Configurer ESA pour privilégier la confidentialité de transmission parfaite

Table des matières

Introduction

Ce document décrit comment configurer la préférence pour PFS dans les connexions chiffrées TLS sur l'appliance de sécurité de la messagerie (ESA).

Conditions préalables

Exigences

Cisco recommande que vous ayez connaissance des protocoles SSL (Secure Sockets Layer) et TLS (Transport Layer Security).

Composants utilisés

Les informations contenues dans ce document sont basées sur AsyncOS for Email version 9.6 et ultérieures.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l’incidence possible des commandes.

Informations générales

ESA prend en charge le protocole PFS (Perfect Forward Secrecy). PFS signifie que les données sont transférées sur un canal qui utilise un chiffrement symétrique avec des secrets éphémères. Même si la clé privée de l’un des hôtes ou des deux est compromise, une session précédemment enregistrée ne peut pas être déchiffrée.

Le secret n'est pas transféré via le canal. Au lieu de cela, le secret partagé est dérivé avec le

problème Diffie-Hellman (DH). Le secret est stocké uniquement dans la mémoire vive (RAM) de l'hôte pendant le délai d'expiration de la session établie ou de la régénération de clé.

ESA prend en charge DH pour l'échange de clés.

Configurer

ENTRANT - ESA fait office de serveur TLS

Ces suites de chiffrement sont disponibles sur ESA pour le trafic entrant SMTP (Simple Mail Transfer Protocol) et fournissent PFS. Dans cet exemple, la sélection de chiffrement autorise uniquement les suites de chiffrement considérées comme ÉLEVÉ ou MOYEN, utilise EDH (Ephemeral Diffie-Hellman) pour l'échange de clés et préfère TLSv1.2. La syntaxe de sélection de chiffrement se trouve après la syntaxe OpenSSL.

Chiffres avec transfert secret sur AsyncOS 9.6+ :

```
"EDH+TLSv1.2:EDH+HIGH:EDH+MEDIUM:!LOW:!EXP:!aNULL:!RC4:!DSS:!SEED:!IDEA:!MD5:!PSK:!3DES:!SRP"

List:
TLS_AES_256_GCM_SHA384  TLSv1.3 Kx=any      Au=any Enc=AESGCM(256) Mac=AEAD
TLS_CHACHA20_POLY1305_SHA256 TLSv1.3 Kx=any Au=any  Enc=CHACHA20/POLY1305(256) Mac=AEAD
TLS_AES_128_GCM_SHA256  TLSv1.3 Kx=any      Au=any  Enc=AESGCM(128) Mac=AEAD
DHE-RSA-AES256-GCM-SHA384 TLSv1.2 Kx=DH       Au=RSA  Enc=AESGCM(256) Mac=AEAD
DHE-RSA-CHACHA20-POLY1305 TLSv1.2 Kx=DH       Au=RSA  Enc=CHACHA20/POLY1305(256) Mac=AEAD
DHE-RSA-AES256-CCM      TLSv1.2 Kx=DH       Au=RSA  Enc=AESCCM(256) Mac=AEAD
DHE-RSA-AES128-GCM-SHA256 TLSv1.2 Kx=DH       Au=RSA  Enc=AESGCM(128) Mac=AEAD
DHE-RSA-AES128-CCM      TLSv1.2 Kx=DH       Au=RSA  Enc=AESCCM(128) Mac=AEAD
DHE-RSA-AES256-SHA256   TLSv1.2 Kx=DH       Au=RSA  Enc=AES(256)  Mac=SHA256
DHE-RSA-CAMELLIA256-SHA256 TLSv1.2 Kx=DH       Au=RSA  Enc=Camellia(256) Mac=SHA256
DHE-RSA-AES128-SHA256   TLSv1.2 Kx=DH       Au=RSA  Enc=AES(128)  Mac=SHA256
DHE-RSA-CAMELLIA128-SHA256 TLSv1.2 Kx=DH       Au=RSA  Enc=Camellia(128) Mac=SHA256
DHE-RSA-AES128-SHA      SSLv3 Kx=DH       Au=RSA  Enc=AES(128)  Mac=SHA1
DHE-RSA-CAMELLIA128-SHA SSLv3 Kx=DH       Au=RSA  Enc=Camellia(128) Mac=SHA1
```

Le champ Kx (échange de clés) indique que DH est utilisé pour dériver le secret.

ESA prend en charge ces chiffrements avec le paramètre sslconfig par défaut (:ALL), mais ne les préfère pas. Pour préférer les chiffrements qui offrent PFS, modifiez sslconfig et ajoutez EDH ou une combinaison telle que EDH+<nom du groupe de chiffrage> à la sélection du chiffrage.

1. Vérifiez la configuration par défaut.

```
ESA> sslconfig

sslconfig settings:
  Inbound SMTP method:  tlsv1/tlsv1.2
  Inbound SMTP ciphers:
        RC4-SHA
        RC4-MD5
        ALL
```

2. Configurez la nouvelle préférence de chiffrement entrant.

```
ESA> sslconfig

Inbound SMTP method:  tlsv1/tlsv1.2
  Inbound SMTP ciphers:
        EDH+TLSv1.2
        EDH+HIGH
        EDH+MEDIUM
        RC4-SHA
        RC4-MD5
        ALL
```

 Remarque : RC4 comme chiffrement et MD5 comme MAC sont considérés comme faibles et hérités. Évitez leur utilisation avec SSL/TLS, en particulier pour les volumes de données plus élevés sans régénération de clé.

Paramètres sslconfig recommandés pour INBOUND

Cet exemple montre une configuration recommandée qui autorise uniquement les chiffrements généralement considérés comme forts et sécurisés.

Une configuration entrante recommandée supprime RC4 et MD5, ainsi que d'autres options héritées et faibles telles que Export (EXP), Low (LOW), IDEA, SEED, 3DES, les certificats DSS, l'échange de clés anonymes (aNULL), les clés pré-partagées (PSK) et SRP. Elle désactive également l'algorithme ECDH (Elliptic Curve Diffie-Hellman) pour l'échange de clés et l'algorithme ECDSA (Elliptic Curve Digital Signature Algorithm).

1. Entrez la chaîne de chiffrement suivante dans sslconfig.

```
EDH+TLSv1.2:EDH+HIGH:EDH+MEDIUM:HIGH:MEDIUM:!ECDH:!ECDSA:!LOW:!EXP:!aNULL:!RC4:!DSS:!SEED:!IDEA:!MD5:!P
```

La chaîne entrée dans sslconfig donne cette liste de chiffrements pris en charge pour INBOUND :

```
TLS_AES_256_GCM_SHA384  TLSv1.3 Kx=any      Au=any  Enc=AESGCM(256) Mac=AEAD
TLS_CHACHA20_POLY1305_SHA256 TLSv1.3 Kx=any      Au=any  Enc=CHACHA20/POLY1305(256) Mac=AEAD
TLS_AES_128_GCM_SHA256  TLSv1.3 Kx=any      Au=any  Enc=AESGCM(128) Mac=AEAD
DHE-RSA-AES256-GCM-SHA384 TLSv1.2 Kx=DH       Au=RSA  Enc=AESGCM(256) Mac=AEAD
DHE-RSA-CHACHA20-POLY1305 TLSv1.2 Kx=DH       Au=RSA  Enc=CHACHA20/POLY1305(256) Mac=AEAD
DHE-RSA-AES256-CCM      TLSv1.2 Kx=DH       Au=RSA  Enc=AESCCM(256) Mac=AEAD
DHE-RSA-AES128-GCM-SHA256 TLSv1.2 Kx=DH       Au=RSA  Enc=AESGCM(128) Mac=AEAD
DHE-RSA-AES128-CCM      TLSv1.2 Kx=DH       Au=RSA  Enc=AESCCM(128) Mac=AEAD
DHE-RSA-AES256-SHA256   TLSv1.2 Kx=DH       Au=RSA  Enc=AES(256)  Mac=SHA256
DHE-RSA-CAMELLIA256-SHA256 TLSv1.2 Kx=DH       Au=RSA  Enc=Camellia(256) Mac=SHA256
DHE-RSA-AES128-SHA256   TLSv1.2 Kx=DH       Au=RSA  Enc=AES(128)  Mac=SHA256
DHE-RSA-CAMELLIA128-SHA256 TLSv1.2 Kx=DH       Au=RSA  Enc=Camellia(128) Mac=SHA256
DHE-RSA-AES128-SHA      SSLv3 Kx=DH       Au=RSA  Enc=AES(128)  Mac=SHA1
DHE-RSA-CAMELLIA128-SHA SSLv3 Kx=DH       Au=RSA  Enc=Camellia(128) Mac=SHA1
AES256-GCM-SHA384       TLSv1.2 Kx=RSA      Au=RSA  Enc=AESGCM(256) Mac=AEAD
AES256-CCM              TLSv1.2 Kx=RSA      Au=RSA  Enc=AESCCM(256) Mac=AEAD
AES128-GCM-SHA256       TLSv1.2 Kx=RSA      Au=RSA  Enc=AESGCM(128) Mac=AEAD
AES128-CCM              TLSv1.2 Kx=RSA      Au=RSA  Enc=AESCCM(128) Mac=AEAD
AES256-SHA256           TLSv1.2 Kx=RSA      Au=RSA  Enc=AES(256)  Mac=SHA256
CAMELLIA256-SHA256      TLSv1.2 Kx=RSA      Au=RSA  Enc=Camellia(256) Mac=SHA256
AES128-SHA256           TLSv1.2 Kx=RSA      Au=RSA  Enc=AES(128)  Mac=SHA256
CAMELLIA128-SHA256      TLSv1.2 Kx=RSA      Au=RSA  Enc=Camellia(128) Mac=SHA256
AES256-SHA              SSLv3 Kx=RSA      Au=RSA  Enc=AES(256)  Mac=SHA1
AES128-SHA              SSLv3 Kx=RSA      Au=RSA  Enc=AES(128)  Mac=SHA1
CAMELLIA128-SHA         SSLv3 Kx=RSA      Au=RSA  Enc=Camellia(128) Mac=SHA1
```

Remarque : ESA agissant en tant que serveur TLS pour le trafic entrant ne prend actuellement pas en charge l'ECDHE (Elliptic Curve Diffie-Hellman Ephemeral) pour l'échange de clés ou les certificats ECDSA.

SORTANT - ESA agit comme client TLS

Pour le trafic SMTP sortant, ESA prend en charge les certificats ECDHE et ECDSA en plus des fonctionnalités entrantes.

Remarque : Les certificats ECC (Elliptic Curve Cryptography) qui utilisent ECDSA ne sont pas très répandus.

Lorsqu'un e-mail sortant est envoyé, ESA est le client TLS. Un certificat client TLS est facultatif. Si le serveur TLS n'exige pas qu'ESA fournisse un certificat client ECDSA, ESA peut poursuivre une session sécurisée ECDSA. Lorsque le client TLS ESA est invité à fournir son certificat, il fournit le certificat RSA configuré pour la direction sortante.

Mise en garde : Le magasin de certificats d'autorité de certification approuvée préinstallé (liste système) sur ESA n'inclut pas les certificats racines ECC (ECDSA). Vous pouvez ajouter manuellement des certificats racines ECC approuvés à la liste personnalisée pour

⚠ rendre la chaîne de confiance ECC vérifiable.

Pour préférer les chiffrements DHE/ECDHE qui offrent PFS, modifiez la sélection de chiffrement sslconfig.

1. Ajoute cette chaîne à la sélection de chiffrement en cours.

```
"EDH+TLSv1.2:ECDH+TLSv1.2:EDH+HIGH:EDH+MEDIUM:ECDH+HIGH:ECDH+MEDIUM"
```

Paramètres sslconfig recommandés pour OUTBOUND

Cet exemple montre une configuration recommandée qui autorise uniquement les chiffrements généralement considérés comme forts et sécurisés.

```
EDH+TLSv1.2:ECDH+TLSv1.2:EDH+HIGH:EDH+MEDIUM:ECDH+HIGH:ECDH+MEDIUM:HIGH:MEDIUM:!LOW:!EXP:!aNULL:!RC4:!D
```

La chaîne entrée dans sslconfig aboutit à cette liste de chiffrements pris en charge pour OUTBOUND :

```
TLS_AES_256_GCM_SHA384  TLSv1.3 Kx=any      Au=any  Enc=AESGCM(256) Mac=AEAD
TLS_CHACHA20_POLY1305_SHA256 TLSv1.3 Kx=any      Au=any  Enc=CHACHA20/POLY1305(256) Mac=AEAD
TLS_AES_128_GCM_SHA256  TLSv1.3 Kx=any      Au=any  Enc=AESGCM(128) Mac=AEAD
DHE-RSA-AES256-GCM-SHA384 TLSv1.2 Kx=DH Au=RSA Enc=AESGCM(256) Mac=AEAD
DHE-RSA-AES256-SHA256 TLSv1.2 Kx=DH Au=RSA Enc=AES(256) Mac=SHA256
DHE-RSA-AES128-GCM-SHA256 TLSv1.2 Kx=DH Au=RSA Enc=AESGCM(128) Mac=AEAD
DHE-RSA-AES128-SHA256 TLSv1.2 Kx=DH Au=RSA Enc=AES(128) Mac=SHA256
ECDHE-RSA-AES256-GCM-SHA384 TLSv1.2 Kx=ECDH Au=RSA Enc=AESGCM(256) Mac=AEAD
ECDHE-ECDSA-AES256-GCM-SHA384 TLSv1.2 Kx=ECDH Au=ECDSA Enc=AESGCM(256) Mac=AEAD
ECDHE-RSA-AES256-SHA384 TLSv1.2 Kx=ECDH Au=RSA Enc=AES(256) Mac=SHA384
ECDHE-ECDSA-AES256-SHA384 TLSv1.2 Kx=ECDH Au=ECDSA Enc=AES(256) Mac=SHA384
ECDHE-RSA-AES128-GCM-SHA256 TLSv1.2 Kx=ECDH Au=RSA Enc=AESGCM(128) Mac=AEAD
ECDHE-ECDSA-AES128-GCM-SHA256 TLSv1.2 Kx=ECDH Au=ECDSA Enc=AESGCM(128) Mac=AEAD
ECDHE-RSA-AES128-SHA256 TLSv1.2 Kx=ECDH Au=RSA Enc=AES(128) Mac=SHA256
ECDHE-ECDSA-AES128-SHA256 TLSv1.2 Kx=ECDH Au=ECDSA Enc=AES(128) Mac=SHA256
DHE-RSA-AES256-SHA SSLv3 Kx=DH Au=RSA Enc=AES(256) Mac=SHA1
DHE-RSA-CAMELLIA256-SHA SSLv3 Kx=DH Au=RSA Enc=Camellia(256) Mac=SHA1
DHE-RSA-AES128-SHA SSLv3 Kx=DH Au=RSA Enc=AES(128) Mac=SHA1
DHE-RSA-CAMELLIA128-SHA SSLv3 Kx=DH Au=RSA Enc=Camellia(128) Mac=SHA1
ECDHE-RSA-AES256-SHA SSLv3 Kx=ECDH Au=RSA Enc=AES(256) Mac=SHA1
ECDHE-ECDSA-AES256-SHA SSLv3 Kx=ECDH Au=ECDSA Enc=AES(256) Mac=SHA1
ECDHE-RSA-AES128-SHA SSLv3 Kx=ECDH Au=RSA Enc=AES(128) Mac=SHA1
ECDHE-ECDSA-AES128-SHA SSLv3 Kx=ECDH Au=ECDSA Enc=AES(128) Mac=SHA1
AES256-GCM-SHA384 TLSv1.2 Kx=RSA Au=RSA Enc=AESGCM(256) Mac=AEAD
AES256-SHA256 TLSv1.2 Kx=RSA Au=RSA Enc=AES(256) Mac=SHA256
```

```
AES256-SHA SSLv3 Kx=RSA Au=RSA Enc=AES(256) Mac=SHA1
CAMELLIA256-SHA SSLv3 Kx=RSA Au=RSA Enc=Camellia(256) Mac=SHA1
AES128-GCM-SHA256 TLSv1.2 Kx=RSA Au=RSA Enc=AESGCM(128) Mac=AEAD
AES128-SHA256 TLSv1.2 Kx=RSA Au=RSA Enc=AES(128) Mac=SHA256
AES128-SHA SSLv3 Kx=RSA Au=RSA Enc=AES(128) Mac=SHA1
CAMELLIA128-SHA SSLv3 Kx=RSA Au=RSA Enc=Camellia(128) Mac=SHA1
```

Vérifier

Suivez ces étapes pour vérifier que ESA préfère les suites de chiffrement PFS.

1. Exécutez sslconfig et vérifiez les chaînes de chiffrement entrantes et sortantes configurées.
2. Vérifiez que la sélection de chiffrement inclut EDH pour la préférence DHE et, pour le trafic sortant, les entrées ECDH, le cas échéant.
3. Examinez la liste de chiffrement résultante affichée par sslconfig et confirmez que les suites de chiffrement DHE ou ECDHE apparaissent avant les suites de chiffrement d'échange de clés RSA.
4. Testez une connexion TLS et vérifiez que la suite de chiffrement négociée utilise DHE ou ECDHE lorsque l'homologue TLS distant la prend en charge.

Dépannage

Utilisez ces vérifications si ESA ne négocie pas de suite de chiffrement PFS.

- Vérifiez que la chaîne de chiffrement sslconfig configurée inclut EDH pour le trafic entrant et les valeurs EDH ou ECDH requises pour le trafic sortant.
- Vérifiez que le serveur ou le client TLS distant prend en charge les suites de chiffrement DHE ou ECDHE. Si l'homologue ne les prend pas en charge, ESA peut négocier une suite de chiffrement non-PFS qui est toujours autorisée par la configuration.
- Pour les sessions ECDSA sortantes, vérifiez que l'homologue ne nécessite pas de certificat client ECDSA. ESA présente le certificat RSA configuré pour la direction sortante lorsqu'un certificat client est demandé.
- Pour la validation de l'approbation ECC sortante, vérifiez que les certificats racine ECC requis sont présents dans la liste personnalisée s'ils ne sont pas inclus dans la liste système.

Informations connexes

- Chiffres OpenSSL
- Cryptographie de nouvelle génération
- Assistance technique et documentation Cisco

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.