在协作环境中设置RADKit

目录

简介

本文档介绍设置RADKit的步骤，并显示了开始将其用于协作产品所需的配置。

要求

思科建议您了解以下主题：

- 任何VOS协作产品的基本知识
- CLI/SSH访问基础知识

使用的组件

本文档中的信息基于以下软件和硬件版本：

- Cisco Unified Communications Manager 12.5和14.0

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。

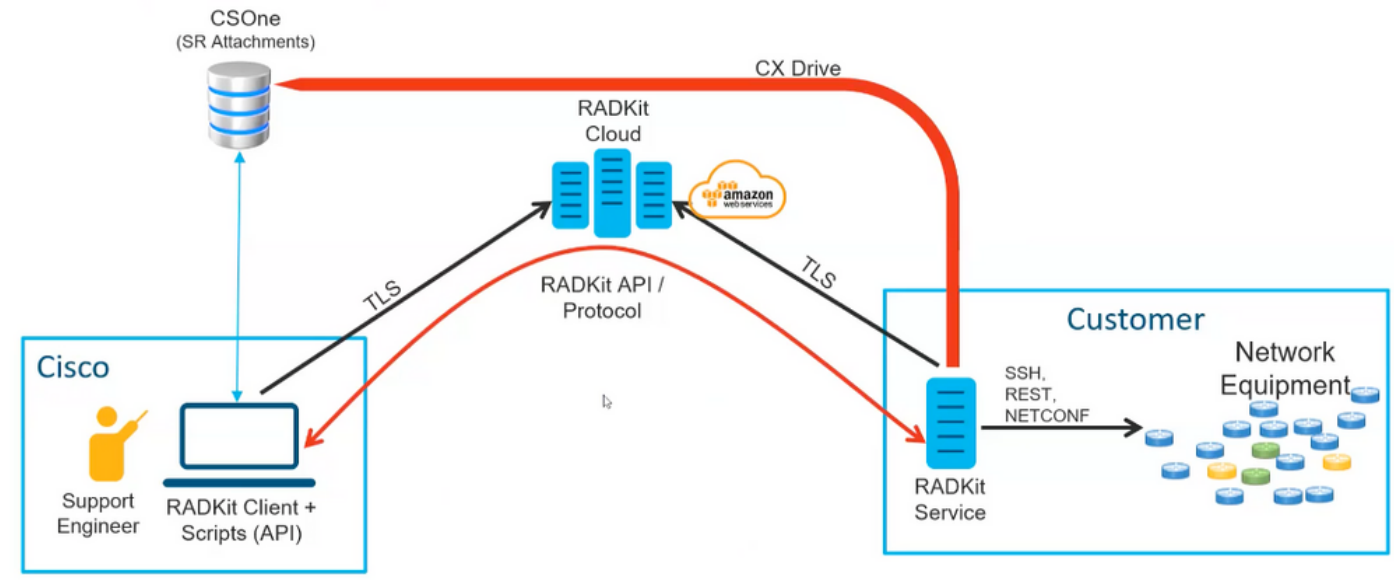
术语

RADKit:该连接器为Cisco TAC工程师和合作伙伴提供对用户设备的安全远程访问。它支持多种协议与设备交互，例如SSH或HTTP/HTTP。

RADKit服务:这是Server端。 它由用户处理和完全管理。从服务器端，用户控制谁可以访问设备以及访问时间。Radkit服务必须连接到网络中的设备才能提供对这些设备的访问。

RADKit客户端:这是Client端。它是用于连接到用户网络中设备的PC。

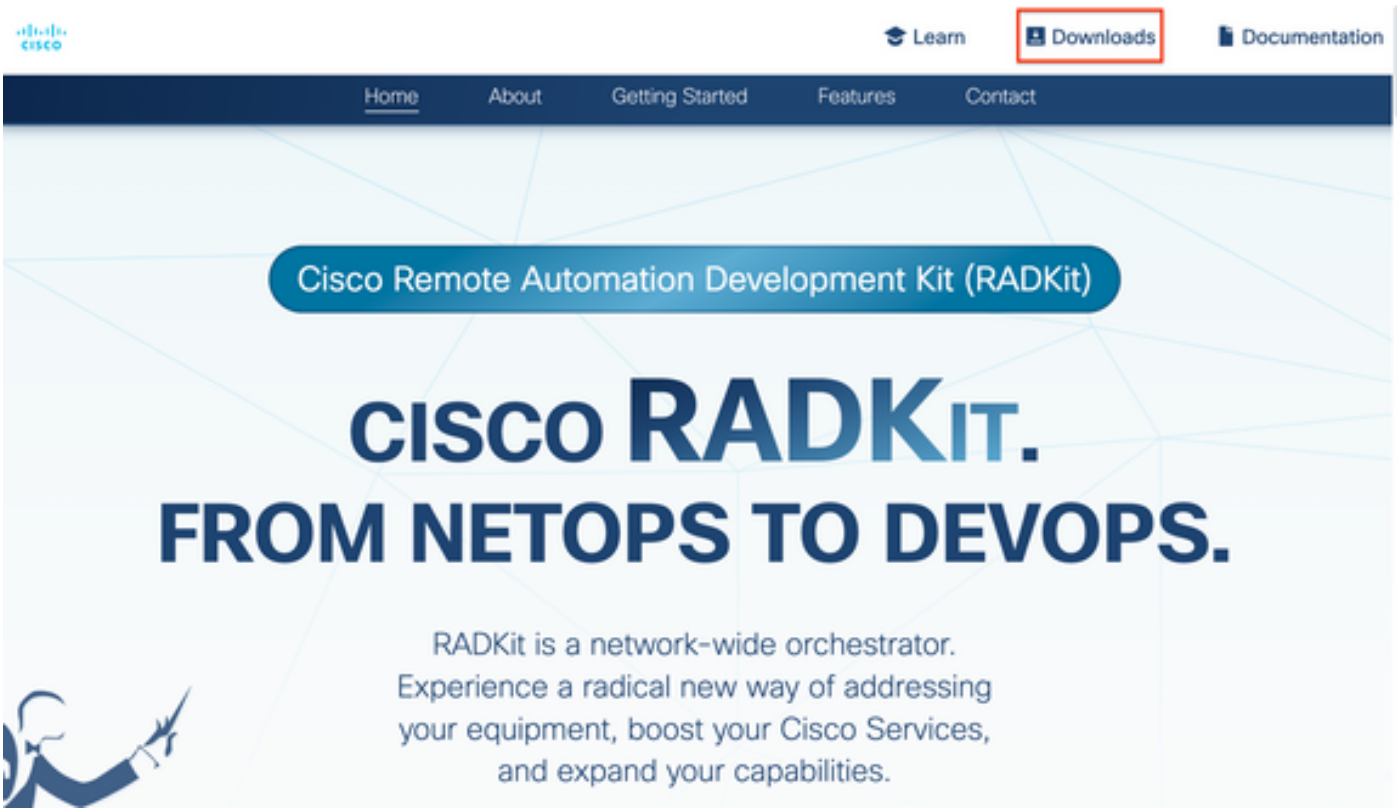
RADKit架构



RADKit架构

RADKit安装

步骤1.导航到https://radkit.cisco.com，然后单击Downloads，然后转到release文件夹。



步骤2.点击最新版本。

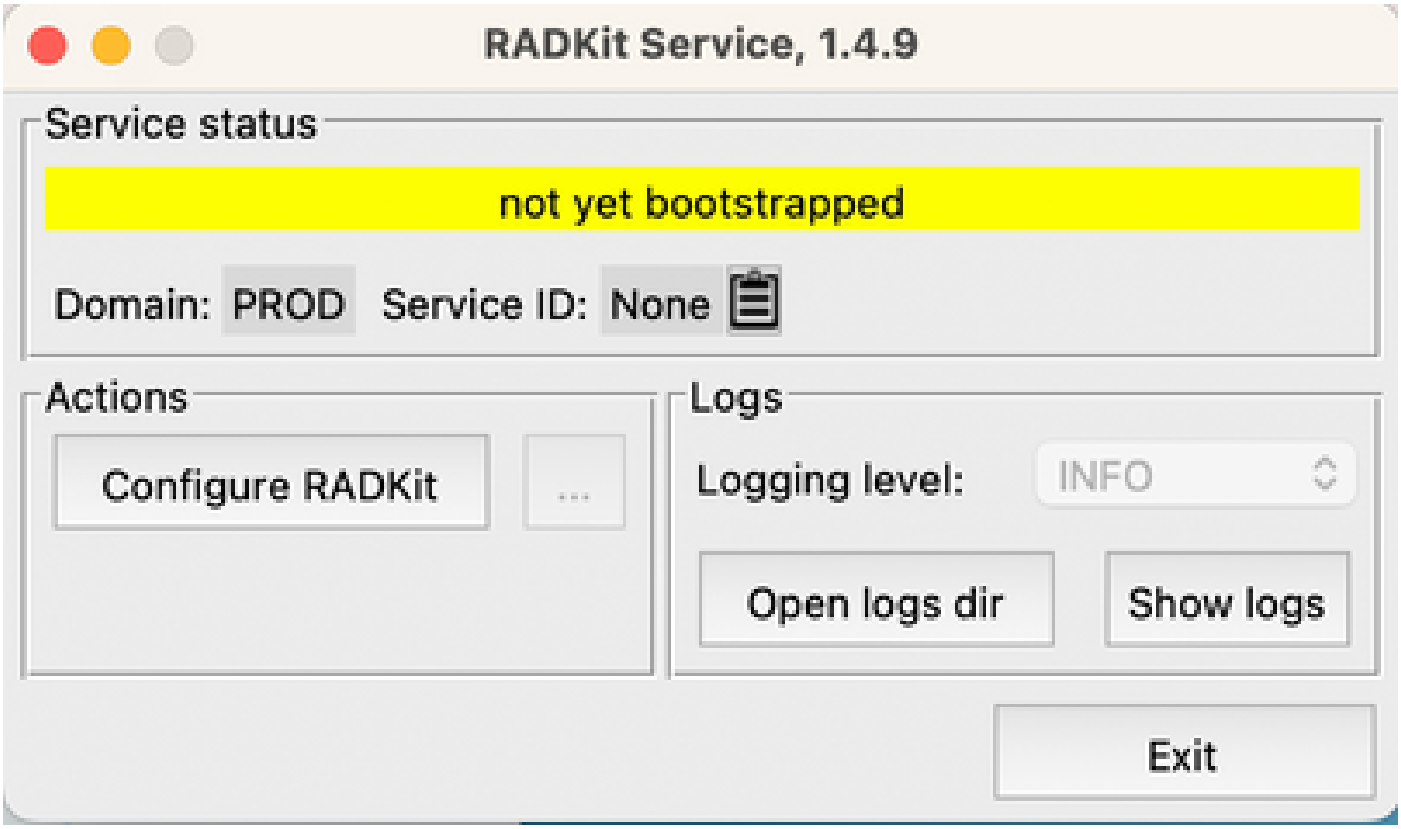
步骤3.根据您的操作系统下载正确的文件。

第 4 步： 在PC或服务器上运行安装程序。在安装过程中，Radkit需要安装三个应用程序：Radkit服务、Radkit客户端和Radkit网络控制台。

RADKit服务（用户端）

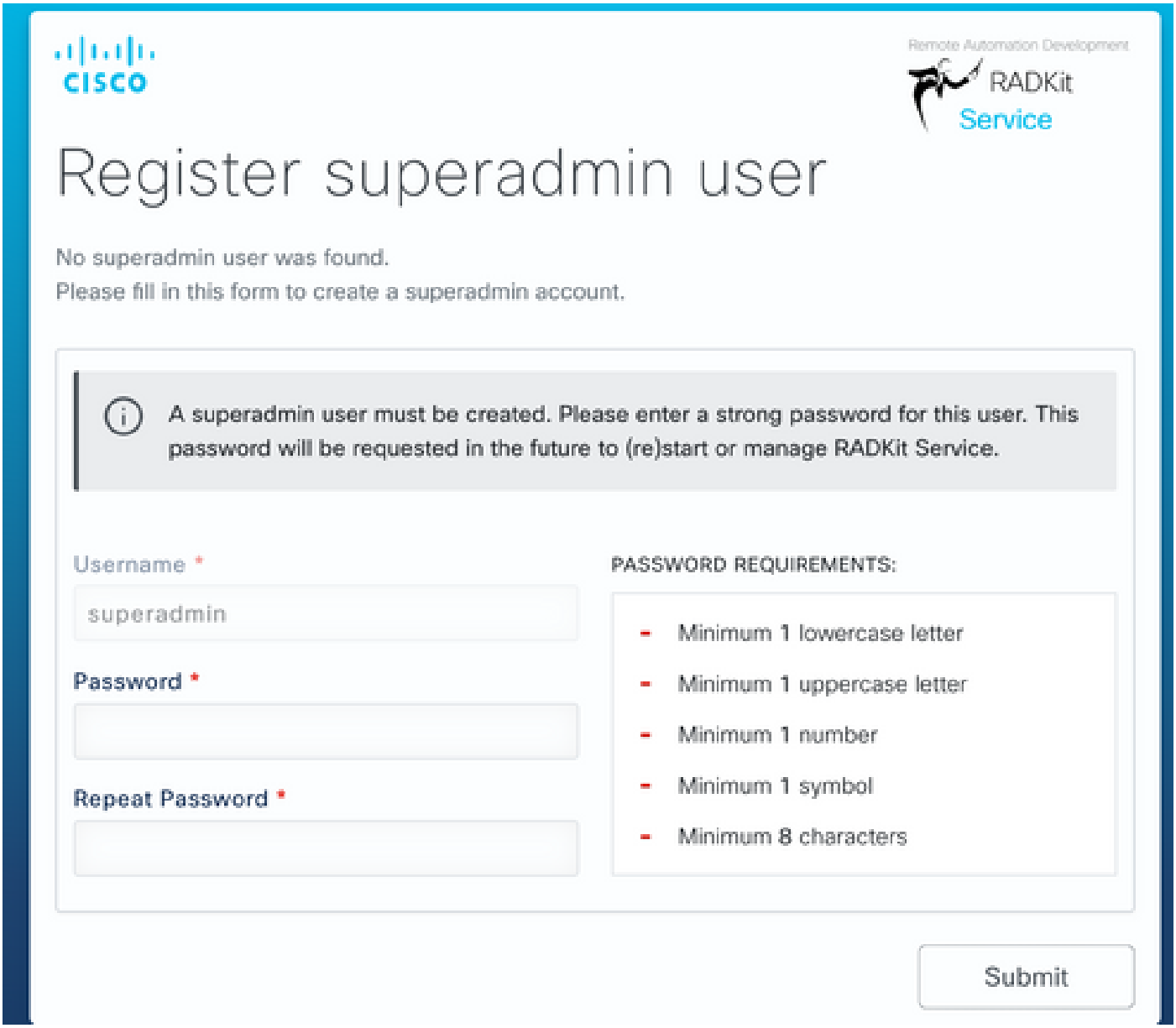
自注册

第1步：要开始配置RADKit服务，请导航到应用并找到RADKit服务。第一次运行它时会显示消息“not yet bootstrapped”。



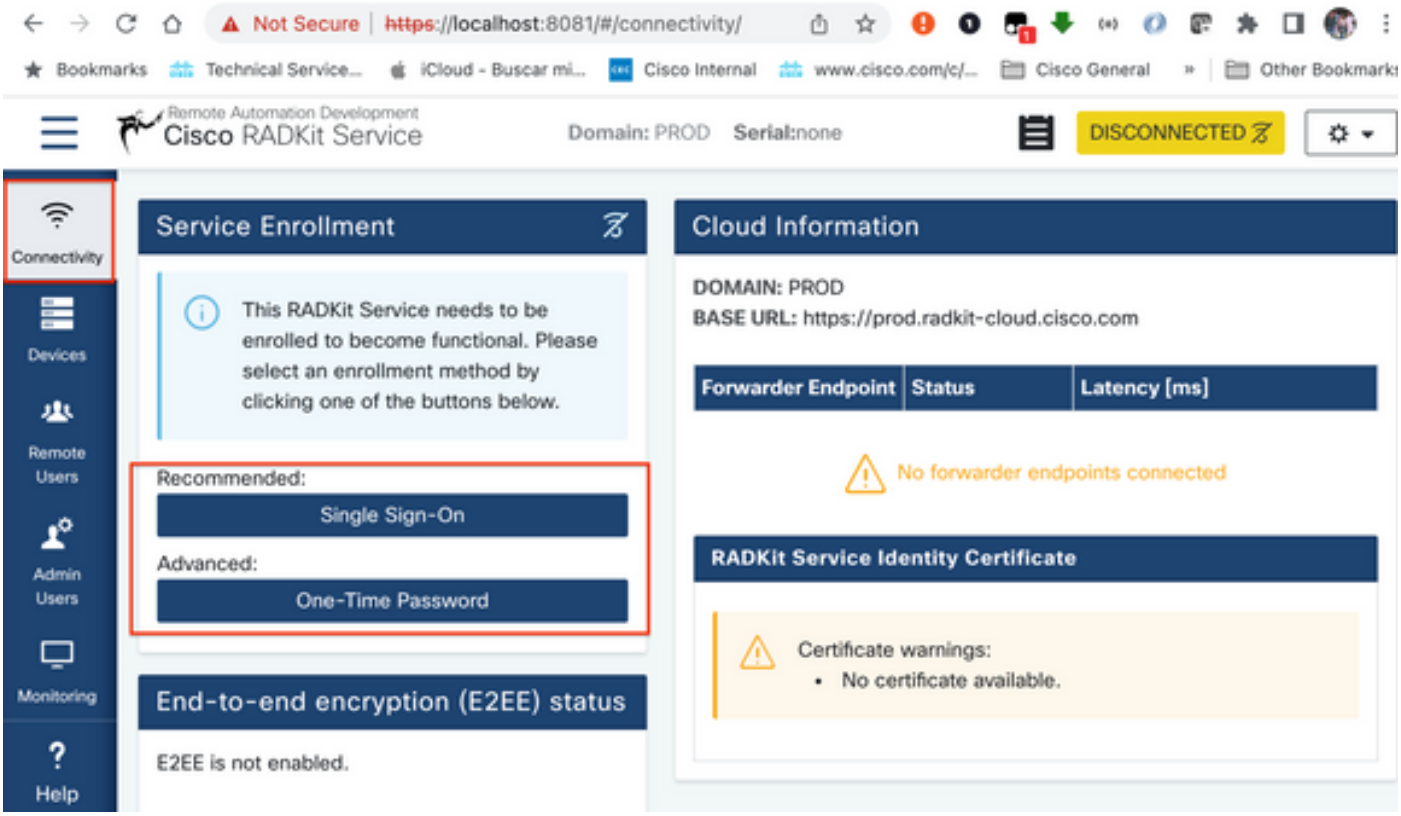
步骤2.单击配置RADKit，浏览器会自动弹出并显示URL https://localhost:8081/bootstrap。

- 为superadmin用户创建密码，然后单击Submit。
- 每次启动或配置服务时都会请求此superadmin用户名和密码。

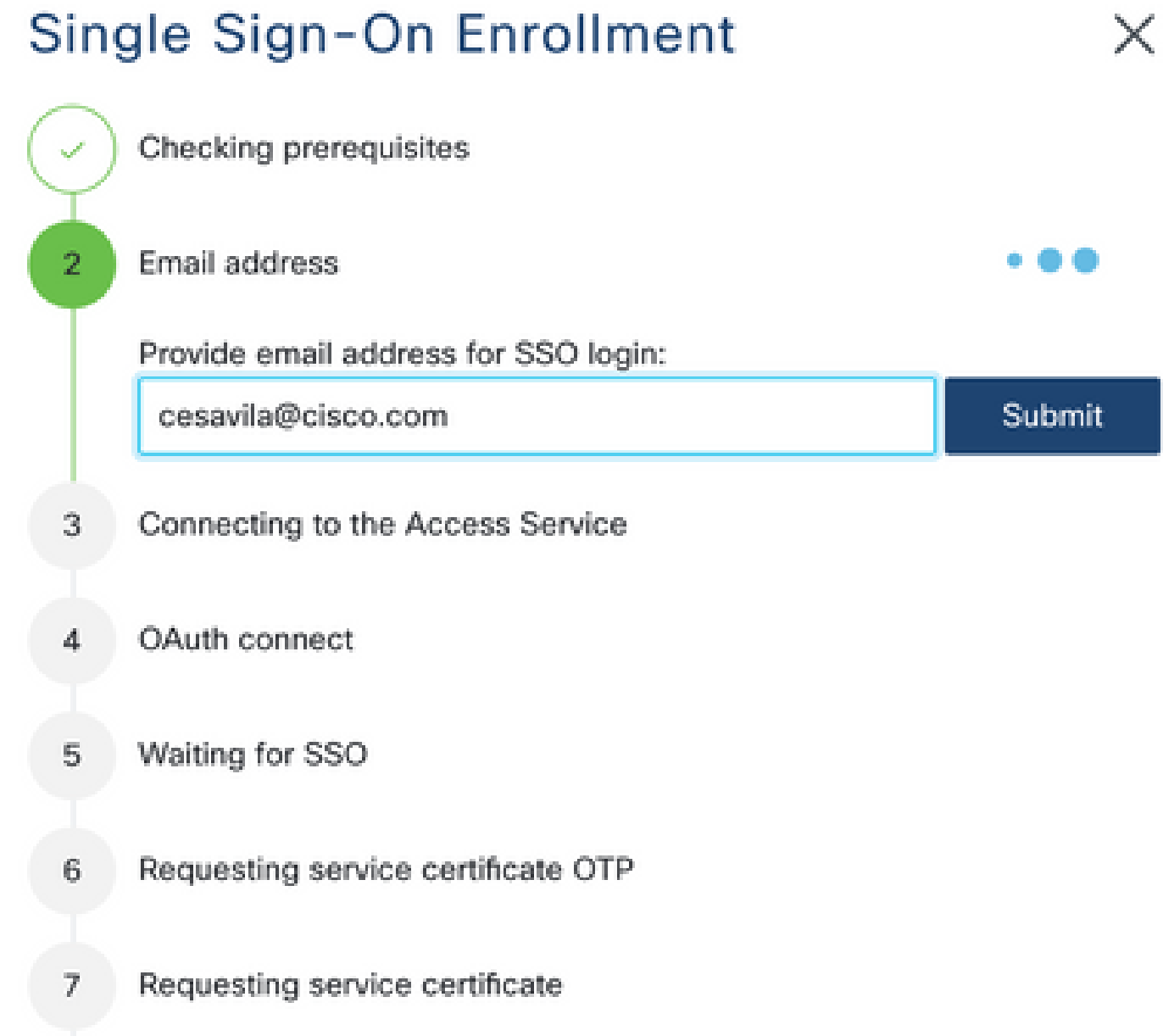


第3步：单击Submit后，浏览器将您重定向到https://localhost:8081/#/connectivity/。

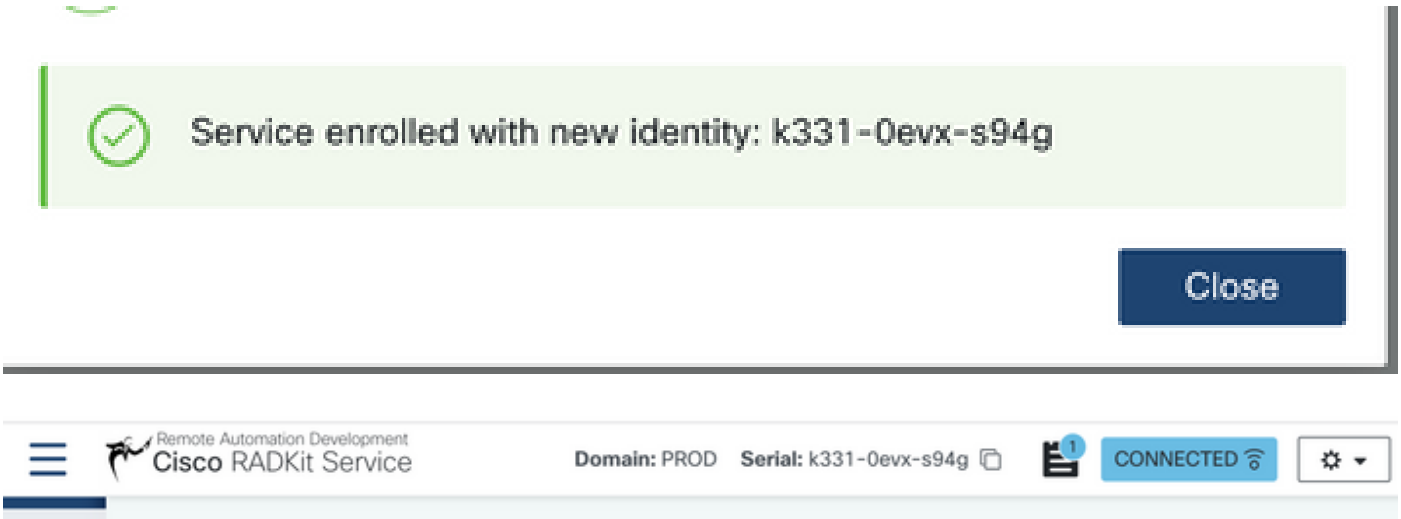
在Connectivity > Service Enrollment下，有两种身份验证方法：单一登录和一次性密码。

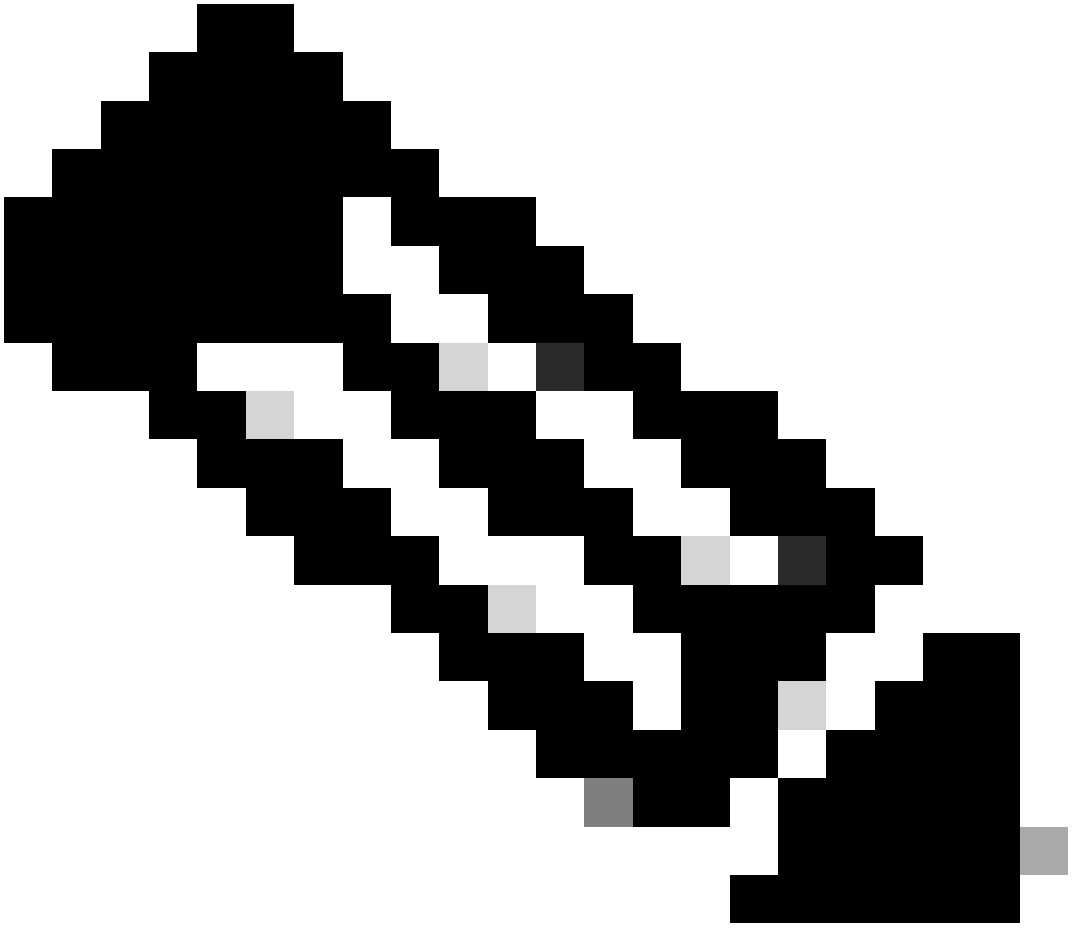


步骤4.您可以使用单点登录提供您的CEC。

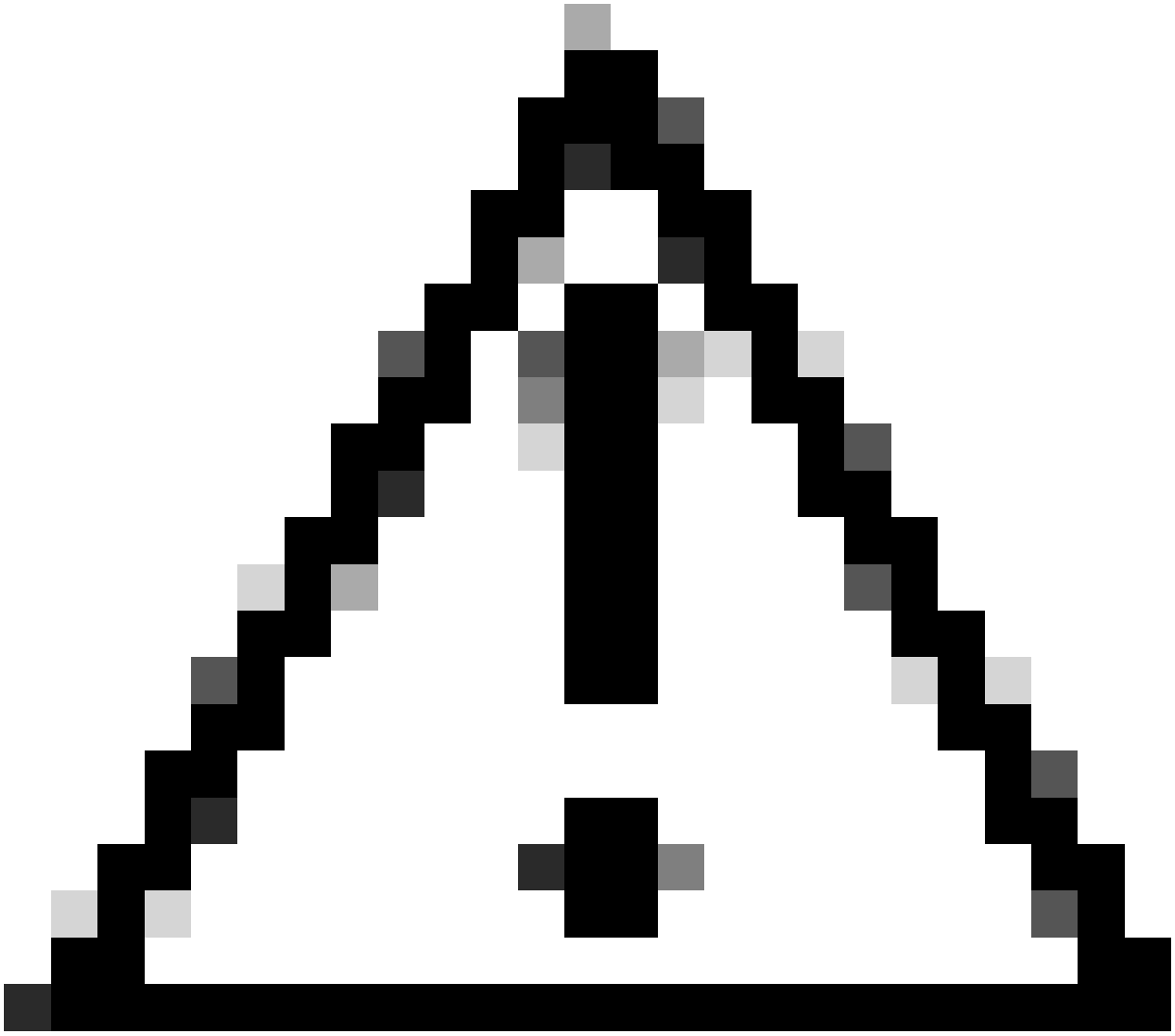


步骤5.完成向导并完成步骤，直到显示“使用新身份注册的服务：xxxx-xxxx-xxxx”，并在点击Close时，服务显示为Connected。





注意：激活RADKit服务需要思科帐户。

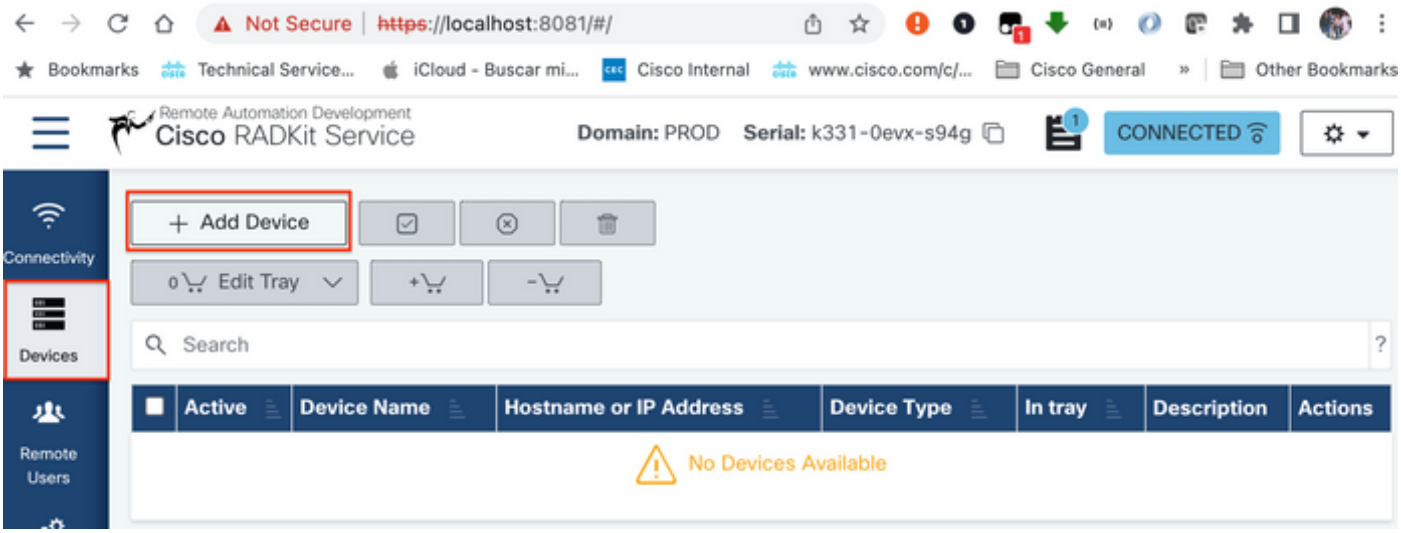


警告：

- 如果运行RADKit服务的服务器需要定义代理，除了在服务器/PC上定义代理外，还需要为RADKit服务定义环境变量以使RADKIT_CLOUD_CLIENT_PROXY_URL=http://proxy.example.com:80。

添加设备

第1步：导航到Devices，然后点击Add Device。

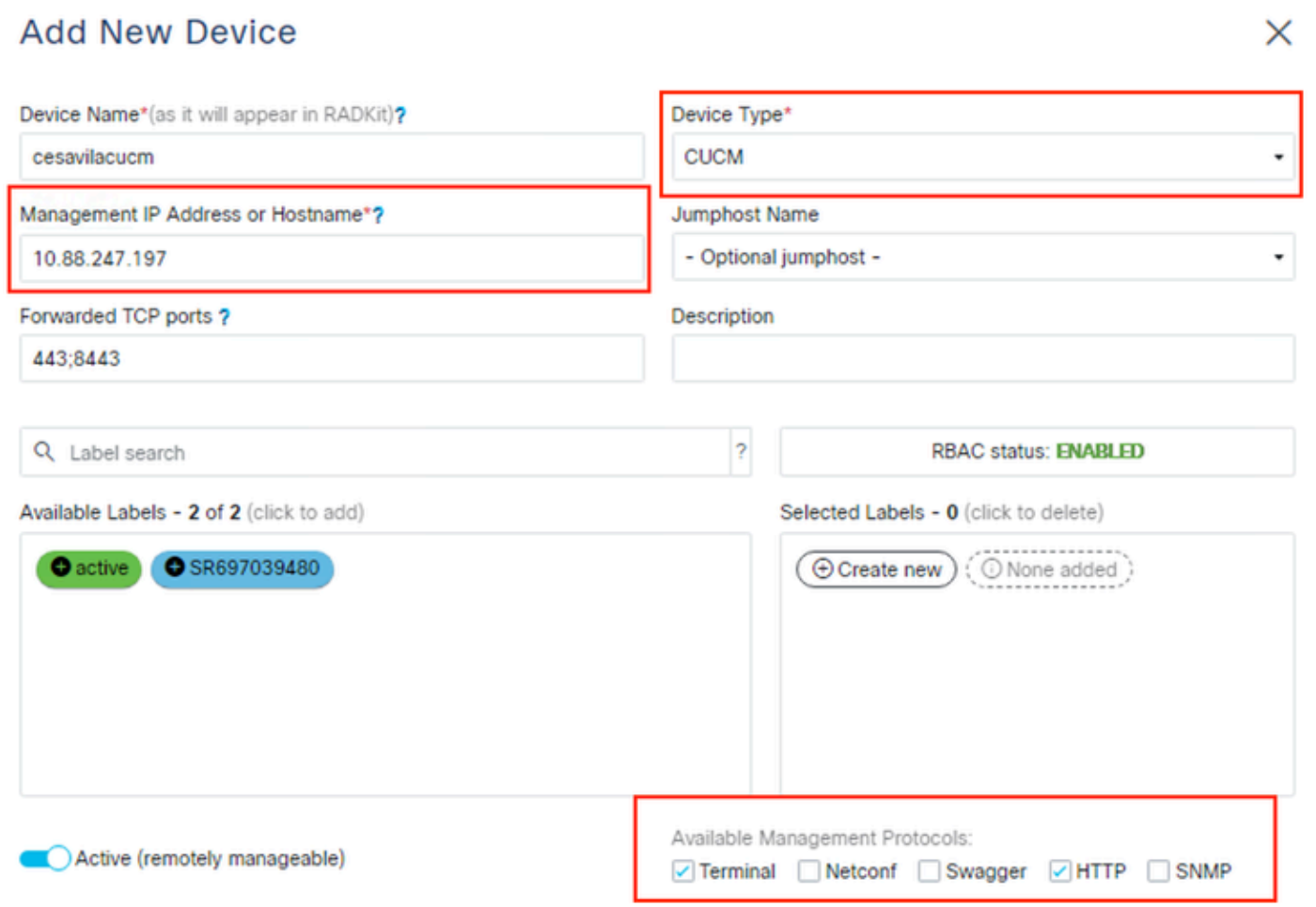


步骤2.您需要配置以下详细信息：

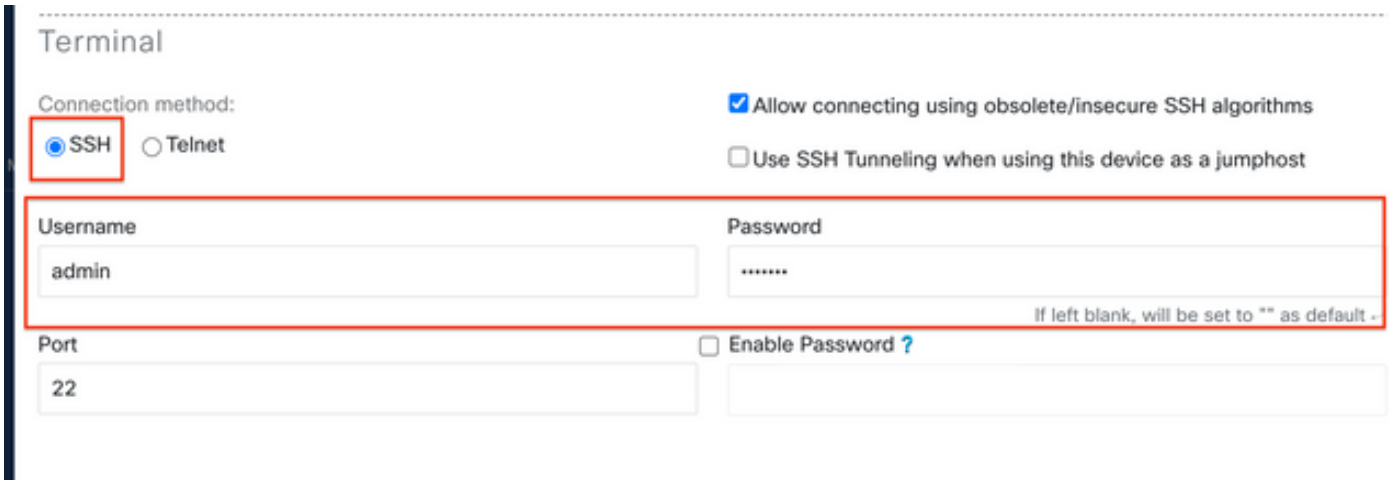
- 设备名
- 管理IP地址或主机名
- 设备类型

此外，还必须配置Forwarded TCP ports，这是设备使用的端口，需要从RADKit客户端对其进行访问。在本示例中，使用的端口是443用于GUI访问，8443用于RTMT。

最后，选择可用的管理协议，在本例中为Terminal和HTTP。



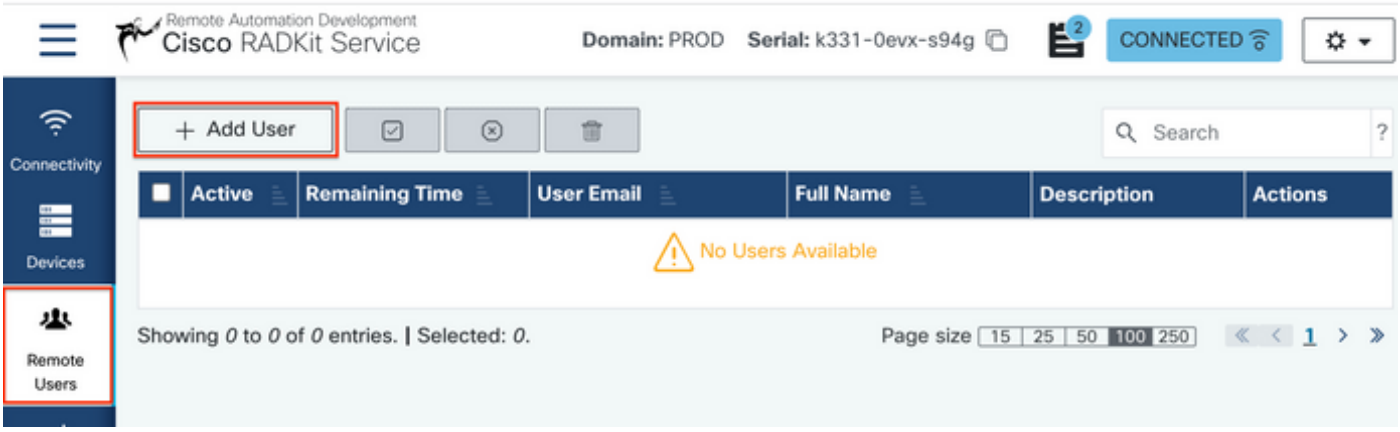
步骤3.对于每个管理协议，配置正确的设置，然后点击Add & Close。



步骤4.添加设备后，必须在设备列表中显示该设备，可以启用/禁用该设备以进行远程访问。

授权远程用户

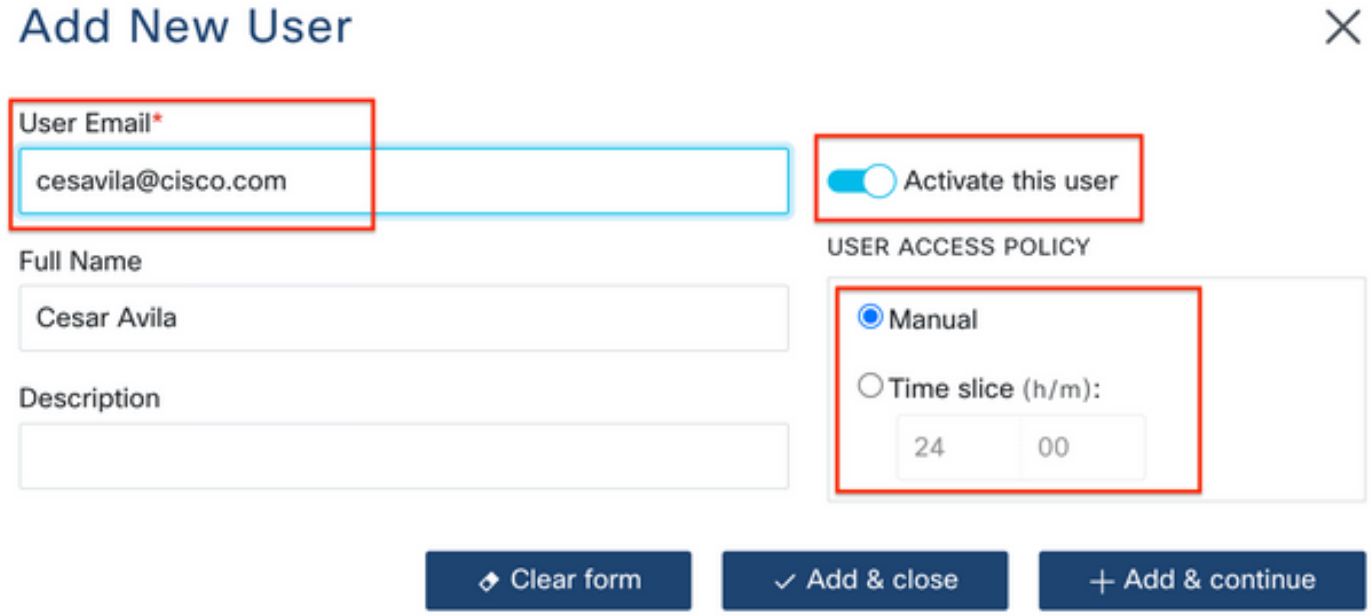
第1步：要授予用户对RADKit服务中配置的设备的访问权限，请转至Remote Users，然后选择Add Users。



步骤2.配置用户详细信息：

- 电子邮件地址

- 全名（可选）
- 激活用户。
- 指定必须手动控制激活还是设置时间范围以授予该用户的访问权限。



第3步：选择添加并关闭。

RADkit客户端（TAC端）

登录

步骤1.在客户端PC上，导航到应用并找到RADkit客户端。

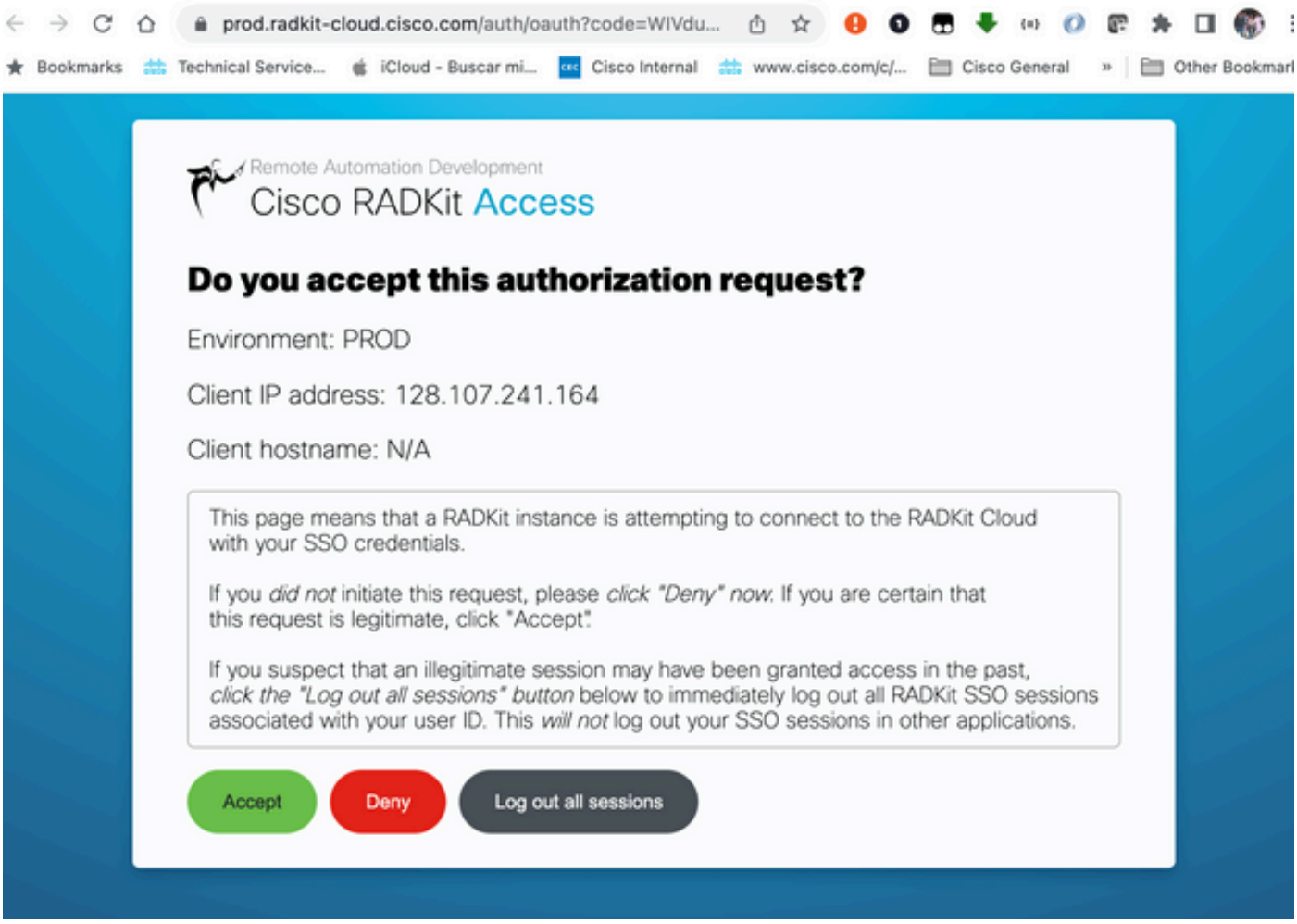
步骤2.使用SSO登录创建客户端实例。

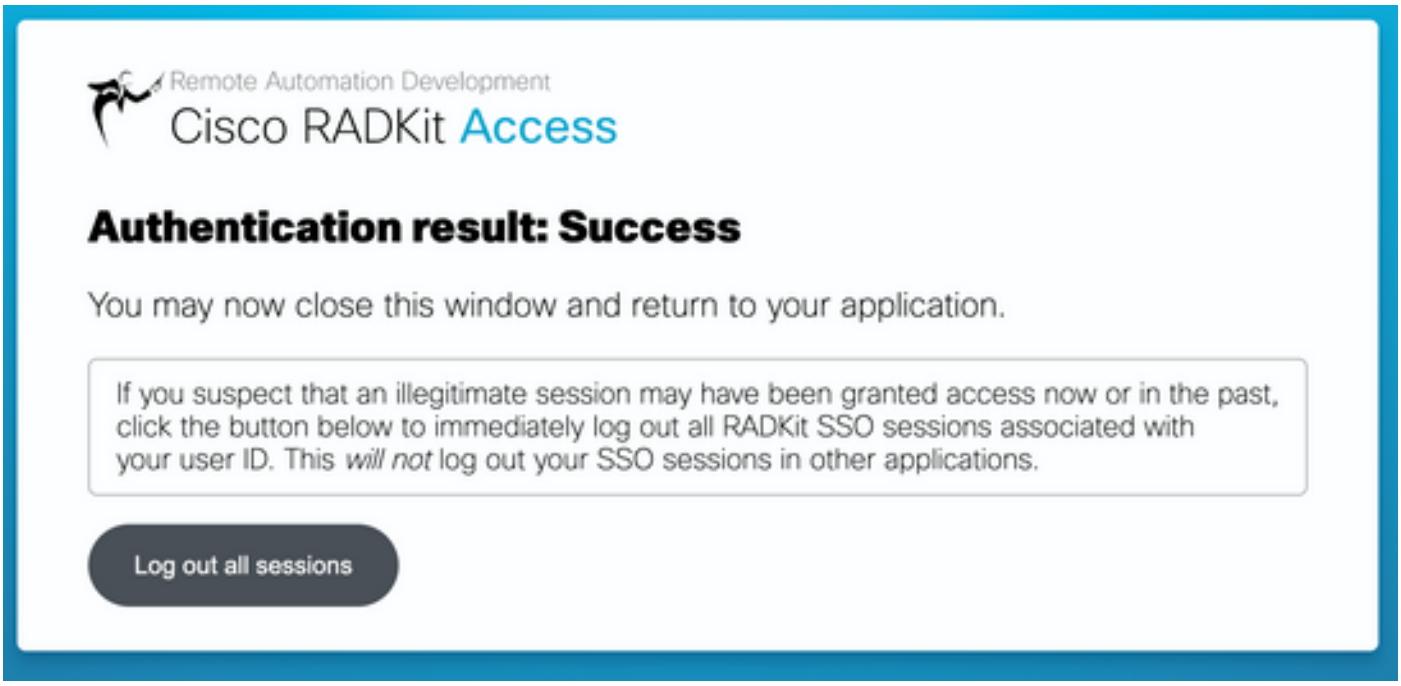
<#root>

>>>

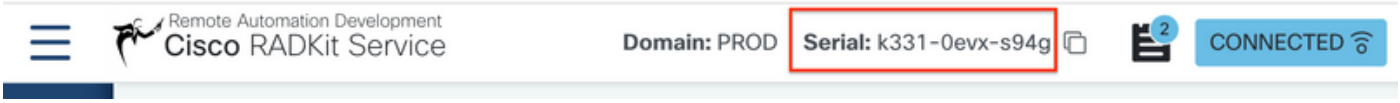
```
client = sso_login("cesavila@cisco.com")
```

步骤3.接受在浏览器上自动打开的SSO授权请求。





步骤4.使用用户从RADKit服务 — 自注册阶段生成的序列号创建服务实例。

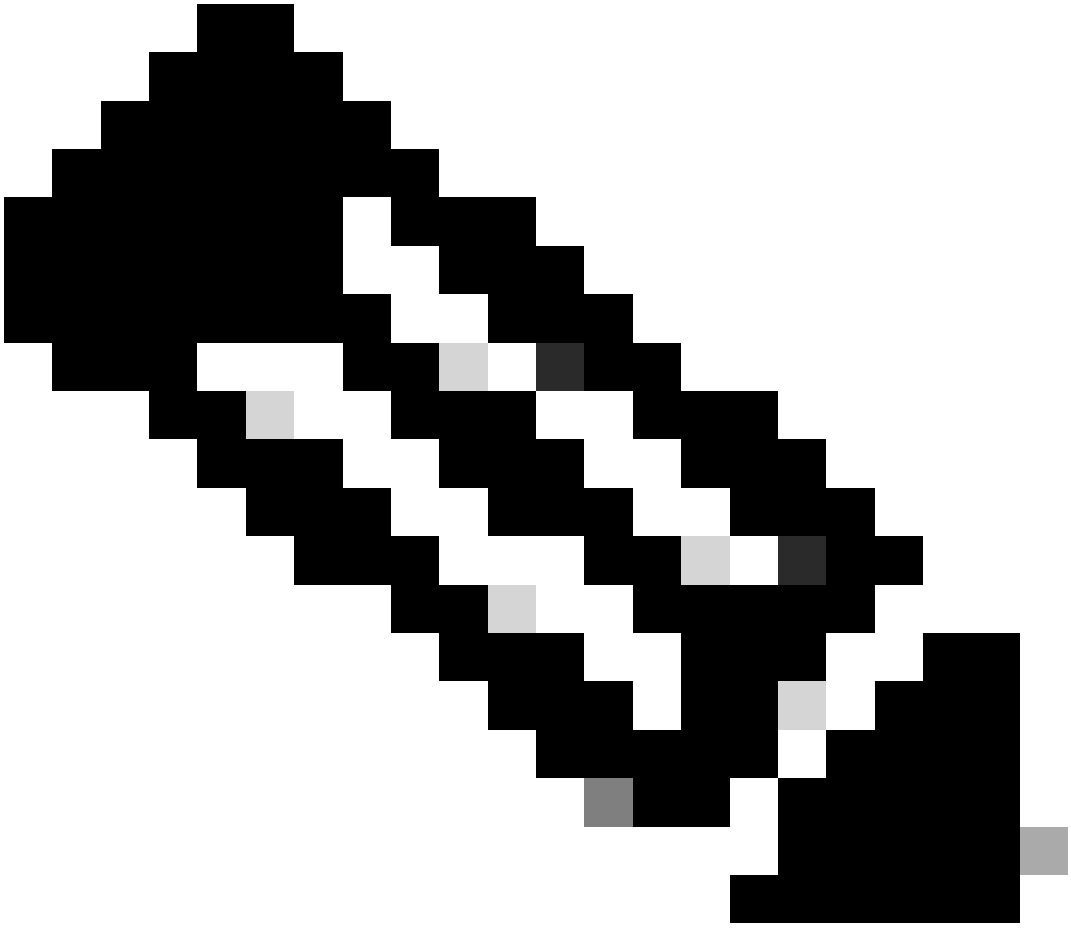


<#root>

>>>

```
service = client.service("k331-0evx-s94g")
```

```
>>> service = client.service("k331-0evx-s94g")
05:16:36.349Z INFO  | internal | Connecting to forwarder [uri='wss://prod.radkit-cloud.cisco.com/forwarder-2/websocket/']
05:16:37.153Z INFO  | internal | Connection to forwarder successful [uri='wss://prod.radkit-cloud.cisco.com/forwarder-2/websocket/']
05:16:39.523Z INFO  | internal | Connecting to forwarder [uri='wss://prod.radkit-cloud.cisco.com/forwarder-3/websocket/']
05:16:40.333Z INFO  | internal | Connection to forwarder successful [uri='wss://prod.radkit-cloud.cisco.com/forwarder-3/websocket/']
```



注意：service是一个变量，可以是任何变量。

步骤5.检查可供访问的设备。

<#root>

>>>

service.inventory

```
>>>
>>> service.inventory
<radkit_client.sync.device.DeviceDict object at 0x10d7728e0>
name         host          device_type    Terminal    Netconf    Swagger    HTTP    description    failed
-----------  ------------  -------------  ----------  ---------  ---------  ------  -------------  --------
cesavilacucm  10.88.247.197  UNKNOWN        True        False      False      True                   False
```

要刷新资产列表，请使用命令update_inventory。

<#root>

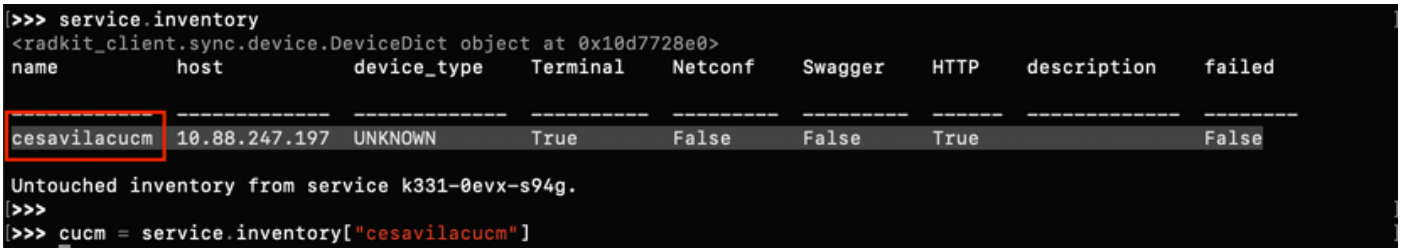
```
>>> service.update_inventory().wait()
```

SSH访问

步骤1.从资产列表创建对象。

<#root>

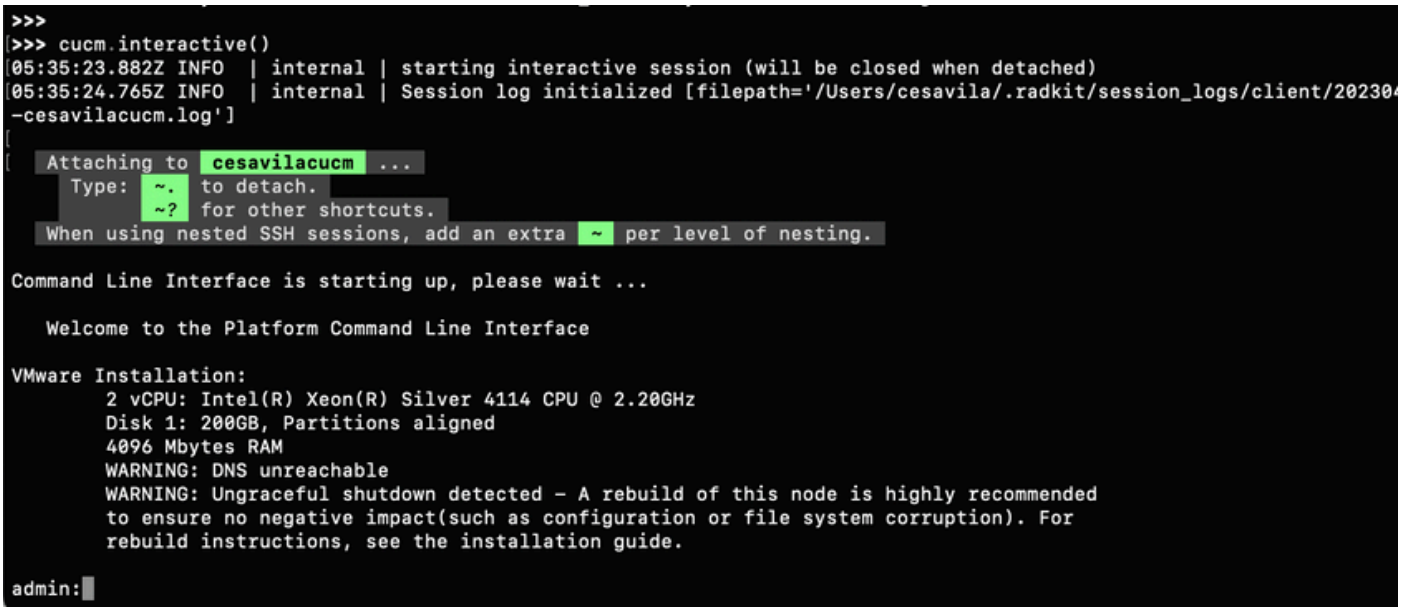
```
>>> cucm = service.inventory['cesavilacucm']
```



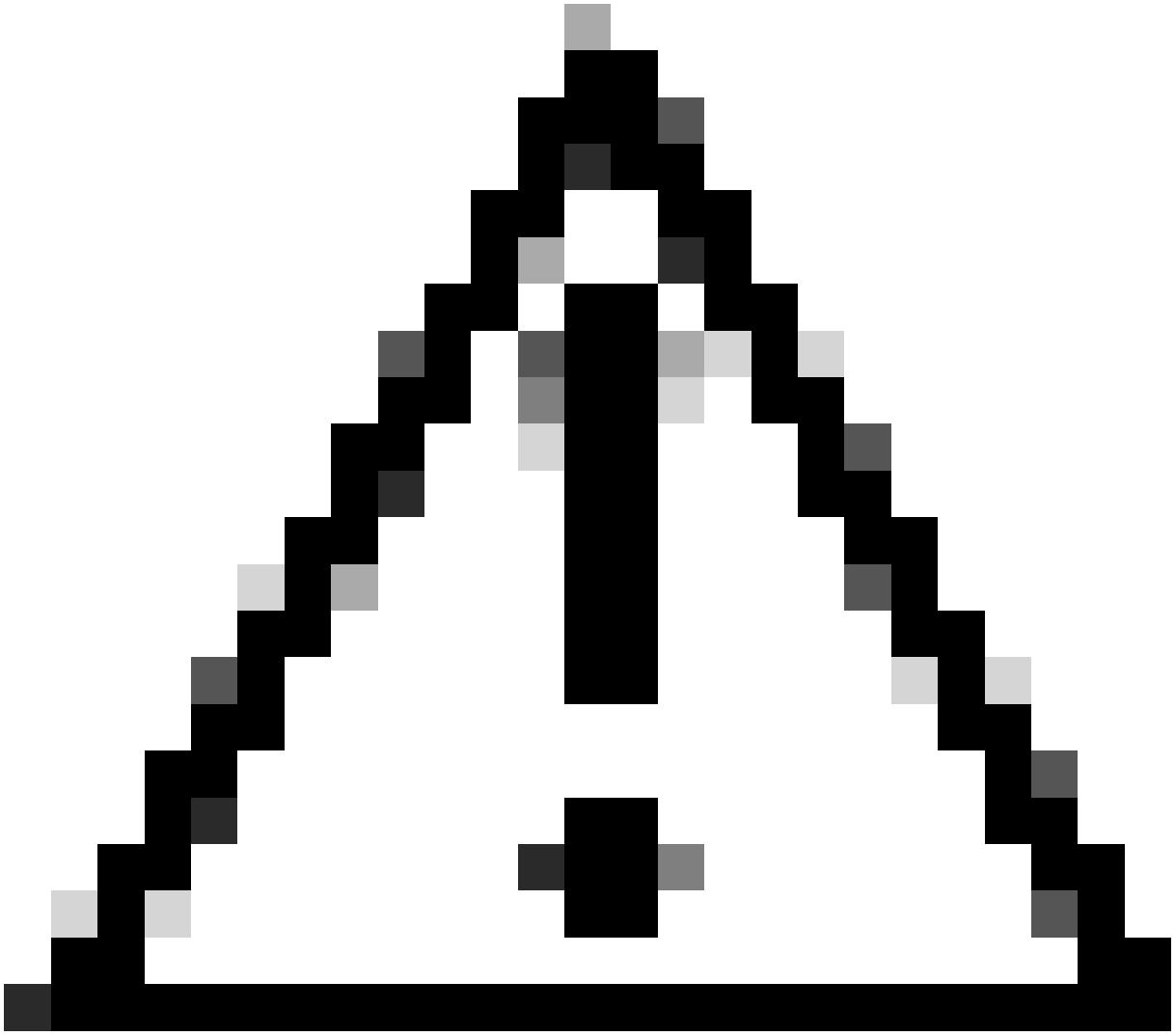
步骤2.使用交互式命令启动SSH会话。

<#root>

```
>>> cucm.interactive()
```



步骤3.现在您可以正常管理设备。



警告：

- 在用户环境中操作时，请始终牢记我们的责任。
- RADKit必须用作数据收集工具。
- 未经用户许可，请勿进行任何更改。
- 在案例说明中记录所有调查结果。

GUI访问

- HTTP 代理

步骤1.确保在设备配置上的RADKit服务中添加了HTTP凭证。

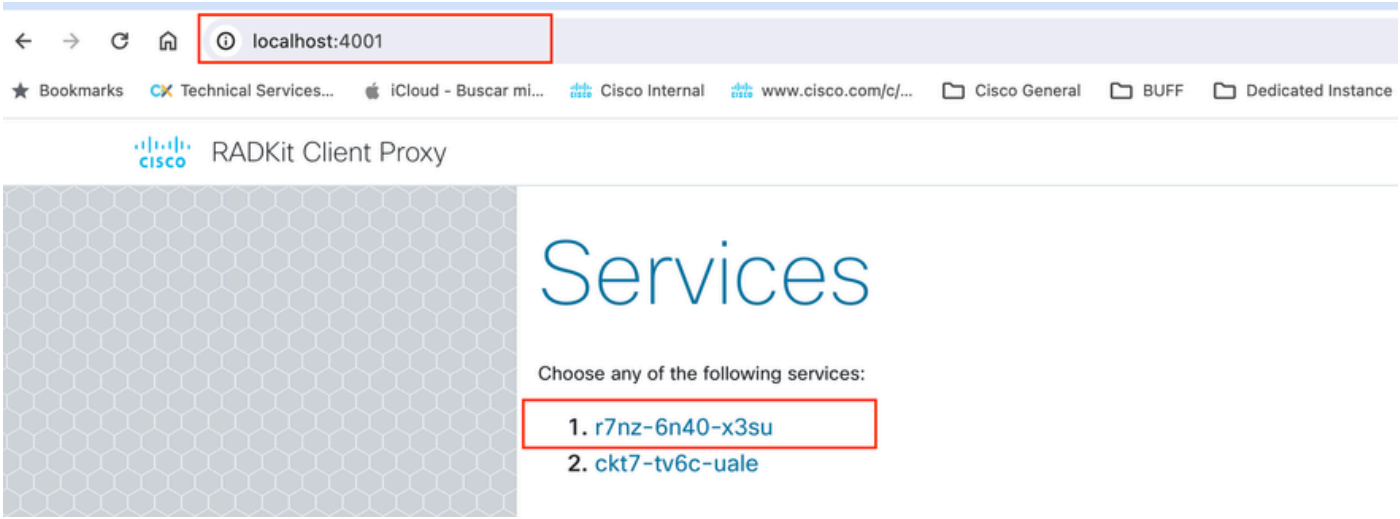
步骤2.在Radkit客户端上启动HTTP代理，并定义用于连接到代理的本地端口。

<#root>

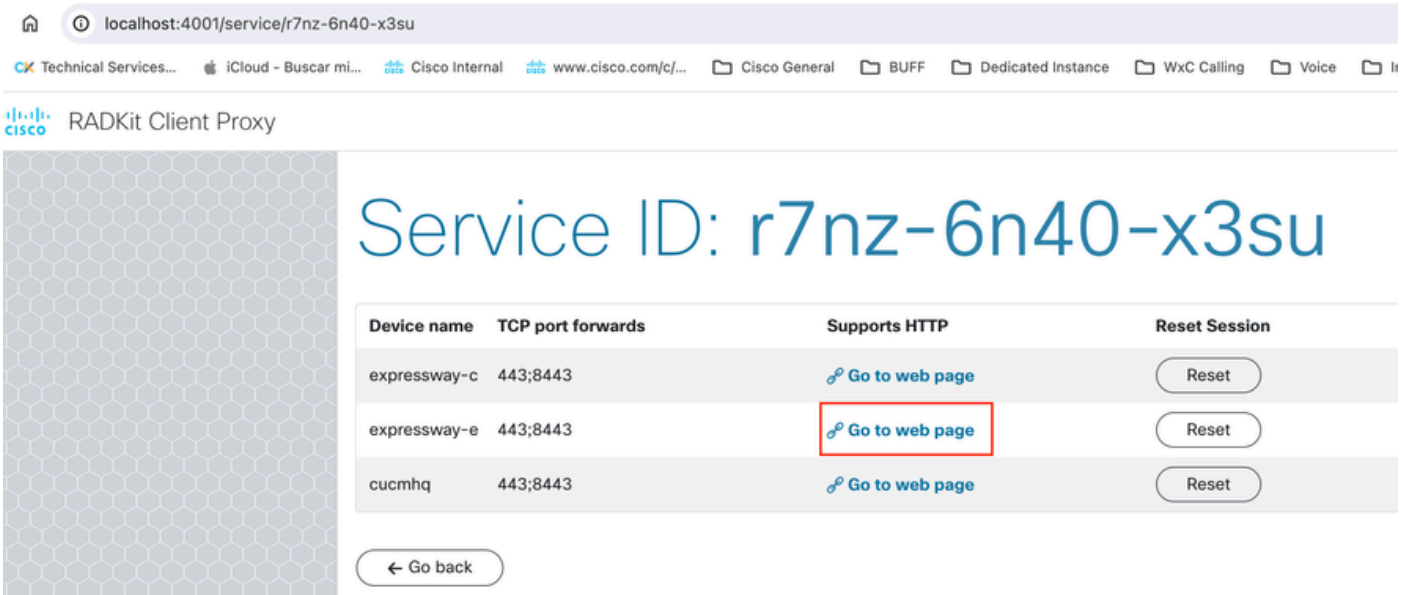
```
>>> http_proxy = client.start_http_proxy(4001)
```

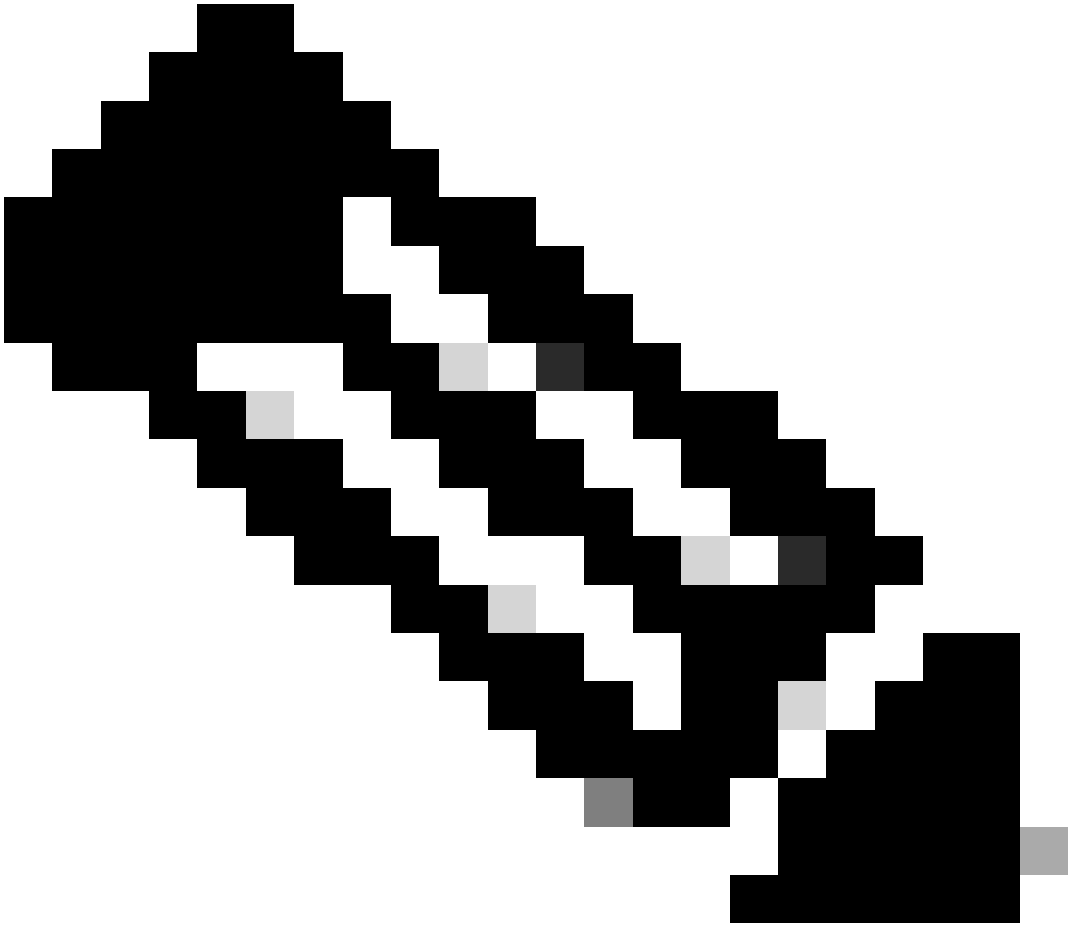
```
>>>
>>> http_proxy = client.start_http_proxy(4001)
22:24:19.981Z WARNI | HTTP proxy is NOT PROTECTED by username/password
>>>
```

第3步：从Web浏览器导航到https://localhost:4001，然后选择要连接到的服务。



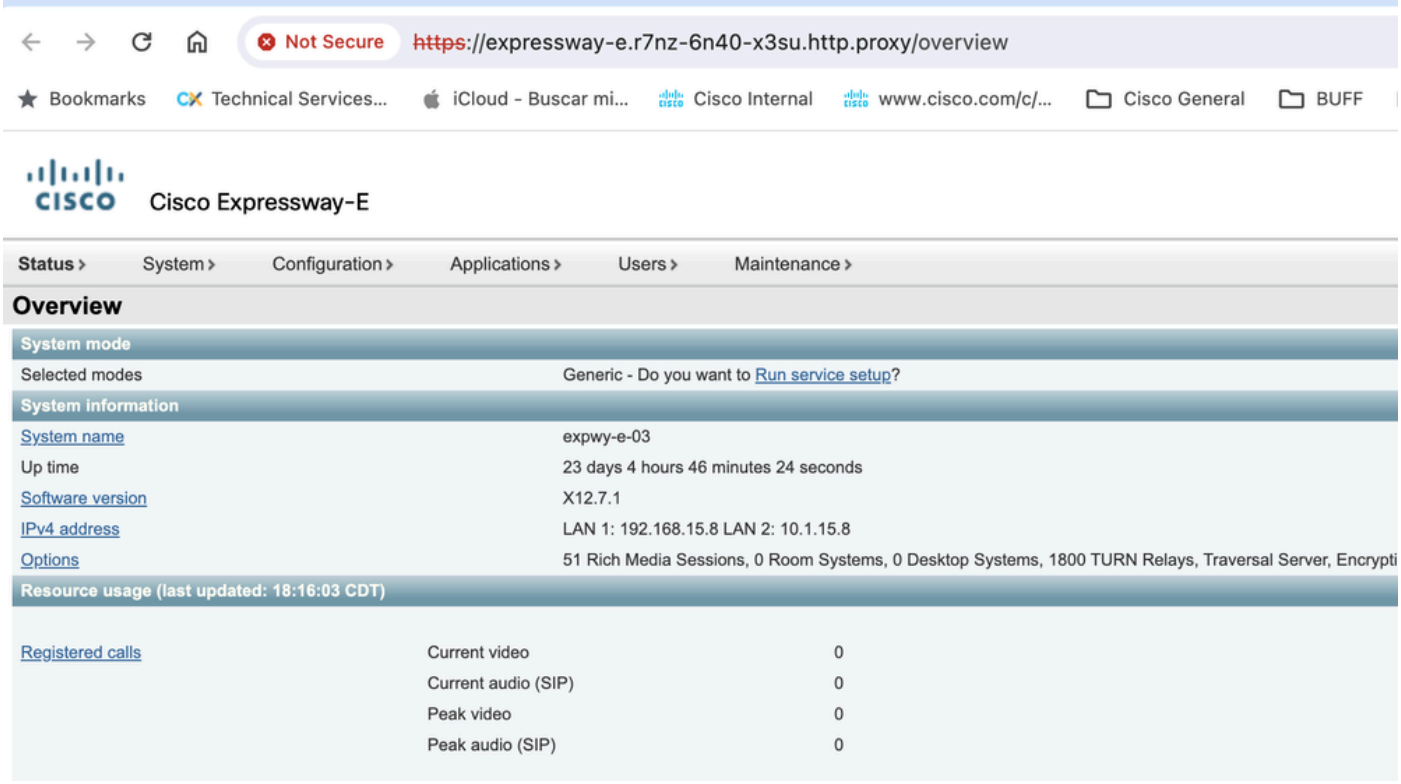
步骤4.点击正确设备上的选项Go to Web Page以连接到其网页。





注：第一次在RADKit客户端上设置HTTP代理时，建议先单击选项“重置每台设备”，然后再尝试打开设备网页。

步骤5.显示网页。



- 端口转发

步骤1.检验为设备配置的TCP转发端口。

<#root>

```
>>> cucm.forwarded_tcp_ports
```

步骤2.配置要与设备的目标端口映射的本地端口，您必须使用本地端口访问设备GUI。

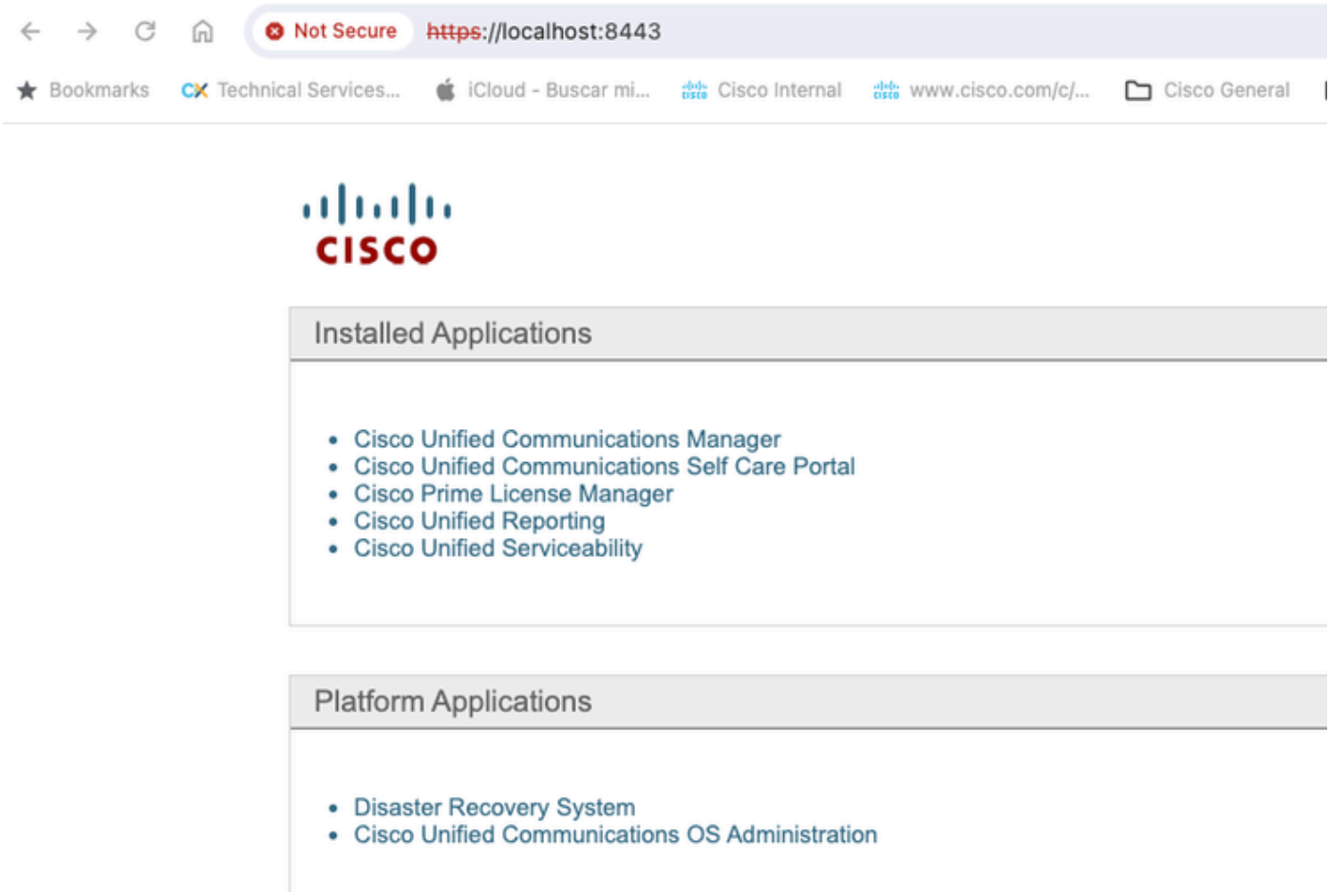
<#root>

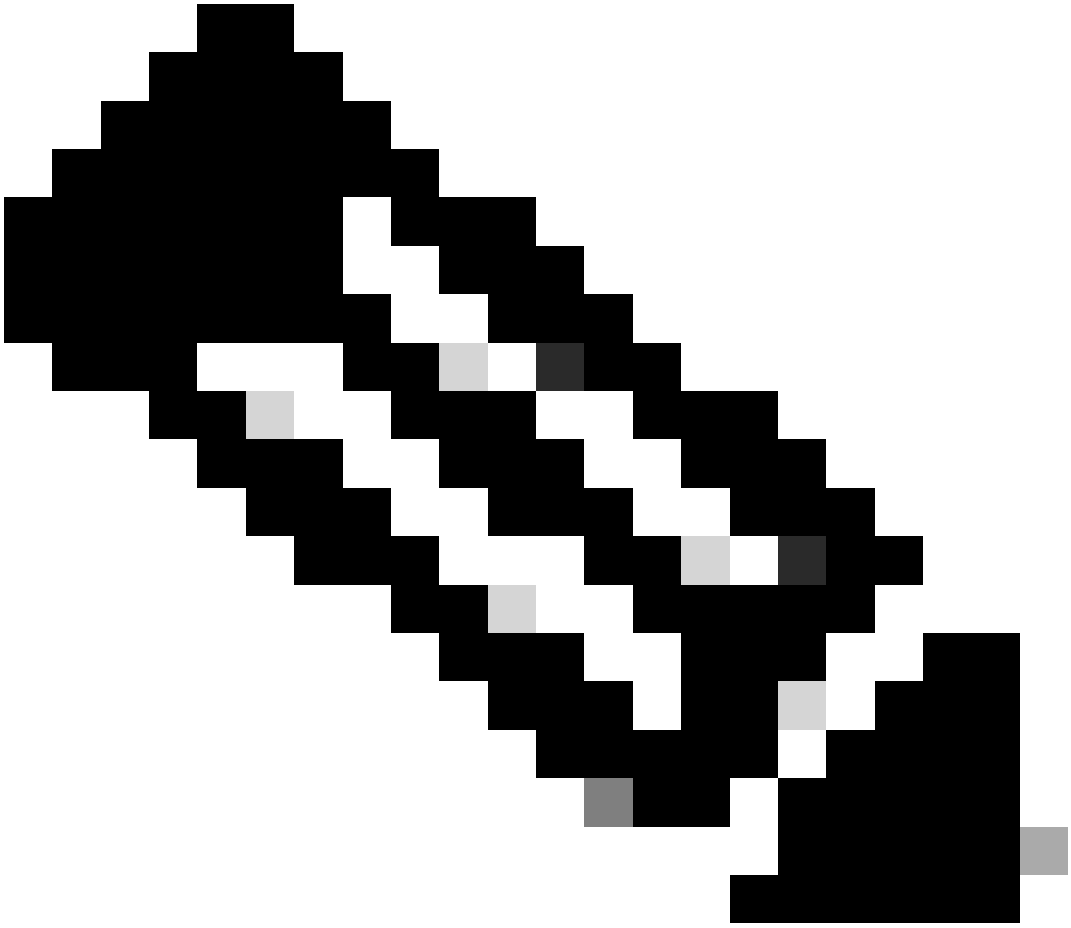
```
>>> cucm.forward_tcp_port(local_port=8443, destination_port=443)
```

```
>>>
>>> cucm.forward_tcp_port(12443,443)
[RUNNING] <radkit_client.sync.port_forwarding.TCPPortForwarder object at 0x10ceb3d60>
----------------  ------------
status            RUNNING
serial            None
device_name       cesavilacucm
local_port        12443
destination_port  443
#active           0
#failed           0
#closed           0
#total            0
bytes up          0
bytes down        0
exception         None
----------------  ------------
```

步骤3.打开浏览器并键入URL以及步骤2中配置的端口：https://localhost:8443。

现在可以访问设备的GUI。





注意：要访问产品的GUI，您仍需要凭据才能登录，因此建议用户创建只读用户帐户以进行访问。

日志收集

- RTMT

步骤1.检验端口8443是否列在为该设备配置的TCP转发端口中。

<#root>

```
>>> cucm.forwarded_tcp_ports
```

```
>>> cucm.forwarded_tcp_ports
'443;8443'
>>>
```

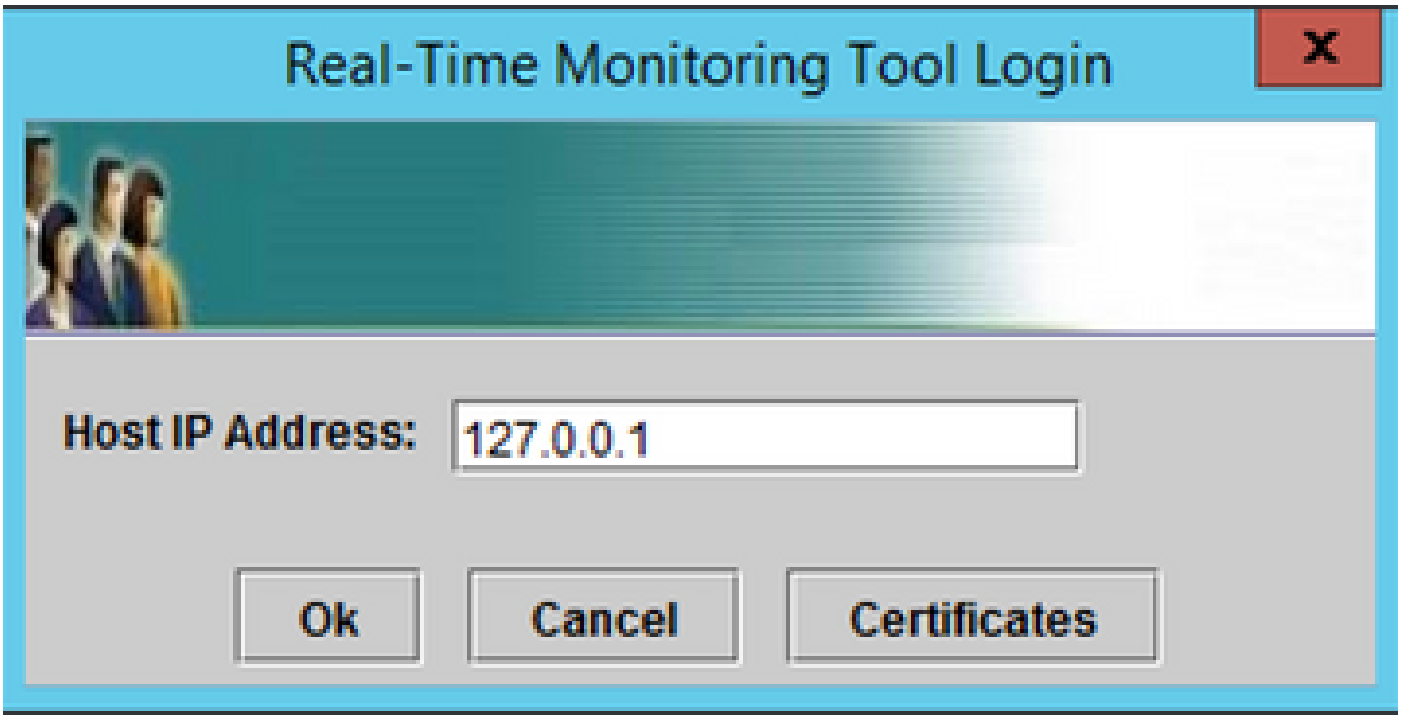
步骤2.将同一端口8443配置为本地端口，以便与端口8443作为设备的目标端口进行映射。

<#root>

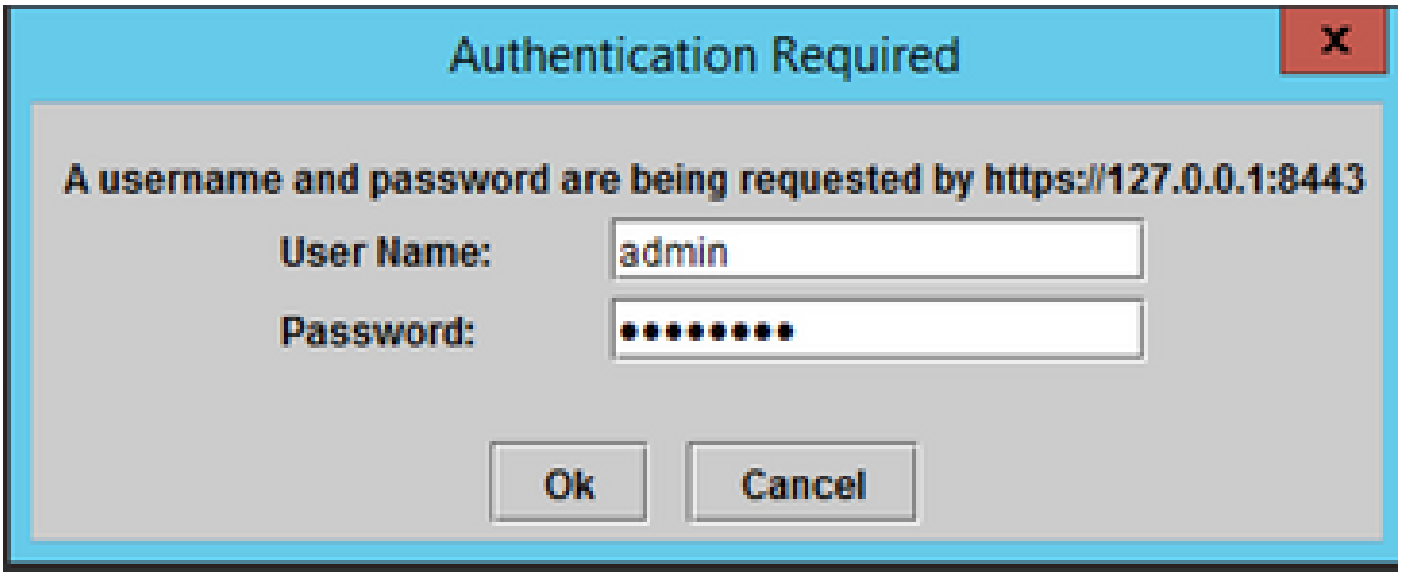
```
>>> cucm.forward_tcp_port(local_port=8443, destination_port=8443)
```

```
>>> cucm.forward_tcp_port(8443,8443)
[RUNNING] <radkit_client.sync.port_forwarding.TCPPortForwarder object at 0x1077defa0>
----------------  ------------
status            RUNNING
serial            None
device_name       cesavilacucm
local_port        8443
destination_port  8443
#active           0
#failed           0
#closed           0
#total            0
bytes up          0
bytes down        0
exception         None
----------------  ------------
```

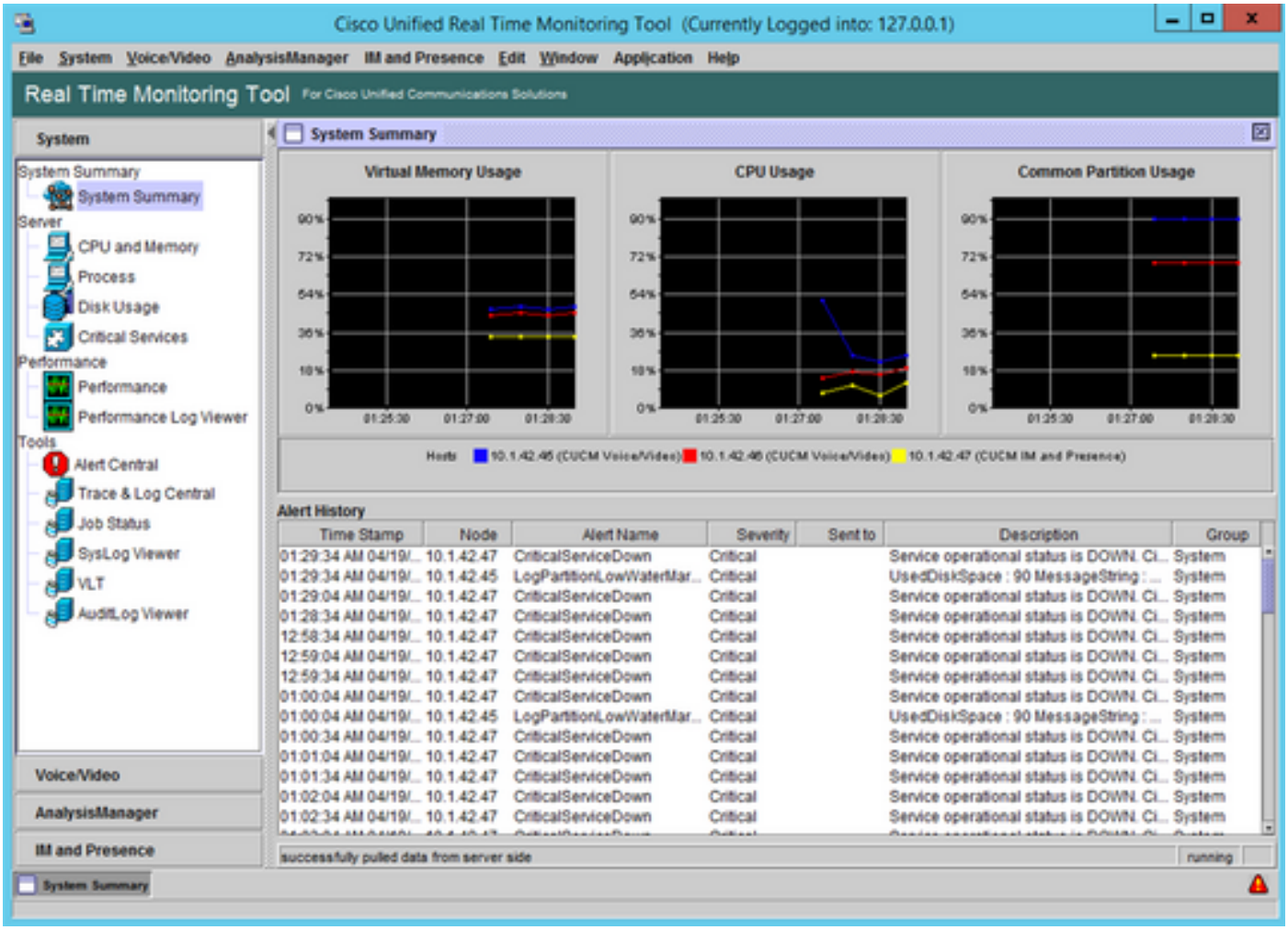
步骤3.打开RTMT并在主机IP地址中键入127.0.0.1，它会自动使用端口8443。



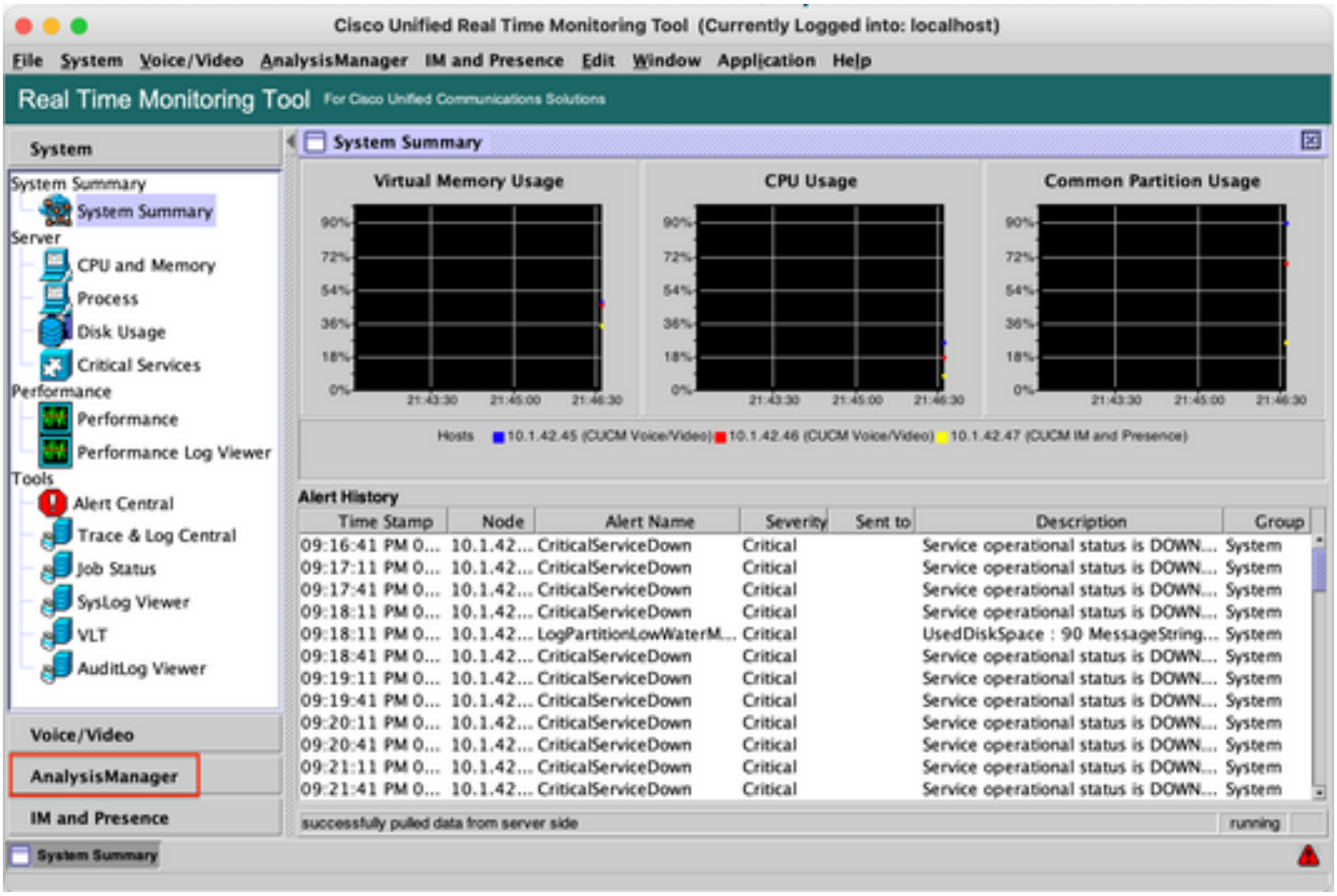
步骤4.使用正确的凭证登录。



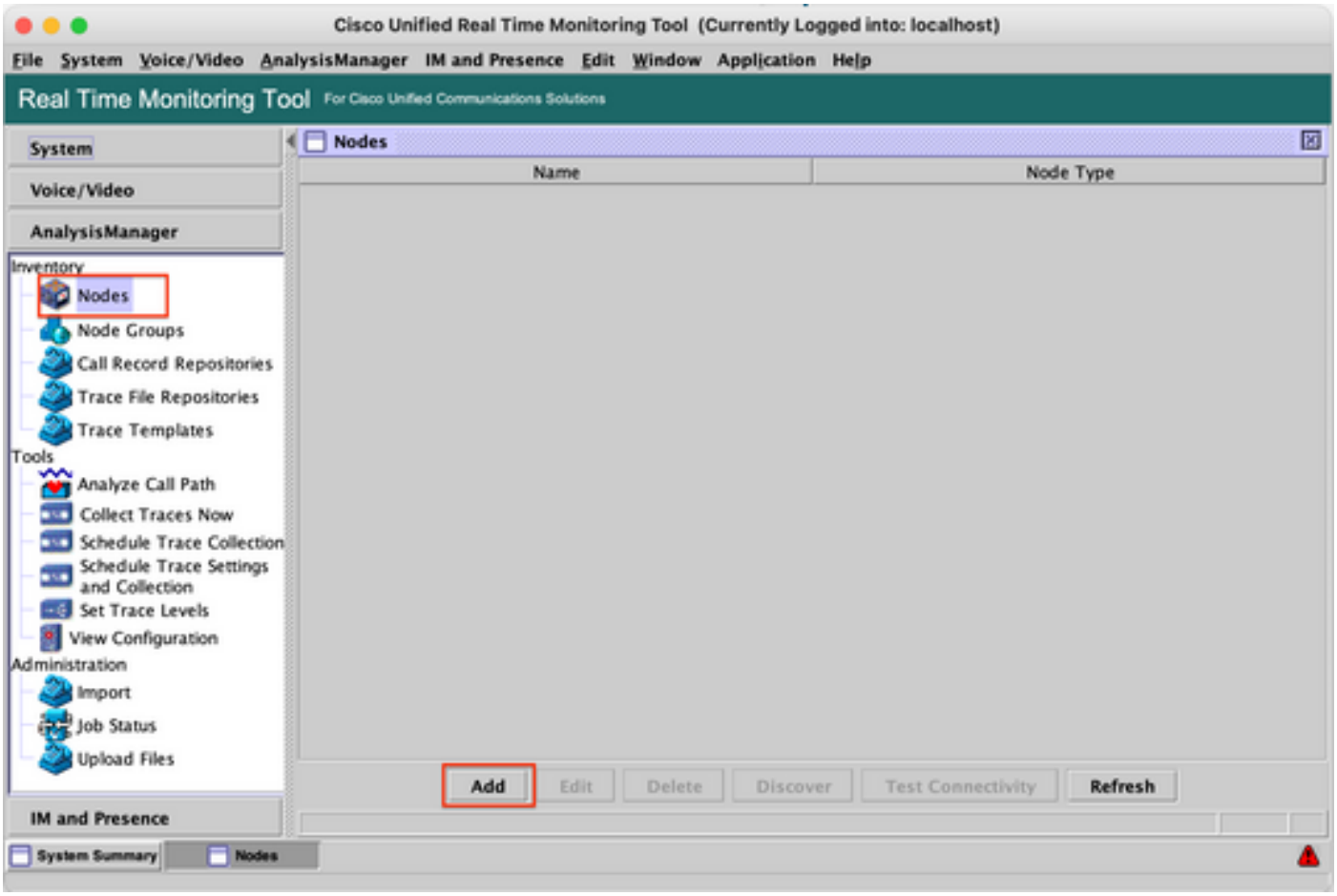
步骤5. RTMT显示。

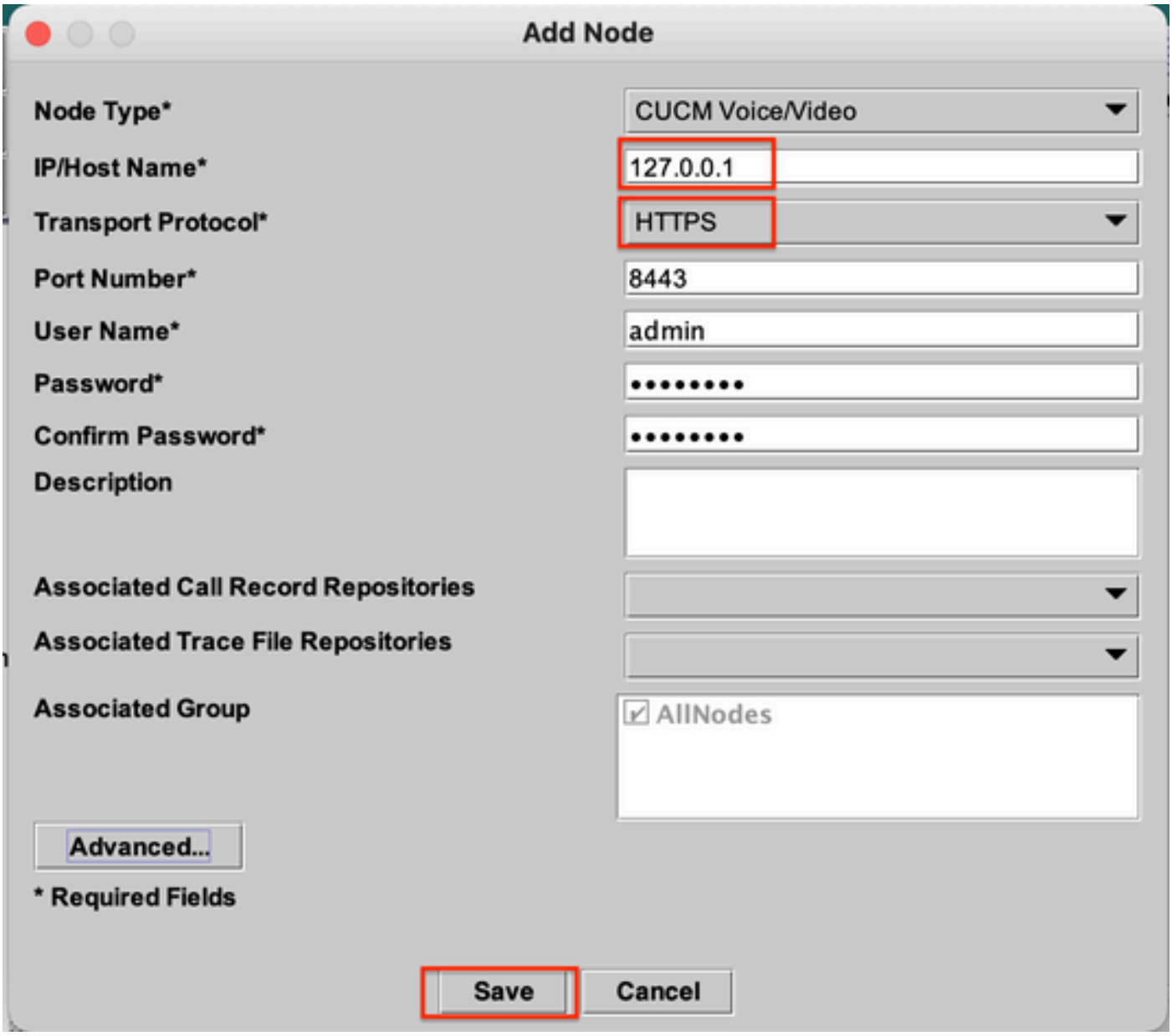


步骤6.转到左面板上的AnalysisManager。

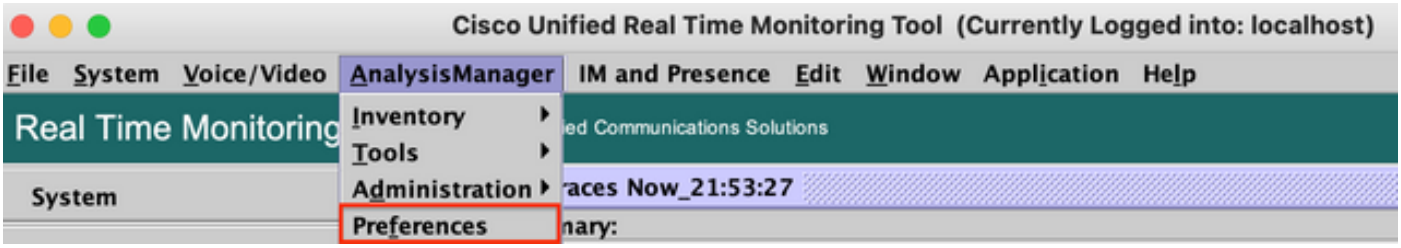


步骤7.单击节点和添加以配置要使用localhost和转发的TCP端口添加的设备的详细信息。

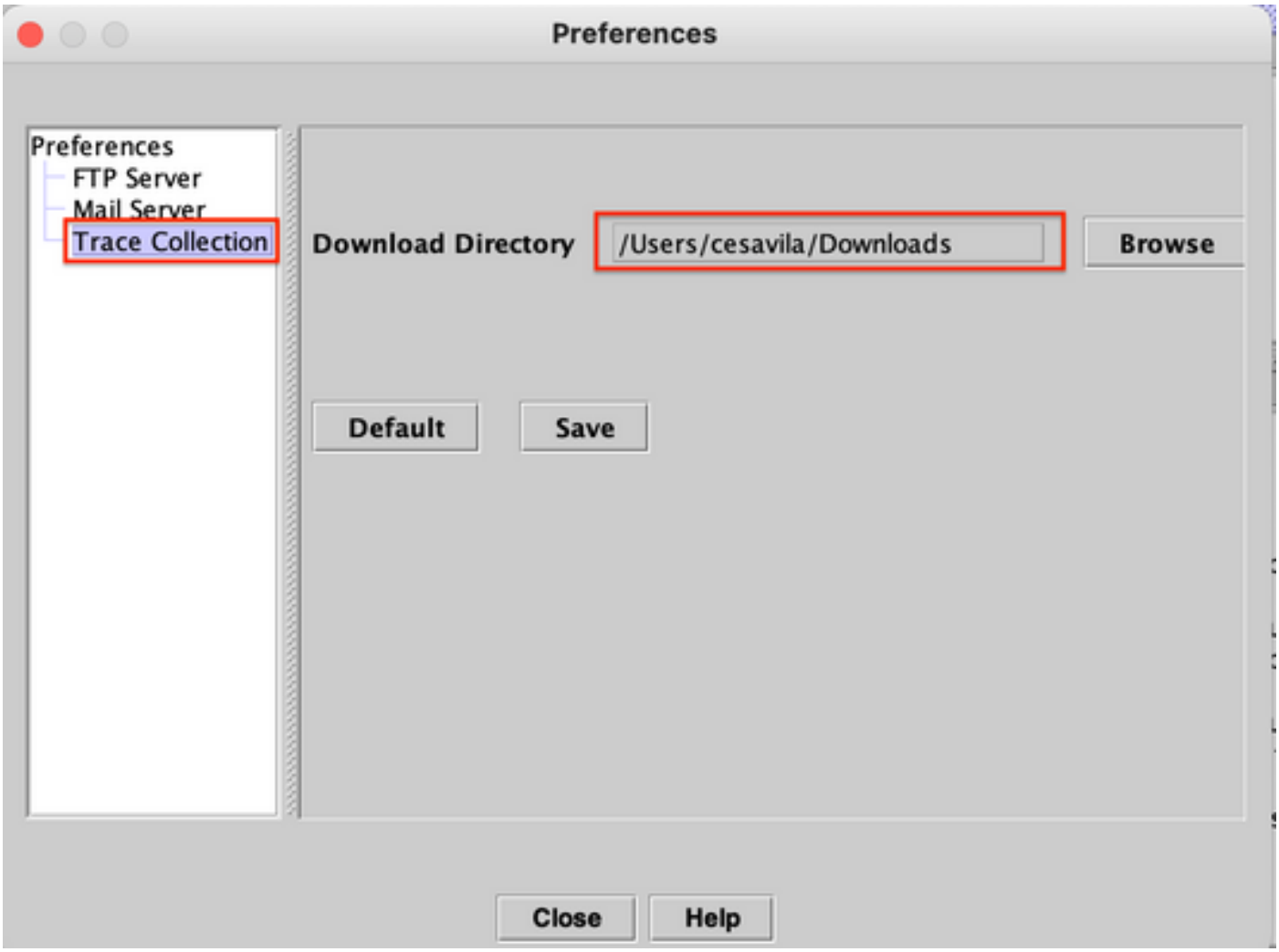




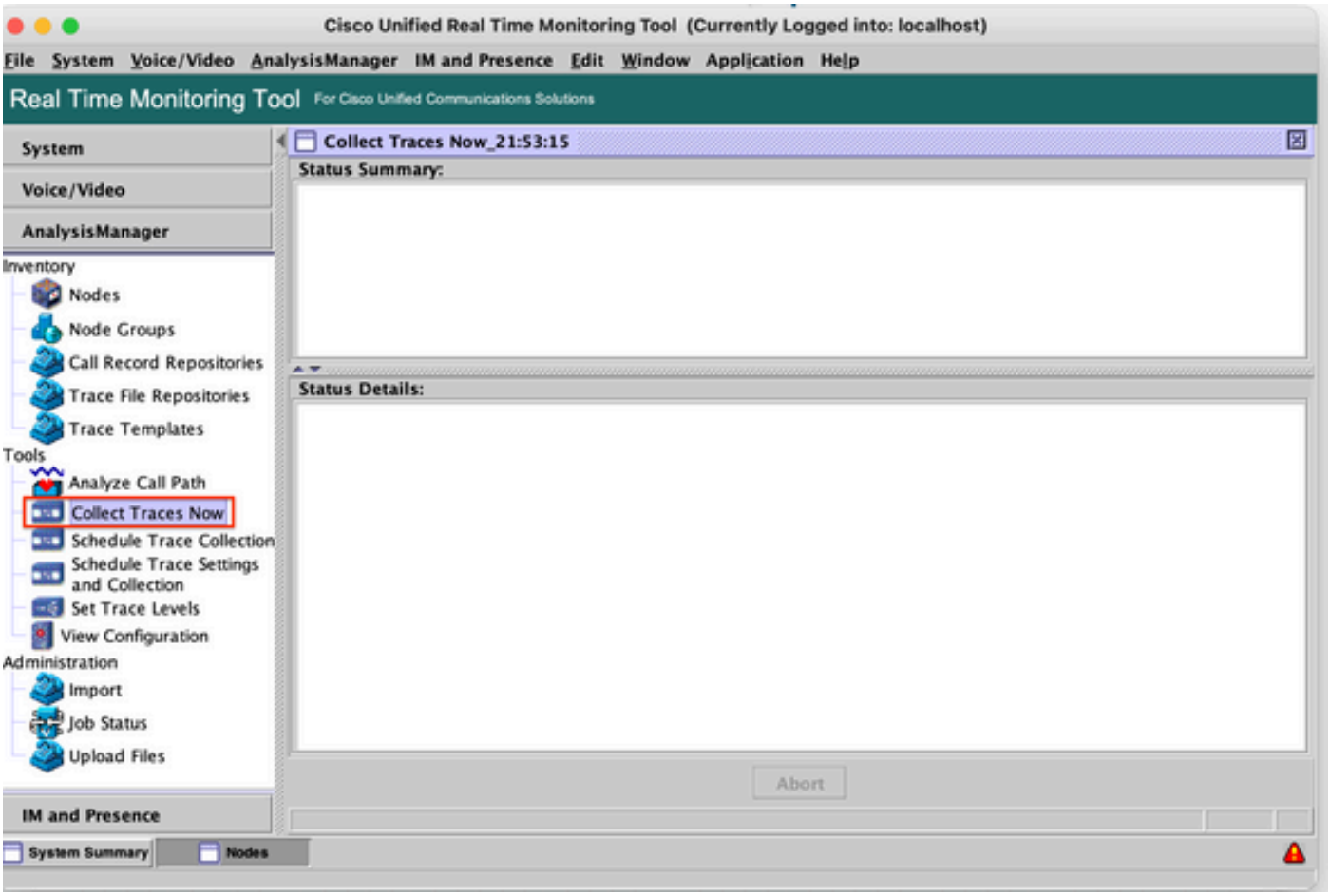
步骤8.在顶部菜单上单击“分析管理器”(Analysis Manager)，然后选择“首选项”(Preferences)。



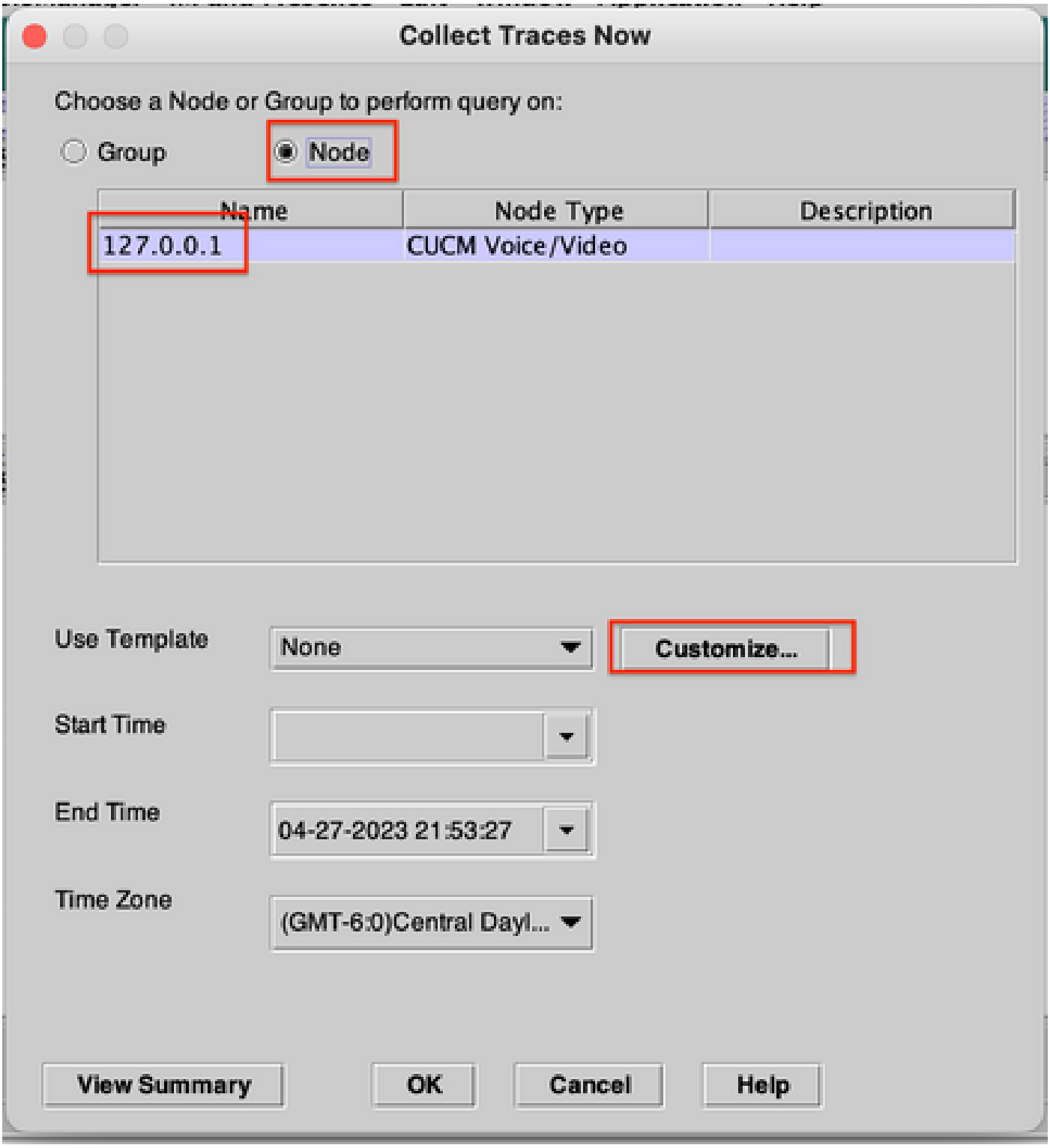
第9步：转至Trace Collection并选择Correct文件夹以下载日志，单击Save，然后单击Close。



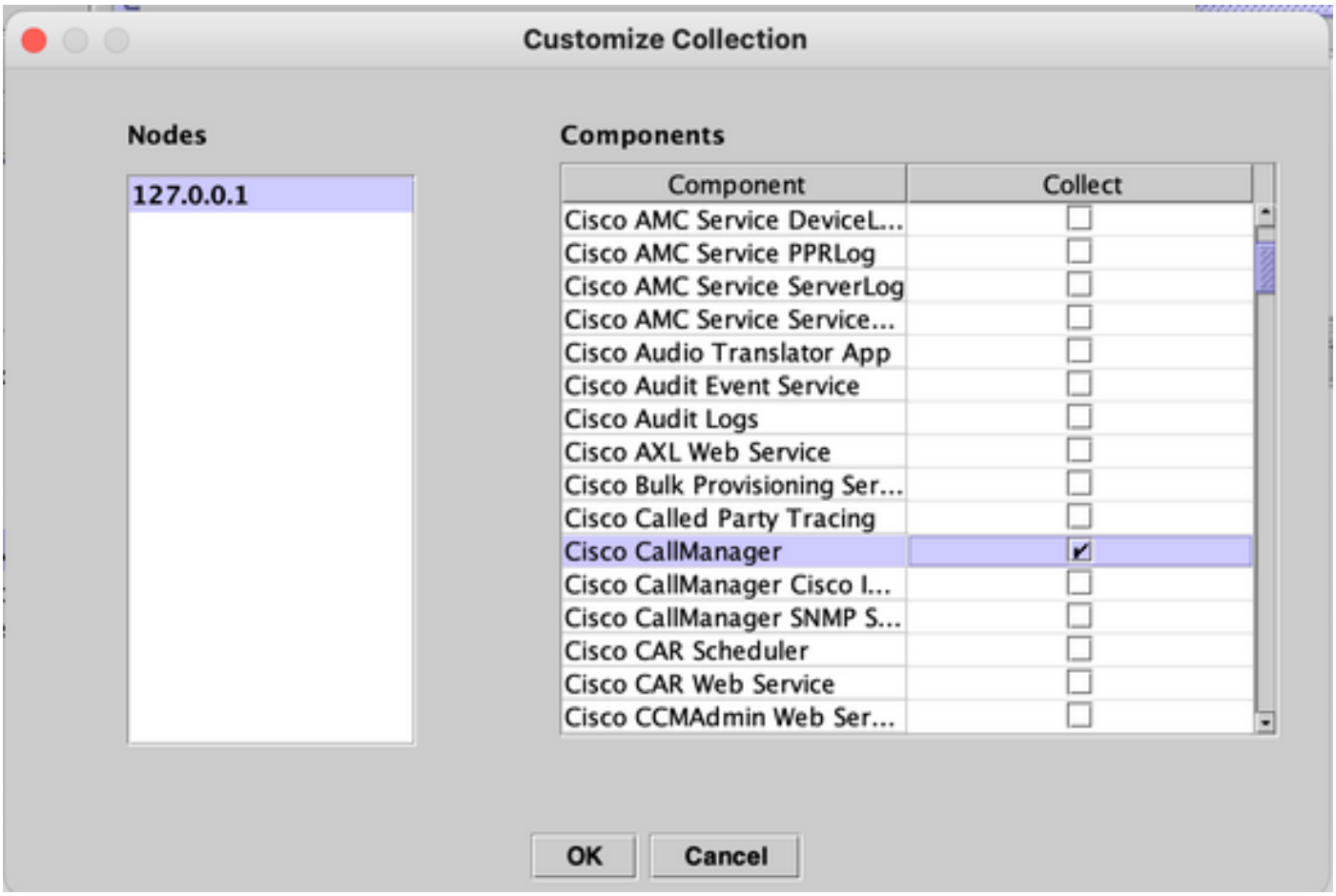
步骤10.立即转至Collect Traces。



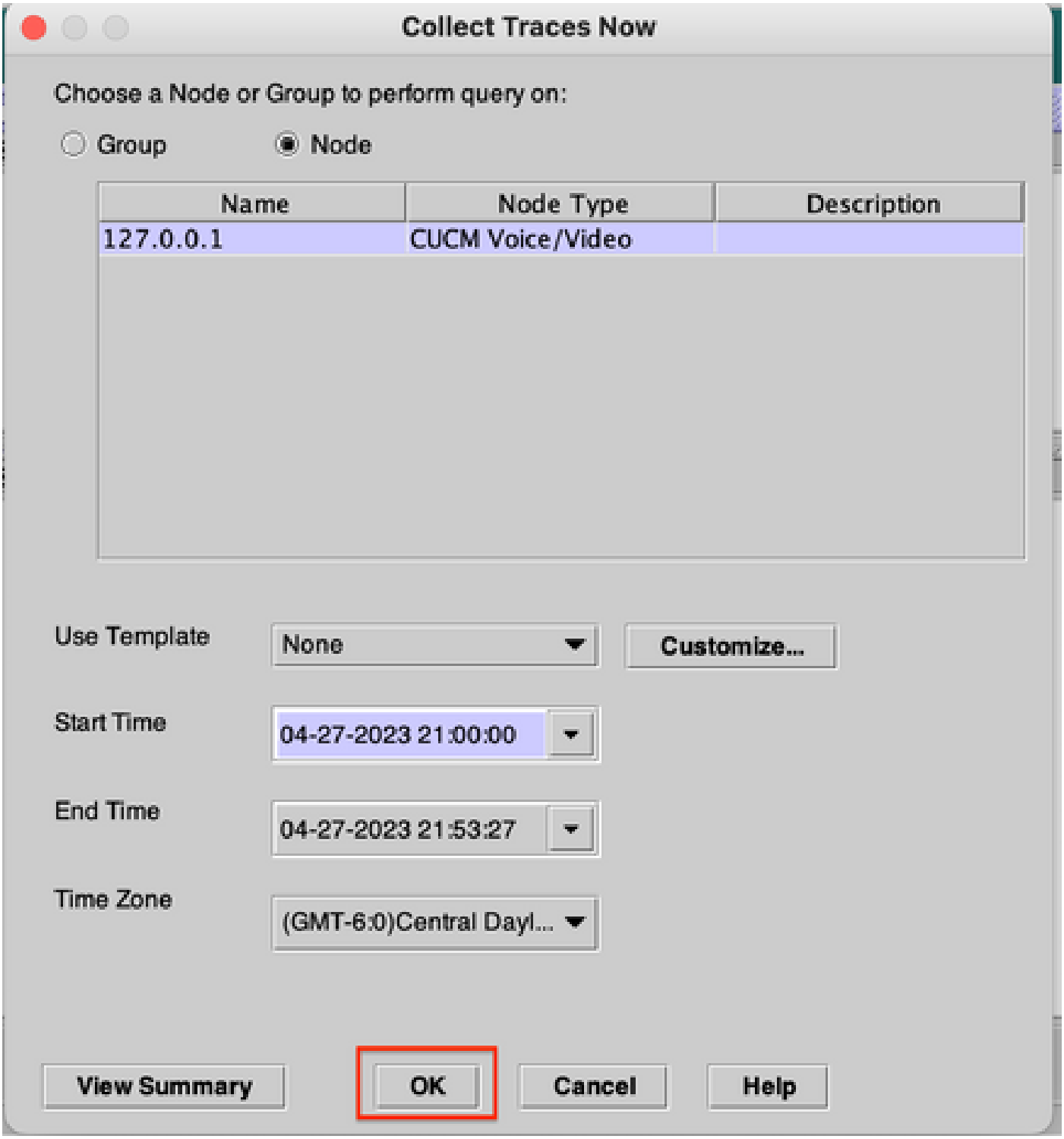
步骤11.选择选项Node，选择在步骤7中添加的设备，然后单击Customize。



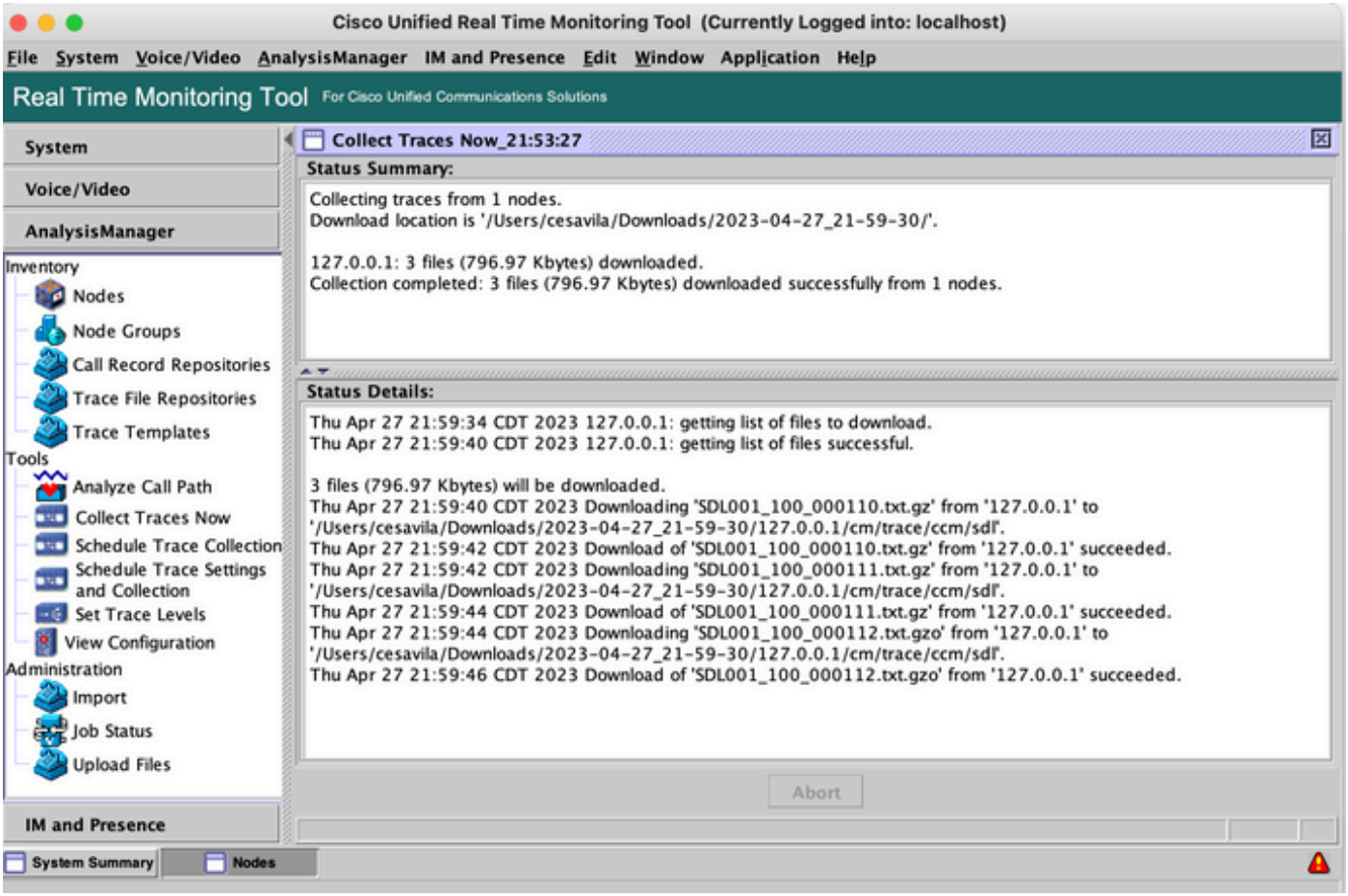
步骤12.选择要从设备收集的日志，然后点击OK。



第13步：最终选择要收集的日志的开始时间和结束时间，然后单击确定。



步骤14.文件已成功下载到本地PC（RADKit客户端PC）。



- SOAP API

CUCM当前支持SOAP API。此外，CMS、Expressway、CVP等支持Swagger。

步骤1.确保在设备配置上的RADKit服务中添加了HTTP凭证。

步骤2.在RADKit Client上运行HTTP Post命令，使用必要的参数和报头指定资源路径和请求正文。

```
>>>
... r = cucm.http.post('/logcollectionservice2/services/LogCollectionPortTypeService', content = '''<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/so
... ap/envelope/" xmlns:soap="http://schemas.cisco.com/ast/soap">
... <soapenv:Header/>
... <soapenv:Body>
... <soap:FileName>/var/log/active/cm/trace/ccm/sdl/SDL002_100_000819.txt.gz</soap:FileName>
... </soapenv:Body>
... </soapenv:Envelope>''', headers = {"Content-Type": "text/xml; charset=utf-8", "SOAPAction": "GetOneFile"}, postprocessors = ['cucm-extract'])
>>>
>>>
```

注意：postprocessors选项“cucm-extract”用于删除HTTP响应报头，以便将日志保存到文件中。

步骤3.将内容保存到文件中，以获取Trace File保存在本地PC中。

<#root>

```
>>> content = r.content
```

```
>>> with open('SDL002_100_000819.txt.gz', 'wb') as file:
        file.write(content)
```

RADKit使用案例

如前所述，RADKit无需在Webex上即可提供到网络设备（包括协作服务器）的安全连接。其理念是通过对所需设备进行按需访问，简化数据收集方面的一些挑战。

谈到协作部署，RADKit目前对于各种问题都非常有用，例如：

- 数据库复制问题。
- 证书重新生成过程。
- 系统运行状况检查。
- 在GUI/CLI中进行配置验证。
- 通过Web界面（例如CER、Expressway、CIMC等）进行日志收集。
- 通过语音网关上的CLI调试日志。

相关信息

- RADKit主页https://radkit.cisco.com/
- 外部RADKit支持页https://community.cisco.com/t5/radkit-discussions/bd-p/disc-radkit

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）。