# 配置ESA以偏好完全向前保密

## 目录

## 简介

本文档介绍如何在邮件安全设备(ESA)上配置TLS加密连接中的PFS首选项。

## 先决条件

### 要求

Cisco建议您了解安全套接字层(SSL)/传输层安全(TLS)。

### 使用的组件

本文档中的信息基于AsyncOS for Email版本9.6及更高版本。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

## 背景信息

ESA支持完全向前保密(PFS)。PFS意味着数据将通过使用具有临时机密的对称加密的信道传输。即使一台或两台主机上的私钥已被破坏，之前记录的会话仍无法解密。

密钥不会通过信道传输。相反，共享密钥是通过Diffie-Hellman(DH)问题派生的。密钥仅在已建立的会话或密钥重新生成超时期间存储在主机随机访问存储器(RAM)中。

ESA支持DH进行密钥交换。

## 配置

## 入站 — ESA用作TLS服务器

这些密码套件在ESA上可用于入站简单邮件传输协议(SMTP)流量并提供PFS。在本示例中，密码选择仅允许被视为HIGH或MEDIUM的密码套件，使用短暂Diffie-Hellman(EDH)进行密钥交换，并首选TLSv1.2。密码选择语法在OpenSSL语法之后。

在AsyncOS 9.6+上具有前向保密的密码：

```
"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
```

Kx（密钥交换）字段显示DH用于派生密钥。

ESA使用默认sslconfig设置(:ALL)支持这些密码，但不首选它们。要首选提供PFS的密码，请修改sslconfig并将EDH或组合（例如EDH+<cipher or cipher group name>）添加到密码选择中。

1. 检查默认配置。

```
ESA> sslconfig

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

2.配置新的入站密码首选项。

```
ESA> sslconfig

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



注意：RC4作为密码，而MD5作为MAC被视为弱和传统。避免将其用于SSL/TLS，尤其是对于没有密钥重新生成的更高数据量。

## 建议的INBOUND的sslconfig设置

此示例显示仅允许一般认为强而安全的密码的推荐配置。

推荐的入站配置可删除RC4和MD5以及其他传统和弱选项，例如导出(EXP)、低(LOW)、IDEA、SEED、3DES、DSS证书、匿名密钥交换(aNULL)、预共享密钥(PSK)和SRP。它还禁用密钥交换的椭圆曲线Diffie-Hellman(ECDH)和椭圆曲线数字签名算法(ECDSA)。

1. 在sslconfig中输入下一个密码字符串。

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

在sslconfig中输入的字符串将生成此受支持的入站密码列表：

```
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
```

注意：用作入站流量的TLS服务器的ESA当前不支持密钥交换或ECDSA证书的Elliptic Curve Diffie-Hellman Ephemeral(ECDHE)。

## 出站 — ESA充当TLS客户端

对于出站SMTP流量，ESA除支持入站功能外，还支持ECDHE和ECDSA证书。

注意：使用ECDSA的椭圆曲线加密(ECC)证书并未被广泛采用。

发送出站邮件时，ESA是TLS客户端。TLS客户端证书是可选的。如果TLS服务器不需要ESA提供ECDSA客户端证书，则ESA可以继续执行ECDSA安全会话。当ESA作为TLS客户端请求其证书时，它提供已配置的RSA证书用于出站方向。

警告：ESA上预安装的受信任CA证书存储（系统列表）不包括ECC(ECDSA)根证书。您可能需要手动将受信任ECC根证书添加到自定义列表，以使ECC信任链可验证。

要首选提供PFS的DHE/ECDHE密码，请修改sslconfig密码选择。

1. 将此字符串添加到当前密码选择。

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

## 建议的OUTBOUND的sslconfig设置

此示例显示仅允许一般认为强而安全的密码的推荐配置。

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

在sslconfig中输入的字符串将生成以下支持OUTBOUND密码的列表：

```
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
```

# 验证

使用以下步骤验证ESA是否优先于PFS密码套件。

1. 运行sslconfig并检查已配置的入站和出站密码字符串。
2. 确认密码选择包括DHE首选项的EDH，以及出站流量的ECDH条目（如果适用）。
3. 查看sslconfig显示的结果密码列表，并确认DHE或ECDHE密码套件出现在RSA密钥交换密码套件之前。
4. 测试TLS连接，并确认当远程TLS对等体支持时，协商的密码套件使用DHE或ECDHE。

# 故障排除

如果ESA不协商PFS密码套件，请使用这些检查。

- 验证已配置的sslconfig密码字符串包括EDH（用于入站流量）和所需EDH（用于出站流量）或ECDH（用于出站流量）。
- 确认远程TLS服务器或TLS客户端支持DHE或ECDHE密码套件。如果对等体不支持它们，ESA可以协商配置仍允许的非PFS密码套件。
- 对于基于ECDSA的出站会话，请确认对等设备不需要ECDSA客户端证书。当请求客户端证书时，ESA为出站方向提供已配置的RSA证书。
- 对于出站ECC信任验证，如果所需的ECC根证书未包含在系统列表中，请确认其存在于自定义列表中。

## 相关信息

- OpenSSL密码
- 下一代加密
- Cisco技术支持和文档

关于此翻译
思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。