华东师范大学计算机科学技术系上机实践报告

课程名称：计算机网络	**年级**：2021	**上机实践成绩**：
指导教师：洪道诚	**姓名**：朱宇笑	**创新实践成绩**：
实验名称：简单网络管理协议(SNMP)	**学号**：10225001410	**上机实践日期**：2023.11.17
座位编号：F	**组号**：6	**上机实践时间**：2 学时

组员：黄棋锐，刘玥涵，段欣语，吴杉镐，廖颖鑫，朱宇笑

一、 实验目的

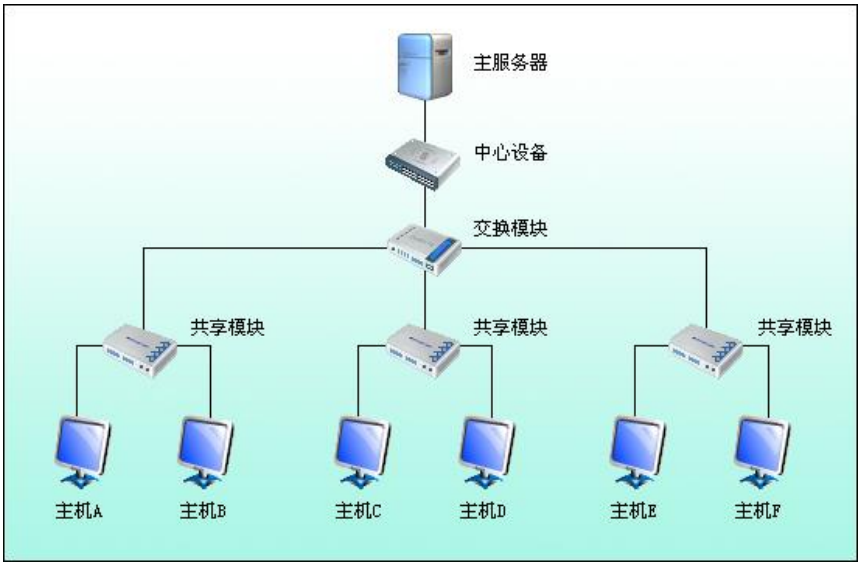
1. 掌握SNMP的报文格式
2. 掌握SMI定义的规则
3. 掌握MIB定义的结构
4. 理解SNMP工作原理

二、 实验设备

1. PC机
2. 仿真编辑器和协议分析器

三、 实验原理

实验采用网络结构一



（一）.SNMP 简介

SNMP（简单网络管理协议）是专门用于在 IP 网络中管理网络节点的一种标准协议。它

用于在 SNMP 代理和 SNMP 管理器之间传送管理信息。SNMP 使网络管理员能够管理网络，发现并解决网络问题以及规划网络增长。

（二）.SNMP 报文格式

SNMP 的 5 种报文，它们封装在 UDP 数据报中，它们都有公共 SNMP 首部，然后是不同的 PDU（其中 get，get-next，set 的 PDU 部分是相同的）。管理进程发出的 get，get-next，set 操作采用 UDP 端口 161，代理进程发出的 trap 操作采用 UDP 的 162 端口。另外，SNMP 报文的编码采用 ASN.1 和 BER。如下图所示。

公共 SNMP 首部格式如下：

版本	共同体	PDU 类型

get，get-next，setPDU 格式如下：

请求标识	0	0	变量绑定

get-responsePDU 格式如下：

请求标识	差错状态	差错索引	变量绑定

trapPDU 格式如下：

企业	代理地址	Trap 类型	特定代码	时间戳	变量绑定

变量绑定字段格式如下：

名称 1	值 1	名称 2	值 2	

图 1-1SNMP 报文格式

●版本：SNMP 版本，值为 0，是通过 SNMP 版本号减去 1 得到的。

●共同体：SNMP 代理和一些 SNMP 应用程序实体的任意集合的组合。是一个字符串，默认值为 public。

●PDU 类型：PDU 类型如下表所示：

表 1-1PDU 类型

PDU 类型	名称
0	get
1	get-next
2	get-response
3	set
4	trap

●请求标识：通过给每个请求提供一个独有的 ID 来区分处理的请求。

●变量绑定：变量名称和相应取值的列表。

●差错状态：用于表示在处理请求时出现的异常。

●差错索引：当差错状态是非零时，可能由列表中导致异常的变量来提供附加的信息。

●企业：产生 trap 的对象的类型。

●代理地址：产生 trap 的对象的地址。

●Trap 类型：通用 trap 类型。

●特定代码：具体 trap 代码。

●时间戳：上次初始化网络实体和产生 trap 之间的所持续的时间。

（三）.SNMP 管理器和代理

SNMP 使用管理器和代理的概念。管理器（通常是主机）控制和监视一组代理（通常是路由器）。如下图所示：

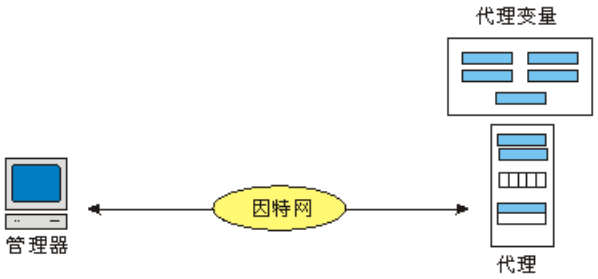


图 1-2 管理器和代理

管理器是运行 SNMP 客户程序的主机。代理是运行 SNMP 服务器程序的路由器(或主机)。管理是通过在管理器和代理之间的简单交互来实现的。

代理在数据库中保存了性能信息。管理器可以使用这个数据库中数值。例如，路由器可以把已收到的和已转发的数据包数存储成适当的变量。管理器可以读取和比较这两个变量，以便发现路由器是否拥塞。

管理器还可以使路由器完成某些动作。例如，路由器定期地检查重新引导计数器的值，看它何时应当重新引导自己。当计数器的值为 0 时就应当重新引导自己。管理器可以随时使用这个特性从远程重新引导这个代理。它只要发送一个数据包，迫使这个计数器的值为 0 即可。

代理也可以参加到管理过程中。在代理上运行的服务器程序可以检查环境，若发现有异常现象可以向管理器发送告警报文（叫做陷阱 trap）。

总之，使用 SNMP 的管理是基于以下 3 个基本思想：

●管理器检查代理的方法是发出请求能够反映代理的行为的信息。

●管理器可以用重新设置在代理数据库中的某些值的方法，强迫代理完成某个任务。

●代理参与管理过程的方法是向管理器发出对异常情况的告警。

（四）.SNMP 管理构件

为了完成管理任务，SNMP 使用另外两个协议：管理信息结构(SMI)和管理信息库(MIB)。如下图所示：

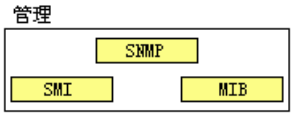


图 1-3SNMP 管理构件

（1）.SNMP 的作用

SNMP 在网络管理中起着非常特殊的作用。它定义了从管理器发送到代理以及从代理发送到管理器的数据包格式。它还解释结果和产生统计。所交换的数据包包含对象（变量）名和它们的状态（值）。SNMP 负责读取和改变这些数值。

（2）.SMI 的作用

要使用 SNMP，就需要命名对象的规则和定义对象类型的规则。SMI 是定义这些规则的协议。SMI 只是定义了这些规则，它并没有定义在一个实体中可以管理多少个对象或哪个对象使用哪一种类型。SMI 是许多通用规则的集合，这些规则用来命名对象和列出它们的类型清单。对象和类型的关联并不是 SMI 应当做的事。

（3）.MIB 的作用

MIB 协议定义了对象的数目，按照 SMI 定义的规则给这些对象命名，并且将对象和一种类型联系起来。

（五）.管理信息结构 SMI

（1）.SMI

SMIv2（管理信息结构版本 2）是用在网络管理中的一个构件。它的功能是：

●定义给对象命名的规则。

●定义可在对象中存储的数据类型。

●给出如何对网络上传输的数据进行编码的方法。

SMI 是 SNMP 的一个指南。它强调处理对象的 3 个属性：名字、数据类型和编码方法，如下图所示：

图 1-4 对象属性

（2）. 名字

SMI 要求一个被管对象具有惟一的名字。为了在全局给对象命名，SMI 使用对象标识符，它是基于树结构的分层次的标识符。如下图所示：

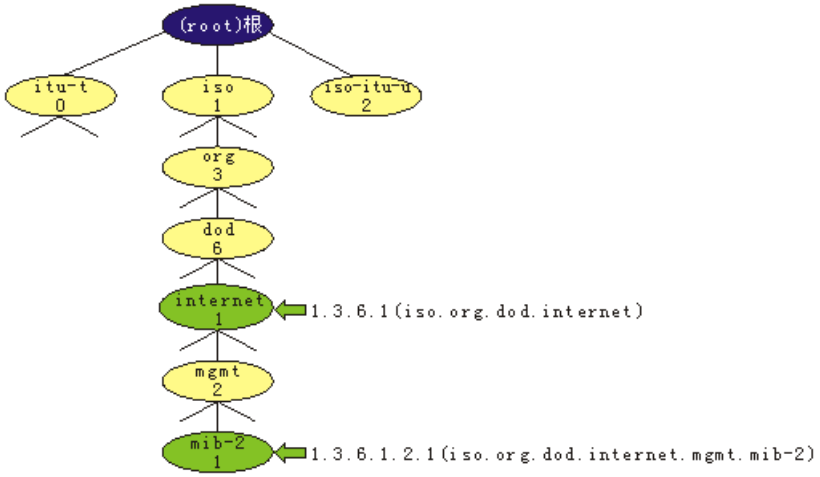


图 1-5 对象标识符

树结构从不命名的根开始。每一个对象可用点分隔开的整数序列定义。这种树结构也可以使用点分隔开的文本名字序列来定义对象。名字——点的表示方法是用户使用的。

SNMP 使用的各对象都位于 mib-2 对象下面的，它们的标识符永远从 1.3.6.1.2.1 开始。

（3）. 类型

对象的第二个属性是对象中存储的数据的类型。SMI 使用 ASN.1（抽象语法 1）的基本定义定义了数据类型，同时也增加了几个新的类型。SMI 既是 ASN.1 的一个子集，也是 ASN.1 的一个超集。

SMI 使用两大数据类型：简单的和结构化的。

① 简单类型

简单数据类型是原子数据类型。这些类型中的一些是直接取自 ASN.1 的；另一些是 SMI 增加的。下表给出了一些重要的类型，前 5 个取自 ASN.1；后 7 个是 SMI 定义的。

表 1-2SMI 简单类型

类型	大小	说明
INTEGER	4 字节	在 -2^{31} ～ $2^{31}-1$ 之间的整数
Interger32	4 字节	和 INTEGER 相同
Unsigned32	4 字节	在 0～ $2^{32}-1$ 之间的无符号数
OCTETSTRING	可变	不超过 65535 字节长的字节串
OBJECTIDENTIFIER	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

类型	大小	说明
Counter64	8 字节	64 位的计数器
Gauge32	4 字节	与 COUNTER32 相同，但当它到达最大值时不返回；它保持在这个数值到复位
TimrTicks	4 字节	记录时间的计数值，以 1/100s 为单位
BITS		位串
Opaque	可变	不解释的串

② 结构化类型

把简单的和结构化的数据类型组合起来，就可以构成新的结构化数据类型。SMI 定义了两种结构化数据类型：sequence 和 sequenceof。

Sequence：sequence 数据类型是一些简单数据类型的组合，但不必都是相同的类型。它和 C 语言中使用的 struct 或 record 的概念相似。

Sequenceof：sequenceof 数据类型是所有相同类型的简单数据类型的组合，或所有相同类型的 sequence 数据类型的组合。它和 C 语言中使用的 array 的概念相似。

（4）.编码方法

SMI 编码方法使用 BER（基本编码规则），把数据编码后在网络上传输。BER 将每一块数据编码成三元组格式：标记、长度和值，如下图所示：

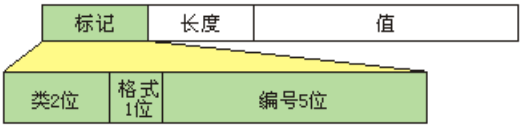


图 1-6 编码格式

① 标记

标记是定义数据类型的 1 字节字段。它由 3 个子字段组成：类字段（2 位）、格式字段（1 位）和编号字段（5 位）。类字段定义数据的作用域（scope）。共定义了 4 个类：通用类（00）、应用类（01）、特定上下文类(10)和专用类(11)。通用数据类型是来自 ASN.1 的。应用数据类型是由 SMI 增加的。有 5 种特定上下文数据类型，它们的意义随着协议的不同而不同。专用数据类型是特定厂商使用的。格式字段指出数据是简单的（0）还是结构化的(1)。编号字段将简单的或结构化的数据进一步划分为一些子组（subgroup）。例如在通用类的简单格式，INTEGER 的值是 2，OCTETSTRING 的值是 4，等等。

② 长度

长度字段是 1 或多字节。若它是 1 字节，则最高位必定为 0，其余的 7 位定义数据长度。若大于 1 字节，则第一字节的最高位必定为 1。第一个字节的其余 7 位则定义所需的字节数。下图说明了长度字段的意义：

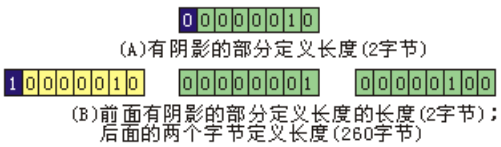


图 1-7 长度的格式

③ 值

值字段按照在 BER 中定义的规则把数据的值进行编码。

（六）.管理信息库 MIB

MIB2（管理信息库版本 2）是网络管理中的第二个构件。每一个代理有它自己的 MIB2，这是管理器能够管理的所有对象的集合。在 MIB2 中的对象分成 10 个不同的组：system（系统）、interface（接口）、addresstranslation（地址转换）、ip、icmp、tcp、udp、egp、transmission（传输）和 snmp。这些组都在对象标识符树中 mib-2 对象的下面，如下图所示。每一个组

定义一些变量和表。

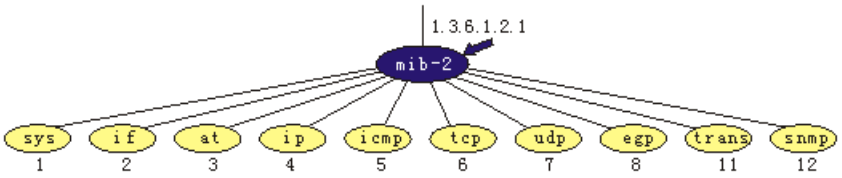


图 1-8MIB

下面是一些对象的简单描述。

sys：系统定义有关这个节点（系统）的一般信息，如名字、位置和寿命。

if：接口定义有关这个节点的所有接口的信息，如接口号、物理地址和 IP 地址。

at：地址转换定义有关 ARP 表的信息。

ip：这个对象定义有关 IP 的信息，如路由表和 IP 地址。

icmp：这个对象定义有关 ICMP 的信息，如已发送和已收到的数据包数，以及产生的差错总数。

tcp：这个对象定义有关 TCP 的信息，如连接表、超时值、端口号，以及已发送和已收到的数据包数。

udp：这个对象定义有关 UDP 的信息，如端口号、已发送和已收到的数据包数。

snmp：这个对象定义有关 SNMP 本身的信息。

（1）. 访问 MIB 变量

下面以 udp 组作为例子来说明如何访问不同的变量。在 udp 组有 4 个简单变量和一个记录序列（表）。下图给出了这些变量和表。

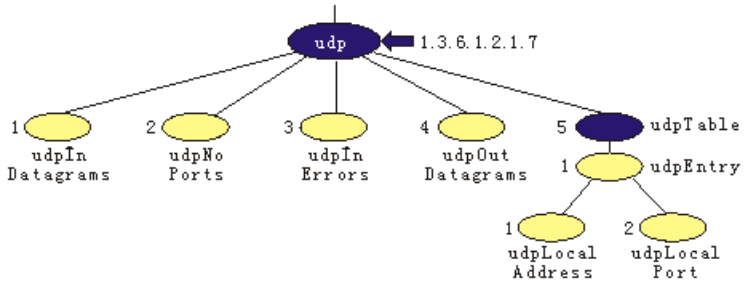


图 1-9udp 组

下面说明如何访问每一个实体。

访问简单变量时使用组的 id（1.3.6.1.2.1.7）后面跟着该变量的 id。下面给出了如何访问每一个变量。

udpInDatagrams->1.3.6.1.2.1.7.1

udpNoPorts-> 1.3.6.1.2.1.7.2

udpInErrors-> 1.3.6.1.2.1.7.3

udpOutDatagrams-> 1.3.6.1.2.1.7.4

但是，这些对象标识符定义的是变量而不是内容。要给出每一个变量的内容，应该增加内容的后缀。简单变量的内容后缀就是零。例如：

udpInDatagrams-> 1.3.6.1.2.1.7.1.0

udpNoPorts-> 1.3.6.1.2.1.7.2.0

udpInErrors-> 1.3.6.1.2.1.7.3.0

udpOutDatagrams-> 1.3.6.1.2.1.7.4.0

MIB 使用表 id 标志表。如下图所示，udp 只有一个表（id 是 5）。

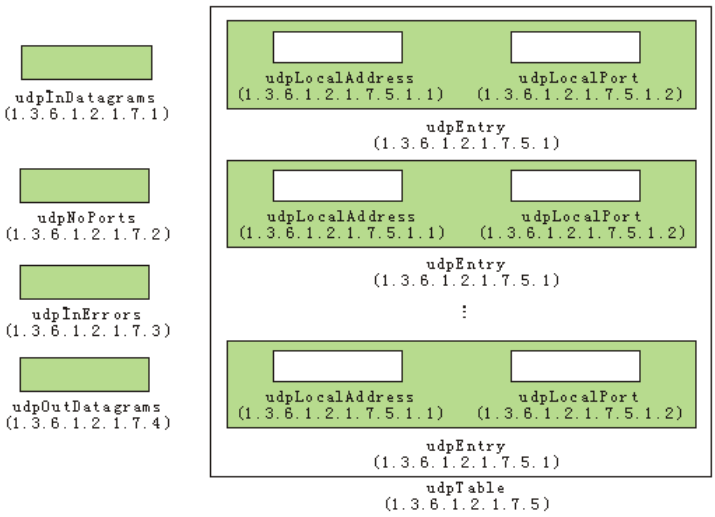


图 1-10udp 的变量和表

要访问这个表，应该使用如下的表 id：

udpTable-> 1.3.6.1.2.1.7.5

但是，这个表还是在树结构的树叶级。因此不能访问这个表，下面定义在这个表（id 是 1）中的项目：

udpEntry-> 1.3.6.1.2.1.7.5.1

这个项目也不是树叶，因此不能访问它。还需要定义这个项目中的每一个实体（字段）。

udpLocalAddress-> 1.3.6.1.2.1.7.5.1.1

udpLocalPort-> 1.3.6.1.2.1.7.5.1.2

这时两个变量都是在树叶上了。这个表对每一个“本地地址/本地端口对”可以有不同的值。要访问表中的特定行，应当给以上的 id 加上索引。在 MIB 中，数组的索引不是整数。这些索引是基于在这些项目中的一个或多个字段的值。在例子中，udpTable 的索引基于本地地址和本地端口号。

（七）.SNMP 通信过程

SNMP 在两个熟知端口 161 和 162 上使用 UDP 的服务。熟知端口 161 由代理使用，而熟知端口 162 由管理器使用。

代理在端口 161 上发出被动打开。然后它就等待从管理器来的连接。管理器使用短暂端口发出主动打开。客户向服务器发送请求报文，使用短暂端口作为源端口而熟知端口 161 作为目的端口。服务器向客户发送响应报文，使用熟知端口 161 作为源端口而短暂端口作为目的端口。

管理器在端口 162 发出被动打开。然后它就等待从代理来的连接。代理只要有 Trap 报文要发送，就使用短暂端口发出主动打开。这个连接只是单向的，从服务器到客户，如下图所示：

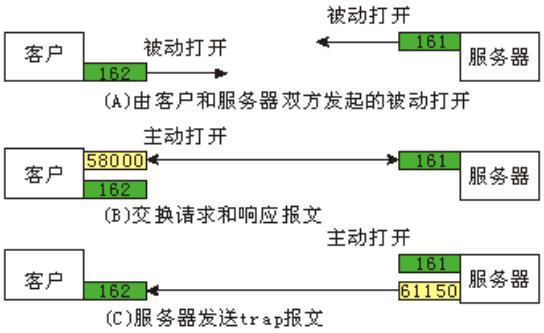


图 1-11SNMP 的端口号

在SNMP中的客户/服务器机制与其它协议的不同。这里的客户和服务器都使用熟知端

口。此外，客户和服务器都必须无限制地运行下去。这个原因就是请求报文是由管理器发出的，但Trap报文则是由代理发出的。

四、 实验步骤

练习一获取代理服务器信息

各主机打开工具区的“拓扑验证工具”，选择相应的网络结构，配置网卡后，进行拓扑验证，如果通过拓扑验证，关闭工具继续进行实验，如果没有通过，请检查网络连接。

本练习将主机 A 和 B 作为一组，主机 C 和 D 作为一组，主机 E 和 F 作为一组。现仅以主机 A、B 所在组为例，其它组的操作参考主机 A、B 所在组的操作。

其中主机 B 作为 SNMP 代理服务器，主机 A 作为 SNMP 管理器。主机 B 在实验前应当通过管理工具中的“服务”工具查看是否已经安装了名为“SNMPService”服务，如果没有安装，应当参考实验附录 C 完成“SNMPService”服务的安装。

1. 主机 B 启动 SNMP 服务，并创建具有“只读”权利的团体“public”接受来自任何主机的 SNMP 数据包。配置方法如下：

(1)启动“服务”管理器，找到“控制面板/管理工具/服务”程序，双击启动。

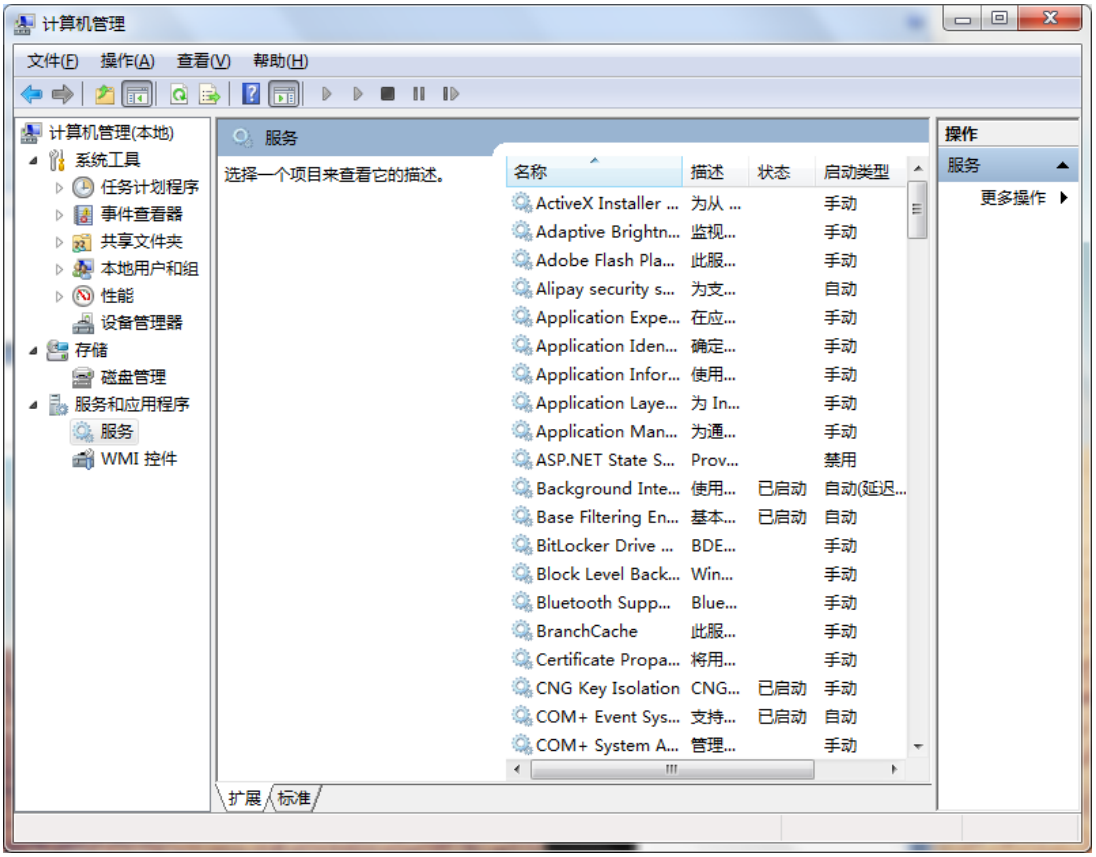


图 1-12 服务

(2)启动“SNMPService”和“SNMPTrap”。

①在服务程序列表中找到“SNMPService”和“SNMPTrap”。

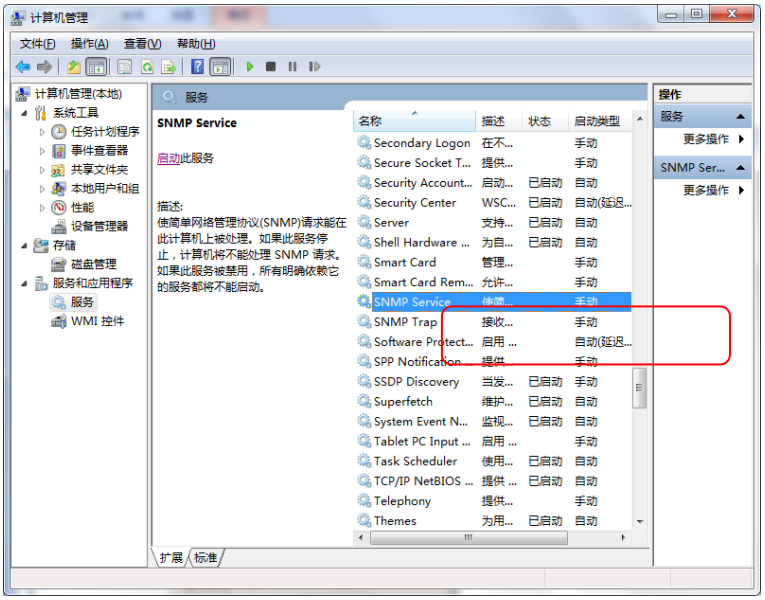


图 1-13SNMP 服务

②选中“SNMPService”条目，单击鼠标右键，选择“属性”菜单项，并修改“启动类型”为“手动”，点击[确定]按钮保存设置。

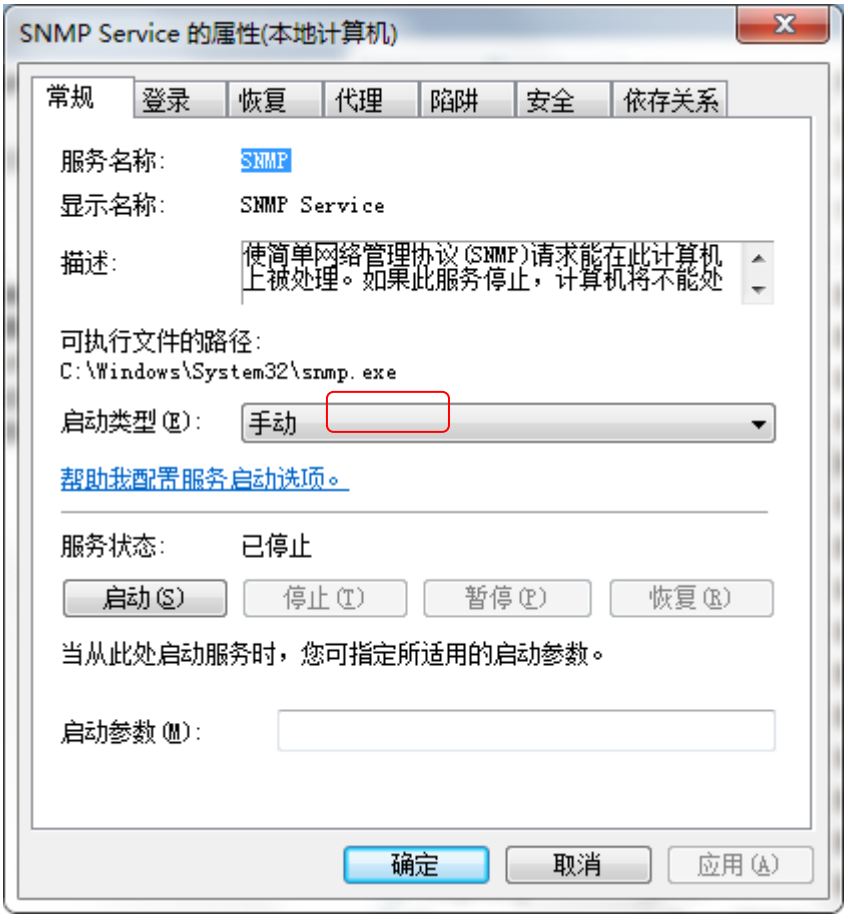


图 1-14 更改服务属性

③单击“服务”管理器菜单栏上的[启动]按钮启动服务。

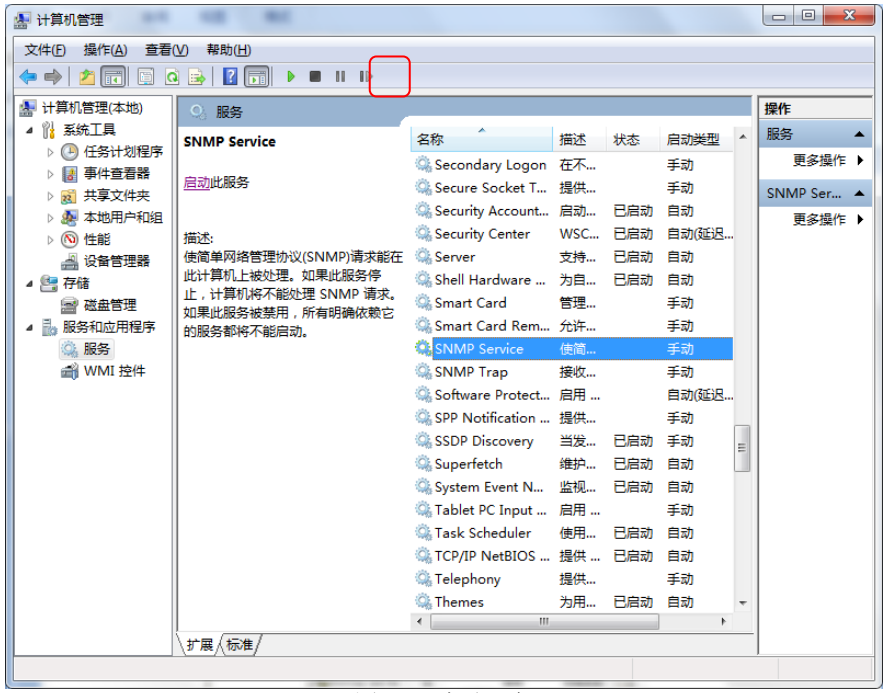


图 1-15 启动服务

④按同样的方法启动“SNMPTrap”。启动后的状态如下图所示：

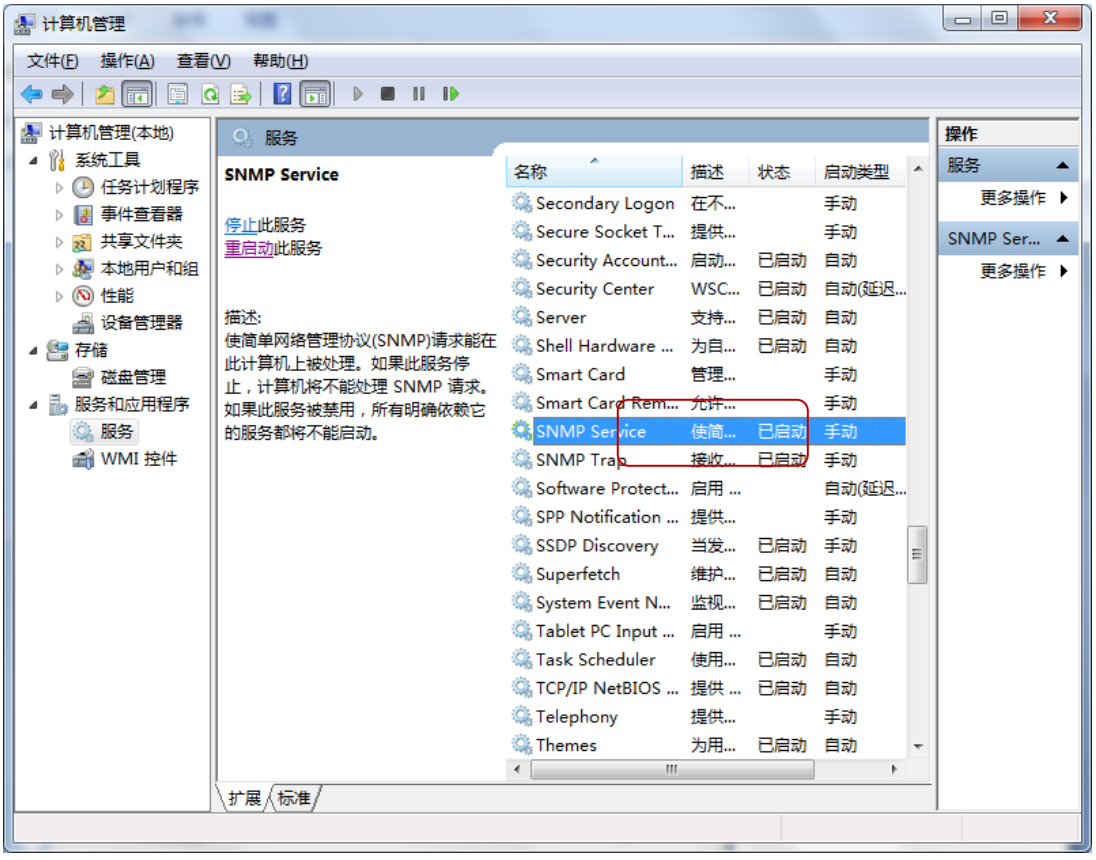


图 1-16 启动服务

(3)设置“代理”属性页。

选中“SNMPService”条目，单击鼠标右键，选择“属性”菜单项。在属性页集合中找到“代理”属性页，并按照下图所示设置：

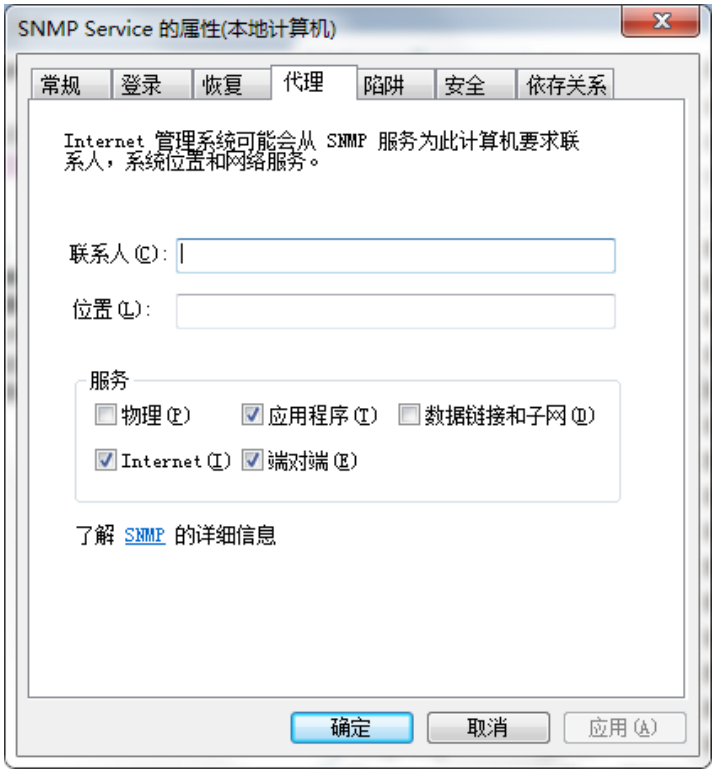


图 1-17SNMPService 属性

(4)设置“陷阱”属性页。

在属性页集合中找到“陷阱”属性页，并按照下图所示设置：

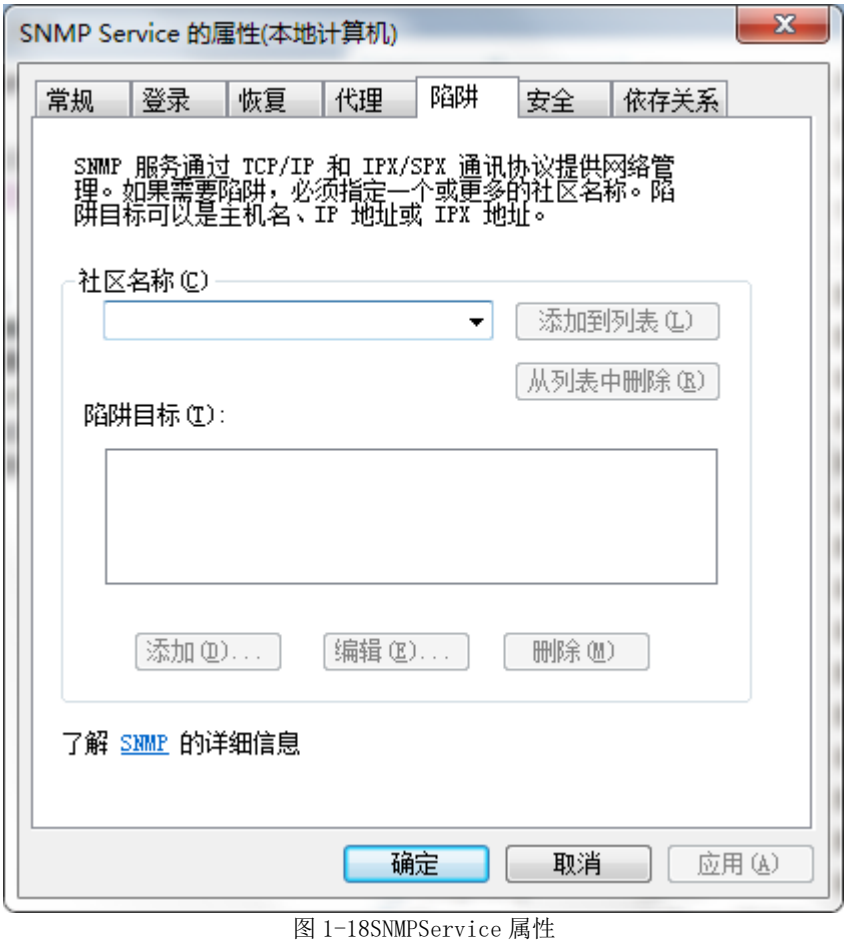


图 1-18SNMPService 属性

(5)设置“安全”属性页。

在属性页集合中找到“安全”属性页，并按照下图所示设置：

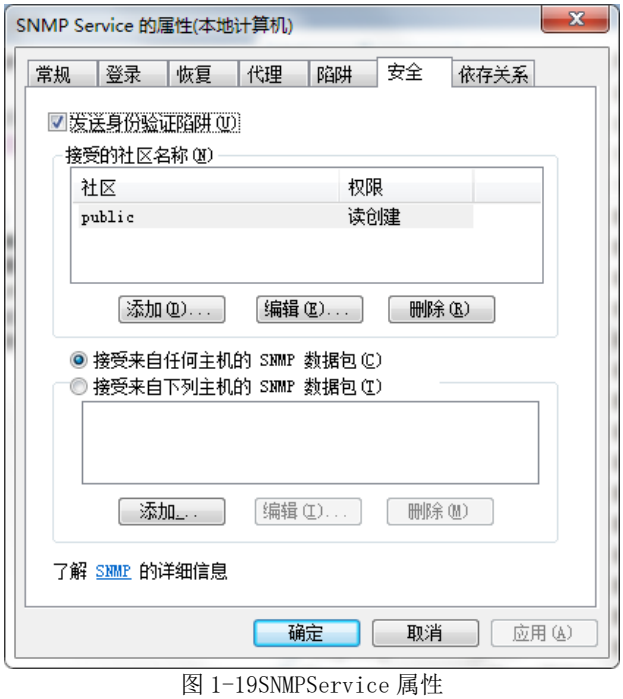


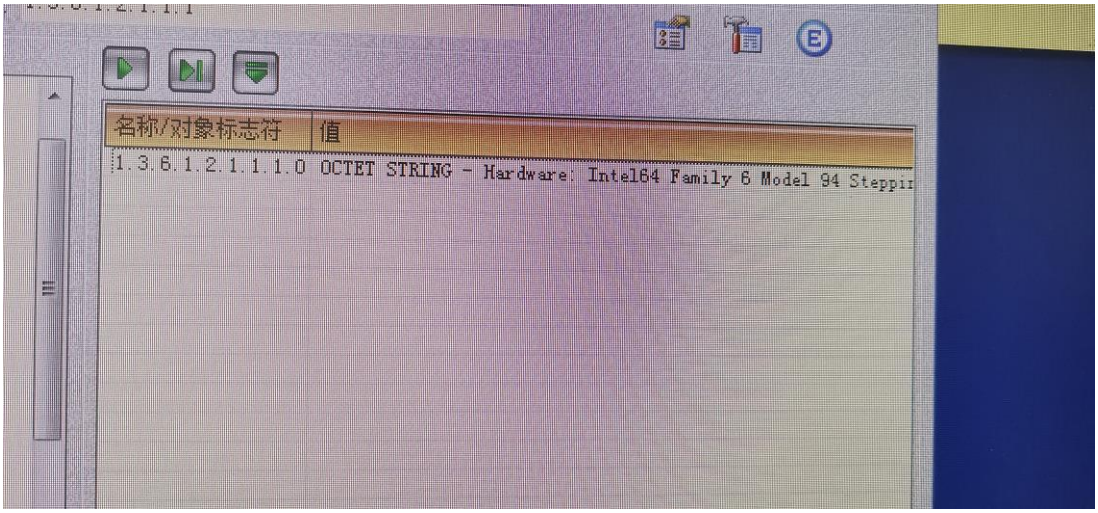
图 1-19SNMPService 属性

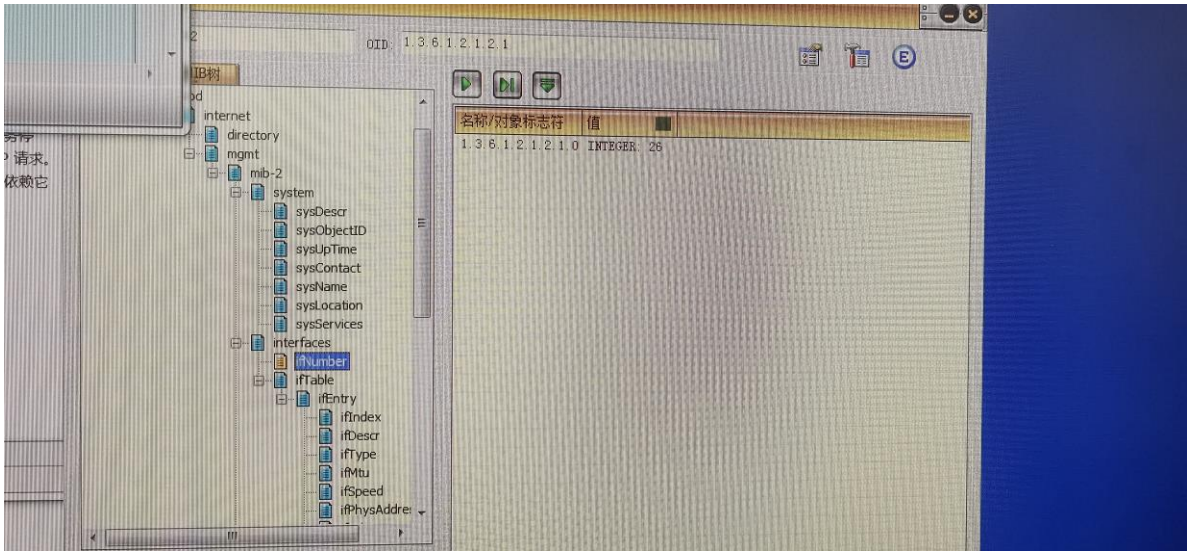
2. 主机 B 启动协议分析器捕获数据，并设置过滤条件（提取 SNMP 协议）。

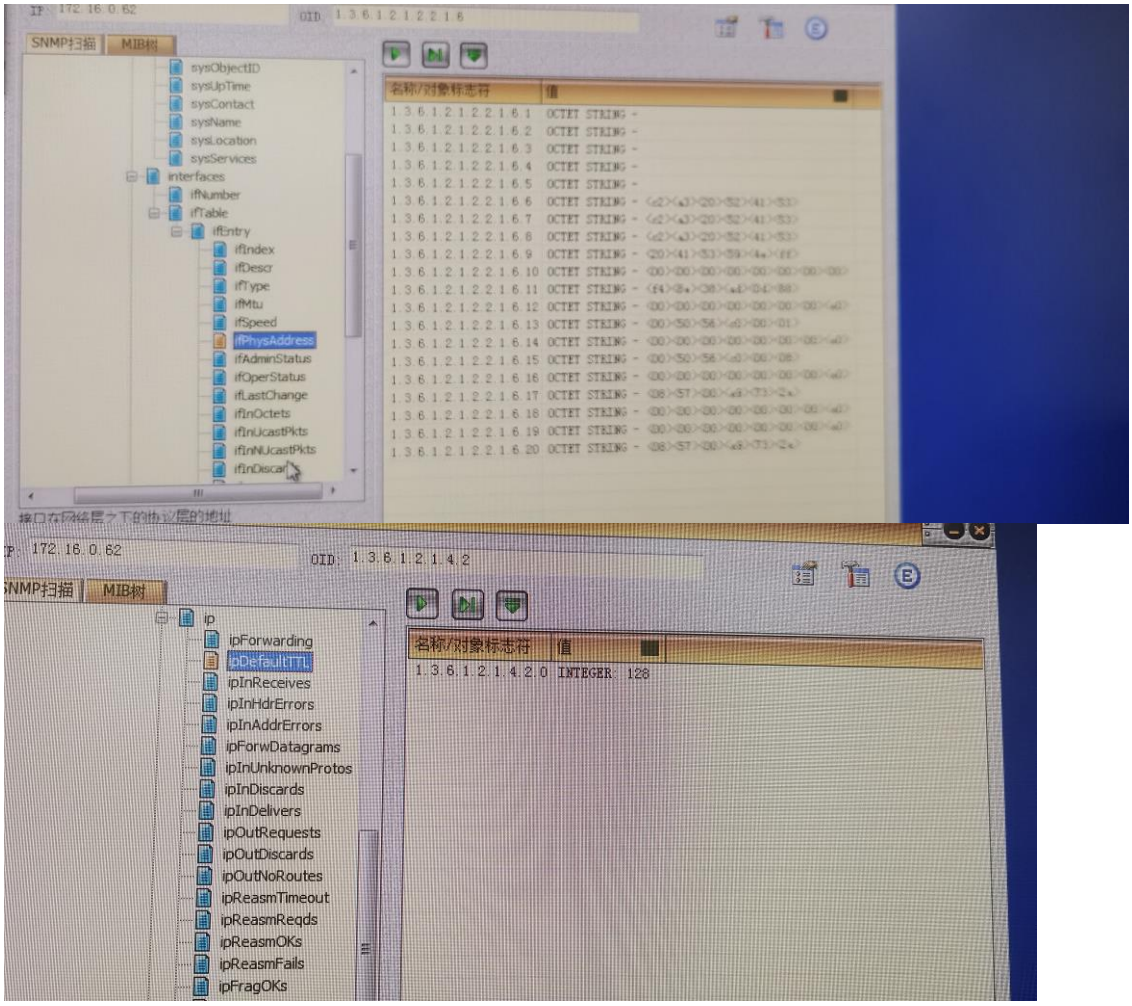
3. 主机 A 扫描 SNMP 主机，填入主机 B 的 IP 地址。

主机 A 启动“实验平台工具栏中的 SNMP 工具”在“SNMP 扫描”页签下点击[开始扫描 SNMP 主机]按钮，在列表中找到主机 B 的 IP 地址，双击该地址，使其添加到工具栏中的“IP”文本框中。

4. 在主机 A 上，展开 MIB 树，通过双击树节点来获取代理服务器信息。







●按照返回的代理服务器信息，填写下表：

表 1-3 实验结果

代理服务器信息	OID	返回值类型	返回值
操作系统类型 (sysDescr)	1.3.6.1.2.1.1.1.0	OCTET STRING	Hardware:Intel64 Family 6 Model 94 Stepping 3 AT/AT COMPATIBLE - Software:

			Windows Version 6.1 (Build 7601 Multiprocessor Free)
网卡数 (ifNumber)	1.3.6.1.2.1.2.1.0	INTEGER	26
物理地址 (ifPhysAddress)	1.3.6.1.2.1.2.2.1.6	OCTET STRING	主机 B 的接口 1 的 MAC 主机 B 的接口 2 的 MAC
IP 默认 TTL 值 (ipDefaultTTL)	1.3.6.1.2.1.4.2.0	INTEGER	128

●通过对代理服务器信息的获取，推测该代理服务器的路由表。

答：主要查看 iso.org.dod.internet.mgmt.mib-2.ip.ipRouteTable.ipRouteEntry.ipRouteDest 的信息。

5. 主机 B 停止捕获数据。通过分析捕获到的报文，回答以下问题：

●为加深对 SMI（管理信息结构）的理解，现给出某一报文中 SNMP 协议的数据。

302602010004067075626c6963a11902020a52020100020100300d300b06072b0601020101010500

结合 SNMP 报文格式和 SMI 定义的规则，绘制出树形的结构图（用树来表现 sequence 和 sequenceof 的关系）。

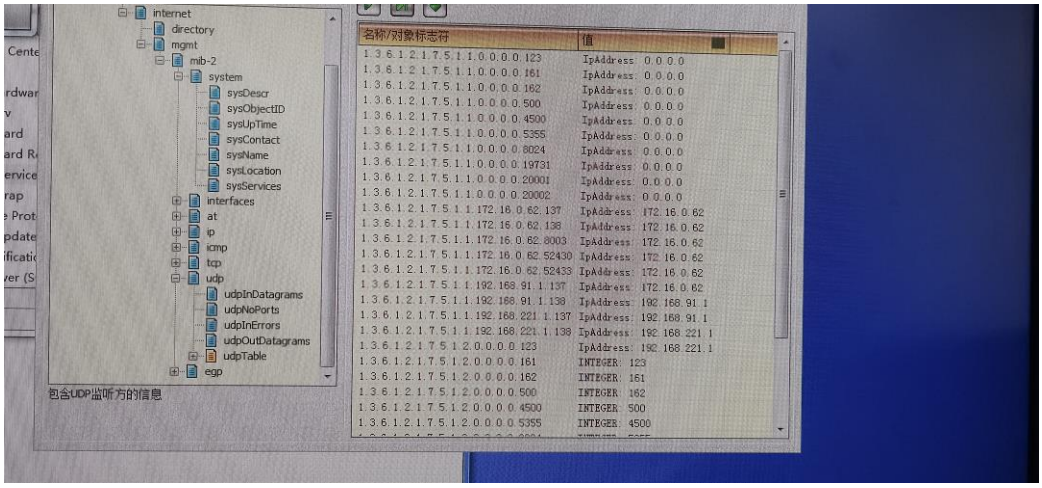
```
Sequence
Length = 38
    Integer
    Length = 1
    SNMP version = 1
    Octetstring
    Length = 6
    Community = public
    Command = Get Next Request
    Length = 25
    Integer
    Length = 2
    Request ID = 10
    Integer
    Length = 1
    Error Status = 0(noError)
    Integer
    Length = 1
    Error Index = 0
    Sequence
    Length = 13
        Sequence
        Length = 11
            Object
            Length = 7
            Object = 1.3.6.1.2.1.1.1 (sysDescr)
            Null
            Length = 0
            Null
```

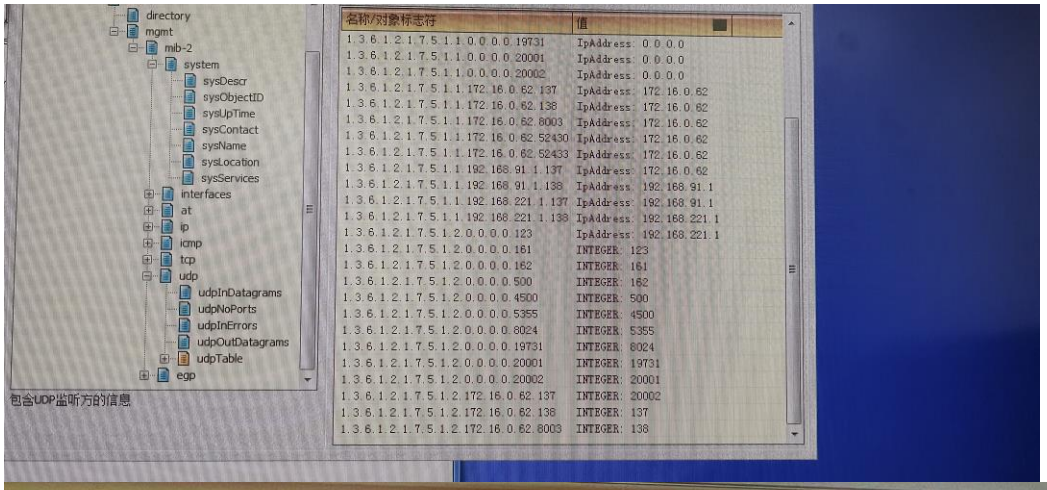
6. 主机 B 重新启动数据捕获。

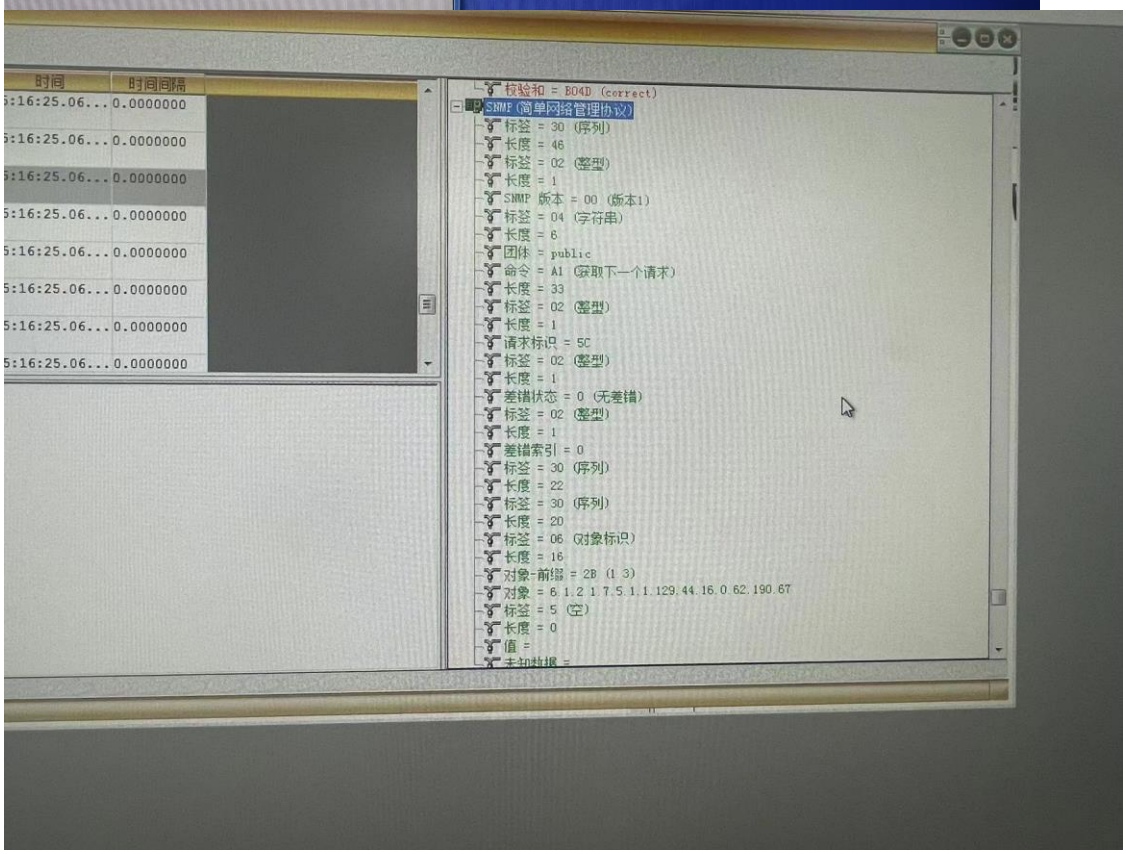
7. 在主机 A 上，用鼠标选中

“iso.org.dod.internet.mgmt.mib-2.udp.udptable”节点，单击工具条上的[获取子树]按钮（按钮图标为  ）。

结果如图：


internet
directory
mgmt
mib-2
system
sysDescr
sysObjectID
sysUpTime
sysContact
sysName
sysLocation
sysServices
interfaces
at
ip
icmp
tcp
udp
udpInDatagrams
udpNoPorts
udpInErrors
udpOutDatagrams
udpTable
egp
包含UDP监听方的信息
名称/对象标志符
值


directory
mgmt
mib-2
system
sysDescr
sysObjectID
sysUpTime
sysContact
sysName
sysLocation
sysServices
interfaces
at
ip
icmp
tcp
udp
udpInDatagrams
udpNoPorts
udpInErrors
udpOutDatagrams
udpTable
egp
包含UDP监听方的信息
名称/对象标志符
值


时间
时间间隔
校验和 = B04D (correct)
SNMP (简单网络管理协议)
标签 = 30 (序列)
长度 = 46
标签 = 02 (整型)
长度 = 1
SNMP 版本 = 00 (版本1)
标签 = 04 (字符串)
长度 = 6
团体 = public
命令 = A1 (获取下一个请求)
长度 = 33
标签 = 02 (整型)
长度 = 1
请求标识 = 5C
标签 = 02 (整型)
长度 = 1
差错状态 = 0 (无差错)
标签 = 02 (整型)
长度 = 1
差错索引 = 0
标签 = 30 (序列)
长度 = 22
标签 = 30 (序列)
长度 = 20
标签 = 06 (对象标识)
长度 = 16
对象=前缀 = 2B (1 3)
对象 = 6 1 2 1 7 5 1 1 129 44 16 0 62 190 67
标签 = 5 (空)
长度 = 0
值 =

●通过察看代理服务器返回的结果 OID 列表和主机 B 上捕获的 SNMP 报文类型，简述字典式排序在 SNMP 查询方式中的意义。

答：MIB 变量的对象标识符（包括实例标识符）是按照字典的顺序排列的。表的顺序是按照列-行规则，这就是说，要按逐列的顺序走。在每一个列中，必须是从顶向底走。字典式的排序可以使管理器在定义了第一个变量后，可以一个接一个地访问一组变量。

●列出你所熟悉的代理服务器开放的 UDP 端口和 UDP 服务名。

答：
161	SNMP
162	SNMP Trap
137	NetBios Name
138	NetBios DGM

8. 关闭 SNMP 工具。

练习二设置代理服务器信息

本练习将主机 A 和 B 作为一组，主机 C 和 D 作为一组，主机 E 和 F 作为一组。现仅以主机 A、B 所在组为例，其它组的操作参考主机 A、B 所在组的操作。

1. 主机 B 修改 SNMP 服务配置，为团体“public”开放“读/写”权利。

选中“SNMPService”条目，单击鼠标右键，选择“属性”菜单项。在属性页集合中找到“安全”属性页，并按照下图所示设置：

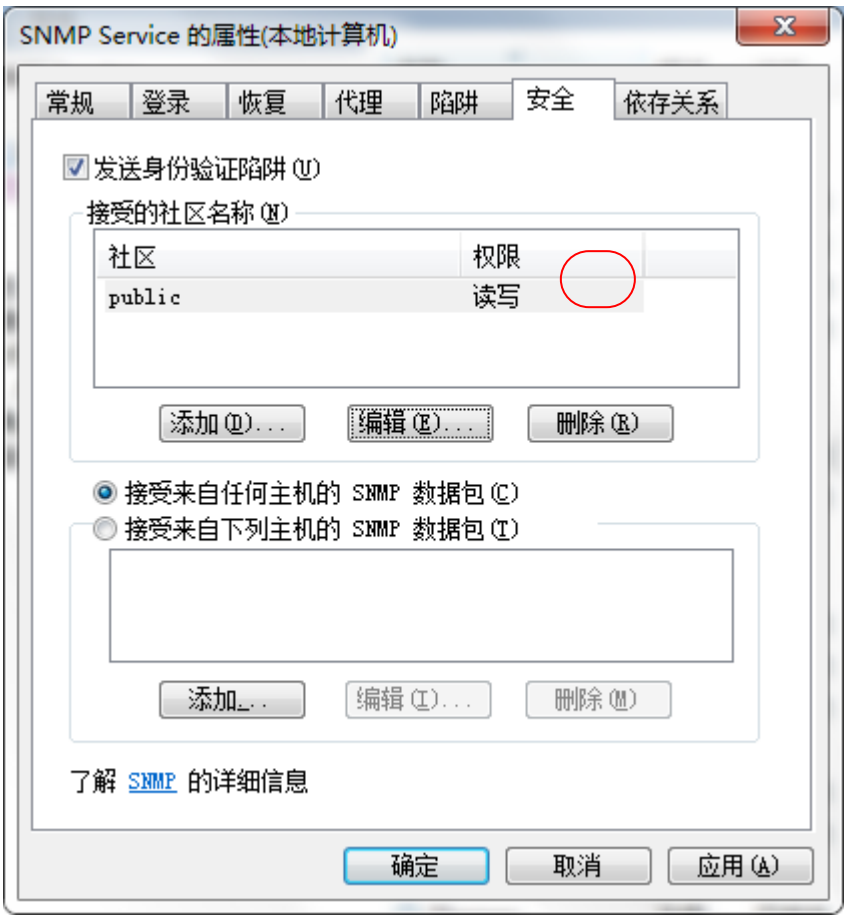
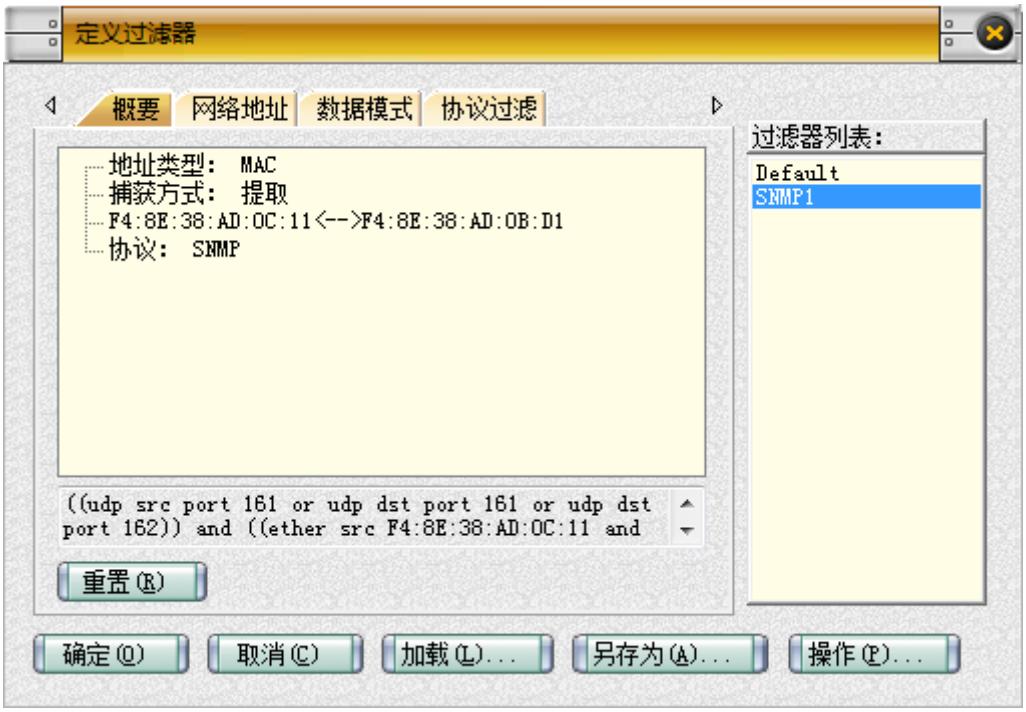


图 1-20SNMPService 属性

2. 主机 B 启动协议分析器开始捕获数据并设置过滤条件（提取 SNMP 协议）。



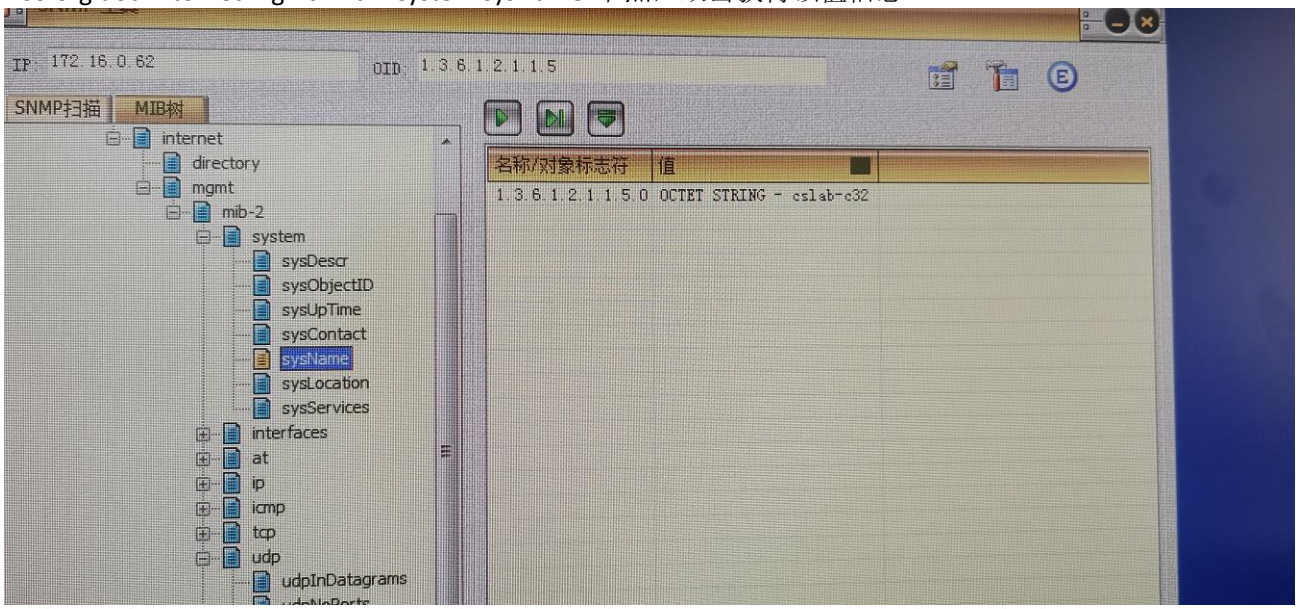
3. 主机 A 更改主机 B 的 sysName 节点值。

(1)主机 A 启动“实验平台工具栏中的 SNMP 工具”。

(2)单击“SNMP 扫描”页签后点击[开始扫描 SNMP 主机]按钮，在列表中找到主机 B 的 IP 地址，双击该地址，使其添加到工具栏中的“IP”文本框中。

(3)单击“MIB 树”页签获得树列表，在列表中找到。

“iso.org.dod.internet.mgmt.mib-2.system.sysName”节点，双击获得该值信息。



(4)在工具条上，点击[设置 SNMP 主机的相关信息]按钮，设置“公共体名称”下拉框为“public”；在“节点值”文本框中输入任意字符串；“变量绑定类型”单选按钮选择[OctetString]，如下图所示。点击[确定]按钮。

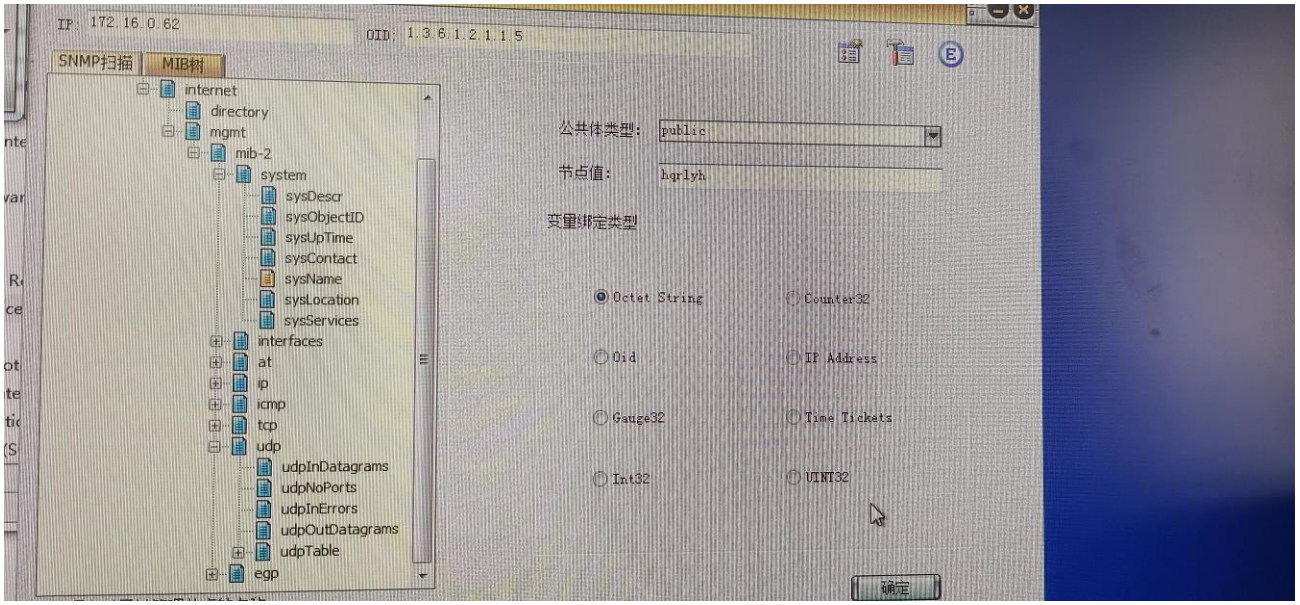
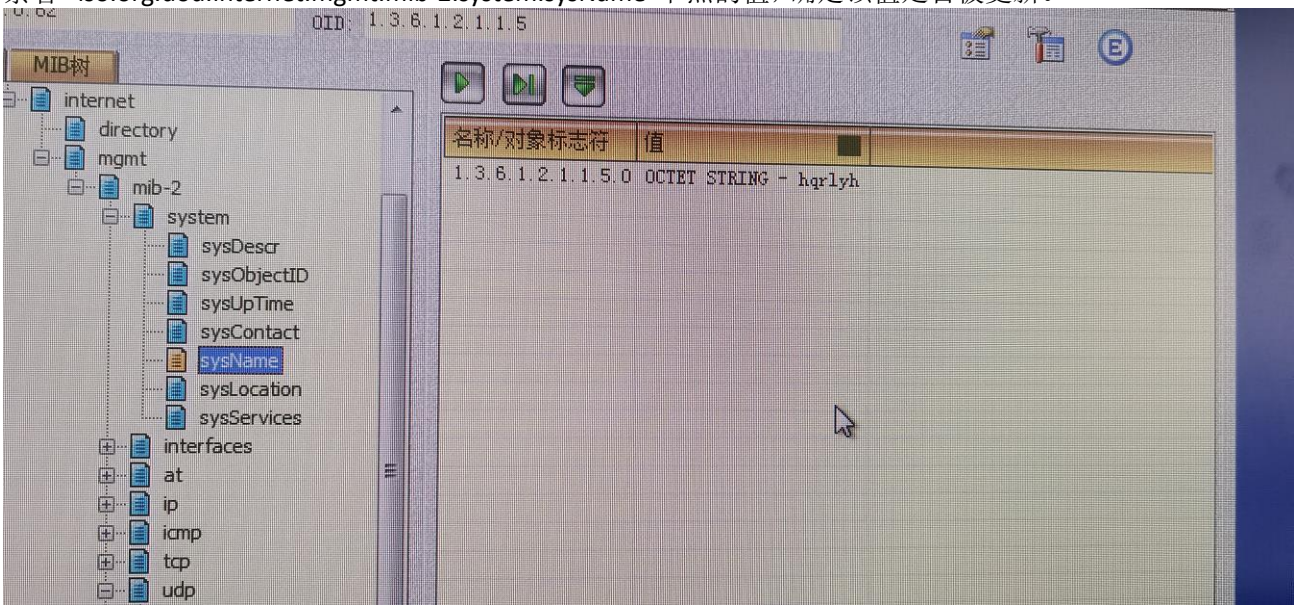


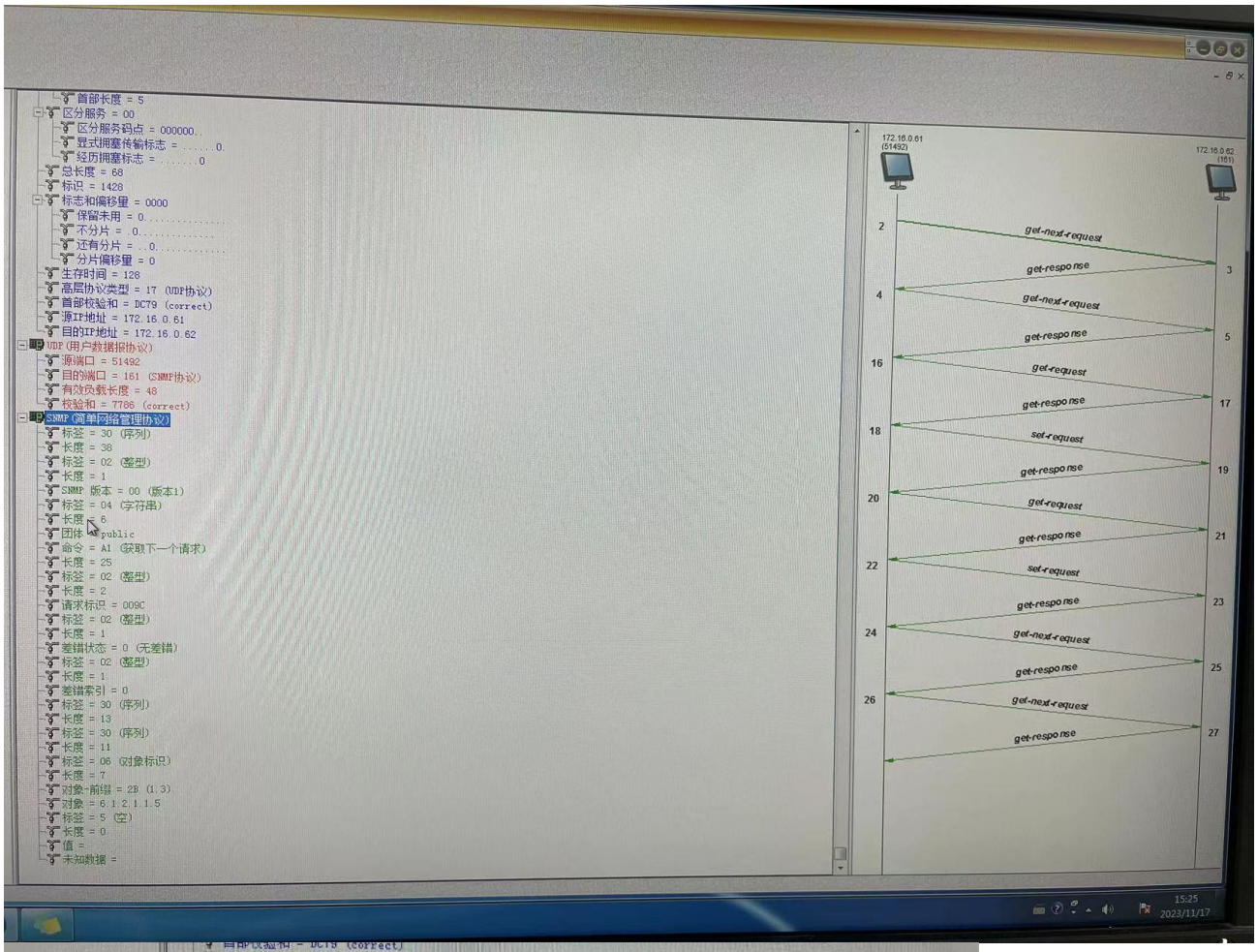
图 1-21SNMP 工具

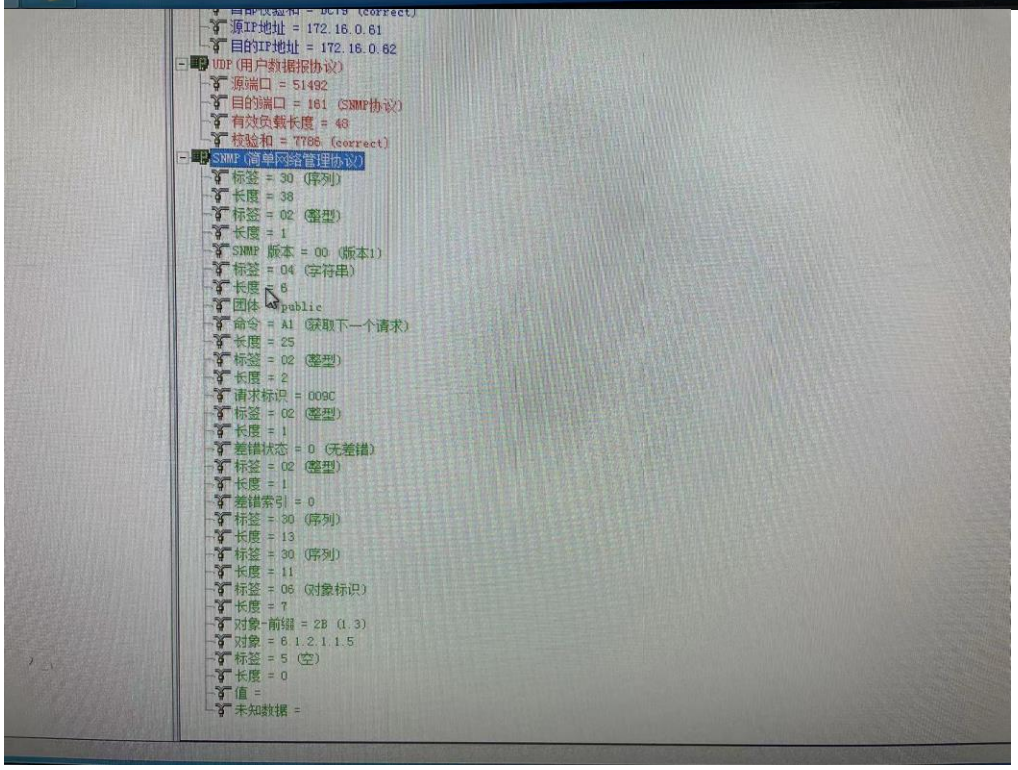
4. 主机 A 点击工具条上的[显示 SNMP 主机的相关信息]按钮，

察看“iso.org.dod.internet.mgmt.mib-2.system.sysName”节点的值，确定该值是否被更新。



5. 主机 B 停止捕获数据，分析捕获到的数据。





6. 关闭 SNMP 工具。

练习三代理服务器的事件报告

本练习将主机 A 和 B 作为一组，主机 C 和 D 作为一组，主机 E 和 F 作为一组。现仅以主机 A、B 所在组为例，其它组的操作参考主机 A、B 所在组的操作。

1. 主机 B 修改 SNMP 服务配置，为团体“public”设置“陷阱”。

(1)选中“SNMPService”条目，单击鼠标右键，选择“属性”菜单项。在属性页集合中找到“陷阱”属性页；添加“团体名称”为 public；点击[添加到列表]按钮；点击[添加(D)...]按钮来设置“陷阱目标”（注意陷阱目标使用主机 A 的 IP 地址）。如下图所示设置：

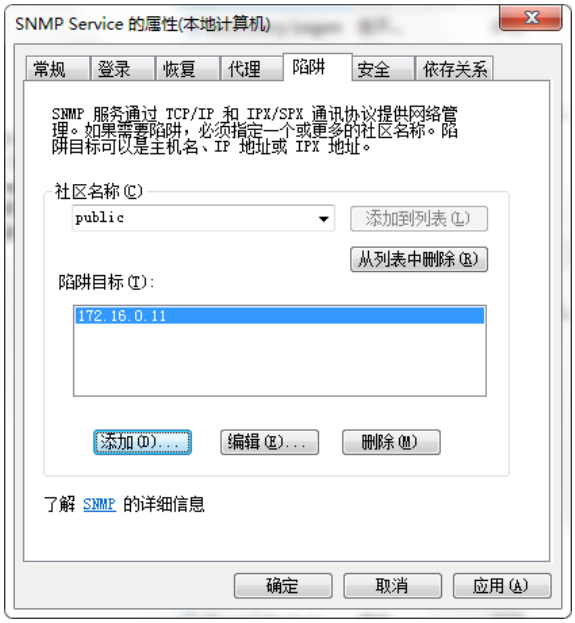


图 1-22SNMPService 属性

(2)在属性页集合中找到“安全”属性页；选中“接受来自这些主机的 SNMP 包”项；点击[添加(D)...]按钮来设置 SNMP 管理器的 IP 地址（注意此地址使用一个非组内主机的 IP 地址）。如下图所示设置：

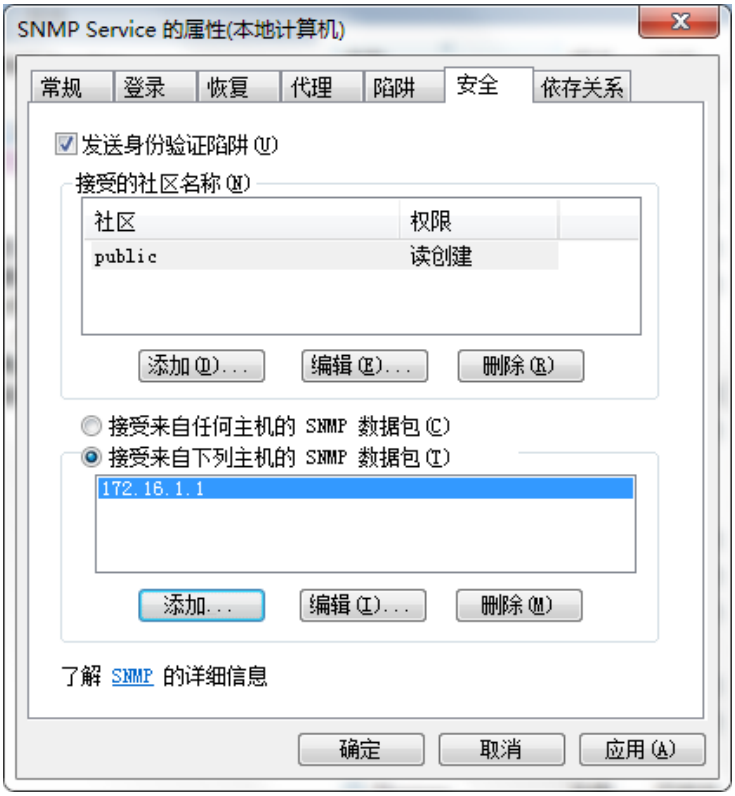
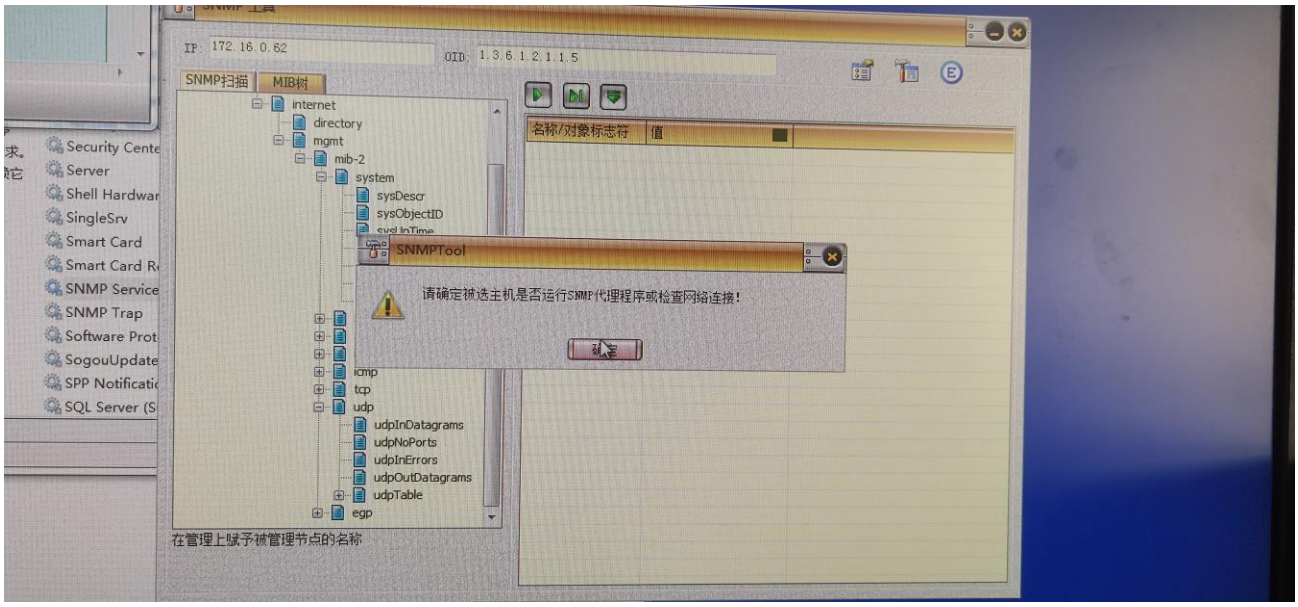


图 1-23SNMPService 属性

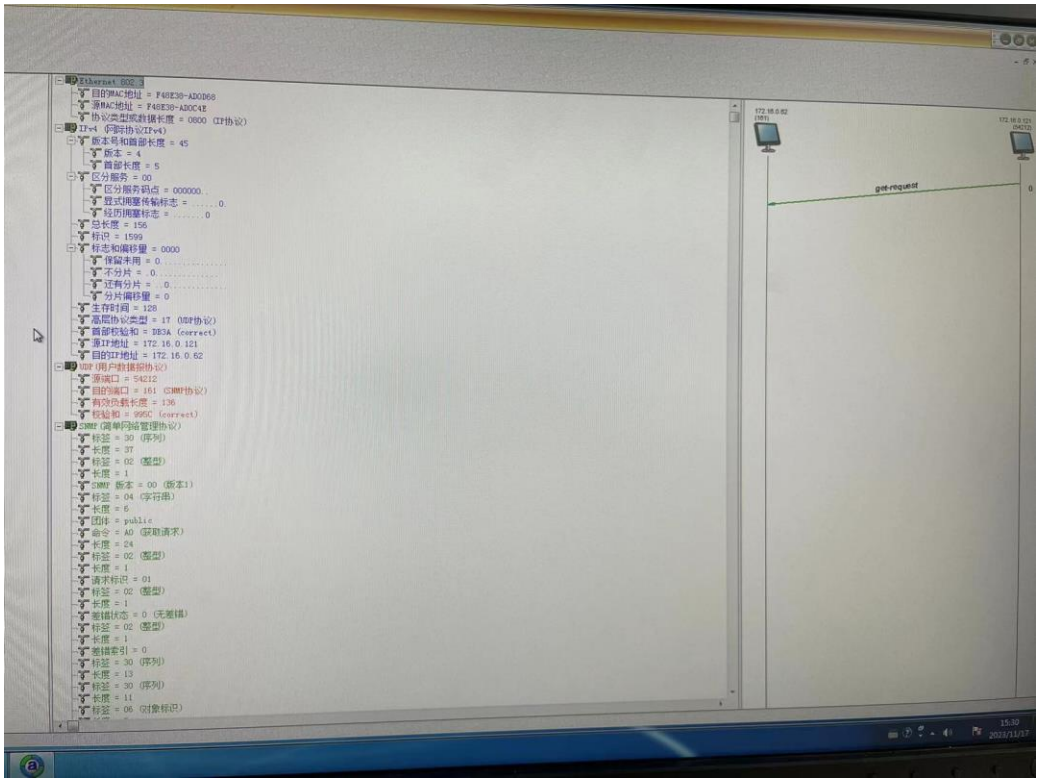
2. 主机 B 启动协议分析器开始捕获数据并设置过滤条件（提取 SNMP 协议）。

3. 主机 A 启动“实验平台工具栏中的 SNMP 工具”，在“IP”文本框中输入主机 B 的 IP 地址。

4. 主机 A 通过 SNMP 工具尝试获取主机 B（SNMP 代理服务器）的信息。



主机 B 停止捕获数据，并分析捕获到的数据。



5.关闭 SNMP 工具。

五、 实验结果总结

思考问题:

1. SNMP 使用 UDP 协议进行封装，分析为什么不使用 TCP 进行封装？

   答：UDP 的开销较 TCP 要小的多。

2. 为什么 SNMP 的管理进程使用探询掌握全网状态属于正常情况，而代理进程用陷阱向管理进程报告属于较少发生的异常情况？

   答：SNMP 的功能通过探询操作来实现，即 SNMP 管理进程定时向被管理设备周期性的发送探询信息。时间间隔可通过 SNMP 的管理信息库 MIB 来建立。
   探询的好处是
   （1）可以使系统相对简单。
   （2）能限制通过网络所产生的管理信息的通信量。但 SNMP 不是完全的探询协议，它允许不经过询问就能发送某些信息。这种信息成为陷阱。
   陷阱的好处是
   （1）仅在严重事件发生时才发送陷阱。
   （2）陷阱信息很简单且所需字节数很少。

3. 假如你是网络管理人员，你能否通过 SNMP 协议和以前所学的知识，实现网络拓扑发现？

   答：通过以前所学知识和 SNMP 协议结合能了解简单的网络拓扑。结合 ping、traceroute 以及 SNMP 的 MIB 树上的结点信息，可以对网络拓扑进行推测。