SSL, TLS et IBM MQ
Quel chiffrement utiliser ?

French MQ User Groupe du 10 Novembre 2023

Une production : Demey Consulting

Version 1.0 - Novembre 2023

Avertissement

- Cet article traite des différents types de cipherspecs existants, et des critères de choix pour leur utilisation.
- Il s'agit d'un article de vulgarisation sur la sécurité IBM MQ
- Pour une meilleure compréhension, certains éléments ont été simplifiés
- Une lecture attentive et complète de la documentation IBM est fortement conseillée avant toute décision de mise en œuvre

Chiffrement

- Objectif : Protection des flux MQ
  - Lors du transit via des canaux Client, DQM ou Cluster
  - Confidentialité, non modification, identification des partenaires
- Moyens :
  - Certificats SSL
  - Dans des magasins de certificats
  - CipherSpec
    - A spécifier dans les canaux à protéger

SSL

- SSL (Secure Sockey Layer)
  - Un des premiers protocoles de chiffrement utilisé à grande échelle
  - Aujourd'hui remplacé par TLS
  - On parle toujours de "chiffrement SSL" et de "certificats SSL".
- Versions successives :
  - SSL V1 : développé par Netscape en 1994, jamais déployée
  - SSL V2 (1995)
  - **SSL V3** (1996) : Première version de SSL supportée par MQSeries
    - MQ version 5.3 en 2002 (avec le FixPack 4 !)

SSL

- SSL V3 a été le protocole de chiffrement le plus utilisé
- Officiellement déprécié (deprecated) en 2014, suite à la découverte de la faille POODLE.
- L'utilisation de SSL V3 est particulièrement déconseillée avec MQ, cette version est dépréciée depuis MQ Version 8.0.0.2.

https://www.ibm.com/docs/en/ibm-mq/9.3?topic=cipherspecs-deprecated

TLS

- TLS (Transport Layer Security)
  - Successeur naturel de SSL
- Versions successives :
  - TLS 1.0 (1999) : Largement utilisé dans le monde MQ, encore aujourd'hui)
  - TLS 1.1 (2006) : Pas de support par MQ
  - **TLS 1.2** (2008) : Disponible à partir de MQ 9.0.0.3 & 9.0.5
  - **TLS 1.3** (2018) : Disponible à partir de MQ 9.1.4

TLS

- TLS 1.0 est déprécié depuis MQ 9.2
  - Eviter son utilisation avec MQ
  - Arrêt de support par certain OS (RHEL 9 par exemple)
  - Planifier son remplacement le cas échéant
- En environnement MQ :
  - TLS 1.2 (minima)
    - MQ 9.1+
  - TLS 1.3 (de plus en plus)
    - MQ 9.2+
- A partir de TLS 1.2 :
  - Taille de clé 2048 minimum et signature SHA256+ pour le certificat

CipherSpec

(ou CipherSuite dans la littérature non-IBM)

- Définit une combinaison d'algorithmes qui vont être utilisés pour le chiffrement
- Trois algorithmes différents :
  - Algorithme d'échange de clés et d'authentification
  - Algorithme de chiffrement des données
  - Algorithme de hachage des données (MAC - Message Authentication Code)
- Exemple : TLS_RSA_WITH_AES_128_CBC_SHA
  - RSA : protocole d'échange des clés
  - Chiffrement AES, clé de 128 bits avec Cipher Block Chaining
  - SHA (en fait SHA-1) pour le MAC

CipherSpec

- En fonction de la combinaison d'algorithmes utilisés pour un cipherspec :
  - Type SSL V3, TLS 1.0, TLS 1.2, ...
- IBM MQ :
  - Cipherspec précisé canal par canal
- Canaux Java, JMS et MQTT
  - Ciphersuite (même principe)
- Disponibilité des cipherspecs en fonction :
  - De la version de MQ
  - De la plateforme (et de ses réglages - iSeries)

→ Planifier !

Evolution du nombre de cipherspecs disponibles

Version MQ	Cipherspec SSL V3	Cipherspec TLS 1.0	Cipherspec TLS 1.2	Cipherspec TLS 1.3	Total Cipherspec
MQ 7.5	10	3	19	0	32
MQ 8.0.0.3	0	2	19	0	21
MQ 9.0	0	2	16	0	18
MQ 9.1	0	0	12	5	17
MQ 9.2	0	0	12	5	17
MQ 9.3	0	0	12	5	17

Extrait de https://www.ibm.com/docs/en/ibm-mq/9.0?topic=messages-enabling-cipherspecs

Platform support	CipherSpec name	Protocol used	Data integrity	Encryption algorithm	Encryption bits
ULW z/OS	TLS_RSA_WITH_AES_128_CBC_SHA	TLS 1.0	SHA-1	AES	128
ULW z/OS	TLS_RSA_WITH_AES_256_CBC_SHA	TLS 1.0	SHA-1	AES	256
All	ECDHE_ECDSA_AES_128_CBC_SHA256	TLS 1.2	SHA-256	AES	128
All	ECDHE_ECDSA_AES_256_CBC_SHA384	TLS 1.2	SHA-384	AES	256
Multi	ECDHE_ECDSA_AES_128_GCM_SHA256	TLS 1.2	AEAD AES-128 GCM	AES	128
Multi	ECDHE_ECDSA_AES_256_GCM_SHA384	TLS 1.2	AEAD AES-128 GCM	AES	256
All	ECDHE_RSA_AES_128_CBC_SHA256	TLS 1.2	SHA-256	AES	128
All	ECDHE_RSA_AES_256_CBC_SHA384	TLS 1.2	SHA-384	AES	256
Multi (LTS) All (V9.0.5 and later)	ECDHE_RSA_AES_128_GCM_SHA256 4	TLS 1.2	AEAD AES-128 GCM	AES	128
Multi (LTS) All (V9.0.5 and later)	ECDHE_RSA_AES_256_GCM_SHA384	TLS 1.2	AEAD AES-128 GCM	AES	SHA384
IBM i	ECDHE_ECDSA_RC4_128_SHA256	TLS 1.2	AEAD AES-128 GCM	AES	SHA

TLS 1.2, extrait de :
https://www.ibm.com/docs/en/ibm-mq/9.3?topic=messages-enabling-cipherspecs

Platform support	CipherSpec name	Protocol used	Data integrity	Encryption algorithm
All	TLS_RSA_WITH_AES_128_CBC_SHA2569	TLS 1.2	SHA-256	AES (128)
All	TLS_RSA_WITH_AES_256_CBC_SHA256	TLS 1.2	SHA-256	AES (256)
All	TLS_RSA_WITH_AES_128_GCM_SHA256	TLS 1.2	SHA-256 and AEAD GCM	AES (128)
All	TLS_RSA_WITH_AES_256_GCM_SHA384	TLS 1.2	SHA-384 and AEAD GCM	AES (256)
All	ECDHE_ECDSA_AES_128_CBC_SHA256	TLS 1.2	SHA-256	AES (128)
All	ECDHE_ECDSA_AES_256_CBC_SHA384	TLS 1.2	SHA-384	AES (256)
All	ECDHE_RSA_AES_128_CBC_SHA256	TLS 1.2	SHA-256	AES (128)
All	ECDHE_RSA_AES_256_CBC_SHA384	TLS 1.2	SHA-384	AES (256)
Multi	ECDHE_ECDSA_AES_128_GCM_SHA256	TLS 1.2	SHA-256 and AEAD GCM	AES (SHA384)
Multi	ECDHE_ECDSA_AES_256_GCM_SHA384	TLS 1.2	SHA-384 and AEAD GCM	AES (SHA384)
All	ECDHE_RSA_AES_128_GCM_SHA256	TLS 1.2	SHA-256 and AEAD GCM	AES (128)
All	ECDHE_RSA_AES_256_GCM_SHA384	TLS 1.2	AEAD AES-128 GCM	AES (SHA384)

Cipherspecs Alias

- IBM MQ : pas de liste de cipherspec à utiliser dans un canal
  - Contrairement à d'autres logiciels (serveur HTTP, …)
- Même cipherspec défini (et utilisé) par les deux extrémités
- Depuis MQ 911, possibilité d'utiliser des alias de cipherspec
  - Exemple : ANY_TLS12
  - Résultat :
    - Négociation d'un cipherspec entre les deux extrémités
    - En restant dans le TLS 1.2
- Autres alias disponibles :
  - ANY_TLS13_OR_HIGHER, ANY_TLS13, ANY_TLS12_OR_HIGHER, ANY_TLS12 , ANY
  - Dans tous les cas, le cipherspec négocié fera partie de ceux supportés et autorisés par les deux extrémités du canal.

Ré-activation de cipherpecs dépréciés

https://www.ibm.com/docs/en/ibm-mq/9.3?topic=cipherspecs-deprecated

- Peut être nécessaire dans le cadre de migrations
- Doit être une situation temporaire
- Deux solutions :
  - Variable d'environnement AMQ_SSL_WEAK_CIPHER_ENABLE

```
export AMQ_SSL_WEAK_CIPHER_ENABLE=ECDHE_RSA_RC4_128_SHA256
```

    - nom de cipherspec
    - Liste de cipherspec
    - ALL (SSL V3, TLS 1.0, ...)
  - Stanza dans qm.ini :

```
SSL:
 AllowTLSV1=Y
 AllowWeakCipherSpec=ECDHE_RSA_RC4_128_SHA256
```

SSL, TLS, cipherspecs et qm.ini

https://www.ibm.com/docs/en/ibm-mq/9.3?topic=qmini-ssl-stanza-file

- TLS 1.3 activé par défaut pour les QM créés à partir de MQ 9.2 (MQ 9.1.4)
- Fichier qm.ini à modifier pour les QM migrés :

```
SSL:
   AllowTLSV13=Yes
```

- Incompatible avec AllowWeakCipherSpec
- Possibilité de limiter les ciphers disponibles :
  - AllowedCipherSpecs=*name*|*name list*|ALL
  - Désactive AllowWeakCipherSpec

FIPS & Suite B

- FIPS et Suite B sont des listes de cipherspecs qui correspondent à un usage spécifique
- FIPS
  - Federal Information Processing Standard
  - Standards de sécurité US
  - Peut être imposé par le partenaire
- Suite B
  - Défini une liste d'algorithmes de chiffrement en TLS 1.2
  - Les certificats doivent être signés avec une clé ECDSA

Perfect Forward Secrecy

- "PFS", "Confidentialité persistante"
- Propriété cryptographique
- Garanti que la découverte de la clé privée à un instant T ne permet pas le décodage des échanges antérieurs
- Implicite avec TLS 1.3, possible en TLS 1.2
- De plus en plus demandé par les SSI
- Nécessite l'utilisation des algorithmes de chiffrement ECDHE ou DHE

Choix du cipherspec pour les canaux MQ

- Liens DQM, MQ Client, MQ Cluster
- Trois cas :
  - Tous les participants peuvent faire du TLS 1.3
  - Tous les participants peuvent faire au moins du TLS 1.2
  - Un des participants ne peut pas faire de TLS 1.2

TLS 1.3

- ANY_TLS13 , ANY_TLS13_OR_HIGHER
  - Limitations avec Clients MQ + non-IBM JRE
    - Ajouter -Dcom.ibm.mq.cfg.useIBMCipherMappings=false
- TLS_CHACHA20_POLY1305_SHA256
  - Par défaut si ANY_TLS13*
  - Performances +

TLS 1.2

- ANY_TLS12, ANY_TLS12_OR_HIGHER
  - Permettra le passage à TLS 1.3
  - Limitations avec Clients MQ + non-IBM JRE
- ECDHE_RSA_AES_128_GCM_SHA256
  - Compatible MQ Explorer
- ECDHE_ECDSA_AES_128_CBC_SHA256
  - PFS compliant

Un des participants ne peut pas faire de TLS 1.2

- Planifier la migration vers une version de MQ le permettant
- Réactivation de TLS 1.0
  - Incompatible avec TLS 1.3
- Modification du qm.ini + :
  - TLS_RSA_WITH_AES_128_CBC_SHA
    - TLS 1.0
    - Non compatible iSeries
  - TLS_RSA_WITH_DES_CBC_SHA
    - TLS 1.0

Autres points

- Contraintes sur les certificats
  - Suite à une migration TLS 1.0 → TLS 1.2
  - Avec les cipherspecs elliptiques
- Attention aux clients MQ
  - Souvent back level
  - Installation incomplète

Questions ?

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 (1700 slides)
- Des supports de cours totalement francisés et au dernier niveau technique (MQ version 9.3.4)
- Des travaux pratiques sur Windows, Linux et iSeries avec les corrigés
- Des filières prédéfinies de 2 à 5 jours, ou à la carte

https://demey-consulting.fr/formationMQ