利用EEM自动向用户发送安全邮件

目录

简介

本文档介绍在Cisco IOS® XE中的嵌入式事件管理器(EEM)中使用“邮件服务器”(mail server)操作所需的过程，以使用端口587上的传输层安全(TLS)将安全邮件发送到简单邮件传输协议(SMTP)服务器。

在此过程中，您可能会遇到许多警告，因此编写本文的目的是记录完成此任务所需的步骤。

使用案例

许多客户认为，在特定事件发生后自动接收电邮通知很有用。 EEM子系统是网络事件检测和板载自动化的强大工具，并且可提供在Cisco IOS XE设备上自动发送邮件通知的有效方法。 例如，您可能希望监控IPSLA跟踪，并响应指示状态更改的系统日志，采取某种操作并通过电子邮件向网络管理员发出事件警报。 这种“电邮通知”的想法可以应用到许多其他场景中，以吸引对您要突出显示的任何特定事件的关注。

背景

PEM代表“隐私增强邮件”，是一种通常用于表示证书和密钥的格式。 这是Cisco IOS XE设备使用的证书格式。 安全应用（如HTTPS或安全SMTP）通常具有“堆叠PEM”，其中涉及多个证书，包括：

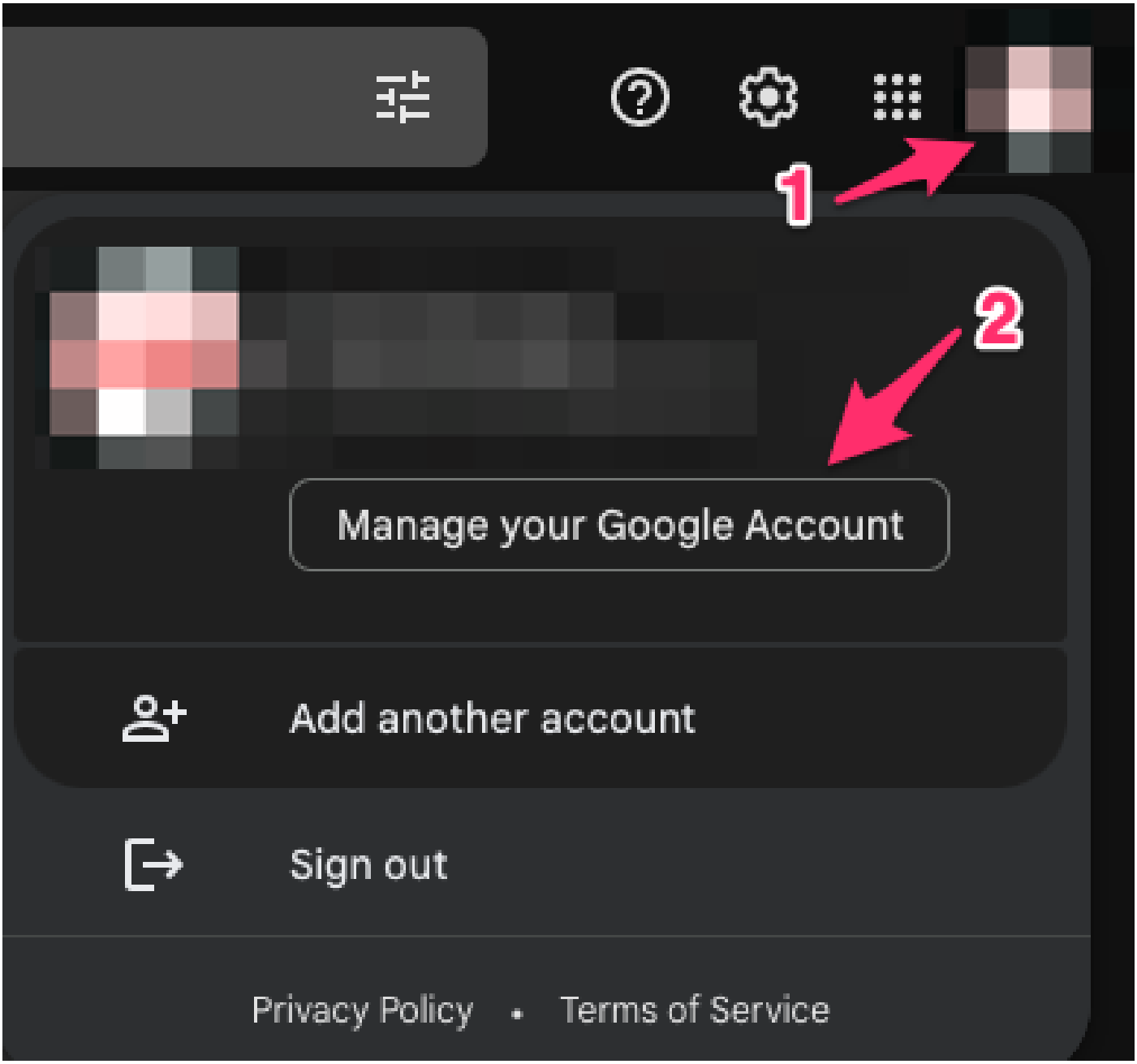
- 根证书
- 签名（中间）证书
- 最终用户（或服务器）证书

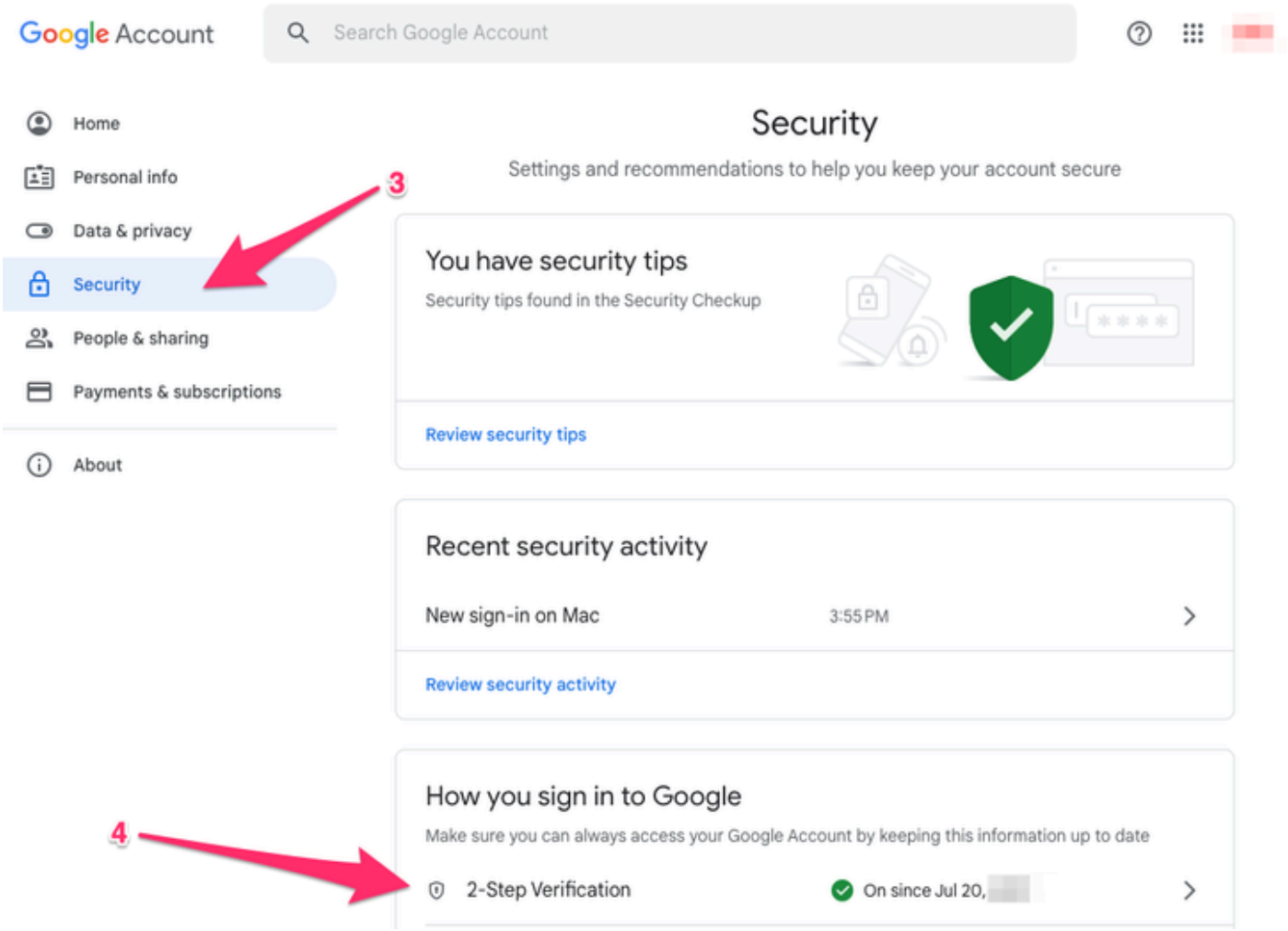
Gmail帐户设置

本文以Google的SMTP服务为例。 前提条件是您之前设置了Gmail帐户。

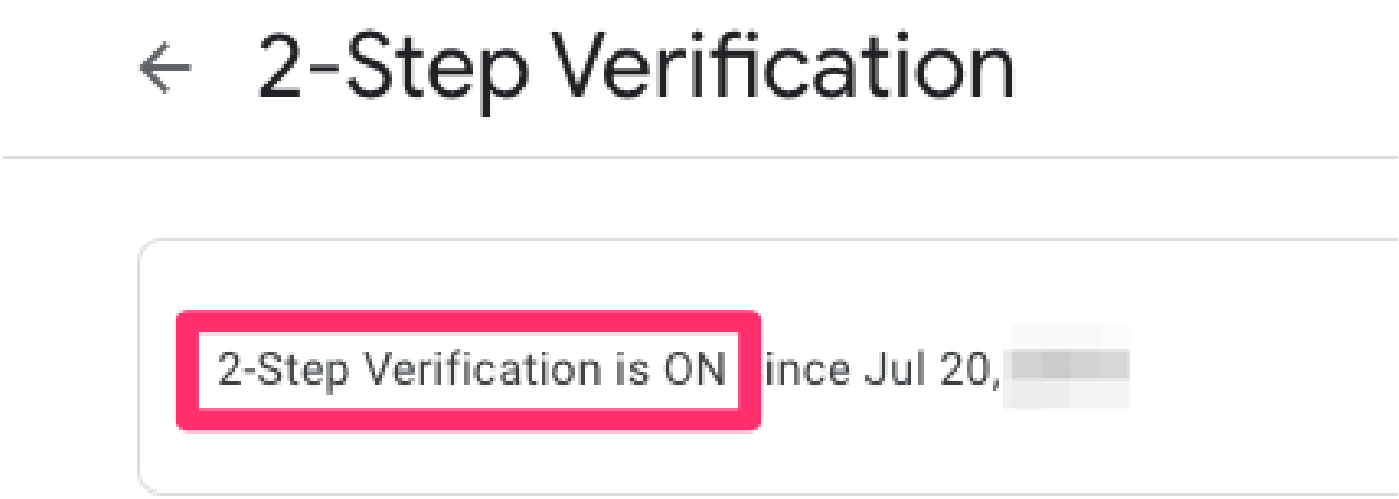
Google允许您从远程客户端向Gmail发送电子邮件。 Gmail中曾经有一个针对“不安全的应用”的设置，如果Google端不允许该设置，该应用将面临错误。 该设置已删除，取而代之的是“安全应用程序”选项。可通过以下方式访问：

mail.google.com >点击您的简档(#1) >管理您的Google帐户(#2) >安全(#3) >您如何登录Google > 2步骤验证(#4)





在此页面上，确保“两步验证”处于打开状态。



然后，您可以向下滚动到“应用密码”，让Gmail生成一个密码，该密码可用于从不支持2步验证的应用程序登录到您的Google帐户。

App passwords

App Passwords aren't recommended and are unnecessary in most cases. To help keep your account secure, use "Sign in with Google" to connect apps to your Google Account.

App passwords

None

← App passwords

App passwords let you sign in to your Google Account from apps on devices that don't support 2-Step Verification. You'll only need to enter it once so you don't need to remember it. Learn more

You don't have any app passwords.

Select the app and device you want to generate the app password for.

Mail | Select device

- **iPhone**
- **iPad**
- **BlackBerry**
- **Mac**
- **Windows Phone**
- **Windows Computer**
- **Other *(Custom name)***

GENERATE

App passwords

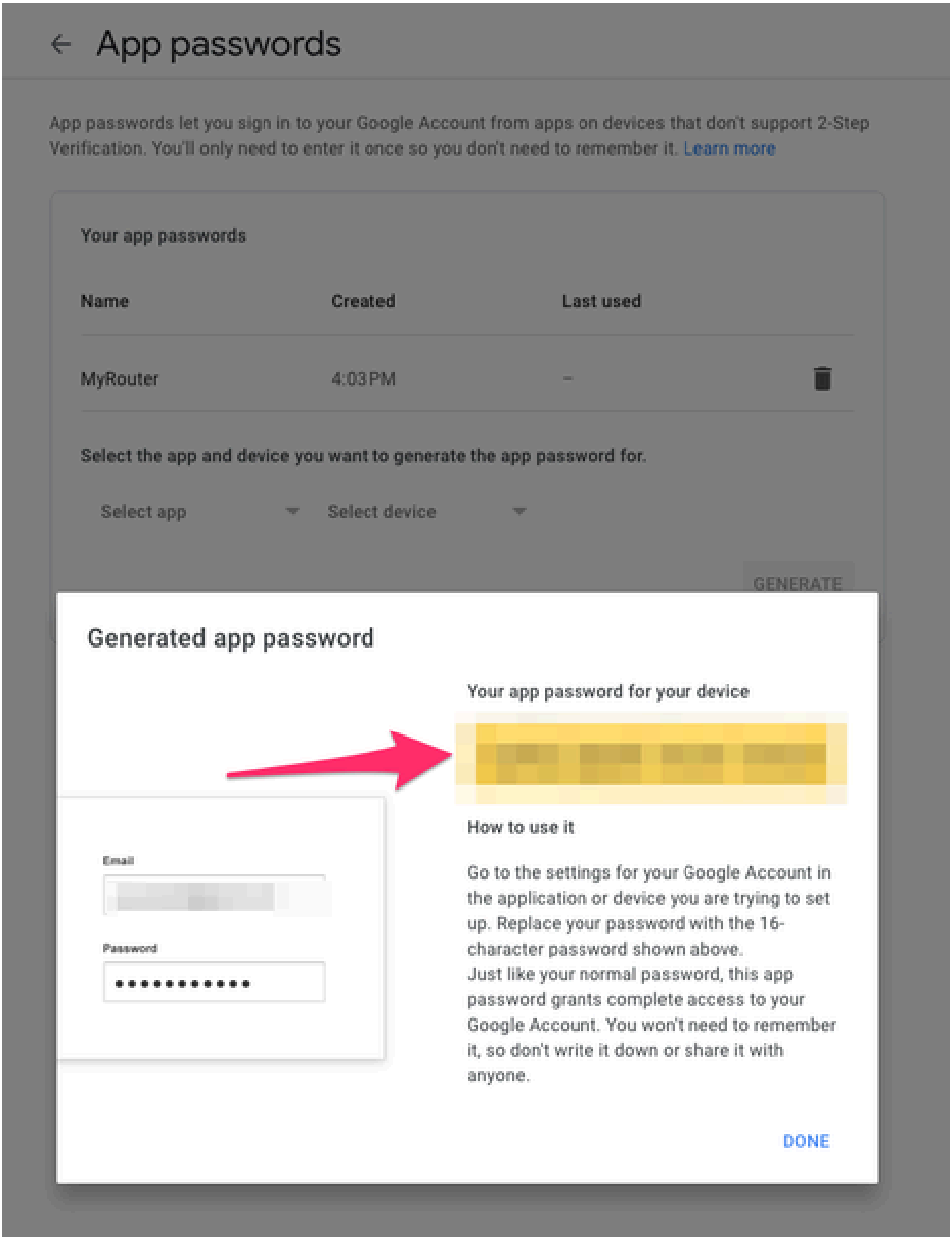
App passwords let you sign in to your Google Account from apps on devices that don't support 2-Step Verification. You'll only need to enter it once so you don't need to remember it. Learn more

You don't have any app passwords.

Select the app and device you want to generate the app password for.

MyRouter

GENERATE



此屏幕截图中的16个字符的应用程序密码被模糊了，因为它与个人Gmail帐户相关联。

现在您已有Gmail应用程序密码，您可以将此密码与Gmail帐户名称一起用作电子邮件服务器，以转

发电子邮件。 指定服务器的格式为“username：password@host”。

基本EEM配置

您可以通过许多方式自定义EEM脚本以满足您的确切需求，但本示例是运行安全邮件功能的基本EEM脚本：

```
(config)# event manager environment _email_from <username@gmail.com>
(config)# event manager environment _email_to <EMAIL@domain.com>
(config)# event manager environment _email_server <username>:<password>@smtp.gmail.com

(config)# event manager applet SendSecureEmailEEM
(config-applet)#  event none
(config-applet)#    action 0010 mail server "$_email_server" to "$_email_to" from "$_email_from" cc "$_
```

配置首先创建三个EEM环境变量：_email_from、_email_to和_email_server。 每个变量都定义在一个变量中，以便更轻松地更改配置。 然后创建SendSecureEmailEEM脚本。 此处触发事件为“none”，因此您可以使用“# event manager run SendSecureEmailEEM”（而不是等待特定事件触发）手动运行EEM脚本。 接下来，您只需执行一个“邮件服务器”操作即可处理电子邮件生成。 “安全tls”和“端口587”选项告知设备在端口587上协商TLS，Gmail服务器将侦听该端口。

您还需要确保“from”字段有效。 如果您正在验证身份“Alice”，但正在尝试从“Bob”发送电子邮件，则会因Alice欺骗他人的电子邮件地址而出错。 “发件人”字段需要与服务器上用于发送电子邮件的帐户保持一致。

仅安装了默认证书时出现问题

EEM使用openssl与SMTP服务器建立连接。 为了安全通信，服务器将证书发送回在Cisco IOSd中运行的openssl。 然后，IOSd将查找与该证书关联的信任点。

在Cisco IOS XE设备上，默认情况下不安装Gmail SMTP服务器的证书。 必须手动导入它们才能建立信任。 如果未安装证书，则TLS握手将由于“证书错误”而失败。

以下调试对于调试任何证书问题非常有用：

```
debug event manager action mail
debug crypto pki API
debug crypto pki callbacks
debug crypto pki messages
debug crypto pki scep
debug crypto pki server
debug crypto pki transactions
debug crypto pki validation
debug ssl openssl errors
debug ssl openssl ext
debug ssl openssl msg
debug ssl openssl states
```

在触发EEM时，您可以在路由器上启动嵌入式数据包捕获(EPC)，以捕获传入或传出电子邮件服务器的任何流量：

```
! Trigger the EEM:
# event manager run SendSecureEmailEEM

<SNIP>

*Mar 15 21:51:32.798: CRYPTO_PKI: (A0693) Check for identical certs
*Mar 15 21:51:32.798: CRYPTO_PKI(Cert Lookup) issuer="cn=GlobalSign Root CA,ou=Root CA,o=GlobalSign nv-

*Mar 15 21:51:32.798: CRYPTO_PKI: looking for cert in handle=7F41EE523CE0, digest=
94 40 D1 90 A0 A3 5D 47 E5 B5 31 F6 63 AD 1B 0A

*Mar 15 21:51:32.799: CRYPTO_PKI: Cert record not found for issuer serial.
*Mar 15 21:51:32.799: CRYPTO_PKI : (A0693) Validating non-trusted cert
*Mar 15 21:51:32.799: CRYPTO_PKI: (A0693) Create a list of suitable trustpoints
*Mar 15 21:51:32.799: CRYPTO_PKI: crypto_pki_get_cert_record_by_issuer()
*Mar 15 21:51:32.799: CRYPTO_PKI: Unable to locate cert record by issuername
*Mar 15 21:51:32.799: CRYPTO_PKI: No trust point for cert issuer, looking up cert chain

*Mar 15 21:51:32.799: CRYPTO_PKI: crypto_pki_get_cert_record_by_subject()
*Mar 15 21:51:32.799: CRYPTO_PKI: (A0693) No suitable trustpoints found
*Mar 15 21:51:32.799: CRYPTO_PKI: (A0693) Removing verify context

*Mar 15 21:51:32.799: CRYPTO_PKI: destroying ca_req_context type PKI_VERIFY_CHAIN_CONTEXT,ident 32, ref
*Mar 15 21:51:32.799: CRYPTO_PKI: ca_req_context released
*Mar 15 21:51:32.799: CRYPTO_OPSSL: Certificate verification has failed
*Mar 15 21:51:32.799: CRYPTO_PKI: Rcvd request to end PKI session A0693.
*Mar 15 21:51:32.799: CRYPTO_PKI: PKI session A0693 has ended. Freeing all resources.
*Mar 15 21:51:32.800: >>> ??? [length 0005]
*Mar 15 21:51:32.800: 15 03 03 00 02
*Mar 15 21:51:32.800:
*Mar 15 21:51:32.800: >>> TLS 1.2 Alert [length 0002], fatal bad_certificate
*Mar 15 21:51:32.800: 02 2A
*Mar 15 21:51:32.800:
*Mar 15 21:51:32.800: SSL3 alert write:fatal:bad certificate
*Mar 15 21:51:32.801: P11:C_OpenSession slot 1 flags 6
*Mar 15 21:51:32.801: SSL_connect:error in error
*Mar 15 21:51:32.801: 0:error:1416F086:SSL routines:tls_process_server_certificate:certificate verify f
```

最后，openssl无法与SMTP服务器建立安全TLS会话，因此会抛出“不良证书”错误，这会导致EEM停止运行：

```
*Mar 15 21:51:32.801: %HA_EM-3-FMPD_SMTP: Error occurred when sending mail to SMTP server: username:pas
*Mar 15 21:51:32.802: %HA_EM-3-FMPD_ERROR: Error executing applet SendSecureEmailEEM statement 0010
```

从此交换记录的数据包捕获附加为“NoCertificateInstalled.pcap”。 从路由器(10.122.x.x)到Gmail SMTP服务器(142.251.163.xx)的最终TLS数据包显示，由于在调试中看到的相同“Bad Certificate”消息，TLS协商已终止。

```
Frame 33: 61 bytes on wire (488 bits), 61 bytes captured (488 bits)
Ethernet II, Src: Cisco_a3:c5:f0 (74:86:0b:a3:c5:f0), Dst: Cisco_f0:44:45 (00:08:30:f0:44:45)
Internet Protocol Version 4, Src: 10.122.xx.xx, Dst: 142.251.163.xx
Transmission Control Protocol, Src Port: 13306, Dst Port: 587, Seq: 189, Ack: 4516, Len: 7
Transport Layer Security
TLSv1.2 Record Layer: Alert (Level: Fatal, Description: Bad Certificate)
Content Type: Alert (21)
Version: TLS 1.2 (0x0303)
Length: 2
Alert Message
Level: Fatal (2)
Description: Bad Certificate (42)
```

用于保护SMTP的证书

由于缺少允许Cisco IOS XE设备信任Gmail服务器的证书，因此解决方法是将这些证书中的一个/所有安装到设备上的信任点中。

例如，上一个测试的完整调试显示发生了以下证书查找：

```
CRYPTO_PKI(Cert Lookup) issuer="cn=GTS CA 1C3,o=Google Trust Services LLC,c=US" serial number= 52 87 E0
CRYPTO_PKI(Cert Lookup) issuer="cn=GTS Root R1,o=Google Trust Services LLC,c=US" serial number= 02 03 B
CRYPTO_PKI(Cert Lookup) issuer="cn=GlobalSign Root CA,ou=Root CA,o=GlobalSign nv-sa,c=BE" serial number
```

需要在信任点下为每台此类颁发机构安装证书，以便设备能够与Gmail SMTP服务器建立安全会话。 可以使用以下配置为每个颁发者创建信任点：

```
crypto pki trustpoint CA-GTS-1C3
  enrollment terminal
  revocation-check none
  chain-validation stop

crypto pki trustpoint CA-GTS-Root-R1
  enrollment terminal
  revocation-check none
  chain-validation stop

crypto pki trustpoint CA-GlobalSign-Root
  enrollment terminal
  revocation-check none
  chain-validation stop

crypto pki trustpoint CA-gmail-SMTP
  enrollment terminal
  revocation-check none
  chain-validation stop
```

您现在已为每个颁发者设置了信任点；但是，目前还没有实际证书与其关联。 它们基本上是空白信任点：

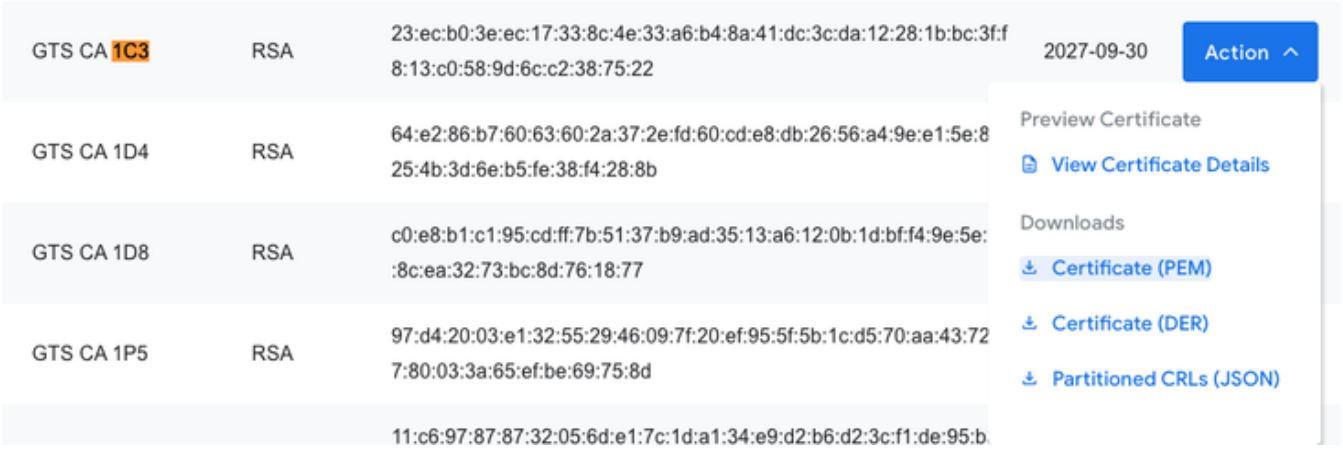
```
# show run | sec crypto pki certificate chain CA-
crypto pki certificate chain CA-GTS-1C3
crypto pki certificate chain CA-GTS-Root-R1
crypto pki certificate chain CA-GlobalSign-Root
crypto pki certificate chain CA-gmail-SMTP
```

您必须找到这些证书的位置，然后在设备上安装它们。

在在线搜索“Google Trust Services 1C3”时，我们很快发现证书的Google Trust Services存储库：

https://pki.goog/repository/

展开该页面上的所有证书后，您可以搜索以查找“1C3”，点击“操作”下拉菜单，然后下载PEM证书：



使用文本编辑器打开已下载的PEM文件，表明这只是可以在您之前创建的信任点下导入到Cisco IOS XE设备上的证书：

```
-----BEGIN CERTIFICATE-----
MIIFljCCA36gAwIBAgINAgO8U1lrNMcY9QFQZjANBgkqhkiG9w0BAQsFADBHMQsw
CQYDVQQGEwJVUzEiMCAGA1UEChMZR29vZ2xlIFRydXN0IFNlcnZpY2VzIExMQzEU
<snip>
AJ2xDx8hcFH1mtOG/FX0Kw4zd8NLQsLxdxP8c4CU6x+7Nz/OAipmsHMdMqUybDKw
juDEI/9bfU1lcKwrmz3O2+BtjjKAvpafkmO8l7tdufThcV4q5O8DIrGKZTqPwJNl
1IXNDw9bg1kWRxYtnCQ6yICmJhSFm/Y3m6xv+cXDBlHz4n/FsRC6UfTd
-----END CERTIFICATE-----
```

您可以使用配置命令在“CA-GTS-1C3”信任点下导入它：

```
(config)# crypto pki authenticate CA-GTS-1C3
```

```
Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

MIIFljCCA36gAwIBAgINAgO8U1lrNMcY9QFQZjANBgkqhkiG9w0BAQsFADBHMQsw
CQYDVQQGEwJVUzEiMCAGA1UEChMZR29vZ2xlIFRydXN0IFNlcnZpY2VzIExMQzEU
<snip>
juDEI/9bfU1lcKwrmz3O2+BtjjKAvpafkmO8l7tdufThcV4q5O8DIrGKZTqPwJNl
1IXNDw9bg1kWRxYtnCQ6yICmJhSFm/Y3m6xv+cXDBlHz4n/FsRC6UfTd

Certificate has the following attributes:
Fingerprint MD5: 178EF183 43CCC9E0 ECB0E38D 9DEA03D8
Fingerprint SHA1: 1E7EF647 CBA15028 1C608972 57102878 C4BD8CDC
Certificate validated - Signed by existing trustpoint CA certificate.

Trustpoint CA certificate accepted.
% Certificate successfully imported

(config)#
```

然后您可以确认证书已安装：

```
# show run | sec crypto pki certificate chain CA-GTS-1C3
crypto pki certificate chain CA-GTS-1C3
 certificate ca 02D3BC53596B34C718F5015066
  30820596 3082037E A0030201 02020D02 03BC5359 6B34C718 F5015066 300D0609
  2A864886 F70D0101 0B050030 47310B30 09060355 04061302 55533122 30200603
  55040A13 19476F6F 676C6520 54727573 74205365 72766963 6573204C 4C433114
  <snip>
  E1715E2A E4EF0322 B18A653A 8FC09365 D485CD0F 0F5B8359 1647162D 9C243AC8
  80A62614 859BF637 9BAC6FF9 C5C30651 F3E27FC5 B110BA51 F4DD
    quit
```

```
#show crypto pki certificates verbose CA-GTS-1C3
CA Certificate
  Status: Available
  Version: 3
  Certificate Serial Number (hex): 02D3BC53596B34C718F5015066
  Certificate Usage: Signature
  Issuer:
    cn=GTS Root R1
    o=Google Trust Services LLC
    c=US
  Subject:
    cn=GTS CA 1C3
    o=Google Trust Services LLC
    c=US
  CRL Distribution Points:
    http://crl.pki.goog/gtsr1/gtsr1.crl
  Validity Date:
    start date: 00:00:42 UTC Aug 13 2020
    end date: 00:00:42 UTC Sep 30 2027
  Subject Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (2048 bit)
```

```
  Signature Algorithm: SHA256 with RSA Encryption
  Fingerprint MD5: 178EF183 43CCC9E0 ECB0E38D 9DEA03D8
  Fingerprint SHA1: 1E7EF647 CBA15028 1C608972 57102878 C4BD8CDC
  X509v3 extensions:
    X509v3 Key Usage: 86000000
      Digital Signature
      Key Cert Sign
      CRL Signature
    X509v3 Subject Key ID: 8A747FAF 85CDEE95 CD3D9CD0 E24614F3 71351D27
    X509v3 Basic Constraints:
        CA: TRUE
    X509v3 Authority Key ID: E4AF2B26 711A2B48 27852F52 662CEFF0 8913713E
    Authority Info Access:
        OCSP URL: http://ocsp.pki.goog/gtsr1
        CA ISSUERS: http://pki.goog/repo/certs/gtsr1.der
    X509v3 CertificatePolicies:
        Policy: 2.23.140.1.2.2
        Policy: 2.23.140.1.2.1
        Policy: 1.3.6.1.4.1.11129.2.5.3
            Qualifier ID: 1.3.6.1.5.5.7.2.1
            Qualifier Info: https://pki.goog/repository/
    Extended Key Usage:
        Client Auth
        Server Auth
  Cert install time: 02:31:20 UTC Mar 16 2023
Cert install time in nsec: 1678933880873946880
  Associated Trustpoints: CA-GTS-1C3
```

接下来，您可以为其他两个颁发者安装证书。

CA-GTS-Root-R1：

配置:

Spoiler（突出显示以便阅读）

```
(config)# crypto pki authenticate CA-GTS-Root-R1

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

MIIFVzCCAz+gAwIBAgINAgPlk28xsBNJiGuiFzANBgkqhkiG9w0BAQwFADBHMQsw
CQYDVQQGEwJVUzEiMCAGA1UEChMZR29vZ2xlIFRydXN0IFNlcnZpY2VzIExMQzEU
<snip>
2tIMPNuzjsmhDYAPexZ3FL//2wmUspO8IFgV6dtxQ/PeEMMA3KgqlbbC1j+Qa3bb
bP6MvPJwNQzcmRk13NfIRmPVNnGuV/u3gm3c

Certificate has the following attributes:
Fingerprint MD5: 05FED0BF 71A8A376 63DA01E0 D852DC40
Fingerprint SHA1: E58C1CC4 913B3863 4BE9106E E3AD8E6B 9DD9814A

% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported

(config)# end
```

(config)# crypto pki authenticate CA-GTS-Root-R1输入base 64编码的CA证书。以空白行或单词“quit”结尾
MIIFVzCCAz+gAwIBAgINAgPlk28xsBNJiGuiFzANBgkqhkiG9w0BAQwFADBHMQswCQYDVQQQQQQG
zEiMCAGA1UEChMZR29vZ2xlIFRydXN0IFNlcnZpY2VzIExMQzEU<snip>2tIMPNuzjsmhDYAPexZ3FL//2
bbbP6MvPJwNQzcmRk13NfIRmPVNnGuV/u3gm3c证书具有以下属性：指纹MD5： 05FED0BF 71A8A376 63DA01E0 D852DC40指纹SHA1： E58C1CC4 913B3863 4BE9106E E3AD8E6B 9DD9814A%您是否接受此证书？[yes/no]： yesTrustpoint CA证书已接受。%证书已成功导入
(config)#结束

运行配置验证：

Spoiler（突出显示以便阅读）

```
# show run | sec crypto pki certificate chain CA-GTS-Root-R1
crypto pki certificate chain CA-GTS-Root-R1
 certificate ca 0203E5936F31B01349886BA217
  30820557 3082033F A0030201 02020D02 03E5936F 31B01349 886BA217 300D0609
  2A864886 F70D0101 0C050030 47310B30 09060355 04061302 55533122 30200603
  <snip>
  6775C119 3A2B474E D3428EFD 31C81666 DAD20C3C DBB38EC9 A10D800F 7B167714
  BFFFDB09 94B293BC 205815E9 DB7143F3 DE10C300 DCA82A95 B6C2D63F 906B76DB
  6CFE8CBC F270350C DC991935 DCD7C846 63D53671 AE57FBB7 826DDC
    quit
```

show run | sec crypto pki certificate chain CA-GTS-Root-R1crypto pki certificate chain CA-GTS-Root-R1 certificate ca 0203E5936F31B01349886BA217 30820557 3082033F A0030201 02020D02 03E5936F 31B01349 886BA217 300D0609 2A864886 F7 d0101 0C050030 47310B30 09060355 04061302 55533122 <snip> 6775C119 3A2B474E D3428EFD 31C30200603 DAD20C3C DBB38EC9 A10D800F 7B81666 BFFFDB09 94B293BC 167714E9 DB7 143F3 DE10C300 DCA82A95 B6C2D63F 906B76DB 6CFE8CBC F205815C DC270350 DCD7C846 63D991935 53671 AE57FBB7 826DDC quit

显示加密验证：

Spoiler（突出显示以便阅读）

```
# show crypto pki certificates verbose CA-GTS-Root-R1
CA Certificate
  Status: Available
  Version: 3
  Certificate Serial Number (hex): 0203E5936F31B01349886BA217
  Certificate Usage: Signature
  Issuer:
    cn=GTS Root R1
    o=Google Trust Services LLC
    c=US
  Subject:
    cn=GTS Root R1
    o=Google Trust Services LLC
    c=US
  Validity Date:
    start date: 00:00:00 UTC Jun 22 2016
    end date: 00:00:00 UTC Jun 22 2036
```

```
  Subject Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (4096 bit)
  Signature Algorithm: SHA384 with RSA Encryption
  Fingerprint MD5: 05FED0BF 71A8A376 63DA01E0 D852DC40
  Fingerprint SHA1: E58C1CC4 913B3863 4BE9106E E3AD8E6B 9DD9814A
  X509v3 extensions:
    X509v3 Key Usage: 86000000
      Digital Signature
      Key Cert Sign
      CRL Signature
    X509v3 Subject Key ID: E4AF2B26 711A2B48 27852F52 662CEFF0 8913713E
    X509v3 Basic Constraints:
        CA: TRUE
    Authority Info Access:
  Cert install time: 14:39:38 UTC Mar 13 2023
Cert install time in nsec: 1678718378546968064
  Associated Trustpoints: CA-GTS-Root-R1 Trustpool
```

show crypto pki certificates verbose CA-GTS-Root-R1CA证书状态：可用版本： 3证书序列号（十六进制）： 0203E5936F31B01349886BA217证书用法：签名颁发者： cn=GTS根R1 o=Google Trust Services LLC c=US主题： cn=GTS根R1 o=Google Trust Services LLC c=US有效日期：日期：00:00:00 UTC 6月22日2016结束日期：00:00:00 UTC 6月22日2036主题密钥信息：公钥算法：rsa加密RSA公钥：（4096位）签名算法：带RSA加密指纹的SHA384 MD5：05FED0BF 71A8A376 63DA01E0 852DC40指纹SHA1：E58C1CC4 913B3863 4BE9106E E3AD8E6B 9DD9814A X509v3扩展：X509v3密钥用法：86000000数字签名密钥证书签名CRL签名X509v3主题密钥ID：E4AF2B 26 711A2B48 27852F52 662CEFF0 8913713E X509v3基本限制： CA：真实授权信息访问：证书安装时间： 14:39:38 UTC 3月13日2023年nsec格式的证书安装时间： 1678718378546968064关联信任点： CA-GTS-Root-R1 Trustpool

CA-GlobalSign-Root：

在此位置找到此证书：

https://support.globalsign.com/ca-certificates/root-certificates/globalsign-root-certificates

配置:

Spoiler（突出显示以便阅读）

```
(config)# crypto pki authenticate CA-GlobalSign-Root

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

MIIDdTCCAl2gAwIBAgILBAAAAAABFUtaw5QwDQYJKoZIhvcNAQEFBQAwVzELMAkG
A1UEBhMCQkUxGTAXBgNVBAoTEEdsb2JhbFNpZ24gbnYtc2ExEDAOBgNVBAsTB1Jv
<snip>
DKqC5JlR3XC321Y9YeRq4VzW9v493kHMB65jUr9TU/Qr6cf9tveCX4XSQRjbgbME
HMUfpIBvFSDJ3gyICh3WZlXi/EjJKSZp4A==

Certificate has the following attributes:
Fingerprint MD5: 3E455215 095192E1 B75D379F B187298A
Fingerprint SHA1: B1BC968B D4F49D62 2AA89A81 F2150152 A41D829C
```

```
% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported

(config)# end
```

(config)# crypto pki authenticate CA-GlobalSign-Root输入base 64编码的CA证书。以空白行或单词“quit”结尾
MIIDdTCCAl2gAwIBAgILBAAAAAAABFUtaw5QwDQYJKoZIhvcNAQEFBQAwVzELMAkGA1UEBhMCQ
MEHMUfpIBvFSDJ3gyICh3WZlXi/EjJKSZp4A==证书具有以下属性：指纹MD5： 3E455215
095192E1 B75D33 79F B187298A指纹SHA1：B1BC968B D4F49D62 2AA89A81 F2150152
A41D829C%是否接受此证书？[yes/no]： yesTrustpoint CA证书已接受。%证书已成功导入
(config)#结束

运行配置验证：

Spoiler （突出显示以便阅读）

```
# show run | sec crypto pki certificate chain CA-GlobalSign-Root
crypto pki certificate chain CA-GlobalSign-Root
 certificate ca 040000000001154B5AC394
  30820375 3082025D A0030201 02020B04 00000000 01154B5A C394300D 06092A86
  <snip>
  2AC45631 95D06789 852BF96C A65D469D 0CAA82E4 9951DD70 B7DB563D 61E46AE1
  5CD6F6FE 3DDE41CC 07AE6352 BF5353F4 2BE9C7FD B6F7825F 85D24118 DB81B304
  1CC51FA4 806F1520 C9DE0C88 0A1DD666 55E2FC48 C9292669 E0
    quit
```

show run | sec crypto pki certificate chain CA-GlobalSign-Rootcrypto pki certificate chain CA-GlobalSign-Root certificate ca 040000000001154B5AC3082025D A0030201 02020B04 00000000 01154B5A C394300D 06092A86 <snip> 2AC45631 95D30820375 852BF96C A65D469D 0CAA82E4 951DD 70 B7DB563D 61E46AE1 5CD6F6FE 3DDE41CC 07AE6352 BF5353F4 2BE9C7FD B6F7825F 85D06789 DB81B304 1CC51FA4 806F1520 9DE0C88 0A1DD666 55E2FC48 C24118 9292669 E0 quit

显示加密验证：

Spoiler （突出显示以便阅读）

```
#show crypto pki certificates verbose CA-GlobalSign-Root
CA Certificate
Status: Available
Version: 3
Certificate Serial Number (hex): 040000000001154B5AC394
Certificate Usage: Signature
Issuer:
cn=GlobalSign Root CA
ou=Root CA
o=GlobalSign nv-sa
c=BE
Subject:
```

```
cn=GlobalSign Root CA
ou=Root CA
o=GlobalSign nv-sa
c=BE
Validity Date:
start date: 12:00:00 UTC Sep 1 1998
end date: 12:00:00 UTC Jan 28 2028
Subject Key Info:
Public Key Algorithm: rsaEncryption
RSA Public Key: (2048 bit)
Signature Algorithm: SHA1 with RSA Encryption
Fingerprint MD5: 3E455215 095192E1 B75D379F B187298A
Fingerprint SHA1: B1BC968B D4F49D62 2AA89A81 F2150152 A41D829C
X509v3 extensions:
X509v3 Key Usage: 6000000
Key Cert Sign
CRL Signature
X509v3 Subject Key ID: 607B661A 450D97CA 89502F7D 04CD34A8 FFFCFD4B
X509v3 Basic Constraints:
CA: TRUE
Authority Info Access:
Cert install time: 03:03:01 UTC Mar 16 2023
Cert install time in nsec: 1678935781942944000
Associated Trustpoints: CA-GlobalSign-Root
```

#show crypto pki certificates verbose CA-GlobalSign-RootCA CertificateStatus：AvailableVersion： 3证书序列号（十六进制）： 040000000001154B5AC394证书用法：SignatureIssuer： cn=GlobalSign Root CAou=Root CAo=GlobalSign nv-sac=BESsubject：cn=GlobalSign Root CAou=Root CAo=GlobalSign-sac=BEValidity日期：开始日期： 12 00:00 UTC 9月1日1998结束日期： 12:00:00 UTC 1月28日2028主题密钥信息：公钥算法：rsaEncryptionRSA公钥：（2048位）签名算法：SHA1带RSA加密指纹MD5： 3E455215095192E1 B75D379F B187298A指纹SHA1： B1BC96 8B D4F49D62 2AA89A81 F2150152 A41D829C X509v3扩展：X509v3密钥用法： 6000000密钥证书签名CRL签名X509v3主题密钥ID：607B661A 450D97CA 89502F7D 04CD3 A8 FFFCFD4B X509v3基本限制：CA： TRUEAauthority Info Access：证书安装时间： 03:03:01 UTC 3月16日2023证书安装时间（以nsec为单位）：1678935781942944000关联信任点： CA-GlobalSign-Root

CA-gmail-SMTP：

通过此处记录的步骤：使用TLS证书进行安全传输找到Gmail服务器(CA-gmail-SMTP)的TLS证书

配置:

Spoiler（突出显示以便阅读）

```
(ca-trustpoint)# crypto pki authenticate CA-gmail-SMTP

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself

MIIEhjCCA26gAwIBAgIQUofgQKT+9wcSaLBP3d3w9DANBgkqhkiG9w0BAQsFADBG
MQswCQYDVQQGEwJVUzEiMCAGA1UEChMZR29vZ2xlIFRydXN0IFNlcnZpY2VzIExM
<snip>
b1J2gZAyjyd4nfFRG1jeL5KrsfUR9hIXufqySvlPUoPuKSi3fvsIS21BYEXEe8uZ
gBxJaeTUjncvow==
```

```
Trustpoint 'CA-gmail-SMTP' is a subordinate CA.
but certificate is not a CA certificate.
Manual verification required
Certificate has the following attributes:
Fingerprint MD5: 19651FBE 906A414D 6D57B783 946F30A2
Fingerprint SHA1: 4EF392CB EEB46D5E 47433953 AAEF313F 4C6D2825

% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported

(config)#
```

(ca-trustpoint)# crypto pki authenticate CA-gmail-SMTPE输入base 64编码的CA证书。以空白行或单词“quit”结束 MIIEhjCCA26gAwIBAgIQUofgQKT+9wcSaLBP3d3w9DANBgkqhkiG9w0BAQsFADBGMQswCQYDVQQG zgBxJaeTUjncvow==信任点“CA-gmail-SMTP”是下级CA。但证书不是CA证书。需要手动验证证书具有以下属性：指纹MD5： 19651FBE 906A414D 6D 57B783 946F30A2指纹SHA1:4EF392CB EEB46D5E 47433953 AAEF313F 4C6D2825%您是否接受此证书？[yes/no]： yesTrustpoint CA证书已接受。%证书已成功导入(config)#

运行配置验证：

Spoiler （突出显示以便阅读）

```
# show run | sec crypto pki certificate chain CA-gmail-SMTP
 crypto pki certificate chain CA-gmail-SMTP
  certificate ca 5287E040A4FEF7071268B04FDDDDF0F4
  30820486 3082036E A0030201 02021052 87E040A4 FEF70712 68B04FDD DDF0F430
  0D06092A 864886F7 0D01010B 05003046 310B3009 06035504 06130255 53312230
  <snip>
  92ABB1F5 11F61217 B9FAB24A F94F5283 EE2928B7 7EFB084B 6D416045 C47BCB99
  801C4969 E4D48E77 2FA3
    quit
```

show run | sec crypto pki certificate chain CA-gmail-SMTP crypto pki certificate chain CA-gmail-SMTP certificate ca 5287E040A4FEF7071268B04FDDDF0F4 30820486 3082036E A0030201 02021052 87E040A4 FEF70712 68B04FDD DDF0F430 0D06092A 864886F7 0D01010B 05003046 310B 3009 06035504 06130255 <snip> 92ABB1F5 11F53312230 B9FAB24A F94F5283 EE2928B7 7EFB084B 6D61217 416045 C47BCB99 801C4969 E4D48E77 2FA3 quit

显示加密验证：

Spoiler （突出显示以便阅读）

```
# show crypto pki certificates verbose CA-gmail-SMTP
CA Certificate
Status: Available
Version: 3
Certificate Serial Number (hex): 5287E040A4FEF7071268B04FDDDDF0F4
Certificate Usage: Signature
Issuer:
```

```
cn=GTS CA 1C3
o=Google Trust Services LLC
c=US
Subject:
cn=smtp.gmail.com
CRL Distribution Points:
http://crls.pki.goog/gts1c3/moVDfISia2k.crl
Validity Date:
start date: 09:15:03 UTC Feb 20 2023
end date: 09:15:02 UTC May 15 2023
Subject Key Info:
Public Key Algorithm: ecEncryption
EC Public Key: (256 bit)
Signature Algorithm: SHA256 with RSA Encryption
Fingerprint MD5: 19651FBE 906A414D 6D57B783 946F30A2
Fingerprint SHA1: 4EF392CB EEB46D5E 47433953 AAEF313F 4C6D2825
X509v3 extensions:
X509v3 Key Usage: 80000000
Digital Signature
X509v3 Subject Key ID: 5CC36972 D07FE997 510E1A67 8A8ECC23 E40CFB68
X509v3 Basic Constraints:
CA: FALSE
X509v3 Subject Alternative Name:
smtp.gmail.com
IP Address :
OtherNames :
X509v3 Authority Key ID: 8A747FAF 85CDEE95 CD3D9CD0 E24614F3 71351D27
Authority Info Access:
OCSP URL: http://ocsp.pki.goog/gts1c3
CA ISSUERS: http://pki.goog/repo/certs/gts1c3.der
X509v3 CertificatePolicies:
Policy: 2.23.140.1.2.1
Extended Key Usage:
Server Auth
Cert install time: 03:10:41 UTC Mar 16 2023
Cert install time in nsec: 1678936241822955008
Associated Trustpoints: CA-gmail-SMTP
```

show crypto pki certificates verbose CA-gmail-SMTPCA CertificateStatus： AvailableVersion：3证书序列号（十六进制）： 5287E040A4FEF7071268B04FDDDF0F4证书用法：SignatureIssuer： cn=GTS CA 1C3o=Google Trust Services LLCc=USSsubject：cn=smtp.gmail.comCRLDistribution Poination：http://crls.pki.goog/gts1c3/moVDfISia2k.crlValidity分发点：日期： 15:03 UTC 2月20日2023年结束日期： 09:15:02 UTC 5月15日2023主题密钥信息：公钥算法： ec加密EC公钥：（256位）签名算法：SHA256 with RSA加密指纹MD5： 19651FBE 906A414D 6D57B783 946F3 0A2指纹SHA1:4EF392CB EEB46D5E 47433953 AAEF313F 4C6D2825 X509v3扩展：X509v3密钥用法：80000000数字签名X509v3主题密钥ID：5CC36972 D07FE997 510E1A67 8A ECC 23 E40CFB68 X509v3基本限制：CA： FALSEX509v3主题备用名称：smtp.gmail.com IP地址：其他名称：X509v3授权密钥ID：8A747FAF 85CDEE95 CD3D9CD0 E24614F3 71351D27授权信息访问：OCSP URL： http://ocsp.pki.goog/gts1c3CA证书颁发者：http://pki.goog/repo/certs/gts1c3.derX509v3策略：策略：2.23.140.1.2.1扩展密钥用法：服务器AuthCert安装时间：03:10:41 UTC 3月16日2023年3月16日以nsec表示的证书安装时间：1678936241822955008关联信任点： CA-gmail-SMTP

更简单的证书查找方法

或者，您可以尝试使用来自服务器/笔记本电脑的openssl呼叫作为从SMTP服务器获取证书的简单方法，而无需使用调试和搜索Google来跟踪这些证书：

```
openssl s_client -showcerts -verify 5 -connect gmail-smtp-in.l.google.com:25 -starttls smtp
```

您还可以use smtp.gmail.com：

```
openssl s_client -showcerts -verify 5 -connect smtp.gmail.com:25 -starttls smtp
```

该呼叫的输出将包括可用于“crypto pki authenticate <trustpoint>”配置的实际证书本身。

再次使用安全SMTP测试EEM

现在，证书已应用到Cisco IOS XE设备，EEM脚本将按预期发送安全SMTP消息。

```
# event manager run SendSecureEmailEEM
```

检查剧透器中的完全加密和ssl调试输出：

Spoiler（突出显示以便阅读）

```
# event manager run SendSecureEmailEEM

*Mar 16 03:28:50.673: CRYPTO_OPSSL: Allocated the memory for OPSSLContext
*Mar 16 03:28:50.673: CRYPTO_OPSSL: Set cipher specs to mask 0x02FC0000 for version 128
*Mar 16 03:28:50.674: Set the Default EC Curve list: 0x70Set the EC curve list: secp521r1:secp384r1:prin
*Mar 16 03:28:50.674: opssl_SetPKIInfo entry
*Mar 16 03:28:50.674: CRYPTO_PKI: (A069B) Session started - identity selected (TP-self-signed-486541296)
*Mar 16 03:28:50.674: CRYPTO_PKI: Begin local cert chain retrieval.
*Mar 16 03:28:50.674: CRYPTO_PKI(Cert Lookup) issuer="cn=IOS-Self-Signed-Certificate-486541296" serial n

*Mar 16 03:28:50.674: CRYPTO_PKI: looking for cert in handle=7F41EE523CE0, digest=
1C 7F 3D 52 67 66 D5 59 E2 66 58 E7 8B E7 9B 8E

*Mar 16 03:28:50.675: CRYPTO_PKI: Done with local cert chain fetch 0.
*Mar 16 03:28:50.675: CRYPTO_PKI: Rcvd request to end PKI session A069B.
*Mar 16 03:28:50.675: CRYPTO_PKI: PKI session A069B has ended. Freeing all resources.TP-self-signed-486
*Mar 16 03:28:50.675: opssl_SetPKIInfo done.
*Mar 16 03:28:50.675: CRYPTO_OPSSL: Common Criteria is disabled on this session.Disabling Common Criter

*Mar 16 03:28:50.675: CRYPTO_OPSSL: ciphersuites ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-SHA
```

```
*Mar 16 03:28:50.676: Handshake start: before SSL initialization
*Mar 16 03:28:50.676: SSL_connect:before SSL initialization
*Mar 16 03:28:50.676: >>> ??? [length 0005]
*Mar 16 03:28:50.676: 16 03 01 00 95
*Mar 16 03:28:50.676:
*Mar 16 03:28:50.676: >>> TLS 1.2 Handshake [length 0095], ClientHello
*Mar 16 03:28:50.676: 01 00 00 91 03 03 26 4B 9F B3 44 94 FD 5F FD A1
<snip>
*Mar 16 03:28:50.679: 03 03 01 02 01
*Mar 16 03:28:50.679:
*Mar 16 03:28:50.679: SSL_connect:SSLv3/TLS write client hello
*Mar 16 03:28:50.692: <<< ??? [length 0005]
*Mar 16 03:28:50.692: 16 03 03 00 3F
*Mar 16 03:28:50.692:
*Mar 16 03:28:50.692: SSL_connect:SSLv3/TLS write client hello
*Mar 16 03:28:50.692: <<< TLS 1.2 Handshake [length 003F], ServerHello
*Mar 16 03:28:50.692: 02 00 00 3B 03 03 64 12 7E 05 25 F6 7A BD A0 2E
*Mar 16 03:28:50.692: 58 83 12 7F 90 CD F4 AB E2 69 53 A8 C7 FC 44 4F
*Mar 16 03:28:50.692: 57 4E 47 52 44 01 00 C0 2B 00 00 13 00 17 00 00
*Mar 16 03:28:50.693: FF 01 00 01 00 00 0B 00 02 01 00 00 23 00 00
*Mar 16 03:28:50.693: TLS server extension "unknown" (id=23), len=0
TLS server extension "renegotiate" (id=65281), len=1

*Mar 16 03:28:50.693: 00
*Mar 16 03:28:50.693: TLS server extension "EC point formats" (id=11), len=2

*Mar 16 03:28:50.693: 01 00
*Mar 16 03:28:50.693: TLS server extension "session ticket" (id=35), len=0

*Mar 16 03:28:50.693: <<< ??? [length 0005]
*Mar 16 03:28:50.693: 16 03 03 0F 9A
*Mar 16 03:28:50.694:
*Mar 16 03:28:50.702: SSL_connect:SSLv3/TLS read server hello
*Mar 16 03:28:50.702: <<< TLS 1.2 Handshake [length 0F9A], Certificate
*Mar 16 03:28:50.702: 0B 00 0F 96 00 0F 93 00 04 8A 30 82 04 86 30 82
*Mar 16 03:28:50.702: 03 6E A0 03 02 01 02 02 10 52 87 E0 40 A4 FE F7
<snip>
*Mar 16 03:28:50.763: 82 35 CF 62 8B C9 24 8B A5 B7 39 0C BB 7E 2A 41
*Mar 16 03:28:50.763: BF 52 CF FC A2 96 B6 C2 82 3F
*Mar 16 03:28:50.763:
*Mar 16 03:28:50.765: CC_DEBUG: Entering shim layer app callback function
*Mar 16 03:28:50.765: CRYPTO_PKI: (A069C) Session started - identity not specified
*Mar 16 03:28:50.765: CRYPTO_PKI: (A069C) Adding peer certificate
*Mar 16 03:28:50.767: CRYPTO_PKI: Added x509 peer certificate - (1162) bytes
*Mar 16 03:28:50.767: CRYPTO_PKI: (A069C) Adding peer certificate
*Mar 16 03:28:50.768: CRYPTO_PKI: Added x509 peer certificate - (1434) bytes
*Mar 16 03:28:50.768: CRYPTO_PKI: (A069C) Adding peer certificate
*Mar 16 03:28:50.770: CRYPTO_PKI: Added x509 peer certificate - (1382) bytes
*Mar 16 03:28:50.770: CRYPTO_OPSSL: Validate Certificate Chain Callback
*Mar 16 03:28:50.770: CRYPTO_PKI(Cert Lookup) issuer="cn=GTS CA 1C3,o=Google Trust Services LLC,c=US" s

*Mar 16 03:28:50.770: CRYPTO_PKI: looking for cert in handle=7F41EE523CE0, digest=
A7 CC 4B 0F 36 C3 AC D1 2F 77 DD 1D 9A 37 DC FC

*Mar 16 03:28:50.770: CRYPTO_PKI(Cert Lookup) issuer="cn=GTS Root R1,o=Google Trust Services LLC,c=US"

*Mar 16 03:28:50.771: CRYPTO_PKI: looking for cert in handle=7F41EE523CE0, digest=
03 9F CF 59 82 EE 09 CC 4F 53 AE D8 02 7E 4B AF

*Mar 16 03:28:50.771: CRYPTO_PKI(Cert Lookup) issuer="cn=GlobalSign Root CA,ou=Root CA,o=GlobalSign nv-

*Mar 16 03:28:50.771: CRYPTO_PKI: looking for cert in handle=7F41EE523CE0, digest=
```

```
94 40 D1 90 A0 A3 5D 47 E5 B5 31 F6 63 AD 1B 0A

*Mar 16 03:28:50.771: CRYPTO_PKI: Cert record not found for issuer serial.
*Mar 16 03:28:50.772: CRYPTO_PKI: crypto_pki_get_cert_record_by_subject()
*Mar 16 03:28:50.772: CRYPTO_PKI: Found a subject match
*Mar 16 03:
#28:50.772: CRYPTO_PKI: ip-ext-val: IP extension validation not required:Incrementing refcount for cont
*Mar 16 03:28:50.773: CRYPTO_PKI: create new ca_req_context type PKI_VERIFY_CHAIN_CONTEXT,ident 35
*Mar 16 03:28:50.773: CRYPTO_PKI: (A069C)validation path has 1 certs

*Mar 16 03:28:50.773: CRYPTO_PKI: (A069C) Check for identical certs
*Mar 16 03:28:50.773: CRYPTO_PKI(Cert Lookup) issuer="cn=GlobalSign Root CA,ou=Root CA,o=GlobalSign nv-

*Mar 16 03:28:50.774: CRYPTO_PKI: looking for cert in handle=7F41EE523CE0, digest=
94 40 D1 90 A0 A3 5D 47 E5 B5 31 F6 63 AD 1B 0A

*Mar 16 03:28:50.774: CRYPTO_PKI: Cert record not found for issuer serial.
*Mar 16 03:28:50.774: CRYPTO_PKI : (A069C) Validating non-trusted cert
*Mar 16 03:28:50.774: CRYPTO_PKI: (A069C) Create a list of suitable trustpoints
*Mar 16 03:28:50.774: CRYPTO_PKI: crypto_pki_get_cert_record_by_issuer()
*Mar 16 03:28:50.774: CRYPTO_PKI: Found a issuer match
*Mar 16 03:28:50.774: CRYPTO_PKI: (A069C) Suitable trustpoints are: CA-GlobalSign-Root,
*Mar 16 03:28:50.775: CRYPTO_PKI: (A069C) Attempting to validate certificate using CA-GlobalSign-Root p
*Mar 16 03:28:50.775: CRYPTO_PKI: (A069C) Using CA-GlobalSign-Root to validate certificate
*Mar 16 03:28:50.775: CRYPTO_PKI(make trusted certs chain)
*Mar 16 03:28:50.775: CRYPTO_PKI: Added 1 certs to trusted chain.
*Mar 16 03:28:50.775: CRYPTO_PKI: Prepare session revocation service providers
*Mar 16 03:28:50.776: P11:C_CreateObject:
*Mar 16 03:28:50.776: CKA_CLASS: PUBLIC KEY
*Mar 16 03:28:50.776: CKA_KEY_TYPE: RSA
*Mar 16 03:28:50.776: CKA_MODULUS:
DA 0E E6 99 8D CE A3 E3 4F 8A 7E FB F1 8B 83 25
6B EA 48 1F F1 2A B0 B9 95 11 04 BD F0 63 D1 E2
<snip>

*Mar 16 03:28:50.780: CKA_PUBLIC_EXPONENT: 01 00 01
*Mar 16 03:28:50.780: CKA_VERIFY_RECOVER: 01
*Mar 16 03:28:50.780: CRYPTO_PKI: Deleting cached key having key id 45
*Mar 16 03:28:50.781: CRYPTO_PKI: Attempting to insert the peer's public key into cache
*Mar 16 03:28:50.781: CRYPTO_PKI:Peer's public inserted successfully with key id 46
*Mar 16 03:28:50.781: P11:C_CreateObject: 131118
*Mar 16 03:28:50.781: P11:C_GetMechanismInfo slot 1 type 3 (invalid mechanism)
*Mar 16 03:28:50.781: P11:C_GetMechanismInfo slot 1 type 1
*Mar 16 03:28:50.781: P11:C_VerifyRecoverInit - 131118
*Mar 16 03:28:50.781: P11:C_VerifyRecover - 131118
*Mar 16 03:28:50.781: P11:found pubkey in cache using index = 46
*Mar 16 03:28:50.781: P11:public key found is :
30 82 01 22 30 0D 06 09 2A 86 48 86 F7 0D 01 01
01 05 00 03 82 01 0F 00 30 82 01 0A 02 82 01 01
<snip>
CF 02 03 01 00 01

*Mar 16 03:28:50.788: P11:CEAL:CRYPTO_NO_ERR
*Mar 16 03:28:50.788: P11:C_DestroyObject 2:2002E
*Mar 16 03:28:50.788: CRYPTO_PKI: Expiring peer's cached key with key id 46
*Mar 16 03:28:50.788: CRYPTO_PKI: (A069C) Certificate is verified
*Mar 16 03:28:50.788: CRYPTO_PKI: Remove session revocation service providers
*Mar 16 03:28:50.788: CRYPTO_PKI: Remove session revocation service providersCA-GlobalSign-Root:validat
*Mar 16 03:28:50.788: CRYPTO_PKI: (A069C) Certificate validated without revocation check:cert refcount
*Mar 16 03:28:50.790: CRYPTO_PKI: Populate AAA auth data
*Mar 16 03:28:50.790: CRYPTO_PKI: Unable to get configured attribute for primary AAA list authorization
*Mar 16 03:28:50.790: PKI: Cert key-usage: Digital-Signature , Certificate-Signing , CRL-Signing
```

```
*Mar 16 03:28:50.790: CRYPTO_PKI: (A069C)chain cert was anchored to trustpoint CA-GlobalSign-Root, and
*Mar 16 03:28:50.790: CRYPTO_PKI: (A069C) Removing verify context

*Mar 16 03:28:50.790: CRYPTO_PKI: destroying ca_req_context type PKI_VERIFY_CHAIN_CONTEXT,ident 35, ref
*Mar 16 03:28:50.790: CRYPTO_PKI: ca_req_context released
*Mar 16 03:28:50.790: CRYPTO_PKI: (A069C) Validation TP is CA-GlobalSign-Root
*Mar 16 03:28:50.790: CRYPTO_PKI: (A069C) Certificate validation succeeded
*Mar 16 03:28:50.790: CRYPTO_OPSSL: Certificate verification is successful
*Mar 16 03:28:50.790: CRYPTO_PKI: Rcvd request to end PKI session A069C.
*Mar 16 03:28:50.790: CRYPTO_PKI: PKI session A069C has ended. Freeing all resources.:cert refcount aft
*Mar 16 03:28:50.791: <<< ??? [length 0005]
*Mar 16 03:28:50.791: 16 03 03 00 93
*Mar 16 03:28:50.791:
*Mar 16 03:28:50.791: SSL_connect:SSLv3/TLS read server certificate
*Mar 16 03:28:50.791: <<< TLS 1.2 Handshake [length 0093], ServerKeyExchange
*Mar 16 03:28:50.791: 0C 00 00 8F 03 00 17 41 04 3D 49 34 A3 52 D4 EB
*Mar 16 03:28:50.791: DE A2 9E CC B0 91 AA CB 1B 39 D0 26 1B 7D FF 31
*Mar 16 03:28:50.792: E0 D7 D5 9C 75 C0 7D 5B D6 B2 0A B5 CC EA E1 4B
*Mar 16 03:28:50.792: 4E E5 72 7B 54 5D 9B B2 95 91 E0 CC D6 A5 8E CE
*Mar 16 03:28:50.792: 8D 36 C9 83 42 B0 4D AC 0C 04 03 00 46 30 44 02
*Mar 16 03:28:50.792: 20 67 B3 F1 DA D1 BF 13 72 DD B6 B2 11 3B 6E 6F
*Mar 16 03:28:50.793: 87 52 D9 00 F7 44 31 C3 C2 5E BE 2D FF 93 4E F0
*Mar 16 03:28:50.793: A8 02 20 24 42 91 BE B7 10 1C D1 C0 12 28 FB 1F
*Mar 16 03:28:50.793: E4 DE 81 0B AA 66 19 CD 28 5A A0 30 7D 3C 4A 56
*Mar 16 03:28:50.793: 0D 94 E2
*Mar 16 03:28:50.793:
*Mar 16 03:28:50.794: P11:C_FindObjectsInit:
*Mar 16 03:28:50.794: CKA_CLASS: PUBLIC KEY
*Mar 16 03:28:50.794: CKA_KEY_TYPE: : 00 00 00 03

*Mar 16 03:28:50.794: CKA_ECDSA_PARAMS:
30 59 30 13 06 07 2A 86 48 CE 3D 02 01 06 08 2A
86 48 CE 3D 03 01 07 03 42 00 04 63 B6 D3 1A 28
<snip>

*Mar 16 03:28:50.796: P11:C_FindObjectsFinal
*Mar 16 03:28:50.796: P11:C_VerifyInit - Session found
*Mar 16 03:28:50.796: P11:C_VerifyInit - key id = 131073
*Mar 16 03:28:50.796: P11:C_Verify
*Mar 16 03:28:50.800: P11:CEAL:CRYPTO_NO_ERR
*Mar 16 03:28:50.800: <<< ??? [length 0005]
*Mar 16 03:28:50.800: 16 03 03 00 04
*Mar 16 03:28:50.800:
*Mar 16 03:28:50.800: SSL_connect:SSLv3/TLS read server key exchange
*Mar 16 03:28:50.800: <<< TLS 1.2 Handshake [length 0004], ServerHelloDone
*Mar 16 03:28:50.801: 0E 00 00 00
*Mar 16 03:28:50.801:
*Mar 16 03:28:50.801: SSL_connect:SSLv3/TLS read server done
*Mar 16 03:28:50.810: >>> ??? [length 0005]
*Mar 16 03:28:50.810: 16 03 03 00 46
*Mar 16 03:28:50.811:
*Mar 16 03:28:50.811: >>> TLS 1.2 Handshake [length 0046], ClientKeyExchange
*Mar 16 03:28:50.811: 10 00 00 42 41 04 26 C3 EF 02 05 6C 82 D1 90 B3
*Mar 16 03:28:50.811: 17 31 9A CD DD 8C 81 91 BA E8 C0 86 40 7B 2C E4
*Mar 16 03:28:50.811: 9A 2C 18 9D D1 6A C0 56 A0 98 2E B7 3B AB B3 EB
*Mar 16 03:28:50.811: BB CD 5E 42 C5 76 C0 C4 BF 15 F4 87 F2 7C AD 74
*Mar 16 03:28:50.812: 97 0A 97 2B 06 B5
*Mar 16 03:28:50.812:
*Mar 16 03:28:50.812: SSL_connect:SSLv3/TLS write client key exchange
*Mar 16 03:28:50.812: >>> ??? [length 0005]
*Mar 16 03:28:50.812: 14 03 03 00 01
*Mar 16 03:28:50.812:
```

```
*Mar 16 03:28:50.812: >>> TLS 1.2 ChangeCipherSpec [length 0001]
*Mar 16 03:28:51.116: >>> ??? [length 0005]
*Mar 16 03:28:51.116: 17 03 03 00 35
*Mar 16 03:28:51.116:
*Mar 16 03:28:51.116: >>> ??? [length 0005]
*Mar 16 03:28:51.116: 17 03 03 00 1A
*Mar 16 03:28:51.116:
*Mar 16 03:28:51.116: >>> ??? [length 0005]
*Mar 16 03:28:51.116: 17 03 03 00 30
*Mar 16 03:28:51.116:
*Mar 16 03:28:51.116: >>> ??? [length 0005]
*Mar 16 03:28:51.116: 17 03 03 00 1B
*Mar 16 03:28:51.117:
*Mar 16 03:28:51.713: <<< ??? [length 0005]
*Mar 16 03:28:51.713: 17 03 03 00 6D
*Mar 16 03:28:51.713:
*Mar 16 03:28:51.714: >>> ??? [length 0005]
*Mar 16 03:28:51.714: 17 03 03 00 1E
*Mar 16 03:28:51.714:
*Mar 16 03:28:51.732: <<< ??? [length 0005]
*Mar 16 03:28:51.732: 17 03 03 00 71
*Mar 16 03:28:51.732:
```

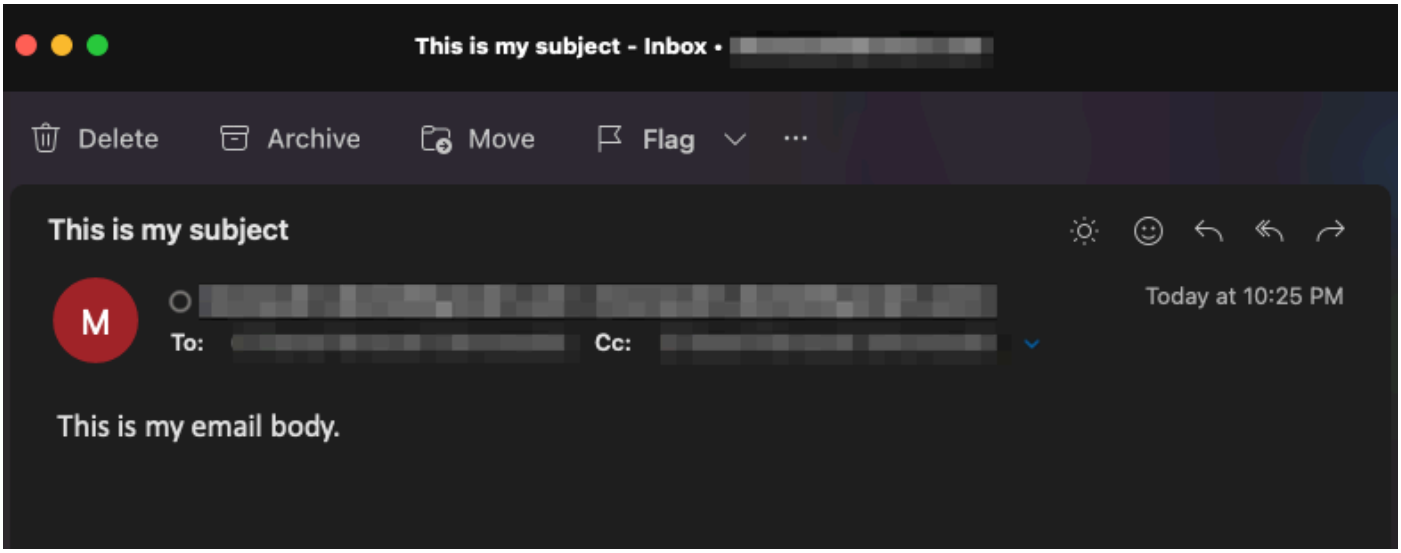
486541296 #事件管理器运行SendSecureEmailEEM*Mar 16 03:28:50.673：CRYPTO_OPSSL：为OPSSLContext*Mar 16 03:28:50.673： CRYPTO_OPSSL：将密码规范设置为掩码0x02FC0000（版本128*Mar 16 03:28:50.674）：设置默认密码曲线列表： 0x70设置EC曲线列表： secp521r1：secp384r1：prime256v1*Mar 16 03:28:50.674： opssl_SetPKIInfo entry*Mar 16 03:28:50.674： CRYPTO_PKI： (A069B)会话已启动-已选择身份(TP-self-signed) -486541296：refcount after increment = 1*Mar 16 03:28:50.674： CRYPTO_PKI： Begin local cert chain retrieval.*Mar 16 03:28:50.674： CRYPTO_PKI(Cert Lookup) issuer="cn=IOS-Self-Signed-Certificate-486541296" serial number= 01*Mar 16 03:28:50.674： CRYPTO：查找句柄=F1EE523CE0， digest=1C 7F 3D 52 67 66 D5 59 E2 66 58 E7 8B E7 9B 8E*Mar 16 03:28:50.675： CRYPTO_PKI：完成本地证书链获取0.*Mar 16 03:28:50.675：CRYPTO_PKI：Rcvd请求结束PKI会话9A06 B.*Mar 16 03:28:50.675： CRYPTO_PKI： PKI session A069B has ended.释放所有资源。TP-self-signed-486541296：unlocked trustpoint TP-self-signed-486541296， refcount为0*Mar 16 03:28:50.675： opssl_SetPKIInfo done.*Mar 16 03:28:50.675： CRYPTO_OPSSL：此会话禁用通用标准。在SSL CTX 0x7F41F28EAFF8上禁用通用标准模式功能3月16日03:28:50.675：CRYPTO_OPSSL：密码套件ECDHE-RSA-AES256-GCM-SHA384：ECDHE-ECDSA-AES256-GCM-SHA384：ECDHE-RSA-AES256-GCM -SHA384：DHE-RSA-AES256-SHA256：AES256-GCM-SHA384：AES256-SHA256：ECDHE-RSA-AES128-GCM-SHA256：ECDHE-ECDSA-AES128-GCM-SHA256：ECDHE RSA-AES128-SHA256：DHE-RSA-AES128-GCM-SHA256：DHE-RSA-AES128-SHA256：AES128-GCM-SHA256：AES128-SHA256*Mar 16:28:50.676：握手start： before SSL initialization*Mar 16 03:28:50.676： SSL_connect：before SSL initialization*Mar 16 03:28:50.676： >>> ???[长度0005]*3月16日03:28:50.676:16 03 01 00 95*3月16日03:28:50.676： *3月16日03:28:50.676： >>> TLS 1.2握手[长度0095]，ClientHello*Mar 16 03:28:50.676:000000 1 03 03 26 4B 9F B3 44 94 FD 5F FD A1<截图>*3月16日03:28:50.679： 03 03 01 02 01*3月16日0 3:28:50.679： *Mar 16 03:28:50.679： SSL_connect：SSLv3/TLS write client hello*Mar 16 03:28:50.692： << ???[长度0005]*3月16日03:28:50.692： 16 03 03 00 3F*3月16日03:28:50.692： *3月16日03:28:50.692：SSL_connect：SSLv3/TLS写入客户端hello*3月16日03:28:50.692：< TLS 1.2握手[长度003f]，ServerHello*Mar 16 03:28:50.692： 02 00 00 3B 03 03 64 12 7E 05 25 F6 7A BD A0 2E*Mar 16 03:28:50.692： 58 83 12 7F 90 CD F4 AB E2 69 53 A8 C7 FC 44 4F*Mar 16 3:28:50.692： 57

4E 47 52 44 01 00 C0 2B 00 00 13 00 17 00 00*3月16日03:28:50.693： FF 01 00 00 0B 00 02 01 00 00 23 00 00*3月16 03:28:50.693：服务器扩展“unknown” (id=23)， len=0TLS服务器扩展“renegotiate” (id=65281)， len=1*Mar 16 03:28:50.693： 00*Mar 16 03:28:50.693： TLS服务器扩展“EC点格式” (id=11)， len=2*Mar 16 03:28:50.693： 01 0*Mar 16 03:28:50.69 3： TLS服务器扩展“会话票证” (id=35)， len=0*Mar 16 03:28:50.693： <<< ???[长度0005]*Mar 16 03:28:50.693： 16 03 03 0F 9A*Mar 16 03:28:50.694： *Mar 16 03:28:50.702： SSL_connect：SSLv3/TLS读取服务器hello*Mar 16 03:28:50.702： << TLS 1.2握手[长度0F9a]， Certificate*Mar 16 03:28:50.702： 0B 00 0F 96 00 0F 93 00 04 8A 30 82 04 86 30 82*Mar 16 03:28:50.702： 03 6E A0 03 02 01 02 02 10 52 87 E0 40 A4 FE F7<snip>*Mar 16 3:28:50.763： 82 35 CF 62 8B C9 24 8B A5 B7 39 0C BB 7E 2A 41*3月16日03:28:50.763： BF 52 CF FC A2 96 B6 C2 82 3F*Mar 16 03:28:50.763： *Mar 16 03:28:50 765： CC_DEBUG：正在进入填充层应用程序回调函数*Mar 16 03:28:50.765： CRYPTO_PKI： (A069C)会话已启动-未指定标识*Mar 16 03:28:50.765： CRYPTO_PKI： (A069C)正在添加对等体证书*Mar 16 03:28:50.767： CRYPTO_PKI：已添加x 509对等证书- (1162)字节*Mar 16 03:28:50.767： CRYPTO_PKI： (A069C)添加对等证书*Mar 16 03:28:50.768： CRYPTO_PKI：添加了x509对等证书- (1434)字节*Mar 16 03:28:50.768： CRYPTO_PKI： (A0 69C)添加对等证书*Mar 16 03:28:50.770： CRYPTO_PKI：已添加x509对等证书- (1382)字节*Mar 16 03:28:50.770： CRYPTO_OPSSL：验证证书链回调*Mar 16 03:28:50.770： CRYPTO_PKI（证书查找） issuer="cn=GTS 1CA C3，o=Google Trust Services LLC，c=US"序列号= 52 87 E0 40 A4 FE F7 07 12 68 B0 4F DD F0 F4*Mar 16 03:28:50.770： CRYPTO_PKI：查找句柄中的证书=7F41EE523CE0， digest=A7 CC 4B 0F 36 C3 Ac d 1 2F 77 DD 1D 9A 37 DC FC*Mar 16 03:28:50.770： CRYPTO_PKI（证书查找）issuer="cn=GTS Root R1，o=Google Trust Services LLC，c=US"序列号= 02 03 BC 53 59 6B 34 C7 18 F5 01 50 66*Mar 16 03:28:50.7 71： CRYPTO_PKI：查找句柄中的证书=7F41EE523CE0， digest=03 9F CF 59 82 EE 09 CC 4F 53 AE D8 02 7E 4B AF*Mar 16 03:28:50.771： CRYPTO_PKI（证书查找）issuer="cn=GlobalSign Root CA，ou=Root CA，o=Global sign nv-sa，c=BE"序列号= 77 BD 0D 6C DB 36 F9 1A EA 21 0F C4 F0 58 D3 0D*Mar 16 03:28:50.771： CRYPTO_PKI：查找句柄中的证书=7F41EE523CE0， digest=94 40 D1 90 A0 A3 5D 47 E5 B5 31 6 63 AD 1B 0A*Mar 16 03:28:50.771： CRYPTO_PKI： Cert record not found for issuer serial.*Mar 16 03:28:50.772： CRYPTO_PKI： crypto_pki_get_cert_record_by_subject()*Mar 16 03:28:50.772： CRYPTO_PKI： Found a subject match*Mar 16 03：#28：50.50 772： CRYPTO_PKI： ip-ext-val： IP extension validation not required：Increasing refcount for context id-35 to 1*Mar 16 03:28:50.773： CRYPTO_PKI： create new ca_req_context type PKI_VERIFY_CHAIN_CONTEXT，ident 35*Mar 16 03:28:50.773： CRYPTO_PKI： (A069C)验证路径有1个证书16 03:28:50.773： CRYPTO_PKI： (A069C)检查相同的证书*Mar 16 03:28:50.773： CRYPTO_PKI（证书查找）issuer="cn=GlobalSign Root CA，ou=Root CA，o=GlobalSign-sa，c=BE"序列号= 77 BD 0D 6C DB 36 F9 1A EA 2 1 0F C4 F0 58 D3 0D*Mar 16 03:28:50.774： CRYPTO_PKI： looking cert in handle=7F41EE523CE0， digest=94 40 D1 90 A0 A3 5D 47 E5 B5 31 F6 63 AD 1B 0A*Mar 16 03:28:50.774： CRYPTO PKI：未找到颁发者序列号的证书记录。*Mar 16 03:28:50.774： CRYPTO_PKI ： (A069C)正在验证不受信任的证书*Mar 16 03:28:50.774： CRYPTO_PKI： (A069C)创建合适的信任点列表*Mar 16 03:28:50.774： CRYPTO_pki_get_cert_record_by_issuer()*Mar 16 03:28:50.774： CRYPTO_PKI： Found a issuer match*Mar 16 03:28:50.774： CRYPTO_PKI： (A069C)合适的信任点是： CA-GlobalSign-Root，*Mar 16 03:28:50.775： CRYPTO_PKI： (A069C)正在尝试验证证书使用CA-GlobalSign-Root策略*Mar 16 03:28:50.775： CRYPTO_PKI： (A069C)使用CA-GlobalSign-Root验证证书*Mar 16 03:28:50.775： CRYPTO_PKI（建立受信任的证书链）*Mar 16 03:28:50.775： CRYPTO_PKI：添加了1个证书到受信任链。*Trusted 16 03:28:50.775： CRYPTO_PKI：准备会话撤销服务提供商*3月16日03:28:50.776： P11：C_CreateObject：*Mar 16 03:28:50.776：

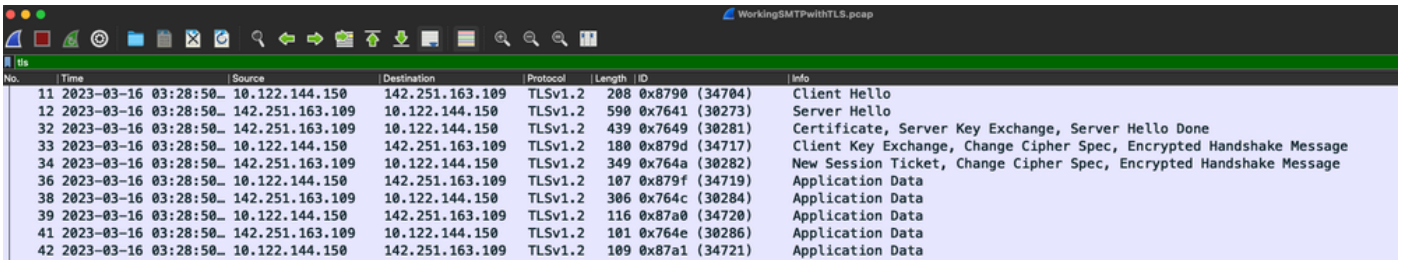
CKA_CLASS： PUBLIC KEY*Mar 16 03:28:50.776： CKA_KEY_TYPE： RSA*Mar 6 03:28:50.776： CKA_MODULUS： DA 0E E6 99 8D CE A3 E3 4F 8A 7E FB F1 8B 83 25 6B EA 48 1F F1 2A B0 B9 95 11 04 BD F0 63 D1 E2 <snip>*Mar 16 03:28:50.780： CKA_public_EXPONENT： 01 00 01*Mar 16 03:28:50.780： CKA_VERIFY_RECOVER： 01*Mar 16 03:28:50.780： CRYPTO_PKI：删除密钥ID为45*Mar 16 03:28:50.781： CRYPTO_PKI：尝试将对等体的公钥插入缓存*Mar 16 03:28:50 781： CRYPTO_PKI：Peer's public inserted with key id 46*Mar 16 03:28:50.781： P11：C_CreateObject： 131118*Mar 16 03:28:50.781： P11：C_GetMechanismInfo slot 1 type 3 (invalid mechanism)*Mar 16 03:28:50.781： P111：P1：C_Get1：Mechanism info slot 1类型1*Mar 16 03:28:50.781： P11：C_VerifyRecoverInit - 131118*Mar 16 03:28:50.781： P11：C_VerifyRecover - 131118*Mar 16 03:28:50.781： P11：found pubkey in cache using index = 46*Mar 16 03:28:50.781： P 1：找到的公钥是：30 82 01 22 30 0D 06 09 2A 86 48 86 F7 0D 01 01 01 05 00 03 82 01 0F 00 30 82 01 0A 02 82 01 01 <snip>CF 02 03 01 00 01*Mar 16 03:28:50.788： P1：CEAL ：CRYPTO_NO_ERR*Mar 16 03:28:50.788： P11：C_DestroyObject 2:2002E*Mar 16 03:28:50.788： CRYPTO_PKI：密钥id为46*Mar 16 03:28:50.788： CRYPTO_PKI： (A069C)证书已验证*Mar 16 03:28:50.788： CRYPTO_PKI：删除会话撤销服务提供商*Mar 16 03:28:50.788： CRYPTO_PKI：删除会话撤销服务提供商CA-GlobalSign-Root：validation status - CRYPTO_VALID_CERT_WITH_WARNING*Mar 16 03:28:50.788： CRYPTO_PKI： (A069C)证书验证无需撤销检查：证书refcount after increment = 1*Mar 16 03:28:50.790： CRYPTO_PKI： Populate AAA auth data*Mar 16 03:28:50.790： CRYPTO_PKI： Unable to get configured attribute for primary AAA list authorization.*Mar 16 03:28:50.790： PKI： Cert key-usage： Signature， Certificate-Signing， CRL-SiGNING 16 03:28:50.790： CRYPTO_PKI： (A069C)链证书已锚定到信任点CA-GlobalSign-Root，并且链验证结果为： CRYPTO_VALID_CERT_WITH_WARNING*Mar 16 03:28:50.790： CRYPTO_PKI： (A069C)正在删除验证上下文*Mar 16 03:28:50.7 0： CRYPTO_PKI：销毁ca_req_context type PKI_VERIFY_CHAIN_CONTEXT，ident 35， ref count 1：将上下文id-35的refcount减少至0*Mar 16 03:28:50.790： CRYPTO_PKI： ca_req_context released*Mar 16 03:28:50.790： CRYPTO_PKI： (A069C)验证TPCA-Global sign-Root*Mar 16 03:28:50.790： CRYPTO_PKI： (A069C)证书验证成功*Mar 16 03:28:50.790： CRYPTO_OPSSL：证书验证成功*Mar 16 03:28:50.790： CRYPTO_PKI： Rcvd请求结束PKI会话A069C.*Mar 16:28 50.790： CRYPTO_PKI： PKI session A069C has ended.释放所有资源。：cert refcount after decrement = 0*Mar 16 03:28:50.791： <<< ???[长度0005]*3月16日03:28:50.791:16 03 03 00 93*3月16日03:28:50.791：*3月16日03:28:50.791：SSL_connect：SSLv3/TLS读取服务器证书*3月16日03:28:50.791：<< TLS 1.2 Handshake [length0009 3]， ServerKeyExchange*Mar 16 03:28:50.791： 0C 00 00 8F 03 00 17 41 04 3D 49 34 A3 52 D4 EB*Mar 16 03:28:50.791： DE A2 9E CC B0 91 AA CB 1B 39 D0 26 1B 7D FF 31*1 6 03:28:50.792： E0 D7 D5 9C 75 C0 7D 5B D6 B2 0A B5 CC EA E1 4B*Mar 16 03:28:50.792： 4E E5 72 7B 54 5D 9B B2 95 91 E0 CC D6 A5 8E CE*Mar 16 03:28:50.7 2： 8D 36 C9 83 42 B0 4D AC 0C 04 03 00 46 30 44 02*3月16日03:28:50.792： 20 67 B3 F1 DA D1 BF 13 72 DD B6 B2 11 3B 6E 6F*Mar 16 03:28:50.793： 8752 D9 0 f7 44 31 C3 C2 5E BE 2D FF 93 4E F0*Mar 16 03:28:50.793： A8 02 20 24 42 91 BE B7 10 1C D1 C0 12 28 FB 1F*Mar 16 03:28:50.793： E4 DE 810B AA 66 19 CD 28 a A0 30 7D 3C 4A 56*Mar 16 03:28:50.793： 0D 94 E2*Mar 16 03:28:50.793： *Mar 16 03:28:50.794： P11：C_FindObjectsInit：*Mar 16 03:28:50.794： CKA_CLASS：公钥3月16日03:28:50.794： CKA_KEY_TYPE： ： 00 00 00 03*3月16日03:28:50.794： CKA_ECDSA_PARAMS： 30 59 30 13 06 07 2A 86 48 CE 3D 02 01 06 08 2A 86 48 CE 3D 03 01 073 2 00 04 63 B6 D3 1A 28 <snip>*Mar 16 03:28:50.796： P11：C_FindObjectsFinal*Mar 16 03:28:50.796： P11：C_VerifyInit -找到的会话*Mar 16 03:28:50.796： P11：C_VerifyInit -密钥id = 131073*Mar 6

03:28:50.796： P11：C_Verify*Mar 16 03:28:50.800： P11：CEAL：CRYPTO_NO_ERR*Mar 16 03:28:50.800： << ???[长度0005]*3月16日03:28:50.800:16 03 03 00 04*3月16日03:28:50.800： *3月16日03:28:50.800：SSL_connect：SSLv3/TLS读取服务器密钥交换*3月16日03:28:50.800： << TLS 1.2握手[长度00 4]， ServerHelloDone*Mar 16 03:28:50.801： 0E 00 00 00*Mar 16 03:28:50.801： *Mar 16 03:28:50.801： SSL_connect：SSLv3/TLS读取服务器完成*Mar 16 03:28:50.810： >>> ???[长度0005]*3月16日03:28:50.810:16 03 03 00 46*3月16日03:28:50.811： *3月16日03:28:50.811： >>> TLS 1.2握手[长度0046]，客户端密钥交换*3月16日03:28:50.811:100 0 42 41 04 26 C3ef 02 05 6C 82 D1 90 B3*Mar 16 03:28:50.811： 17 31 9A CD DD 8C 81 91 BA E8 C0 86 40 7B 2C E4*Mar 16 03:28:50.811： 9A 2C 18 9D1 6A C0 5 A0 98 2E B7 3B AB B3 EB*Mar 16 03:28:50.811： BB CD 5E 42 C5 76 C0 C4 BF 15 F4 87 F2 7C AD 74*Mar 16 03:28:50.812： 97 0A 97 2B 06 B5*Mar 16 03:28:50.81 2：*Mar 16 03:28:50.812： SSL_connect：SSLv3/TLS写入客户端密钥交换*Mar 16 03:28:50.812： >>> ???[长度0005]*3月16日03:28:50.812:14 03 03 00 01*3月16日03:28:50.812： *3月16日03:28:50.812： >>> TLS 1.2 ChangeCipherSpec [长度0001]*3月16日03:28:51.16：>> ???[长度0005]*3月16日03:28:51.116:17 03 00 35*3月16日03:28:51.116： *3月16日03:28:51.116： >> ???[长度0005]*3月16日 03:28:51.116:17 03 03 00 1A*3月16日03:28:51.116： *3月16日03:28:51.116： >> ???[长度 0005]*3月16日03:28:51.116:17 03 03 00 30*3月16日03:28:51.116： *3月16日03:28:51.116： >> ???[长度0005]*3月16日03:28:51.116:17 03 03 00 1B*3月16日03:28:51.117： *3月16日 03:28:51.713： << ???[长度0005]*3月16日03:28:51.713:17 03 03 00 6D*3月16日03:28:51.713： *3月16日03:28:51.714： >> ???[长度0005]*3月16日03:28:51.714： 17 03 03 00 1E*3月16日 03:28:51.714： *3月16日03:28:51.732： << ???[长度0005]*3月16日03:28:51.732： 17 03 03 00 71*3月16日03:28:51.732：

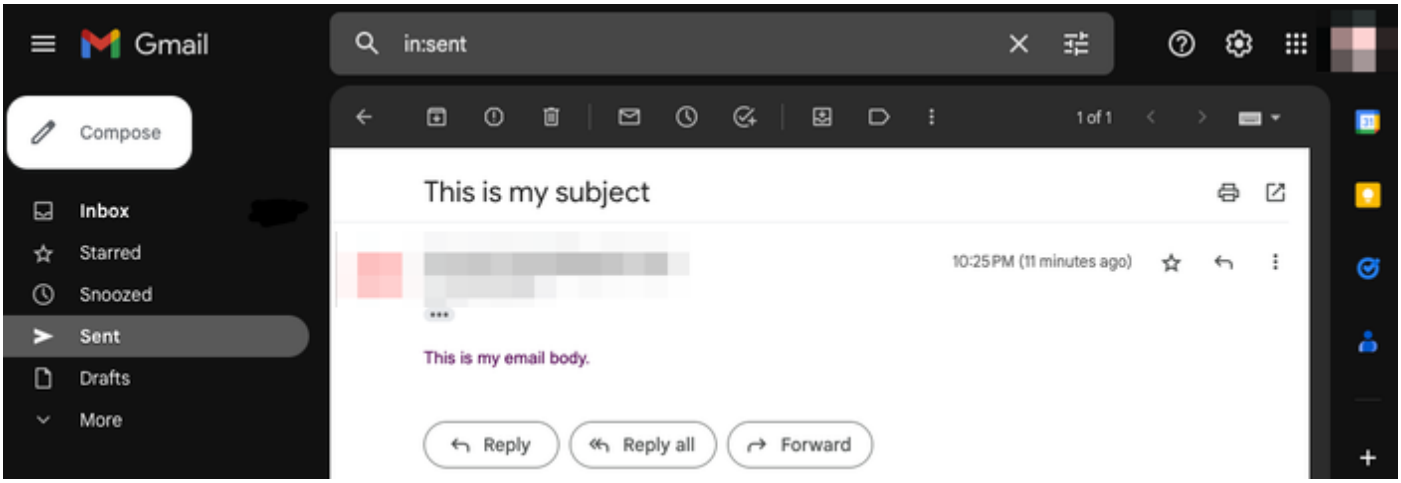
您可以验证是否已收到邮件，并且所有字段（收件人、发件人、抄送、主题、正文）都已正确填充：



您还可以验证从Cisco IOS XE设备上的数据包捕获进行的TLS握手和会话（附加为“WorkingSMTPwithTLS.pcap”）：



您甚至可以验证电子邮件是否反映在所用电子邮件帐户的“已发送”文件夹中：



其他注意事项和注意事项

带@符号的用户名

尝试使用SMTP中继时可能会出现问题。 由于SMTP中继，服务器字符串的格式如下（用户名中为“@”）：

```
event manager environment _email_server email.relay@My.Domain.Name:MyPasswordString@smtp-relay.gmail.co
```

解析用户名和密码的代码会在第一次出现“@”符号时拆分字符串。因此，系统认为服务器主机名在字符串的其余部分第一个“@”符号之后立即启动，并将之前的所有内容解释为“username：password”。

SMTP的TCL实现使用正则表达式(regex)，它以不同方式处理此用户名/密码/服务器信息。由于这种差别，TCL允许用户名带有“@”符号；但是，Cisco IOS XE TCL不支持加密，因此没有通过TLS发送安全邮件的选项。

综述：

- 如果电子邮件需要安全，则不能与TCL一起发送。
- 如果您的用户名中有“@”，则无法通过EEM发送。

我们通过归档思科漏洞ID CSCwe75439来应对此改进EEM电子邮件功能的契机；但目前尚无针对

此增强请求的规划图。

结论

如此处所示，使用嵌入式事件管理器(EEM)小程序，可以通过SMTP使用TLS发送安全邮件。 它需要在服务器端进行某些设置，并配置必要的证书以允许信任，但如果您想要生成自动、安全的邮件通知，它还是可行的。

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）。