

# 通过内部报告模式对Catalyst SD-WAN Manager上的许可证同步进行故障排除

## 目录

---

[简介](#)

[要求](#)

[Error](#)

[故障排除方法](#)

[解决方法](#)

---

## 简介

本文档介绍如何通过内部报告模式对Catalyst SD-WAN Manager上同步许可证时遇到的错误进行故障排除。

## 要求

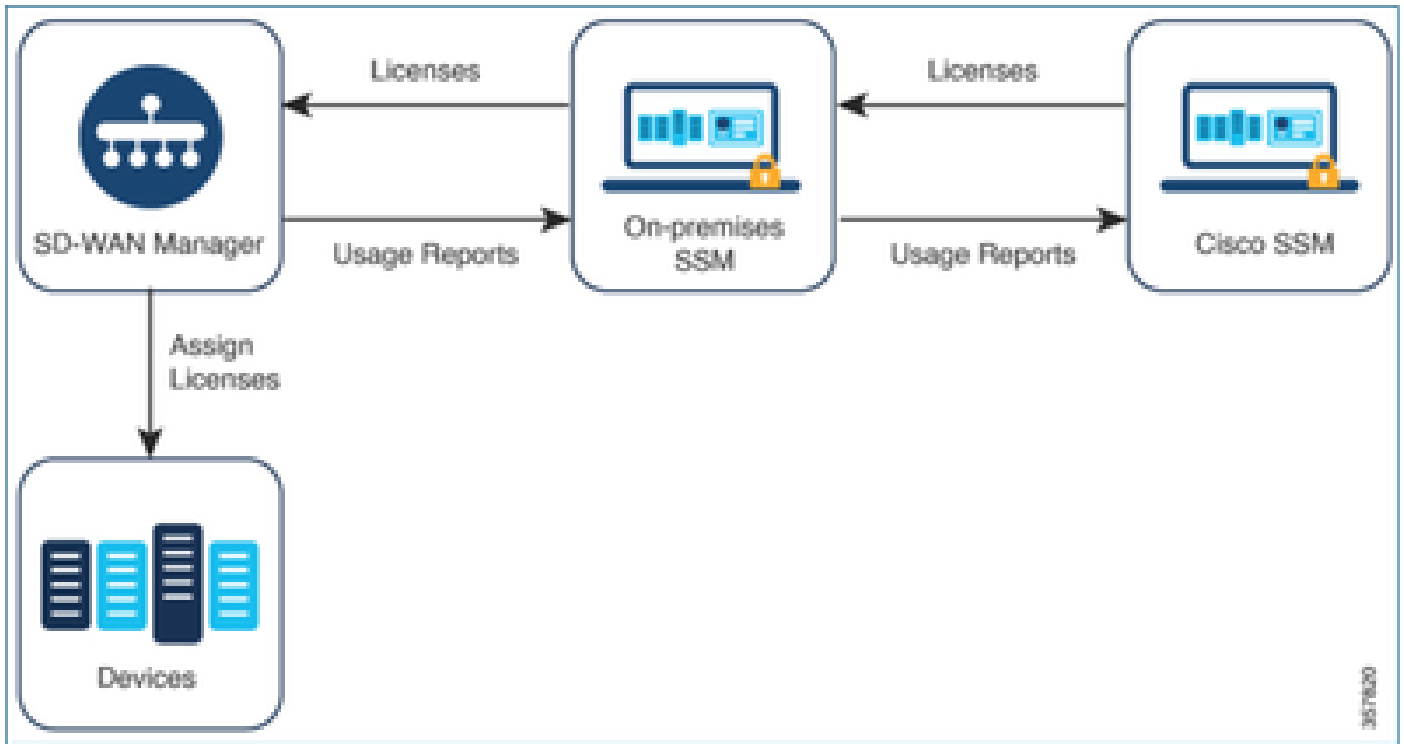
对于Catalyst SD-WAN Manager未直接连接到互联网的情况，使用代理服务器可以提供对基于互联网的服务（例如Cisco SSM）或本地内部SSM的访问。

最低版本：Catalyst SD-WAN Manager版本20.9.1

思科本地智能软件管理器(SSM on-prem)是思科智能许可解决方案，使您能够从本地服务器管理许可证，而不必直接连接到思科SSM。该解决方案涉及设置思科SSM内部许可证服务器，该服务器定期将其许可证数据库与思科SSM同步，并且功能与思科SSM类似，同时在本地运行。

Catalyst SD-WAN Manager支持使用思科SSM内部服务器管理许可证，使用一种称为内部服务器的模式。内部模式对于使用Cisco SSM内部模式以满足严格安全策略的组织（该策略不允许网络设备通过直接互联网连接与Cisco SSM通信）非常有用。

在内部模式中运行时，Catalyst SD-WAN Manager每24小时与思科SSM内部许可证服务器同步许可证信息。在此同步过程中，Catalyst SD-WAN Manager接收对可用许可证的任何更新，并将许可证使用情况报告发送到Cisco SSM内部许可证服务器。您可以随时同步许可证。



## 使用本地思科智能软件管理器的优势

安全策略或其他情况要求Catalyst SD-WAN Manager未连接到互联网的组织有两个选项可用于使用策略管理智能许可证的许可证：

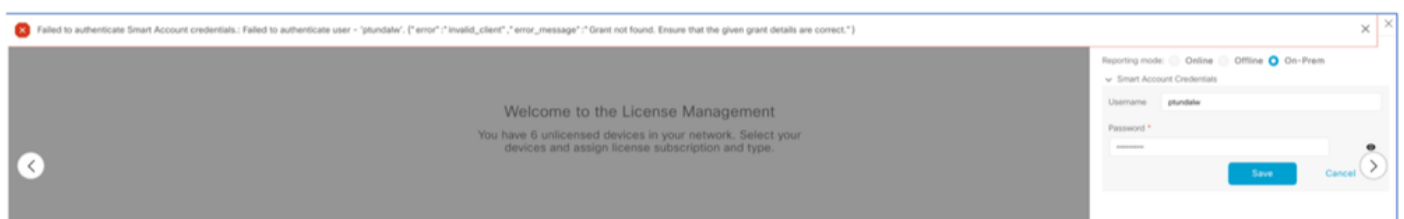
- 使用离线模式，这需要在Catalyst SD-WAN Manager和Cisco SSM之间手动传输文件。
- 使用可通过本地连接访问Catalyst SD-WAN Manager的Cisco SSM内部服务器。

这两种方法都解决了在Cisco SSM和Catalyst SD-WAN Manager之间传输许可证信息的需要。只要可以使用内部模式，此模式就可以显著减少在Catalyst SD-WAN Manager和Cisco SSM之间手动传输文件的维护开销，这是离线模式所必需的。

## Error

从Catalyst SD-WAN Manager GUI同步智能凭证时，我们会收到以下错误：

Failed to authenticate Smart Account credentials.: Failed to authenticate user - 'admin'. {"error": "inv



## 故障排除方法

- vManage必须位于代码20.9.1或更高版本上。
- 在Catalyst SD-WAN Manager License Management部分放置智能帐户凭证时，检查Catalyst SD-WAN Manager (vmanage-server.logs)上的日志。
- 确保本地SSM团队共享的客户端ID和密钥正确。
- CSSM服务器IP的vManage上的TCPDUMP
- 验证Catalyst SD-WAN Manager上的DNS配置正确且能够ping通cloudsso.cisco.com
- 让内部SSM团队参与，并请求SSM团队在内部服务器端进行调试。

Catalyst SD-WAN Manager IP : 10.66.76.81 / 192.168.10.1

CSSM服务器IP : 10.106.66.55

SSM服务器IP的vManage上的TCPDump :

```
um8_vManage# tcpdump vpn 0 interface eth0 options "host 10.106.66.55 -nn -vv"
```

```
tcpdump -p -i eth0 -s 128 host 10.106.66.55 -nn -vv in VPN 0
```

```
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 128 bytes
```

```
12:15:06.407513 IP (tos 0x0, ttl 64, id 24618, offset 0, flags [DF], proto TCP (6), length 52)
```

```
192.168.10.1.57886 > 10.106.66.55.8443: Flags [S], cksum 0xfadb (incorrect -> 0xdf91), seq 74638621
```

```
12:15:06.651698 IP (tos 0x20, ttl 44, id 0, offset 0, flags [DF], proto TCP (6), length 52)
```

```
10.106.66.55.8443 > 192.168.10.1.57886: Flags [S.], cksum 0x1b34 (correct), seq 2758352947, ack 746
```

```
12:15:06.651768 IP (tos 0x0, ttl 64, id 24619, offset 0, flags [DF], proto TCP (6), length 40)
```

```
192.168.10.1.57886 > 10.106.66.55.8443: Flags [.] , cksum 0xfacf (incorrect -> 0xcce1), seq 1, ack 1
```

```
12:15:06.654592 IP (tos 0x0, ttl 64, id 24620, offset 0, flags [DF], proto TCP (6), length 212)
```

```
192.168.10.1.57886 > 10.106.66.55.8443: Flags [P.], seq 1:173, ack 1, win 229, length 172
```

```
12:15:06.899695 IP (tos 0x0, ttl 41, id 44470, offset 0, flags [DF], proto TCP (6), length 40)
```

```
10.106.66.55.8443 > 192.168.10.1.57886: Flags [.] , cksum 0xcc2d (correct), seq 1, ack 173, win 237,
```

```
12:15:06.911484 IP (tos 0x0, ttl 41, id 44471, offset 0, flags [DF], proto TCP (6), length 1420)
```

```
10.106.66.55.8443 > 192.168.10.1.57886: Flags [.] , seq 1:1381, ack 173, win 237, length 1380
```

```
12:15:06.911542 IP (tos 0x0, ttl 41, id 44472, offset 0, flags [DF], proto TCP (6), length 254)
```

```
10.106.66.55.8443 > 192.168.10.1.57886: Flags [P.], seq 1381:1595, ack 173, win 237, length 214
```

```
12:15:06.911573 IP (tos 0x0, ttl 64, id 24621, offset 0, flags [DF], proto TCP (6), length 40)
```

```
192.168.10.1.57886 > 10.106.66.55.8443: Flags [.] , cksum 0xfacf (incorrect -> 0xc6bb), seq 173, ack
```

```
12:15:06.911598 IP (tos 0x0, ttl 64, id 24622, offset 0, flags [DF], proto TCP (6), length 40)
```

```
192.168.10.1.57886 > 10.106.66.55.8443: Flags [.] , cksum 0xfacf (incorrect -> 0xc5cf), seq 173, ack
```

```
12:15:06.923929 IP (tos 0x0, ttl 64, id 24623, offset 0, flags [DF], proto TCP (6), length 234)
```

192.168.10.1.57886 > 10.106.66.55.8443: Flags [P.], seq 173:367, ack 1595, win 273, length 194

## 内部服务器日志：

```
[root@SSM-On-Prem log]# tail -f messages
```

```
Jan 13 11:13:36 SSM-On-Prem chronyd[1319]: Source 172.20.226.229https://172.20.226.229 replaced with 172.20.226.229
Jan 13 11:14:09 SSM-On-Prem b09c1e3b5d81: 1:M 13 Jan 2023 11:14:09.049 * 100 changes in 300 seconds. Saving to disk
Jan 13 11:14:09 SSM-On-Prem b09c1e3b5d81: 1:M 13 Jan 2023 11:14:09.050 * Background saving started by process 1319
Jan 13 11:14:09 SSM-On-Prem b09c1e3b5d81: 4617:C 13 Jan 2023 11:14:09.052 * DB saved on disk
Jan 13 11:14:09 SSM-On-Prem b09c1e3b5d81: 4617:C 13 Jan 2023 11:14:09.053 * RDB: 0 MB of memory used by sorted set
Jan 13 11:14:09 SSM-On-Prem b09c1e3b5d81: 1:M 13 Jan 2023 11:14:09.150 * Background saving terminated with success
Jan 13 11:14:46 SSM-On-Prem 1a1fca641d0a: Redis#exists(key) will return an Integer in redis-rb 4.3. exists() returns a boolean
Jan 13 11:14:46 SSM-On-Prem 1a1fca641d0a: [active_model_serializers] Rendered UserSerializer with ActiveModelSerializers::Adapter::JSON
Jan 13 11:14:46 SSM-On-Prem 1a1fca641d0a: method=GET path=/sessions/get_user format=json controller=SessionsController
Jan 13 11:14:46 SSM-On-Prem 504f06c0d581: 10.110.35.124https://10.110.35.124 - - [13/Jan/2023:11:14:46 +0000] "GET /sessions/get_user HTTP/1.1" 200 1000
Jan 13 11:17:01 SSM-On-Prem 504f06c0d581: 2023/07/13 11:17:01 [error] 47#47: *1576 connect() failed (111: Connection refused) while connecting to upstream, client: 10.110.35.124, server: , request: "GET /sessions/get_user HTTP/1.1", upstream: "http://127.0.0.1:6379/redis", host: "10.110.35.124"
Jan 13 11:17:01 SSM-On-Prem 504f06c0d581: 2023/07/13 11:17:01 [warn] 47#47: *1576 upstream server temporarily disabled while connecting to upstream, client: 10.110.35.124, server: , request: "GET /sessions/get_user HTTP/1.1", upstream: "http://127.0.0.1:6379/redis", host: "10.110.35.124"
Jan 13 11:17:01 SSM-On-Prem 1a1fca641d0a: [active_model_serializers] Rendered ActiveModel::Serializer::JSON
Jan 13 11:17:01 SSM-On-Prem 1a1fca641d0a: method=POST path=/oauth/token format=json controller=Doorkeeper::SessionsController
Jan 13 11:17:01 SSM-On-Prem 504f06c0d581: 10.66.76.85https://10.66.76.85 - - [13/Jan/2023:11:17:01 +0000] "POST /oauth/token HTTP/1.1" 200 1000
Jan 13 11:17:14 SSM-On-Prem 1a1fca641d0a: [INFO] Session expiring outcome=success
```

在vManage License Management ( vManage许可证管理 ) 部分放置智能帐户详细信息时，登录vManage：

```
13-Jan-2023 17:29:02,775 IST INFO [um8_vManage] [SmartLicensingIntegrationManager] (default task-24) |default|
13-Jan-2023 17:29:02,776 IST INFO [um8_vManage] [SmartLicensingIntegrationManager] (default task-24) |default|
13-Jan-2023 17:29:02,780 IST INFO [um8_vManage] [AbstractSettingsManager] (default task-24) |default|
13-Jan-2023 17:29:02,781 IST INFO [um8_vManage] [SmartLicensingUtil] (default task-24) |default| initialia
13-Jan-2023 17:29:02,781 IST INFO [um8_vManage] [SmartLicensingUtil] (default task-24) |default| Getti
13-Jan-2023 17:29:02,793 IST INFO [um8_vManage] [RestAPIClient] (default task-24) |default| RestAPI pro
```

```
13-Jan-2023 17:29:02,793 IST INFO [um8_vManage] [RestAPIClient] (default task-24) |default| RestAPI pr
13-Jan-2023 17:29:02,798 IST INFO [um8_vManage] [SmartLicensingUtil] (default task-24) |default| URL b
13-Jan-2023 17:29:02,798 IST INFO [um8_vManage] [SmartLicensingUtil] (default task-24) |default| Query
13-Jan-2023 17:29:03,490 IST ERROR [um8_vManage] [RestAPIClient] (default task-24) |default| Failed to
13-Jan-2023 17:29:03,491 IST ERROR [um8_vManage] [SmartLicensingUtil] (default task-24) |default| Faile
13-Jan-2023 17:29:03,491 IST ERROR [um8_vManage] [SmartLicensingIntegrationRestfulResource] (default ta
```

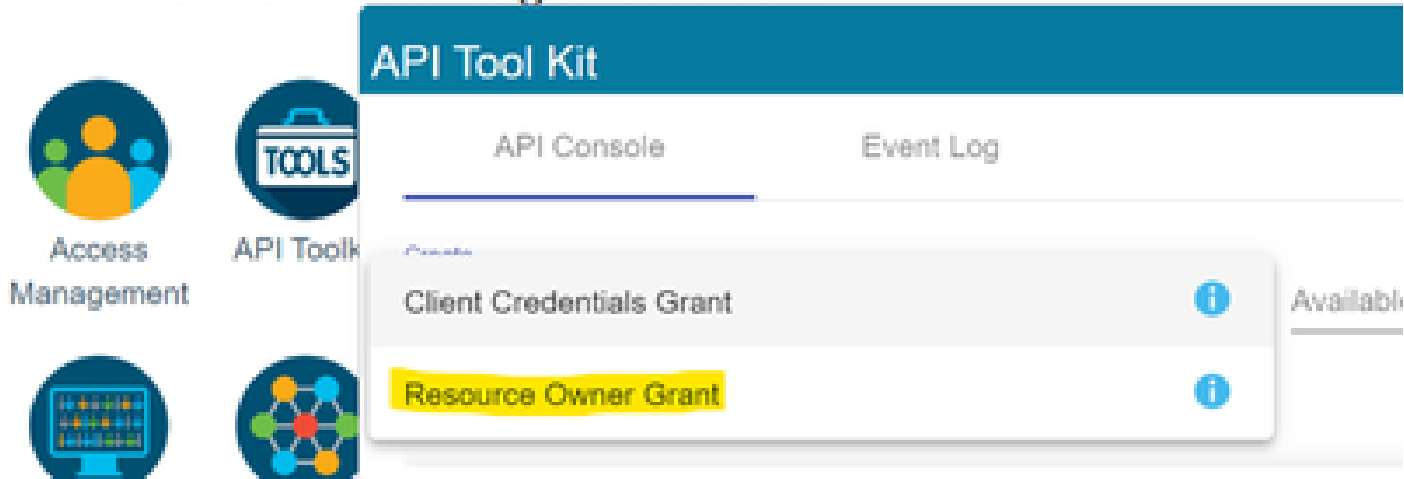
注意：从vManage GUI同步智能帐户时出现错误403，表明服务器理解该请求，但拒绝对其进行授权。

## 解决方法

1. 登录到Prem Server。
2. 导航到API工具包。
3. 选择“Resource Owner Grant”，输入详细信息作为Name并保存。

## On-Prem Admin Workspace

### Smart Software Manager On-Prem



## Resource Owner Grant



Name \*

Test5

Description

Expiration Date

Client ID \*

z92Dss3\_SVnlhUXURJV97gdf03ukxSE5\_shD3vB7tlyl2YKAaJkGh6nbYSRWYCzN

Client Secret \*

.....

Regenerate Client Secret

Save

Cancel

4. 选择保存的记录（在以前的快照中提及）并选中“客户端ID”和“客户端密码”。

## API Tool Kit



API Console

Event Log

Create

Available Actions

API Console Enabled

Search by Name

Showing All Records

<input type="checkbox"/>	Name	Creation Date	Type	Description	Client ID
<input type="checkbox"/>	Test5	Aug 04 2023	Resource Owner Grant		z92Dss3_SVnlhUXU...

5. 在Catalyst SD-WAN Manager门户中共享并输入共享客户端ID和客户端密码。
6. 转至vManage中的“同步许可证和刷新设备”，并使用您登录所使用的内部凭证生成客户端ID和客户端密码。

## 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）。