Exp 23：Firewall

目的： 了解防火墙(Firewall)的运行模式及如何执行。

摘要： 本实验将介绍防止遭受网络黑客攻击的防火墙的作业模式及其理论架构。另外实验中也借着MDDL程序语言，解析防火墙机制的作业流程及原理。

时间： 4.5 hrs。

一、网络拓扑

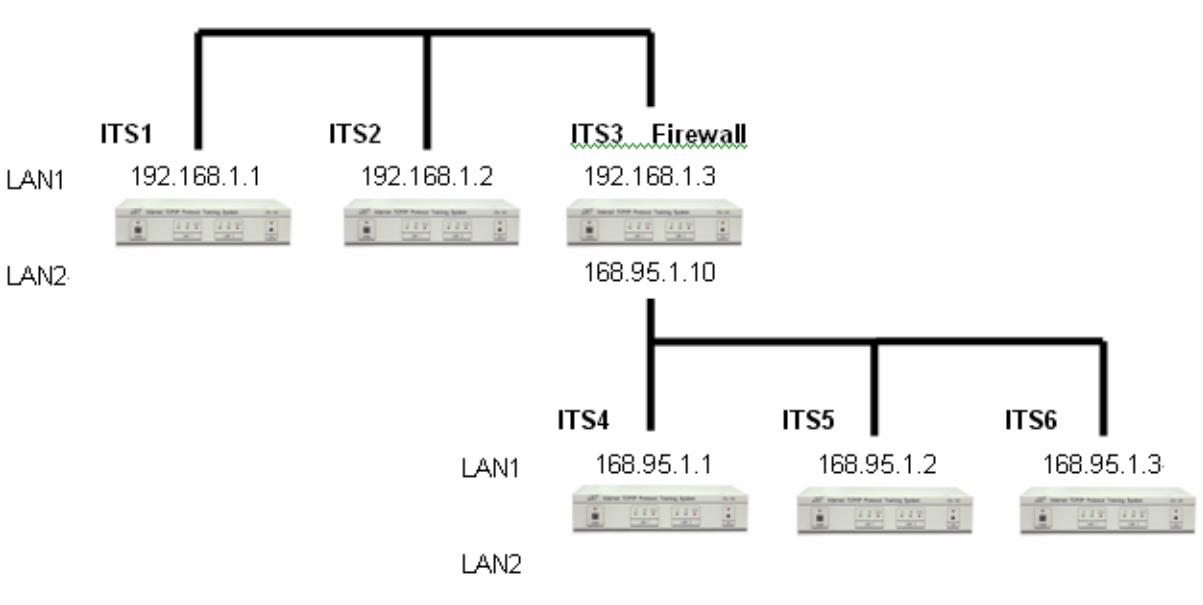

图 23.1

二、技术背景

1、Firewall

防火墙(firewall)是由一群相关联的程序所组成，通常位于与网络网关(gateway)上，主要功用就是在网关前把欲侵入或破坏未开放资源的黑客阻绝在外。一般来说：

1）防火墙可能是一只程序或一个装置用来控制一台计算机或一个网络的门禁，有点像大楼的保全一般的功能。

2）防火墙随时监视并限制进出系统的网络联机。

3）防火墙是挡在系统与英特网中间，所以所有的进出都需经过它。

4）防火墙会盘查所有进出的数据及封包。

5）如果防火墙接获一未遵循防火墙订定规则的封包时，防火墙会毫不客气的将它挡

在门外。

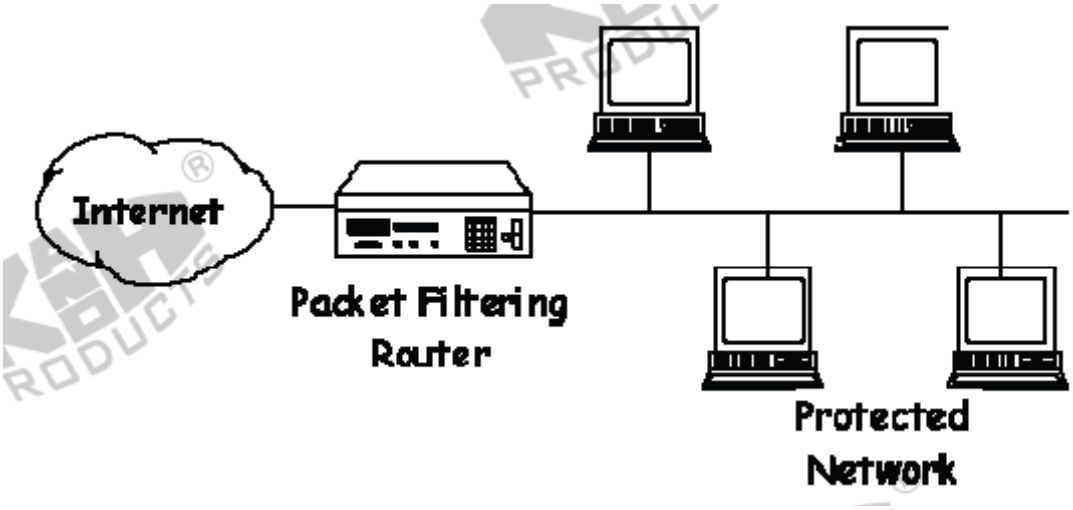

图 23.2

2、Packet Filtering

封包过滤(packet filtering)是防火墙中一般最常见的防火墙技术，顾名思义此技术会建立一些规则来过滤数种通讯协议的封包，限制其进或出。一般来说，防火墙会把封包内的信息拿来和其规则做比较，封包内的信息通常包含：IP地址、协议种类(protocol type)、TCP或UDP的通讯端口号码、实体地址(hardware address)…等等，透过这些信息的比对，防火墙就可以判断要让封包通过或阻挡在外。举几个简单的例子：防火墙透过比对，允许所有IP地址都可以传送封包给邮件服务器的TCP 25号通讯端口，也就是表示开放使用SMTP传送e-mail；又或者将所有欲连接至TCP 23号通讯端口的封包拦截下来，也就表示禁示了Telnet功能。

三、实验步骤

1、了解网络拓扑

1）在Hubox上将网络连接如图23.3所示。

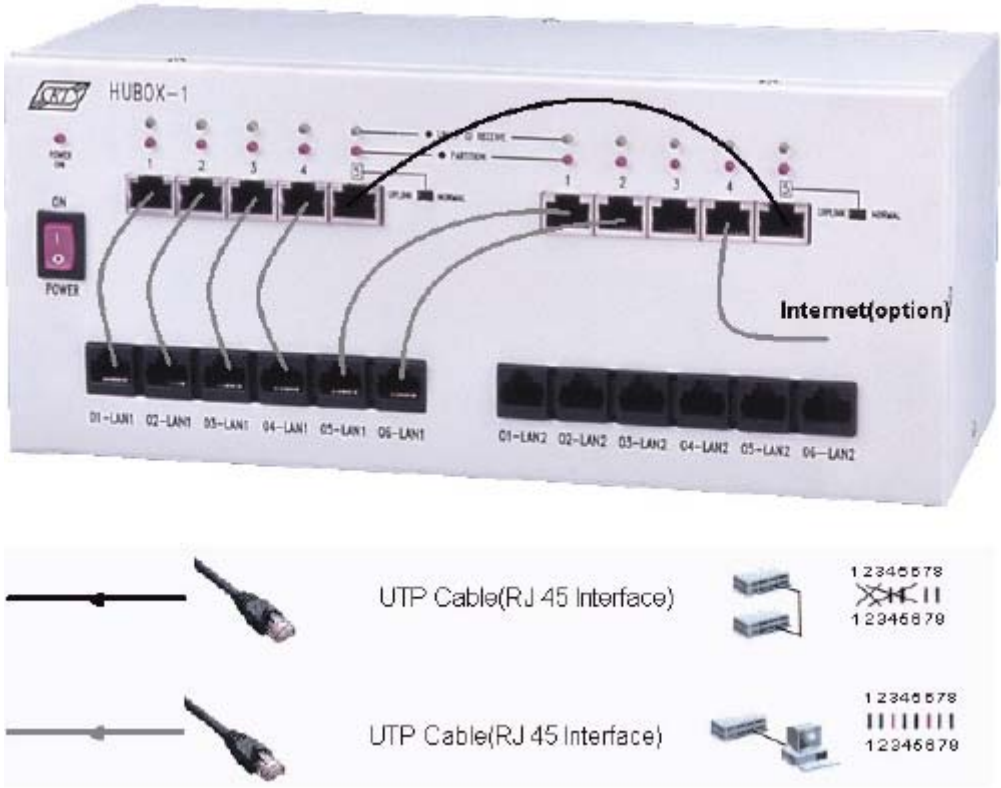

图 23.3

2、Firewall Rules of IP Address Filter 实验

A. 初始设置：

2）执行 **XCLIENT.BAT，**打开 ITS 应用软件 KCodes Network Explorer。

3）从 Tool 菜单部分打开“网络配置”对话框（**Network Configuration**） 见图 23.4。

ITS1 设置如下：

4）定义 Interface 1 的 IP 地址为 “**192.168.1.1**”。在路由表部分（Routing Table），“Destination ”and “Mask” 部分输入“**0.0.0.0**”；“Gateway” 部分输入“**192.168.1.3**”；模式设定为“**Host”**, 然后点击“**Set & Close**”按钮。

ITS2 设置如下：

5）定义 Interface 1 的 IP 地址为“**192.168.1.2**” 。在路由表部分（Routing Table），“Destination ”and “Mask” 部分输入“**0.0.0.0**”；“Gateway” 部分输入“**192.168.1.3**”； 模式设定为“**Host”**, 然后点击“**Set & Close**”按钮。

ITS3 (Firewall) 设置如下：

6）定义 Interface 1 的 IP 地址为“**192.168.1.3**” ，定义 Interface2 的 IP 地址为“**168.95.1.10**” 模式设定为“**Gateway”**, 然后点击“**Set & Close**”按钮。

ITS4 设置如下：

7）定义 Interface 1 的 IP 地址为“**168.95.1.1**”。 在路由表部分（Routing Table），“Destination ”and “Mask” 部分输入“**0.0.0.0**”；“Gateway” 部分输入“**168.95.1.10**”；模式设定为“**Host”**，然后点击“**Set & Close**”按钮。

ITS5 设置如下：

8）定义 Interface 1 的 IP 地址为”**168.95.1.2**”。 在路由表部分（Routing Table），“Destination ”and “Mask” 部分输入“**0.0.0.0**”；“Gateway” 部分输入“**168.95.1.10**”；模式设定为“**Host”**，然后点击“**Set & Close**”按钮。

ITS6 设置如下：

9）定义 Interface 1 的 IP 地址为“**168.95.1.3**”。 在路由表部分（Routing Table），“Destination ”and “Mask” 部分输入“**0.0.0.0**”；“Gateway” 部分输入“**168.95.1.10**”；模式设定为“**Host”**，然后点击“**Set & Close**”按钮。

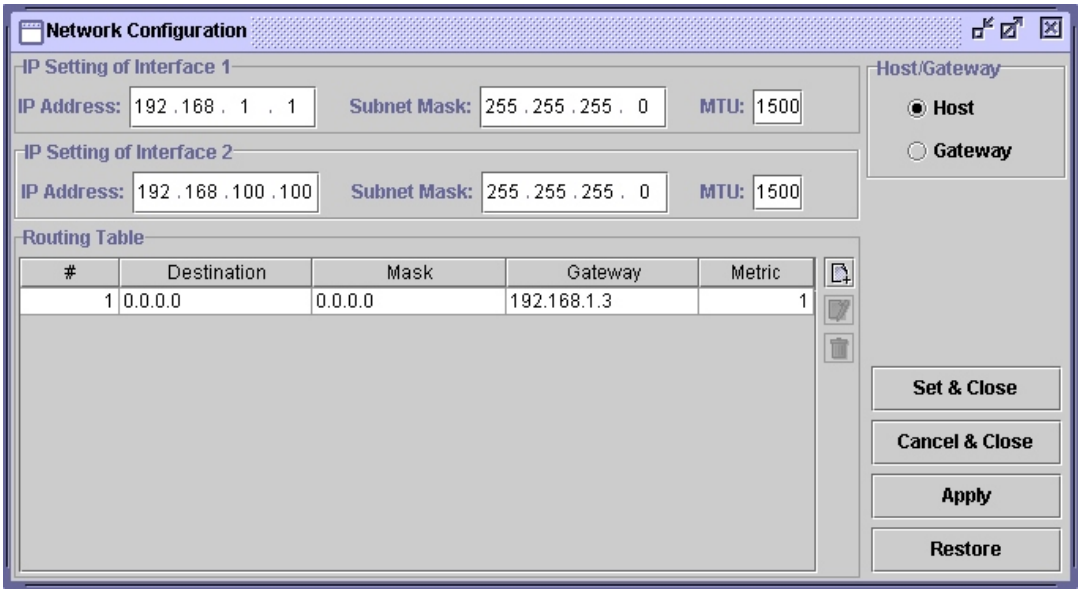

图 23.4

B. Reject by Firewall 实验

ITS 3 设置如下：

10）打开网络封包浏览器（Network Message Browser）. 检查是否打开 **Listening**。

11）打开 MDDL 编辑界面。

12）点击 **Load** 按钮。调用 C: \XClient \Data \Mddl \Tutorial \Ex23 \Firewall.mddl，然后点击 **Upld** 按钮。

ITS 1, 2, 4, 5 and 6 设置如下：

13）打开网络封包浏览器（Network Message Browser）．检查是否打开 **Listening**。

14）根据前面的实验操作，ITS4 打开“TCP Session”界面，设定 Source IP Address 为 **168.95.1.1** 且 Source port 为 **21**，然后点击 **Listen** 按钮。由 ITS1 打开“TCP Session”界面，设定 destination IP address 为 ITS4 的 IP（**168.95.1.1**） 且 destination port 为 **21** (见图 23.5)。然后点击 **Connect** 按钮。我们可以看见 ITS3 作为防火墙将会阻止这个 IP 的请求封包，见图 23.6。

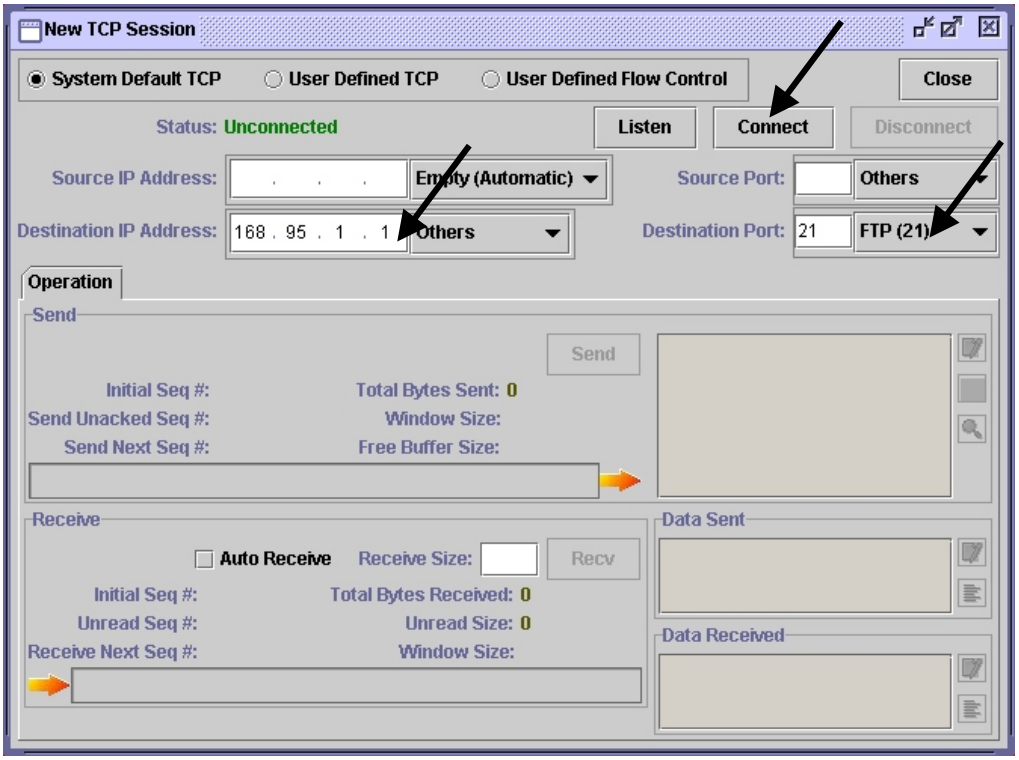

图 23.5

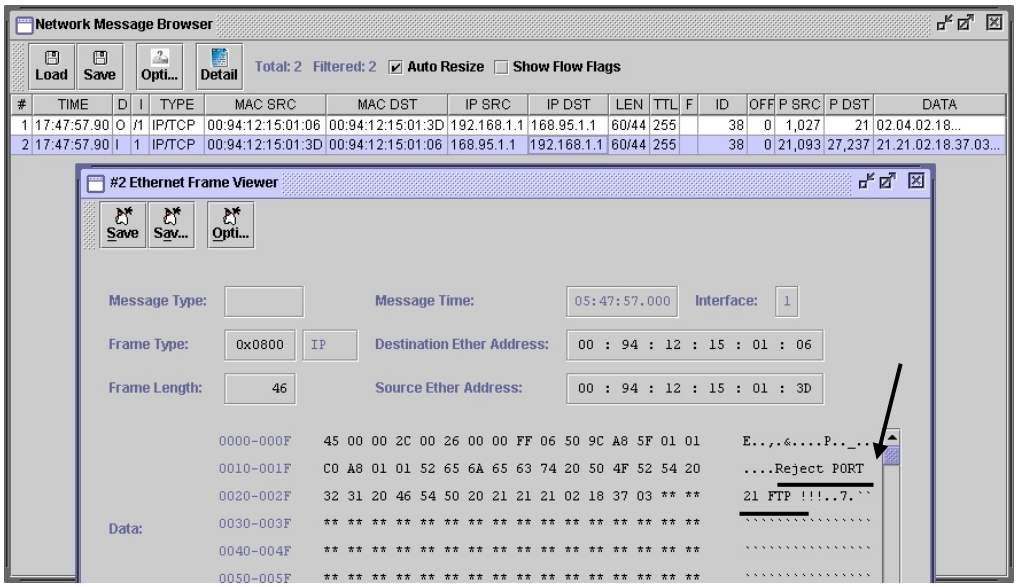

图 23.6

15）根据前面的实验操作，ITS4 发送“ICMP Echo Request”至 ITS1。我们可以看见 ITS3 作为防火墙将会阻止这个 IP 的请求封包。见图 23.7。

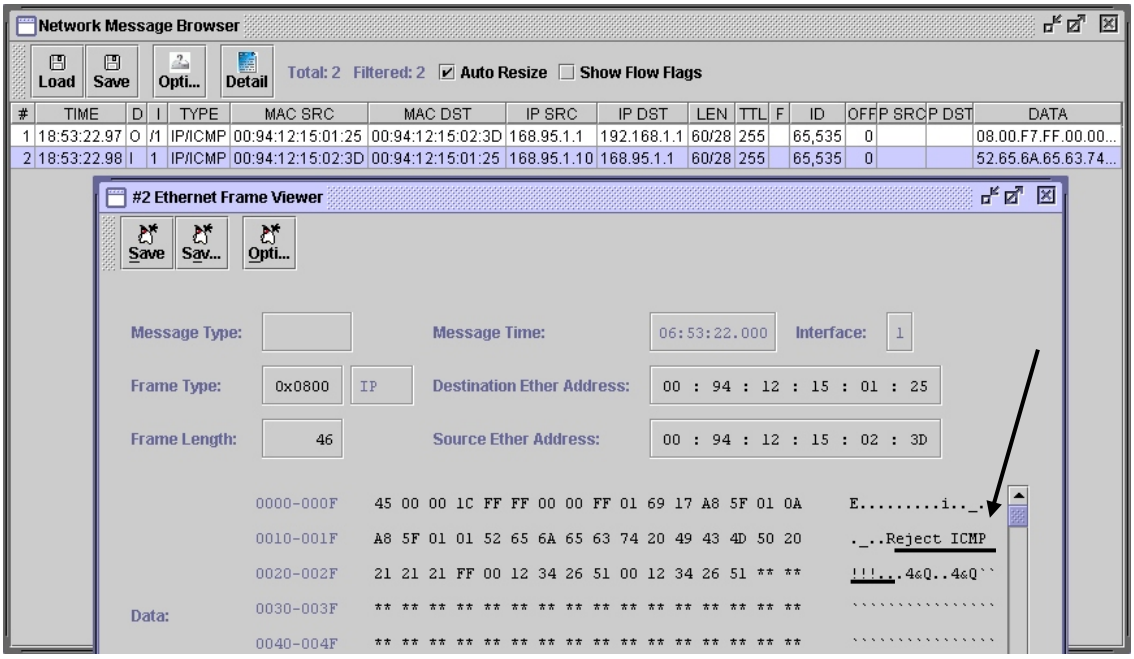

图 23.7

四、实验讨论

1、什么是「IP 欺骗(IP spoofing)」？我们是不是可以用ITS设计一个防火墙解决这个问题？

提示：「IP欺骗」是一种黑客使用TCP/IP封包入侵他人计算机的高阶黑客技巧，主要是改更封包的来源端的IP地址，让整个攻击封包看起来像是来自可信任的网域。同时一般路由器是依据「目的端IP地址(destination IP address)」来传递数据包，而会忽略来源端IP地址，这都是防火墙可以再加以细心处理的项目。

REACTOR PROGRAM

1、Firewall.mddl

```
// Firewall
// Reject Port 80/21 from interface 1
// Reject ICMP from interface 2

ETHER_IN_HANDLER 1
```

```
{
    IF(S.ETHER_TYPE==CNST_ETHER_TYPE_IP)
    {
            IF(S.ETHER_DATA.[22,23]== 80W)   // REJECT   PORT 80     HTTP
           {
                DISCARD_MESSAGE;
                SEND_OUT_ETHER_FROM_INTERFACE 1 WITH_DATA
     {
     T = S ,
     T.ETHER_MACADDRDST          = S.ETHER_MACADDRSRC             ,
     T.ETHER_MACADDRSRC          = MYMAC(1)              ,
     T.ETHER_DATA.IP_ADDRSRC = S.ETHER_DATA.IP_ADDRDST ,
     T.ETHER_DATA.IP_ADDRDST =    S.ETHER_DATA.IP_ADDRSRC ,
     T.ETHER_DATA.IP_DATA        = "Reject PORT 80 HTTP!!!"        ,
     T.ETHER_DATA.IP_HEADERCHKSUM = 0W                                                ,
     T.ETHER_DATA.IP_HEADERCHKSUM = CHECKSUM(T.ETHER_DATA.IP_HEADER)
      }
           }
           ELSE IF(S.ETHER_DATA.[22,23]== 21W)   // REJECT   PORT 21   FTP
           {
                DISCARD_MESSAGE;
                SEND_OUT_ETHER_FROM_INTERFACE 1 WITH_DATA
      {
      T = S ,
     T.ETHER_MACADDRDST          = S.ETHER_MACADDRSRC             ,
      T.ETHER_MACADDRSRC          = MYMAC(1)              ,
     T.ETHER_DATA.IP_ADDRSRC = S.ETHER_DATA.IP_ADDRDST ,
     T.ETHER_DATA.IP_ADDRDST =    S.ETHER_DATA.IP_ADDRSRC ,
     T.ETHER_DATA.IP_DATA        = "Reject PORT 21 FTP !!!"        ,
     T.ETHER_DATA.IP_HEADERCHKSUM = 0W                                                ,
```

```
        T.ETHER_DATA.IP_HEADERCHKSUM = CHECKSUM(T.ETHER_DATA.IP_HEADER)
          }
                }
      }
}

ETHER_IN_HANDLER 2
{
      IF(S.ETHER_TYPE==CNST_ETHER_TYPE_IP)
      {
                  IF(S.ETHER_DATA.IP_PROT==CNST_IP_PROT_ICMP)
                  {
                      DISCARD_MESSAGE;
                      SEND_OUT_ETHER_FROM_INTERFACE 2 WITH_DATA
          {
          T = S ,
        T.ETHER_MACADDRDST          = S.ETHER_MACADDRSRC              ,
          T.ETHER_MACADDRSRC        = MYMAC(1)              ,
        T.ETHER_DATA.IP_ADDRSRC = MYIP(2) ,
        T.ETHER_DATA.IP_ADDRDST =    S.ETHER_DATA.IP_ADDRSRC ,
        T.ETHER_DATA.IP_DATA        = "Reject ICMP !!!"         ,
        T.ETHER_DATA.IP_HEADERCHKSUM = 0W                       ,
        T.ETHER_DATA.IP_HEADERCHKSUM = CHECKSUM(T.ETHER_DATA.IP_HEADER)
          }
                  }
      }
}
```