF9603
4F
05F41699A7A1D36DEC048CE18F49A195BB762020D420EB628568D30120F12538B53D3F91939EF8851863EC7B87B
6E
B0F95B57B6AB68B61D4324FAA3DFDE05AC956556736F8A9CA5BAF89BC2174B0EF5CE04E65646626F788F1CE2284
EF
1562868C5A800B8BF4BFB8FB6C3FCD194EA6EB2FF43A3CFB57CCF9F5EF76F0E724FAB645B8E14CD3D9484BF799B
3B
090CCD67B6CE8C8DAB552018A538903B0CD0B9FD747F2F4C18A80A65A2C3AE2A0D631B298AF"
-H "Content-Type: application/json" --data '{"name":"Q1"}'
```

## Utilizzo dell'autenticazione basata su token con REST API

Come configurare l'autenticazione basata su token con REST API da IBM MQ 9.0.5e per IBM MQ 9.0.4 e versioni precedenti.

### **V 9.0.5** Utilizzo dell'autenticazione basata su token con REST API da IBM MQ 9.0.5

Gli utenti di REST API possono eseguire l'autenticazione fornendo un ID utente e una password alla risorsa REST API `login` con il metodo HTTP POST. Viene generato un token LTPA che consente all'utente di autenticare richieste future. L'utente può scollegarsi utilizzando il metodo HTTP DELETE e può interrogare le informazioni di accesso dell'utente corrente con il metodo HTTP GET.

#### Prima di iniziare

- Configurare gli utenti, i gruppi e i ruoli per essere autorizzati all'utilizzo di REST API. Per ulteriori informazioni, consultare [“Configurazione di utenti e ruoli”](#) a pagina 466.
- Facoltativamente, configurare la scadenza per il token LTPA. Per ulteriori informazioni, consultare [Configurazione del timer di scollegamento utente](#).
- Assicurarsi di utilizzare una connessione sicura quando si inviano richieste REST. Quando utilizzi il metodo HTTP POST sulla risorsa `login`, la combinazione di nome utente e password inviata con la richiesta non viene codificata. Pertanto, è necessario utilizzare una connessione protetta (HTTPS) quando si utilizza l'autenticazione basata su token con REST API.
- È possibile interrogare le credenziali dell'utente corrente utilizzando il metodo HTTP GET sulla risorsa `login`, fornendo il token LTPA, `LtpaToken2`, per autenticare la richiesta. Questa richiesta restituisce informazioni sul metodo di autenticazione, il nome utente e i ruoli assegnati all'utente. Per ulteriori informazioni, vedi [GET /login](#).

#### Procedura

1. Accedi a un utente:

a) Utilizzare il metodo HTTP POST sulla risorsa `login`:

```
https://host:port/ibmmq/rest/v1/login
```

Includere il nome utente e la password nel corpo della richiesta JSON, nel formato seguente:

```
{
  "username" : name,
  "password" : password
}
```

- b) Memorizzare il token LTPA, `LtpaToken2` restituito dalla richiesta nell'archivio cookie locale.
2. Autenticare le richieste REST con il token LTPA memorizzato, `LtpaToken2`, come cookie con ogni richiesta.

Per le richieste che utilizzano i metodi HTTP PUT, PATCH o DELETE, includere un'intestazione `ibm-mq-rest-csrf-token`. Il valore di questa intestazione può essere qualsiasi cosa, incluso uno spazio.

3. Disconnettere un utente:

- a) Utilizzare il metodo HTTP DELETE sulla risorsa `login` :

```
https://host:9443/ibmmq/rest/v1/login
```

È necessario fornire il token LTPA, `LtpaToken2`, come cookie per autenticare la richiesta e includere un'intestazione `ibm-mq-rest-csrf-token`. Il valore di questa intestazione può essere qualsiasi cosa, incluso il campo vuoto

- b) Elaborare l'istruzione per eliminare il token LTPA dall'archivio cookie locale.

**Nota:** Se l'istruzione non viene elaborata e il token LTPA rimane nell'archivio cookie locale, il token LTPA può essere utilizzato per autenticare future richieste REST. In altre parole, quando l'utente tenta di eseguire l'autenticazione con il token LTPA una volta terminata la sessione, viene creata una nuova sessione che utilizza il token esistente.

## Esempio

Il seguente esempio cURL mostra come creare una nuova coda Q1, sul gestore code QM1, con autenticazione basata su token, sui sistemi Windows :

- Accedere e aggiungere il token LTPA, `LtpaToken2`, all'archivio cookie locale. Le informazioni su nome utente e password sono incluse nel corpo JSON. L'indicatore `-c` specifica l'ubicazione del file in cui memorizzare il token:

```
curl -k https://localhost:9443/ibmmq/rest/v1/login -X POST
-H "Content-Type: application/json" --data
"{\"username\": \"mqadmin\", \"password\": \"mqadmin\"}"
-c c:\cookiejar.txt
```

- Creare una coda. Utilizzare il metodo HTTP POST con la risorsa della coda, autenticandosi con il token LTPA. Il token LTPA, `LtpaToken2`, viene richiamato dal file `cookiejar.txt` utilizzando l'indicatore `-b`. La protezione CSRF è fornita dalla presenza dell'intestazione HTTP `ibm-mq-rest-csrf-token` :

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X POST -b
c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json"
--data "{\"name\": \"Q1\"}"
```

- Scollegarsi ed eliminare il token LTPA dall'archivio cookie locale. Il token LTPA, `LtpaToken2`, viene richiamato dal file `cookiejar.txt` utilizzando l'indicatore `-b`. La protezione CSRF è fornita dalla presenza dell'intestazione HTTP `ibm-mq-rest-csrf-token`. L'ubicazione del file `cookiejar.txt` è specificata dall'indicatore `-c` in modo che il token LTPA venga eliminato dal file:


```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X DELETE
-H "ibm-mq-rest-csrf-token: value" -b c:\cookiejar.txt
-c c:\cookiejar.txt
```


## Operazioni successive

Se entrambe le porte HTTP e HTTPS sono abilitate per il server mqweb, considerare la possibilità di applicare l'utilizzo di una connessione HTTPS sicura con il token LTPA. È possibile configurare il server mqweb per richiedere una connessione HTTPS quando viene utilizzato un token LTPA:

1. Come utente privilegiato, aprire il file `mqwebuser.xml`.

Il file `mqwebuser.xml` può essere trovato in una delle seguenti directory:

-  Su UNIX, Linux, and Windows: `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb`

-  Su z/OS: `WLP_user_directory/servers/mqweb`

dove `WLP_user_directory` è la directory specificata quando è stato eseguito lo script `crtmqweb.sh` per creare la definizione del server `mqweb`.

2. Aggiungere la riga seguente al file `mqwebuser.xml`:

```
<webAppSecurity ssoRequiresSSL="true"/>
```

### Informazioni correlate

[PUBBLICA /login](#)

[GET /login](#)

[Elimina /login](#)

## Utilizzo dell'autenticazione basata su token con la API REST per IBM MQ 9.0.4 e versioni precedenti

Gli utenti dell'API REST possono eseguire l'autenticazione fornendo un ID utente e una password alla risorsa `login` dell'API REST con il metodo HTTP POST. Viene generato un token LTPA che consente all'utente di autenticare richieste future. L'utente può scollegarsi utilizzando il metodo HTTP DELETE e può interrogare le informazioni di accesso dell'utente corrente con il metodo HTTP GET.

### Prima di iniziare

- Configurare utenti, gruppi e ruoli da autorizzare all'utilizzo dell'API REST. Per ulteriori informazioni, consultare [“Configurazione di utenti e ruoli”](#) a pagina 466.
- Facoltativamente, configurare la scadenza per il token LTPA. Per ulteriori informazioni, consultare [Configurazione del timer di scollegamento utente](#).
- Assicurarsi di utilizzare una connessione sicura quando si inviano richieste REST. Quando utilizzi il metodo HTTP POST sulla risorsa `login`, la combinazione di nome utente e password inviata con la richiesta non viene codificata. Pertanto, è necessario utilizzare una connessione protetta (HTTPS) quando si utilizza l'autenticazione basata su token con l'API REST.
- È possibile interrogare le credenziali dell'utente corrente utilizzando il metodo HTTP GET sulla risorsa `login`, fornendo il token LTPA, `LtpaToken2`, per autenticare la richiesta. Questa richiesta restituisce informazioni sul metodo di autenticazione, il nome utente e i ruoli assegnati all'utente. Per ulteriori informazioni, vedi [GET /login](#).

### Procedura

1. Accedi a un utente:

a) Utilizzare il metodo HTTP POST sulla risorsa `login`:

```
https://host:port/ibmmq/v1/login
```

Includere il nome utente e la password nel corpo della richiesta JSON, nel formato seguente:

```
{
  "username" : name,
  "password" : password
}
```

b) Memorizzare il token LTPA `LtpaToken2` e il cookie CSRF, `csrfToken`, restituiti dalla richiesta nell'archivio cookie locale.

## 2. Autenticare le richieste REST con i token archiviati:

- Fornire il token LTPA, `LtpaToken2`, come cookie con ogni richiesta.
- Per le richieste che utilizzano i metodi HTTP PUT, PATCH o DELETE, includere il contenuto del token CSRF, `csrfToken`, in un'intestazione `ibm-mq-rest-csrf-token`.

Il contenuto del cookie `csrfToken` viene utilizzato per confermare che le credenziali utilizzate per autenticare la richiesta vengono utilizzate dal proprietario delle credenziali. Ovvero, il token viene utilizzato per evitare attacchi cross - site request forgery.

Non è possibile utilizzare una versione memorizzata nella cache del contenuto del cookie perché il contenuto del cookie può cambiare. È necessario utilizzare l'ultimo valore del cookie per ogni richiesta.

## 3. Disconnettere un utente:

- a) Utilizzare il metodo HTTP DELETE sulla risorsa `login` :

```
https://host:9443/ibmmq/v1/login
```

È necessario specificare il token LTPA, `LtpaToken2`, come cookie per autenticare la richiesta. È necessario includere anche il contenuto del token CSRF, `csrfToken`, in un'intestazione `ibm-mq-rest-csrf-token`.

- b) Elaborare l'istruzione per eliminare il token LTPA dall'archivio cookie locale.

**Nota:** Se l'istruzione non viene elaborata e il token LTPA rimane nell'archivio cookie locale, il token LTPA può essere utilizzato per autenticare future richieste REST. In altre parole, quando l'utente tenta di eseguire l'autenticazione con il token LTPA una volta terminata la sessione, viene creata una nuova sessione che utilizza il token esistente.

## Esempio

Il seguente esempio cURL mostra come creare una nuova coda Q1, sul gestore code QM1, con autenticazione basata su token, sui sistemi Windows :

- Accedere e aggiungere il token LTPA, `LtpaToken2` e CSRF, `csrfToken`, all'archivio cookie locale. Le informazioni su nome utente e password sono incluse nel corpo JSON. L'indicatore `-c` specifica l'ubicazione del file in cui memorizzare il token:

```
curl -k https://localhost:9443/ibmmq/rest/v1/login -X POST
-H "Content-Type: application/json" --data
"{\"username\": \"mqadmin\", \"password\": \"mqadmin\"}"
-c c:\cookiejar.txt
```

- Creare una coda. Utilizzare il metodo HTTP POST con la risorsa della coda, autenticandosi con il token LTPA e includendo il contenuto del token CSRF in un'intestazione. Il token LTPA, `LtpaToken2`, viene richiamato dal file `cookiejar.txt` utilizzando l'indicatore `-b`. Il token CSRF, `csrfToken`, è incluso in un'intestazione HTTP `ibm-mq-rest-csrf-token`. Il valore del token CSRF viene copiato dal file `cookiejar.txt`:

**V 9.0.4** IBM MQ 9.0.4:

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X POST -b c:\cookiejar.txt
-H "ibm-mq-rest-csrf-token:
D82EEBAF1D52F51FE14766474282D3573A432F30D5CD730DB04B1B6187940DF9138B69
381DD68E7B0042ABA0C3D6EFA2F6DBE13E1F67AEFE309E7AA02AD6013FFCB265210281C7949690E215750A1F55FD
BA8B
16B25EAA6F915F7F2299CC2B87EFB9AD4BAAFD28210DDAA9563AC23DDAB259C8992079A7917194E0A6A6ABE1B3DDA5
E2D0
3187FF8CEEC8C707E012D730F2B278ADF6E67A3F4AC1FD5586DEF91C7EC04F5969138D929B7CC118B9EBC74D2733EF9
0E90
3E0A4792A198AF5281F1CFB6E500F72EECDB63B43FED5813708FE1EAC518CA88DFCF687A5AA41BC2BCD3B6C173A605
C6A7
2E7C49F60113B6D171FDCAF7ED85D14FF32761D5BC771796BF" -H "Content-Type: application/json"
--data "{\"name\": \"Q1\"}"
```



IBM MQ 9.0.3 e precedenti:

```
curl -k https://localhost:9443/ibmmq/rest/v1/qmgr/QM1/queue -X POST -b c:\cookiejar.txt
-H "ibm-mq-rest-csrf-token:
D82EEBAF1D52F51FE14766474282D3573A432F30D5CD730DB04B1B6187940DF9138B69
381DD68E7B0042ABA0C3D6EFA2F6DBE13E1F67AEFE309E7AA02AD6013FFCB6265210281C7949690E215750A1F55FD
BA8B
16B25EAA6F915F7F2299CC2B87EFB9AD4BAAF28210DDAA9563AC23DDAB259C8992079A7917194E0A6A6ABE1B3DDA5
E2D0
3187FF8CEE8C707E012D730F2B278ADF6E67A3F4AC1FD5586DEF91C7EC04F5969138D929B7CC118B9EBC74D2733EF9
0E90
3E0A4792A198AF5281F1CFB6E500F72EECDB63B43FED5813708FE1EAC518CA88DFCF687A5AA41BC2BCD3B6C173A605
C6A7
2E7C49F60113B6D171FDCAF7ED85D14FF32761D5BC771796BF" -H "Content-Type: application/json"
--data "{ \"name\": \"Q1\" }"
```

- Scollegarsi ed eliminare il token LTPA dall'archivio cookie locale. Il token LTPA, `LtpaToken2`, viene richiamato dal file `cookiejar.txt` utilizzando l'indicatore `-b`. Il token CSRF, `csrfToken`, è incluso in un'intestazione HTTP `ibm-mq-rest-csrf-token`. Il valore del token CSRF viene copiato dal file `cookiejar.txt`. L'ubicazione del file `cookiejar.txt` è specificata dall'indicatore `-c` in modo che il token LTPA venga eliminato dal file:



```
curl -k https://localhost:9443/ibmmq/rest/v1/login -X DELETE
-H "ibm-mq-rest-csrf-token: D82EEBAF1D52F51FE14766474282D3573A432F30D5C
D730DB04B1B6187940DF9138B69381DD68E7B0042ABA0C3D6EFA2F6DBE13E1F67AEFE3
09E7AA02AD6013FFCB6265210281C7949690E215750A1F55FDBA8B16B25EAA6F915F7F2
299CC2B87EFB9AD4BAAF28210DDAA9563AC23DDAB259C8992079A7917194E0A6A6ABE1
B3DDA5E2D03187FF8CEE8C707E012D730F2B278ADF6E67A3F4AC1FD5586DEF91C7EC04F5
969138D929B7CC118B9EBC74D2733EF90E903E0A4792A198AF5281F1CFB6E500F72EECDB
63B43FED5813708FE1EAC518CA88DFCF687A5AA41BC2BCD3B6C173A605C6A72E7C49F6011
3B6D171FDCAF7ED85D14FF32761D5BC771796BF" -b c:\cookiejar.txt
-c c:\cookiejar.txt
```

## Operazioni successive

Se entrambe le porte HTTP e HTTPS sono abilitate per il server mqweb, considerare la possibilità di applicare l'utilizzo di una connessione HTTPS sicura con il token LTPA. È possibile configurare il server mqweb per richiedere una connessione HTTPS quando viene utilizzato un token LTPA:

1. Come utente privilegiato, aprire il file `mqwebuser.xml`.

Il file `mqwebuser.xml` può essere trovato in una delle seguenti directory:

-  Su UNIX, Linux, and Windows: `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb`
-  Su z/OS: `WLP_user_directory/servers/mqweb`

dove `WLP_user_directory` è la directory specificata quando è stato eseguito lo script `crtmqweb.sh` per creare la definizione del server mqweb.

2. Aggiungere la riga seguente al file `mqwebuser.xml`:

```
<webAppSecurity ssoRequiresSSL="true"/>
```

## Informazioni correlate

[PUBBLICA /login](#)

[GET /login](#)

[Elimina /login](#)

## V 9.0.2 Configurazione di CORS per REST API

Per impostazione predefinita, un browser Web non consente agli script, come JavaScript, di richiamare REST API quando lo script non proviene dalla stessa origine di REST API. In altre parole, le richieste tra origini non sono abilitate. È possibile configurare CORS (Cross Origin Resource Sharing) per consentire richieste di origine incrociata da origini specificate.

## Informazioni su questa attività

È possibile accedere a REST API tramite un browser Web, ad esempio tramite uno script. Poiché queste richieste provengono da un'origine differente rispetto a REST API, il browser Web rifiuta la richiesta perché si tratta di una richiesta di origine incrociata. L'origine è diversa se il dominio, la porta o lo schema non sono uguali.

Ad esempio, se hai uno script ospitato in `http://localhost:1999/`, fai una richiesta di origine incrociata se immetti un HTTP GET su un sito web ospitato in `https://localhost:9443/`. Questa richiesta è di origine incrociata perché i numeri di porta e lo schema (HTTP) sono diversi.

Puoi abilitare le richieste tra origini configurando CORS e specificando le origini a cui è consentito accedere a REST API.

## Procedura

### • V 9.0.4

Utilizzare uno dei seguenti metodi per configurare il nome host:

- Da IBM MQ 9.0.4, utilizzare il comando **setmqweb properties** :

- Visualizzare la configurazione attuale immettendo il seguente comando e visualizzando le voci `mqRestCorsAllowedOrigins` e `mqRestCorsMaxAgeInSeconds` :

```
dspmweb properties -a
```

- Specificare le origini consentite per accedere a REST API immettendo il seguente comando:

```
setmqweb properties -k mqRestCorsAllowedOrigins -v allowedOrigins
```

dove *allowedOrigins* specifica l'origine da cui si desidera consentire le richieste tra origini. È possibile utilizzare un asterisco racchiuso tra virgolette doppie, "\*", per consentire tutte le richieste tra origini. È possibile immettere più di un'origine in un elenco separato da virgole, racchiuso tra virgolette. Per non consentire richieste tra origini, immettere virgolette vuote come valore per *allowedOrigins*.

- Specificare il tempo, in secondi, per cui si desidera consentire a un browser Web di memorizzare nella cache i risultati di qualsiasi controllo pre - volo CORS immettendo il seguente comando:


```
setmqweb properties -k mqRestCorsMaxAgeInSeconds -v time
```

- Per IBM MQ 9.0.3 e versioni precedenti, modificare il file `mqwebuser.xml` :

1. Assicurati di essere un utente privilegiato.

2. Aprire il file `mqwebuser.xml`.

Il file `mqwebuser.xml` può essere trovato in una delle seguenti directory:

-  Su UNIX, Linux, and Windows: `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb`

-  Su z/OS: `WLP_user_directory/servers/mqweb`

dove *WLP\_user\_directory* è la directory specificata quando è stato eseguito lo script **crtmqweb.sh** per creare la definizione del server `mqweb`.

3. Configurare CORS aggiungendo o decommentando le seguenti righe nel file `mqwebuser.xml` :

```
<variable name="mqRestCorsAllowedOrigins" value="https://localhost:9883"/>  
<variable name="mqRestCorsMaxAgeInSeconds" value="120"/>
```

4. Modifica il valore della variabile **mqRestCorsAllowedOrigins** con l'origine da cui vuoi consentire le richieste di origine incrociata. È possibile utilizzare un asterisco, \*, per consentire tutte le richieste tra origini oppure è possibile immettere più di un'origine in un elenco separato da virgole.

5. Modificare il valore della variabile **mqRestCorsMaxAgeInSeconds** con il tempo, in secondi, che si desidera consentire a un browser Web di memorizzare nella cache i risultati di qualsiasi controllo pre - volo CORS.

### Esempio

**V 9.0.4** Il seguente esempio mostra le richieste tra origini abilitate per `http://localhost:9883`, `https://localhost:1999` e `https://localhost:9663`. La durata massima dei risultati memorizzati nella cache di qualsiasi controllo pre - volo CORS è impostata su 90 secondi:

```
setmqweb -k mqRestCorsAllowedOrigins -v "http://localhost:9883,https://localhost:1999,https://localhost:9663"
setmqweb -k mqRestCorsMaxAgeInSeconds -v 90
```

## ULW **V 9.0.1** Audit

Su UNIX, Linux, and Windows, IBM MQ Console registra le modifiche di stato significative come messaggi nei log del server mqweb. Ogni messaggio indica il nome principal autenticato che ha richiesto l'operazione.

Le modifiche di stato significative, ad esempio quando i gestori code vengono creati, avviati, terminati o eliminati, vengono registrate nei file `messages.log` e `console.log` del server mqweb al livello di registrazione [ AUDIT]. Ogni voce di log indica il nome principal autenticato che ha richiesto l'operazione.

I file `messages.log` e `console.log` si trovano nella seguente ubicazione:

- **ULW** Su UNIX, Linux, and Windows:  
`MQ_DATA_DIRECTORY\web\installations\installationName\servers\mqweb\logs`

**V 9.0.1** Per ulteriori informazioni sulla configurazione dei livelli di registrazione, consultare [Configurazione della registrazione](#).

Facoltativamente, è possibile abilitare gli eventi di comando e configurazione per fornire informazioni sulla maggior parte delle attività IBM MQ Console . Ad esempio, la creazione di canali e l'interrogazione delle code generano eventi di comando e configurazione. Per ulteriori informazioni sull'abilitazione degli eventi di comando e configurazione, consultare [Controllo degli eventi di configurazione, comando e programma di registrazione](#). Per questi messaggi di comandi e di eventi di configurazione, il campo `MQIACF_EVENT_ORIGIN` è impostato su `MQEVO_REST` e il campo `MQCACF_EVENT_APPL_IDENTITY` riporta i primi 32 caratteri del nome principal autenticato. Se un utente ha il ruolo **MQWebAdmin** o **MQWebAdminRO** , il campo `MQCACF_EVENT_USER_ID` riporta il nome utente del principal che ha avviato il server Web, non il nome utente del principal che ha emesso il comando. Tuttavia, se l'utente dispone del ruolo **MQWebUser** , `MQCACF_EVENT_USER_ID` riporta il nome utente del principal che ha emesso il comando.

### Concetti correlati

[“Audit” a pagina 431](#)

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 MQ Explorer.

## Considerazioni sulla protezione per IBM MQ Console e REST API su z/OS

Su z/OS, sono disponibili ulteriori opzioni per configurare la sicurezza per IBM MQ Console e REST API. È possibile configurare un registro LDAP. Puoi configurare TLS per IBM MQ Console e REST API per abilitare un utente ad accedere con un certificato. È possibile configurare l'interfaccia SAF (System Authorization Facility) in modo che un utente possa accedere con ID utente e password z/OS .

### Prima di iniziare

IBM MQ Console e REST API dispongono di funzioni di sicurezza che controllano se un utente può immettere, visualizzare o modificare i comandi. I comandi vengono quindi passati al gestore code e la

sicurezza del gestore code viene quindi utilizzata per controllare se all'utente è consentito immettere il comando per quel gestore code specifico.

È necessario essere consapevoli delle seguenti considerazioni sulla sicurezza:

- Se il gestore code è stato configurato per richiedere che tutte le applicazioni batch forniscano un ID utente e una parola d'ordine validi, impostando CHKLOCL (REQUIRED), è necessario fornire all'ID utente dello spazio di indirizzo *UPDATE* l'accesso al profilo h1q.BATCH nella classe MQCONN.

Ciò fa in modo che l'autenticazione della connessione operi in modalità CHKLOCL (OPTIONAL) per l'ID utente dello spazio di indirizzi del server mqweb.

Se il gestore code non è stato configurato per richiedere che tutte le applicazioni batch forniscano un ID utente e una password validi, è sufficiente fornire all'ID utente dello spazio di indirizzo del server mqweb *READ* l'accesso al profilo h1q.BATCH nella classe MQCONN.

Per ulteriori informazioni su CHCKLOCL, consultare [“Utilizzo di CHCKLOCL su applicazioni associate localmente”](#) a pagina 189.

- L'ID utente dello spazio di indirizzo del server mqweb necessita dell'autorizzazione per emettere determinati comandi PCF e per accedere a determinate code

Per ulteriori informazioni, consultare:

- [“IBM MQ Console - profili di sicurezza del comando richiesti”](#) a pagina 222
- [“Sicurezza della coda di sistema”](#) a pagina 198
- [“Profili per la sicurezza del contesto”](#) a pagina 209

- Gli utenti di IBM MQ Console e REST API assegnati al ruolo MQWebUser operano nel contesto di sicurezza del principal.

Questi ID utente possono eseguire solo le operazioni che l'ID utente può eseguire sul gestore code e devono avere accesso alle stesse code di sistema dello spazio di indirizzo del server mqweb.

L'ID utente dello spazio di indirizzi del server mqweb deve avere accesso alternativo a tutti gli utenti assegnati al ruolo MQWebUser. Per ulteriori informazioni sulla sicurezza dell'utente alternativo, consultare [“Profili per la sicurezza utente alternativa”](#) a pagina 207

## Procedura

- [Autorizzazione richiesta dall'ID utente dell'attività avviata del server mqweb](#)
- [Accesso alle risorse IBM MQ, richieste per utilizzare la console MQ o l'API REST](#)
- [Configurazione di TLS per l'API REST e la console IBM MQ su z/OS](#)
- [Configurazione dell'interfaccia di System Authorization Facility](#)

## **Autorizzazione richiesta dall'ID utente dell'attività avviata del server mqweb**

Su z/OS, l'ID utente dell'attività avviata dal server mqweb richiede alcune autorizzazioni per emettere comandi PCF e accedere alle risorse del sistema.

L'ID utente dell'attività avviata del server mqweb richiede:

- Un UID (z/OS UNIX user identifier) per poter utilizzare z/OS UNIX System Services.
- Accedere ai dataset h1q.SCSQAUTH e h1q.SCSQANL\* nell'installazione di IBM MQ.
- Accesso in lettura ai file di installazione IBM MQ in z/OS UNIX System Services.
- Accesso in lettura e scrittura alla directory utente Liberty creata dallo script **crtmqweb**.
- Autorizzazione per connettersi al gestore code. Concedere all'ID utente dell'attività avviata del server mqweb l'accesso *READ* al profilo h1q.BATCH nella classe MQCONN.

- Autorizzazione per immettere comandi IBM MQ e accedere a determinate code. Questi dettagli sono descritti in [“IBM MQ Console - profili di sicurezza del comando richiesti”](#) a pagina 222, [“Sicurezza della coda di sistema”](#) a pagina 198e [“Profili per la sicurezza del contesto”](#) a pagina 209.
- Autorizzazione alla sottoscrizione all'argomento SYSTEM.FTE , per utilizzare REST API per MFT. Concedere l'accesso *ALTER* dell'ID utente dell'attività avviata del server mqweb al profilo h1q.SUBSCRIBE.SYSTEM.FTE nella classe MXTOPIC.
- Se si sta configurando un registro SAF, accedere a vari profili di protezione. Per ulteriori informazioni, fare riferimento a [“Configurazione dell'interfaccia di System Authorization Facility”](#) a pagina 488.

## Autenticazione connessione

Se il gestore code è stato configurato per richiedere che tutte le applicazioni batch forniscano un ID utente e una password validi, impostando CHKLOCL (REQUIRED), è necessario fornire all'ID utente dell'attività avviata del server mqweb l'accesso *UPDATE* al profilo h1q.BATCH nella classe MQCONN.

Questa autorizzazione fa in modo che l'autenticazione della connessione operi in modalità CHKLOCL (OPTIONAL) per l'ID utente del task avviato del server mqweb.

Se il gestore code non è stato configurato in modo da richiedere che tutte le applicazioni batch forniscano un ID utente e una password validi, è sufficiente fornire all'ID utente che avvia l'attività del server mqweb *READ* l'accesso al profilo h1q.BATCH nella classe MQCONN.

Per ulteriori informazioni su CHCKLOCL, consultare [“Utilizzo di CHCKLOCL su applicazioni associate localmente”](#) a pagina 189.

## Accesso alle risorse IBM MQ , richieste per utilizzare MQ Console o REST API

Le operazioni eseguite in MQ Console, o REST API, da un utente nel ruolo MQWebUser si verificano nel contesto di sicurezza dell'utente.

### Informazioni su questa attività

Consultare [“Ruoli su IBM MQ Console e su REST API”](#) a pagina 469 per ulteriori informazioni sui ruoli in MQ Console e REST API.

Utilizzare la seguente procedura per concedere a un utente, nel ruolo MQWebUser , l'accesso alle risorse del gestore code richieste per utilizzare MQ Console o REST API.

### Procedura

1. Concedere l'ID utente dell'attività avviata mqweb, l'accesso utente alternativo a ogni ID utente nel ruolo MQWebUser .

Eseguire questa operazione su ogni gestore code che gli utenti gestiranno tramite MQ Console o REST API.

È possibile utilizzare i seguenti comandi RACF di esempio per concedere, l'ID utente dell'attività avviata da mqweb, l'accesso utente alternativo a un utente nel ruolo MQWebUser :

```
RDEFINE MQADMIN h1q.ALTERNATE.USER.userId UACC(NONE)
PERMIT h1q.ALTERNATE.USER.userId CLASS(MQADMIN) ACCESS(UPDATE) ID(mqwebUserId)
SETROPTS RACLIST(MQADMIN) REFRESH
```

dove:

#### **h1q**

È il prefisso del profilo, che può essere il nome del gestore code o il nome del gruppo di condivisione code

#### **userId**

L'utente è nel ruolo MQWebUser

## mqwebUserId

È l'ID utente dell'attività avviata mqweb

**Nota:** Se si utilizza la sicurezza con caratteri maiuscoli e minuscoli, utilizzare la classe MXADMIN piuttosto che la classe MQADMIN.

2. Concedere a ogni utente nel ruolo MQWebUser , l'accesso alle code di sistema necessarie per utilizzare MQ Console e REST API.

A tale scopo, per entrambi i sistemi SYSTEM.ADMIN.COMMAND.QUEUE e SYSTEM.REST.REPLY.QUEUE, fornisce a ciascun utente l'accesso UPDATE alle classi MQQUEUE o MXQUEUE, a seconda che la sicurezza sia in uso o meno.

È necessario eseguire questa operazione su ogni gestore code che l'utente gestirà tramite il REST API, incluso da IBM MQ 9.0.4, o sui gestori code remoti gestiti tramite il [gateway amministrative REST API](#).

3. Da IBM MQ 9.0.4, è possibile utilizzare REST API per gestire i gestori code remoti.

Per consentire a un utente con il ruolo MQWebUser di amministrare i gestori code remoti, concedere all'utente l'accesso UPDATE al profilo nella classe MQQUEUE o MXQUEUE, proteggendo la coda di trasmissione utilizzata per inviare i comandi al gestore code remoto. Tenere presente che è necessario fornire all'utente l'accesso UPDATE sul gestore code gateway.

Sul gestore code remoto, concedere l'accesso allo stesso utente per inserire nella coda di trasmissione utilizzata per inviare i messaggi di risposta del comando al gestore code gateway.

4. Concedere agli utenti in MQWebUserRole l'accesso a tutte le altre risorse richieste per eseguire le operazioni supportate da MQ Console e REST API.

L'accesso necessario per:

- Eseguire operazioni in REST API, è descritto nelle sezioni *Requisiti di protezione* delle singole risorse REST API
- L'emissione di comandi da parte di MQ Console è descritta in [“IBM MQ Console - profili di sicurezza del comando richiesti”](#) a pagina 222

V 9.0.2

## Configurazione di TLS per REST API e IBM MQ Console su z/OS

Un metodo di configurazione TLS per IBM MQ Console e REST API su z/OS.

### Prima di iniziare

Assicurati di avere un IBM MQ Console e un REST API di lavoro come descritto in [Introduzione](#).

### Informazioni su questa attività

Per configurare l'utilizzo dell'interfaccia TLS, è necessario specificare le istruzioni in un file XML.

È possibile aggiungerle al file mqwebuser.xml o creare un file separato, ad esempio ssl.xml e aggiungere un'istruzione `<include location="ssl.xml"/>` nella parte inferiore del file mqwebuser.xml.

### Procedura

1. In mqwebuser.xml, commentare le definizioni esistenti per:
  - mqDefaultSSLConfig
  - defaultKeyStore
2. In mqwebuser.xml, aggiungere le seguenti istruzioni di codice:

```
<?xml version="1.0" encoding="UTF-8"?>
<server>
  <featureManager>
    <feature>ssl-1.0</feature>
  </featureManager>
  <sslDefault sslRef="mqDefaultSSLConfig"/>
</server>
```

```

<ssl id="mqDefaultSSLConfig" keyStoreRef="defaultKeyStore"
    sslProtocol="TLSv1.2"
    serverKeyAlias="def2"
    clientAuthentication="true"
  />

<keyStore id="defaultKeyStore"
    filebased="false"
    location="safkeyring://userid/keyring"
    password="password"
    readOnly="true"
    type="JCERACFKS"
  />
</server>

```

**Note:**

- a. Il testo in grassetto è richiesto per definire l'interfaccia TLS.
- b. Il valore di `sslRef="mqDefaultSSLConfig"` in **sslDefault** deve corrispondere a uno dei valori `<ssl id = ..... >`.
- c. Il valore di `<ssl keyStoreRef="defaultKeyStore">` in **<ssl>** deve corrispondere a `id = value` in un **<keystore>**.
- d. Specificare l'ID utente e il keyring dell'ID utente da utilizzare nell'istruzione `location="safkeyring://userid/keyring"`.
- e. Il valore di **serverKeyAlias** (ad esempio, `serverKeyAlias="def2"`) è il nome del certificato nel keyring che deve essere utilizzato da IBM MQ Console.
- f. Il valore di **keystore password** viene ignorato.

Consultare [Liberty: Keystore](#) per informazioni sul keyring RACF.

**Definizioni RACF**

Con un valore di:

- a. `location="safkeyring://SCENSTC/MYKEYRING"`
- b. `serverKeyAlias="def2"`

l'output del comando `RACF RACDCERT LISTRING(MYKEYRING) ID(SCENSTC)` è:

```

Digital ring information for user SCENSTC:

Ring:
  >MYKEYRING<
Certificate Label Name          Cert Owner      USAGE          DEFAULT
-----
SCENCA                          CERTAUTH       CERTAUTH       NO
def2                             ID(SCENSTC)    PERSONAL       NO

```

**Nota:** Se si utilizzano certificati autofirmati, questi devono essere connessi al keyring.

3. Riavviare il server mqweb.

Non devono essere presenti messaggi in `// STDERR`

Ci dovrebbero essere messaggi in `// STDOUT` simili a quelli elencati in [Introduzione alla Console di IBM MQ](#) ..

**Note:**

- a. Se si stanno utilizzando solo certificati per l'autenticazione con IBM MQ Console, il browser potrebbe visualizzare un elenco di certificati da cui effettuare la selezione.
- b. Se si desidera utilizzare un certificato diverso, è necessario chiudere e riavviare il browser.
- c. Se si utilizzano certificati che non si trovano nel database RACF, è possibile utilizzare il filtro del nome del certificato RACF, per associare gli attributi nel certificato a un ID utente, ad esempio:

```
RACDCERT ID(DEPT3USR) MAP SDNFILTER(OU=DEPT1.C=US)
```

associa i certificati con OU=DEPT1 e C=US all'ID utente DEPT3USR

## Risultati

Hai configurato un'interfaccia TLS per IBM MQ Console e REST API.

### V 9.0.2 Configurazione dell'interfaccia di System Authorization Facility

L'interfaccia SAF (System Authorization Facility) consente al server mqweb di richiamare il gestore sicurezza esterno per l'autenticazione e il controllo delle autorizzazioni.

## Prima di iniziare

Accertarsi di avere:

- Un lavoro IBM MQ Console e REST API come descritto in [Introduzione](#)
- Il processo Angel WebSphere Application Server Liberty in esecuzione per utilizzare l'interfaccia autorizzata per SAF. Per ulteriori informazioni, consultare [Abilitazione dei servizi autorizzati z/OS su Liberty for z/OS](#).

## Informazioni su questa attività

L'interfaccia SAF consente al server mqweb di richiamare il gestore della sicurezza esterno per l'autenticazione e il controllo dell'autorizzazione sia per IBM MQ Console che per REST API.

Notare che è necessario essere un [utente privilegiato](#) per eseguire alcune delle fasi di questa attività.

## Procedura

1. Segui la procedura in [Abilitazione dei servizi autorizzati z/OS su Liberty for z/OS](#) per fornire al tuo server mqweb Liberty l'accesso per utilizzare i servizi autorizzati z/OS.

Il JCL di esempio per l'avvio del processo angel si trova in `USS_ROOT/web/templates/zos/procs/bbgzang1.jcl`, dove `USS_ROOT` è il percorso in Unix System Services in cui sono installati i componenti USS IBM MQ for z/OS.

In `bbgzang1.jcl`, modificare l'istruzione `SET ROOT` in modo che punti a `USS_ROOT/web`, ad esempio, `/usr/lpp/mqm/V9R0M0/web`.

Consultare [Amministrazione di Liberty su z/OS](#) per ulteriori informazioni sull'arresto e l'avvio del processo angel.

2. Attieniti alla procedura in [Liberty: Configurazione dell'utente non autenticato SAF \(System Authorization Facility\)](#) per creare l'utente non autenticato richiesto da Liberty.
3. Utilizzare il file `zos_saf_registry.xml`.

### V 9.0.5

Da IBM MQ 9.0.5, copiare il file `zos_saf_registry.xml` dal seguente percorso:

PathPrefix `/web/mq/samp/configuration`, dove PathPrefix è il percorso di installazione di IBM MQ Unix System Services Components.

Per IBM MQ 9.0.4 e versioni precedenti, utilizzare questo file:

```
<?xml version="1.0" encoding="UTF-8"?>
<server>
  <!-- ***** -->
  <!-- IBM MQ security configuration for MQ Console and REST API. -->
  <!-- Name: zos_saf_registry.xml -->
  <!-- Description: SAF based registry for z/OS -->
  <!-- ***** -->
  <!-- <copyright -->
  <!-- notice='lm-source-program' -->
  <!-- pids='5724-H72' -->
```



```

<!-- years='2017' -->
<!-- crc='0' > -->
<!-- -->
<!-- Licensed Materials - Property of IBM -->
<!-- -->
<!-- 5724-H72 -->
<!-- -->
<!-- (C) Copyright IBM Corp. 2017, 2023. All Rights Reserved. -->
<!-- -->
<!-- US Government Users Restricted Rights - Use, duplication or -->
<!-- disclosure restricted by GSA ADP Schedule Contract with -->
<!-- IBM Corp. -->
<!-- </copyright> -->

<!--
Role mappings are granted by giving users and groups READ access to the
following profiles in the EJBROLE class:

1) MQWEB.com.ibm.mq.console.MQWebAdmin

MQWebAdmin role access for the MQ Console. All MQ commands issued by the
MQ Console use the security context of the operating system user running
the application server.

2) MQWEB.com.ibm.mq.console.MQWebAdminRO

MQWebAdminRO role access for the MQ Console. The security context of
the operating system user running the application server is used for
all read-only MQ commands, such as DISPLAY CHANNEL, QUEUE, etc,
issued by the MQ Console.

3) MQWEB.com.ibm.mq.console.MQWebUser

MQWebUser role access for the MQ Console. All MQ commands issued by
the MQ Console use the security context of the principal and so the
user must be known to the queue manager and authorized to issue the
command.

4) MQWEB.com.ibm.mq.rest.MQWebAdmin

MQWebAdmin role access for the MQ REST API. All MQ commands issued by the
REST API use the security context of the operating system user running
the application server.

5) MQWEB.com.ibm.mq.rest.MQWebAdminRO

MQWebAdminRO role access for the MQ REST API. The security context of
the operating system user running the application server is used for
all read-only MQ commands, such as DISPLAY CHANNEL, QUEUE, etc,
issued by the REST API.

6) MQWEB.com.ibm.mq.rest.MQWebUser

MQWebUser role access for the MQ REST API. All MQ commands issued by
the REST API use the security context of the principal and so the
user must be known to the queue manager and authorized to issue the
command.

In addition the sample enables HTTP Basic Authentication.
-->

<!--
Enable features
-->
<featureManager>
  <feature>appSecurity-2.0</feature>
  <feature>zosSecurity-1.0</feature>
  <feature>basicAuthenticationMQ-1.0</feature>
</featureManager>

<!--
The MQ Console
-->
<enterpriseApplication id="com.ibm.mq.console"/>

<!--
The MQ REST API
-->
<enterpriseApplication id="com.ibm.mq.rest"/>

<!--
Example SAF Registry
-->
<safAuthorization racRouteLog="ASIS"/>
<safRegistry id="saf"/>
<safAuthorization id="saf"/>
<safCredentials unauthenticatedUser="WSGUEST" profilePrefix="MQWEB"/>

<!--
Enable HTTP by uncommenting the line below.
-->
<!--
<variable name="httpPort" value="9080"/>

```

```

-->
<!--
By default the server listens for HTTP/HTTPS requests on localhost only. To
listen on all available network interfaces uncomment the line below. To listen
on a specific IP address or hostname replace the * with an appropriate value.
-->
<!--
<variable name="httpHost" value="*"/>
-->

<!--
Default MQ SSL configuration allows TLS v1.2 ONLY, refer to the
IBM Documentation section on "IBM MQ Console and REST API security"
for details of how to configure security.
-->
<sslDefault sslRef="mqDefaultSSLConfig"/>

<!--
Enable client certificate authentication by uncommenting the
block below and creating a trust.jks store. Basic registry
maps the common name (CN=) issued by a trusted CA to
users names in the registry. For example a certificate with
a distinguished name of 'CN=mqadmin,O=IBM,C=GB' will be granted
a MQWebAdmin role under the 'mqadmin' user.

The default, auto-generated certificate held in key.jks is
intended for developer convenience only, it is not intended for
production use.

Passwords for both defaultKeyStore and defaultTrustStore should
be changed and encoded using the securityUtility tool, refer
to the following IBM Developer article for further information;
https://developer.ibm.com/wasdev/docs/configuring-ssl-liberty/
-->
<!--
<keyStore id="defaultKeyStore" location="key.jks" type="JKS" password="password"/>
<keyStore id="defaultTrustStore" location="trust.jks" type="JKS" password="password"/>
<ssl id="thisSSLConfig" clientAuthenticationSupported="true" keyStoreRef="defaultKeyStore"
trustStoreRef="defaultTrustStore" sslProtocol="TLSv1.2" serverKeyAlias="default"/>
<sslDefault sslRef="thisSSLConfig"/>
-->

<!--
Uncomment the following two variables, and adjust them, to change
the default CORS settings.
-->
<!--
<variable name="mqRestCorsAllowedOrigins" value="https://localhost:9883"/>
<variable name="mqRestCorsMaxAgeInSeconds" value="120"/>
-->
</server>

```

4. Inserire il file di esempio nella directory *WLP\_user\_directory/servers/mqweb*, dove *directory\_utente\_WLP* è la directory specificata quando lo script **crtmqweb.sh** è stato eseguito per creare la definizione del server mqweb.
5. Opzionale: Se in precedenza sono state modificate le impostazioni di configurazione in *mqwebuser.xml*, copiarle nel file di esempio.
6. Eliminare il file *mqwebuser.xml* esistente e ridenominarlo come *mqwebuser.xml*.
7. Personalizzare l'elemento **safCredentials** in *mqwebuser.xml*.
  - a. Impostare **profilePrefix** su un nome univoco per il server Liberty. Se si dispone di più server mqweb in esecuzione su un singolo sistema, sarà necessario scegliere un nome diverso per ogni server; ad esempio MQWEB903 e MQWEB904.
  - b. Impostare **unauthenticatedUser** sul nome dell'utente non autenticato creato nel passo [“2”](#) a [pagina 488](#).
8. Definire l'APPLID del server mqweb in RACF.

Il nome della risorsa APPLID è il valore specificato nell'attributo **profilePrefix** nel passo [“7”](#) a [pagina 490](#). Il seguente esempio definisce l'APPLID del server mqweb in RACF:

```
RDEFINE APPL profilePrefix UACC(NONE)
```
9. Concedere a tutti gli utenti o gruppi di essere autenticati per l'accesso READ MQ Console o REST API all'APPLID del server mqweb nella classe APPL.

È necessario eseguire questa operazione anche per l'utente non autenticato definito nel passo [“2” a pagina 488](#). L'esempio riportato di seguito concede a un utente l'accesso READ all'APPLID del server mqweb in RACF:

```
PERMIT profilePrefix CLASS(APPL) ACCESS(READ) ID(userID)
```

10. Definire i profili nella classe EJBROLE necessari per fornire agli utenti l'accesso ai ruoli in MQ Console e REST API.

Il seguente esempio definisce i profili in RACF, dove **profilePrefix** è il valore specificato per l'attributo **profilePrefix** nel passo [“7” a pagina 490](#).

```
RDEFINE EJBROLE profilePrefix.com.ibm.mq.console.MQWebAdmin UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.console.MQWebAdminRO UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.console.MQWebUser UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MQWebAdmin UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MQWebAdminRO UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MQWebUser UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MFTWebAdmin UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MFTWebAdminRO UACC(NONE)
```

Per IBM MQ 9.0.4 e versioni precedenti:

```
RDEFINE EJBROLE profilePrefix.com.ibm.mq.console.MQWebAdmin UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.console.MQWebAdminRO UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.console.MQWebUser UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MQWebAdmin UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MQWebAdminRO UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MQWebUser UACC(NONE)
```

11. Concedere agli utenti l'accesso ai ruoli in MQ Console e REST API.

Per eseguire questa operazione, fornire agli utenti o ai gruppi l'accesso READ a uno o più profili nella classe EBJROLE creati nel passo [“10” a pagina 491](#).

Il seguente esempio fornisce a un utente l'accesso al ruolo MQWebAdmin per il REST API in RACF, dove **profilePrefix** è il valore specificato per l'attributo **profilePrefix** nel passo [“7” a pagina 490](#).

```
PERMIT profilePrefix.com.ibm.mq.rest.MQWebAdmin CLASS(EJBROLE) ACCESS(READ) ID(userID)
```

## Risultati

È stata impostata l'autenticazione SAF per IBM MQ Console e REST API.

## **V 9.0.5** Configurazione della sicurezza MFTREST API

La configurazione per la sicurezza basata sul ruolo Managed File Transfer REST API viene eseguita tramite `mqwebuser.xml`, come avviene per IBM MQ REST API.

## Prima di iniziare

Consultare [“Sicurezza IBM MQ Console e REST API” a pagina 464](#) per informazioni su come impostare la sicurezza.

## Informazioni su questa attività

Per gestire gli aspetti delle risorse MFT per l'interfaccia MFT REST API, i nuovi ruoli di MFTWebAdmin e MFTWebAdminRO sono stati aggiunti ai ruoli specifici IBM MQ esistenti di MQWebAdmin, MQWebAdminRO e MQWebUser.

### MFTWebAdmin

Un utente o un gruppo a cui è assegnato questo ruolo può eseguire tutte le operazioni REST MFT e opera nel contesto di sicurezza dell'ID utente del sistema operativo utilizzato per avviare il server mqweb.

## MFTWebAdminRO

Questo ruolo fornisce accesso in sola lettura a MFT REST API. Un utente o un gruppo a cui è assegnato questo ruolo può eseguire operazioni di sola lettura (richieste GET) come trasferimento elenco e agent elenco.

Un utente o un gruppo a cui è assegnato questo ruolo opera nel contesto di sicurezza dell'ID utente del sistema operativo utilizzato per avviare il server mqweb .

### Importante:

1. Un principal con il ruolo RO MFTWebAdmin o MFTWebAdminnon ha accesso ai servizi IBM MQ REST API .
2. Se un utente MFTWebAdmin ha bisogno di accedere a IBM MQ REST API, tale utente deve essere membro di uno dei gruppi MQWebAdmin, MQWebAdminRO o MQWebUser .
3. I membri dei gruppi MQWebAdmin, MQWebAdminRO e MQWebUser non hanno accesso a MFT REST API.

Il seguente esempio concede:

- MQWebAdmin, MQWebAdminRO e i ruoli MQWebUser , l'accesso descritto in [“Ruoli su IBM MQ Console e su REST API”](#) a pagina 469.
- Accesso del ruolo MFTWebAdmin all'utente "mftadmin". L'utente "mftadmin" può eseguire tutti i servizi REST MFT .
- MFTWebAdminAccesso del ruolo RO all'utente "mftreader". L'utente "mftreader" può eseguire operazioni di sola lettura, come l'agent di elenco e il trasferimento di elenco.

```
<?xml version="1.0" encoding="UTF-8"?>
<server>
  <!-- Enable features -->
  <featureManager>
    <feature>appSecurity-2.0</feature>
  </featureManager>

  <!-- Role Mappings -->
  <enterpriseApplication id="com.ibm.mq.rest">
    <application-bnd>
      <security-role name="MQWebAdmin">
        <group name="MQWebUI" realm="defaultRealm"/>
      </security-role>
      <security-role name="MQWebAdminRO">
        <user name="reader" realm="defaultRealm"/>
      </security-role>
      <security-role name="MQWebUser">
        <special-subject type="ALL_AUTHENTICATED_USERS"/>
      </security-role>
      <security-role name="MFTWebAdmin">
        <user name="mftadmin" realm="defaultRealm"/>
      </security-role>
      <security-role name="MFTWebAdminRO">
        <user name="mftreader" realm="defaultRealm"/>
      </security-role>
    </application-bnd>
  </enterpriseApplication>
```

## Gestione di chiavi e certificati su UNIX, Linux, and Windows

Utilizzare il comando `runmqckm` (UNIX e Windows) e il comando `runmqakm` (UNIX, Linux, and Windows) per gestire le chiavi, i certificati e le richieste di certificati.

### Comando `runmqckm`

Il comando **`runmqckm`** è disponibile su UNIX e Windows.

Il comando **`runmqckm`** fornisce funzioni simili a quelle di iKeyman, descritte in [Sicurezza](#).

Per utilizzare il comando **runmqckm**, assicurarsi che le variabili di ambiente di sistema siano correttamente configurate eseguendo il comando **setmqenv**.

**V 9.0.2** Il comando **runmqckm** richiede l'installazione del componente JRE IBM MQ. Se questo componente non è installato, è possibile utilizzare il comando **runmqackm**.

### Comando runmqackm

Il comando **runmqackm** è disponibile su UNIX, Linux e Windows.

Per utilizzare il comando **runmqackm**, assicurarsi che le variabili di ambiente di sistema siano correttamente configurate eseguendo il comando **setmqenv**.

Se devi gestire i certificati TLS in modo conforme a FIPS, utilizza il comando **runmqackm** invece dei comandi **runmqckm**. Ciò è dovuto al fatto che il comando **runmqackm** supporta una codifica più rigida.

Utilizzare i comandi **runmqckm** e **runmqackm** per effettuare le seguenti operazioni:

- Creare il tipo di file database delle chiavi CMS richiesto da IBM MQ
- Crea richieste di certificato
- Importa certificati personali
- Importa certificati CA
- Gestisci certificati autofirmati

### Informazioni correlate

[keytool](#)

## ULW comandi runmqckm e runmqackm

Questa sezione descrive i comandi **runmqckm** e **runmqackm** in base all'oggetto del comando.

Le principali differenze tra i due comandi sono le seguenti:

- **ULW runmqackm**
  - È disponibile su UNIX, Linux e Windows.
  - Supporta la creazione di certificati e richieste di certificato con chiavi pubbliche Elliptic Curve mentre il comando **runmqckm** non lo fa.
  - Supporta una crittografia più rigida del file del repository delle chiavi rispetto al comando **runmqckm** tramite il parametro **-strong**.
  - È stato certificato come conforme a FIPS 140-2 e può essere configurato per funzionare in modo conforme a FIPS, utilizzando il parametro **-fips**, a differenza del comando **runmqckm**.
- **Windows UNIX runmqckm**
  - È disponibile su UNIX e Windows.
  - Supporta i formati file del repository di chiavi JKS e JCEKS, mentre il comando **runmqackm** non lo supporta.



**Attenzione:** **V 9.0.2** Il comando **runmqckm** richiede l'installazione della funzione JRE (IBM MQ Java runtime environment).

Ogni comando specifica almeno un *oggetto*. I comandi per le operazioni di periferica PKCS #11 potrebbero specificare oggetti aggiuntivi. I comandi per gli oggetti di richiesta certificato, certificato e database delle chiavi specificano anche un'azione. L'oggetto può essere uno dei seguenti:

#### -keydb

Le azioni si applicano a un database di chiavi

#### -cert

Le azioni si applicano a un certificato

### **-certreq**

Le azioni si applicano a una richiesta di certificato

### **-help**

visualizza la guida

### **-versione**

Visualizza le informazioni sulla versione

I seguenti argomenti secondari descrivono le azioni che è possibile eseguire su oggetti di richiesta di certificati, certificati e database delle chiavi; consultare [“opzioni runmqckm e runmqakm”](#) a pagina 503 per una descrizione delle opzioni per questi comandi.

## **Comandi solo per un database di chiavi CMS**

È possibile utilizzare i comandi **runmqckm** e **runmqakm** per gestire le chiavi e i certificati per un database di chiavi CMS.

### **-keydb -modificapw**

Modificare la parola d'ordine per un database di chiavi CMS:

```
-keydb -changepw -db filename -pw password -new_pw new_password  
  
-stash
```

### **-keydb -crea**

Creare un database di chiavi CMS:

```
-keydb -create -db filename  
-pw password -type cms -expire days -stash
```

### **-keydb -stashpw**

Eseguire lo stash della password di un database di chiavi CMS in un file:

```
-keydb -stashpw -db filename  
-pw password
```

### **-cert -getdefault**

**Nota:** Il certificato predefinito non è supportato da IBM MQ 8.0. È necessario utilizzare la configurazione dell'etichetta del certificato come descritto in [Etichette del certificato digitale, che comprende i requisiti](#).

Ottenere il certificato personale predefinito:

```
-cert -getdefault -db filename  
-pw password
```

### **-cert - modifica**

Modificare un certificato.

**Nota:** Attualmente, l'unico campo che può essere modificato è il campo Certificato attendibile.

```
-cert -modify -db filename  
-pw password -label label  
-trust enable|disable
```

## **-cert -setdefault**

**Nota:** Il certificato predefinito non è supportato da IBM MQ 8.0 o versioni successive. È necessario utilizzare la configurazione dell'etichetta del certificato come descritto in [Etichette del certificato digitale, che comprende i requisiti](#).

Impostare il certificato personale predefinito:

```
-cert -setdefault -db filename  
-pw password -label label
```

## **ULW**

## **Comando per database di chiavi CMS o PKCS #12**

È possibile utilizzare i comandi `runmqckm` e `runmqakm` per gestire chiavi e certificati per un database di chiavi CMS o PKCS #12 .

**Nota:** IBM 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 .

## **-keydb -modificapw**

Modificare la password per un database di chiavi:

```
-keydb -changepw -db filename -pw password -new_pw  
new_password -expire days
```

## **-keydb -convertire**

convertire il database delle chiavi da un formato all'altro:

```
-keydb -convert -db filename -pw password  
-old_format cms | pkcs12 -new_format cms
```

## **-keydb -crea**

Creare un database delle chiavi:

```
-keydb -create -db filename -pw password -type cms  
| pkcs12
```

## **-keydb -elimina**

Eliminare un database delle chiavi:

```
-keydb -delete -db filename -pw password
```

## **-keydb -elenco**

Elencare i tipi di database delle chiavi attualmente supportati:

```
-keydb -list
```

## **-cert -aggiungi**

Aggiungere un certificato da un file in un database di chiavi:

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

## **-cert -crea**

Creare un certificato autofirmato:

```

-cert -create -db filename -pw password -label label
-dn distinguished_name
-size 1024 | 512 -x509version 3 | 1
| 2
-expire days -sig_alg MD2_WITH_RSA | MD2WithRSA
| MD5_WITH_RSA | MD5WithRSA
| SHA1WithDSA | SHA1WithRSA
| SHA256_WITH_RSA | SHA256WithRSA
| SHA2WithRSA | SHA384_WITH_RSA
| SHA384WithRSA | SHA512_WITH_RSA
| SHA512WithRSA | SHA_WITH_DSA
| SHA_WITH_RSA | SHAWithDSA
| SHAWithRSA

```

### **-cert -elimina**

Eliminare un certificato:

```
-cert -delete -db filename -pw password -label label
```

### **-cert -dettagli**

Elencare le informazioni dettagliate per un certificato specifico:

```
-cert -details -db filename -pw password -label label
```

### **-cert -esporta**

Esportare un certificato personale e la chiave privata associata da un database di chiavi in un file PKCS #12 o in un altro database di chiavi:

```

-cert -export -db filename -pw password -label label
-type cms | pkcs12
-target filename -target_pw password -target_type
cms | pkcs12

```

### **-cert -estrai**

Estrarre un certificato da un database di chiavi:

```

-cert -extract -db filename -pw password -label label
-target filename
-format ascii | binary

```

### **-cert -importa**

Importare un certificato personale da un database di chiavi:

```

-cert -import -file filename -pw password -type
pkcs12 -target filename
-target_pw password -target_type cms -label
label

```

L'opzione `-label` è obbligatoria e specifica l'etichetta del certificato che deve essere importato dal database delle chiavi di origine.

L'opzione `-new_label` è facoltativa e consente al certificato importato di ottenere un'etichetta diversa nel database delle chiavi di destinazione dall'etichetta nel database di origine.

### **-cert -elenco**

Elenca tutti i certificati in un database delle chiavi:



```
-cert -list all | personal | CA
-db filename -pw password
```

### **-cert -ricevere**

Ricevere un certificato da un file:

```
-cert -receive -file filename -db filename -pw password
-format ascii | binary -default_cert yes |
no
```

### **-cert -segno**

Firma un certificato:

```
-cert -sign -db filename -file filename -pw password
-label label -target filename
-format ascii | binary -expire days
-sig_alg MD2_WITH_RSA | MD2WithRSA | MD5_WITH_RSA
|
MD5WithRSA | SHA1WithDSA | SHA1WithRSA
|
SHA256_WITH_RSA | SHA256WithRSA |
SHA2WithRSA | SHA384_WITH_RSA |
SHA384WithRSA | SHA512_WITH_RSA |
SHA512WithRSA | SHA_WITH_DSA |
SHA_WITH_RSA | SHAWithDSA |
SHAWithRSA
```

### **-certreq -crea**

Creare una richiesta di certificato:

```
-certreq -create -db filename -pw password
-label label -dn distinguished_name
-size 1024 | 512 -file filename
-sig_alg MD2_WITH_RSA | MD2WithRSA |
MD5_WITH_RSA | MD5WithRSA |
SHA1WithDSA | SHA1WithRSA |
SHA256_WITH_RSA | SHA256WithRSA |
SHA2WithRSA | SHA384_WITH_RSA |
SHA384WithRSA | SHA512_WITH_RSA |
SHA512WithRSA | SHA_WITH_DSA |
SHA_WITH_RSA | SHAWithDSA |
SHAWithRSA
```

### **-certreq -elimina**

Eliminare una richiesta certificato:

```
-certreq -delete -db filename -pw password -label
label
```

### **-certreq -dettagli**

Elencare le informazioni dettagliate di una richiesta di certificato specifica:

```
-certreq -details -db filename -pw password -label
label
```

Elencare le informazioni dettagliate su una richiesta di certificato e visualizzare la richiesta di certificato completa:

```
-certreq -details -showOID -db filename
-pw password -label label
```

### -certreq -estrazione

Estrarre una richiesta di certificato da un database di richieste di certificato in un file:

```
-certreq -extract -db filename -pw password  
-label label -target filename
```

### -certreq -elenco

Elencare tutte le richieste di certificato nel database delle richieste di certificato:

```
-certreq -list -db filename -pw password
```

### -certreq -ricrea

Ricreare una richiesta di certificato:

```
-certreq -recreate -db filename -pw password  
-label label -target filename
```

ULW

## Comandi per operazioni unità crittografiche

È possibile utilizzare i comandi `runmqckm` e `runmqakm` per la gestione di chiavi e certificati per le operazioni delle periferiche di crittografia.

**Nota:** IBM 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`.

### -keydb -modificapw

Modificare la password per un'unità crittografica:

```
-keydb -changepw -crypto module_name -tokenlabel token_label  
-pw password -new_pw new_password
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che `runmqckm` e `strmqikm` 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, in quanto i programmi `strmqikm` e `runmqckm` sono a 32 bit su tali piattaforme.

### -keydb -elenco

Elencare i tipi di database delle chiavi attualmente supportati:

```
-keydb -list
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che `runmqckm` e `strmqikm` 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, in quanto i programmi `strmqikm` e `runmqckm` sono a 32 bit su tali piattaforme.

### -cert -aggiungi

Aggiungere un certificato da un file a una unità crittografica:

```
-cert -add -crypto module_name -tokenlabel token_label
```

```
-pw password -label label -file filename -format  
ascii | binary
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

#### **-cert -crea**

Creare un certificato autofirmato su un'unità crittografica:

```
-cert -create -crypto module_name -tokenlabel token_label  
  
-pw password -label label -dn distinguished_name  
-size 1024 | 512  
-x509version 3 | 1 | 2 -default_cert no  
| yes -expire days  
-sig_alg MD2_WITH_RSA | MD2WithRSA |  
MD5_WITH_RSA | MD5WithRSA |  
SHA1WithDSA | SHA1WithRSA |  
SHA256_WITH_RSA | SHA256WithRSA |  
SHA2WithRSA | SHA384_WITH_RSA |  
SHA384WithRSA | SHA512_WITH_RSA |  
SHA512WithRSA | SHA_WITH_DSA |  
SHA_WITH_RSA | SHAWithDSA |  
SHAWithRSA
```

**Nota:** Non è possibile importare un certificato contenente più attributi OU (organizational unit) nel DN (distinguished name).

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

#### **-cert -elimina**

Eliminare un certificato su un'unità crittografica:

```
-cert -delete -crypto module_name -tokenlabel token_label  
-pw password -label label
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

#### **-cert -dettagli**

Elencare le informazioni dettagliate per un certificato specifico su una unità crittografica:

```
-cert -details -crypto module_name -tokenlabel token_label  
  
-pw password -label label
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

Elencare le informazioni dettagliate e visualizzare il certificato completo per un certificato specifico su una periferica crittografica:

```
-cert -details -showOID -crypto module_name -tokenlabel  
token_label  
-pw password -label label
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

#### **-cert -estrai**

Estrarre un certificato da un database di chiavi:

```
-cert -extract -crypto module_name -tokenlabel token_label  
-pw password -label label -target filename  
-format ascii | binary
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

#### **-cert -importa**

Importare un certificato in una unità crittografica con il supporto del database di chiavi secondario:

```
-cert -import -db filename -pw password -label label  
-type cms  
-crypto module_name -tokenlabel token_label -pw  
password  
-secondaryDB filename -secondaryDBpw password
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

```
-cert -import -db filename -pw password -label label  
-type cms  
-crypto module_name -tokenlabel token_label -pw  
password  
-secondaryDB filename -secondaryDBpw password -fips
```

Importare un certificato PKCS #12 in una periferica crittografica con il supporto del database delle chiavi secondario:

```
-cert -import -file filename -pw password -type pkcs12  
-crypto module_name -tokenlabel token_label -pw  
password  
-secondaryDB filename -secondaryDBpw password
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

```
-cert -import -file filename -pw password -type pkcs12
-crypto module_name -tokenlabel token_label -pw
password
-secondaryDB filename -secondaryDBpw password -fips
```

**Nota:** Non è possibile importare un certificato contenente più attributi OU (organizational unit) nel DN (distinguished name).

#### **-cert -elenco**

Elencare tutti i certificati su un'unità crittografica:

```
-cert -list all | personal | CA
-crypto module_name -tokenlabel token_label -pw
password
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

#### **-cert -ricevere**

Ricevere un certificato da un file a una unità crittografica con supporto database di chiavi secondario:

```
-cert -receive -file filename -crypto module_name -tokenlabel
token_label
-pw password -default_cert yes | no
-secondaryDB filename -secondaryDBpw password -format
ascii | binary
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

Utilizzando il comando **runmqakm** :

#### **-certreq -crea**

Creare una richiesta di certificato su un'unità crittografica:

```
-certreq -create -crypto module_name -tokenlabel token_label
-pw password -label label -dn distinguished_name
-size 1024 | 512 -file filename
-sig_alg MD2_WITH_RSA | MD2WithRSA | MD5_WITH_RSA
|
MD5WithRSA | SHA1WithDSA | SHA1WithRSA
|
SHA256_WITH_RSA | SHA256WithRSA
SHA2WithRSA | SHA384_WITH_RSA |
SHA384WithRSA | SHA512_WITH_RSA |
SHA512WithRSA | SHA_WITH_DSA |
```

```
SHA_WITH_RSA | SHAWithDSA |  
SHAWithRSA
```

**Nota:** Non è possibile importare un certificato contenente più attributi OU (organizational unit) nel DN (distinguished name).

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

#### **-certreq -elimina**

Cancellare una richiesta di certificato da un'unità crittografica:

```
-certreq -delete -crypto module_name -tokenlabel token_label  
-pw password -label label
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

#### **-certreq -dettagli**

Elencare le informazioni dettagliate di una specifica richiesta di certificato su una periferica crittografica:

```
-certreq -details -crypto module_name -tokenlabel token_label  
-pw password -label label
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

Elencare le informazioni dettagliate su una richiesta di certificato e visualizzare la richiesta di certificato completa su una periferica crittografica:

```
-certreq -details -showOID -crypto module_name -tokenlabel  
token_label  
-pw password -label label
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

#### **-certreq -estrazione**

Estrarre una richiesta di certificato da un database di richieste di certificato su una unità crittografica in un file:

```
-certreq -extract -crypto module_name -tokenlabel token_label
```

```
-pw password -label label -target filename
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

#### -certreq -elenco

Elencare tutte le richieste di certificato nel database delle richieste di certificato su una unità crittografica:

```
-certreq -list -crypto module_name -tokenlabel token_label  
-pw password
```

Se si stanno utilizzando certificati o chiavi memorizzati su hardware di crittografia PKCS #11, tenere a mente che **runmqckm** e **strmqikm** 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, in quanto i programmi **strmqikm** e **runmqckm** sono a 32 bit su tali piattaforme.

## ULW opzioni runmqckm e runmqakm

È possibile utilizzare le opzioni della riga comandi **runmqckm** (iKeycmd) e **runmqakm** per la gestione di chiavi, certificati e richieste di certificato.

ULW

Il comando **runmqakm** è disponibile su UNIX, Linux, and Windows.

Windows

UNIX

Il comando **runmqckm** è disponibile su UNIX e Windows.

**Nota:** IBM 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 .

Il significato di un'opzione può dipendere dall'oggetto e dall'azione specificati nel comando.

Parametro	Descrizione
-crea	Opzione per creare un database delle chiavi.
-crypto	Nome del modulo per gestire un'unità crittografica PKCS #11 . Il valore dopo <b>-crypto</b> è facoltativo se si specifica il nome del modulo nel file delle proprietà. Se si utilizzano certificati o chiavi memorizzati sull'hardware crittografico PKCS #11 , tenere presente che <b>runmqckm</b> e <b>strmqikm</b> vengono eseguiti utilizzando la JVM ( Java virtual machine) fornita con l'installazione di IBM MQ . I moduli esterni richiesti per il supporto PKCS #11 verranno caricati nel processo JVM, pertanto è necessario disporre di una libreria PKCS #11 installata per la gestione dell'hardware crittografico che corrisponda al bit della JVM e specificare questa libreria su <b>runmqckm</b> o <b>strmqikm</b> .
-db	Nome percorso completo di un database di chiavi.

Tabella 83. Opzioni che possono essere utilizzate con **runmqckm** e **runmqakm** (Continua)

Parametro	Descrizione
<b>-default_cert</b>	Imposta un certificato come certificato predefinito. Il valore può essere yes o no. Il valore predefinito è No.
<b>-dn</b>	Nome distinto X.500 . Il valore è una stringa racchiusa tra virgolette, ad esempio "CN=John Smith,O=IBM,OU=Test,C=GB". Si noti che sono richiesti solo gli attributi O e C. La specifica di un nome comune (CN) è facoltativa.
<b>-encryption</b>	Livello di crittografia utilizzato nel comando di esportazione certificato. Il valore può essere strong o weak. Il valore predefinito è strong.
<b>-expire</b>	Il tempo di scadenza in giorni di un certificato o di una password del database. Il valore predefinito è 365 giorni per una password di certificato.  Non esiste un'ora predefinita per una password del database: utilizzare il parametro <b>-expire</b> per impostare esplicitamente un'ora di scadenza della password del database.
<b>-file</b>	Nome file di un certificato o di una richiesta di certificato.
<b>-fips</b>	specifica che il comando viene eseguito 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 <b>runmqakm</b> non riesce.
<b>-format</b>	Formato di un certificato. Il valore può essere ascii per Base64_encoded ASCII o binary per i dati DER binari. Il valore predefinito è ascii.
<b>-label</b>	Etichetta allegata a un certificato o a una richiesta di certificato. Se il certificato è un certificato personale utilizzato per identificare un'applicazione client IBM MQ o un gestore code, l'etichetta deve corrispondere all'impostazione CERTLABL ( IBM MQ certificate label); per ulteriori informazioni, consultare <a href="#">"Etichette dei certificati digitali, comprensione dei requisiti"</a> a pagina 26.
<b>-new_format</b>	Nuovo formato del database di chiavi.
<b>-new_label</b>	Utilizzata in un comando di importazione certificati, questa opzione consente di importare un certificato con un'etichetta diversa da quella che aveva nel database delle chiavi di origine. Se il certificato è un certificato personale utilizzato per identificare un'applicazione client IBM MQ o un gestore code, l'etichetta deve corrispondere all'impostazione CERTLABL ( IBM MQ certificate label); per ulteriori informazioni, consultare <a href="#">"Etichette dei certificati digitali, comprensione dei requisiti"</a> a pagina 26.
<b>-new_pw</b>	Nuova password database.
<b>-old_format</b>	Vecchio formato del database di chiavi.
<b>-pw</b>	Password per il database delle chiavi o il file PKCS #12 .
<b>-secondaryDB</b>	Nome di un database di chiavi secondario per le operazioni della periferica PKCS #11 .
<b>-secondaryDBpw</b>	Password per il database delle chiavi secondario per le operazioni della periferica PKCS #11 .
<b>-showOID</b>	Visualizza il certificato completo o la richiesta di certificato.



Tabella 83. Opzioni che possono essere utilizzate con **runmqckm** e **runmqakm** (Continua)

Parametro	Descrizione
<b>-sig_alg</b>	<p>L'algoritmo di hash utilizzato durante la creazione di una richiesta di certificato, di un certificato autofirmato o la firma di un certificato. Questo algoritmo di hash viene utilizzato per creare la firma associata al certificato appena creato o alla richiesta di certificato.</p> <p>Per <b>runmqckm</b>, il valore può essere, MD2_WITH_RSA, MD2WithRSA, MD5_WITH_RSA, MD5WithRSA, SHA1WithDSA, SHA1WithECDSA, SHA1WithRSA, SHA2/ECDSA, SHA224WithECDSA, SHA256_WITH_RSA, SHA256WithECDSA, SHA256WithRSA, SHA2WithECDSA, SHA3/ECDSA, SHA384_WITH_RSA, SHA384WithECDSA, SHA384WithRSA, SHA3WithECDSA, SHA5/ECDSA, SHA512_WITH_RSA, SHA512WithECDSA, SHA512WithRSA, SHA5WithECDSA, SHA_WITH_DSA, SHA_WITH_RSA, SHAWithDSA, SHAWithRSA. Il valore di default è SHA1WithRSA.</p> <p>Per <b>runmqakm</b>, 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_SHA384o EC_ecdsa_with_SHA512. Il valore di default è SHA1WithRSA.</p>
<b>-size</b>	<p>Dimensione chiave.</p> <p>Per <b>runmqckm</b>, il valore può essere 512, 1024 o 2048. Il valore predefinito è 1024 bit.</p> <p>Per <b>runmqakm</b>, il valore dipende dall'algoritmo di firma:</p> <ul style="list-style-type: none"> <li>• Per gli algoritmi di firma RSA (l'algoritmo predefinito utilizzato se non viene specificato <b>-sig_alg</b>), il valore può essere 512, 1024, 2048 o 4096. Una dimensione chiave RSA di 512 bit non è consentita se il parametro <b>-fips</b> è abilitato. La dimensione della chiave RSA predefinita è 1024 bit.</li> <li>• Per gli algoritmi della curva ellittica, il valore può essere 256, 384 o 512. La dimensione della chiave della curva ellittica predefinita dipende dall'algoritmo di firma. Per SHA256, è 256; per SHA384, è 384; e per SHA512, è 512.</li> </ul>
<b>-stash</b>	<p>Eseguire lo stash della password del database delle chiavi in un file. Applicabile solo ai database di tipo CMS e PKCS12.</p> <p><b>Nota:</b> <b>-stash</b> è valido sui comandi <b>-keydb -create</b> per indicare a <b>runmqckm/ runmqakm</b> di creare un file stash contenente la password.</p> <p>L'immissione del comando \$ runmqakm -help elenca solo i parametri della guida di alto livello.</p>

Tabella 83. Opzioni che possono essere utilizzate con **runmqckm** e **runmqakm** (Continua)

Parametro	Descrizione
<b>-stashed</b>	Indica che la password per il database delle chiavi o il file PKCS #12 si trova in un file stash.  <b>Nota:</b> L'opzione <b>-stashed</b> è valida per le chiamate a parte i comandi <b>-keydb -create</b> . Se non si specifica questa opzione, è necessario fornire la password utilizzando <b>-pw</b> .  Inoltre, solo quando si indica al comando il tipo di azione che si sta eseguendo, viene visualizzata la guida dettagliata che mostra <b>-stashed</b> .
<b>-target</b>	File di destinazione o database.
<b>-target_pw</b>	La password per il database di chiavi se <b>-target</b> specifica un database di chiavi.
<b>-target_type</b>	Tipo di database specificato dall'operando <b>-target</b> . Consultare il parametro <b>-type</b> per i valori consentiti.
<b>-tokenLabel</b>	Etichetta di un'unità crittografica PKCS #11.
<b>-trust</b>	Stato di attendibilità di un certificato CA. Il valore può essere enable o disable. Il valore predefinito è enable.
<b>-type</b>	Tipo di database. Il valore può essere uno dei seguenti: <ul style="list-style-type: none"> <li>• cms per un database di chiavi CMS</li> <li>• pkcs12 per un file PKCS #12.</li> </ul>
<b>-x509version</b>	Versione del certificato X.509 da creare. Il valore può essere 1, 2 o 3. Il valore predefinito è 3.
<b>-rfc3339</b>	Utilizzare questo parametro per emettere la data in formato RFC 3339 per il comando <code>runmqakm -cert -details</code> , che ha il seguente formato: <pre>Not Before : 2015-08-26T08:53:37Z Not After  : 2016-08-26T08:53:37Z</pre> <p>Tenere presente che il parametro <b>-rfc3339</b> deve essere visualizzato nel comando dopo i parametri aggiuntivi:</p> <pre>runmqakm -cert -details -db exampleDB -stashed -label certificateLabel -rfc3339</pre>

**Nota:** Le proprietà fornite con IBM Global Security Kit (GSKit) relative al parametro di codifica della chiave simmetrica **-seckey** nel programma di utilità **runmqckm** vengono ignorate e non supportate da IBM MQ.

## ULW codici di errore runmqakm

Una tabella dei codici di errore numerici emessi da runmqakm e il loro significato.

Codice di errore	Messaggio di errore
0	Operazione riuscita
1	Si è verificato un errore sconosciuto
2	Si è verificato un errore di codifica / decodifica ASN.1.

<b>Codice di errore</b>	<b>Messaggio di errore</b>
3	Si è verificato un errore durante l'inizializzazione del codificatore / decodificatore ASN.1 .
4	Si è verificato un errore di codifica / decodifica ASN.1 a causa di un indice fuori intervallo o di un campo facoltativo non esistente.
5	Si è verificato un errore del database.
6	Si è verificato un errore durante l'apertura del file di database, verificare l'esistenza del file e l'autorizzazione.
7	Si è verificato un errore in fase di riapertura del file database.
8	Creazione del database non riuscita.
9	Il database esiste già.
10	Si è verificato un errore in fase di eliminazione del file database.
11	Impossibile aprire il database.
12	Si è verificato un errore in fase di lettura del file database.
13	Si è verificato un errore durante la scrittura dei dati nel file database.
14	Si è verificato un errore di convalida del database.
15	È stata rilevata una versione di database non valida.
16	È stata rilevata una password database non valida.
17	È stato rilevato un tipo di file database non valido.
18	Il database specificato è stato danneggiato.
19	È stata fornita una password non valida oppure il database delle chiavi è stato alterato o danneggiato.
20	Si è verificato un errore di integrità della voce della chiave del database.
21	Un certificato duplicato già esiste nel database.
22	Una chiave duplicata esiste già nel database (ID record).
23	Un certificato con la stessa etichetta già esisteva nel database delle chiavi.
24	Una chiave duplicata esiste già nel database (Firma).
25	Una chiave duplicata esiste già nel database (certificato non firmato).
26	Una chiave duplicata esiste già nel database (Emittente e Numero di serie).

<b>Codice di errore</b>	<b>Messaggio di errore</b>
27	Una chiave duplicata esiste già nel database (Info chiave pubblica oggetto).
28	Una chiave duplicata esiste già nel database (CRL senza firma).
29	L'etichetta è stata utilizzata nel database.
30	Si è verificato un errore di codifica della parola d'ordine.
31	Si è verificato un errore relativo a LDAP. (LDAP non è supportato da questo programma)
32	Si è verificato un errore crittografico.
33	Si è verificato un errore di codifica / decodifica.
34	È stato rilevato un algoritmo crittografico non valido.
35	Si è verificato un errore in fase di firma dei dati.
36	Si è verificato un errore in fase di verifica dei dati.
280	Si è verificato un errore durante il calcolo del digest dei dati.
38	È stato trovato un parametro crittografico non valido.
39	È stato rilevato un algoritmo crittografico non supportato.
40	La dimensione di input specificata è maggiore della dimensione modulo supportata.
41	È stata trovata una dimensione modulo non supportata.
42	Si è verificato un errore di convalida del database.
43	Convalida della voce chiave non riuscita.
44	Esiste un campo di estensione duplicato.
45	La versione della chiave è errata.
46	Non esiste un campo estensione richiesto.
47	Il periodo di validità non comprende oggi o non rientra nel periodo di validità dell'emittente
48	Il periodo di validità non include oggi o non rientra nel periodo di validità dell'emittente.
49	Si è verificato un errore durante la convalida dell'estensione di utilizzo della chiave privata.
50	Non è stato trovato l'emittente della chiave.
51	Manca un'estensione del certificato richiesta.
52	È stata trovata un'estensione di vincolo base non valida.

<b>Codice di errore</b>	<b>Messaggio di errore</b>
53	La convalida della firma chiave non è riuscita.
54	La chiave root della chiave non è attendibile.
55	La chiave è stata revocata.
56	Si è verificato un errore durante la convalida dell'estensione dell'identificativo della chiave di autorizzazione.
57	Si è verificato un errore durante la convalida dell'estensione di utilizzo della chiave privata.
58	Si è verificato un errore durante la convalida dell'estensione del nome alternativo dell'oggetto.
59	Si è verificato un errore durante la convalida dell'estensione del nome alternativo dell'emittente.
60	Si è verificato un errore durante la convalida dell'estensione di utilizzo della chiave.
61	È stata trovata un'estensione critica sconosciuta.
62	Si è verificato un errore durante la convalida delle voci della coppia di chiavi.
63	Si è verificato un errore durante la convalida del CRL.
64	Si è verificato un errore mutex.
65	È stato rilevato un parametro non valido.
66	È stato rilevato un parametro null o un errore di assegnazione della memoria.
67	Il numero o la dimensione sono troppo grandi o troppo piccoli.
68	La vecchia password non è valida.
69	La nuova password non è valida.
70	La password è scaduta.
69	Si è verificato un errore relativo al thread.
84	Si è verificato un errore durante la creazione dei thread.
73	Si è verificato un errore mentre un thread era in attesa di uscita.
74	Si è verificato un errore I/O.
75	Si è verificato un errore in fase di carico di CMS.
76	Si è verificato un errore relativo all'hardware di crittografia.
77	La routine di inizializzazione della libreria non è stata richiamata correttamente.

<b>Codice di errore</b>	<b>Messaggio di errore</b>
78	La tabella di gestione del database interno è danneggiata.
79	Si è verificato un errore di allocazione memoria.
80	È stata trovata un'opzione non riconosciuta.
81	Si è verificato un errore durante il richiamo delle informazioni sull'ora.
82	Si è verificato un errore di creazione mutex.
83	Si è verificato un errore durante l'apertura del catalogo messaggi.
84	Si è verificato un errore durante l'apertura del catalogo dei messaggi di errore
85	È stato trovato un nome file null.
86	Si è verificato un errore durante l'apertura dei file, verificare l'esistenza del file e le autorizzazioni.
87	Si è verificato un errore in fase di apertura dei file da leggere.
88	Si è verificato un errore in fase di apertura dei file da da scrivere.
89	Questo file non esiste.
90	Impossibile aprire il file a causa della relativa impostazione di autorizzazione.
91	Si è verificato un errore in fase di scrittura dei dati sui file.
92	Si è verificato un errore durante l'eliminazione dei file.
93	Sono stati trovati dati Base64-encoded non validi.
94	È stato rilevato un tipo di messaggio Base64 non valido.
95	Si è verificato un errore durante la codifica dei dati con la regola di codifica Base64 .
96	Si è verificato un errore in fase di decodifica dati con codifica Base64.
97	Si è verificato un errore durante il richiamo di una tag DN (distinguished name).
98	Il campo nome comune richiesto è vuoto.
99	Il campo del nome del paese o della regione richiesto è vuoto.
100	È stato trovato un handle di database non valido.
101	Il database delle chiavi non esiste.
102	Il database coppia di chiavi richiesta non esiste.
103	Il file delle password non esiste.

<b>Codice di errore</b>	<b>Messaggio di errore</b>
104	La nuova password è identica a quella precedente.
105	Nessuna chiave trovata nel database delle chiavi.
106	Non è stata trovata alcuna chiave di richiesta.
107	Non è stata trovata alcuna CA attendibile.
108	Nessuna chiave di richiesta trovata per il certificato.
109	Non esiste alcuna chiave privata nel database delle chiavi.
110	Non esiste alcuna chiave predefinita nel database delle chiavi.
111	Non c'è alcuna chiave privata nel record chiave.
112	Non è presente alcun certificato nel record chiave.
113	Non esiste alcuna voce CRL.
114	È stato trovato un nome file di database delle chiavi non valido.
115	È stato trovato un tipo di chiave privata non riconosciuto.
116	È stato trovato un input di nome distinto non valido.
117	Non è stata trovata alcuna voce chiave con l'identificativo chiave specificato.
118	L'elenco di etichette chiave è stato danneggiato.
119	I dati di input non sono dati PKCS12 validi.
120	La password non è valida oppure i dati PKCS12 sono stati danneggiati o creati con la versione successiva di PKCS12
121	È stato rilevato un tipo di esportazione chiave non riconosciuto.
122	È stato rilevato un algoritmo di crittografia basato su password non supportato.
123	Si è verificato un errore durante la conversione del file di chiavi in un database di chiavi CMS.
124	Si è verificato un errore durante la conversione del database di chiavi CMS in un file key ring.
125	Si è verificato un errore durante la creazione di un certificato per la richiesta di certificato.
126	Impossibile creare una catena di emittenti completa.
127	Sono stati trovati dati WEBDB non validi.
128	Non ci sono dati da scrivere nel file key ring.

<b>Codice di errore</b>	<b>Messaggio di errore</b>
129	Il numero di giorni immesso si estende oltre il periodo di validità consentito.
130	La password è troppo breve; deve contenere almeno {0} caratteri.
131	Una password deve contenere almeno una cifra numerica.
132	Tutti i caratteri nella password sono caratteri alfabetici o numerici.
133	È stato specificato un algoritmo di firma non riconosciuto o non supportato.
134	È stato rilevato un tipo di database non valido.
135	Il database di chiavi secondario specificato è utilizzato da un altro dispositivo PKCS#11 .
136	Non è stato specificato alcun database di chiavi secondario.
137	L'etichetta non esiste sulla periferica PKCS#11 .
138	Password richiesta per accedere alla periferica PKCS#11 .
139	Password non richiesta per accedere al dispositivo PKCS#11 .
140	Impossibile caricare la libreria crittografica.
141	PKCS#11 non è supportato per questa operazione.
142	Un'operazione su una periferica PKCS#11 non è riuscita.
143	L'utente LDAP non è un utente valido. (LDAP non è supportato da questo programma)
144	L'utente LDAP non è un utente valido. (LDAP non è supportato da questo programma)
145	La query LDAP non è riuscita. (LDAP non è supportato da questo programma)
146	È stata trovata una catena di certificati non valida.
147	Il certificato root non è attendibile.
148	È stato rilevato un certificato revocato.
149	Una funzione dell'oggetto crittografico ha avuto esito negativo.
4	Non è disponibile alcuna origine dati CRL (Certificate Revocation List).
151	Non è disponibile alcun token crittografico.
152	La modalità FIPS non è disponibile.
153	Esiste un conflitto con le impostazioni della modalità FIPS.



<b>Codice di errore</b>	<b>Messaggio di errore</b>
154	La password immessa non soddisfa il livello minimo richiesto.
200	Si è verificato un errore durante l'inizializzazione del programma.
201	La tokenizzazione degli argomenti passati al programma runmqakm non è riuscita.
202	L'oggetto identificato nel comando non è un oggetto riconosciuto.
203	L'azione passata non è un'azione -keydb nota.
204	L'azione passata non è un'azione -cert nota.
205	L'azione passata non è un'azione -certreq nota.
206	Manca una tag per il comando richiesto.
207	Il valore passato con la tag -version non è un valore riconosciuto.
208	Il valore passato con la tag -size non è un valore riconosciuto.
209	Il valore passato con la tag -dn non ha il formato corretto.
210	Il valore passato con la tag -format non è un valore riconosciuto.
211	Si è verificato un errore associato all'apertura del file.
212	PKCS12 non è supportato in questa fase.
213	Il token crittografico per cui si sta tentando di modificare la password non è protetto da password.
214	PKCS12 non è supportato in questa fase.
215	La password immessa non soddisfa il livello minimo richiesto.
216	La modalità FIPS non è disponibile.
217	Il numero di giorni immesso come data di scadenza non è compreso nell'intervallo consentito.
218	Il livello di sicurezza della password non ha superato i requisiti minimi.
219	Non è stato trovato alcun certificato predefinito nel database delle chiavi richiesto.
220	È stato rilevato uno stato di attendibilità non valido.
221	È stato rilevato un algoritmo di firma non supportato. A questo punto sono supportati solo MD5 e SHA1 .
222	PCKS11 non supportato per quella particolare operazione.

Codice di errore	Messaggio di errore
223	L'azione passata non è un'azione -casuale nota.
224	Non è consentita una lunghezza inferiore a zero.
225	Quando si utilizza la tag -strong, la lunghezza minima della password è 14 caratteri.
226	Quando si utilizza la tag -strong, la lunghezza massima della password è 300 caratteri.
227	L'algoritmo MD5 non è supportato in modalità FIPS.
228	La tag site non è supportata per il comando -cert -list. Questo attributo viene aggiunto per la compatibilità con le versioni precedenti e per potenziali miglioramenti futuri.
229	Il valore associato alla tag -ca non è riconosciuto. Il valore deve essere 'true' o 'false'.
230	Il valore passato con la tag -type non è valido.
231	Il valore passato con la tag -expire è inferiore all'intervallo consentito.
232	L'algoritmo di codifica utilizzato o richiesto non è supportato.
233	La destinazione esiste già.

## Protezione dei dettagli di autenticazione del database

Se si sta utilizzando l'autenticazione nome utente e password per connettersi al gestore database, è possibile memorizzarli nell'archivio delle credenziali MQ XA per evitare di memorizzare la password in testo semplice nel file `qm.ini`.

### Aggiorna XAOpenString per il gestore risorse

Per utilizzare l'archivio credenziali è necessario modificare XAOpenString nel file `qm.ini`. La stringa viene utilizzata per collegarsi al gestore database. Specificare i campi sostituibili per identificare dove il nome utente e la password vengono sostituiti all'interno della stringa XAOpenString.

- Il campo `+USER+` viene sostituito con il valore del nome utente memorizzato nell'archivio XACredentials.
- Il campo `+PASSWORD+` viene sostituito con il valore della password memorizzato nell'archivio XACredentials.

I seguenti esempi mostrano come modificare una XAOpenString per utilizzare il file delle credenziali per connettersi al database.

#### Connessione a un database Db2

```
XAResourceManager:
  Name=mydb2
  SwitchFile=db2swit
  XAOpenString=db=mydbname,uid=+USER+,pwd=+PASSWORD+,toc=t
  ThreadOfControl=THREAD
```

## Connessione ad un database Oracle

```
XAResourceManager:  
Name=myoracle  
SwitchFile=oraswit  
XAOpenString=Oracle_XA+Acc=P/+USER+//PASSWORD++SesTm=35  
+LogDir=/tmp+threads=true  
ThreadOfControl=THREAD
```

## Gestire le credenziali per il database nell'archivio credenziali XA di MQ

Dopo aver aggiornato il file `qm.ini` con le stringhe di credenziale sostituibile, è necessario aggiungere il nome utente e la password all'archivio credenziali MQ utilizzando il comando **setmqxacred**. È inoltre possibile utilizzare **setmqxacred** per modificare le credenziali esistenti, eliminare le credenziali o elencare le credenziali. I seguenti esempi forniscono alcuni casi di utilizzo tipici:

### Aggiunta di credenziali

Il seguente comando salva in modo sicuro il nome utente e password per il gestore code QM1 per la risorsa `mqdb2`.

```
setmqxacred -m QM1 -x mydb2 -u user1 -p Password2
```

### Aggiornamento delle credenziali

Per aggiornare il nome utente e la password utilizzati per connettersi a un database, immettere nuovamente il comando **setmqxacred** con i nuovi nome utente e password:

```
setmqxacred -m QM1 -x mydb2 -u user3 -p Password4
```

È necessario riavviare il gestore code per rendere effettive le modifiche.

### Eliminazione delle credenziali

Il seguente comando cancella le credenziali:

```
setmqxacred -m QM1 -x mydb2 -d
```

### Elenco delle credenziali

Il seguente comando elenca le credenziali:

```
setmqxacred -m QM1 -l
```

### Informazioni correlate

#### **setmqxacred**

ULW

## Protezione dei client AMQP

È possibile utilizzare una serie di meccanismi di sicurezza per proteggere le connessioni dai client AMQP e garantire che i dati siano adeguatamente protetti sulla rete. È possibile creare la sicurezza nelle applicazioni MQ Light. È anche possibile utilizzare le funzioni di sicurezza esistenti di IBM MQ con client AMQP, nello stesso modo in cui le funzioni vengono utilizzate per altre applicazioni.

### Regole di autenticazione di canale (CHLAUTH)

È possibile utilizzare regole di autenticazione di canale per limitare le connessioni TCP a un gestore code. I canali AMQP supportano l'utilizzo delle regole di autenticazione di canale configurate per il proprio gestore code. Se le regole di autenticazione di canale sono definite con un profilo che corrisponde a qualsiasi canale AMQP sul gestore code, queste regole vengono applicate a tali canali. Per impostazione

predefinita, l'autenticazione di canale è abilitata sui nuovi gestori code di IBM MQ , pertanto è necessario completare almeno una configurazione prima di poter utilizzare un canale AMQP.

Per ulteriori informazioni su come configurare le regole di autenticazione di canale per consentire le connessioni AMQP al tuo gestore code, vedi [Creazione e utilizzo di canali AMQP](#).

## Autenticazione della connessione (CONNAUTH)

È possibile utilizzare l'autenticazione della connessione per autenticare le connessioni a un gestore code. I canali AMQP supportano l'utilizzo dell'autenticazione della connessione per controllare l'accesso al gestore code dalle applicazioni AMQP.

Il protocollo AMQP utilizza il framework SASL (Simple Authentication and Security Layer) per specificare come viene autenticata una connessione. Esistono vari meccanismi SASL e IBM MQ supporta due meccanismi SASL: ANONYMOUS e PLAIN.

Nel caso di ANONYMOUS, non vengono trasmesse credenziali dal client al gestore code per l'autenticazione. Se l'oggetto MQ AUTHINFO specificato nell'attributo CONNAUTH ha un valore CHCKCLNT di REQUIRED o REQDADM (se ci si connette come utente di gestione), la connessione viene rifiutata. Se il valore di CHCKCLNT è NONE o OPTIONAL, la connessione viene accettata.

Nel caso di PLAIN, un nome utente e una password vengono trasmessi dal client al gestore code per l'autenticazione. Se l'oggetto MQ AUTHINFO specificato nell'attributo CONNAUTH ha un valore CHCKCLNT di NONE, la connessione viene rifiutata. Se il valore di CHCKCLNT è OPTIONAL, REQUIRED o REQDADM (se ci si connette come utente di gestione), il nome utente e la parola d'ordine vengono controllati dal gestore code. Il gestore code controlla il sistema operativo (se l'oggetto AUTHINFO è di tipo IDPWOS) o un repository LDAP (se l'oggetto AUTHINFO è di tipo IDPWLDAP).

La seguente tabella riepiloga questo comportamento di autenticazione:

<i>Tabella 84. Riepilogo dei meccanismi SASL e autenticazione della connessione</i>		
<b>Meccanismo SASL</b>	<b>Credenziali passate dal client al gestore code?</b>	<b>valore CHCKCLNT</b>
anonimo	No	REQUIRED o REQDADM - connessione rifiutata  NONE o OPTIONAL - connessione accettata
NORMALE	Sì, nome utente e password	REQUIRED, REQDADM o OPTIONAL - nome utente e password controllati dal gestore code  NONE - connessione rifiutata

Se si sta utilizzando un client MQ Light , è possibile specificare le credenziali includendo tali credenziali nell'indirizzo AMQP a cui ci si connette, ad esempio:

```
amqp://mwhitehead:mYp4ssw0rd@localhost:5672/sports/football
```

## Impostazione MCAUSER su un canale

I canali AMQP hanno un attributo MCAUSER, che è possibile utilizzare per impostare l'ID utente IBM MQ con cui sono autorizzate tutte le connessioni a tale canale. Tutte le connessioni dai client AMQP a tale canale adottano l'ID MCAUSER configurato. Tale ID utente viene utilizzato per l'autorizzazione della messaggistica su argomenti differenti.

Si consiglia di utilizzare l'autenticazione di canale (CHLAUTH) per proteggere le connessioni ai gestori code. Se si utilizza l'autenticazione di canale, si consiglia di configurare il valore di MCAUSER per un utente non privilegiato. Ciò garantisce che se una connessione a un canale non corrisponde a una regola CHLAUTH, la connessione non è autorizzata ad eseguire alcuna messaggistica sul gestore code.

**Nota:** **Windows** Su Windows, l'impostazione ID utente MCAUSER è supportata solo per gli ID utente con una lunghezza massima di 12 caratteri.

## Supporto SSL/TLS

I canali AMQP supportano la codifica SSL/TLS utilizzando le chiavi dal repository delle chiavi configurato per il tuo gestore code. Le opzioni di configurazione del canale AMQP per la crittografia SSL/TLS supportano le stesse opzioni di altri tipi di canale MQ; è possibile specificare una specifica di crittografia e se il gestore code richiede certificati dalle connessioni client AMQP.

Utilizzando gli attributi FIPS del gestore code, è possibile controllare le suite di cifratura SSL/TLS, che è possibile utilizzare per proteggere le connessioni dai client AMQP.

Per informazioni su come configurare un repository delle chiavi per il gestore code, consultare [Utilizzo di SSL o TLS su sistemi UNIX, Linux e Windows](#).

Per informazioni su come configurare il supporto SSL/TLS per una connessione client AMQP, vedi [Creazione e utilizzo dei canali AMQP](#).

## JAAS (Java Authentication and Authorization Service)

**V 9.0.0**

Facoltativamente, è possibile configurare i canali AMQP con un modulo di login JAAS, che può controllare il nome utente e la password forniti da un client AMQP. Vedere [“Configurazione di JAAS per canali AMQP”](#) a pagina 518.

### Informazioni correlate

[Sviluppo di applicazioni client AMQP](#)

[Creazione e utilizzo di canali AMQP](#)

**ULW**

## Limitazione del takeover del client AMQP

Quando viene effettuata una connessione client AMQP con lo stesso identificativo client di una connessione client AMQP esistente, la connessione client esistente viene disconnessa per impostazione predefinita. Tuttavia, è possibile configurare il gestore code per limitare il comportamento del takeover del client in modo che il takeover sia possibile solo quando vengono soddisfatti determinati criteri.

Ad esempio, la disconnessione della connessione client esistente potrebbe non essere appropriata se ci sono applicazioni AMQP sviluppate da team differenti e si verifica che utilizzino lo stesso ID client. Per risolvere questo problema è possibile limitare il takeover del client in base al nome del canale AMQP utilizzato, l'indirizzo IP del client e l'ID utente del client (quando è abilitata l'autenticazione SASL).

Utilizzare le impostazioni degli attributi del gestore code **AdoptNewMCA** e **AdoptNewMCACheck** per specificare il livello richiesto di limitazione del takeover client, come descritto nella seguente tabella:

<i>Tabella 85. Impostazioni di <b>AdoptNewMCA</b> e <b>AdoptNewMCACheck</b> per limitare il takeover del client</i>		
<b>AdoptNewMCA</b>	<b>AdoptNewMCACheck</b>	<b>Criteri controllati prima che sia consentito il takeover del client</b>
NO o non definito	Non applicabile	Nessuna. Il takeover client è consentito per tutte le connessioni client autenticate e che passano tutte le regole CHLAUTH.

Tabella 85. Impostazioni di **AdoptNewMCA** e **AdoptNewMCACheck** per limitare il takeover del client (Continua)

<b>AdoptNewMCA</b>	<b>AdoptNewMCACheck</b>	<b>Criteri controllati prima che sia consentito il takeover del client</b>
ALL (o un valore diverso da NO)	QM o non definito	Nessuna. Il takeover client è consentito per tutte le connessioni client autenticate e che passano tutte le regole CHLAUTH.
ALL (o un valore diverso da NO)	NOME	ID utente (quando SASL è abilitato) Nome canale
ALL (o un valore diverso da NO)	ADDRESS	ID utente (quando SASL è abilitato) Indirizzo IP
ALL (o un valore diverso da NO)	TUTTO	ID utente (quando SASL è abilitato) Nome canale Indirizzo IP

Gli attributi del gestore code **AdoptNewMCA** e **AdoptNewMCACheck** fanno parte della configurazione del gestore code, definita nella stanza CHANNELS. Sui sistemi IBM MQ per Windows e IBM MQ for Linux x86-64, modificare le informazioni di configurazione utilizzando IBM MQ Explorer. Su altri sistemi, modificare le informazioni modificando il file di configurazione `qm.ini`. Per informazioni su come modificare le informazioni sui canali del gestore code, vedere [Attributi della stanza dei canali](#).

#### Informazioni correlate

[Sviluppo di applicazioni client AMQP](#)

[Creazione e utilizzo di canali AMQP](#)

## **ULW** Configurazione di JAAS per canali AMQP

I moduli personalizzati JAAS (Java Authentication and Authorization Service) (JAAS) possono essere utilizzati per autenticare le credenziali di nome utente e password inoltrate a un canale AMQP da parte di un client AMQP quando si connette.

### Informazioni su questa attività

È possibile utilizzare un modulo JAAS personalizzato se si utilizzano già i moduli JAAS per l'autenticazione in altri sistemi basati su Java e si desidera riutilizzare tali moduli per l'autenticazione delle connessioni AMQP a MQ. In alternativa, è possibile scrivere un modulo JAAS personalizzato se le funzioni di autenticazione integrate in MQ non supportano il meccanismo di autenticazione che si desidera utilizzare.

La configurazione dei moduli JAAS per canali AMQP viene eseguita a livello di gestore code. Ciò significa che, se si configura un modulo JAAS per l'autenticazione delle connessioni AMQP al gestore code, il modulo verrà applicato a tutti i canali AMQP. Il nome del canale che ha richiamato il modulo JAAS viene passato al modulo, consentendo di codificare un comportamento di log JAAS differente per i diversi canali.



Altre informazioni vengono trasmesse anche al modulo JAAS :

- L'ID client del client AMQP che sta tentando di autenticare.
- L'indirizzo di rete del client AMQP.

- Il nome del canale che ha richiamato il modulo JAAS .

## Procedura

Configurare un modulo di configurazione JAAS per i canali AMQP completando la seguente procedura:

1. Definire un file `jaas.config` contenente una o più stanze di configurazione del modulo JAAS .  
La stanza deve specificare il nome completo della classe Java che implementa l'interfaccia JAAS `javax.security.auth.spi.LoginModule` .
  - Un file `jaas.config` predefinito viene fornito con il prodotto e si trova in `QM_data_directory/amqp/jaas.config`.
  - Una sezione preconfigurata denominata `MQXRConfig` è già definita nel file `jaas.config` predefinito.
2. Specificare il nome della stanza da utilizzare per canali AMQP.
  -  Aggiungere una proprietà al file `amqp_unix.properties` .
  -  Aggiungere una proprietà al file `amqp_win.properties` .

La proprietà ha il formato seguente:

```
com.ibm.mq.MQXR.JAASConfig=JAAS_stanza_name
```

Ad esempio:

```
com.ibm.mq.MQXR.JAASConfig=MQXRConfig
```

3. Configurare l'ambiente del gestore code per includere la classe del modulo personalizzato. Il servizio AMQP deve avere accesso alla classe Java configurata nella sezione di configurazione JAAS .  
A tale scopo, aggiungere il percorso alla classe JAAS nel file `MQ.service.env` . Modificare il file `service.env` nella directory di configurazione di MQ (`MQ_config_directory`) o nella directory di configurazione del gestore code (`QM_config_directory`) per impostare la variabile `CLASSPATH` sul percorso della classe del modulo JAAS .

## Operazioni successive

Un modulo di login JAAS di esempio viene fornito con il prodotto nella directory `mq_installation_directory/amqp/samples` . Il modulo di login JAAS di esempio autentica tutte le connessioni client, indipendentemente dal nome utente o dalla password con cui si connette il client.

È possibile modificare il codice di origine dell'esempio e ricompilarlo per provare ad autenticare solo utenti specifici con una particolare password. Per configurare il canale AMQP su un sistema UNIX per utilizzare il modulo login JAAS di esempio fornito con il prodotto:

1. Modificare il file `/var/mqm/qmgrs/QMNAME/amqp/amqp_unix.properties` e impostare la proprietà `com.ibm.mq.MQXR.JAASConfig=MQXRConfig`.
2. Modificare il file `/var/mqm/service.env` e impostare la proprietà `CLASSPATH=mq_installation_location/amqp/samples`

Il file `jaas.config` contiene già una stanza denominata `MQXRConfig` che specifica la classe di esempio `samples.JAASLoginModule` come classe del modulo di login. Non sono necessarie modifiche a `jaas.config` prima di provare il modulo di esempio.

### Informazioni correlate

[Sviluppo di applicazioni client AMQP](#)

[Creazione e utilizzo di canali AMQP](#)

## Advanced Message Security

---

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

### Panoramica di AMS

Le applicazioni IBM MQ possono utilizzare 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.

#### Informazioni correlate

[Codici di ritorno GSKit utilizzati nei AMS messaggi](#)

### Comportamento modificato da IBM WebSphere MQ 7.0.1

Advanced Message Security è diventato un componente in IBM MQ da IBM WebSphere MQ 7.5. Alcuni aspetti della funzione AMS sono stati modificati, il che potrebbe influire sulle applicazioni, sugli script di gestione o sulle procedure di gestione.


Esaminare attentamente il seguente elenco di modifiche prima di aggiornare i gestori code a IBM WebSphere MQ 7.5 o versioni successive. Decidere se è necessario pianificare modifiche alle applicazioni, agli script e alle procedure esistenti prima di avviare la migrazione dei sistemi:

- L'installazione di AMS fa parte del processo di installazione IBM MQ.
- La funzionalità di protezione AMS è abilitata con la relativa installazione e controllata con le politiche di protezione. Non è necessario abilitare gli intercettatori per consentire a AMS di avviare l'intercettazione dei dati.
- AMS in IBM MQ non richiede l'utilizzo del comando **cfgmq** come nella versione autonoma di Advanced Message Security.

### Caratteristiche e funzioni di Advanced Message Security

Advanced Message Security espande IBM 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 posizionati 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.

AMS fornisce le seguenti funzioni:

- Protegge le transazioni sensibili o di valore elevato elaborate da IBM 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 IBM MQ.
-  Da IBM MQ 9.0.0 Fix Pack 8, è stato aggiunto un controllo al codice della libreria IBM MQ eseguito all'interno del programma applicativo del cliente. Il controllo viene eseguito all'inizio dell'inizializzazione per leggere il valore della variabile di ambiente `AMQ_AMS_FIPS_OFF` e, se è impostato su un qualsiasi valore, il codice GSKit verrà eseguito in modalità non FIPS in tale applicazione.



## Gestione degli errori

IBM MQ 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 Advanced Message Security .
- 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.

Advanced Message Security utilizza SYSTEM.PROTECTION.ERROR.QUEUE come coda di gestione errori. Tutti i messaggi inseriti da IBM MQ AMS nel SISTEMA SYSTEM.PROTECTION.ERROR.QUEUE è preceduto da un'intestazione MQDLH.

L'amministratore IBM MQ può definire anche il SISTEMA SYSTEM.PROTECTION.ERROR.QUEUE come coda alias che punta ad un'altra coda.

## **Qualità di protezione disponibili con AMS**

Le tre qualità di protezione per Advanced Message Security sono Integrity, Privacy, da IBM MQ 9.0, Confidentiality.

La protezione Integrity è fornita dalla firma digitale, che fornisce la garanzia su chi ha creato il messaggio e che il messaggio non è stato modificato o manomesso.

La protezione Privacy è fornita da una combinazione di firma digitale e crittografia.

La crittografia garantisce che i dati del messaggio siano visualizzabili solo per il destinatario o i destinatari previsti. Anche se i destinatari non autorizzati ottengono una copia dei dati del messaggio crittografati, non sono in grado di visualizzare i dati effettivi del messaggio.

La protezione Confidentiality è fornita solo dalla cifratura.

## Effetto sulle prestazioni

AMS utilizza una combinazione di routine crittografiche simmetriche e asimmetriche per fornire la firma digitale e la crittografia. Poiché le operazioni della chiave simmetrica sono molto veloci rispetto alle operazioni della chiave asimmetrica, che utilizzano intensamente la CPU, questo a sua volta può avere un impatto significativo sui costi di protezione di un gran numero di messaggi con AMS.

### Routine crittografiche asimmetriche

Ad esempio, quando si immette un messaggio firmato, l'hash del messaggio viene firmato utilizzando un'operazione di chiave asimmetrica.

Quando si riceve un messaggio firmato, viene utilizzata un'ulteriore operazione di chiave asimmetrica per verificare l'hash firmato.

Pertanto, sono richieste almeno due operazioni di chiave asimmetrica per messaggio per firmare e verificare i dati del messaggio.

## Routine crittografiche asimmetriche e simmetriche

Quando si colloca un messaggio codificato, viene generata una chiave simmetrica e quindi codificata utilizzando un'operazione di chiave asimmetrica per ciascun destinatario previsto del messaggio.

I dati del messaggio vengono quindi codificati con la chiave simmetrica. Quando si ottiene il messaggio codificato, il destinatario previsto deve utilizzare un'operazione di chiave asimmetrica per rilevare la chiave simmetrica in uso per il messaggio.

Tutte e tre le qualità di protezione, quindi, contengono diversi elementi delle operazioni chiave asimmetriche ad alta intensità di CPU, che avranno un impatto significativo sulla massima velocità di messaggistica raggiungibile per le applicazioni che inseriscono e ricevono messaggi.

## Riutilizzo chiave

Le politiche Confidentiality, tuttavia, consentono il riutilizzo della chiave simmetrica su una sequenza di messaggi.


È possibile utilizzare questo approccio per ridurre in modo significativo i costi relativi alla crittografia di un numero di messaggi destinati allo stesso destinatario o destinatario.

Ad esempio, quando si inserendo 10 messaggi crittografati nella stessa serie di destinatari, viene generata una chiave simmetrica e quindi crittografata per il primo messaggio, utilizzando un'operazione di chiave asimmetrica per ogni destinatario previsto del messaggio.

In base ai limiti controllati dalla politica, la chiave simmetrica codificata può essere riutilizzata da messaggi successivi destinati agli stessi destinatari. Un'applicazione che sta ricevendo messaggi codificati può applicare la stessa ottimizzazione, in quanto l'applicazione può rilevare quando una chiave simmetrica non è stata modificata ed evitare il costo del richiamo della chiave simmetrica.

In questo esempio, il 90% delle operazioni della chiave asimmetrica può essere evitato sia dalle applicazioni di immissione che di acquisizione riutilizzando la stessa chiave.

Per ulteriori informazioni su come utilizzare il riutilizzo delle chiavi, consultare:

- Comando MQSC [SET POLICY](#)
- Comando di controllo [setmqspl](#)
-  IBM i comando [SETMQ](#)

## Concetti chiave in AMS

Scopri i concetti chiave in Advanced Message Security per comprendere come funziona lo strumento e come gestirlo in modo efficace.

### **Infrastruttura chiave pubblica e Advanced Message Security**

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 ai 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 549

Per fornire una protezione crittografica trasparente alle applicazioni IBM MQ , Advanced Message Security utilizza il file keystore, in cui vengono memorizzati i certificati della chiave pubblica e una chiave privata. Su z/OS, viene utilizzato un file di chiavi SAF invece di un file keystore.

## Certificati digitali in AMS

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 sia nelle applicazioni Java che native

## Informazioni correlate

[Crittografia](#)

### **Gestore Autorizzazione Oggetto**

Su Multiplatforms, OAM (Object Authority Manager) è un componente del servizio di autorizzazione fornito con i prodotti IBM MQ .

L'accesso alle entità Advanced Message Security è controllato tramite i gruppi di utenti IBM 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 (Message Queue Interface). 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](#)

[Panoramica su Message Queue Interface](#)

## **Tecnologia supportata da Advanced Message Security**

Advanced Message Security dipende da diversi componenti tecnologici per fornire un'infrastruttura di sicurezza.

Advanced Message Security supporta le seguenti API (application programming interface) IBM MQ :

- Message Queue Interface (MQI)
- IBM MQ Java Message Service (JMS) 1.0.2 e 1.1.
- IBM MQ Classi base per Java
- Classi IBM MQ per .NET in modalità non gestita

**Nota:** Advanced Message Security supporta le autorità di certificazione X.509 .

### **Limitazioni note**

Informazioni sulle limitazioni di Advanced Message Security.

- Le seguenti opzioni IBM MQ non sono supportate o hanno limitazioni:

- Pubblicazione / sottoscrizione.

**Nota:** Uno dei principali vantaggi di un modello di messaggistica di pubblicazione / sottoscrizione su point - to - point è che le applicazioni di invio e ricezione non hanno bisogno di sapere nulla l'uno sull'altro per i dati da inviare e ricevere. Questo vantaggio viene negato dall'utilizzo delle politiche Advanced Message Security che devono definire i destinatari previsti o i firmatari autorizzati. È possibile per un'applicazione pubblicare in un argomento tramite una definizione di coda alias protetta da una politica, è anche possibile per un'applicazione di sottoscrizione ottenere i messaggi da una coda protetta dalla politica. Non è possibile assegnare una politica direttamente a una stringa di argomenti, le politiche possono essere assegnate solo a definizioni di coda.

- Conversione dati del canale.

**Nota:** Il payload protetto di un messaggio protetto Advanced Message Security viene trasmesso utilizzando il formato binario, ciò garantisce che la conversione dei dati su un canale tra le applicazioni non invalidi il digest del messaggio. Le applicazioni che richiamano i messaggi da una coda protetta della politica devono richiedere la conversione dei dati, la conversione del payload protetto verrà tentata dopo che i messaggi sono stati correttamente verificati e non protetti.

- Elenchi di distribuzione.

**Nota:** Le politiche Advanced Message Security possono essere utilizzate quando si proteggono le applicazioni che inserendo i messaggi negli elenchi di distribuzione, purché ciascuna coda di destinazione nell'elenco abbia una politica identica definita. Se vengono identificate delle politiche incongruenti quando un'applicazione apre un elenco di distribuzione, l'operazione di apertura avrà esito negativo e verrà restituito un errore di sicurezza all'applicazione.

- Segmentazione del messaggio dell'applicazione.

**Nota:** La dimensione dei messaggi protetti dalla politica aumenterà e non è possibile per le applicazioni specificare in modo accurato i limiti del segmento di un messaggio.

- Le applicazioni senza thread su piattaforme HP-UX non sono supportate.
- Le applicazioni che utilizzano IBM MQ classes for .NET in modalità gestita (connessioni client) non sono supportate.

**Nota:** L'intercettazione MCA può essere utilizzata per consentire ai client non supportati di utilizzare AMS.

- Il client Message Service per le applicazioni .NET (XMS) in modalità gestita non è supportato.

**Nota:** L'intercettazione MCA può essere utilizzata per consentire ai client non supportati di utilizzare AMS.

- Le code IBM MQ elaborate dal bridge IMS non sono supportate.

**Nota:** AMS è supportato sulle code bridge CICS . È necessario utilizzare lo stesso ID utente per MQPUT (codifica) e MQGET (decodifica) sulle code CICS Bridge.

- Gli utenti devono evitare di inserire più di un certificato con lo stesso DN (Distinguished Name) in un singolo file keystore, poiché la scelta del certificato da utilizzare quando si protegge un messaggio non è definita.
- AMS non è supportato in JMS se la proprietà **WMQ\_PROVIDER\_VERSION** è impostata su 6.
- L'interceptor AMS non è supportato per i canali AMQP o MQTT.

## Scenari utente

Familiarizzare con gli scenari possibili per comprendere quali obiettivi di business è possibile raggiungere con Advanced Message Security.

### **Guida rapida per AMS su piattaforme Windows**

Utilizzare questa guida per configurare rapidamente Advanced Message Security per fornire la sicurezza dei messaggi su 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 MQ per i sistemi Windows](#) .

Per informazioni sull'utilizzo del comando **setmqenv** per inizializzare l'ambiente corrente in modo che i comandi IBM MQ appropriati possano essere ubicati ed eseguiti dal sistema operativo, consultare [setmqenv \(set IBM MQ environment\)](#).

### 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 intercettatori per firmare e codificare i messaggi nel momento in cui entrano nell'infrastruttura IBM MQ tramite l'interfaccia standard IBM MQ . La configurazione di base viene eseguita in IBM MQ e viene configurata nei seguenti passi.

È possibile utilizzare IBM 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 presenti in C:\Programmi\IBM\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 `HOME` 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
```



**Attenzione:** IBM MQ ottimizza le prestazioni memorizzando nella cache le politiche in modo da non dover sfogliare i record per i dettagli della politica sul `SYSTEM.PROTECTION.POLICY.QUEUE` in tutti i casi.

IBM MQ non memorizza nella cache tutte le politiche disponibili. Se è presente un numero elevato di politiche, IBM MQ memorizza nella cache un numero limitato di politiche. Quindi, se il gestore code ha un numero basso di politiche definite, non è necessario fornire l'opzione di ricerca al SISTEMA `SYSTEM.PROTECTION.POLICY.QUEUE`.

Tuttavia, è necessario concedere l'autorizzazione di ricerca a questa coda, nel caso in cui sia definito un numero elevato di politiche o se si utilizzano vecchi client. Il SISTEMA `SYSTEM.PROTECTION.ERROR.QUEUE` viene utilizzato per inserire i messaggi di errore generati dal codice AMS. L'autorizzazione all'inserimento rispetto a questa coda viene controllata solo quando si tenta di inserire un messaggio di errore nella coda. L'autorizzazione di inserimento

rispetto alla coda non viene controllata quando si tenta di inserire o richiamare un messaggio da una coda protetta AMS.

## Risultati

Gli utenti vengono ora creati e le autorizzazioni richieste vengono loro concesse.

## Operazioni successive

Per verificare se i passi sono stati eseguiti correttamente, utilizzare gli esempi `amqsput` e `amqsget` come descritto nella sezione [“7. Verifica della configurazione”](#) a pagina 530.

### 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 bind client, è necessario creare un keystore JKS e i certificati utilizzando il comando **keytool**, che fa parte del JRE (per ulteriori dettagli, consultare [“Guida rapida per AMS con client Java”](#) a pagina 537). Per tutte le altre lingue e per le applicazioni Java che utilizzano i bind locali, i passi in questa guida sono corretti.

## Procedura

1. Utilizzare la GUI di IBM Key Management (`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 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 = 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" (con uno spazio alla fine), ad esempio, sono riconosciuti come etichette di due diversi certificati. Tuttavia, per evitare confusione, è meglio non utilizzare spazi nel nome dell'etichetta.
- Sono presenti i seguenti formati keystore: CMS (Cryptographic Message Syntax), JKS (Java Keystore) e JCEKS (Java Cryptographic Extension Keystore). Per ulteriori informazioni, fare riferimento a [“Struttura del file di configurazione del keystore \(keystore.conf\) per AMS”](#) a pagina 550.
- %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 549.
- 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 passwd -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 passwd -label Alice_Cert -file alice_public.arm
```

## 3. Ripetere i passi per bob:

```
runmqakm -cert -extract -db "C:/Documents and Settings/bob/AMS/bobkey.kdb" -pw passwd -label Bob_Cert -target bob_public.arm
```

```
runmqakm -cert -add -db "C:/Documents and Settings/alice/AMS/alicekey.kdb" -pw passwd -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 passwd -label Alice_Cert
```

```
runmqakm -cert -details -db "C:/Documents and Settings/alice/AMS/alicekey.kdb" -pw passwd -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 `setmqsp1`. Fare riferimento a `setmqsp1` 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:

```
setmqsp1 -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 certificato dell'utente ricettivo dal database delle chiavi.

## Operazioni successive

Per verificare la politica definita, immettere il seguente comando:

```
dspmqsp1 -m QM_VERIFY_AMS
```

Per stampare i dettagli della politica come una serie di comandi `setmqsp1`, utilizzare 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 AMS su UNIX

Utilizzare questa guida per configurare rapidamente Advanced Message Security per fornire la sicurezza dei messaggi su 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 GSK (Global Security Kit)
- MQ Advanced Message Security

Fare riferimento ai seguenti argomenti per i nomi componente su ciascuna piattaforma specifica:

- [Componenti IBM MQ per sistemi Linux .](#)
- [Componenti IBM MQ per sistemi HP-UX .](#)
- [Componenti IBM MQ per sistemi AIX .](#)
- [Componenti IBM 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 intercettatori per firmare e codificare i messaggi nel momento in cui entrano nell'infrastruttura IBM MQ tramite l'interfaccia standard IBM MQ . La configurazione di base viene eseguita in IBM MQ e viene configurata nei seguenti passi.

È possibile utilizzare IBM 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_INSTALLATION_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
```



**Attenzione:** IBM MQ ottimizza le prestazioni memorizzando nella cache le politiche in modo da non dover sfogliare i record per i dettagli della politica sul SYSTEM.PROTECTION.POLICY.QUEUE in tutti i casi.

IBM MQ non memorizza nella cache tutte le politiche disponibili. Se è presente un numero elevato di politiche, IBM MQ memorizza nella cache un numero limitato di politiche. Quindi, se il gestore code ha un numero basso di politiche definite, non è necessario fornire l'opzione di ricerca al SISTEMA SYSTEM.PROTECTION.POLICY.QUEUE.

Tuttavia, è necessario concedere l'autorizzazione di ricerca a questa coda, nel caso in cui sia definito un numero elevato di politiche o se si utilizzano vecchi client. Il SISTEMA SYSTEM.PROTECTION.ERROR.QUEUE viene utilizzato per inserire i messaggi di errore generati dal codice AMS. L'autorizzazione all'inserimento rispetto a questa coda viene controllata solo quando si tenta di inserire un messaggio di errore nella coda. L'autorizzazione di inserimento

rispetto alla coda non viene controllata quando si tenta di inserire o richiamare un messaggio da una coda protetta AMS.

## 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 i passi sono stati eseguiti correttamente, utilizzare gli esempi `amqsput` e `amqsget` come descritto nella sezione [“8. Verifica della codifica”](#) a pagina 536.

### 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 bind client, è necessario creare un keystore JKS e i certificati utilizzando il comando **keytool**, che fa parte del JRE (per ulteriori dettagli, consultare [“Guida rapida per AMS con client Java”](#) a pagina 537). Per tutte le altre lingue e per le applicazioni Java che utilizzano i bind locali, i passi in questa guida sono corretti.

## Procedura

1. Creare un nuovo database di chiavi per l'utente `alice`

```
mkdir /home/alice/.mqc -p
runmqakm -keydb -create -db /home/alice/.mqc/alicekey.kdb -pw passwd -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/.mqc/alicekey.kdb
```

3. Crea un certificato che identifica l'utente `alice` da utilizzare nella cifratura

```
runmqakm -cert -create -db /home/alice/.mqc/alicekey.kdb -pw passwd
-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 leggibile 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.
- Sono presenti i seguenti formati *keystore*: CMS (Cryptographic Message Syntax), JKS (Java Keystore) e JCEKS (Java Cryptographic Extension Keystore). Per ulteriori informazioni, fare riferimento a [“Struttura del file di configurazione del keystore \(keystore.conf\) per AMS”](#) a pagina 550.
- *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 549.

### 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/.mq5/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/.mq5/bobkey.kdb -pw passw0rd -label Alice_Cert -file  
alice_public.arm
```

3. Ripetere il passo per bob:

```
runmqakm -cert -extract -db /home/bob/.mq5/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/.mq5/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/.mq5/bobkey.kdb -pw passw0rd -label Alice_Cert  
runmqakm -cert -details -db /home/alice/.mq5/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 `setmqsp1`. Fare riferimento a `setmqsp1` 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:

```
setmqsp1 -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 certificato dell'utente ricettivo dal database delle chiavi.

## Operazioni successive

Per verificare la politica definita, immettere il seguente comando:

```
dspmqsp1 -m QM_VERIFY_AMS
```

Per stampare i dettagli della politica come una serie di comandi `setmqsp1` , utilizzare l'indicatore `-export` . Ciò consente di memorizzare le politiche già definite:

```
dspmqsp1 -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 a`lice`, 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 AMS con client Java

Utilizzare questa guida per configurare rapidamente 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

Assicurarsi di disporre dei componenti appropriati installati come descritto nella **Guida di avvio rapido** ([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 intercettatori per firmare e codificare i messaggi nel momento in cui entrano nell'infrastruttura IBM MQ tramite l'interfaccia standard IBM MQ. La configurazione di base viene eseguita in IBM 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** visualizza i dettagli su TEST.Q:

```
DISPLAY Q(TEST.Q)
```

### 2. Creazione e autorizzazione degli utenti

## Informazioni su questa attività

In questo scenario sono presenti 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 correttamente le politiche di protezione definite in questo scenario, a tali 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 +browse
```

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
```



**Attenzione:** IBM MQ ottimizza le prestazioni memorizzando nella cache le politiche in modo da non dover sfogliare i record per i dettagli della politica sul SYSTEM.PROTECTION.POLICY.QUEUE in tutti i casi.

IBM MQ non memorizza nella cache tutte le politiche disponibili. Se è presente un numero elevato di politiche, IBM MQ memorizza nella cache un numero limitato di politiche. Quindi, se il gestore code ha un numero basso di politiche definite, non è necessario fornire l'opzione di ricerca al SISTEMA SYSTEM.PROTECTION.POLICY.QUEUE.

Tuttavia, è necessario concedere l'autorizzazione di ricerca a questa coda, nel caso in cui sia definito un numero elevato di politiche o se si utilizzano vecchi client. Il SISTEMA SYSTEM.PROTECTION.ERROR.QUEUE viene utilizzato per inserire i messaggi di errore generati

dal codice AMS. L'autorizzazione all'inserimento rispetto a questa coda viene controllata solo quando si tenta di inserire un messaggio di errore nella coda. L'autorizzazione di inserimento rispetto alla coda non viene controllata quando si tenta di inserire o richiamare un messaggio da una coda protetta AMS.

## Risultati

Gli utenti vengono ora creati e le autorizzazioni richieste vengono loro concesse.

## Operazioni successive

Per verificare se i passi sono stati eseguiti correttamente, utilizzare gli esempi `JmsProducer` e `JmsConsumer` come descritto nella sezione [“7. Verifica della configurazione”](#) a pagina 542.

### 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, utilizziamo le applicazioni di esempio scritte in Java che si collegano utilizzando i collegamenti client. Se si pianifica di utilizzare le applicazioni Java utilizzando i bind locali o le applicazioni C, è necessario creare un keystore CMS e i certificati 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/.mqsc`. È possibile crearla nella stessa directory utilizzata dalla **Guida rapida** ([Windows](#) o [UNIX](#)) per la propria piattaforma.

**Nota:** Questa directory viene 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 invece 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. Su UNIX, accertarsi che il keystore 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* è il seguente:

```
JKS.keystore = keystore-dir/keystore
JKS.certificate = Alice_Java_Cert
JKS.encrypted = no
JKS.keystore_pass = password
JKS.key_pass = password
JKS.provider = IBMJCE
```

Per questo scenario, il contenuto di *keystore.conf* per *bob* è il seguente:

```
JKS.keystore = keystore-dir/keystore
JKS.certificate = Bob_Java_Cert
JKS.encrypted = no
JKS.keystore_pass = password
JKS.key_pass = password
JKS.provider = IBMJCE
```

#### Nota:

- Il percorso del file *keystore* deve essere fornito senza estensione file.
- Se si dispone già di un file *keystore.conf* perché sono state seguite le istruzioni della Guida di avvio rapido ([Windows](#) o [UNIX](#)), è possibile modificare il file esistente per aggiungere queste righe.
- Per ulteriori informazioni, consultare [“Struttura del file di configurazione del keystore \(keystore.conf\) per AMS”](#) a pagina 550.

### 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 di comando IBM GSKit **strmqikm** (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 IBM 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 `setmqsp1` . Fare riferimento a [setmqsp1](#) 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:

```
setmqsp1 -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 certificato dell'utente ricettivo dal database delle chiavi.

## Operazioni successive

Per verificare la politica definita, immettere il seguente comando:

```
dspmqsp1 -m QM_VERIFY_AMS
```

Per stampare i dettagli della politica come una serie di comandi `setmqsp1` , l'indicatore `-export` . Ciò consente di memorizzare le politiche già definite:

```
dspmqsp1 -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 IBM 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 rapida** ([Windows](#) o [UNIX](#)) per la propria piattaforma, per dettagli sull'esecuzione di programmi con 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 da IBM MQ classes for 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 a una serie di politiche per la coda remota. Ad esempio, per una politica di codifica, il messaggio viene codificato prima di essere passato a IBM MQ per gestirlo. Dopo che Advanced Message Security ha elaborato il messaggio inserito in una coda remota, IBM 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 politica impostata 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 utilizzando IBM Integration Bus**

Advanced Message Security può proteggere i messaggi in un'infrastruttura in cui è installato IBM Integration Bus o WebSphere Message Broker 8.0.0.1 (o successivo). È necessario comprendere la natura di entrambi i prodotti prima di applicare la protezione nell'ambiente IBM Integration Bus .

### 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 IBM Integration Bus, è possibile consentire a IBM Integration Bus di elaborare i messaggi senza conoscerne il contenuto ([Scenario 1](#)) oppure renderlo un utente autorizzato a ricevere e inviare messaggi ([Scenario 2](#)).

*Scenario 1 - Integration Bus non può visualizzare il contenuto del messaggio*

### Prima di iniziare

IBM Integration Bus deve essere 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),<sup>1</sup>(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 IBM Integration Bus. Advanced Message Security viene utilizzato solo da *alice*, *bob* e *cecil*. Non è necessario installarlo o configurarlo per IBM Integration Bus.

IBM Integration Bus riceve il messaggio protetto dalla coda `alias` non protetta per evitare qualsiasi tentativo di decodificare il 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 IBM Integration Bus e arriva nella coda di destinazione non modificata. Quindi 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). IBM Integration Bus inserisce il messaggio instradato in un `alias` non protetto. I destinatari richiameranno il messaggio da una coda di output protetta in cui 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
  - 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.

---

<sup>1</sup> cecil

Eeguire 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*.

Eeguire 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 protezione per la coda QIN su una configurazione idonea. Utilizzare la configurazione identica per le code QBOB, QCECIL e QDEF .

```
setmqsp1 -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.

```
dspmqsp1 -m QMgrName -p AIN
```

8. In IBM Integration Bus creare un flusso di messaggi per instradare i messaggi in arrivo nella 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 `bob` `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 IBM Integration Bus .
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 formato protetto da IBM Integration Bus dalla coda alias AIN . IBM Integration Bus decide dove instradare il messaggio leggendo la proprietà `routeTo` che, come tutte le proprietà, non è codificata. IBM Integration Bus inserisce 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 - Integration Bus può visualizzare il contenuto del messaggio*

## Informazioni su questa attività

In questo scenario, un gruppo di individui può inviare messaggi a IBM Integration Bus. Un altro gruppo è autorizzato a ricevere messaggi creati da IBM Integration Bus. La trasmissione tra le parti e IBM Integration Bus non può essere interrotta.

Tenere presente che IBM Integration Bus 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 IBM Integration Bus è considerato una parte autorizzata a leggere o firmare il payload del messaggio, devi configurare Advanced Message Security per l'utente che avvia il servizio IBM Integration Bus . Tenere presente che non è necessariamente lo stesso utente che inserisce / richiama i messaggi nelle code né l'utente che crea e distribuisce applicazioni IBM Integration Bus .

## Procedura

1. Configura *alice*, *bob*, *cecil* e *dave* e l'utente del servizio IBM Integration Bus , 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. Fornisci i certificati *alice*, *bob*, *cecil* e *dave* all'utente del servizio IBM Integration Bus .

Eseguire questa operazione estraendo ciascuno dei certificati che identificano *alice*, *bob*, *cecil* e *dave* nei file esterni, quindi aggiungendo i certificati estratti al keystore IBM Integration Bus . È importante utilizzare il metodo descritto in **Attività 5. Condivisione dei certificati** nella **Guida di avvio rapido** ([Windows](#) o [UNIX](#)).

3. Fornisci il certificato dell'utente del servizio IBM Integration Bus 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 IBM Integration Bus per crittografare correttamente i messaggi. L'utente del servizio IBM Integration Bus ha bisogno dei certificati *alice* e *bob* per verificare gli autori dei messaggi. L'utente del servizio IBM Integration Bus ha bisogno dei certificati *cecil* e *dave* per crittografare i messaggi. *cecil* e *dave* hanno bisogno del certificato dell'utente del servizio IBM Integration Bus per verificare se il messaggio proviene da IBM Integration Bus.

4. Definire una coda locale denominata IN e definire la politica di sicurezza con *alice* e *bob* specificati come autori e l'utente del servizio per il IBM Integration Bus specificato come destinatario:

```
setmqsp1 -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 sicurezza con l'utente del servizio per IBM Integration Bus specificato come autore e *cecil* e *dave* specificati come destinatari:

```
setmqsp1 -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 IBM Integration Bus 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 IBM Integration Bus .
8. L'esecuzione come utente *alice* o *bob* inserisce un messaggio nella coda IN utilizzando l'applicazione di esempio **amqsp1**.
9. L'esecuzione come utente *cecil* o *dave* richiama il messaggio dalla coda OUT utilizzando l'applicazione di esempio **amqsgt**.

## Risultati

I messaggi inviati da *alice* o *bob* alla coda di input IN sono crittografati consentendo solo a IBM Integration Bus di leggerli. IBM Integration Bus accetta solo i messaggi da *alice* e *bob* e rifiuta tutti gli altri. I messaggi accettati vengono elaborati in modo appropriato, quindi firmati e crittografati 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 IBM Integration Bus vengono rifiutati.

## Utilizzo di AMS con Managed File Transfer

Questo scenario spiega come configurare Advanced Message Security per fornire la privacy dei messaggi per i dati inviati tramite Managed File Transfer.

## Prima di iniziare

Verificare che il componente Advanced Message Security sia installato nell'installazione IBM MQ che ospita le code utilizzate da Managed File Transfer che si desidera proteggere.

Se gli agent Managed File Transfer si collegano in modalità bind, verificare che il componente GSKit sia installato anche sull'installazione locale.

## Informazioni su questa attività

Quando il trasferimento dei dati tra due agent Managed File Transfer viene interrotto, è possibile che i dati riservati rimangano non protetti sulle code IBM MQ sottostanti utilizzate per gestire il trasferimento. Questo scenario spiega come configurare e utilizzare Advanced Message Security per proteggere tali dati sulle code Managed File Transfer .

In questo scenario si considera una topologia semplice che comprende una macchina con due code Managed File Transfer e due agenti AGENT1 e AGENT2, che condividono un singolo gestore code, come descritto nello scenario [Panoramica sullo scenario](#). 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 Managed File Transfer Agent . 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 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 di avvio rapido** ( [Windows](#) o [UNIX](#) ) per la propria piattaforma.
- Se gli agent 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 AMS con client Java”](#) a pagina 537.

## 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 **setmqsp1** . 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 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` .

```
setmqsp1 -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. Verificare che l'utente che esegue il processo Managed File Transfer Agent disponga dell'accesso per sfogliare la coda della politica di sistema e inserire 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 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.

## Installazione Advanced Message Security

Installare il componente Advanced Message Security su varie piattaforme.

## Informazioni su questa attività

Per le procedure di installazione complete, vedere [Installazione di Advanced Message Security](#).

### Informazioni correlate

[disinstallazione Advanced Message Security](#)

z/OS

## Controllo su z/OS

Advanced Message Security per z/OS fornisce un mezzo per il controllo facoltativo delle operazioni MQI sulle code protette dalla politica. Quando abilitata, i record di controllo SMF (IBM System Management Facility) vengono creati per l'esito positivo e negativo di queste operazioni sulle code protette dalla politica. Le operazioni controllate includono MQPUT, MQPUT1e MQGET.

Il controllo è disabilitato per impostazione predefinita, tuttavia, è possibile attivare il controllo configurando `_AMS_SMF_TYPE` e `_AMS_SMF_AUDIT` nel file Language Environment `_CEE_ENVFILE` configurato per lo spazio di indirizzo AMS. Per ulteriori informazioni, consultare [Attività 24: Creare procedure per Advanced Message Security](#). La variabile `_AMS_SMF_TYPE` viene usata per designare il tipo di record SMF ed è un numero compreso tra 128 e 255. Un tipo di record SMF di 180 è usuale, tuttavia non è obbligatorio. Il controllo è disabilitato specificando il valore 0. La variabile `_AMS_SMF_AUDIT` configura se i record di controllo vengono creati per le operazioni MQI riuscite, le operazioni MQI non riuscite o entrambe. Le opzioni di controllo possono anche essere modificate in modo dinamico mentre AMS è attivo utilizzando i comandi operatore. Per ulteriori informazioni, consultare [Funzionamento di Advanced Message Security](#).

Il record SMF viene definito utilizzando sottotipi, con il sottotipo 1 che rappresenta un evento di verifica generale. Il record SMF contiene tutti i dati rilevanti per la richiesta in fase di elaborazione.

Il record SMF viene mappato dalla macro CSQ0KSMF (notare lo zero nel nome della macro), fornita nella libreria di destinazione SCSQMACS. Se si stanno scrivendo programmi di riduzione dei dati per i dati SMF, è possibile includere questa macro di mappatura per facilitare lo sviluppo e la personalizzazione delle routine di post - elaborazione SMF.

Nei record SMF prodotti da Advanced Message Security per z/OS, i dati sono organizzati in sezioni. Il record è composto da:

- un'intestazione SMF standard
- un'estensione di intestazione definita da Advanced Message Security per z/OS
- una sezione del prodotto
- una sezione dati

La sezione del prodotto del record SMF è presente sempre nei record prodotti da Advanced Message Security per z/OS. La sezione dei dati varia in base al tipo secondario. Attualmente, viene definito un sottotipo e quindi viene utilizzata una singola sezione di dati.

SMF è descritto nel manuale z/OS System Management Facilities (SA22-7630). I tipi di record validi sono descritti nel membro SMFPRMxx del dataset PARMLIB del sistema. Per ulteriori informazioni, consultare la documentazione SMF.

## Generatore report di verifica Advanced Message Security (CSQ0USMF)

Advanced Message Security (AMS) per z/OS fornisce uno strumento del generatore di report di verifica denominato CSQ0USMF fornito nella libreria SCSQAUTH di installazione. Il JCL di esempio per eseguire il programma di utilità CSQ0USMF denominato CSQ40RSM viene fornito nella libreria di installazione SCSQPROC.

Come esempio, il seguente JCL esegue il dump dei record SMF di tipo 180 da un dataset SMF e li trasferisce a un dataset di destinazione.

```
//IFAUDUMP EXEC PGM=IFASMFDP
//INDD1 DD DSN=SYSn.MANn.syst,DISP=SHR
//OUTDD1 DD DSN=your.target.dataset,DISP=SHR
//SYSPRINT DD SYSOUT=*
```

```
//SYSIN DD *
INDD(INDD1,OPTIONS(DUMP))
OUTDD(OUTDD1,TYPE(180))
/*
```

È necessario verificare i nomi dei dataset SMF effettivi utilizzati dall'installazione. Il dataset di destinazione per i record di cui è stato eseguito il dump deve avere un formato record VBS e una lunghezza record di 32760.

**Nota:** Se vengono utilizzati i flussi di log SMF, è necessario utilizzare il programma IFASMF DL per eseguire il dump di un flusso di log in un dataset sequenziale. Consultare [Tipo di elaborazione 116 record SMF](#) per un esempio di JCL utilizzato.

Il dataset di destinazione può quindi essere utilizzato come input per l'utility CSQ0USMF per produrre un report di verifica AMS. Ad esempio:

```
//STEP1 EXEC PGM=CSQ0USMF,
// PARM=('/' -SMFTYPE 180 -M qmgr')
//STEPLIB DD DSN=thlqual.SCSQANLE,DISP=SHR
// DD DSN=thlqual.SCSQAUTH,DISP=SHR
//SMFIN DD DSN=your.target.dataset,DISP=SHR
//
```

Il programma CSQ0USMF accetta due parametri facoltativi, elencati nella seguente tabella:

<i>Tabella 86. Parametri facoltativi CSQ0USMF</i>		
<b>Parametro</b>	<b>Valore</b>	<b>Descrizione</b>
PE SMFT	nnn	Il tipo di record SMF applicabile al report di controllo. Il programma CSQ0USMF utilizza solo record SMF che corrispondono al valore SMFTYPE durante la generazione del report. Se non si specifica SMFTYPE, viene utilizzato il valore predefinito 180.
M	qmgr	Il nome del gestore code WMQ applicabile al report di controllo. Se non si specifica il parametro -M, il report di controllo includerà tutti i record di controllo per tutti i gestori code rappresentati nel dataset SMFIN.




## Utilizzo di keystore e certificati

Per fornire una protezione crittografica trasparente alle applicazioni IBM MQ , Advanced Message Security utilizza il file keystore, in cui vengono memorizzati i certificati della chiave pubblica e una chiave privata. Su z/OS, viene utilizzato un file di chiavi SAF invece di un file keystore.

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.

Su Windows e UNIX , l'ubicazione del keystore viene fornita nel file di configurazione del keystore, che è `keystore.conf` per default. Ogni utente Advanced Message Security deve avere il file di configurazione keystore che fa riferimento 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 UNIX e IBM i: \$HOME/.mqsc/keystore.conf
-  Su Windows: %HOMEDRIVE%%HOMEPATH%\mqsc\keystore.conf

Se si sta utilizzando un percorso e un nome file keystore specificati, è necessario utilizzare i seguenti comandi

- Per Java `java -DMQS_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 in AMS” a pagina 576

I DN (distinguished name) del mittente identificano gli utenti autorizzati a inserire messaggi in una coda.

“DN (Distinguished Name) destinatario in AMS” a pagina 577

Il DN (distinguished name) del destinatario identifica gli utenti autorizzati a richiamare i messaggi da una coda.

## Struttura del file di configurazione del keystore (keystore.conf) per AMS

Il file di configurazione del keystore (keystore.conf) punta Advanced Message Security all'ubicazione del keystore appropriato.

Ognuno dei seguenti tipi di file di configurazione ha un prefisso:

### cms

Certificate Management System, le voci di configurazione hanno come prefisso: cms.

### PKCS#11

Standard di crittografia della chiave pubblica #11, le voci di configurazione hanno come prefisso: pkcs11.

### PEM

Privacy Enhanced Mail format, le voci di configurazione hanno come prefisso: pem.

### JKS

Java KeyStore, le voci di configurazione hanno come prefisso: jks.

### JCEKS

Java Crittografia KeyStore, le voci di configurazione hanno come prefisso: jceks.

### JCEKCFKS

Java Keyring RACF di crittografia KeyStore, le voci di configurazione hanno come prefisso: jceracfks.

### 

**Importante:** Da IBM MQ 9.0 i valori JCEKS.provider e JKS.provider vengono ignorati. Viene utilizzato il provider Bouncy Castle, insieme a qualsiasi fornitura JCE/JCE fornita dal JRE in uso. Per ulteriori informazioni, consultare “Supporto per JRE nonIBM con AMS” a pagina 553.

Strutture di esempio per i keystore:

cms

```
cms.keystore = /dir/keystore_file
cms.certificate = certificate_label
```

## PKCS#11

```
pkcs11.library = dir\cryptoki.dll
pkcs11.certificate = certificatelabel
pkcs11.token = tokenlabel
pkcs11.token_pin = tokenpin
pkcs11.secondary_keystore = dir\signers
```

## PEM

```
pem.private = /dir/keystore_file_private_key
pem.public = /dir/keystore_file_public_keys
pem.password = password
```

## Java JKS

```
jks.keystore = dir/Keystore
jks.certificate = certificate_label
jks.encrypted = no
jks.keystore_pass = password
jks.key_pass = password
jks.provider = IBMJCE
```

## Java JCEKS

```
jceks.keystore = dir/Keystore
jceks.certificate = certificate_label
jceks.encrypted = no
jceks.keystore_pass = password
jceks.key_pass = password
jceks.provider = IBMJCE
```

## V 9.0.5 Java JCERACFKS

```
jceracfks.keystore = safkeyring://user/keyring
jceracfks.certificate = certificate_label
```

Tabella 87. Riepilogo dei parametri necessari per ciascun tipo di file di configurazione

Parametri	Tipo di file di configurazione			
	V 9.0.5 Java (JKS, JCEKS e JCERACFKS)	PEM	PKCS#11	cms
keystore	✓			✓
private		✓		
public		✓		
password		✓		
library			✓	
certificate	✓		✓	✓
token			✓	
token_pin			✓	

Tabella 87. Riepilogo dei parametri necessari per ciascun tipo di file di configurazione (Continua)

Parametri	Tipo di file di configurazione			
	V 9.0.5 Java (JKS, JCEKS e JCERACFKS)	PEM	PKCS#11	cms
secondary_keystore			✓	
encrypted	✓			
keystore_pass	✓			
provider	✓			

I parametri del file di configurazione sono definiti come segue:

### keystore

Solo configurazione CMS e Java . Percorso del file keystore per la configurazione CMS, JKS e JCEKS.

V 9.0.5 z/OS MQ,Adv.VUE URI per il keyring RACF per la configurazione JCERACFKS.

### Importante:

- Il percorso del file keystore non deve includere l'estensione file.
- V 9.0.5 z/OS MQ,Adv.VUE L'URI del keyring RACF deve essere nel formato:

```
safkeyring://user/keyring
```

dove:

- *user* è l'ID utente che possiede il keyring
- *keyring* è il nome del keyring.

### private

Solo configurazione PEM. Nome file di un file che contiene la chiave privata e il certificato in formato PEM.

### public

Solo configurazione PEM. Nome file di un file che contiene certificati pubblici attendibili in formato PEM.

### password

Solo configurazione PEM. Password utilizzata per decodificare una chiave privata codificata.

### library

PKCS#11 . Nome percorso della libreria PKCS#11 .

### certificate

Solo configurazione CMS, PKCS#11 e Java . Etichetta del certificato.

### token

PKCS#11 . Etichetta token.

### token\_pin

PKCS#11 . PIN per sbloccare il token.

### secondary\_keystore

PKCS#11 . Nome percorso del keystore CMS, fornito senza l'estensione .kdb , che contiene i certificati di ancoraggio (certificati root) richiesti dai certificati memorizzati sul token PKCS #11 . Il keystore secondario può contenere anche i certificati intermedi nella catena di attendibilità, nonché i certificati dei destinatari definiti nella politica di sicurezza della privacy. Questo keystore CMS deve



essere accompagnato da un file stash che deve essere ubicato nella stessa directory del keystore secondario.

### encrypted

Solo configurazione Java . Stato della password.

### keystore\_pass

Solo configurazione Java . Password per il file keystore.

#### Nota:

- Per il keystore CMS, 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.
- **Importante:** La memorizzazione delle password in formato testo semplice rappresenta un rischio per la sicurezza.



**Nota:** Ignorato per jceracfks poiché l'accesso non è controllato da una password.

### key\_pass

Solo configurazione Java . Password per la chiave privata dell'utente.

**Importante:** La memorizzazione delle password in formato testo semplice rappresenta un rischio per la sicurezza.



**Nota:** Ignorato per jceracfks poiché l'accesso non è controllato da una password.

### provider

Solo configurazione Java . Il fornitore della sicurezza Java che implementa gli algoritmi crittografici richiesti dal certificato keystore.

**Importante:** Le informazioni memorizzate nel keystore sono fondamentali per il flusso sicuro di dati inviati utilizzando IBM MQ. Gli amministratori della sicurezza devono prestare particolare attenzione quando assegnano le autorizzazioni file a questi file.

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 = passw0rd
jceks.key_pass = passw0rd
jceks.provider = IBMJCE
```

### Attività correlate

[“Protezione delle password in Java” a pagina 567](#)

La memorizzazione di password del keystore e della chiave privata come testo semplice rappresenta un rischio per la sicurezza, quindi Advanced Message Security fornisce uno strumento che può codificarle utilizzando una chiave utente, disponibile nel file keystore.

### Supporto per JRE nonIBM con AMS

Da IBM MQ 9.0, AMS è supportato in JRE nonIBM in client Java .

Advanced Message Security implementa [CMS \(Cryptographic Message Syntax\)](#). La sintassi CMS viene utilizzata per firmare, digest, autenticare o codificare in modo digitale il contenuto arbitrario del messaggio.

Nei rilasci precedenti del prodotto, il supporto Advanced Message Security in IBM MQ classes for Java e IBM MQ classes for JMS dipendeva dal supporto CMS fornito specificamente dall'implementazione IBM

di JCE ( Java Cryptography Extensions). A causa di questa restrizione, la funzione era disponibile solo quando si utilizza un Java runtime environment (JRE) che includeva il provider Java JCE.

È importante sottolineare che il supporto su piattaforme come Solaris richiedeva un JRE ibrido. Vale a dire, il JRE standard per la piattaforma con ulteriori elementi forniti da IBM; in particolare, il provider IBM JCE era richiesto piuttosto che quello fornito dal JRE standard per la piattaforma.

Da IBM MQ 9.0, il supporto AMS in IBM MQ classes for Java e IBM MQ classes for JMS è stato modificato e ora utilizza i package Bouncy Castle open-source per supportare CMS. Ciò significa che queste classi possono ora supportare l'operazione AMS durante l'esecuzione con JRE nonIBM .

Il file JAR di Bouncy Castle necessario è incluso come parte del package di installazione di IBM MQ classes for Java e IBM MQ classes for JMS .

Per IBM MQ 9.0.x Continuous Delivery e per Long Term Support per IBM MQ 9.0.0 Fix Pack 5 e precedenti, i file JAR Bouncy Castle utilizzati sono i seguenti:

#### **Il barattolo "provider", che è fondamentale per le operazioni di Bouncy Castle.**

Questo jar è denominato `bcprov-jdk15on-VER.jar` dove "VER" è un numero di versione di 3 cifre, che rappresenta il numero di versione di Bouncy Castle senza punti incorporati. Ad esempio, il jar del provider per Bouncy Castle versione 1.5.2 è `bcprov-jdk15on-152.jar`.

#### **Il jar "PKIX", che contiene il supporto per le operazioni CMS utilizzate da AMS .**

Si chiama `bcpkix-jdk15on-VER.jar` dove "VER" rappresenta lo stesso numero di versione a 3 cifre del jar del fornitore.

La versione dei file jar di Bouncy Castle varia a seconda della release IBM MQ .

- In IBM MQ 9.0.3 e versioni precedenti, VER è 152.

- **V 9.0.4** Da IBM MQ 9.0.4, VER è 157.

**LTS** Per Long Term Support per IBM MQ 9.0.0 Fix Pack 6 e versioni successive, i file JAR di Bouncy Castle utilizzati sono i seguenti:

#### **V 9.0.0.6 Da IBM MQ 9.0.0 Fix Pack 6: il file JAR del provider, che è fondamentale per le operazioni di Bouncy Castle.**

Questo file JAR è denominato `bcprov-jdk15on.jar`.

#### **V 9.0.0.6 Da IBM MQ 9.0.0 Fix Pack 6: il file JAR "PKIX", che contiene il supporto per le operazioni CMS utilizzate da Advanced Message Security.**

Questo file JAR è denominato `bcpkix-jdk15on.jar`.

#### **V 9.0.0.12 Da IBM MQ 9.0.0 Fix Pack 12: il file JAR "UTIL", che contiene le classi utilizzate dalle altre API Bouncy Castle.**

Questo file JAR è denominato `bcutil-jdk15on.jar`.

Bouncy Castle 1.69 ha introdotto un nuovo file JAR, `bcutil-VER.jar`. Il file JAR "BCUTIL" è una raccolta di classi che non devono necessariamente trovarsi nel file JAR del provider JCE, ma sono utilizzate dalle altre API Bouncy Castle.

Le classi modificate sono state verificate con JRE IBM e JRE Oracle . È anche probabile che vengano eseguiti correttamente in qualsiasi JRE J2SE-compliant . Tuttavia, è necessario notare le seguenti dipendenze:

- Non ci sono modifiche alla configurazione AMS
- Le classi Bouncy Castle sono utilizzate solo per le operazioni CMS. Tutte le altre operazioni relative alla sicurezza, ad esempio l'accesso al keystore, la codifica effettiva dei dati e il calcolo dei checksum della firma, utilizzano la funzionalità fornita da JRE.

**Importante:** Per questo motivo, il JRE utilizzato deve includere un'implementazione del fornitore JCE.

- Per utilizzare alcuni algoritmi di codifica *strong* , potrebbe essere necessario installare i file della politica *unrestricted* per l'implementazione JRE

Fare riferimento alla documentazione JRE per ulteriori dettagli.

- Se è stata abilitata la sicurezza Java :
  - Aggiungere `java.security.SecurityPermissioninsertProvider.BC` all'applicazione, in modo che le classi Bouncy Castle possano essere utilizzate come provider di sicurezza.
  - Concedere `java.security.AllPermission` ai file JAR di Bouncy Castle, che sono:

```

V9.0.0.12 mq_install_dir/java/lib/bcutil-jdk15on.jar
mq_install_dir/java/lib/bcpkix-jdk15on.jar
mq_install_dir/java/lib/bcprov-jdk15on.jar

```

### Informazioni correlate

[Cosa è installato per le classi IBM MQ per JMS](#)

[Cosa è installato per le classi IBM MQ per Java](#)

## Multi Intercettazione MCA (Message Channel Agent)

L'intercettazione MCA consente a un gestore code in esecuzione in IBM MQ di abilitare in modo selettivo le politiche da applicare per i canali di connessione server.

L'intercettazione MCA consente ai client esterni a AMS di essere ancora connessi a un gestore code e ai relativi messaggi di essere crittografati e decrittografati.

L'intercettazione MCA è progettata per fornire la funzione AMS quando AMS non può essere abilitata sul client. Notare che l'utilizzo dell'intercettazione MCA e di un client abilitato a AMSporta a una doppia protezione dei messaggi che potrebbe essere problematica per le applicazioni di ricezione. Per ulteriori informazioni, consultare [“Disabilitazione di Advanced Message Security sul client”](#) a pagina 557.

**Nota:** Gli intercettatori MCA non sono supportati per i canali AMQP o MQTT.

### 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 `.mq` 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 AMS , consultare [“Utilizzo di keystore e certificati”](#) a pagina 549.

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.

Consultare [“Esempio di intercettazione MCA Advanced Message Security”](#) a pagina 555 per un esempio di impostazione dell'intercettazione MCA.



**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 i

Se la propria azienda utilizza IBM i ed è stata selezionata una CA (Certificate Authority) commerciale per firmare il certificato, il Certificate Manager digitale crea una richiesta di certificato in formato PEM (Privacy - Enhanced Mail). È necessario inoltrare la richiesta alla CA scelta.

A tale scopo, è necessario utilizzare il seguente comando per selezionare il certificato corretto per il canale specificato in `channelname`:

```
pem.certificate.channel.channelname
```

### Esempio di intercettazione MCA Advanced Message Security

Un'attività di esempio su come impostare un'intercettazione MCA 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.

Se la propria azienda utilizza IBM i ed è stata selezionata una CA (Certificate Authority) commerciale per firmare il certificato, il Certificate Manager digitale crea una richiesta di certificato in formato PEM (Privacy - Enhanced Mail). È necessario inoltrare la richiesta alla CA scelta.

## 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 7.5, AMS era un prodotto aggiuntivo che doveva essere installato separatamente e gli intercettatori configurati per proteggere le applicazioni. Da IBM WebSphere MQ 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 IBM WebSphere MQ 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/mq90
KEYSTORELOC=/home/testusr/ssl/ams1
mkdir -p $KEYSTORELOC
chmod -R 777 $KEYSTORELOC
chown -R mqm:mqm $KEYSTORELOC
export PATH=$PATH:$INSTLOC/gskit8/bin
echo "PATH = $PATH"
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$INSTLOC/gskit8/lib64

gsk8capicmd_64 -keydb -create -db $KEYSTORELOC/alicekey.kdb -pw passw0rd -stash
gsk8capicmd_64 -keydb -create -db $KEYSTORELOC/bobkey.kdb -pw passw0rd -stash
gsk8capicmd_64 -cert -create -db $KEYSTORELOC/alicekey.kdb -pw passw0rd
-label alice_cert -dn "cn=alice,0=IBM,c=IN" -default_cert yes
gsk8capicmd_64 -cert -create -db $KEYSTORELOC/bobkey.kdb -pw passw0rd
-label bob_cert -dn "cn=bob,0=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 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 MQ 9.0 :

```
/opt/mq90/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 MQ 9.0 :

```
/opt/mqm/samp/bin/amqsgetc TESTQ TESTQMGR
```

### Attività correlate

“Guida rapida per AMS con client Java” a pagina 537

Utilizzare questa guida per configurare rapidamente 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 524

Informazioni sulle limitazioni di Advanced Message Security.

## Disabilitazione di Advanced Message Security sul client

È necessario disabilitare IBM MQ Advanced Message Security (AMS) se si sta utilizzando un client IBM WebSphere MQ 7.5 o 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 IBM WebSphere MQ 7.5, IBM MQ Advanced Message Security (AMS) viene abilitato automaticamente in un client IBM MQ e 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 IBM WebSphere MQ 7.1, non hanno AMS abilitato e ciò causa la notifica dell'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 nel modo seguente:

- Per i client Java , in uno dei seguenti modi:
  - Impostando una variabile di ambiente AMQ\_DISABLE\_CLIENT\_AMS.
  - Impostando la proprietà di sistema Java com.ibm.mq.cfg.AMQ\_DISABLE\_CLIENT\_AMS.
  - Utilizzando la proprietà AMS DisableClient, nella stanza **Security** del file mqclient.ini .
- Per i client C, in uno dei modi riportati di seguito:
  - Impostando una variabile di ambiente MQS\_DISABLE\_ALL\_INTERCEPT.
  - Utilizzando la proprietà AMS DisableClient, nella stanza **Security** del file mqclient.ini .

**Nota:** In IBM WebSphere MQ 7.5, è possibile utilizzare anche la variabile di ambiente AMQ\_DISABLE\_CLIENT\_AMS. per i client C. Da IBM MQ 8.0, non è più possibile utilizzare la variabile di ambiente AMQ\_DISABLE\_CLIENT\_AMS per i client C. È necessario utilizzare invece la variabile di ambiente MQS\_DISABLE\_ALL\_INTERCEPT.

## Procedura

- Per disabilitare AMS sul client, utilizzare una delle seguenti opzioni:

### 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 IBM WebSphere MQ 7.5, o un client IBM MQ classes for JMS o IBM MQ classes for Java successivo.

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
```

### Java proprietà di sistema com.ibm.mq.cfg.AMQ\_DISABLE\_CLIENT\_AMS

Per i clienti IBM MQ classes for JMS e IBM MQ classes for Java , è possibile impostare la Java proprietà di sistema 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.

### Variabile di ambiente MQS\_DISABLE\_ALL\_INTERCEPT

È necessario impostare questa variabile se si utilizza IBM MQ 8.0 o versioni successive con i client nativi ed è necessario disabilitare AMS sul client.

Creare la variabile di ambiente MQS\_DISABLE\_ALL\_INTERCEPT e impostarla su TRUE nell'ambiente in cui è in esecuzione il client. Ad esempio:

```
export MQS_DISABLE_ALL_INTERCEPT =TRUE
```

È possibile utilizzare la variabile di ambiente MQS\_DISABLE\_ALL\_INTERCEPT solo per i client C. Per i client Java , è necessario utilizzare invece la variabile di ambiente AMQ\_DISABLE\_CLIENT\_AMS.

### Proprietà `DisableClientAMS` nel file `mqclient.ini`

È possibile utilizzare questa opzione per client IBM MQ classes for JMS e IBM MQ classes for Java e per client C.

Aggiungere il nome proprietà `DisableClientAMS` nella stanza **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 di apertura delle code protette AMS, consultare [“Problemi nell'apertura di code protette quando si usa JMS”](#) a pagina 597.

### Concetti correlati

[“Intercettazione MCA \(Message Channel Agent\)”](#) a pagina 555

L'intercettazione MCA consente a un gestore code in esecuzione in IBM MQ di abilitare in modo selettivo le politiche da applicare per i canali di connessione server.

### Informazioni correlate

[Il file di configurazione IBM MQ classes for JMS](#)

[Configurazione di un client utilizzando un file di configurazione](#)

## 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 MQ”](#) a pagina 42.

## 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 579

Le politiche di protezione dati Advanced Message Security implicano una qualità di protezione (QOP).

## Metodi di convalida dei certificati in AMS

È possibile utilizzare 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.

AMS consente di verificare la validità di un certificato utilizzando OCSP (Online Certificate Status Protocol) o CRL (Certificate Revocation List).

AMS può essere configurato per il controllo OCSP o CRL o per entrambi. Se entrambi i metodi sono abilitati, per motivi di prestazioni AMS utilizza prima OCSP per lo stato di revoca. Se lo stato di revoca di un certificato non è determinato dopo il controllo OCSP, AMS utilizza il controllo CRL.

Si noti che il controllo OCSP e CRL sono abilitati per impostazione predefinita.

### Concetti correlati

“OCSP (Online Certificate Status Protocol) in AMS” a pagina 560

OCSP (Online Certificate Status Protocol) determina se un certificato è stato revocato e, pertanto, consente di determinare se il certificato può essere considerato attendibile. OCSP è abilitato per impostazione predefinita.

“CRL (Certificate Revocation List) in AMS” a pagina 562

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 (Online Certificate Status Protocol) in AMS

OCSP (Online Certificate Status Protocol) determina se un certificato è stato revocato e, pertanto, consente di determinare se il certificato può essere considerato attendibile. OCSP è abilitato per impostazione predefinita.

OCSP non è supportato sui sistemi IBM i.

*Abilitazione del controllo OCSP negli intercettatori nativi di Advanced Message Security*

Il check-in OCSP (Online Certificate Status Protocol) in Advanced Message Security è abilitato per impostazione predefinita, in base alle informazioni contenute nei certificati utilizzati.

### Procedura

Aggiungere le seguenti opzioni al file di configurazione del keystore:

**Nota:** Tutte le stanze OCSP sono facoltative e possono essere specificate in modo indipendente.

Opzione	Descrizione
<code>ocsp.enable=off</code>	Abilitare il controllo OCSP se il certificato che si sta controllando ha un'estensione AIA (Authority Info Access) con un metodo di accesso PKIX_AD_OCSP contenente un URI in cui si trova il responder OCSP.  Valori possibili: <code>on</code> o <code>off</code> .
<code>ocsp.url=responder_URL</code>	L'indirizzo URL del responder OCSP. Se questa opzione viene omessa, il controllo OCSP non AIA è disabilitato.
<code>ocsp.http.proxy.host=OCSP_proxy</code>	L'indirizzo URL del server proxy OCSP. Se questa opzione viene omessa, non viene utilizzato un proxy per i controlli dei certificati online non AIA.
<code>ocsp.http.proxy.port=port_number</code>	Il numero di porta del server proxy OCSP. Se questa opzione viene omessa, viene utilizzata la porta predefinita 8080.
<code>ocsp.nonce.generation=on/off</code>	Generare un parametro nonce durante le query su OCSP.  Il valore predefinito è <code>off</code> .



Opzione	Descrizione
<code>ocsp.nonce.check=on/off</code>	Verificare il parametro nonce dopo la ricezione di una risposta da OCSP. Il valore predefinito è <code>off</code> .
<code>ocsp.nonce.size=8</code>	Dimensione nonce in byte.
<code>ocsp.http.get=on/off</code>	Specificare HTTP GET come metodo di richiesta. Se questa opzione è impostata su <code>off</code> , viene utilizzato HTTP POST. Il valore predefinito è <code>off</code> .
<code>ocsp.max_response_size=20480</code>	Dimensione massima della risposta dal responder OCSP fornita in byte.
<code>ocsp.cache_size=100</code>	Abilitare la memorizzazione della risposta OCSP interna nella cache e impostare il limite per il numero di voci della cache.
<code>ocsp.timeout=30</code>	Tempo di attesa per una risposta del server, in secondi, dopo il quale Advanced Message Security va in timeout.
<code>ocsp.unknown=ACCEPT</code>	Definisce il funzionamento quando non è possibile raggiungere un server OCSP entro un periodo di timeout. I valori possibili sono: <ul style="list-style-type: none"> <li>• <code>ACCEPT</code> Consente il certificato</li> <li>• <code>WARN</code> Consente il certificato e registra un'avvertenza</li> <li>• <code>REJECT</code> Impedisce l'uso del certificato e registra un errore</li> </ul>

#### Abilitazione del check-in OCSP Java in AMS

Per abilitare il controllo 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 check-in OCSP in Advanced Message Security:

#### Utilizzo di `java.security`

Verificare se il proprio certificato contiene un'estensione del certificato AIA (Authority Information Access).

### 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 utilizza 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 List) in AMS

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 dell'autorità di certificazione (CA) fino a un ancoraggio sicuro. 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. AMS verifica il percorso del certificato utilizzando un algoritmo di convalida PKIX. Quando il concatenamento viene creato e verificato, 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 accodata al certificato, AMS verifica se **digitalSignature** o **nonRepudiation** sono impostati. In caso contrario, il `MQRC_SECURITY_ERROR` viene riportato e registrato. Successivamente, 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 AMS. Se non viene trovata alcuna configurazione relativa al CRL nel file di configurazione del keystore, AMS non esegue alcun controllo di validità CRL. Per ciascun certificato CA, AMS interroga LDAP per i CRL utilizzando i DN (Distinguished Name) di una CA per trovare il proprio 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 keystore in modo che Advanced Message Security possa scaricare i CLR dal server LDAP (Lightweight Directory Access Protocol).

### Informazioni su questa attività

 L'abilitazione della convalida del certificato e del supporto dell'elenco di revoca del certificato negli intercettatori nativi non è supportata per Advanced Message Security su IBM i.

### Procedura

Aggiungere le seguenti opzioni al file di configurazione:

**Nota:** Tutte le stanze CRL sono facoltative e possono essere specificate indipendentemente.

Opzione	Descrizione
<code>crl.ldap.host=host_name</code>	Nome host del server LDAP.
<code>crl.ldap.port=port_number</code>	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 di AMS Java si connette correttamente a un server LDAP, non tenta di scaricare i CRL dai restanti server forniti.
<code>crl.cdp=off</code>	Utilizzare questa opzione per controllare o utilizzare le estensioni CRLDistributionPoints nei certificati.
<code>crl.ldap.version=3</code>	Numero di versione protocollo LDAP. Valori possibili: 2 o 3.
<code>crl.ldap.user=cn=username</code>	Accedere al server LDAP. Se questo valore non è specificato, gli attributi CRL in LDAP devono essere leggibili
<code>crl.ldap.pass=password</code>	Password per il server LDAP.
<code>crl.ldap.cache_lifetime=0</code>	Durata della cache LDAP in secondi. Valori possibili: 0-86400.
<code>crl.ldap.cache_size=50</code>	Dimensione cache LDAP. Questa opzione può essere specificata solo se il valore <code>crl.ldap.cache_lifetime</code> è maggiore di 0.
<code>crl.http.proxy.host=some.host.com</code>	Porta del server proxy Http per il recupero CRL CDP.
<code>crl.http.proxy.port=8080</code>	Numero di porta del server proxy Http.
<code>crl.http.max_response_size=204800</code>	La dimensione massima di CRL, in byte, che può essere richiamata da un server HTTP accettato da GSKit.
<code>crl.http.timeout=30</code>	Tempo di attesa per una risposta del server, in secondi, dopo il quale AMS va in timeout.

Opzione	Descrizione
<code>cr1.http.cache_size=0</code>	Dimensione cache HTTP, in byte.
<code>cr1.unknown=ACCEPT</code>	Definisce il comportamento quando non è possibile raggiungere un server CRL entro un periodo di timeout. I valori possibili sono: <ul style="list-style-type: none"> <li>• ACCEPT Consente il certificato</li> <li>• WARN Consente il certificato e registra un'avvertenza</li> <li>• REJECT Impedisce l'uso del certificato e registra un errore</li> </ul>

#### Abilitazione del supporto CRL (Certificate Revocation List) in Java in AMS

Per abilitare il supporto CRL in Advanced Message Security, è necessario modificare il file di configurazione del keystore per consentire a 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
<code>cr1.ldap.host=host_name</code>	Nome host LDAP.
<code>cr1.ldap.port=port_number</code>	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 di AMS Java si connette correttamente a un server LDAP, non tenta di scaricare i CRL dai restanti server forniti. Java non utilizza valori <code>cr1.ldap.user</code> e <code>cr1.ldap.worldp.pass</code> . Non utilizza un utente e una password durante la connessione a un server LDAP. Di conseguenza, gli attributi CRL in LDAP devono essere leggibili.
<code>cr1.cdp=on/off</code>	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
<code>com.ibm.security.enableCRLDP</code>	<p>Questa proprietà assume i seguenti valori: <code>true</code>, <code>false</code>.</p> <p>Se è impostato su <code>true</code>, durante il controllo della revoca del certificato, i CRL vengono individuati utilizzando l'URL dall'estensione dei punti di distribuzione CRL del certificato.</p> <p>Se è impostato su <code>false</code> o non è impostato, il controllo di CRL utilizzando l'estensione dei punti di distribuzione CRL è disabilitato.</p>
<code>ibm.security.certpath.ldap.cache.lifetime</code>	<p>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.</p>
<code>com.ibm.security.enableAIAEXT</code>	<p>Questa proprietà assume i seguenti valori: <code>true</code>, <code>false</code>.</p> <p>Se è impostato su <code>true</code>, 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.</p> <p>Se è impostato su <code>false</code> o non è impostato, non vengono creati ulteriori oggetti LDAPCertStore .</p>

### **Abilitazione dei CRL (Certificate Revocation List) su z/OS**

Advanced Message Security supporta la verifica CRL (Certificate Revocation List) dei certificati digitali utilizzati per proteggere i messaggi di dati

#### **Informazioni su questa attività**

Quando abilitato, Advanced Message Security convaliderà i certificati dei destinatari quando i messaggi vengono inseriti in una coda protetta dalla privacy e convaliderà i certificati dei mittenti quando i messaggi vengono richiamati da una coda protetta (integrità o riservatezza). La convalida in questo caso include la verifica che i relativi certificati non sono registrati in un CRL pertinente.

Advanced Message Security utilizza i servizi IBM System SSL per convalidare i certificati del mittente e del destinatario. La documentazione dettagliata relativa alla convalida del certificato SSL di sistema è disponibile nel manuale *z/OS Cryptographic Services System Secure Sockets Layer Programming (SC24-5901)*.

Per abilitare il controllo CRL, specificare l'ubicazione di un file di configurazione CRL tramite CRLFILE DD nel JCL dell'attività avviata per lo spazio di indirizzo AMS. Un file di configurazione CRL di esempio che può essere personalizzato viene fornito in *thlqual.SCSQPROC (CSQ40CRL)*. Le impostazioni consentite in questo file sono le seguenti:

Tabella 88. Variabili di configurazione CRL Advanced Message Security

Variabile	Valori validi	Descrizione
crl.ldap.host[.n]	hostname -or - hostname: porta	L'ipaddr / nome host del server LDAP che ospita i CRL dei certificati dell'emittente. Se non si specifica un numero di porta per il server LDAP, viene utilizzato il numero di porta specificato da crl.ldap.port .
crl.ldap.port	porta	Il numero di porta TCP/IP del server LDAP.
crl.ldap.user	utente_ldap	Il nome utente LDAP da utilizzare durante la connessione al server LDAP.
crl.ldap.pass	password_ldap	La password LDAP associata a crl.ldap.user.

È possibile specificare più porte e nomi host del server LDAP nel modo seguente:

```
crl.ldap.host.1 = hostname -or hostname:port
crl.ldap.host.2 = hostname -or hostname:port
crl.ldap.host.3 = hostname -or hostname:port
```

È possibile specificare fino a 10 nomi host. Se non si specifica un numero di porta per i propri server LDAP, viene utilizzato il numero di porta specificato da crl.ldap.port . Ciascun server LDAP deve utilizzare la stessa combinazione crl.ldap.user/password per accedere.

Quando viene specificato CRLFILE DD, la configurazione viene caricata durante l'inizializzazione dello spazio di indirizzi Advanced Message Security e il controllo CRL è abilitato. Se il CRLFILE DD non è specificato, il file di configurazione CRL non è disponibile o non è valido, il controllo CRL è disabilitato.

AMS esegue un controllo CRL utilizzando i servizi di convalida del certificato SSL del sistema IBM come segue:

Tabella 89. Controlli CRL Advanced Message Security

Operazione	QoP (Quality of protection)	Certificati controllati
PUT	Riservatezza	Destinatario/i
GET	Integrità / Privacy	Mittente

Se un'operazione del messaggio non riesce, un controllo CRL Advanced Message Security effettua le seguenti azioni:

Tabella 90. Comportamento dell'errore di controllo CRL Advanced Message Security

Operazione	Errore di controllo CRL
PUT	Il messaggio non viene inserito nella coda di destinazione. All'applicazione viene restituito un codice di completamento MQCC_FAILED e un codice motivo MQRC_SECURITY_ERROR.

Tabella 90. Comportamento dell'errore di controllo CRL Advanced Message Security (Continua)

Operazione	Errore di controllo CRL
GET	Il messaggio viene rimosso dalla coda di destinazione e spostato nella coda di errori di protezione del sistema. All'applicazione viene restituito un codice di completamento MQCC_FAILED e un codice motivo MQRC_SECURITY_ERROR.

AMS per z/OS utilizza i servizi IBM System SSL per convalidare i certificati, che includono CRL e il controllo dell'affidabilità. IBM System SSL fornisce la variabile di ambiente GSK\_CRL\_SECURITY\_LEVEL per moderare l'operazione di controllo CRL. Ad esempio:

```
GSK_CRL_SECURITY_LEVEL=MEDIUM
```

Questa variabile è documentata nel manuale z/OS Cryptographic Services System Secure Sockets Layer Programming. Le assegnazioni valide includono:

- LOW - La convalida del certificato non avrà esito negativo se non è possibile contattare il server LDAP.
- MEDIUM - La convalida del certificato richiede che il server LDAP sia contattabile, ma non richiede la definizione di un CRL.
- HIGH - La convalida del certificato richiede che il server LDAP sia contattabile e che sia definito un CRL.

Il valore predefinito di SSL di sistema IBM è MEDIUM. È possibile impostare questa variabile nel file di configurazione specificato tramite ENVARS DD nel JCL del task avviato per lo spazio di indirizzo AMS. Un file di configurazione della variabile di ambiente di esempio è fornito in *thlqual.SCSQPROC* (CSQ40ENV).

**Nota:** È responsabilità degli amministratori garantire che i relativi servizi LDAP siano disponibili e gestire le voci CRL per le autorità di certificazione.

## Protezione delle password in Java

La memorizzazione di password del keystore e della chiave privata come testo semplice rappresenta un rischio per la sicurezza, quindi Advanced Message Security fornisce uno strumento che può codificarle 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.es.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 549.

#### Esempio

Di seguito è riportato un esempio di tale output:

```
#Fri Jul 30 15:20:29 CEST 2010
jceks.key_pass=MMXh997n5Z0r8uRlJmc5qity9MN2CggGBMKCDxDbn1AyPk1vdgTsOLG6X3C1YT7oDzwaqZF1OR4t\i\nm
Zsc7JGAX8nqqxLnAucdGn0Nwo6xnjZB1n501YGo12k/
PhaQHhFXKMAU9dKg0f8dj0tCA01X4ETe\i\nfY19LBUt2wk87uM7dSs\=
jceks.keystore_pass=0IdeayBnSCfLG4cFuxEVrk6SYyAsdSPpDqgPf16s9s1M04cqZjNbhgjoA2EXonudHZHH+4s2drvQ
\i\nCUvQgu9GuaBMJK2F20jtHJJ1Y4BVeLW2c2okgawo/
W2J1AdUYKkJ0raYTkDouLaTYTQeuLyG0xIl\i\niD2si1xUCxhYvvyhbbY\=
jceks.encrypted=yes
```

## Utilizzo dei certificati su z/OS

### Informazioni su questa attività

Advanced Message Security implementa due livelli di protezione: integrità e privacy. Con il livello di integrità, i messaggi vengono firmati utilizzando la chiave privata del mittente (l'applicazione che esegue MQPUT). L'integrità fornisce il rilevamento della modifica del messaggio, ma il testo del messaggio stesso non è codificato.

Con il livello di privacy, il messaggio non solo è firmato, ma è anche crittografato. Il messaggio viene codificato utilizzando una chiave simmetrica e un algoritmo specificato nella politica Advanced Message Security pertinente. La chiave simmetrica stessa viene codificata con la chiave pubblica di ciascun destinatario (l'applicazione che esegue MQGET). Le chiavi pubbliche sono associate ai certificati memorizzati nei keyring.

Quando un messaggio protetto con privacy viene eliminato dalla coda da un'applicazione del destinatario che esegue un MQGET, il messaggio deve essere decodificato. Poiché è stata codificata utilizzando la chiave pubblica del destinatario, deve essere decodificata utilizzando la chiave privata del destinatario trovata in un keyring.

## Uso dei portachiavi SAF

Advanced Message Security utilizza i servizi key ring SAF esistenti per definire e gestire i certificati necessari per la firma e la codifica. I prodotti di sicurezza funzionalmente equivalenti a RACF possono essere utilizzati al posto di RACF se forniscono lo stesso livello di supporto.

L'uso efficiente dei portachiavi può ridurre l'amministrazione necessaria per gestire i certificati.

Dopo che un certificato è stato generato (o importato), deve essere collegato a un keyring per diventare accessibile. Lo stesso certificato può essere collegato a più di un keyring.

Advanced Message Security utilizza due serie di portachiavi. Una serie è costituita da keyring di proprietà dei singoli ID utente che originano o ricevono messaggi. Ogni keyring contiene la chiave privata associata al certificato dell'ID utente proprietario. La chiave privata di ogni certificato viene utilizzata per firmare i messaggi per le code protette dall'integrità o dalla privacy. Viene utilizzato anche per decodificare i messaggi dalle code protette dalla riservatezza quando si ricevono i messaggi.



L'altra serie è in realtà un singolo keyring associato allo spazio di indirizzo AMS. Per la protezione dell'integrità o della privacy, contiene la catena di certificati CA di firma necessari per convalidare la firma e il certificato di firma del messaggio del creatore del messaggio.

Quando si utilizza la protezione della privacy, questo keyring contiene anche i certificati dei destinatari del messaggio. Le chiavi pubbliche in questi certificati vengono utilizzate per codificare la chiave simmetrica utilizzata per codificare i dati del messaggio quando il messaggio è stato inserito nella coda protetta. Quando questi messaggi vengono richiamati, la chiave privata dei destinatari rilevanti viene utilizzata per decodificare la chiave simmetrica che viene quindi utilizzata per decodificare i dati del messaggio.

Advanced Message Security utilizza un nome keyring di **drq.ams.keyring** durante la ricerca di certificati e chiavi private. Questo è il caso sia per l'utente che per i keyring dello spazio di indirizzo AMS.

Per un'illustrazione e per un'ulteriore spiegazione dei certificati e del key ring e del loro ruolo nella protezione dei dati, fare riferimento a [Riepilogo delle operazioni correlate ai certificati](#).

La chiave privata utilizzata per la firma e la decodifica può avere qualsiasi etichetta ma deve essere connessa come certificato predefinito.

I certificati digitali e i keyring sono gestiti in RACF principalmente utilizzando il comando RACDCERT.

Per ulteriori informazioni su certificati, etichette e sul comando RACDCERT, consultare *z/OS: Security Server RACF Command Language Reference* e *z/OS: Security Server RACF Security Administrator's Guide*.

## **Autorizzazione dell'accesso al comando RACDCERT**

L'autorizzazione ad utilizzare il comando RACDCERT è un'attività di post - installazione che dovrebbe essere stata completata dal programmatore del sistema z/OS . Questa attività implica la concessione di autorizzazioni pertinenti all'amministratore della sicurezza Advanced Message Security .

Come riepilogo, questi comandi sono richiesti per consentire l'accesso al comando RACDCERT RACF :

```
RDEFINE FACILITY IRR.DIGTCERT.* UACC(NONE)
PERMIT IRR.DIGTCERT.* CLASS(FACILITY) ID( admin ) ACCESS(CONTROL)
SETROPTS RACLIST(FACILITY) REFRESH
```

In questo esempio, *admin* specifica l'ID utente dell'amministratore della sicurezza o qualsiasi utente che si desidera utilizzare il comando RACDCERT.

## **Creazione dei certificati e dei portachiavi**

Questa sezione documenta i passi richiesti per creare i certificati e i file di chiavi necessari per gli utenti z/OS di Advanced Message Security, utilizzando una CA (Certificate Authority) RACF .

## **Risoluzione dei problemi con i certificati quando si utilizzano Advanced Message Security su z/OS**

Se si verificano problemi con il certificato e le voci mancanti nei keystore, è possibile abilitare una traccia GSKIT.

Nel file //ENVARS aggiungere:

```
GSK_TRACE_FILE=/u/... /gsktrace
GSK_TRACE=0xiff
```

Per ulteriori informazioni, vedi [Variabili di ambiente](#) .

Per ogni accesso al keystore, i dati vengono scritti nel file di traccia specificato nel file GSK\_TRACE\_FILE .

Per formattare il file di traccia utilizzare il comando:

```
gsktrace inputtrace file > output_file
```

## Scenario

Uno scenario di un'applicazione di invio e di ricezione viene utilizzato per spiegare i passi richiesti.

Negli esempi che seguono, user1 è il creatore di un messaggio e user2 è il destinatario. L'ID utente dello spazio di indirizzo Advanced Message Security è WMQAMSD.

Tutti i comandi negli esempi riportati di seguito vengono emessi dall'opzione 6 di ISPF dall'ID utente amministrativo admin.

### Definizione di un certificato CA (Certificate Authority) locale

Se si utilizza RACF come CA, è necessario creare un certificato CA (Certificate Authority), se non è già stato fatto. Il seguente comando crea un certificato dell'autorità di certificazione (o del firmatario). Questo esempio crea un certificato denominato AMSCA da utilizzare quando si creano certificati successivi che riflettono l'identità di utenti e applicazioni Advanced Message Security .

Questo comando può essere modificato, in particolare SUBJECTSDN, per riflettere la struttura di denominazione e le convenzioni utilizzate durante l'installazione:

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('AMSCA') O('ibm') C('us'))
KEYUSAGE(CERTSIGN) WITHLABEL('AMSCA')
```

**Nota:** I certificati firmati con questo certificato di autorità di certificazione locale mostrano un emittente di CN=AMSCA, O=ibm, C=us quando elencato con il comando RACDCERT LIST.

### Creazione di un certificato digitale con una chiave privata

È necessario generare un certificato digitale con una chiave privata per ogni utente di Advanced Message Security . Nell'esempio qui mostrato, i comandi RACDCERT sono utilizzati per creare certificati per user1 e user2, firmati con il certificato CA locale identificato dall'etichetta AMSCA.

```
RACDCERT ID(user1) GENCERT SUBJECTSDN(CN('user1') O('ibm') C('us'))
WITHLABEL('user1') SIGNWITH(CERTAUTH LABEL('AMSCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)

RACDCERT ID(user2) GENCERT SUBJECTSDN(CN('user2') O('ibm') C('us'))
WITHLABEL('user2') SIGNWITH(CERTAUTH LABEL('AMSCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)

RACDCERT ID(user1) ALTER (LABEL('user1')) TRUST
RACDCERT ID(user2) ALTER (LABEL('user2')) TRUST
```

Il comando RACDCERT ALTER è richiesto per aggiungere l'attributo TRUST al certificato. Quando un certificato viene creato per la prima volta utilizzando questa procedura, ha un intervallo di date valido diverso dal certificato di firma. Di conseguenza, RACF lo contrassegna come NOTRUST, il che significa che non deve essere utilizzato il certificato. Utilizzare il comando RACDCERT ALTER per impostare l'attributo TRUST.

Gli attributi KEYUSAGE HANDSHAKE, DATAENCRYPT e DOCSIGN devono essere specificati per i certificati utilizzati da Advanced Message Security.

Tabella 91. Valori e indicatori RACDCERT KEYUSAGE	
Valore KEYUSAGE	Serie di indicatori
Handshake	digitalSignature e keyEncipherment
CODIFICADATI	dataEncipherment
SIGN DOCG	nonRepudiation
CERTSIGN	keyCertSign e cRLSign

## Creazione dei keyring RACF

I comandi qui riportati creano un keyring per gli ID utente definiti da RACF user1, user2 e l'utente WMQAMSD dell'attività dello spazio di indirizzo Advanced Message Security. Il nome del key ring è fisso da Advanced Message Security e deve essere codificato come mostrato, senza virgolette. Questo valore è sensibile al maiuscolo/minuscolo.

```
RACDCERT ID(user1) ADDRING(drq.ams.keyring)
RACDCERT ID(user2) ADDRING(drq.ams.keyring)
RACDCERT ID(WMQAMSD) ADDRING(drq.ams.keyring)
```

## Connessione dei certificati ai keyring

Connettere l'utente e i certificati CA ai keyring:

```
RACDCERT ID(WMQAMSD) CONNECT(CERTAUTH LABEL('AMSCA')
RING(drq.ams.keyring))
RACDCERT ID(user1) CONNECT(ID(user1) LABEL('user1')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
RACDCERT ID(user2) CONNECT(ID(user2) LABEL('user2')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
RACDCERT ID(WMQAMSD) CONNECT(ID(user2) LABEL('user2')
RING(drq.ams.keyring) USAGE(SITE))
```

Il certificato contenente la chiave privata utilizzata per la decrittografia deve essere connesso al keyring dell'utente come certificato predefinito.

L'attributo RACDCERT USAGE (SITE) impedisce l'accesso alla chiave privata nel keyring, mentre l'attributo RACDCERT USAGE (PERSONAL) consente l'utilizzo della chiave privata, se esiste. Il certificato User2 deve essere connesso al keyring dello spazio di indirizzo AMS perché la chiave pubblica è necessaria per codificare i messaggi quando vengono inseriti nella coda. USAGE (SITE) limita l'esposizione della chiave privata di user2.

Il certificato CERTAUTH con etichetta AMSCA deve essere connesso al keyring dello spazio di indirizzi Advanced Message Security perché è stato utilizzato per firmare il certificato di user1, che è il mittente del messaggio. Viene utilizzato per convalidare il certificato di firma di user1.

## Verifica key ring

Il keyring dovrebbe apparire come mostrato di seguito, dopo che sono stati immessi tutti i comandi:

```
RACDCERT ID(user1) LISTRING(drq.ams.keyring)
Digital ring information for user USER1:
Ring:
>drq.ams.keyring<:

Certificate Label Name          Cert Owner  USAGE    DEFAULT
-----
user1                          ID(USER1)  PERSONAL YES

RACDCERT ID(user2) LISTRING(drq.ams.keyring)
Digital ring information for user USER2:
Ring:
>drq.ams.keyring<:

Certificate Label Name          Cert Owner  USAGE    DEFAULT
-----
user2                          ID(USER2)  PERSONAL YES

RACDCERT ID(WMQAMSD) LISTRING(drq.ams.keyring)
Digital ring information for user WMQAMSD:
Ring:
>drq.ams.keyring<:

Certificate Label Name          Cert Owner  USAGE    DEFAULT
-----
AMSCA                          CERTAUTH   CERTAUTH NO
```

user2	ID(USER2)	SITE	NO
User1	ID(USER1)	SITE	NO

L'elenco dei singoli certificati mostra anche l'associazione dell'anello.

```
RACDCERT ID(user2) LIST(label('user2'))
Digital certificate information for user USER2:
```

```
***
Label: user2
Certificate ID: 2QfH8Pny9/LzpKKFmfFA
Status: TRUST
Start Date: 2010/05/03 22:59:53
End Date: 2011/05/04 22:59:52
Serial Number:
>15<:
Issuer's Name:
>OU=AMSCA.O=ibm.C=us<:
Subject's Name:
>CN=user2.O=ibm.C=us<:
Key Usage: HANDSHAKE, DATAENCRYPT, DOCSIGN
Private Key Type: Non-ICSF
Private Key Size: 1024
Ring Associations:
Ring Owner: USER2
Ring:
>drq.ams.keyring<:
Ring Owner: WMQAMSD
Ring:
>drq.ams.keyring<:
```

Per migliorare le prestazioni, il contenuto di drq.ams.keyring associato allo spazio di indirizzo AMS viene memorizzato nella cache per la durata dello spazio di indirizzo. Le modifiche a tale keyring non diventano effettive automaticamente. L'amministratore può aggiornare la cache:

- Arresto e riavvio del gestore code.
- Utilizzando il comando z/OS MODIFY:

```
F qmgt AMSM,REFRESH KEYRING
```

### Informazioni correlate

[Funzionamento di Advanced Message Security](#)

### **Riepilogo delle operazioni relative al certificato**

Figura 32 a pagina 573 illustra le relazioni tra le applicazioni di invio e ricezione e i relativi certificati. Lo scenario illustrato implica l'accodamento in remoto tra due gestori code z/OS utilizzando una politica di privacy di protezione dati. In [Figura 32 a pagina 573](#), "AMS" indica "Advanced Message Security".

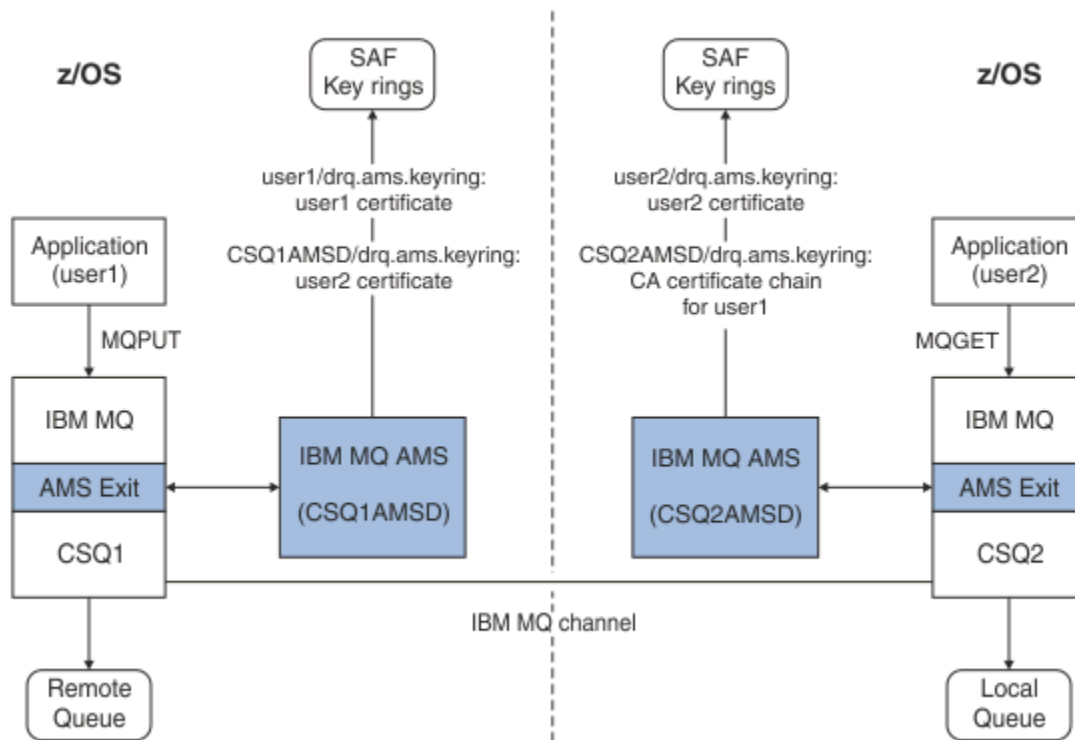


Figura 32. Relazioni applicazione e certificato

In questo diagramma, un'applicazione in esecuzione come 'user1' inserisce un messaggio in una coda remota gestita dal gestore code CSQ1, che deve essere richiamato da un'applicazione in esecuzione come 'user2' da una coda locale gestita dal gestore code CSQ2. Il diagramma presuppone una politica di riservatezza Advanced Message Security, che significa che il messaggio è firmato e codificato.

Advanced Message Security intercetta il messaggio quando si verifica un'operazione di inserimento e utilizza un certificato user2 (memorizzato nel key ring dell'utente dello spazio di indirizzo AMS) per codificare una chiave simmetrica utilizzata per codificare i dati del messaggio.

Notare che il certificato di user2 è connesso al file di chiavi utente dello spazio di indirizzo AMS con l'opzione USAGE (SITE). Ciò significa che l'utente dello spazio di indirizzo AMS può accedere al certificato e alla chiave pubblica, ma non alla chiave privata.

Sul lato ricevente, Advanced Message Security intercetta il get emesso da user2 e utilizza il certificato di user2 per decodificare la chiave simmetrica in modo che possa decodificare i dati del messaggio. Convalida quindi la firma di user1 utilizzando la catena di certificati CA del certificato di user1 memorizzato nel file di chiavi dell'utente dello spazio indirizzi AMS.

Dato questo scenario, ma con una politica di protezione dei dati di integrità, non sarebbero richiesti i certificati per user2.

Per utilizzare Advanced Message Security per accodare i messaggi su code IBM MQ protette con una politica di protezione dei messaggi di privacy o integrità, Advanced Message Security deve avere accesso a questi elementi dati:

- Il certificato X.509 V2 o V3 e la chiave privata per l'utente che accoda il messaggio.
- La catena di certificati utilizzata per firmare i certificati digitali di tutti i firmatari di messaggi.
- Se la politica di protezione dei dati è privacy, il certificato X.509 V2 o V3 dei destinatari previsti. I destinatari previsti sono riportati nella politica Advanced Message Security associata alla coda.

Per i processi e le applicazioni in esecuzione su z/OS, Advanced Message Security deve avere certificati in due posizioni:

- In un keyring gestito da SAF associato all'identità RACF dell'applicazione di invio (l'applicazione che accoda il messaggio protetto) o dell'applicazione di ricezione (se si utilizza la riservatezza).

Il certificato individuato da Advanced Message Security è il certificato predefinito e deve includere la chiave privata. Advanced Message Security assume l'identità utente z/OS dell'applicazione mittente. Cioè, funziona come un surrogato, in modo da poter accedere alla chiave privata dell'utente.

- In un keyring gestito da SAF associato all'utente dello spazio di indirizzo AMS.

Quando si inviano messaggi protetti con riservatezza, questo keyring contiene i certificati di chiave pubblica dei destinatari del messaggio. Quando si ricevono i messaggi, contiene la catena di certificati dell'autorità di certificazione necessari per convalidare la firma del mittente del messaggio.

Gli esempi precedenti mostrati hanno utilizzato RACF come CA locale. Tuttavia, è possibile utilizzare un altro provider PKI (Certificate Authority) durante l'installazione. Se si intende utilizzare un altro prodotto PKI, tenere presente che la chiave privata e il certificato devono essere importati in un file di chiavi associato agli ID utente z/OS RACF che originano i messaggi IBM MQ protetti da Advanced Message Security.

È possibile utilizzare il comando RACF RACDCERT come meccanismo per generare richieste di certificato, che possono essere esportate e inviate al provider PKI di propria scelta da emettere.

Di seguito è riportato un riepilogo delle operazioni relative al certificato:

1. Richiedere la creazione di un certificato CA, uno in cui RACF è la CA locale. Omettere questa fase se si sta utilizzando un altro provider PKI.
2. Generare certificati utente firmati dalla CA.
3. Crea i file di chiavi per gli utenti e l'ID spazio di indirizzo AMS Advanced Message Security .
4. Collegare il certificato utente al keyring utente con l'attributo predefinito.
5. Connettere i certificati dei destinatari al keyring utente dello spazio di indirizzo Advanced Message Security AMS utilizzando l'attributo di utilizzo (sito) (questo passo è necessario solo per i certificati utente che saranno i destinatari dei messaggi protetti dalla privacy).
6. Connettere le catene di certificati CA per i mittenti di messaggi al keyring utente spazio di indirizzi AMS Advanced Message Security . Questo passo è necessario solo per le attività AMS che verificheranno le firme del mittente.

## **Configurazione di una PKI residente nonz/OS**

Advanced Message Security per z/OS, utilizza i certificati digitali X.509 V3 nella protezione - elaborazione dei messaggi inseriti o ricevuti dalle code IBM MQ . Advanced Message Security non crea o gestisce il ciclo di vita di questi certificati; tale funzione è fornita da una PKI (Public Key Infrastructure). Gli esempi in questa pubblicazione che illustrano l'utilizzo dei certificati utilizzano z/OS Security Server RACF per soddisfare richieste di certificato.

Se viene utilizzata o meno una PKI residente z/OS o nonz/OS , AMS per z/OS utilizza solo i keyring gestiti da RACF o il suo equivalente. Questi file di chiavi si basano su SAF (Security Authorization Facility) e sono il repository utilizzato da AMS per z/OS per richiamare i certificati per i creatori e i destinatari dei messaggi inseriti o ricevuti dalle code IBM MQ .

Per i messaggi originati da z/OS, che sono protetti dalla politica di integrità o di codifica, il certificato e la chiave privata dell'ID utente di origine devono essere memorizzati in un keyring gestito da SAF associato all'ID utente z/OS del creatore del messaggio.

RACF include la capacità di importare certificati e chiavi private in keyring gestiti da RACF. Consultare le pubblicazioni z/OS Server di protezione RACF per i dettagli e gli esempi su come caricare i certificati nei file di chiavi gestiti RACF .

Se l'installazione sta utilizzando uno dei prodotti PKI supportati, fare riferimento alle pubblicazioni che accompagnano il prodotto per informazioni su come distribuirlo.

## Gestione delle politiche di sicurezza Advanced Message Security

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 per AMS

Le politiche di sicurezza Advanced Message Security sono oggetti concettuali che descrivono il modo in cui un messaggio è crittograficamente 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 579](#)

Le politiche di protezione dati Advanced Message Security implicano una qualità di protezione (QOP).

[“Attributi della politica di sicurezza in AMS” a pagina 578](#)

È possibile utilizzare Advanced Message Security per selezionare un particolare algoritmo o metodo per proteggere i dati.

#### Nomi delle politiche in AMS

Il nome della normativa è un nome univoco che identifica una normativa Advanced Message Security specifica e la coda a cui viene applicata.

Il nome della politica deve corrispondere al nome della coda a cui si applica. Esiste una mappatura uno - a - uno tra un Advanced Message Security ( 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 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 di firma in AMS

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 crittografia in AMS

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 in AMS**

L'attributo di tolleranza indica se 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 in AMS**

I DN (distinguished name) del mittente identificano gli utenti autorizzati a inserire messaggi in una coda.

Advanced Message Security ( 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, 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.



Nome componente	Valore
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.
- AMS supporta i DN con valori solo dalla serie di caratteri Latin-1 . Per creare i DN con i caratteri della serie, è necessario prima creare un certificato con un DN creato nella codifica UTF-8 utilizzando UNIX con la codifica UTF-8 attivata o con la GUI **strmqikm** . Quindi, è necessario creare una politica da una piattaforma UNIX con la codifica UTF-8 attivata oppure utilizzare il plug-in AMS per IBM MQ.
- Il metodo utilizzato da AMS, per convertire il nome del mittente dal formato x.509 in formato DN, utilizza sempre ST = per il valore Stato o Provincia.
- I seguenti caratteri speciali richiedono caratteri di escape:

```

, (comma)
+ (plus)
" (double quote)
\ (backslash)
< (less than)
> (greater than)
; (semicolon)

```

- Se DN (Distinguished Name) contiene spazi, è necessario racchiudere il DN tra virgolette doppie.

### Concetti correlati

“DN (Distinguished Name) destinatario in AMS” a pagina 577

Il DN (distinguished name) del destinatario identifica gli utenti autorizzati a richiamare i messaggi da una coda.

### ***DN (Distinguished Name) destinatario in AMS***

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 i DN con i caratteri della serie, è necessario prima creare un certificato con un DN creato nella codifica UTF-8 utilizzando UNIX con la codifica UTF-8 attivata o con la GUI **strmqikm** . Quindi, è necessario creare una politica da una piattaforma UNIX con la codifica UTF-8 attivata oppure utilizzare il plug-in Advanced Message Security per IBM MQ.

### Concetti correlati

“Nomi distinti del mittente in AMS” a pagina 576

I DN (distinguished name) del mittente identificano gli utenti autorizzati a inserire messaggi in una coda.

### Attributi della politica di sicurezza in AMS

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.

<i>Tabella 92. Attributi della politica di sicurezza in AMS</i>	
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 **V 9.0.0** includono un quarto livello da IBM MQ 9.0 e tutti 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.
- **V 9.0.0** Riservatezza - i messaggi inseriti nella coda devono essere codificati. Per ulteriori informazioni, consultare [“Qualità di protezione disponibili con AMS” a pagina 521](#)
- 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".

**V 9.0.0** Una normativa che stabilisce che i messaggi devono essere codificati quando vengono inseriti in una coda ha un QOP di RISERVATEZZA. Un QOP di RISERVATEZZA significa che una normativa stabilisce un algoritmo di codifica.

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 un QOP di 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.

L'ubicazione da cui vengono eseguite tutte le attività di gestione relative alle politiche di sicurezza dipende dalla piattaforma utilizzata.

- **ULW** Su UNIXe Windows, utilizzare i comandi `DELETE POLICY`, `DISPLAY POLICY`e `SET POLICY` (o PCF equivalente) per gestire le politiche di sicurezza.

- **UNIX** Su UNIX, le attività di amministrazione possono essere eseguite da `MQ_INSTALLATION_PATH/bin`.

- **Windows** Su piattaforme Windows , le attività di gestione possono essere eseguite da qualsiasi ubicazione quando la variabile di ambiente `PATH` viene aggiornata durante l'installazione.

- **IBM i** Su IBM i, i comandi `DSPMQMSPL`, `SETMQMSPL`e `WRKMQMSPL` vengono installati nella libreria di sistema `QSYS` per la lingua principale del sistema quando è installato IBM MQ .

Ulteriori versioni della lingua nazionale vengono installate nelle librerie `QSYS29xx` in base al caricamento della funzione lingua. Ad esempio, una macchina con l'inglese americano come lingua principale e il coreano come lingua secondaria ha i comandi dell'inglese americano installati in `QSYS` e il carico della lingua secondaria coreana in `QSYS2962` come 2962 è il carico della lingua per il coreano.

- **z/OS** Su z/OS, i comandi di gestione vengono eseguiti utilizzando il programma di utilità della politica di sicurezza dei messaggi (`CSQOUTIL`). Quando le politiche vengono create, modificate o eliminate su z/OS, le modifiche non vengono riconosciute da Advanced Message Security finché il gestore code non viene arrestato e riavviato oppure il comando z/OS `MODIFY` viene utilizzato per aggiornare la configurazione della politica Advanced Message Security . Ad esempio:

```
F <qmgr ssid>AMSM,REFRESH POLICY
```

### Attività correlate

[“Creazione di politiche di sicurezza in AMS” a pagina 580](#)

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 in AMS” a pagina 581](#)

È possibile utilizzare Advanced Message Security per modificare dettagli delle politiche di sicurezza già definite.

[“Visualizzazione e dump delle politiche di sicurezza in AMS” a pagina 582](#)

Utilizzare il comando `dspmqsp1` 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 in AMS” a pagina 584](#)

Per rimuovere le politiche di sicurezza in Advanced Message Security, devi utilizzare il comando `setmqsp1` .

### Informazioni correlate

[Il programma di utilità della politica di sicurezza dei messaggi \(CSQOUTIL\)](#)

[Funzionamento di Advanced Message Security](#)

### Creazione di politiche di sicurezza in AMS

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 normativa di sicurezza deve seguire [Regole per la denominazione di oggetti IBM MQ](#).
  - È necessario disporre dell'autorizzazione necessaria per connettersi al gestore code e creare una politica di sicurezza:
    - **z/OS** Su z/OS, concedere le autorizzazioni documentate in [The message security policy utility \(CSQOUTIL\)](#).
    - **Multi** Su altre piattaforme diverse da z/OS, è necessario concedere le autorizzazioni + connect, + inq e + chg necessarie utilizzando il comando [setmqaut](#).
- Per ulteriori informazioni sulla configurazione della sicurezza, consultare [“Configurazione della sicurezza”](#) a pagina 118.
- **z/OS** Su z/OS, assicurarsi che gli oggetti di sistema richiesti siano stati definiti in base alle definizioni in CSQ4INSM.

### 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 SHA256 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 SHA256 -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 normativa specifica che i messaggi devono essere codificati utilizzando l'algoritmo 3DES per i certificati con DN: CN=john, O=IBM, C=US e CN=jeff, O=IBM, C=US e firmati con l'algoritmo SHA256 per il certificato con DN: CN=phil, O=IBM, C=US

```
setmqspl -m QMGR -p MY.OTHER.QUEUE -s SHA256 -e 3DES -r CN=john,O=IBM,C=US -r CN=jeff,O=IBM,C=US -a CN=phil,O=IBM,C=US
```

### 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.

### Informazioni correlate

[Elenco completo degli attributi del comando setmqspl](#)

### Modifica delle politiche di sicurezza in AMS

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 dell'autorizzazione necessaria per connettersi al gestore code e creare una politica di sicurezza.
  - **z/OS** Su z/OS, concedere le autorizzazioni documentate in [The message security policy utility \(CSQOUTIL\)](#).
  - **Multi** Su altre piattaforme diverse da z/OS, è necessario concedere le autorizzazioni + connect, + inq e + chg necessarie utilizzando il comando [setmqaut](#).

Per ulteriori informazioni sulla configurazione della sicurezza, consultare [“Configurazione della sicurezza”](#) a pagina 118.

## Informazioni su questa attività

Per modificare le politiche di sicurezza, applicare il comando `setmqsp1` a una politica già esistente che fornisce nuovi attributi.

### Esempio

Di seguito è riportato un esempio di creazione di una politica denominata MYQUEUE su un gestore code denominato QMGR, che specifica che i messaggi devono essere codificati utilizzando l'algoritmo 3DES per gli autori (-a) che hanno certificati con DN (Distinguished Name) di CN=alice, O=IBM, C=US e firmati con l'algoritmo SHA256 per i destinatari (-r) che hanno certificati con DN CN=jeff, O=IBM, C = US.

```
setmqsp1 -m QMGR -p MYQUEUE -e 3DES -s SHA256 -a CN=jeff,O=IBM,C=US -r CN=alice,O=IBM,C=US
```

Per modificare questa politica, emettere il comando `setmqsp1` 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:

```
setmqsp1 -m QMGR -p MYQUEUE -e AES256 -s SHA256 -a CN=jeff,O=IBM,C=US -r CN=alice,O=IBM,C=US
```



### Riferimenti correlati

[setmqsp1 \(impostazione politica di sicurezza\)](#)

## Visualizzazione e dump delle politiche di sicurezza in AMS

Utilizzare il comando `dspmqsp1` 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 dell'autorizzazione necessaria per connettersi al gestore code e creare una politica di sicurezza.
  -  Su z/OS, concedere le autorizzazioni documentate in [The message security policy utility \(CSQOUTIL\)](#).
  -  Su altre piattaforme diverse da z/OS, è necessario concedere le autorizzazioni + connect, + inq e + chg necessarie utilizzando il comando `setmqaut`.

Per ulteriori informazioni sulla configurazione della sicurezza, consultare [“Configurazione della sicurezza”](#) a pagina 118.

## Informazioni su questa attività

Di seguito è riportato l'elenco di indicatori di comando `dspmqsp1` :

Tabella 93. indicatori del comando <code>dspmqsp1</code> .	
Indicatore comando	Spiegazione
-m	Nome gestore code (obbligatorio).
-p	Il nome della politica.
-export	L'aggiunta di questo indicatore genera un output che può essere facilmente applicato a un gestore code differente.

## Esempio

L'esempio seguente mostra come creare due criteri di sicurezza per `venus.queue.manager`:

```
setmqsp1 -m venus.queue.manager -p AMS_POL_04_ONE -s sha256 -a "CN=signer1,O=IBM,C=US" -e NONE
setmqsp1 -m venus.queue.manager -p AMS_POL_06_THREE -s sha256 -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:

```
dspmqsp1 -m venus.queue.manager
```

```
Policy Details:
Policy name: AMS_POL_04_ONE
Quality of protection: INTEGRITY
Signature algorithm: SHA256
Encryption algorithm: NONE
Signer DNS:
  CN=signer1,O=IBM,C=US
Recipient DNS: -
Toleration: 0
-----
Policy Details:
Policy name: AMS_POL_06_THREE
Quality of protection: INTEGRITY
Signature algorithm: SHA256
Encryption algorithm: NONE
Signer DNS:
  CN=another signer,O=IBM,C=US
Recipient DNS: -
Toleration: 0
```

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:


```
dspmqsp1 -m venus.queue.manager -p AMS_POL_06_THREE
```

```
Policy Details:
Policy name: AMS_POL_06_THREE
Quality of protection: INTEGRITY
Signature algorithm: SHA256
Encryption algorithm: NONE
Signer DNS:
  CN=another signer,O=IBM,C=US
Recipient DNS: -
Toleration: 0
```

Nell'esempio successivo, creiamo prima una politica di sicurezza e poi la esportiamo utilizzando l'indicatore **-export** :



```
setmqsp1 -m venus.queue.manager -p AMS_POL_04_ONE -s SHA256 -a "CN=signer1,O=IBM,C=US" -e NONE
dspmqsp1 -m venus.queue.manager -export
```

 Su z/OS, le informazioni sulla politica esportata vengono scritte da CSQOUTIL in EXPORT DD.

 Su piattaforme diverse dal z/OS, reindirizzare l'emissione a un file, ad esempio:

```
dspmqsp1 -m venus.queue.manager -export > policies.[bat|sh]
```

Per importare una politica di sicurezza:

-  Su Windows, eseguire `policies.bat`.
-  Su UNIX:

1. Collegarsi come utente appartenente al gruppo di amministrazione mqm IBM MQ .
  2. Immettere `. policies.sh`.
- **z/OS** Su z/OS utilizzare il programma CSQOUTIL , specificando per SYSIN il data set contenente le informazioni della politica esportata.

### Informazioni correlate

Elenco completo degli attributi del comando `dspmqspl`

### Rimozione delle politiche di sicurezza in AMS

Per rimuovere le politiche di sicurezza in Advanced Message Security, devi 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 dell'autorizzazione necessaria per connettersi al gestore code e creare una politica di sicurezza.
  - **z/OS** Su z/OS, concedere le autorizzazioni documentate in [The message security policy utility \(CSQOUTIL\)](#).
  - **Multi** Su altre piattaforme diverse da z/OS, è necessario concedere le autorizzazioni + connect, + inq e + chg necessarie utilizzando il comando `setmqaut` .

Per ulteriori informazioni sulla configurazione della sicurezza, consultare [“Configurazione della sicurezza”](#) a pagina 118.

### 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
```

### Informazioni correlate

Elenco completo degli attributi del comando `setmqspl`

### Protezione della coda di sistema in AMS

Le code di sistema consentono la comunicazione tra IBM 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 IBM 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. Consultare [“Creazione di politiche di sicurezza in AMS”](#) a pagina 580.

- **Windows** Per utilizzare la protezione della coda di sistema su Windows, copiare il file `keystore.conf` nella seguente directory:

```
c:\Documents and Settings\Default User\.mq\keystore.conf
```









► **z/OS** Su z/OS, per fornire 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 IBM MQ Explorer non sono protetti da AMS.

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
- ► **z/OS** SYSTEM.ADMIN.COMMAND.QUEUE
- SYSTEM.ADMIN.CONFIG.EVENT
- SYSTEM.ADMIN.LOGGER.EVENT
- SYSTEM.ADMIN.PERFM.EVENT
- SYSTEM.ADMIN.PUBSUB.EVENT
- SYSTEM.ADMIN.QMGR.EVENT
- SYSTEM.ADMIN.STATISTICS.QUEUE
- SYSTEM.ADMIN.TRACE.ROUTE.QUEUE
- SYSTEM.AUTH.DATA.QUEUE
- SYSTEM.BROKER.ADMIN.STREAM
- ► **z/OS** SYSTEM.BROKER.CLIENTS.DATA
- SYSTEM.BROKER.CONTROL.QUEUE
- SYSTEM.BROKER.DEFAULT.STREAM
- SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS
- ► **z/OS** SYSTEM.BROKER.SUBSCRIPTIONS.DATA
- SYSTEM.CHANNEL.INITQ
- SYSTEM.CHANNEL.SYNCQ
- ► **z/OS** SYSTEM.CHLAUTH.DATA.QUEUE
- SYSTEM.CICS.INITIATION.QUEUE
- SYSTEM.CLUSTER.COMMAND.QUEUE
- SYSTEM.CLUSTER.HISTORY.QUEUE
- SYSTEM.CLUSTER.REPOSITORY.QUEUE
- SYSTEM.CLUSTER.TRANSMIT.QUEUE
- ► **z/OS** SYSTEM.COMMAND.INPUT
- ► **z/OS** SYSTEM.DDELAY.LOCAL.QUEUE
- SYSTEM.DEAD.LETTER.QUEUE
- SYSTEM.DURABLE.SUBSCRIBER.QUEUE
- SYSTEM.HIERARCHY.STATE

- SYSTEM.INTER.QMGR.CONTROL
- SYSTEM.INTER.QMGR.FANREQ
- SYSTEM.INTER.QMGR.PUBS
- SYSTEM.INTERNAL.REPLY.QUEUE
-  SYSTEM.JMS.PS.STATUS.QUEUE
-  SYSTEM.JMS.REPORT.QUEUE
- SYSTEM.PENDING.DATA.QUEUE
- SYSTEM.PROTECTION.ERROR.QUEUE
- SYSTEM.PROTECTION.POLICY.QUEUE
-  SYSTEM.QSG.CHANNEL.SYNCQ
-  SYSTEM.QSG.TRANSMIT.QUEUE
-  SYSTEM.QSG.UR.RESOLUTION.QUEUE
- SYSTEM.RETAINED.PUB.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 `setmqsp1` e `dspmqspl`. Tuttavia, 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 gestione IBM MQ, o che non dispone delle autorizzazioni per leggere le impostazioni della politica di sicurezza concesse, genera 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
```

**Nota:** È necessario impostare queste autorizzazioni OAM solo se si intende connettere i client al gestore code utilizzando Advanced Message Security 7.0.1.



**Attenzione:** Ricercare l'autorizzazione al SISTEMA `SYSTEM.PROTECTION.POLICY.QUEUE` non è obbligatorio in tutte le situazioni. IBM MQ ottimizza le prestazioni memorizzando nella cache le politiche in modo da non dover sfogliare i record per i dettagli della politica sul `SYSTEM.PROTECTION.POLICY.QUEUE` in tutti i casi.

IBM MQ non memorizza nella cache tutte le politiche disponibili. Se è presente un numero elevato di politiche, IBM MQ memorizza nella cache un numero limitato di politiche. Quindi, se il gestore code ha un numero basso di politiche definite, non è necessario fornire l'opzione di ricerca al SISTEMA `SYSTEM.PROTECTION.POLICY.QUEUE`.

Tuttavia, è necessario concedere l'autorizzazione di ricerca a questa coda, nel caso in cui sia definito un numero elevato di politiche o se si utilizzano vecchi client. Il SISTEMA `SYSTEM.PROTECTION.ERROR.QUEUE` viene utilizzato per inserire i messaggi di errore generati dal codice AMS. L'autorizzazione all'inserimento rispetto a questa coda viene controllata solo quando si tenta di inserire un messaggio di errore nella coda. L'autorizzazione di inserimento rispetto alla

coda non viene controllata quando si tenta di inserire o richiamare un messaggio da una coda protetta AMS.

## Concessione delle autorizzazioni di protezione

Quando si utilizza la sicurezza della risorsa comando, è necessario impostare le autorizzazioni per consentire il funzionamento di Advanced Message Security . Questo argomento utilizza i comandi RACF negli esempi. Se l'azienda utilizza un ESM (External Security Manager) differente, è necessario utilizzare i comandi equivalenti per tale ESM.

Ci sono tre aspetti per concedere le autorizzazioni di protezione:

- [“Lo spazio di indirizzo AMSM” a pagina 587](#)
- [“CSQOUTIL” a pagina 587](#)
- [“Utilizzo di code che hanno una politica Advanced Message Security definita” a pagina 588](#)

**Note:** I comandi di esempio utilizzano le seguenti variabili.

1. *QMgrName* - il nome del gestore code.



Su z/OS, questo valore può essere anche il nome di un gruppo di condivisione code.

2. *username* - può essere un nome gruppo.
3. Gli esempi mostrano la classe MQQUEUE. può essere anche MXQUEUE, GMQUEUE o GMXQUEUE. Consultare [“Profili per la sicurezza della coda” a pagina 192](#) per ulteriori informazioni.

Inoltre, se il profilo esiste già, non è necessario il comando RDEFINE.

## Lo spazio di indirizzo AMSM

È necessario emettere della sicurezza IBM MQ per il nome utente con cui viene eseguito lo spazio di indirizzo Advanced Message Security .

- Per la connessione batch al gestore code, immettere

```
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
PERMIT QMgrName.BATCH CLASS(MQCONN) ID(username) ACCESS(READ)
```

- Per l'accesso al SISTEMA SYSTEM.PROTECTION.POLICY.QUEUE, emissione:

```
RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
PERMIT QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(READ)
```

## CSQOUTIL

Il programma di utilità che consente agli utenti di eseguire i comandi **setmqsp1** e **dspmqsp1** richiede le seguenti autorizzazioni, dove il nome utente è l'ID utente del lavoro:

- Per la connessione batch al gestore code, immettere:

```
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
PERMIT QMgrName.BATCH CLASS(MQCONN) ID(username) ACCESS(READ)
```

- Per l'accesso al SISTEMA SYSTEM.PROTECTION.POLICY.QUEUE, richiesto per il comando **setmqpol** , immettere:

```
RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
PERMIT QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(ALTER)
```

- Per l'accesso al SISTEMA SYSTEM.PROTECTION.POLICY.QUEUE, richiesto per il comando **dspmqp01**, immettere:

```
RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
          PERMIT QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(READ)
```

## Utilizzo di code che hanno una politica Advanced Message Security definita

Quando un'applicazione esegue operazioni con le code su cui è definita una politica, tale applicazione richiede ulteriori autorizzazioni per consentire a Advanced Message Security di proteggere i messaggi.

L'applicazione richiede:

- Accesso in lettura a SYSTEM.PROTECTION.POLICY.QUEUE. Eseguire questa operazione emettendo:

```
RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
          PERMIT QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(READ)
```

- Inserire l'accesso al SISTEMA SYSTEM.PROTECTION.ERROR.QUEUE. Eseguire questa operazione emettendo:

```
RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.ERROR.QUEUE UACC(NONE)
          PERMIT QMgrName.SYSTEM.PROTECTION.ERROR.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(READ)
```

## Impostazione dei certificati e del file di configurazione keystore su IBM i

La tua prima attività quando configuri la protezione Advanced Message Security è creare un certificato e associarlo al tuo ambiente. L'associazione viene configurata tramite un file contenuto nell'IFS (integrated filesystem).

### Procedura

1. Per creare un certificato autofirmato utilizzando la strumentazione OpenSSL fornita con IBM i, immettere il seguente comando da QShell:

```
/QOpenSys/usr/bin/openssl req -x509 -newkey rsa:2048 -keyout
$HOME/private.pem -out $HOME/mycert.pem -nodes -days 365
```

Il comando richiede diversi attributi DN (Distinguished Name) per un nuovo certificato autofirmato, inclusi:

- Nome comune (CN =)
- Organizzazione (O =)
- Paese (C =)

Questo crea una chiave privata non crittografata e un certificato corrispondente, entrambi in formato PEM (Privacy Enhanced Mail).

Per semplicità, immettere semplicemente i valori per nome comune, organizzazione e paese. Tali attributi e valori sono importanti quando viene creata una politica.

2. AMS richiede che il certificato e la chiave privata siano conservati nello stesso file. Immettere il seguente comando per ottenere questo risultato:

```
cat $HOME/mycert.pem >> $HOME/private.pem
```

Il file `private.pem` in `$HOME` ora contiene una chiave privata e un certificato corrispondenti, mentre il file `mycert.pem` contiene tutti i certificati pubblici per i quali è possibile codificare i messaggi e convalidare le firme.

I due file devono essere associati all'ambiente creando un file di configurazione keystore, `keystore.conf`, nell'ubicazione predefinita.

Per impostazione predefinita, AMS cerca la configurazione del keystore in una directory secondaria `.mqc` della propria directory home.

3. In QShell creare il file `keystore.conf` :

```
mkdir -p $HOME/.mqc
echo "pem.private = $HOME/private.pem" > $HOME/.mqc/keystore.conf
echo "pem.public = $HOME/mycert.pem" >> $HOME/.mqc/keystore.conf
echo "pem.password = unused" >> $HOME/.mqc/keystore.conf
```

## IBM i **Creazione di una politica su IBM i**

Prima di creare una politica, è necessario creare una coda per conservare i messaggi protetti.

### Procedura

1. Da un prompt della riga comandi immettere;

```
CRTMQMQ QNAME(PROTECTED) QTYPE(*LCL) MQMNAME (mqmname)
```

dove `mqmname` è il nome del tuo gestore code.

Utilizzare il comando `DSPMQM` per controllare che il gestore code sia in grado di utilizzare le politiche di sicurezza. Assicurarsi che **Security Policy Capability** mostri *\*YES*.

La politica più semplice che è possibile definire è una politica di integrità, che si ottiene creando una politica con un algoritmo di firma digitale ma senza algoritmo di codifica.

I messaggi sono firmati ma non codificati. Se i messaggi devono essere codificati, è necessario specificare un algoritmo di crittografia e uno o più destinatari dei messaggi previsti.

Un certificato nel keystore pubblico per un destinatario del messaggio previsto viene identificato mediante un DN (distinguished name).

2. Visualizzare i nomi distinti dei certificati nel keystore pubblico, `mycert.pem` in `$HOME`, utilizzando il seguente comando in QShell:

```
/QOpenSys/usr/bin/openssl x509 -in $HOME/mycert.pem -noout -subject -nameopt RFC2253
```

È necessario immettere il DN (distinguished name) come destinatario previsto e il nome della politica deve corrispondere al nome della coda da proteggere.

3. Da una richiesta comandi CL, immettere, ad esempio:

```
SETMQMSPL POLICY(PROTECTED) MQMNAME (mqmname) SIGNALG(*SHA256) ENCALG(*AES256) RECIP('CN=.., O=.., C=..')
```

dove `mqmname` è il nome del tuo gestore code.

Una volta creata la politica, tutti i messaggi inseriti, sfogliati o rimossi in modo distruttivo tramite tale nome coda sono soggetti alla politica AMS.

### Informazioni correlate

[Visualizza gestore code messaggi \(DSPMQM\)](#)

[Imposta politica di sicurezza MQM \(SETMQMSPL\)](#)

Utilizzare le applicazioni di esempio fornite con il prodotto per verificare le politiche di sicurezza.

## Informazioni su questa attività

È possibile utilizzare le applicazioni di esempio fornite con IBM MQ , come ad esempio AMQSPUT4, AMQSGET4, AMQSGBR4 e strumenti come WRKMQMMSG per inserire, sfogliare e richiamare i messaggi utilizzando il nome della coda PROTECTED.

Se tutto è stato configurato correttamente, non dovrebbe esserci alcuna differenza nel comportamento dell'applicazione rispetto a quello di una coda non protetta per questo utente.

Un utente non impostato per Advanced Message Security che non dispone della chiave privata richiesta per decodificare il messaggio, tuttavia, non sarà in grado di visualizzare il messaggio. L'utente riceve un codice di completamento di RCFAIL, equivalente a MQCC\_FAILED (2) e codice motivo RC2063 (MQRC\_SECURITY\_ERROR).

Per verificare che la protezione AMS sia attiva, inserire alcuni messaggi di prova nella coda PROTECTED, ad esempio utilizzando AMQSPUT0. È possibile quindi creare una coda alias per sfogliare i dati protetti non elaborati mentre sono inattivi.

## Procedura

Per concedere le autorizzazioni necessarie a un utente, eseguire:

```
CRTMQMQ QNAME(ALIAS) QTYPE(*ALS) TGTQNAME(PROTECTED) MQMNAME(yourqm)
```

L'esplorazione mediante il nome della coda ALIAS, ad esempio utilizzando AMQSBCG4 o WRKMQMMSG, dovrebbe rivelare messaggi scrambled più grandi in cui un'esplorazione della coda PROTECTED mostra messaggi in testo non codificato.

I messaggi criptati sono visibili, ma il cleartext originale non è decifrabile utilizzando la coda ALIAS, poiché non esiste alcuna politica per AMS che applichi la corrispondenza con questo nome. Quindi, vengono restituiti i dati protetti non elaborati.

### Informazioni correlate

[Imposta politica di sicurezza MQM \(SETMQMSPL\)](#)

[Gestione messaggi MQ \(WRKMQMMSG\)](#)

## 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 IBM MQ sono messaggi in formato PCF inviati a code dedicate sul gestore code in cui si verifica l'evento.

I messaggi degli eventi di configurazione vengono inviati a SYSTEM.ADMIN.CONFIG.EVENT .

I messaggi di eventi comando vengono inviati al SISTEMA SYSTEM.ADMIN.COMMAND.EVENT .

Gli eventi vengono generati indipendentemente dagli strumenti che stai 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 in AMS” a pagina 580](#), che genera due messaggi di eventi IBM MQ :
  - Un evento di configurazione
  - Un evento di comando
- [“Modifica delle politiche di sicurezza in AMS” a pagina 581](#), che genera tre messaggi di eventi IBM 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 in AMS” a pagina 582](#), che genera un messaggio di evento IBM MQ :
  - Un evento di comando
- [“Rimozione delle politiche di sicurezza in AMS” a pagina 584](#), che genera due messaggi di eventi IBM MQ :
  - Un evento di configurazione
  - Un evento di comando

### **Abilitazione e disabilitazione della registrazione eventi**

È possibile controllare gli eventi di comando e configurazione 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 di comando per i comandi tranne i comandi **DISPLAY MQSC** e i comandi 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)
```

### **Informazioni correlate**

[Controllo degli eventi di configurazione, comando e logger in IBM 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:	<b>CONSOLLE MQEVO</b> Riga comandi della console. <b>MQEVO_MSG</b> Messaggio di comando dal plugin IBM 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:	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.

### Dati EventIdentity

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.

### **Comando**

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**  
Sono definiti in IBM MQ 8.0 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 di formato Advanced Message Security standard.

I valori possibili per il descrittore del messaggio MQMD possono essere trovati in [MQMD del messaggio di evento \(descrittore del messaggio\)](#).

Di seguito sono riportati i valori MQMD selezionati:

```
Format = MQFMT_EVENT
Persistence = MQPER_PERSISTENCE_AS_Q_DEF
PutAppType = 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. I valori MQCFH possibili possono essere trovati in [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:	<b>ID UTENTE MQACF_EVENT_</b>
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:	<b>ID_MQBACF_EVENT_SECURITY_ID</b>
Tipo di dati:	MQCBS.
Lunghezza massima:	MQ_SECURITY_ID_LENGTH.
Restituito:	Sempre.

### EventOrigin

Descrizione:	L'origine dell'azione che causa l'evento.
Identificatore:	<b>MQIACF_EVENT_ORIGIN</b>
Tipo di dati:	MQCFIN.

Valori: **CONSOLE MQEVO**  
Riga comandi della console.  
**MQEVO\_MSG**  
Messaggio di comando dal plugin Esplora risorse di IBM 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: 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 IBM MQ 8.0 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 IBM MQ 8.0 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 IBM MQ 8.0 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 IBM MQ 8.0 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 IBM MQ 8.0 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 IBM MQ 8.0 o nel file cmqc . h .  
Restituito: Ogni volta che è definito un algoritmo di codifica nella politica IBM MQ

### ***SignerDNs***

Descrizione: Oggetto DistinguishedName dei firmatari consentiti.  
Identificatore: **DN\_SIGNER\_MQCA**  
Tipo di dati: MQCFSL  
Valore: **2113** - un valore numerico definito in IBM MQ 8.0 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 IBM MQ .

### ***RecipientDNs***

Descrizione: Oggetto DistinguishedName dei firmatari consentiti.  
Identificatore: **DN\_MQCA\_RECIPIENT\_**  
Tipo di dati: MQCFSL  
Valore: **2114** - un valore numerico definito in IBM MQ 8.0 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 IBM MQ .

## **Problemi e soluzioni**

Le informazioni vengono fornite per identificare e risolvere i problemi relativi a Advanced Message Security.

Per problemi relativi a Advanced Message Security , controllare prima il log degli errori del gestore code.

## com.ibm.security.pkcsutil.PKCSException: Errore durante la codifica del contenuto

L'errore com.ibm.security.pkcsutil.PKCSException: Error encrypting contents suggerisce che Advanced Message Security ha problemi con l'accesso agli algoritmi crittografici.

Se il seguente errore viene restituito da Advanced Message Security:

```
DRQJP0103E The 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, per la propria versione del prodotto, dalla seguente ubicazione:[Fix in IBM Developer Kits](#).

## Supporto OSGi

Per utilizzare il bundle OSGi con Advanced Message Security sono richiesti ulteriori parametri.

Eeguire 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:** AMS supporta la comunicazione utilizzando solo classi MQ Base Java 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 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 AMS come descritto in [“Guida rapida per AMS con client Java”](#) a pagina 537.

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 524.

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 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 ciò si ottiene configurando la codifica TLS sul canale di connessione server utilizzato dall'applicazione.

4. Impostare la variabile di ambiente `AMQ_DISABLE_CLIENT_AMS` se non si desidera utilizzare AMS.

Consultare [“Intercettazione MCA \(Message Channel Agent\)”](#) a pagina 555 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.

## Configurazioni di esempio su z/OS

Questa sezione fornisce configurazioni di esempio di politiche e certificati per gli scenari di accodamento Advanced Message Security su z/OS.

Gli esempi riguardano le politiche Advanced Message Security richieste e i certificati digitali che devono esistere relativi agli utenti e ai file di chiavi. Gli esempi presuppongono che gli utenti coinvolti negli scenari siano stati impostati seguendo le istruzioni fornite in [Task 27: Grant users resource permissions for Advanced Message Security](#).

## Accodamento locale dei messaggi protetti dall'integrità su z/OS

Questo esempio descrive in dettaglio i certificati e le politiche Advanced Message Security necessari per l'invio e il richiamo di messaggi protetti da integrità da e verso una coda, locale per le applicazioni di inserimento e richiamo.

Il gestore code e la coda di esempio sono:

```
BNK6      - Queue manager
FIN.XFER.Q7 - Local queue
```

Vengono utilizzati i seguenti utenti:

```
WMQBNK6  - AMS task user
TELLER5   - Sending user
FINADM2   - Recipient user
```

### Creare i certificati utente

In questo esempio, è necessario un solo certificato utente. Questo è il certificato dell'utente mittente necessario per firmare i messaggi protetti dall'integrità. L'utente mittente è 'TELLER5'.

È richiesto anche il certificato CA (Certificate Authority). Il certificato CA è il certificato dell'autorità che ha emesso il certificato utente. Può essere una catena di certificati. In tal caso, tutti i certificati nella catena sono necessari nel key ring dell'utente dell'attività Advanced Message Security, in questo caso l'utente WMQBNK6.

È possibile creare un certificato CA utilizzando il comando RACF RACDCERT. Questo certificato viene utilizzato per emettere certificati utente. Ad esempio:

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('BCOCA') O('BCO') C('US'))
KEYUSAGE(CERTSIGN) WITHLABEL('BCOCA')
```

Questo comando RACDCERT crea un certificato CA che può quindi essere utilizzato per emettere un certificato utente per l'utente 'TELLER5'. Ad esempio:

```
RACDCERT ID(TELLER5) GENCERT SUBJECTSDN(CN('Te11er5') O('BCO') C('US'))
WITHLABEL('Te11er5') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

L'installazione avrà procedure per la scelta o la creazione di un certificato CA, nonché procedure per l'emissione di certificati e la loro distribuzione ai sistemi pertinenti.

Durante l'esportazione e l'importazione di questi certificati, Advanced Message Security richiede:

- Il certificato CA (catena).
- Il certificato utente e la sua chiave privata.

Se si utilizza RACF, il comando RACDCERT EXPORT può essere utilizzato per esportare i certificati in un dataset e il comando RACDCERT ADD può essere utilizzato per importare i certificati dal dataset. Per ulteriori informazioni su questi e altri comandi RACDCERT, fare riferimento a *z/OS: Security Server RACF Command Language Reference*.

I certificati in questo caso sono richiesti sul sistema z/OS su cui è in esecuzione il gestore code BNK6.

Quando i certificati sono stati importati sul sistema z/OS su cui è in esecuzione BNK6, il certificato utente richiede l'attributo TRUST. Il comando RACDCERT ALTER può essere utilizzato per aggiungere l'attributo TRUST al certificato. Ad esempio:

```
RACDCERT ID(TELLER5) ALTER (LABEL('Te11er5')) TRUST
```

In questo esempio, non è richiesto alcun certificato per l'utente destinatario.

## Connetti i certificati ai portachiavi pertinenti

Quando i certificati richiesti sono stati creati o importati e impostati come attendibili, devono essere connessi ai keyring utente appropriati sul sistema z/OS su cui è in esecuzione BNK6. Per creare i file di chiavi, utilizzare i comandi RACDCERT ADDRING:

```
RACDCERT ID(WMQBNK6) ADDRING(drq.ams.keyring)
RACDCERT ID(TELLER5) ADDRING(drq.ams.keyring)
```

Ciò crea un keyring per l'utente dell'attività Advanced Message Security , WMQBNK6, e un keyring per l'utente mittente, 'TELLER5'. Notare che il nome del file di chiavi drq.ams.keyring è obbligatorio e il nome è sensibile al maiuscolo / minuscolo.

Una volta creati i portachiavi, è possibile collegare i relativi certificati:

```
RACDCERT ID(WMQBNK6) CONNECT(CERTAUTH LABEL('BCOCA')
RING(drq.ams.keyring))
RACDCERT ID(TELLER5) CONNECT(ID(TELLER5) LABEL('Te11er5')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

Il certificato utente mittente deve essere connesso come DEFAULT. Se l'utente mittente dispone di più di un certificato nel proprio drq.ams.keyring, il certificato predefinito viene utilizzato per scopi di firma.

La creazione e la modifica dei certificati non viene riconosciuta da Advanced Message Security fino a quando il gestore code non viene arrestato e riavviato oppure fino a quando non viene utilizzato il comando z/OS **MODIFY** per aggiornare la configurazione del certificato Advanced Message Security . Ad esempio:

```
F BNK6AMSM,REFRESH KEYRING
```

## Creare la politica Advanced Message Security

In questo esempio, i messaggi protetti dall'integrità vengono inseriti nella coda FIN.XFER.Q7 da un'applicazione in esecuzione come utente 'TELLER5' e richiamata dalla stessa coda da un'applicazione in esecuzione come utente 'FINADM2', quindi è richiesta una sola politica Advanced Message Security .

Le politiche Advanced Message Security vengono create utilizzando il programma di utilità CSQ0UTIL documentato in [Il programma di utilità della politica di sicurezza dei messaggi \(CSQ0UTIL\)](#).

Utilizzare il programma di utilità CSQ0UTIL per eseguire il seguente comando:

```
setmqsp1 -m BNK6 -p FIN.XFER.Q7 -s MD5 -a CN=Teller5,O=BCO,C=US
```

In questa politica, il gestore code viene identificato come BNK6. Il nome della politica e la coda associata è FIN.XFER.Q7. L'algoritmo utilizzato per generare la firma del mittente è MD5e il DN (distinguished name) dell'utente mittente è 'CN=Teller5,O=BCO,C=US'.

Dopo aver definito la politica, riavviare il gestore code BNK6 oppure utilizzare il comando z/OS **MODIFY** per aggiornare la configurazione della politica Advanced Message Security . Ad esempio:

```
F BNK6AMSM,REFRESH POLICY
```

## Accodamento locale dei messaggi protetti dalla privacy su z/OS

Questo esempio descrive in dettaglio le politiche Advanced Message Security e i certificati necessari per inviare e richiamare i messaggi protetti dalla privacy da e verso una coda, locale per le applicazioni di inserimento e richiamo. I messaggi protetti dalla privacy sono sia firmati che codificati.

Il gestore code di esempio e la coda locale sono i seguenti:

```
BNK6          - Queue manager  
FIN.XFER.Q8   - Local queue
```

Vengono utilizzati i seguenti utenti:

```
WMQBNK6      - AMS task user  
TELLER5      - Sending user  
FINADM2      - Recipient user
```

I passaggi per configurare questo scenario sono:

### Creare i certificati utente

In questo esempio, sono richiesti due certificati utente. Si tratta del certificato dell'utente mittente necessario per firmare i messaggi e del certificato dell'utente destinatario necessario per codificare e decodificare i dati del messaggio. L'utente mittente è 'TELLER5' e l'utente destinatario è 'FINADM2'.

È richiesto anche il certificato CA (Certificate Authority). Il certificato CA è il certificato dell'autorità che ha emesso il certificato utente. Può essere una catena di certificati. In tal caso, tutti i certificati nella catena sono necessari nel key ring dell'utente dell'attività Advanced Message Security , in questo caso l'utente WMQBNK6.

È possibile creare un certificato CA utilizzando il comando RACF RACDCERT. Questo certificato viene utilizzato per emettere certificati utente. Ad esempio:

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('BCOCA') O('BCO') C('US'))  
KEYUSAGE(CERTSIGN) WITHLABEL('BCOCA')
```



Questo comando RACDCERT crea un certificato CA che può essere utilizzato per emettere certificati utente per gli utenti 'TELLER5' e 'FINADM2'. Ad esempio:

```
RACDCERT ID(TELLER5) GENCERT SUBJECTSDN(CN('Teller5') O('BCO') C('US'))
WITHLABEL('Teller5') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)

RACDCERT ID(FINADM2) GENCERT SUBJECTSDN(CN('FinAdm2') O('BCO') C('US'))
WITHLABEL('FinAdm2') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

L'installazione avrà procedure per la scelta o la creazione di un certificato CA, nonché procedure per l'emissione di certificati e la loro distribuzione ai sistemi pertinenti.

Durante l'esportazione e l'importazione di questi certificati, Advanced Message Security richiede:

- Il certificato CA (catena).
- Il certificato utente di invio e la relativa chiave privata.
- Il certificato utente del destinatario e la relativa chiave privata.

Se si utilizza RACF, il comando RACDCERT EXPORT può essere utilizzato per esportare i certificati in un dataset e il comando RACDCERT ADD può essere utilizzato per importare i certificati dal dataset. Per ulteriori informazioni su questi e altri comandi RACDCERT, fare riferimento a *z/OS: Security Server RACF Command Language Reference*.

I certificati in questo caso sono richiesti sul z/OS sistema su cui è in esecuzione il gestore code BNK6.

Quando i certificati sono stati importati sul sistema z/OS su cui è in esecuzione BNK6, i certificati utente richiedono l'attributo TRUST. Il comando RACDCERT ALTER può essere utilizzato per aggiungere l'attributo TRUST al certificato. Ad esempio:

```
RACDCERT ID(TELLER5) ALTER (LABEL('Teller5')) TRUST
RACDCERT ID(FINADM2) ALTER (LABEL('FinAdm2')) TRUST
```

## Connetti i certificati ai portachiavi pertinenti

Quando i certificati richiesti sono stati creati o importati e impostati come attendibili, devono essere connessi ai keyring utente appropriati sul sistema z/OS su cui è in esecuzione BNK6. Per creare i file di chiavi utilizzare il comando RACDCERT ADDRING:

```
RACDCERT ID(WMQBNK6) ADDRING(drq.ams.keyring)
RACDCERT ID(TELLER5) ADDRING(drq.ams.keyring)
RACDCERT ID(FINADM2) ADDRING(drq.ams.keyring)
```

Ciò crea un file di chiavi per l'utente dell'attività Advanced Message Security e i file di chiavi per gli utenti di invio e destinatario. Notare che il nome del file di chiavi drq.ams.keyring è obbligatorio e il nome è sensibile al maiuscolo / minuscolo.

Una volta creati i portachiavi, è possibile collegare i certificati pertinenti.

```
RACDCERT ID(WMQBNK6) CONNECT(CERTAUTH LABEL('BCOCA')
RING(drq.ams.keyring))

RACDCERT ID(WMQBNK6) CONNECT(ID(FINADM2) LABEL('FinAdm2')
RING(drq.ams.keyring) USAGE(SITE))

RACDCERT ID(TELLER5) CONNECT(ID(TELLER5) LABEL('Teller5')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))

RACDCERT ID(FINADM2) CONNECT(ID(FINADM2) LABEL('FinAdm2')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

I certificati utente mittente e destinatario devono essere connessi come DEFAULT. Se uno degli utenti ha più di un certificato nel proprio drq.ams.keyring, il certificato predefinito viene utilizzato per scopi di firma e decodifica.

Il certificato dell'utente destinatario deve essere collegato anche al keyring dell'utente dell'attività Advanced Message Security con USAGE (SITE). Questo perché l'attività Advanced Message Security richiede la chiave pubblica del destinatario quando si codificano i dati del messaggio. Lo USAGE (SITE) impedisce che la chiave privata sia accessibile nel key ring.

La creazione e la modifica dei certificati non viene riconosciuta da Advanced Message Security fino a quando il gestore code non viene arrestato e riavviato oppure fino a quando non viene utilizzato il comando z/OS **MODIFY** per aggiornare la configurazione del certificato Advanced Message Security . Ad esempio:

```
F BNK6AMSM,REFRESH KEYRING
```

## Creare la politica Advanced Message Security

In questo esempio, i messaggi protetti dalla privacy vengono inseriti nella coda FIN.XFER.Q8 da un'applicazione in esecuzione come utente 'TELLER5' e richiamata dalla stessa coda da un'applicazione in esecuzione come utente 'FINADM2', per cui è richiesta una sola politica Advanced Message Security .

Le politiche Advanced Message Security vengono create utilizzando il programma di utilità CSQOUTIL documentato in [Il programma di utilità della politica di sicurezza dei messaggi \(CSQOUTIL\)](#).

Utilizzare il programma di utilità CSQOUTIL per eseguire il seguente comando:

```
setmqsp1 -m BNK6 -p FIN.XFER.Q8 -s SHA1 -e 3DES -a CN=Teller5,O=BCO,C=US -r  
CN=FinAdm2,O=BCO,C=US
```

In questa politica, il gestore code viene identificato come BNK6. Il nome della politica e la coda associata è FIN.XFER.Q8. L'algoritmo utilizzato per generare la firma del mittente è SHA1e il DN (distinguished name) dell'utente mittente è 'CN=Teller5,O=BCO,C=US' e l'utente destinatario è 'CN=FinAdm2,O=BCO,C=US'. L'algoritmo utilizzato per codificare i dati del messaggio è 3DES.

Dopo aver definito la politica, riavviare il gestore code BNK6 oppure utilizzare il comando z/OS **MODIFY** per aggiornare la configurazione della politica Advanced Message Security . Ad esempio:

```
F BNK6AMSM,REFRESH POLICY
```

## Accodamento remoto dei messaggi protetti dall'integrità su z/OS

Questo esempio descrive in dettaglio i certificati e le politiche Advanced Message Security necessari per inviare e richiamare i messaggi protetti dall'integrità da e verso le code gestite da due gestori code differenti. I due gestori code possono essere in esecuzione sullo stesso sistema z/OS o su sistemi z/OS differenti oppure un gestore code può essere su un sistema distribuito su cui è in esecuzione Advanced Message Security.

I gestori e le code di esempio sono:

```
BNK6          - Sending queue manager  
BNK7          - Recipient queue manager  
FIN.XFER.Q7  - Remote queue on BNK6  
FIN.RCPT.Q7  - Local queue on BNK7
```

Nota: per questo esempio, BNK6 e BNK7 sono gestori code in esecuzione su sistemi z/OS differenti.

Vengono utilizzati i seguenti utenti:

```
WMQBNK6 - AMS task user on BNK6
WMQBNK7 - AMStask user on BNK7
TELLER5 - Sending user on BNK6
FINADM2 - Recipient user on BNK7
```

La procedura per configurare questo scenario è la seguente:

## Creare i certificati utente

In questo esempio, è necessario un solo certificato utente. Si tratta del certificato dell'utente mittente necessario per firmare il messaggio protetto dall'integrità. L'utente mittente è 'TELLER5'.

È richiesto anche il certificato CA (Certificate Authority). Il certificato CA è il certificato dell'autorità che ha emesso il certificato utente. Può essere una catena di certificati. In tal caso, tutti i certificati nella catena sono necessari nel key ring dell'utente del task Advanced Message Security, in questo caso l'utente WMQBNK7.

È possibile creare un certificato CA utilizzando il comando RACF RACDCERT. Questo certificato viene utilizzato per emettere certificati utente. Ad esempio:

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('BCOCA') O('BCO') C('US'))
KEYUSAGE(CERTSIGN) WITHLABEL('BCOCA')
```

Questo comando RACDCERT crea un certificato CA che può essere utilizzato per emettere il certificato utente per l'utente 'TELLER5'. Ad esempio:

```
RACDCERT ID(TELLER5) GENCERT SUBJECTSDN(CN('Teller5') O('BCO') C('US'))
WITHLABEL('Teller5') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

L'installazione avrà procedure per la scelta o la creazione di un certificato CA, nonché procedure per l'emissione di certificati e la loro distribuzione ai sistemi pertinenti.

Quando si esportano e importano questi certificati, Advanced Message Security richiede:

- Il certificato CA (catena).
- Il certificato utente di invio e la relativa chiave privata.

Se si utilizza RACF, il comando RACDCERT EXPORT può essere utilizzato per esportare i certificati in un dataset e il comando RACDCERT ADD può essere utilizzato per importare i certificati dal dataset. Per ulteriori informazioni su questi e altri comandi RACDCERT, fare riferimento a *z/OS: Security Server RACF Command Language Reference*.

I certificati in questo caso sono richiesti sul sistema z/OS su cui è in esecuzione il gestore code BNK6 e BNK7.

In questo esempio, il certificato di invio deve essere importato sul sistema z/OS su cui è in esecuzione BNK6 e il certificato CA deve essere importato sul sistema z/OS su cui è in esecuzione BNK7. Una volta importati i certificati, il certificato utente richiede l'attributo TRUST. Il comando RACDCERT ALTER può essere utilizzato per aggiungere l'attributo TRUST al certificato. Ad esempio, su BNK6:

```
RACDCERT ID(TELLER5) ALTER (LABEL('Teller5')) TRUST
```

## Connetti i certificati ai portachiavi pertinenti

Quando i certificati richiesti sono stati creati o importati e impostati come attendibili, devono essere collegati ai keyring utente appropriati sul sistema z/OS su cui sono in esecuzione BNK6 e BNK7.

Per creare i keyring utilizzare il comando RACDCERT ADDRING, su BNK6:

```
RACDCERT ID(TELLER5) ADDRING(drq.ams.keyring)
```

Ciò crea un keyring per l'utente mittente su BNK6. Notare che il nome del file di chiavi drq.ams.keyring è obbligatorio e il nome è sensibile al maiuscolo / minuscolo.

Su BNK7:

```
RACDCERT ID(WMQBNK7) ADDRING(drq.ams.keyring)
```

Questo crea un file di chiavi per l'utente dell'attività Advanced Message Security su BNK7. Non è richiesto alcun keyring utente per 'TELLER5' su BNK7.

Una volta creati i portachiavi, è possibile collegare i certificati pertinenti.

Su BNK6:

```
RACDCERT ID(TELLER5) CONNECT(ID(TELLER5) LABEL('Teller5')  
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

Su BNK7:

```
RACDCERT ID(WMQBNK7) CONNECT(CERTAUTH LABEL('BCOCA')  
RING(drq.ams.keyring))
```

Il certificato utente mittente deve essere connesso come DEFAULT. Se l'utente mittente dispone di più di un certificato nel proprio drq.ams.keyring, il certificato predefinito viene utilizzato per scopi di firma.

La creazione e la modifica dei certificati non viene riconosciuta da Advanced Message Security fino a quando il gestore code non viene arrestato e riavviato oppure fino a quando non viene utilizzato il comando z/OS **MODIFY** per aggiornare la configurazione del certificato Advanced Message Security . Ad esempio:

Su BNK6:

```
F BNK6AMSM, REFRESH, KEYRING
```

Su BNK7:

```
F BNK7AMSM, REFRESH, KEYRING
```

## Creare le politiche Advanced Message Security

In questo esempio, i messaggi protetti dall'integrità vengono inseriti nella coda remota FIN.XFER.Q7 su BNK6 da un'applicazione in esecuzione come utente 'TELLER5' e richiamata dalla coda locale FIN.RCPT.Q7 su BNK7 da un'applicazione in esecuzione come utente 'FINADM2', quindi sono richieste due politiche Advanced Message Security .

Le politiche Advanced Message Security vengono create utilizzando il programma di utilità CSQOUTIL documentato in [Il programma di utilità della politica di sicurezza dei messaggi \(CSQOUTIL\)](#).

Utilizzare il programma CSQOUTIL per eseguire il seguente comando per definire una politica di integrità per la coda remota su BNK6:

```
setmqsp1 -m BNK6 -p FIN.XFER.Q7 -s MD5 -a CN=Teller5,0=BCO,C=US
```

In questa politica, il gestore code viene identificato come BNK6. Il nome della politica e la coda associata è FIN.XFER.Q7. L'algoritmo utilizzato per generare la firma del mittente è MD5e il DN (distinguished name) dell'utente mittente è 'CN=Teller5,O=BCO,C=US'.

Inoltre, utilizzare il programma di utilità CSQ0UTIL per eseguire il seguente comando per definire una politica di integrità per la coda locale su BNK7:

```
setmqsp1 -m BNK7 -p FIN.RCPT.Q7 -s MD5 -a CN=Teller5,O=BCO,C=US
```

In questa politica, il gestore code è identificato come BNK7. Il nome della politica e la coda associata è FIN.RCPT.Q7. L'algoritmo previsto per la firma del mittente è MD5e il DN (distinguished name) dell'utente mittente deve essere 'CN=Teller5,O=BCO,C=US'.

Dopo aver definito le due politiche, riavviare i gestori code BNK6 e BNK7 oppure utilizzare il comando z/OS **MODIFY** per aggiornare le configurazioni della politica Advanced Message Security . Ad esempio:

Su BNK6:

```
F BNK6AMSM,REFRESH,POLICY
```

Su BNK7:

```
F BNK7AMSM,REFRESH,POLICY
```

## **Accodamento remoto dei messaggi protetti dalla privacy su z/OS**

Questo esempio descrive in modo dettagliato le politiche e i certificati Advanced Message Security necessari per inviare e richiamare i messaggi protetti dalla privacy da e verso le code gestite da due gestori code differenti. I due gestori code possono essere in esecuzione sullo stesso sistema z/OS o su sistemi z/OS differenti oppure un gestore code può essere su un sistema distribuito su cui è in esecuzione Advanced Message Security.

I gestori e le code di esempio sono:

```
BNK6      - Sending queue manager
BNK7      - Recipient queue manager
FIN.XFER.Q7 - Remote queue on BNK6
FIN.RCPT.Q7 - Local queue on BNK7
```

Nota: per questo esempio BNK6 e BNK7 sono gestori code in esecuzione su sistemi z/OS differenti con lo stesso nome.

Vengono utilizzati i seguenti utenti:

```
WMQBNK6  - AMS task user on BNK6
WMQBNK7  - AMS task user on BNK7
TELLER5  - Sending user on BNK6
FINADM2  - Recipient user on BNK7
```

La procedura per configurare questo scenario è la seguente:

### **Creare i certificati utente**

In questo esempio, sono richiesti due certificati utente. Si tratta del certificato dell'utente mittente necessario per firmare i messaggi e del certificato dell'utente destinatario necessario per codificare e decodificare i dati del messaggio. L'utente mittente è 'TELLER5' e l'utente destinatario è 'FINADM2'.

È richiesto anche il certificato CA (Certificate Authority). Il certificato CA è il certificato dell'autorità che ha emesso il certificato utente. Può essere una catena di certificati. In tal caso, tutti i certificati nella catena

sono necessari nel key ring dell'utente del task Advanced Message Security , in questo caso l'utente WMQBANK7.

È possibile creare un certificato CA utilizzando il comando RACDCERT. Questo certificato viene utilizzato per emettere certificati utente. Ad esempio:

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('BCOCA') O('BCO') C('US'))  
KEYUSAGE(CERTSIGN) WITHLABEL('BCOCA')
```

Questo comando RACDCERT crea un certificato CA che può essere utilizzato per emettere certificati utente per gli utenti 'TELLER5' e 'FINADM2'. Ad esempio:

```
RACDCERT ID(TELLER5) GENCERT SUBJECTSDN(CN('Teller5') O('BCO') C('US'))  
WITHLABEL('Teller5') SIGNWITH(CERTAUTH LABEL('BCOCA'))  
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)  
  
RACDCERT ID(FINADM2) GENCERT SUBJECTSDN(CN('FinAdm2') O('BCO') C('US'))  
WITHLABEL('FinAdm2') SIGNWITH(CERTAUTH LABEL('BCOCA'))  
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

L'installazione avrà procedure per la scelta o la creazione di un certificato CA, nonché procedure per l'emissione di certificati e la loro distribuzione ai sistemi pertinenti.

Durante l'esportazione e l'importazione di questi certificati, Advanced Message Security richiede:

- Il certificato CA (catena).
- Il certificato utente di invio e la relativa chiave privata.
- Il certificato utente del destinatario e la relativa chiave privata.

Se si utilizza RACF, il comando RACDCERT EXPORT può essere utilizzato per esportare i certificati in un dataset e il comando RACDCERT ADD può essere utilizzato per importare i certificati dal dataset. Per ulteriori informazioni su questi e altri comandi RACDCERT, fare riferimento a *z/OS: Security Server RACF Command Language Reference*.

I certificati in questo caso sono richiesti sul sistema z/OS su cui è in esecuzione il gestore code BNK6 e BNK7.

In questo esempio, i certificati di invio e di destinatario devono essere importati sul sistema z/OS su cui è in esecuzione BNK6 e i certificati CA e di destinatario devono essere importati sul sistema z/OS su cui è in esecuzione BNK7. Una volta importati i certificati, i certificati utente richiedono l'attributo TRUST. Il comando RACDCERT ALTER può essere utilizzato per aggiungere l'attributo TRUST al certificato. Ad esempio:

Su BNK6:

```
RACDCERT ID(TELLER5) ALTER (LABEL('Teller5')) TRUST  
RACDCERT ID(FINADM2) ALTER (LABEL('FinAdm2')) TRUST
```

Su BNK7:

```
RACDCERT ID(FINADM2) ALTER (LABEL('FinAdm2')) TRUST
```

## Connetti i certificati ai portachiavi pertinenti

Quando i certificati richiesti sono stati creati o importati e impostati come attendibili, devono essere connessi ai keyring utente appropriati sui sistemi z/OS su cui sono in esecuzione BNK6 e BNK7.

Per creare i file di chiavi utilizzare il comando RACDCERT ADDRING:

Su BNK6:

```
RACDCERT ID(WMQBNK6) ADDRING(drq.ams.keyring)
RACDCERT ID(TELLER5) ADDRING(drq.ams.keyring)
```

Ciò crea un keyring per l'utente dell'attività Advanced Message Security e un keyring per l'utente mittente su BNK6. Notare che il nome del file di chiavi drq.ams.keyring è obbligatorio e il nome è sensibile al maiuscolo / minuscolo.

Su BNK7:

```
RACDCERT ID(WMQBNK7) ADDRING(drq.ams.keyring)
RACDCERT ID(FINADM2) ADDRING(drq.ams.keyring)
```

Ciò crea un file di chiavi per l'utente dell'attività Advanced Message Security e un file di chiavi per l'utente destinatario su BNK7.

Una volta creati i portachiavi, è possibile collegare i certificati pertinenti.

Su BNK6:

```
RACDCERT ID(WMQBNK6) CONNECT(ID(FINADM2) LABEL('FinAdm2')
RING(drq.ams.keyring) USAGE(SITE))

RACDCERT ID(TELLER5) CONNECT(ID(TELLER5) LABEL('Teller5')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

Su BNK7:

```
RACDCERT ID(WMQBNK7) CONNECT(CERTAUTH LABEL('BCOCA')
RING(drq.ams.keyring))

RACDCERT ID(FINADM2) CONNECT(ID(FINADM2) LABEL('FinAdm2')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

I certificati utente mittente e destinatario devono essere connessi come DEFAULT. Se uno degli utenti ha più di un certificato nel proprio drq.ams.keyring, il certificato predefinito viene utilizzato per scopi di firma e crittografia/decrittografia.

Su BNK6, il certificato dell'utente destinatario deve essere collegato anche al keyring dell'utente del task Advanced Message Security con USAGE (SITE). Questo perché l'attività Advanced Message Security richiede la chiave pubblica del destinatario quando si codificano i dati del messaggio. Lo USAGE (SITE) impedisce che la chiave privata sia accessibile nel key ring.

La creazione e la modifica dei certificati non viene riconosciuta da Advanced Message Security fino a quando il gestore code non viene arrestato e riavviato oppure fino a quando non viene utilizzato il comando z/OS **MODIFY** per aggiornare la configurazione del certificato Advanced Message Security . Ad esempio:

Su BNK6:

```
F BNK6AMSM, REFRESH, KEYRING
```

Su BNK7:

```
F BNK7AMSM, REFRESH, KEYRING
```

## Creare le politiche Advanced Message Security

In questo esempio, i messaggi protetti dalla privacy vengono inseriti nella coda remota FIN.XFER.Q7 su BNK6 da un'applicazione in esecuzione come utente 'TELLER5' e richiamata dalla coda locale

FIN.RCPT.Q7 su BNK7 da un'applicazione in esecuzione come utente 'FINADM2', quindi sono richieste due politiche Advanced Message Security .

Le politiche Advanced Message Security vengono create utilizzando il programma di utilità CSQOUTIL documentato in [Il programma di utilità della politica di sicurezza dei messaggi \(CSQOUTIL\)](#).

Utilizzare il programma di utilità CSQOUTIL per eseguire il seguente comando per definire una normativa sulla riservatezza per la coda remota su BNK6:

```
setmqsp1 -m BNK6 -p FIN.XFER.Q7 -s SHA1 -e 3DES -a CN=Teller5,O=BCO,C=US -r  
CN=FinAdm2,O=BCO,C=US
```

In questa politica, il gestore code viene identificato come BNK6. Il nome della politica e la coda associata è FIN.XFER.Q7. L'algoritmo utilizzato per generare la firma del mittente è SHA1, il DN (distinguished name) dell'utente mittente è 'CN=Teller5,O=BCO,C=US' e l'utente destinatario è 'CN=FinAdm2,O=BCO,C=US'. L'algoritmo utilizzato per codificare i dati del messaggio è 3DES.

Inoltre, utilizzare il programma di utilità CSQOUTIL per eseguire il seguente comando per definire una politica di privacy per la coda locale su BNK7: :

```
setmqsp1 -m BNK7 -p FIN.RCPT.Q7 -s SHA1 -e 3DES -a CN=Teller5,O=BCO,C=US -r  
CN=FinAdm2,O=BCO,C=US
```

In questa politica, il gestore code è identificato come BNK7. Il nome della politica e la coda associata è FIN.RCPT.Q7. L'algoritmo previsto per la firma del mittente è SHA1, il DN (distinguished name) dell'utente mittente è previsto essere 'CN=Teller5,O=BCO,C=US' e l'utente destinatario è 'CN=FinAdm2,O=BCO,C=US'. L'algoritmo utilizzato per decodificare i dati del messaggio è 3DES.

Dopo aver definito le due politiche, riavviare i gestori code BNK6 e BNK7 oppure utilizzare il comando z/OS **MODIFY** per aggiornare la configurazione della politica Advanced Message Security . Ad esempio:

Su BNK6:

```
F BNK6AMSM, REFRESH, POLICY
```

Su BNK7:

```
F BNK7AMSM, REFRESH, POLICY
```



## 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 COMMERCIALIZZABILITÀ 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 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 marchio 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: