

网络运维管理



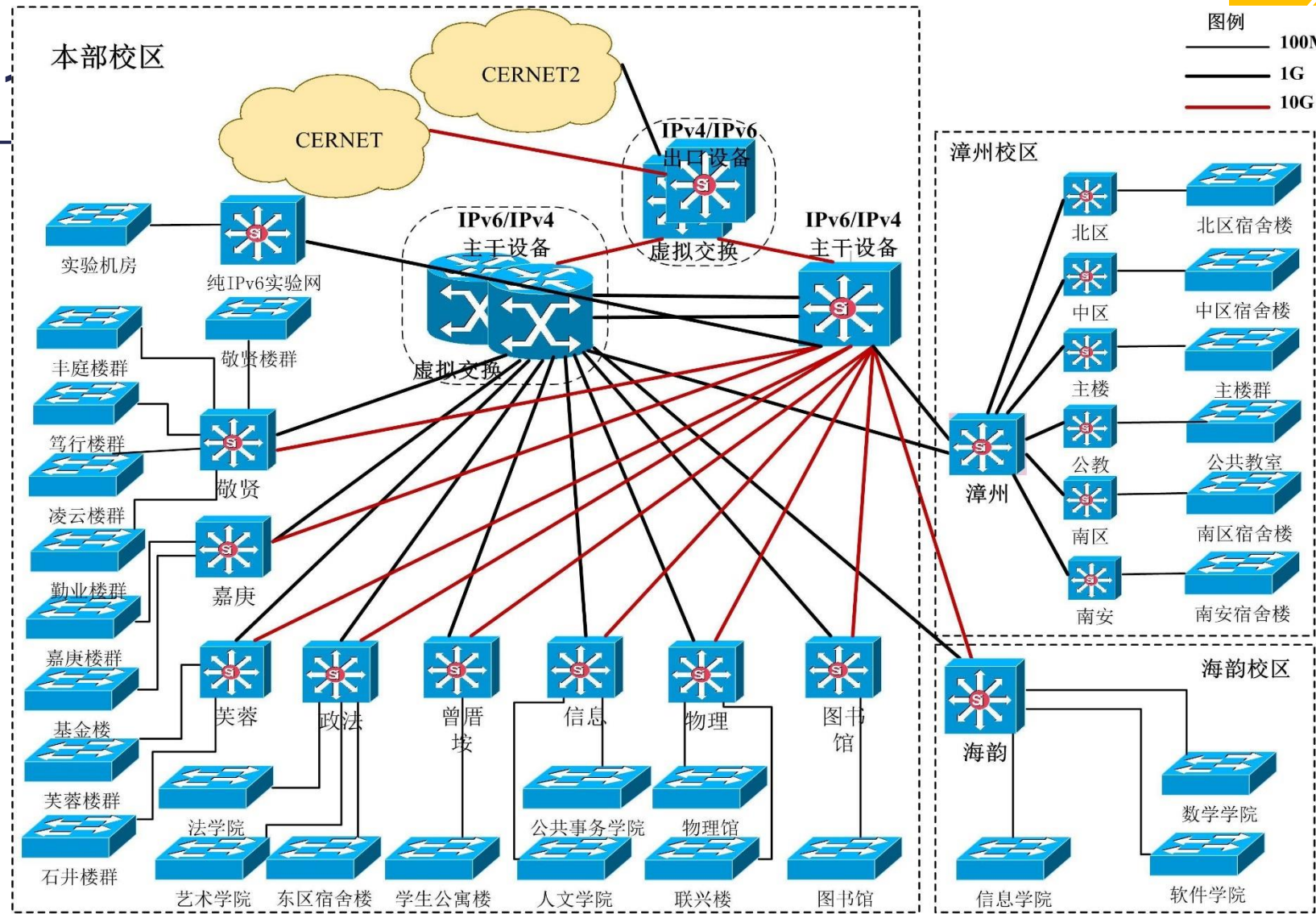
第6讲：网络监控管理

网络监控管理

- SNMP的应用
- 网络监控



图例
— 100M
— 1G
— 10G 述



SNMP的应用

- 每一个网络管理和运维人员都希望能够及时准确地掌握网络的运行状况，例如服务器是否宕机、服务器CPU的使用率有多大、网络中各种协议的流量情况等等，从而实现网络安全稳定地运行。
- 早期的计算机网络规模小，结构简单，网络管理活动也相对简单，但随着计算机网络技术的迅速发展，网络规模日益庞大，结构也越来越复杂。简单、粗陋的管理方式已经不再适应现代的计算机网络，网络管理必须向高度集中和高度智能化的方向发展。

SNMP的应用

- 能否设计一种体系，使管理员能够远程监控到网络中各种设备（服务器、路由器、交换机、防火墙等）当前的状态信息，例如，服务器CPU的利用率、硬盘的使用量、网络接口入分组流量和出分组流量、网络接口的数目等等，则可以大大提高管理效果、降低管理成本。

SNMP的应用

- 这个体系应包含哪些内容？
- 1、网络中有很多被管设备（例如主机、路由器、交换机等），被管理的对象有哪些？（被管理的内容）
- 2、被管对象的结构该如何规定？例如命名规则？如何编码？（内容的表现形式）
- 3、如何对被管对象进行操作？（管理方式）

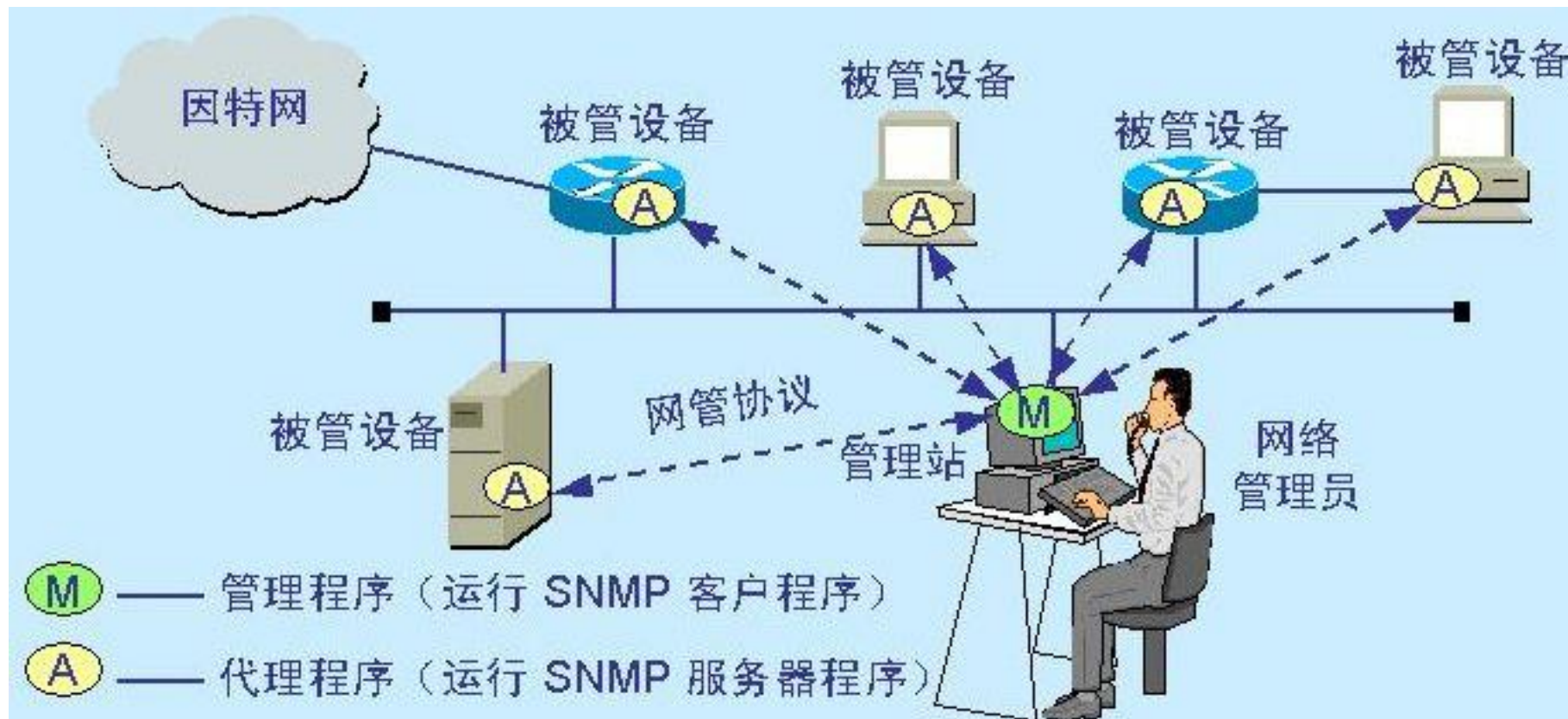
SNMP的应用

□ SNMP的定义：

- SNMP (Simple Network Management Protocol) 为简单网络管理协议。
- 从狭义上讲，是一种专门用于网络管理软件和网络设备之间通信的协议；从广义上讲，是一组为实现网络的自动化管理任务而制定的一系列通用标准，包括管理信息的表示与命名、通信协议等内容。
- 一种应用层协议，使用TCP/IP协议族对互联网上的设备进行管理的框架，它提供一组基本的操作，用来监控和管理网络。

SNMP网络管理模型构件

1.1 SNMP概述



SNMP的应用

1.1 SNMP概述

- 关于网络管理有一个基本原理，就是：

若要管理某个对象，就必然会给该对象添加一些软件或硬件，但这种“添加”必须对原有对象影响尽量小些。

- SNMP正是按照这样的基本原理来设计的。

SNMP的应用

□ SNMP工作的基本思路：

- SNMP中的管理程序和代理程序按照C/S方式工作。
- 管理程序运行SNMP客户程序，而代理程序运行SNMP服务器程序。
- 在被管对象上运行的SNMP服务器程序不停地监听来自管理站的SNMP客户程序的请求，一旦发现，就立即返回管理站所需的信息，或执行某个动作（例如，把某个参数的设置进行更新）。
- 在网管系统中往往是一个（或少数几个）客户程序与很多的服务器程序进行交互。

SNMP的应用

□ SNMP的网络管理由三个部分组成

■ 管理信息结构SMI (Structure of Management Information)

定义对象命名和定义对象类型的通用规则，以及把对象和对象的值进行编码的规则。

■ 管理信息库MIB (Management Information Base)

在被管理的实体中创建命名对象，并按照SMI的规则规定了其类型。

■ SNMP本身

实现对被管对象信息（变量名及其值）的操作，即负责读取和改变这些数值。

SNMP的应用

□ SMI的功能有三个

- 被管对象应该怎样命名（不是命名结果，而是指命名规则）；
- 用来存储被管对象的数据类型有哪些种；
- 在网络上传送的管理数据应如何编码；

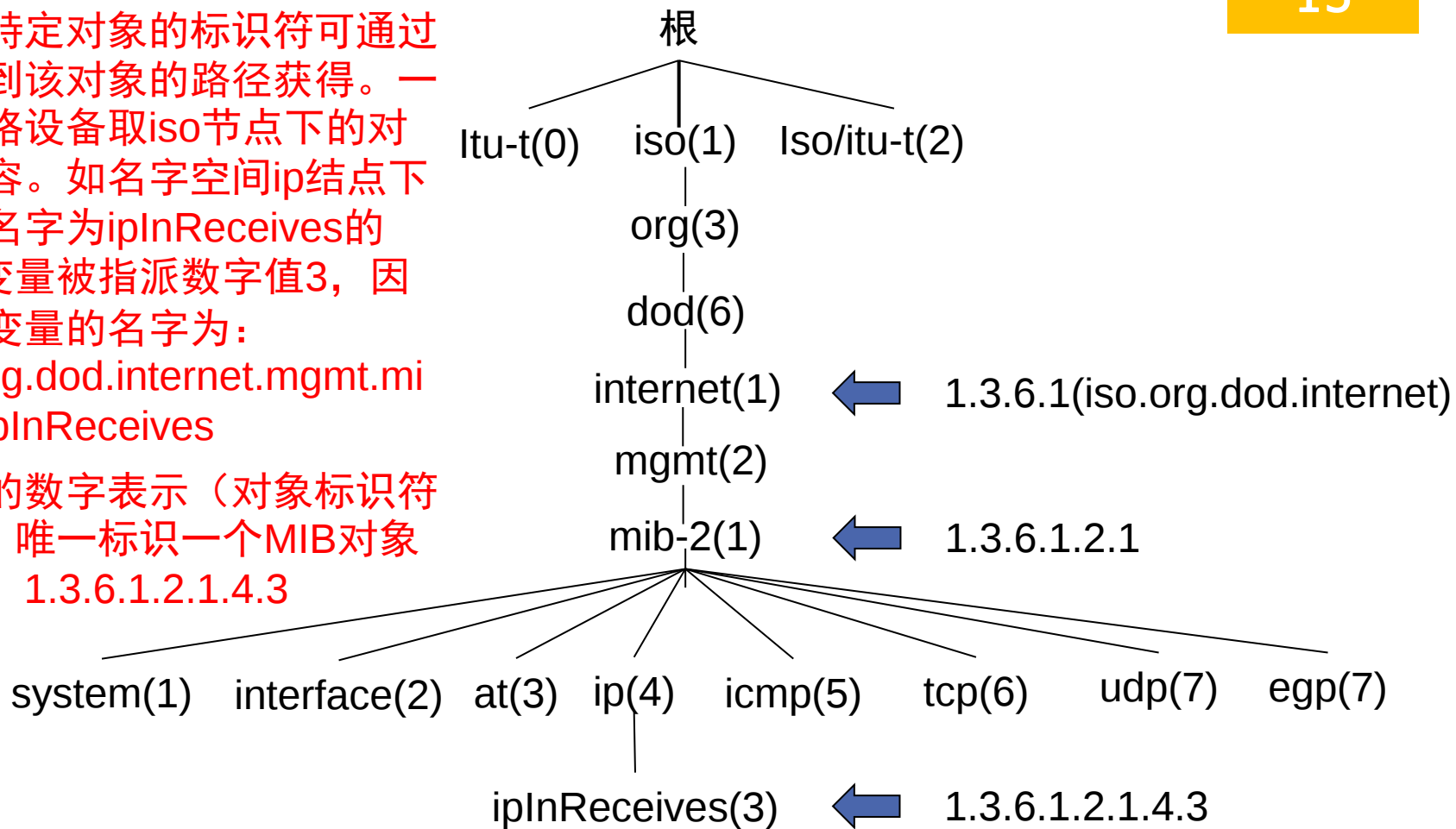
SNMP的应用

- ❑ SMI规定，所有的被管对象都必须处在对象命名树上。
- ❑ 变量使用的名字取自ISO和ITU管理的对象标识符（object identifier, 简称OID）名字空间。它是一种分级树的结构。第一级有三个节点：itu、iso、iso/itu。低级的对象ID分别由相关组织分配。
- ❑ 一个特定对象的标识符可通过由根到该对象的路径获得。一般网络设备取iso节点下的对象内容。如名字空间ip结点下一个名字为ipInReceives的MIB变量被指派数字值3，因而该变量的名字为：
iso.org.dod.internet.mgmt.mib.ip.ipInReceives
- ❑ 相应的数字表示（对象标识符OID，唯一标识一个MIB对象）为：
1.3.6.1.2.1.4.3

一个特定对象的标识符可通过由根到该对象的路径获得。一般网络设备取iso节点下的对象内容。如名字空间ip结点下一个名字为ipInReceives的MIB变量被指派数字值3，因而该变量的名字为：

iso.org.dod.internet.mgmt.mib.ip.inReceives

相应的数字表示（对象标识符OID，唯一标识一个MIB对象）为：1.3.6.1.2.1.4.3



SNMP的应用

- SMI还规定了被管对象中，被管信息的数据类型。

类 型	大 小	说 明
INTEGER	4 字节	在 -2^{31} 到 $2^{31}-1$ 之间的整数
Integer32	4 字节	和 INTEGER 相同
Unsigned32	4 字节	在 0 到 $2^{32}-1$ 之间的无符号数
OCTET STRING	可变	不超过 65535 字节长的字节串
OBJECT IDENTIFIER	可变	对象标识符
IPAddress	4 字节	由 4 个整数组成的 IP 地址
Counter32	4 字节	可从 0 增加到 2^{32} 的整数；当它到达最大值时就返回到 0
TimeTicks	4 字节	记录时间的计数值，以 1/100 秒为单位
BITS	—	比特串
Opaque	可变	不解释的串

SNMP的应用

▣ 管理信息库MIB

- 任何一个被管理的资源都表示成一个对象，称为被管理的对象。MIB是被管理对象的集合。它定义了被管理对象的一系列属性：对象的名称、对象的访问权限和对象的数据类型等。
- 这种定义是依据SMI制定的规则。

SNMP的应用

- ❑ MIB中所有的被管理对象按照所表示的管理信息的不同，被分为不同组。例如，MIB-II中所有和系统有关对象分在system组中；所有和IP有关对象分在ip组中，每个对象组分配一个OID节点，这个节点属于辅助节点。
- ❑ 定义在组中的每个对象，其对象标识符均以组节点的对象标识符作为前缀。

MIB类别所包含的相关信息

MIB类别	包含的相关信息	MIB类别	包含的相关信息
system	被管理对象（如主机、路由器等设备）系统的总体信息	interface	各个网络接口的相关信息
at	地址转换（如：ARP映射）的相关信息	icmp	ICMP协议的实现和运行相关信息
tcp	TCP协议的实现和运行相关信息	udp	UDP协议的实现和运行相关信息
egp	外部网关协议实现和运行相关信息	snmp	描述了SNMP协议自身的一些信息
transmission	根据网络接口，描述相关的管理信息	ip	IP协议的实现和运行相关信息
dot1Bridge	网络中网桥的相关管理信息	host	主机自身上运行的相关信息

MIB变量及类别、含义对比表

MIB对象	类别	含义
sysUpTime	system	系统开启时间
ifNumber	interface	网络接口数
ifMtu	interface	某特定接口的MTU值
icmpInEchos	icmp	接受ICMP发送请求数目
tcpInSegs	tcp	已收到的TCP报文段数目

SNMP的应用

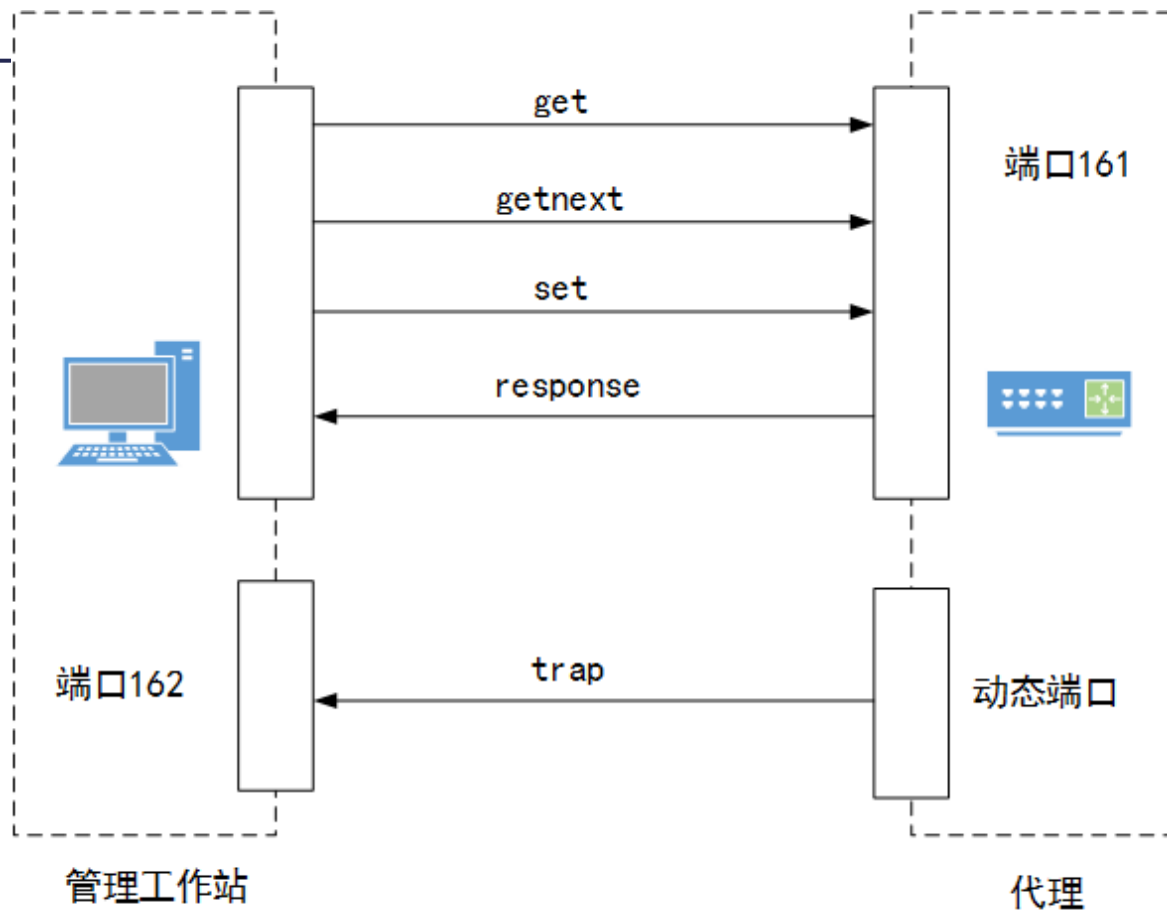
□ SNMP中规定的操作：

- SNMP是管理工作站与代理之间进行数据交互的通信协议。
- 通信协议为完成不同的功能操作，通常是由不同的PDU（协议数据单元）来实现。
- SNMPv1仅规定了5种操作，分别是get、get-next、set、response和trap，在SNMPv2中，增加了get-bulk以及inform两种操作。

SNMP的应用

- SNMPv1的不同操作含义：
 - **get-request操作**：从代理进程处提取一个或多个参数值（从一个具体变量中取出一个值）。
 - **get-next-request操作**：从代理进程处提取一个或多个参数的下一个参数值。
 - **set-request操作**：设置代理进程的一个或多个参数值，即把一个值存入一个具体的变量。
 - **trap操作**：代理进程主动发出的报文，通知管理进程有某些事情发生，即有一个事件所触发的应答。

SNMP的工作过程图



SNMP的应用

□ SNMPv2的不同操作含义：

- **get-bulk-request 操作**：允许SNMPv2管理者请求得到在给定的条件下尽可能大的应答。遵循算法如下：

L = 变量绑定字段中的变量名数量

N = 只返回一个后继变量的变量名数

R = 返回多个后继变量的变量名数

M = 最大返回的后继变量数

在上述变量之间存在着以下三种关系：

$N = \text{MAX} [\text{MIN}(\text{non-repeters}, L), 0]$

$M = \text{MAX} [\text{max-repetitions}, 0]$

$R = L - N$

注： non-repeaters：指出只返回一个后继变量的变量数
max-repetitions：指出其他的变量应返回的最大的后继变量数

SNMP的应用

- SNMPv2的不同操作含义：
 - **inform操作**：与trap操作相同，也是代理主动向管理发送报文。
inform操作相当于trap操作的升级版，因为trap报文发出去之后不会收到响应报文，而inform报文在发送之后能收到响应报文。

SNMP的应用

□ get请求报文：

- get请求的命令有两个

`snmpget -v 2c -c 共同体名 IP地址 OID号`

`snmpwalk -v 2c -c 共同体名 IP地址 OID号`

- snmpget和snmpwalk两者主要的区别

- snmpwalk是对OID值的遍历，比如某个OID值下面有N个子节点，则依次遍历出这N个子节点的值；
- snmpget是取具体的OID的值，适用于OID值是一个叶子节点的情况

SNMP的应用

□ get-next请求:

■ get-next请求命令:

- `snmpgetnext -v 2c -c 共同体名 IP地址 OID号`
- get-next请求主要是为了得到“命令中OID对象”的下一个对象的值，而“下一个对象”是按照MIB库中的对象实例的顺序来返回对象的值。
- 如使用get-next请求采集OID为“1.3.6.1.2.1.1.3.0”的值，而应答报文中返回是OID为“1.3.6.1.2.1.1.4.0”对象实例的值。

3.1 SNMP的应用

4.4 SNMP的操作命令

□ set请求:

■ set请求命令

- `snmpset -v2c -c 共同体名 IP地址 OID号 命令类型 命令设置的值`
- set请求是为了修改MIB库中某个OID对应的值，但不同的OID采集的值是不同类型的，因此在set请求时，应该设置相应值的类型。
- 如使用使用`该命令`修改被采集设备的系统名称，而系统名为字符串型，所以将系统名改为“linux”的具体命令为：

```
snmpset -v2c -c 共同体名 IP地址 .1.3.6.2.1.1.5.0 s linux
```

SNMP的应用

set 请求命令类型对比表

命令类型	说明	命令类型	说明
i	INTEGER (整数, 有符号的 32 位整数, 取值范围为 - 2147483648 到 +2147483648)	u	UNSIGNED (无符号的 32 位整数, 值的范围为: 0 - 4294967295)
s	STRING (字符串类型, 通常限制在 255 个字符内)	x	HEX STRING (十六位整数, 一般用于私有的 MIB 中进行设置)
b	BITS (比特串, 一个无符号的数据类型)	n	NULLOBJ (将对象设置成空对象, 从而不能采集该对象的值)
t	TIMETICKS (表示代表数据的一个无符号整数, 2^{32} 取模 (4294967296), 以百分之一秒为单位)	a	IPADDRESS (表示一个 IP 地址)

SNMP的应用

- get-bulk请求:

- ▣ get-bulk请求命令

- `snmpbulkget -v2c -c 共同体名 IP地址 OID号`

- ▣ get-bulk请求是为了使用户一次请求可以得到多个对象的值，从而减少用户的一些操作。
 - ▣ 如假定发送的get-bulk请求对象实例为1.3.6.1.2.1.1.3.0 (sysUpTime, 系统开机时间)，由于请求报文中non-repeaters的值为0、max-repetitions的值为10，所以在应答报文中将会返回该对象后不重复的10个对象实例的值。

SNMP的应用

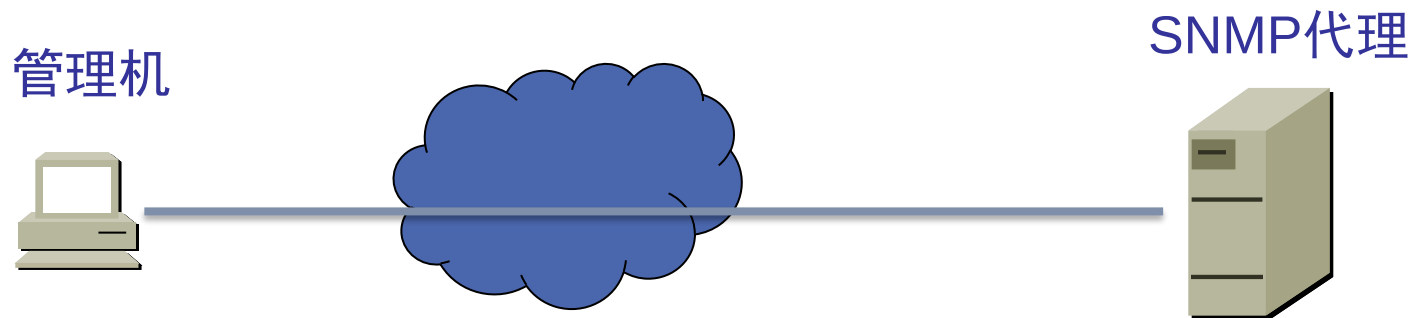
- SNMPv2协议中trap操作：

- trap命令：

`snmptrap -v2c -c 共同体名 管理主机 uptime OID号`

- Trap命令是代理主动采集设备中的相关信息，然后向管理站发送这些信息，上述的命令是在Linux系统下完成代理，并向Windows系统下的主机发送。
 - trap消息的条件为：代理主机安装并开启了SNMP服务，并且确认能够使用SNMP的trap服务，并且在防火墙上开启相关端口允许SNMP发送消息。

SNMP的应用



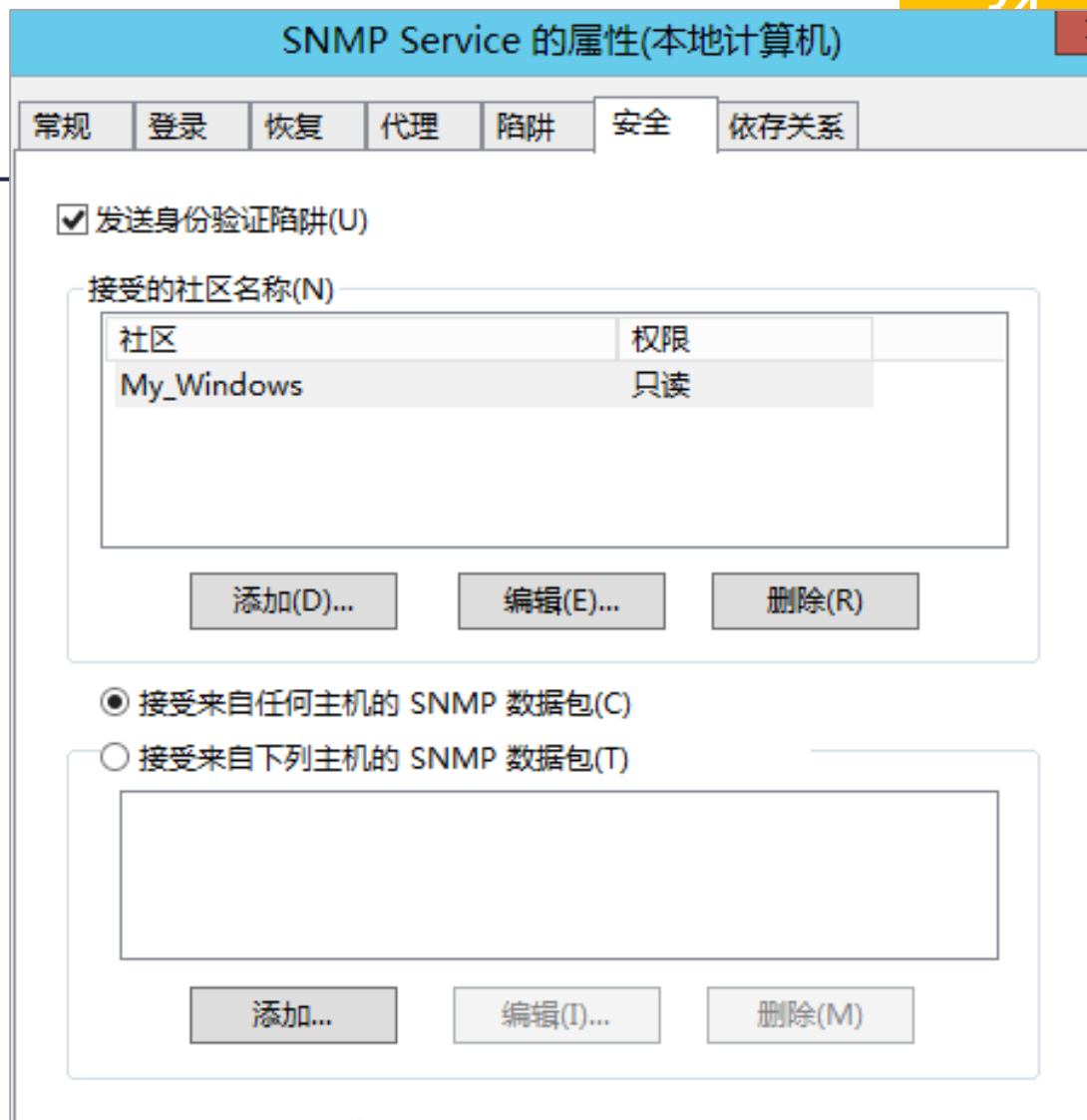
允许管，才能管!

SNMP的应用

□ 设置团体名的意义

- SNMP的团体是一种安全机制，是代理和管理站之间的认证。SNMPv1的安全机制很简单，只是验证团体名(community)。属于同一团体的管理站和被管理站才能相互作用，发送给不同团体的报文被忽略
- 允许访问的团体名是在被管理系统一侧定义的。每一个团体被赋予一个惟一的名称。管理站只能以代理认可的团体名行使其访问权。
- 另一方面，由于团体名的有效范围局限于定义它的代理系统中，所以一个管理站可能以不同的名字出现在不同的代理中，即管理站实体可以用不同的名字对不同的代理实施不同的访问权限。反之，如果两个代理定义了同一团体名，这种名字的相似性也不意味着它们属于同一团体。

- 在Windows中配置共同体（又称为社区）名称



□ 在Linux中配置共同体（community）名称

（1）安装SNMP服务组件

```
# yum -y install net-snmp-libs net-snmp net-snmp-utils net-snmp-devel net-snmp-perl
```

（2）编辑打开snmpd.conf文件，配置共同体名称

```
# vi /etc/snmp/snmpd.conf
```

```
# First, map the community name "public" into a "security name"
#
#      sec.name      source          community
com2sec notConfigUser default        public
```

```
# First, map the community name "public" into a "security name"
#
#      sec.name      source          community
com2sec notConfigUser 192.168.31.100 My_Cacti
```

讨论

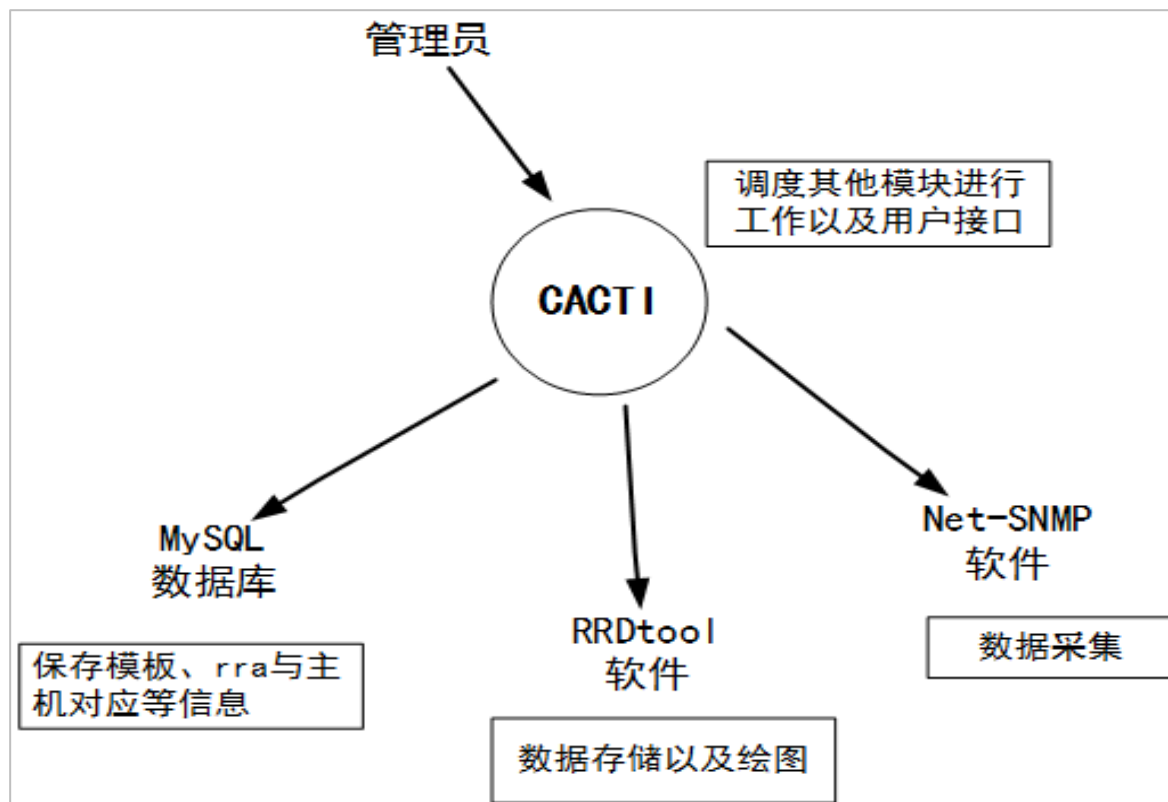
□ Cacti的工作原理

- Cacti是一套基于PHP、MySQL、SNMP及RRDTool的网络流量监测图形分析工具。
- 在Cacti监控体系中，将业务数据（即监控采集的数据）和系统数据（例如账号密码、受监控设备的基本配置信息，像共同体名称等）分开存放。业务数据放置在rrdtool的数据文件（rrd: round robin database）中，系统数据放置在MySQL数据库中

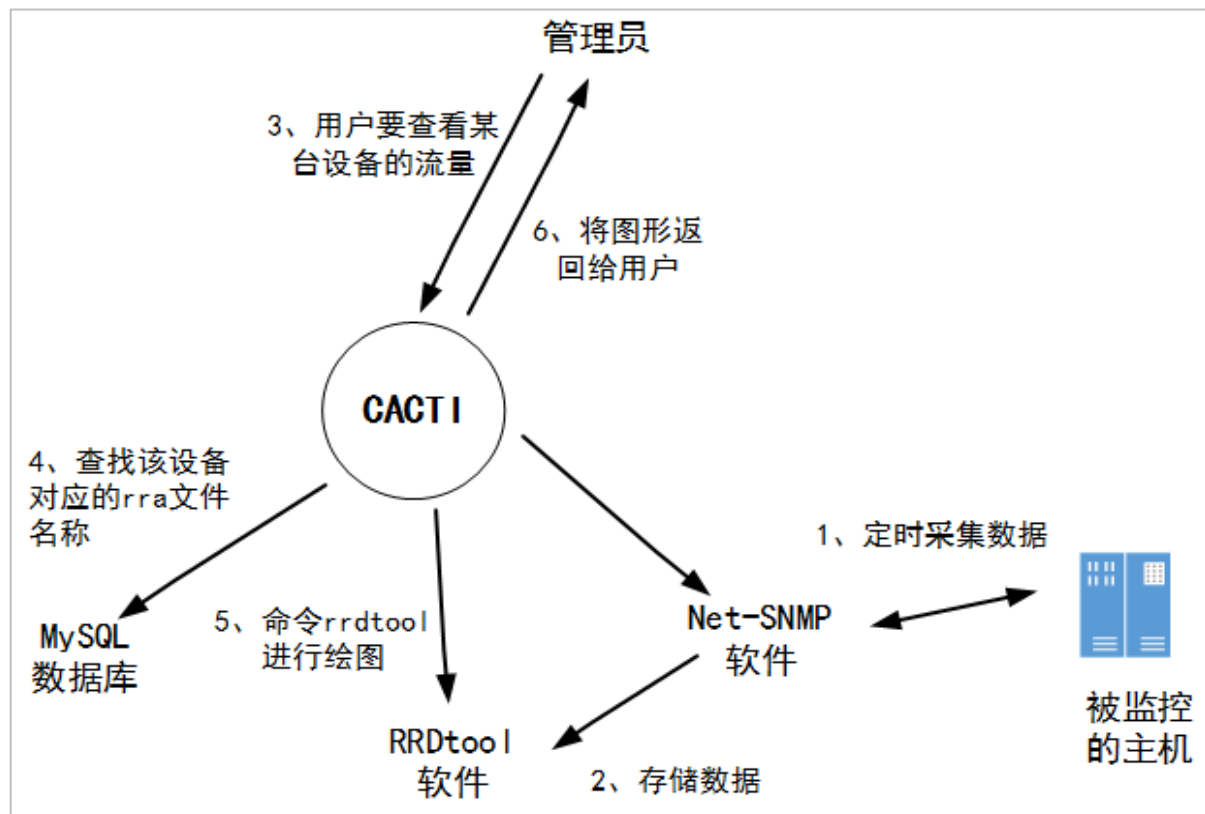
讨论

- Cacti网络监控系统的工作过程
 - 图示说明

讨论



讨论



Thanks.