



French MQ User Group - MQ octobre 2026

Evolutions TLS MQ

Luc-Michel Demey
Demey Consulting
LMD@Demey-Consulting.fr

Ce document a été préparé à partir de différentes sources d'information publiquement disponibles.
Tous les efforts ont été faits pour rendre ce document complet et exact. Demey Consulting décline par avance toute responsabilité quant aux erreurs et omissions présentes dans ce support. Seuls les documents de référence IBM font foi quant aux capacités et au paramétrage des produits cités.

Version 1.04 – Octobre 2026

Limitation de responsabilité :

Tous les efforts ont été faits pour rendre ce support de cours complet et exact. Demey Consulting décline par avance toute responsabilité quant aux erreurs et omissions présentes dans ce support. Seuls les documents de référence IBM font foi quant aux capacités et au paramétrage des produits cités.

IBM, WebSphere, MQSeries sont des marques déposées de International Business Machines.



Copyright 2001-2026 - Demey Consulting – 72, avenue du Général de Gaulle
94170 Le Perreux sur Marne
lmd@demey-consulting.fr

Tous droits réservés.

Toute reproduction, partielle ou totale de ce document par quelque procédé que ce soit, faite sans consentement écrit de Demey Consulting est illicite.



Evolutions TLS : Impact sur IBM MQ

- Certificats
- Magasins
- Paramètres MQ & Utilitaires



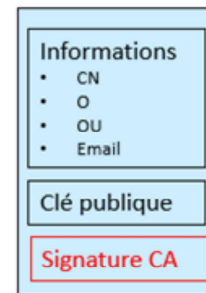


CERTIFICATS



Rappels

- Certificat SSL/TLS : carte d'identité numérique
- Permet d'identifier une entité (utilisateur, serveur, application, ...)
- Contient essentiellement 3 éléments :
 - Information sur l'entité
 - Clé publique associé à l'entité
 - Signature de la CA





Rappels

- Certificat signé par une CA (Certification Authority)
 - Ou par une chaine de certification (CA2 → CA1 → CA)
- Cas des certificats autosignés
 - A bannir en production (et dans les autres environnements)
- Pour IBM MQ, certificats stockés dans des « magasins de certificats »
 - Différents formats



Caractéristiques d'un certificat

- L'algorithme de signature
- L'algorithme de hachage
- La clé publique
- Le Nom Distinctif (Distinguished Name - DN), qui contient le champ CN (Common Name)
 - Possibilité de filtrage
- Les extensions
 - Serveur Authentification, Client Authentification
- La durée de validité



Exemple pour un certificat IBM MQ

- Algorithme de signature : sha256RSA
- Algorithme de hachage : sha256
- Clé publique : RSA 2048 bits
- Extensions : Serveur Authentication
- CN (Common Name) : libre
- Durée de validité : 13 mois

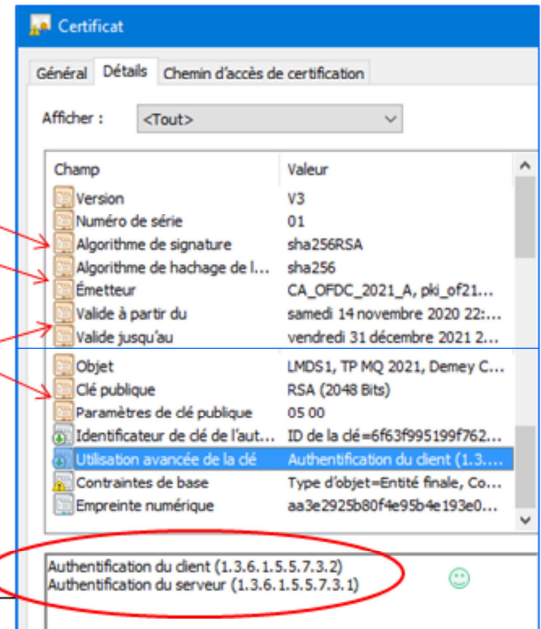
Si MQ Console :

- Certificat de type SAN (Subject Alternative Name) obligatoire



Caractéristique des certificats

- Valeurs recommandées (CA & certificats)
 - Algorithme de signature : **sha256RSA**
 - Algorithme de hachage : **sha256**
 - Clé publique : **RSA 2048 bits**
- Prévoir un label de certificat descriptif
- Extensions
 - Serveur Authentification
 - Client authentification
- Durée de validité
 - **13 mois** (certificat utilisateur)
- Eviter les certificats autosignés
 - Même en recette





Durée de validité d'un certificat

- Historiquement : créés pour des durées très longues
 - 5 ans, 10 ans, ...
- Depuis 2015 : limitation à 1 ou 2 ans
 - Règles de conformité, bonnes pratiques
- Depuis le 01/09/2020, la durée maximale autorisée pour un certificat SSL est de 397 jours (13 mois)
 - Certificate Authority/Browser Forum
- En 2025, le CA Forum vote une nouvelle règle
 - Diminution progressive de la durée de validité



Calendrier de réduction de la validité

- Depuis le 15 mars 2026 : 200 jours maximum
– (environ 6,5 mois)
- À partir du 15 mars 2027 : 100 jours maximum
- À partir du 15 mars 2029 : 47 jours maximum

Vous êtes ici



Solutions ?

- Embaucher massivement ;-)
- Automatiser le renouvellement
 - Let's Encrypt ?
 - Outils internes
 - Outils IBM ???
- Passer en CA privée et garder une durée de 13 mois
(Passer au format PKCS12 pour les magasins)

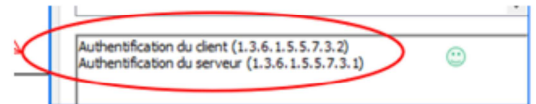


Extensions d'un certificat

L'affaire du clientAuth EKU

- Historiquement :
 - serverAuth & clientAuth dans les certificats publics
 - Certificats utilisables en mTLS avec serveur + client

- A partir du 03/2027 : clientAuth retiré
- Conséquences ?



Article : “IBM MQ and Extended Key Usage” de Robert Parker
<https://community.ibm.com/community/user/blogs/robert-parker1/2025/09/23/ibm-mq-and-extended-key-usage>



Conséquences pour IBM MQ

- La présence d'extensions n'est pas systématiquement testée dans les produits MQ
 - Mais elle va l'être (CERTVPOL)
 - Et peut être débrayée
- Sur certains produits, la présence de l'extension est testée, sa suppression peut avoir des conséquences immédiates :
 - MQ z/OS si TLS 1.3
 - MQ IPT
 - MQ Java / JMS (MQ Explorer)

Platform	Is the ECU value enforced?
IBM MQ on Unix, Linux, Windows	No - But from version 9.4.5 onwards it is enforced when CERTVPOL(RFC5280) is set.
IBM MQ on IBM i	No
IBM MQ Appliance	No - But from version 9.4.5 onwards it is enforced when CERTVPOL(RFC5280) is set.
IBM MQ Containers	No - But from version 9.4.5 onwards it is enforced when CERTVPOL(RFC5280) is set.
IBM MQ on IBM Cloud	No- But from version 9.4.5 onwards it is enforced when CERTVPOL(RFC5280) is set.
IBM MQ on z/OS	Yes, when TLS 1.3 is enabled.
IBM MQ C Client libraries	No - But from version 9.4.5 onwards it is enforced when cerificateValPolicy=RFC5280 is set.
IBM MQ Unmanaged .NET	No - But from version 9.4.5 onwards it is enforced when cerificateValPolicy=RFC5280 is set.
IBM MQIPT	Yes
IBM MQ Java/JMS client libraries	Yes, dependant on JRE configuration
IBM MQ on HP-NS	No



MAGASINS



Le magasin de certificats

- Contient :
 - les certificats
 - Les clés privées
 - les chaînes de certification
 - les CSR en attente de signature
- Terminologie :
 - IBM i : certificate store
 - Java et JMS: keystore & truststore
 - AIX, Linux, Windows : key database file
 - z/OS : keyring



Le magasin de certificats

- Protégé par mot de passe
- Fourni de deux manières :
 - via un fichier stash (.sth)
 - contient une version chiffrée du mot de passe
 - avec un algo propriétaire IBM.
 - via une variable d'environnement



Format des magasins (1)

Au niveau MQ, plusieurs formats de magasins sont disponibles :

- CMS
 - Peut utiliser un fichier stash pour stocker le mot de passe
 - Utilisé historiquement par les Queue Managers, les clients MQ, AMS, ...,
 - en cours de dépréciation / officiellement déprécié pour les applications Java
 - passer au format PKCS12

A noter que la sécurité du mot de passe repose sur un algo non publié, qui a été modifié par IBM à plusieurs reprises. Cette méthode est considérée comme faible, voire comme une vulnérabilité par certains.



Format des magasins

- JKS (monde Java)
 - 1 ou 2 magasins
- PKCS12 (p12)
 - Nouveau depuis MQ version 9.3.0
 - Pour les QM, clients MQ C & clients Java, MQIPT, MQExplorer, MFT, NativeHA, JWT, MQ Appliance, AMQP
 - Pas (encore) pour AMS



Le problème du mot de passe

- Le magasin étant protégé par un mot de passe, il faut fournir ce dernier au Queue Manager, au client MQ, ...
- Magasin CMS : le mot de passe peut être stocké dans un fichier stash
- Magasin PKCS12
 - Pour un Queue Manager : le mot de passe est stocké dans l'attribut KEYRPWD.
 - Il est fourni en clair au QM, et chiffré automatiquement en AES128.
 - Pour un client MQ : le mot de passe est stocké dans la variable d'environnement MQKEYRPWD après chiffrement via runmqicred.
- Le chiffrement utilise une clé par défaut, qui est la même pour toutes les installations.
- Fourniture d'une clé spécifique
 - Attribut INITKEY pour le QM
 - Variable d'environnement attribut MQS_MQI_KEYFILE.



COMMANDES & UTILITAIRES



Commandes magasins

- Jusqu'en version MQ 9.3, pour manipuler les magasins MQ, on disposait des commandes suivantes :
 - strmqikm : interface graphique
 - runmqckm : CLI
 - runmqakm : CLI
- A partir de la version 9.4, on va utiliser :
 - runmqakm
 - runmqktool



Commandes magasins

- runmqakm
 - Permet :
 - la création de CSR
 - l'import / export de certificats
 - la gestion des magasins CMS & PKCS12
- runmqktool
 - Permet :
 - la gestion des magasins JKS, JCEKS & PKCS12
 - La conversion pkcs12 → JKS
 - Nécessite le JRE IBM
 - Utilise le keytool du JRE



- **dspmqcert** : Surveillance expiration certificats
- **Display MQ Certificates**
- Affichage des informations du certificat d'un Queue Manager
 - Démarré ou non
- Affiche :
 - Les caractéristiques du certificat
 - La période de validité
 - Le nombre de jours avant (ou après !) l'expiration
- Disponible depuis 9.4.1



Syntaxe

`dspmqcrt -h`

Usage: `dspmqcrt [-h] [-o <output>] [-d <days>] [-m <pattern>] [-e] [-f <path>]`

Options:

- h this help
- o output: <json|csv|text> default text
- d days: print certs due to expire within these days
- m pattern: match queue manager names, ie. QM1, ERR*, default *
- a include CA intermediate certificates
- e exit error (127) if -d specified and there are certs expiring
- f path: location of INI file mapping QM names and keystore location



Variante « -a »

```
dspmcert -m DC103 -a
```

```
Queue Manager Name: DC103
```

```
Label:          Racine_DC_2021
Type:           Root CA
Version:        3
Serial:         008b87dc81e480ab92
Subject:        CN=CA_OFDC_2021_A,EMAIL=pki_of21@demey-
consulting.fr,OU=Infra,O=PKI OF DC,L=Paris,C=FR
Issuer:         CN=CA_OFDC_2021_A,EMAIL=pki_of21@demey-
consulting.fr,OU=Infra,O=PKI OF DC,L=Paris,C=FR
Keystore Path:  D:\MQ_data\qmgrs\DC103\ssl\key.kdb
Key Size:       2048
Not Before:     2020-11-14 22:40:01
Not After:      2030-11-12 22:40:01
Expiry Days:    2058
```

```
Label:          DC103
Type:           End entity
Version:        3
Serial:         23
Subject:        CN=DC103,OU=Tests 2024,O=Demey
Consulting,L=Paris,C=FR
Issuer:         CN=CA_OFDC_2021_A,EMAIL=pki_of21@demey-
consulting.fr,OU=Infra,O=PKI OF DC,L=Paris,C=FR
Keystore Path:  D:\MQ_data\qmgrs\DC103\ssl\key.kdb
Key Size:       2048
Not Before:     2023-12-30 18:59:09
Not After:      2025-02-14 18:59:09
Expiry Days:    -39
```



Test de lien SSL MQ Client

- Exemple `amqssslc`
- Paramètres :
 - -m QmgrName
 - -c ChannelName
 - -x ConnName
 - -k KeyReposFileName
 - -s CipherSpec
 - -l CertLabel
 - ...
- Permet de tester de manière indépendante un environnement TLS



Autres points (9.4.0)

- TLS Certificate bypass option
 - Permet à un client MQ de ne pas vérifier le certificat envoyé par le serveur
 - Nouvelle valeur pour Certificate validation policy : NONE
 - SET MQCERTVPOL=NONE
 - CertificateValPolicy=NONE

<https://www.ibm.com/docs/en/ibm-mq/9.4?topic=mq-configuring-certificate-validation-policies-in>



Post Quantum Cryptography



- Impact de l'informatique quantique sur le chiffrement classique
- **Harvest Now, Decrypt Later**
- Paramètres :
 - Magasin : `runmqakm -keydb -pqc true`
 - Channels: `SSLQS(NONE | MLKEM | HYBMLKEM)`
 - `SSLQSR(OPTIONAL | REQUIRED)`

<https://demey-consulting.fr/ibm-mq-et-pqc-post-quantum-cryptography/>



Recommandations

- Pas de certificats autosigné
- Durée de validité 13 mois si CA privée
- Suite à un passage en MQ 9.4, remplacer `runmqckm` par `runmqakm`
- Si magasin CMS, remplacer le fichier `.sth` par `KEYRPWD` en MQ 9.4 ou mieux migrer en `PKCS12`
- Utiliser les alias de cipherspec (`ANY_TLS13_OR_HIGHER` par exemple)



La formation avec Demey Consulting



- Un organisme de formation déclaré et « Datadocké »
- Un catalogue de modules sur WebSphere Application Server et IBM MQ
- Plus de 50 modules MQ disponibles (1500 slides)
- Des supports de cours totalement francisés et au dernier niveau technique (MQ version 10.0.1)
- Des travaux pratiques sur Windows, Linux et IBM i avec les corrigés
- Des filières prédéfinies de 2 à 5 jours, ou à la carte.

<https://demey-consulting.fr/formationMQ>