了解 SNMP 中的表索引值

目录

简介

轮询简单网络管理协议(SNMP)对象时，您有时必须确切知道所轮询的内容。要充分理解这一点，您需要了解如何将被轮询的对象与要轮询的对象相关联。本文档介绍如何使用SNMP中的索引将对象分组到表中的基础知识。

先决条件

要求

本文档的读者应掌握以下这些主题的相关知识：

- SNMP的一般知识
- 用于通过SNMP查询思科设备的软件

使用的组件

本文档中的信息基于以下软件和硬件版本：

- UCD SNMP版本4.2
- 带Cisco IOS®软件版本5.5(7)的Cisco Catalyst 5509

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。

规则

有关文件规则的更多信息请参见“ Cisco技术提示规则”。

开始 ifIndex

当您处理SNMP时，首先要学习的内容之一是ifIndex。这是所有对象的主键。假设所有接口（物理和逻辑）都已分解并分配了值。此值在设备启动期间分配，并且不能更改。如果需要轮询该特定接口的任何信息，它必须使用该指定值。

IfIndex在IF-MIB(RFC 1213)中以 下方式定义：

```
InterfaceIndex ::= TEXTUAL-CONVENTION
       DISPLAY-HINT "d"
       STATUS       current
       DESCRIPTION
               "A unique value, greater than zero, for each interface
               or interface sub-layer in the managed system.  It is
               recommended that values are assigned contiguously
               starting from 1.  The value for each interface sub-
               layer must remain constant at least from one re-
               initialization of the entity's network management
               system to the next re-initialization."
       SYNTAX       Integer32 (1..2147483647)
```

对于任何MIB，快速了解组织表的索引的方法是查看表条目：

```
ifEntry OBJECT-TYPE
         SYNTAX      IfEntry
         MAX-ACCESS  not-accessible
         STATUS      current
         DESCRIPTION
             "An entry containing management information applicable
             to a particular interface."
         INDEX   { ifIndex }
        ::= { ifTable 1 }
```

给定MIB和表条目，可确定如何对表进行索引。下一节提供ifIndex的示例。

轮询对象

基于 ifIndex 轮询对象

当您发出**snmpwalk**命令轮询交换机上端口7/4的基于ifIndex的对象(ifName)时，您会得到以下输出：

```
sj-cse-568: snmpwalk 172.16.99.60 public ifname

ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.1 = sc0
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.2 = sl0
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.3 = VLAN-1
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.4 = VLAN-1002
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.5 = VLAN-1004
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.6 = VLAN-1005
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.7 = VLAN-1003
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.8 = 7/1
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.9 = 7/2
```

```
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.10 = 7/3
!--- This is the relevant line: ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.11 = 7/4
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.12 = 7/5
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.13 = 7/6
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.14 = 7/7
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.15 = 7/8
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.16 = 7/9
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.17 = 7/10
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.18 = 7/11
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.19 = 7/12
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.20 = ATM8/0
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.22 = /A
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.23 = /B
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.24 = Nu0
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.25 = LEC/ATM8/0.10
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.532 = 3/1
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.533 = 3/2
!--- Output suppressed.
```

在对ifName(路由器上的ifDescr)的轮询的输出中，请注意，每行后面都附加了一个。这是分配给同一行中实际接口的ifIndex。这意味着轮询的第二行（端口7/4）分配了ifIndex 11。如果要从ifIndexed对象获取端口7/4的信息，请使用索引11。这意味着在MIB对象标识符(OID)的末尾添加 `.11`，以检索与相同ifIndex对应的对象实例值。

ifIndex 未为表创建索引或者创建的是交叉索引时轮询对象

有时，表不会通过ifIndex进行索引，例如与BRIDGE-MIB一起进行索引。此输出将检查其索引方式：

```
dot1dBasePortEntry OBJECT-TYPE
        SYNTAX  Dot1dBasePortEntry
        ACCESS  not-accessible
        STATUS  mandatory
        DESCRIPTION
                "A list of information for each port of the
                bridge."
        REFERENCE
                "IEEE 802.1D-1990: Section 6.4.2, 6.6.1"
        INDEX  { dot1dBasePort }
        ::= { dot1dBasePortTable 1 }
```

该输出显示dot1dBasePortEntry由dot1dBasePort编制索引。这如何转换回ifIndex?BRIDGE-MIB访问名为dot1dBasePortIfIndex的对象。对象的定义方式如下：

```
dot1dBasePortIfIndex OBJECT-TYPE
        SYNTAX  INTEGER
        ACCESS  read-only
        STATUS  mandatory
        DESCRIPTION
                "The value of the instance of the ifIndex object,
                defined in MIB-II, for the interface corresponding
                to this port."
        ::= { dot1dBasePortEntry 2 }
```

该输出显示如何从BRIDGE-MIB关联到IF-MIB。下一个示例展示了它如何结合在一起。

注意：BRIDGE-MIB是按VLAN构建的，因此社区“public@*vlan-id*”必须用于非VLAN1环境。

将 BRIDGE-MIB 关联到 IF-MIB

在BRIDGE-MIB上发出**snmpwalk**时，您将获得索引的下一个示例输出。使用dot1dBasePortIfIndex(.1.3.6.1.2.1.17.1.4.1.2)将其映射回ifIndex。获得ifIndex后，使用它根据ifIndex轮询其他对象。

```
sj-cse-568: snmpwalk 172.16.99.60 public .1.3.6.1.2.1.17.1.4.1.2

17.1.4.1.2.203 = 671
17.1.4.1.2.204 = 672
17.1.4.1.2.205 = 673
17.1.4.1.2.206 = 674
17.1.4.1.2.207 = 675
17.1.4.1.2.208 = 676
17.1.4.1.2.209 = 677
17.1.4.1.2.210 = 678
17.1.4.1.2.211 = 679
17.1.4.1.2.212 = 680
17.1.4.1.2.213 = 681
17.1.4.1.2.214 = 682
17.1.4.1.2.215 = 683
17.1.4.1.2.216 = 684
17.1.4.1.2.257 = 581
17.1.4.1.2.385 = 8
17.1.4.1.2.386 = 9
17.1.4.1.2.387 = 10
17.1.4.1.2.388 = 11
17.1.4.1.2.389 = 12
17.1.4.1.2.390 = 13
17.1.4.1.2.391 = 14
17.1.4.1.2.392 = 15
17.1.4.1.2.393 = 16
17.1.4.1.2.394 = 17
17.1.4.1.2.395 = 18
17.1.4.1.2.396 = 19
17.1.4.1.2.449 = 22
```

粗体文本行(`17.1.4.1.2.388 = 11`)显示`.388`是索引。因为您从BRIDGE-MIB轮询了dot1dBasePortIfIndex对象，所以。388是dot1dBasePortIfIndex。输出的11实际上是ifIndex。如果从此轮询和上一次轮询收集信息，可确定端口7/4的ifIndex为11,dot1dBasePortIfIndex（BRIDGE-MIB的索引）为。388。

相关信息

- 技术支持 - Cisco Systems