

# 目录

## [简介](#)

## [先决条件](#)

## [要求](#)

## [使用的组件](#)

## [背景信息](#)

## [配置](#)

[步骤1:请使用公共CA或设置CA在Windows服务器2003年](#)

[步骤2.验证主机名和设置](#)

[步骤3.生成并且下载证书签名请求\(CSR\)](#)

[步骤4.签署CSR用Microsoft Windows 2003年认证机关](#)

[步骤5.从CA获得根证明](#)

[步骤6.加载CA根证明作为CallManager托拉斯](#)

[步骤7.上传CA符号CallManager CSR证书作为CallManager证书。](#)

[步骤8.创建SIP中继安全配置文件](#)

[步骤9.创建SIP中继](#)

[步骤10.创建路由模式](#)

## [验证](#)

## [故障排除](#)

[收集CUCM的数据包捕获](#)

[收集的CUCM跟踪](#)

[相关的思科支持社区讨论](#)

# 简介

本文描述一逐步进程配置在通信管理器的会话初始化协议(SIP)传输层安全(TLS)中继有

使用TLS，在跟随本文以后，SIP消息在两集群之间将加密。

## [先决条件](#)

## [要求](#)

思科建议您有知识：

- Cisco Unified Communications Manager (CUCM)
- SIP

## [使用的组件](#)

本文档中的信息基于以下软件版本：

- CUCM版本9.1(2)
- CUCM版本10.5(2)
- Microsoft Windows服务器2003作为CA

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。如果您使用的是真实网络，请确保您已经了解所有命令的潜在影响。

## 背景信息

如此镜像所显示，SSL握手使用证书。

### 配置

步骤1:请使用公共CA或设置CA在Windows服务器2003年

参考链路：[在Windows 2003塞弗的设置CA](#)

步骤2.验证主机名和设置

证书根据名称。保证名称在开始前是正确。

为了更改主机名，参考链路：[在CUCM的崔凡吉莱主机名](#)

步骤3.生成并且下载证书签名请求(CSR)

### CUCM 9.1(2)

为了生成CSR，请导航对OS Admin > Security > Certificate Management >生成CSR

在验证名称字段，请选择从下拉列表的CallManager选项。

为了下载CSR，请导航对OS Admin > Security > Certificate Management >下载CSR

在验证名称字段，请选择从下拉列表的CallManager选项。

### CUCM 10.5(2)

为了生成CSR，请导航对OS Admin > Security > Certificate Management >

1. 在证书目的字段，请选择从下拉列表的CallManager。

2. 在密钥长度字段，请选择1024从下拉列表。

3. 在散列算法字段，请选择从下拉列表的SHA1。

为了下载CSR，请导航对OS Admin > Security > Certificate Management >下载CSR

在证书目的字段，请选择从下拉列表的CallManager选项。

**注意：**CallManager CSR生成与1024位Rivest沙米尔Addleman (RSA)密钥。

步骤4.签署CSR用Microsoft Windows 2003年认证机关

这是签署与Microsoft Windows 2003 CA的CSR的可选信息。

1. 打开证书颁发机构。
2. 用鼠标右键单击CA图标并且导航对**所有任务>提交新要求**
3. 选择CSR并且点击**Open选项**(可适用在两CSR (CUCM 9.1(2)和CUCM 10.5(2))
4. 所有在待定请求文件夹的打开的CSR显示。用鼠标右键单击每个CSR并且导航对**所有任务>问题**为了发行证书。(可适用在两CSR (CUCM 9.1(2)和CUCM 10.5(2))
5. 为了下载证书，请选择**已签发证书**文件夹。  
用鼠标右键单击证书并且点击**Open选项**。
6. 证书详细信息显示。为了下载证书，选择**详细信息选项卡**并且单击Copy按钮到文件？
7. 在**证书Export**向导窗口，请点击**Base-64编码X.509(.CER)**单选按钮。
8. 准确地名叫文件。此示例使用**CUCM1052.cer**格式。

对于CUCM 9.1(2)，请遵从同样步骤。

#### 步骤5.从CA获得根证明

打开**证书颁发机构**窗口。

为了下载根CA

1. 用鼠标右键单击CA图标并且点击**属性选项**。
2. 在常规选项卡，请点击**查看证书**。
3. 在**证书**窗口，请点击**详细信息选项卡**。
4. 单击**“Copy”到文件...**

#### 步骤6.加载CA根证明作为CallManager托拉斯

为了上传CA根证明，登录到**OS Admin > Security > Certificate Management > 加载证书/证书链**

**注意：**执行在两个的这些步骤CUCMs (CUCM 9.1(2)和CUCM 10.5(2))

#### 步骤7.上传CA符号CallManager CSR证书作为CallManager证书。

为了上传CA符号CallManager CSR，请登陆到**OS Admin > Security > Certificate Management > 加载证书/证书链**

**注意：**执行在两个的这些步骤CUCMs (CUCM 9.1(2)和CUCM 10.5(2))

#### 步骤8.创建SIP中继安全配置文件

**CUCM 9.1(2)**

为了创建SIP中继安全配置文件，请导航对**系统 > Security > SIP中继安全配置文件**。

复制存在非安全SIP中继配置文件并且给予它新名字。在示例中，非安全SIP中继配置文件重命名与安全SIP中继配置文件TLS。

在**X.509主题名称**请使用CUCM 10.5(2) (CA签名证书)的共同名称(CN)如此镜像所显示。

## CUCM 10.5(2)

导航对**系统 > Security > SIP中继安全配置文件**。

复制存在非安全SIP中继配置文件并且给予它新名字。在示例中，非安全SIP中继配置文件重命名与安全SIP中继配置文件TLS。

在**X.509主题名称**请使用CUCM 9.1(2) (CA签名证书)的CN如突出显示：

两个SIP中继安全配置文件设置传入端口5061，每集群在新的入站SIP TLS呼叫的TCP端口5061侦听。

### 步骤9.创建SIP中继

在安全配置文件创建后，请创建SIP中继并且做下面的配置参数的变动在SIP建立中继。

## CUCM 9.1(2)

1. 在**SIP中继配置**窗口，请检查**允许的**配置参数**SRTP**复选框。

这获取将用于在此中继的呼叫实时传输协议(RTP)。必须只检查此方框，当您使用SIP TLS时，因为Secure实时传输协议的(SRTP)密钥在SIP消息的正文被交换。必须由TLS获取SIP信令，否则任何人与不安全的SIP信令可能解密在中继的对应的SRTP数据流。

1. 在**SIP中继配置**窗口的**SIP信息**部分，请添加**目的地址**、**目的地端口**和**SIP中继安全配置文件**。

## CUCM 10.5(2)

1. 在**SIP中继配置**窗口，请检查**允许的**配置参数**SRTP**复选框。

这允许将用于在此中继的呼叫SRTP。必须只检查此方框，当曾经SIP TLS时，因为SRTP的密钥在SIP消息的正文被交换。因为任何人与一不安全的SIP信令可能解密在中继的对应的安全RTP数据流必须由TLS获取SIP信令。

1. 在**SIP中继配置**窗口的**SIP信息**部分，请添加**目的IP地址**、**目的地端口**和**安全配置文件**

### 步骤10.创建路由模式

简单方法将创建在每集群的一个路由模式，指向直接地到SIP中继。可能也使用路由组和路由列表。

对**路由模式**9898的CUCM 9.1(2)点通过对CUCM的TLS SIP中继10.5(2)

对**路由模式**1018的CUCM 10.5(2)点通过对CUCM的TLS SIP中继9.1(2)

## [验证](#)

当前没有可用于此配置的验证过程。

## 故障排除

SIP TLS呼叫可以调试与这些步骤。

### CUCM的收集的数据包捕获

为了检查CUCM 9.1(2)和CUCM 10.5(2)之间的连接，请采取CUCM服务器的一数据包捕获并且注意SIP TLS流量。

SIP TLS流量在TCP端口5061传送，被看到作为SIP tls。

在以下示例中有SSH CLI会话建立对CUCM 9.1(2)

#### 1. 屏幕的CLI数据包捕获

此CLI打印在屏幕的输出SIP TLS流量的。

```
admin:utils network capture host ip 10.106.95.200Executing command with
options:interface=eth0ip=10.106.95.20019:04:13.410944 IP CUCMA.42387 > 10.106.95.200.sip-tls: P
790302485:790303631(1146) ack 3661485150 win 182 <nop,nop,timestamp 2864697196
5629758>19:04:13.450507 IP 10.106.95.200.sip-tls > CUCMA.42387: . ack 1146 win 249
<nop,nop,timestamp 6072188 2864697196>19:04:13.465388 IP 10.106.95.200.sip-tls > CUCMA.42387: P
1:427(426) ack 1146 win 249 <nop,nop,timestamp 6072201 2864697196>
```

#### 2. 对文件的CLI捕获

此CLI执行根据主机的数据包捕获并且创建名为数据包的文件。

```
admin:utils network capture eth0 file packets count 100000 size all host ip 10.106.95.200
重新启动在CUCM 9.1(2)的SIP中继并且由分机1018 (CUCM 9.1(2))做呼叫对分机9898 (CUCM
10.5(2))
```

为了下载从CLI的文件，请运行此命令：

```
admin:file get activelog platform/cli/packets.cap
```

捕获在标准的.cap格式完成。此示例使用Wireshark打开packets.cap文件，但是可以使用所有数据包捕获显示工具。

1. 传输控制协议(TCP)同步(SYN)建立CUCM 9.1(2)(Client)和CUCM 10.5(2)(Server)之间的TCP通信。
2. CUCM 9.1(2)发送客户端Hello启动TLS会话。
3. CUCM 10.5(2)发送服务器问候、服务器证书和证书请求开始证书交换过程。
4. 客户端CUCM 9.1(2)发送完成证书交换的证书。
5. 是加密的SIP信令的应用程序数据，显示TLS会话建立了。

促进检查正确证书是否交换。在服务器问候以后，服务器CUCM 10.5(2)发送其证书对客户端CUCM 9.1(2)。

序列号和服务器CUCM 10.5(2)有的附属的信息，被提交到客户端CUCM 9.1(2)。The序列号，主题，发布者，并且有效日期全部与关于OS Admin证书管理页的信息比较。

服务器CUCM 10.5(2)提交其验证的自己的证书，它当前检查客户端CUCM 9.1(2)的证书。验证在两

个方向发生。

如果有证书在数据包捕获和证书之间的一不匹配在OS Admin网页，则正确证书没有上传。

必须上传正确证书在OS Admin Cert页上。

#### 收集的CUCM跟踪

CUCM跟踪可以也是有用确定什么消息被交换在CUCM 9.1(2)和CUCM 10.5(2)服务器之间，并且SSL会话是否适当地建立。

在示例中，从CUCM 9.1(2)的跟踪收集。

呼叫流：

Ext 1018 > CUCM 9.1(2) > SIP TLS中继> CUCM 10.5(2) > Ext 9898

#### ++数字分析

```
04530161.009 |19:59:21.185 |AppInfo |Digit analysis: match(pi="2", fqcN="1018",
cn="1018",plv="5", pss="", TodFilteredPss="", dd="9898",dac="0")04530161.010 |19:59:21.185
|AppInfo |Digit analysis: analysis results04530161.011 |19:59:21.185 |AppInfo
||PretransformCallingPartyNumber=1018|CallingPartyNumber=1018|DialingPartition=|DialingPattern=9
898|FullyQualifiedCalledPartyNumber=9898
```

#### ++ SIP TLS在端口5061使用此呼叫。

```
04530191.034 |19:59:21.189 |AppInfo |//SIP/SIPHandler/ccbId=0/scbId=0/SIP_PROCESS_ENQUEUE:
createConnMsg tls_security=304530204.002 |19:59:21.224 |AppInfo
|//SIP/Stack/Transport/0x0/sipConnectionManagerProcessConnCreated: gConnTab=0xb444c150,
addr=10.106.95.200, port=5061, connid=12, transport=TLS Over TCP04530208.001 |19:59:21.224
|AppInfo |SIPtcp - wait_SdlSPISignal: Outgoing SIP TCP message to 10.106.95.200 on port 5061
index 12[131,NET]INVITE sip:9898@10.106.95.200:5061 SIP/2.0Via: SIP/2.0/TLS
10.106.95.203:5061;branch=z9hG4bK144f49a43aFrom: <sip:1018@10.106.95.203>;tag=34~4bd244e4-0988-
4929-9df2-2824063695f5-19024196To: <sip:9898@10.106.95.200>Call-ID: 94fffc00-57415541-7-
cb5f6a0a@10.106.95.203User-Agent: Cisco-CUCM9.1
```

#### ++信号分配层(SDL)消息SIPCertificateInd提供关于主题CN和连接信息的细节。

```
04530218.000 |19:59:21.323 |SdlSig |SIPCertificateInd |wait
|SIPHandler(1,100,72,1) |SIPtcp(1,100,64,1)
|1,100,17,11.3^^^* |[[T:N-H:0,N:1,L:0,V:0,Z:0,D:0] connIdx= 12 --
remoteIP=10.106.95.200 --remotePort = 5061 --X509SubjectName
/C=IN/ST=cisco/L=cisco/O=cisco/OU=cisco/CN=CUCM10 --Cipher AES128-SHA --SubjectAltname
=04530219.000 |19:59:21.324 |SdlSig |SIPCertificateInd
|restart0 |SIPD(1,100,74,16)
|SIPHandler(1,100,72,1) |1,100,17,11.3^^^* |[R:N-
H:0,N:0,L:0,V:0,Z:0,D:0] connIdx= 12 --remoteIP=10.106.95.200 --remotePort = 5061 --
X509SubjectName /C=IN/ST=cisco/L=cisco/O=cisco/OU=cisco/CN=CUCM10 --Cipher AES128-SHA --
SubjectAltname =
```