Cisco IOS XE和GuestShell CA Trust eMudhra不信任指南

目录

目的

本文档提供针对Cisco IOS® XE的指导，以应对对Mudhra根证书颁发机构和中间证书颁发机构的紧急不信任。

使用此指南审计和管理其配置允许受影响层次结构的服务的CA信任。产品适用性和所需的补救在思科信息中定义。在更改信任之前，检查您的Cisco IOS XE版本的命令支持和服务影响。

Cisco信息：cisco-sa-emudhra-info-ScS22A1r

受影响的Mudhra CA证书：原始和交叉签名

库存于2026年10月29日核实。八个eMudhra颁发CA密钥受到影响：6个用于TLS，2个用于代码签名。每个密钥显示在原始的emSign颁发的CA证书和Sectigo交叉签名的CA证书中。因此，该清单包含16个证书：12个TLS和4个代码签名证书。

每个原始/交叉签名对都具有相同的主题和公钥，但具有不同的颁发者、序列号和SHA-256证书指纹。这两种形式提供备用认证路径，并且必须都包括在审核中。

A.发行商和受影响的证书库存

颁发者A — 原始TLS证书：CN=emSign Root TLS CA - G3,O=Mudhra Technologies Limited，C=IN。

签发方B — 原始代码签名证书：CN=emSign Root CS CA - G3,O=Mudhra Technologies Limited，C=IN。

签发方C — 所有交叉签名的证书：CN=AAA Certificate Services、O=Comodo CA Limited、L=Salford、ST=Greater Manchester、C=GB(Sectigo)。

G3B主题：O=Mudhra BV，C=NL。G3C主题：O=Mudhra Inc， C=US。每个变体信元列出其颁发

者引用、十六进制序列和SHA-256指纹（两行）。 前导零填充不会更改串行值。

受影响的发出CA(CN)	原始证书 签发方/串行（十六进制）/SHA-256	Sectigo交叉签名证书 签发方/串行（十六进制）/SHA-256
1. EM DV TLS CA - G3B TLS	发行商A A4AEE678F23362B4C0E460B772F214 42C4ACCD12C4F59597925E1A08A7A901 02608543809308D50EC427E118BA1493	签发方C A024FDF4BA665C785CE4A020483858B7 45A883F349AEDFD5F3A649B5498FFB12 1712BA8B448F3F2A91DBC5A766FCDDBE
2. EM OVER TLS CA - G3B TLS	发行商A 6CADA367FB1C100088120EA202D568 98B896625C0C230ADEEC3EB2CF6AAD50 01ECF9156A68644E82374CE5833BC25B	签发方C E94223BDE92F914E5AB9ACABFDAC3710 5F1DED4DD590AF-74631290073B122A 2AAC6A1FA479778A0BD43579BD832699
3. EM EV TLS CA - G3B TLS	发行商A C7C41EFE92D825F05EAE841A4A82C1 E7568FE6E12331F4FD7DE61DA4C7559D 0F8EB192A218D4741A3BA75184333838	签发方C D8F8685E4F2BD9D-8240285A1C1EB731 579480AEA166290BD3C643589E795FC0 D0E54E86505D180F1DFF99EFD3199512
4. EM DV TLS CA - G3C TLS	发行商A 588B2E3C2BB8CDA13E1A1209254321 0164D7CEAE5BB909C272AF151FF44B11 77AE9B47D9424B94F1D9558C30934D19	签发方C 91F347F1A5B7D39962B95E0EE4E299D1 62B7F9868263F72F6911FC78CEAF72C1 979158B7368BF648683BB2AA39B02E72
5. EM OVER TLS CA - G3C TLS	发行商A 296926C227A3A1526392B19482A504 CB6E15165AC802E9B3F993F16C25D41F 4E81C21EBDE71AE1B012AC00BF9C7C3F	签发方C 8D5A4811939EF447D77111CDD9849AE9 7485B51C0ED192878FA1D2F948FB529A E1842CC5086C4B67F7BE5B2039DE9489

受影响的发出CA(CN)	原始证书 签发方/串行（十六进制）/SHA-256	Sectigo交叉签名证书 签发方/串行（十六进制）/SHA-256
6. EM EV TLS CA - G3C TLS	发行商A 8009928A64107601AF05FA2F19F109 432EC25F0D28DE9006D97BF752F17BBB 2CD41F669933FCB072D9A1148D0261F3	签发方C 05F078752DECE7A1BB8595D303167171 FD9D5C2DBBD4995AB69B8D50C439D952 7B00529199C2FD01B2174EB58F07021F
7. EM OVER CS CA - G3B 代码签名	发行商B 7818933F8F85B4619A2B35FBA83D0A 9E39DE2881273ACC225A149C72BA3E0F 51424963669AB633EE1DC7DF5DB067B4	签发方C 41E32B73AC185D6CE0D7480BACF84E69 A496E23AC0CAB8155FC9B117B97EAE36 8EFCBB9041AEC8804D5A308D2D01C2D8
8. EM OVER CS CA - G3C 代码签名	发行商B 544B7F922C31AE220A4A922916F6E6 4B2460A466F95FD40113BCDCCA2CEB29 B3B093921DE45474DA7DF48874C630B1	签发方C D6A5EF613B390F32EF9CDF349F772841 84112972B8E59D14560AE7E711965447 4AAD119990CF3722BFC1E3B91615E1FD

B.区分的认证途径

原始TLS路径

```
emSign ECC Root CA - G3
  +-- emSign Root TLS CA - G3 (issued by emSign ECC Root CA - G3)
       +-- Original EM DV/OV/EV TLS CA - G3B or G3C
            +-- TLS identity certificate
```

原始代码签名路径（直接信任根CS）

```
emSign Root CS CA - G3
  +-- Original EM OV CS CA - G3B or G3C
       +-- Code-signing identity certificate
```

Sectigo交叉签名路径（TLS和代码签名）

```
AAA Certificate Services
  +-- One of the eight cross-signed CAs listed above
       +-- TLS or code-signing identity certificate
```

当CA直接受信任时，TLS路径也可以终止于emSign Root TLS CA - G3。图表显示了相关的发行人关系；接受的路径取决于消费服务的信任锚点和验证策略。

示例中显示的父/根CA提供链上下文，不计入八个受损的发布密钥。删除或不信任父密钥/根密钥的决定与识别被入侵的中间密钥是分开的。

为了进行识别，请将证书颁发机构和序列号与SHA-256指纹一起使用。仅主题名称是不够的。证书指纹识别一个证书变体；spki SHA-256散列可标识其原始和交叉签名的表单共享的公钥。仅删除缓存的中间路径不会阻止对等体通过剩余的受信任路径呈现该中间路径。

来源和验证日期

2026年10月29日，根据公共事件附件对证书标识符进行了验证。每个原始/交叉符号对包含相同的公钥。使用实时思科信息了解产品适用性和补救状态。

Cisco信息：cisco-sa-emudhra-info-ScS22A1r

Mozilla事件记录：Mudhra中间密钥危害(2083449)

原始CA证书附件(9664066)

AAA交叉签名CA证书附件(9664067)

苹果：原始和交叉签名TLS证书路径和指纹

截至2026年10月31日，在本指南列出的受感染CA中，未识别emSign根CA - G1。其持续信任仍取决于您的组织策略和后续的咨询更新。

Cisco IOS XE如何管理证书信任

配置的PKI信任点保存CA/身份证书和验证策略。PKI信任池包含用于使用该信任池的服务的内置和下载CA证书。确定每个相关服务使用的信任源和策略；更改信任池不会替换单独配置的信任点。

1.审计和准备

检查软件版本、设备时间和已安装的证书：

<#root>

```
show version show clock show crypto pki trustpoints show running-config | section crypto pki trustpoint
```

使用主题作为支持上下文，将完整证书颁发者和十六进制序列号与资产进行比较。如果Cisco IOS XE仅显示MD5和SHA1指纹，则无法在该输出中直接比较库存SHA-256指纹；它用于验证设备外部的证书文件。序列号中的前导00填充可以不同。供应商名称搜索仅识别候选者。记录依赖于每个信任点的服务，并检查原始路径和交叉签名路径，包括直接受信任中间体。

示例：根CA和受影响的子CA

显示详细证书的简短示例；其他字段将被省略。比较Subject（主题）、Issuer（颁发者）和序列号。

根CA示例：EmSign ECC根CA - G3

```
CA Certificate
  Certificate Serial Number (hex): 3CF607A968700EDA8B84
  Issuer:
    cn=emSign ECC Root CA - G3
  Subject:
    cn=emSign ECC Root CA - G3
  Associated Trustpoints:
```

受影响的子CA示例：原始的emSign颁发证书

```
CA Certificate
  Certificate Serial Number (hex): 00A4AEE678F23362B4C0E460B772F214
  Issuer:
    cn=emSign Root TLS CA - G3
  Subject:
```

```
    cn=EM DV TLS CA - G3B
  Associated Trustpoints:
```

受影响的子CA示例：Sectigo交叉签名证书

```
CA Certificate
  Certificate Serial Number (hex): 00A024FDF4BA665C785CE4A020483858B7
  Issuer:
    cn=AAA Certificate Services
  Subject:
    cn=EM DV TLS CA - G3B
  Associated Trustpoints:
```

两个SubCA表单具有相同的Subject，但不同的发布者和序列名称。将完整的发行商和序列与库存配对；前导00填充可能不同。记录关联的信任点，以定位拥有的信任点或池。显示的根是链上下文；它不是八个受损的发布密钥之一。

保持恢复访问方法和批准的更换捆绑包可用。根据需要使用批准的证书/密钥备份过程。备份配置；仅此副本不是私钥备份：

<#root>

copy running-config

使用经认可的特定于设备的值替换尖括号占位符。使用show file systems显示的文件系统，如flash:或bootflash:。

2.检查信任路径

删除直接受信任的受影响SubCA

从信任点和下载的信任池捆绑包中删除直接受信任的受影响SubCA。不要依赖吊销设置来覆盖受感染的CA中的直接信任。

可选根CA删除

删除根CA是提供额外保护的可选建议。当实际服务CRL/OCSP检查获取当前状态、评估受信任的父级下受影响的SubCA、拒绝受影响的原始、交叉签名和适用的更长链时，所需的根可以保持受信任，并且如果配置的方法均未提供可用状态，则验证失败。在这些经过验证的条件下，撤销检查可保护已识别的受损发行者，而不删除根。

考虑删除不必要的根/信任锚点，以减少对撤销服务的依赖。首先评估服务依赖性：删除AAA证书服务可能会中断不相关的合法链。覆盖Mudhra和AAA路径。仅删除缓存的SubCA不会阻止对等体通过剩余的受信任父项提供的副本。

删除所选信任点

对于选择删除的管理员配置的信任点：

<#root>

```
configure terminal no crypto pki trustpoint
```

```
        end
```

接受之前请审核确认。信任点删除会销毁其关联的证书。如果设备身份需要替换，请获取与设备密钥匹配的替换证书，并使用服务注册过程更新其服务绑定。仅安装新的CA证书不会替换设备身份。

对于下载的trustpool条目，请使用捆绑包替换过程。内置或功能管理的锚点需要受支持的产品更新或Cisco TAC指导。

3.如有需要，更新下载的Trustpool

当下载的捆绑包需要更换时，请使用以下步骤。在第2步中经过验证的撤销条件下，所需的父/根可以保持受信任状态。

3.1检查是否安装和使用了下载的捆绑包

<#root>

```
show running-config | section crypto pki certificate pool show crypto pki trustpool verbose show crypto
```

在crypto pki certificate pool和CA记录下查找标记为Trustpool的电缆条目：已下载。捆绑包文件名可能不同于ios.p7b或ios_core.p7b。单独存储的文件或配置的刷新URL并不能证明其证书是服务安装或使用的。确认服务使用信任池并将其CA记录与受影响的资产进行比较。如果未安装已下载的CA，请跳过清理和替换；仍评估其他信任来源。

3.2清理下载的信任池

在清理之前保留恢复访问和已批准的恢复捆绑包。如果传输依赖于正在删除的信任，则稍后下载可能会失败。制定相关服务中断的计划。

<#root>

```
configure terminal crypto pki trustpool clean end
```

```
show crypto pki trustpool verbose
```

确认下载的CA记录已丢失。清除操作会删除所有下载的信任池证书及其管理的捆绑文件；内置CA和独立信任点仍然存在。在更换和服务检查成功之前不要保存。

3.3获取并安装批准的更换捆绑包

获取经批准的思科签名的捆绑包。为所需服务选择ios.p7b或ios_core.p7b，并确认其不包括上面确定的受影响证书和不需要的信任锚。2026年10月31日检查的传统捆绑包在ios.p7b中包含emSign根CA - G1，但在ios_core.p7b中既不包含eMudhra也不包含AAA证书服务CA。Core的信任集较小：在切换之前检查服务依赖性，并验证您安装的捆绑包的内容。使用已批准的HTTPS或SCP传输将其暂存到托管trustpool存储之外。

思科完整CA捆绑包：ios.p7b

思科核心CA捆绑包：ios_core.p7b

<#root>

show file systems copy

configure terminal crypto pki trustpool import url

end show crypto pki trustpool verbose

例如，分级URL可以是flash:approved-ios.p7b。如果导入失败，请在保存之前还原已批准的捆绑包。检验预定的CA和所需的连接。清理不是证书拒绝列表：确认剩余信任点、内置CA和备用路径通过强制撤销或信任路径删除拒绝受影响的链。

4.配置验证和刷新策略

在进行更改之前，请记录现有的信任池策略、受影响的信任点策略、证书映射绑定和刷新URL，以便在需要时恢复它们。

对于相关配置的信任点：

<#root>

configure terminal crypto pki trustpoint

revocation-check crl ocsp exit end

对于使用trustpool策略的服务：

<#root>

```
configure terminal crypto pki trustpool policy chain-validation revocation-check crl ocsp exit end show
```

检查显示的信任池策略是否继续验证到根并列出crl ocsp。当CRL检索不可用时，设备首先尝试CRL并尝试OCSP。当评估受影响的证书时，撤销的结果未通过验证。当需要成功的撤销检查时，请勿添加none。

验证正确的时间、可访问的CA发布状态终端、接受的响应格式以及不存在适用的撤销跳过异常。受影响的SubCA状态必须在可信的父级下评估。SubCA中直接信任可以在检查其父级之前停止验证。验证实际服务是否获得可用的、当前状态并拒绝受影响的链。如果无法验证状态 — 服务器访问和拒绝行为，则不要将配置单独视为已确认保护；根据您的缓解计划需要删除相关信任路径。请勿添加none以绕过不可用状态。

在验证保护之前刷新缓存的CRL

即使CA发布更新的CRL，本地缓存/数据库中已存在的可用CRL也可以重新使用。检查状态 — 服务器可达性，然后在验证保护之前清除旧的CRL:

<#root>

```
show crypto pki crls clear crypto pki crl show crypto pki crls
```

clear命令删除所有信任点的本地CRL;它本身不会下载替代产品。确认旧条目已消失，然后在受影响的服务中启动新的连接或证书验证。再次运行show crypto pki crls，并根据CA的当前发布检查每个相关的颁发者、LastUpdate、NextUpdate和分发点。包括涵盖受影响子CA的父颁发者CRL，而不仅仅包括ID证书颁发者CRL。确认已拒绝受影响的链。清除可能会在获取状态数据时暂时影响新连接；保留失效关闭验证。

查看捆绑包刷新源

对于连接的设备，可选择性地配置批准的到期驱动捆绑刷新源：

<#root>

```
configure terminal crypto pki trustpool policy cabundle url
```

```
exit end show crypto pki trustpool policy
```

查看所有已配置的捆绑包源。到期驱动的刷新不会替换立即安全更新，而且不能恢复不需要的信任路径。

可选的证书映射过滤器：覆盖和限制

Cisco IOS XE证书映射(cert-map)匹配证书属性，例如颁发者名称。此外，该过滤器还可以帮助拒绝由列出的八个CA直接颁发的ID（枝叶）证书。应用现有匹配证书策略之前，请查看信任点。

行为在2026年10月29日使用实验室身份在EM DV TLS CA - G3B下在实验实验室设备上测试。在PKI对等证书验证中，过滤器拒绝直接颁发的身份，但并未阻止通过附加子SubCA颁发的身份；一个未上市的合法发行人通过。行为覆盖使用一个受影响的颁发者，仅在临时实验室信任点上禁用撤销以隔离过滤器。应用握手和其他版本未使用此过滤器进行验证；在依赖实际服务之前，先对其进行测试。

<#root>

```
configure terminal crypto pki certificate map EMSIGN-ISSUER-FILTER 10 issuer-name nc cn=EM DV TLS CA - G
```

```
match certificate EMSIGN-ISSUER-FILTER exit end
```

在此，nc的意思是“不包含”。 将所有8个负发行人名称条件保留在同一条目中，序列10:所有证书必须匹配才能通过此过滤器。匹配任何列出的名称的直接颁发者无法通过过滤器。

Cisco IOS XE可以显示规范化的条目，例如issuer-name nc cn = em dv tls ca - g3b。行为测试中存在此小写和间隔格式。确认映射及其与服务实际使用的验证信任点的绑定：

<#root>

show running-config | section crypto pki certificate map show running-config | section crypto pki trustp

过滤器不提供链范围的CA拒绝列表。例如：

```
Trusted root CA
  +-- Affected SubCA
       +-- New child SubCA
            +-- ID (leaf) certificate
```

ID证书颁发者是新的子SubCA，因此，其issuer-name过滤器不会识别链中更高层受到影响的CA。其他检查（包括CA限制和撤销）仍然适用。请仅将此过滤器用作与本指南中的撤销验证和可选的根删除过程一起使用的附加保护。测试直接颁发、子SubCA以及具有实际服务的合法证书。

第4步的回滚

恢复记录的吊销方法、链验证设置和证书映射绑定。删除通过此过程添加的刷新URL，并恢复任何以前更改的URL。此示例仅删除已添加的刷新URL和过滤器绑定；它不会恢复其他策略设置：

<#root>

configure terminal crypto pki trustpool policy no cabundle url

exit

crypto pki trustpoint

no match certificate EMSIGN-ISSUER-FILTER exit end

仅在此过程添加了URL和绑定的位置运行这些删除操作。如果替换了以前的证书映射绑定，请恢复该绑定。只有在确认没有任何其他配置引用新创建的映射后，才将其删除。

仅当重置为默认值是预期的、已批准的结果时，才使用no crypto pki trustpool policy。此宽重置清除配置的刷新URL，并将撤销检查重置为none;它不会恢复以前的自定义策略。策略回滚不会还原已删除的证书、还原安装的捆绑包或还原清除的CRL。在保存之前重新检查所需的服务和受影响的链拒绝。

5.检验保护并保存

- 确认拒绝受影响的原始链和交叉符号链，包括对等体提供的中间链。如果较长的链相关，当受影响的CA在链中较高时验证拒绝。
- 确认所需的合法连接仍然有效，并且任何替换身份已绑定到目标服务。
- 当吊销数据不可用时检查行为。现有TLS/IPsec会话不会自动通过策略更改进行验证；如果需要，请使用服务支持的重新连接过程。

检查成功后：

<#root>

```
show crypto pki certificates verbose show crypto pki trustpool policy write memory
```

在下次计划重新加载后以及在备用设备上（如果适用）验证持久性。如果结果与预期的信任策略不同，请保留日志并联系思科TAC，然后考虑已验证的信任更改。

进一步阅读

Cisco IOS XE PKI Trustpool管理

思科:使用Cisco IOS XE PKI配置CA签名证书

思科:CRL缓存和撤销检查

Cisco IOS XE GuestShell CA信任指南

步骤 1：审核设备GuestShell部署状态

仅在安装了GuestShell且处于RUNNING或DEPLOYED状态的Cisco IOS XE系统上需要补救。如果GuestShell处于DEPLOYED状态，必须先使用“guestshell enable”命令启用它，然后才能执行补救命令。如果show app-hosting list指示“未找到应用”（GuestShell未部署或未启用），则不需要对GuestShell执行任何操作。

A.处于DEPLOYED状态的GuestShell(需要操作：必须首先启用)

当show app-hosting list报告GuestShell处于DEPLOYED状态时，GuestShell容器存储存在，但容器已停止。必须在输入bash之前运行“guestshell enable”（如果返回停止状态，则可以在完成补救之后运行“guestshell disable”）：

```
<#root>

Router#

show app-hosting list

App id                                   State
------------------------------------------------------------
guestshell                               DEPLOYED
```

B.处于RUNNING状态的GuestShell(需要操作：直接继续)

当show app-hosting list报告GuestShell处于RUNNING状态时，GuestShell处于主动运行状态，您可以直接进入bash:

```
<#root>

Router#

show app-hosting list

App id                                   State
------------------------------------------------------------
guestshell                               RUNNING
```

B.未启用/未部署GuestShell（无需操作）

如果show app-hosting list指示No App found，GuestShell未在系统上安装或启用，并且无需进一步操作：

<#root>

```
Router#

show app-hosting list

No App found
```

步骤 2：访问GuestShell并应用CA不信任

从Cisco IOS XE特权EXEC('enable')提示符执行这些命令。如果GuestShell处于DEPLOYED状态，请先启用它。然后输入bash shell，升级到根目录，并使用动态文件夹选择配置CA阻止列表以确保所有Cisco IOS XE版本的兼容性：

<#root>

```
! 1. If GuestShell is in DEPLOYED state, enable it first:
Router#

guestshell enable


! 2. Enter GuestShell bash shell:
Router#

guestshell run bash


! 3. Escalate to root user inside GuestShell:
[guestshell@guestshell ~]$

sudo -s


! 4. Prepare the ca-trust blocklist directory (compatible with 17.15.x and 17.18.x):
d=/etc/pki/ca-trust/source/blocklist
[ -d "$d" ] || d=/etc/pki/ca-trust/source/blacklist
mkdir -p "$d"
```

不信任受损的eMudhra证书

执行trust extract命令将每个受损的eMudhra证书直接提取到ca-trust blocklist目录中，然后重建系统信任库：

```
trust extract --overwrite --format=pem-bundle \
  --filter='pkcs11:object=emSign%20ECC%20Root%20CA%20-%20G3;type=cert' \
  "$d/emSign_ECC_Root_CA_-_G3.pem"

trust extract --overwrite --format=pem-bundle \
  --filter='pkcs11:object=Comodo%20AAA%20Services%20root;type=cert' \
  "$d/Comodo_AAA_Services_root.pem"

trust extract --overwrite --format=pem-bundle \
  --filter='pkcs11:object=emSign%20ECC%20Root%20CA%20-%20C3;type=cert' \
  "$d/emSign_ECC_Root_CA_-_C3.pem"

trust extract --overwrite --format=pem-bundle \
  --filter='pkcs11:object=emSign%20Root%20CA%20-%20C1;type=cert' \
  "$d/emSign_Root_CA_-_C1.pem"

update-ca-trust extract
```

步骤 3：在GuestShell中验证不受信任的证书

使用下面的验证命令，验证四个提取的证书是否出现在系统不信任/阻止列表策略中，并且已从活动信任锚中排除。该命令确认证书已在阻止列表策略中注册。

```
###Verify with below command
[guestshell@guestshell ~]$ trust list --filter=trust-policy --purpose=server-auth | grep -E -A3 'emSign
    label: AAA Certificate Services
    trust: distrusted
    category: authority

--
    label: emSign ECC Root CA - C3
    trust: distrusted
    category: authority

--
    label: emSign ECC Root CA - G3
    trust: distrusted
    category: authority

--
    label: emSign Root CA - C1
    trust: distrusted
    category: authority

--
    label: emSign Root CA - G1
    trust: anchor
    category: authority

Output will vary based on CentOS Stream version
```

```
###Verify CentOS Stream Version
[guestshell@guestshell ~]$ cat /etc/redhat-release
CentOS Stream release 8
```

OS	CentOS Stream 8	CentOS Stream 9
Directory	blacklist only	blocklist
Result shown	trust: blacklisted	trust: distrusted

步骤 4：状态持久性和容器生命周期

容器持久性注意事项：由于/etc/pki/ca-trust驻留在永久容器存储上，因此这些证书不信任修改在“guestshell enable”和“guestshell disable”操作之间持续存在，并且设备会重新启动。

返回到DEPLOYED状态（如果适用）：如果GuestShell最初处于DEPLOYED状态，并且仅被启用以执行此不信任过程，请通过从IOS特权EXEC提示符退出bash并运行“guestshell disable”来将GuestShell返回到其先前的停止状态：

<#root>

```
[root@guestshell ~]#
```

exit

```
[guestshell@guestshell ~]$
```

exit

```
Router#
```

guestshell disable

重要的生命周期注意事项：如果GuestShell被“guestshell destroy”完全销毁，则删除容器文件系统。如果随后重新启用GuestShell，则从基本系统映像重新创建容器，并且必须重新应用这些不信任的修复步骤。

步骤 5：回滚过程（恢复CA信任）

要撤消不信任配置并恢复所有Cisco IOS XE版本中的默认CA信任，请动态选择active blocklist目录($d)，特别删除通过此过程添加的四个证书文件（包括Comodo AAA Services根），并重建信任存储：

<#root>

```
! Escalate to root user inside GuestShell:
```

```
[guestshell@guestshell ~]$
```

```
sudo -s
```

```
d=/etc/pki/ca-trust/source/blocklist
[ -d "$d" ] || d=/etc/pki/ca-trust/source/blacklist

! Remove only the four certificate files added during remediation:
rm -f "$d/emSign_ECC_Root_CA_-_G3.pem" "$d/Comodo_AAA_Services_root.pem" \
      "$d/emSign_ECC_Root_CA_-_C3.pem" "$d/emSign_Root_CA_-_C1.pem"

! Rebuild the system CA trust store:
update-ca-trust extract
```

验证阻止列表目录不再包含删除的文件：

<#root>

```
[root@guestshell ~]#
```

```
ls -l "$d"/
```

```
total 0
```

确认CA信任恢复：通过重新运行第3步验证命令(所有条目都报告信任：锚):

<#root>

```
[root@guestshell ~]#
```

```
trust list --filter=trust-policy --purpose=server-auth | grep -E -A3 'emSign|AAA'
```

```
    trust: anchor
    trust: anchor
    trust: anchor
    trust: anchor
    trust: anchor
```

端到端执行脚本示例

该脚本说明在Cisco IOS XE设备上进行的完整、成功的GuestShell审核、补救和验证会话：

<#root>

```
Router#
```

```
guestshell run bash

[guestshell@guestshell ~]$ sudo -s
bash-4.4#

d=/etc/pki/ca-trust/source/blocklist

bash-4.4#

[ -d "$d" ] || d=/etc/pki/ca-trust/source/blacklist

bash-4.4#

mkdir -p "$d"

bash-4.4#

trust extract --overwrite --format=pem-bundle \ >   --filter='pkcs11:object=emSign%20ECC%20Root%20CA%20-

bash-4.4#

trust extract --overwrite --format=pem-bundle \ >   --filter='pkcs11:object=Comodo%20AAA%20Services%20ro

bash-4.4#

trust extract --overwrite --format=pem-bundle \ >   --filter='pkcs11:object=emSign%20ECC%20Root%20CA%20-

bash-4.4#

trust extract --overwrite --format=pem-bundle \ >   --filter='pkcs11:object=emSign%20Root%20CA%20-%20C1;

bash-4.4#

update-ca-trust extract

bash-4.4#

trust list --filter=trust-policy --purpose=server-auth | grep -E -A3 'emSign|AAA'

    label: emSign ECC Root CA - G3
    trust: blacklisted
    category: authority

--
    label: AAA Certificate Services
    trust: blacklisted
    category: authority

--
    label: emSign ECC Root CA - C3
    trust: blacklisted
    category: authority
```

```
--
    label: emSign Root CA - C1
    trust: blacklisted
    category: authority

--
    label: emSign Root CA - G1
    trust: anchor
    category: authority

bash-4.4#

cat /etc/redhat-release

CentOS Stream release 8
bash-4.4#

d=/etc/pki/ca-trust/source/blocklist

bash-4.4#

[ -d "$d" ] || d=/etc/pki/ca-trust/source/blacklist

bash-4.4#

rm -f "$d/emSign_ECC_Root_CA_-_G3.pem" "$d/Comodo_AAA_Services_root.pem" \ >       "$d/emSign_ECC_Root_C

bash-4.4#
bash-4.4#

update-ca-trust extract

bash-4.4#

ls -l "$d"/

total 0
```

命令摘要表

运营任务	
审核GuestShell状态	show app-hosting list
启用GuestShell（如	guestshell enable

果已部署）	
输入GuestShell Bash	`guestshell run bash`
切换到根用户	`sudo -s`
准备阻止列表目录	`d=/etc/pki/ca-trust/source/blocklist; [ -d "$d" ] || d=/etc/pki/ca-trust/source/blackli`
将CA提取到阻止列表	`trust extract --overwrite --format=pem-bundle ... (x4 CAs)`
重建CA信任存储	`update-ca-trust extract`
验证不信任策略	`trust list --filter=trust-policy --purpose=server-auth`
恢复已部署状态（可选）	`guestshell disable`
回滚/恢复信任	`rm -f "$d/emSign_ECC_Root_CA_-_G3.pem" "$d/Comodo_AAA_Services_root.pem" "$d/emSign_ECC_`

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）。