安全简单网络管理协议

目录

简介

先决条件

- 要求
- 使用的组件

背景信息

保护SNMP的策略

- 避免不安全的SNMP社区字符串
- 配置SNMP视图
- 使用访问控制列表(ACL)配置SNMP社区
- 配置 SNMPv3
  - 步骤1.识别受信任NMS地址
  - 步骤2.为受信任NMS创建ACL
  - 步骤3.创建SNMP视图
  - 步骤4.创建SNMPv3组
  - 步骤5.创建SNMPv3用户
  - 步骤6.（可选）配置SNMPv3陷阱或通知
  - 步骤7.禁用或避免不必要的SNMPv1和SNMPv2c社区
  - 重要说明
  - 验证
- 在接口上配置ACL

如何检查 SNMP 错误

相关信息

简介

本文档介绍如何为简单网络管理协议 (SNMP) 提供保护。

先决条件

要求

本文档没有任何特定的要求。

使用的组件

本文档中的信息基于SNMP功能，如Cisco IOS®软件版本中引入的SNMP视图和SNMP版本3(SNMPv3)。虽然一些配置示例和命令语法基于Cisco IOS®，但核心SNMP安全概念也适用于较新的Cisco IOS XE平台。命令语法、支持的身份验证和隐私算法、平台功能和管理平面保护功能在Cisco IOS和Cisco IOS XE版本之间可能有所不同。

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。

背景信息

保护SNMP非常重要，因为未经授权的SNMP访问可能会泄露设备信息，在启用读写访问时允许更改配置，或导致拒绝服务(DoS)情况。

早期的SNMP版本（如SNMPv1和SNMPv2c）依赖社区字符串进行访问控制。这些社区字符串的功能类似于共享密码，但它们以明文发送，不提供强身份验证或加密。如果社区字符串被重复使用、在配置文件中公开或从网络中捕获，则未经授权的用户可以轮询设备信息，或者在读写社区的情况下修改设备设置。

一些最常见的SNMPv1/v2c安全限制包括：

- 使用众所周知的社区字符串，例如公共或私有
- 社区字符串的明文传输
- 缺少每用户身份验证
- SNMP负载缺乏加密

对于安全部署，请使用支持身份验证和隐私的SNMPv3。SNMPv3提供基于用户的身份验证和加密，有助于保护访问控制和管理数据。此外，您可以限制对受信任管理站的SNMP访问，避免不必要的读写访问，并使用SNMP视图限制可访问的MIB对象。



注意：在生产网络中部署之前，验证特定平台和软件版本的确切命令行界面(CLI)配置、功能支持和验证命令。

保护SNMP的策略

避免不安全的SNMP社区字符串

请勿使用众所周知的社区字符串，如public或private。如果兼容性需要SNMPv1或SNMPv2c，请使用唯一的、非明显的团体字符串，使用ACL限制它们，并避免读写团体，除非有特定的操作要求。



注意：首选带有authPriv的SNMPv3进行安全部署。authPriv级别提供身份验证和加密。身份验证会验证SNMP消息是否来自有效的SNMPv3用户，并且隐私会加密SNMP负载，因此管理数据不会以明文发送。

配置SNMP视图

SNMP视图限制对特定MIB子树的访问。当视图应用于团体或SNMPv3组时，仅包含的MIB对象可访问。未包含的MIB子树被隐式拒绝。您还可以从更广泛的包含子树中显式排除特定子树。

要创建或更新SNMP视图条目，请使用snmp-server view全局配置命令。要删除指定的SNMP服务器视图条目，请使用此命令的no形式。

<#root>

snmp-server view

{included | excluded}

no snmp-server view

语法说明:

- myview-name — 更新或创建的视图记录的标签。名称用于引用记录。
- oid-tree — 要包括在视图中或从视图中排除的抽象语法表示法1(ASN.1)子树的对象标识符(OID)。要标识子树，请指定包含数字（如1.3.6.2.4）或单词（如system）的文本字符串。用星号(*)通配符替换单个子标识符以指定子树系列；例如1.3.*.4。
- included or excluded — 视图类型。必须指定included或excluded。

传统SNMP文档(包括RFC 1447)引入了MIB视图的概念，以控制MIB树的哪些部分包含或排除在SNMP访问中。尽管RFC 1447及其基于参与方的SNMPv2安全模型已过时，但MIB视图概念仍通过基于RFC 3415中定义的基于视图的访问控制模型(VACM)在现代SNMP中保持相关性。Cisco SNMP视图使用此概念来限制对特定MIB子树的访问。SNMP视图可以应用于SNMPv1/v2c社区或SNMPv3组。

此示例将创建一个视图，其中包含MIB-II系统组中除sysServices(System 7)以外的所有对象以及MIB-II接口组中接口1的所有对象：

<#root>

```
snmp-server view agon system included
```

```
snmp-server view agon system.7 excluded
```

```
snmp-server view agon ifEntry.*.1 included
```

在本示例中，该视图允许访问MIB-II和思科专用MIB对象，但排除了atEntry所表示的ARP表:

<#root>

```
snmp-server view myview mib-2 included
```

```
snmp-server view myview atEntry excluded
```

```
snmp-server view myview cisco included
```

```
snmp-server community C1sc03xample view myview RO 11

snmp-server community C1sc03xample2 view myview RW 11

snmp-server contact ciscouser@cisco.com
```

注意：其中一些示例可以引用来自Cisco IOS版本的旧MIB对象，例如atTable或atEntry。在较新的Cisco IOS XE平台上，等效数据可以使用不同的MIB对象或OID。

注意：在部署SNMP视图之前，请验证目标平台和软件版本上是否存在MIB对象名称或OID。使用SNMP对象导航器获取其他信息。

使用访问控制列表(ACL)配置SNMP社区

如果需要SNMPv1或SNMPv2c，请使用ACL将每个社区字符串限制为受信任的网络管理系统(NMS)地址。与SNMP社区关联的ACL控制哪些源IP地址可以使用该社区字符串向设备发送SNMP请求。

SNMP通信通常包括两种不同的流量类型：

- 轮询/请求：NMS向网络设备发送SNMP GET、GETNEXT、GETBULK或SET请求。这些请求通常以受管设备上的UDP端口161为目标。
- 陷阱/通知：网络设备将未经请求的SNMP陷阱或通知消息发送到NMS，以报告事件、警报或状态更改。这些通知通常以NMS上的UDP端口162为目标。

对轮询请求和陷阱通知使用不同的社区字符串。如果发现一个团体字符串，此分离将限制曝光。例如，如果攻击者从通知数据包获取陷阱社区字符串，则该字符串不得也允许SNMP轮询访问设备。

<#root>

access-list 10 permit

<#root>

snmp-server community

RO 10

<#root>

snmp-server host

version 2c

在本例中：

- NMS使用<RO_POLLING_COMMUNITY>轮询设备。
- 当设备向NMS发送陷阱或通知时，会使用<TRAP_NOTIFICATION_COMMUNITY>。
- ACL 10限制轮询访问，因此只有<NMS_IP>可以使用轮询社区字符串。
- 陷阱通知社区字符串不会重复用于SNMP轮询访问。

如果配置了陷阱社区字符串，请验证它不会无意中允许SNMP轮询访问。在某些Cisco IOS软件版本中，除非明确限制或禁用，否则用于陷阱的社区字符串也可可用于SNMP访问。

要阻止使用陷阱社区字符串的轮询访问，请明确拒绝将其用作SNMP社区：

<#root>

```
access-list 99 deny any
```

<#root>

snmp-server community

RO 99

除非有特定的操作要求，否则应避免配置读写社区。如果需要读写访问，请将其限制为专用管理站，使用单独的社区字符串，并应用限制性ACL。首选具有身份验证和隐私的SNMPv3，以实现安全部署。

配置 SNMPv3

将SNMPv3与身份验证和隐私配合使用以实现安全SNMP部署。SNMPv3提供基于用户的身份验证和加密，与SNMPv1和SNMPv2c不同，后者依赖于社区字符串。

对于安全部署，请使用SNMPv3 authPriv安全级别：

- 身份验证验证SNMP消息是否来自有效的SNMPv3用户。
- 隐私对SNMP负载进行加密，因此不会以明文发送管理数据。
- 对于安全生产部署，请避免noAuthNoPriv和authNoPriv。
- 当平台支持较强选项时，请避免较弱或传统算法（例如MD5和DES）。

注意：算法支持因思科平台和软件版本而异。使用设备支持的最强身份验证和隐私算法，例如使用SHA或SHA-2进行身份验证，使用AES进行隐私。

SNMPv3配置步骤

步骤1.识别受信任NMS地址

确定允许哪个网络管理系统轮询设备。

示例值：

- NMS IP地址：<NMS_IP>
- 设备管理IP地址：<DEVICE_MGMT_IP>



注意：必须只允许受信任的管理站访问SNMP服务。

步骤2.为受信任NMS创建ACL

创建一个ACL，该ACL仅允许来自受信任NMS的SNMP访问。

示例：

<#root>

access-list 10 permit

access-list 10 deny any log

此ACL限制哪些源IP地址可以使用SNMPv3组。deny any log条目有助于识别系统日志中的未授权SNMP访问尝试。

步骤3.创建SNMP视图

创建SNMP视图以限制SNMPv3用户可以访问的MIB对象。

示例：

<#root>

snmp-server view

mib-2 included

<#root>

snmp-server view

cisco included

此示例允许访问MIB-II子树和思科专用MIB对象。如果需要更严格的视图，请明确排除敏感或不必要的MIB子树。

示例：

<#root>

snmp-server view

excluded

注意：SNMP视图通过允许NMS仅访问监控所需的MIB对象来减少暴露情况。

步骤4.创建SNMPv3组

创建一个SNMPv3组，该组使用priv安全级别并引用SNMP视图和ACL。

示例：

```
<#root>

snmp-server group

    v3 priv read

     access 10
```

此命令执行三个重要操作：

- v3指定SNMPv3。
- priv需要身份验证和加密。
- read <VIEW_NAME>使用以前定义的SNMP视图限制可读MIB对象。
- 访问10使用定义的ACL限制对受信任NMS源IP地址的SNMP访问。

注意：对于大多数监控使用情形，只读访问通常就足够了。仅在具有特定操作要求时才配置写访问。

步骤5.创建SNMPv3用户

创建SNMPv3用户并将其分配给SNMPv3组。

示例：

```
<#root>

snmp-server user

      v3 auth sha

       priv aes 256
```

此命令使用以下命令创建用户：

- SNMPv3身份验证
- SHA身份验证（如果支持）
- 支持256位加密的AES
- 受限制的SNMPv3组中的成员资格

注意：使用强、唯一的密码进行身份验证和保密。不要对这两个值重复使用相同的密码。

注意：如果较新的平台支持更强大的选项（例如SHA-2或更强的AES变体），请使用设备和NMS支持的最强选项。

步骤6.（可选）配置SNMPv3陷阱或通知

如果设备需要向NMS发送通知，请使用SNMPv3配置SNMP主机。

示例：

```
<#root>

snmp-server host

     version 3 priv
```

陷阱和通知是从设备发送到NMS的通知。它们不同于轮询请求，NMS在轮询请求中向设备发送SNMP查询。使用SNMPv3通知，这样陷阱或通知流量也能受益于身份验证和隐私。

如果需要，还要配置陷阱源接口：

```
<#root>

snmp-server trap-source
```

注意：使用一致的陷阱源可帮助NMS识别设备并允许防火墙或ACL匹配可预测的源地址。

步骤7.禁用或避免不必要的SNMPv1和SNMPv2c社区

如果不需要SNMPv1或SNMPv2c，请勿配置社区字符串或删除未使用的社区字符串。

示例：

<#root>

no snmp-server community

注意：这降低了通过明文社区字符串进行未授权SNMP访问的风险。

下一个配置片段显示一个示例，该示例仅允许来自NMS 10.189.5.161的SNMPv3轮询，使用SNMP视图限制可读MIB对象，并要求身份验证和加密。

<#root>

Router#configure terminal
Router(config)#

access-list 10 permit 10.189.5.161

Router(config)#

access-list 10 deny any log

Router(config)#!
Router(config)#

snmp-server view ciscoview mib-2 included

Router(config)#

snmp-server view ciscoview cisco included

```
Router(config)#

snmp-server view ciscoview ipNetToMediaEntry.2 excluded

Router(config)#

snmp-server group myciscogroup v3 priv read ciscoview access 10


Router(config)#

snmp-server user ciscouser myciscogroup v3 auth sha C1sc03xampl3 priv aes 256 C1sc03xampl32
```

重要说明

- 将SNMP视图分配到SNMPv3组之前对其进行配置。
- 如果手动配置了本地SNMP引擎ID，请在创建SNMPv3用户之前配置它。在用户创建后更改引擎ID可能需要重新创建SNMPv3用户。
- 出于安全原因，SNMPv3用户信息没有完全显示在show running-config中。使用show snmp user进行验证。
- 确切的MIB对象语法和支持的算法因Cisco IOS和Cisco IOS XE版本而异。在部署之前验证目标平台的命令语法。

验证

使用以下命令验证本地SNMP配置：

show snmp user
show snmp group
show snmp view
show snmp
show ip access-lists
show running-config | section snmp

<#root>

ISP-C#

show snmp user

```
User name: ciscouser
Engine ID: 80000009030000A38EB6F800
storage-type: nonvolatile        active
```

Authentication Protocol:

SHA

Privacy Protocol:

AES256

Group-name: myciscogroup

```
ISP-C#show snmp group
groupname: ILMI                                 security model:v1
contextname:




              storage-type: permanent
readview : *ilmi                                writeview: *ilmi
notifyview:




row status: active

groupname: ILMI                                 security model:v2c
contextname:




                storage-type: permanent
readview : *ilmi                                writeview: *ilmi
notifyview:




row status: active

groupname: myciscogroup                         security model:v3 priv
contextname:




                  storage-type: nonvolatile
readview : ciscoview                            writeview:
```

```
notifyview:




row status: active      access-list: 10

ISP-C#
```

show snmp view

```
*ilmi system - included permanent active
*ilmi atmForumUni - included permanent active
cac_view pimMIB - included read-only active
cac_view msdpMIB - included read-only active
cac_view system - included read-only active
cac_view interfaces - included read-only active
cac_view ip - included read-only active
cac_view ospf - included read-only active
cac_view bgp - included read-only active
cac_view ifMIB - included read-only active
cac_view nhrpMIB - included read-only active
cac_view ipMRouteStdMIB - included read-only active
cac_view igmpStdMIB - included read-only active
cac_view pimStdMIB - included read-only active
cac_view mgmdStdMIB - included read-only active
cac_view ospfv3MIB - included read-only active
cac_view ipForward - included read-only active
cac_view ipTrafficStats - included read-only active
cac_view ospfTrap - included read-only active
cac_view sysUpTime.0 - included read-only active
cac_view ciscoPingMIB - included read-only active
cac_view ciscoPimMIB - included read-only active
cac_view ciscoBgp4MIB - included read-only active
cac_view ciscoIfExtensionMIB - included read-only active
cac_view ciscoEigrpMIB - included read-only active
cac_view ciscoCefMIB - included read-only active
cac_view ciscoNhrpExtMIB - included read-only active
cac_view ciscoIpMRouteMIB - included read-only active
cac_view ciscoDlcSwitchMIB - included read-only active
cac_view ciscoExperiment.101 - included read-only active
cac_view ciscoIetfIsisMIB - included read-only active
cac_view ifIndex - included read-only active
cac_view ifDescr - included read-only active
cac_view ifType - included read-only active
cac_view ifAdminStatus - included read-only active
cac_view ifOperStatus - included read-only active
```

```
cac_view snmpTraps.3 - included read-only active
cac_view snmpTraps.4 - included read-only active
cac_view snmpTrapOID.0 - included read-only active
cac_view snmpMIB.1.4.3.0 - included read-only active
cac_view lifEntry.20 - included read-only active
cac_view cciDescriptionEntry.1 - included read-only active
ciscoview mib-2 - included nonvolatile active
ciscoview cisco - included nonvolatile active
ciscoview ipNetToMediaEntry.2 - excluded nonvolatile active
v1default iso - included permanent active
v1default internet - included permanent active
v1default snmpUsmMIB - excluded permanent active
v1default snmpVacmMIB - excluded permanent active
v1default snmpCommunityMIB - excluded permanent active

ISP-C#
```

show ip access-lists

```
Standard IP access list 10
    10 permit 10.189.5.161 (26 matches)
    20 deny   any log (6 matches)
Extended IP access list meraki-fqdn-dns

ISP-C#
```

show running-config | section snmp

```
snmp-server group myciscogroup v3 priv read ciscoview access 10
snmp-server view ciscoview mib-2 included
snmp-server view ciscoview cisco included
snmp-server view ciscoview ipNetToMediaEntry.2 excluded
```

您还可以对MIB-II系统树执行SNMP遍历：

<#root>

ciscouser@Desktop %

snmpwalk -v3 -l authPriv -u ciscouser -a SHA -A 'C1sc03xampl3' -x AES -X 'C1sc03xampl32' 10.207.214.25 s

```
SNMPv2-MIB::sysDescr.0 = STRING: Cisco IOS Software [Cupertino], ASR1000 Software (X86_64_LINUX_IOSD-UN
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2023 by Cisco Systems, Inc.
Compiled Wed 26-Jul-23 10:29 b
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::enterprises.9.1.2348
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (107487239) 12 days, 10:34:32.39
SNMPv2-MIB::sysContact.0 = STRING:
SNMPv2-MIB::sysName.0 = STRING: ISP-C
SNMPv2-MIB::sysLocation.0 = STRING:
SNMPv2-MIB::sysServices.0 = INTEGER: 78
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (0) 0:00:00.00
```

下一个命令和输出显示无法获取ipNetToMediaEntry.2组：

<#root>

ciscouser@Desktop %

snmpwalk -v3 -l authPriv -u ciscouser -a SHA -A 'C1sc03xampl3' -x AES -X 'C1sc03xampl32' 10.207.214.25 i

```
IP-MIB::ipNetToMediaPhysAddress = No Such Object available on this agent at this OID
```

下一个命令和输出结果用于Cisco本地系统组:

<#root>

ciscouser@Desktop %

snmpwalk -v3 -l authPriv -u ciscouser -a SHA -A 'C1sc03xampl3' -x AES -X 'C1sc03xampl32' 10.207.214.25 .

```
SNMPv2-SMI::enterprises.9.2.1.1.0 = STRING: "
17.15(4r)
"
SNMPv2-SMI::enterprises.9.2.1.2.0 = STRING: "LocalSoft"
SNMPv2-SMI::enterprises.9.2.1.3.0 = STRING: "ISP-C"
SNMPv2-SMI::enterprises.9.2.1.4.0 = ""
```

SNMPv2-SMI::enterprises.9.2.1.5.0 = IpAddress: 10.189.5.161

```
SNMPv2-SMI::enterprises.9.2.1.6.0 = IpAddress: 0.0.0.0
SNMPv2-SMI::enterprises.9.2.1.8.0 = INTEGER: -796648892
SNMPv2-SMI::enterprises.9.2.1.9.0 = INTEGER: 1646
SNMPv2-SMI::enterprises.9.2.1.10.0 = INTEGER: 1000
SNMPv2-SMI::enterprises.9.2.1.11.0 = INTEGER: 1557645
SNMPv2-SMI::enterprises.9.2.1.12.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.13.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.14.0 = INTEGER: 104
SNMPv2-SMI::enterprises.9.2.1.15.0 = INTEGER: 1200
```

```
SNMPv2-SMI::enterprises.9.2.1.16.0 = INTEGER: 1200
SNMPv2-SMI::enterprises.9.2.1.17.0 = INTEGER: 2500
SNMPv2-SMI::enterprises.9.2.1.18.0 = INTEGER: 2684018
SNMPv2-SMI::enterprises.9.2.1.19.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.20.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.21.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.22.0 = INTEGER: 600
SNMPv2-SMI::enterprises.9.2.1.23.0 = INTEGER: 900
SNMPv2-SMI::enterprises.9.2.1.24.0 = INTEGER: 900
SNMPv2-SMI::enterprises.9.2.1.25.0 = INTEGER: 2000
SNMPv2-SMI::enterprises.9.2.1.26.0 = INTEGER: 26757370
SNMPv2-SMI::enterprises.9.2.1.27.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.28.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.29.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.30.0 = INTEGER: 1536
SNMPv2-SMI::enterprises.9.2.1.31.0 = INTEGER: 900
SNMPv2-SMI::enterprises.9.2.1.32.0 = INTEGER: 899
SNMPv2-SMI::enterprises.9.2.1.33.0 = INTEGER: 1800
SNMPv2-SMI::enterprises.9.2.1.34.0 = INTEGER: 8891131
SNMPv2-SMI::enterprises.9.2.1.35.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.36.0 = INTEGER: 1
SNMPv2-SMI::enterprises.9.2.1.37.0 = INTEGER: 1
SNMPv2-SMI::enterprises.9.2.1.38.0 = INTEGER: 5024
SNMPv2-SMI::enterprises.9.2.1.39.0 = INTEGER: 100
SNMPv2-SMI::enterprises.9.2.1.40.0 = INTEGER: 100
SNMPv2-SMI::enterprises.9.2.1.41.0 = INTEGER: 300
SNMPv2-SMI::enterprises.9.2.1.42.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.43.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.44.0 = INTEGER: 1
SNMPv2-SMI::enterprises.9.2.1.45.0 = INTEGER: 1
SNMPv2-SMI::enterprises.9.2.1.46.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.47.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.48.0 = IpAddress: 0.0.0.0
SNMPv2-SMI::enterprises.9.2.1.49.0 = ""
SNMPv2-SMI::enterprises.9.2.1.51.0 = IpAddress: 0.0.0.0
SNMPv2-SMI::enterprises.9.2.1.52.0 = ""
SNMPv2-SMI::enterprises.9.2.1.56.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.57.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.58.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.59.0 = INTEGER: 4972
SNMPv2-SMI::enterprises.9.2.1.60.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.61.0 = STRING: "cisco Systems, Inc.
170 West Tasman Dr.
San Jose, CA  95134-1706
U.S.A.
Ph +1-408-526-4000
Customer service 1-800-553-6387 or +1-408-526-7208
24HR Emergency 1-800-553-2447 or +1-408-526-7209
Email Address tac@cisco.com
World Wide Web http://www.cisco.com"
SNMPv2-SMI::enterprises.9.2.1.62.0 = INTEGER: 18024
SNMPv2-SMI::enterprises.9.2.1.63.0 = INTEGER: 20
SNMPv2-SMI::enterprises.9.2.1.64.0 = INTEGER: 20
SNMPv2-SMI::enterprises.9.2.1.65.0 = INTEGER: 33
SNMPv2-SMI::enterprises.9.2.1.66.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.67.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.68.0 = INTEGER: 1
SNMPv2-SMI::enterprises.9.2.1.69.0 = INTEGER: 1
SNMPv2-SMI::enterprises.9.2.1.70.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.71.0 = INTEGER: 0
SNMPv2-SMI::enterprises.9.2.1.72.0 = IpAddress: 0.0.0.0
SNMPv2-SMI::enterprises.9.2.1.73.0 = STRING: "bootflash:asr1000-universalk9.17.09.04.SPA.bin"
```

```
SNMPv2-SMI::enterprises.9.2.1.74.0 = INTEGER: 3
```

在接口上配置ACL

接口ACL可用作额外的保护层，以限制哪些主机可以向设备接口发送SNMP流量。对于不支持更有针对性的控制平面保护功能(例如控制平面策略(CoPP)或管理平面ACL)的平台，它们尤其有用。

接口ACL可以在未经授权的SNMP数据包被设备处理之前将其阻止，从而减少不需要的轮询、暴力社区字符串尝试或来自不受信任来源的过量SNMP流量的风险。

⚠ 警告：应用接口ACL时请小心。与ip access-group一起应用的ACL会过滤选定方向的流量，并且可能会影响发往路由器的流量和中转流量。在生产部署之前，始终包含所有必需的permit语句并在维护窗口或实验室中验证ACL。

<#root>

ip access-list extended SNMP-MGMT-IN permit udp host

host

eq snmp deny udp any host

eq snmp log permit ip any any

然后，必须使用以下配置命令将ACL应用到所有接口：

<#root>

interface

ip access-group SNMP-MGMT-IN in

在本例中：

- 允许从受信任NMS到设备管理IP地址的SNMP轮询。
- 拒绝并记录从其他源到设备管理IP地址的SNMP轮询。
- 最后的permit ip any any语句允许其他IP流量。

注意：您可以根据网络设计调整此示例。如果该平台支持CoPP、iACL或管理平面ACL，则这些选项可以提供更有针对性的控制平面保护。对于配置了许多IP地址的设备或需要与路由器通信的许多主机，此解决方案并非总是可扩展的。保护您的核心：基础设施保护访问控制列表文档介绍了ACL的准则和推荐的部署技术。

如何检查 SNMP 错误

使用log关键字配置SNMP 社区ACL。监控系统日志的失败尝试，如图所示。

<#root>

```
access-list 10 deny any log
snmp-server community
```

RO 10

当有人尝试通过社区公共访问路由器时，您会看到syslog如下内容：

<#root>

%SEC-6-IPACCESSLOGS: list 10 denied 172.16.1.15 packet

此输出意味着access-list 10已拒绝来自主机172.16.1.15的五个SNMP数据包。

使用show snmp命令定期检查SNMP是否存在错误，如下所示：

<#root>

ISP-C#

show snmp

```
Chassis: TKM21110169
186 SNMP packets input
 0 Bad SNMP version errors 58 Unknown community name
    0 Illegal operation for community name supplied
 0 Encoding errors
    115 Number of requested variables
    0 Number of altered variables
    2 Get-request PDUs
    113 Get-next PDUs
    0 Set-request PDUs
    0 Input queue packet drops (Maximum queue size 1000)
126 SNMP packets output
    0 Too big errors (Maximum packet size 1500)
    0 No such name errors
    0 Bad values errors
    0 General errors
    0 Response PDUs
    0 Trap PDUs
Packets currently in SNMP process input queue: 0
SNMP global trap: disabled

SNMP logging: disabled
```

请观察突出显示的计数器，了解可能表示尝试利用这些漏洞的错误率意外增加的情况。要报告任何安全问题，请参阅思科产品安全事件响应。

相关信息

- 思科安全公告SNMP漏洞
- 思科技术支持和下载

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）。