**第三次网安综合课程设计报告**

**（TCP/IP Attack Lab）**

57117228 陈佑

IP地址：

VM A 10.0.2.4 攻击者

VM B 10.0.2.5 受害者

VM M 10.0.2.6 观察者

# SYN Flooding Attack（SYN泛洪攻击）

SYN Flooding Attack——一种DoS攻击形式，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，攻击者要么使用猜测的IP地址或不继续流程。 通过这次攻击，攻击者可以使受害者的队列中全是半打开的连接，即已完成SYN、SYN-ACK但尚未收到最后的ACK。

当此队列已满时，受害者不能再连接。

在此任务中，您需要演示SYN泛洪攻击。您可以使用Netwox工具进行攻击，然后使用嗅探工具捕获攻击数据包。在攻击进行的时候，在受害者机器上运行“netstat -na”命令，并将结果与攻击前的结果进行比较。

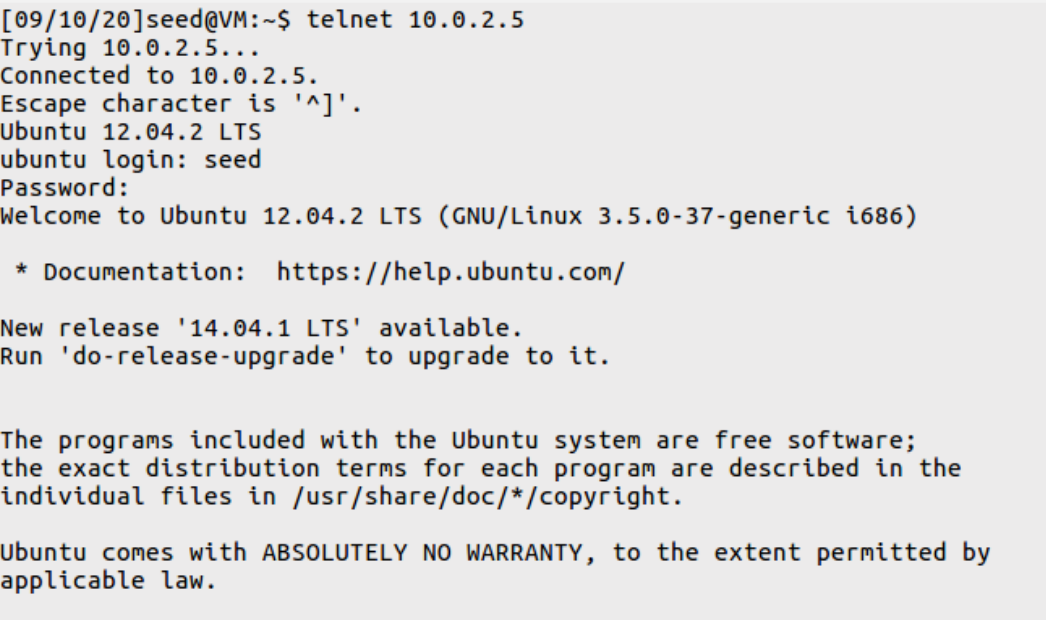
## **1A. 关闭SYN Cookie Countermeasure，进行攻击**

### Step1. 关闭SYN Cookie Countermeasure

STSH%1Y0Q7%9Y[FPWM1[FPX

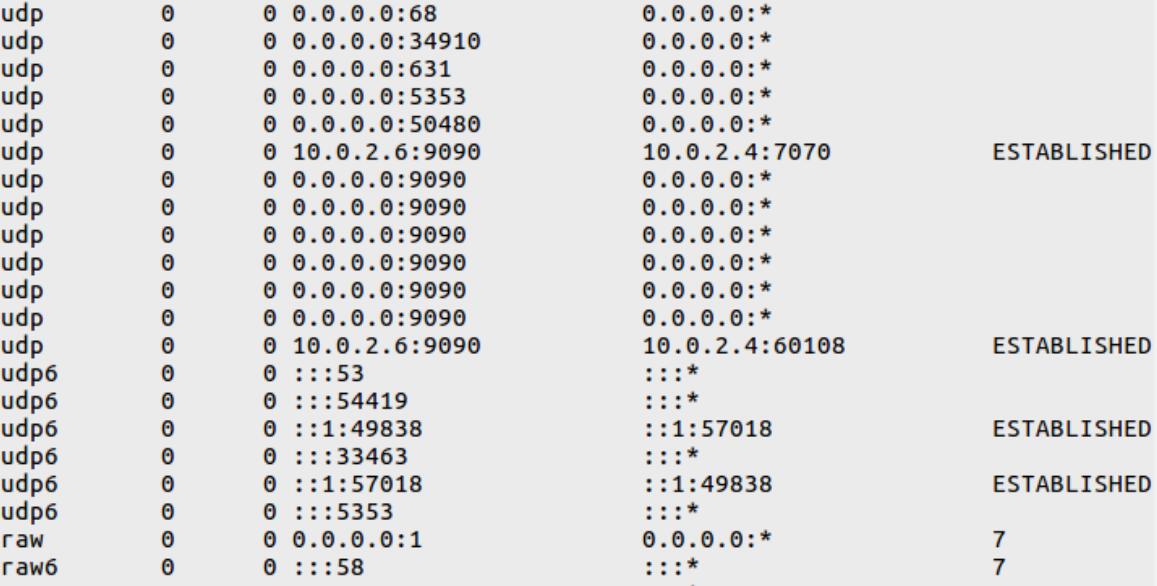
### Step2. 观察者测试受害者的telnet连接是否可用，用观察受害者的连接队列

如上图所示，攻击开始前，观察者VM M可以成功与受害者建立telnet连接。



观察此时受害者的连接队列，发现此时有几个建立的连接，也有的连接处于LISTEN状态。





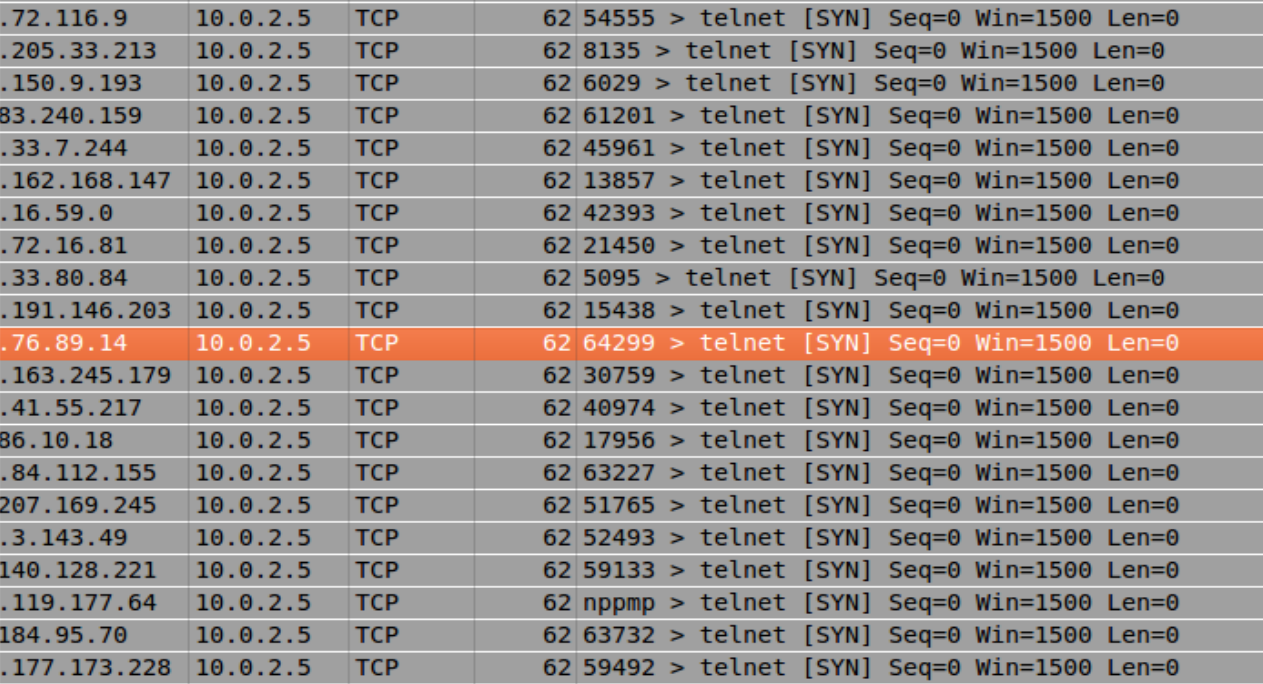
### Step3. 攻击者利用Netwox 76对受害者进行SYN洪泛攻击

Netwox编号为76的工具提供了SYN 洪泛攻击的基本功能。

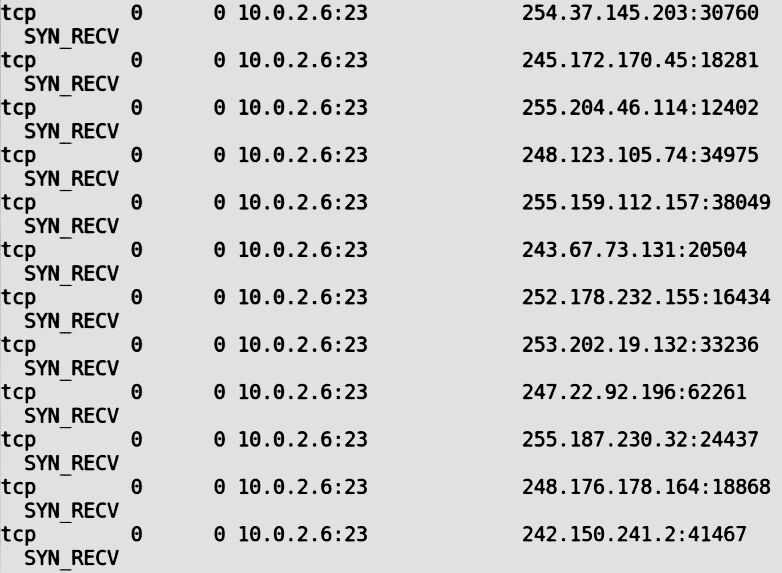
在攻击者VM A上针对受害者的23端口（telnet服务）执行攻击指令：

BM{4AL[WT`BOTHDEW@~2E4H

受害者VM B上打开wireshark抓包，可以看到，受害者收到数量极大的SYN报文，但是并未收到最终的ACK报文，因而其连接队列中有了许多半连接状态的TCP连接。

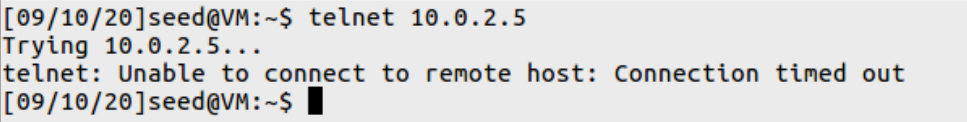


查看受害者当前连接情况，发现有很多半开放连接。



### Step4. 观察者再次测试受害者的telnet连接是否可用

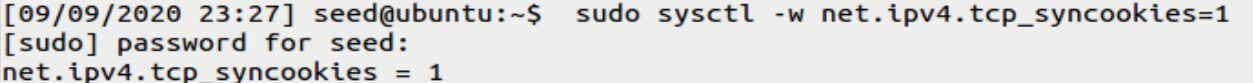
观察者VM M再次尝试与受害者建立telnet连接，发现此时无法建立连接。



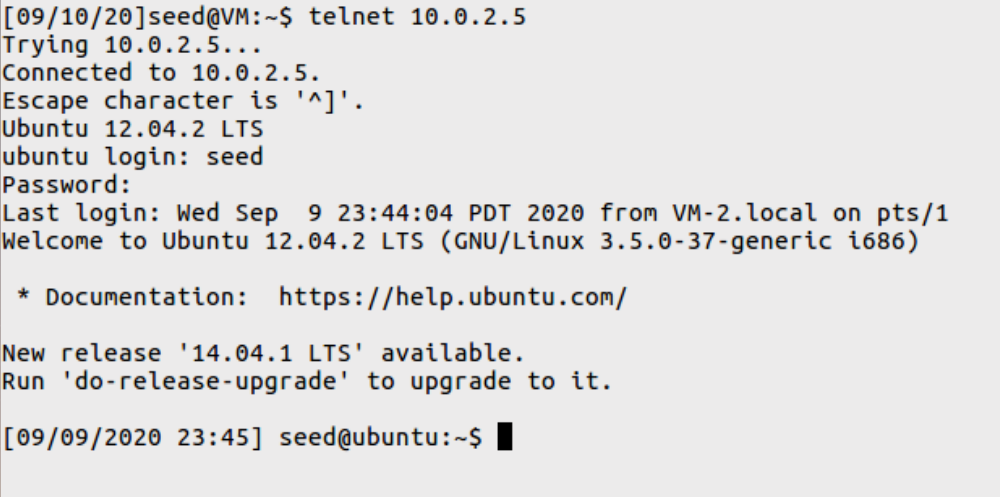
这是因为，被害者的连接队列有固定大小。当这个队列被SYN洪泛攻击造成的半连接填满时，受害者不能再与其他用户建立TCP连接。这说明在SYN cookie Countermeasure机制关闭的前提下，SYN泛洪攻击成功。

## **1B. 打开SYN Cookie Countermeasure，进行攻击**

SYN cookie Countermeasure可以对抗SYN洪泛攻击，在其打开的状态下重新执行SYN攻击。

****

在VM A上执行攻击后，观察者VM M再次尝试与受害者建立telnet连接，发现可以成功连接，说明在SYN cookie Countermeasure机制开启后，SYN泛洪攻击失败。



## **1C. SYN Cookie为什么可以防止SYN泛洪攻击**

在服务器接收到SYN包之后，它会使用只有服务器才知道的密钥，根据包中的信息计算一个哈希值（H）。**哈希值（H）作为服务器的初始序列号发送到客户端，这个H就被称为SYN cookie。**

如果客户端是攻击者，那么攻击者不会返回SYN ACK报文，没有返回就说明对方为攻击者，不会建立socket资源；如果客户端不是攻击者，那么它就会在ack处填上H+1返回一个SYN ACK报文给服务器，服务器通过重新计算H，来确定ack中的数是否正确，若正确，则再建立合法连接。因而，SYN cookie可以有效防止SYN泛洪攻击。

# TCP RST Attacks on telnet and ssh Connections（TCP RST攻击）

RST表示复位，用来异常的关闭连接，在TCP的设计中它是不可或缺的。**发送RST包关闭连接时，不必等缓冲区的包都发出去，直接就丢弃缓存区的包发送RST包。而接收端收到RST包后，也不必发送ACK包来确认。**

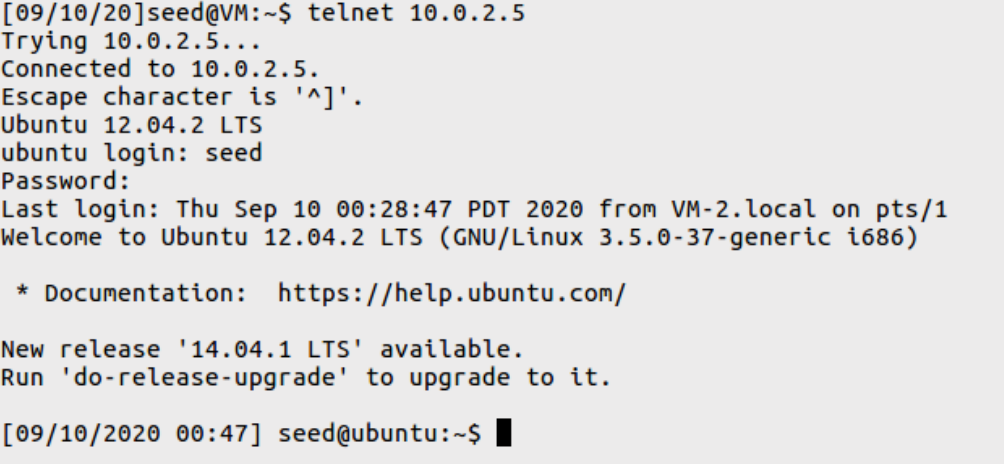
TCP RST攻击可以终止两个受害者之间建立的TCP连接。例如，**如果两个用户A和B之间建立了Telnet连接（TCP），则攻击者可以将A到B的RST数据包进行欺骗，从而破坏现有的连接。**为了在这次攻击中取得成功，攻击者需要正确构造TCP RST数据包。

Netwox编号为78的工具提供了RST攻击的基本功能，输入“netwox 78 –help”可以获取帮助信息。

攻击目的——使用TCP RST攻击来破坏A和B之间现有的telnet连接。之后，对ssh连接尝试相同的攻击。

## **2A. 使用Netwox 78对telnet进行攻击**

### Step1. 观察者先与受害者建立telnet连接

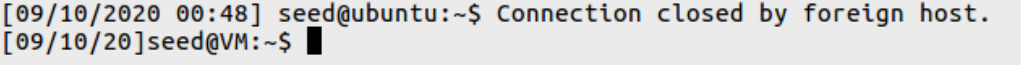


### Step2. 攻击者进行TCP RST攻击

攻击者向受害者发送RST报文，企图中断观察者和受害者间的telnet连接。

UU6BVU4F$G@BS@EOP[{%H~1

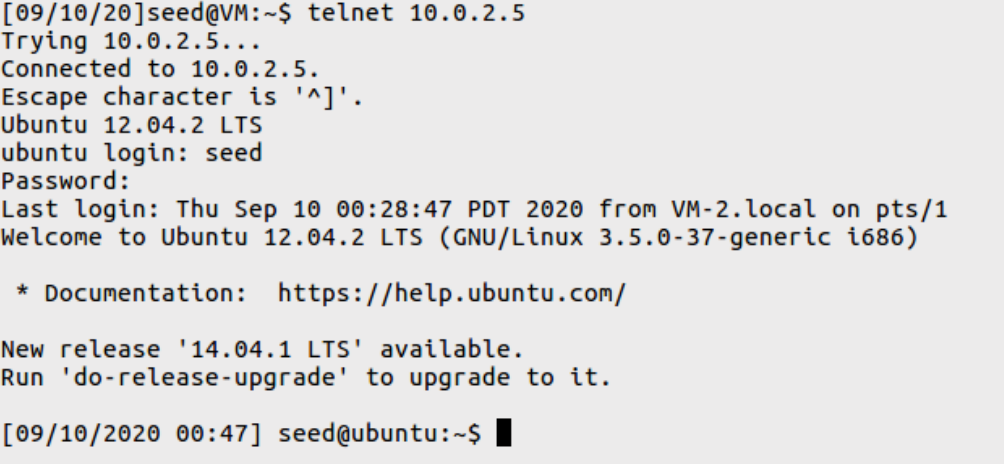
### Step3. 观察者与受害者建立的telnet连接被终止



可以看到，攻击者执行攻击指令后，观察者与受害者之间的telnet连接终止，TCP RST攻击成功。

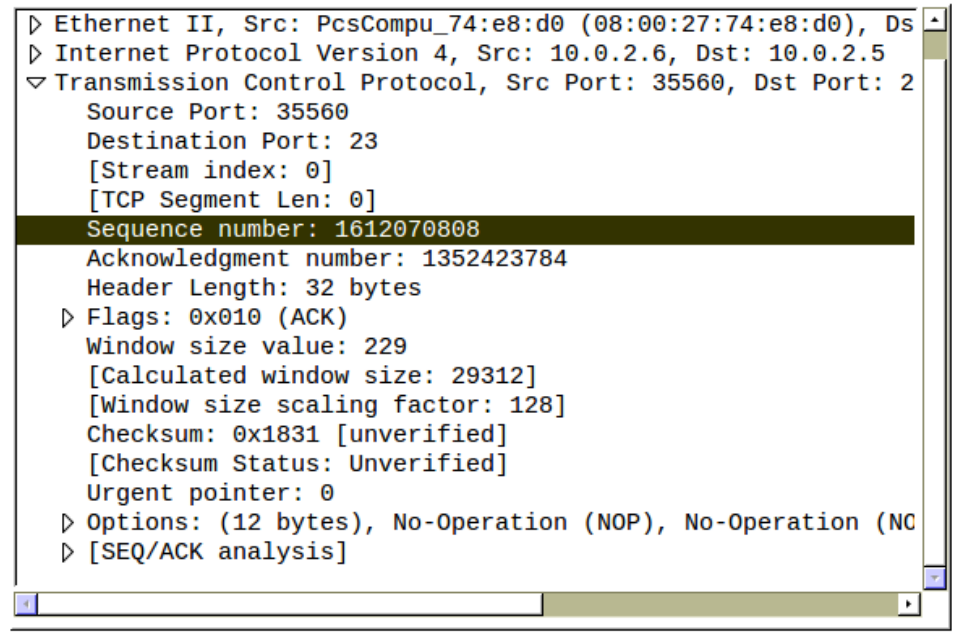
## **2B. 使用scapy对telnet进行攻击**

### Step1. 观察者先与受害者建立telnet连接

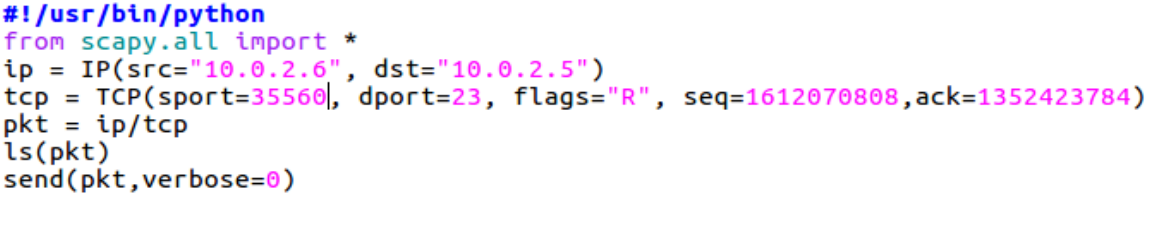


### Step2. 攻击者利用scapy进行TCP RST攻击

攻击者首先用wireshark抓观察者与受害者之间的ssh连接建立数据包，最后一个报文的数据如下：

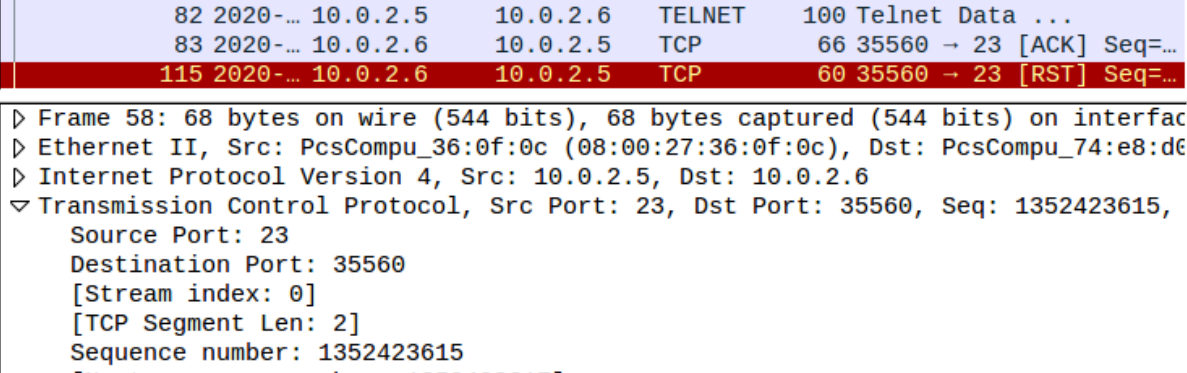


攻击者根据此报文的源地址、宿地址、原宿端口，seq和ack，伪造观察者发往受害者的RST报文，企图中断观察者和受害者间的telnet连接。

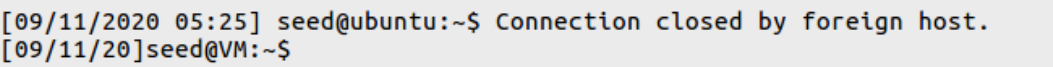


### Step3. 观察者与受害者建立的telnet连接被终止

攻击者用wireshark抓包，可以成功抓到自己伪造的由观察者发送给受害者的RST数据包。

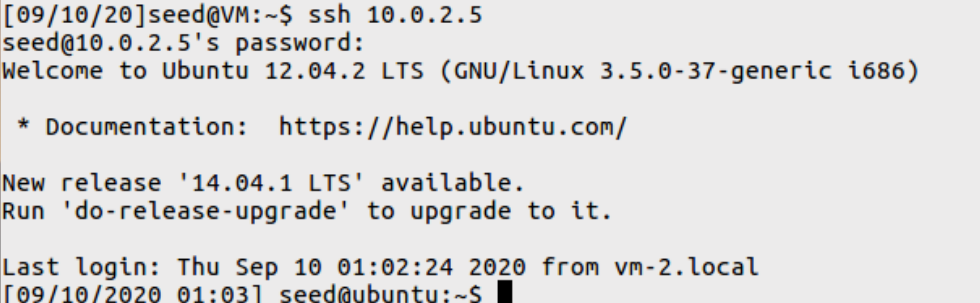


观察者的终端显示如下图所示，观察者与受害者之间的telnet连接终止，针对telnet连接的TCP RST攻击成功。



## **2C. 使用Netwox 78对ssh进行攻击**

### Step1. 观察者先与受害者建立ssh连接

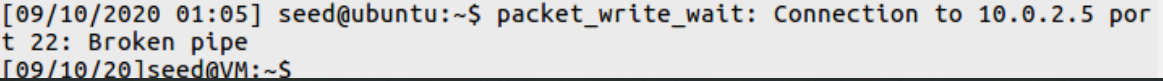


### Step2. 攻击者进行TCP RST攻击

攻击者向受害者发送RST报文，企图中断观察者和受害者间的ssh连接。

UU6BVU4F$G@BS@EOP[{%H~1

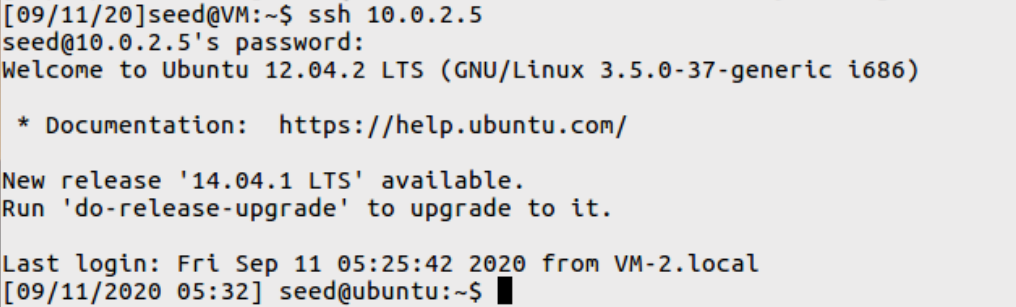
### Step3. 观察者与受害者建立的ssh连接被终止



可以看到，攻击者执行攻击指令后，观察者与受害者之间的ssh连接终止，TCP RST攻击成功。

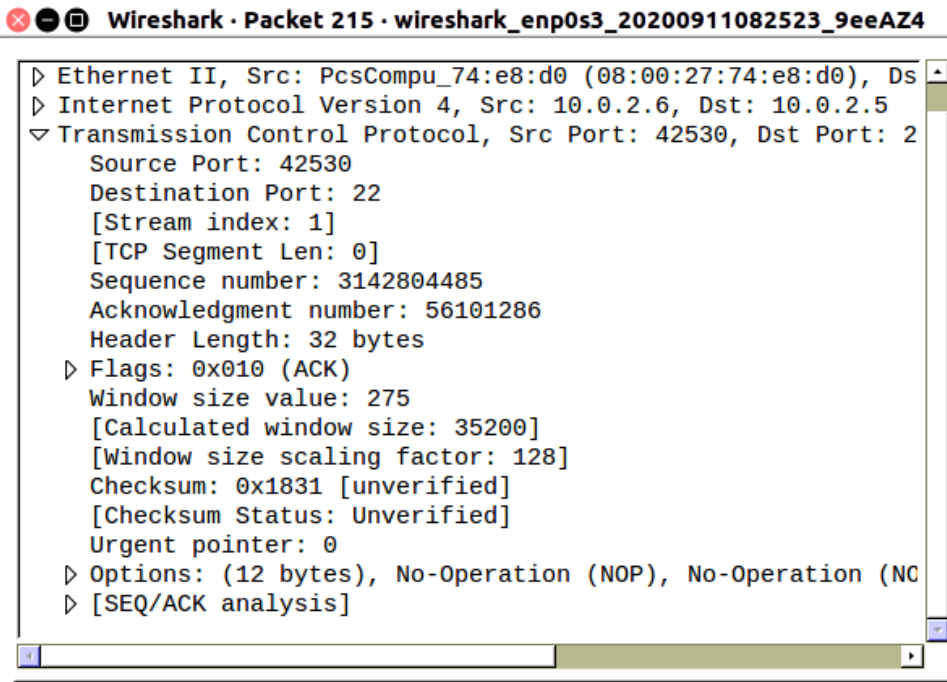
## **2D. 使用scapy对ssh进行攻击**

### Step1. 观察者先与受害者建立ssh连接

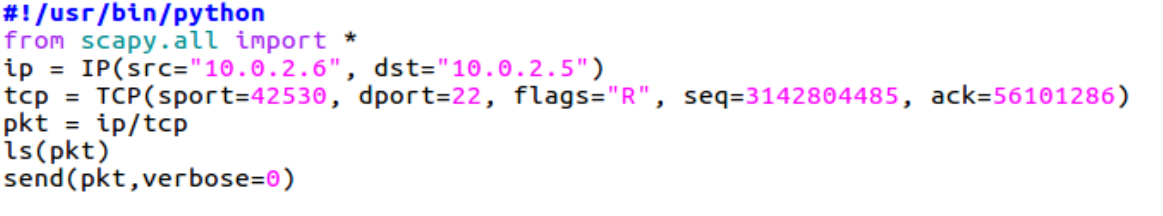


### Step2. 攻击者利用scapy进行TCP RST攻击

攻击者首先用wireshark抓观察者与受害者之间的ssh连接建立数据包，最后一个报文的数据如下：



攻击者根据此报文的源地址、宿地址、原宿端口，seq和ack，伪造观察者发往受害者的RST报文，企图中断观察者和受害者间的ssh连接。

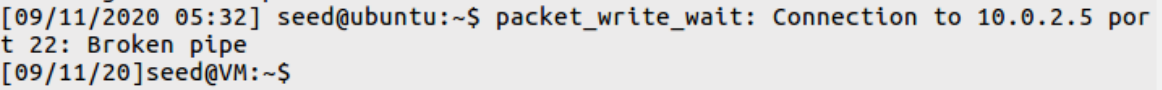


### Step3. 观察者与受害者建立的ssh连接被终止

攻击者用wireshark抓包，可以成功抓到自己伪造的由观察者发送给受害者的RST数据包。



可以看到，攻击者执行攻击指令后，观察者与受害者之间的ssh连接终止，针对ssh连接的TCP RST攻击成功。



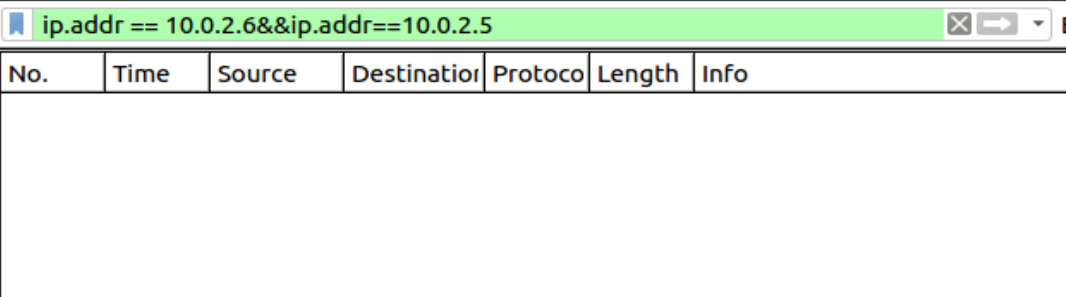
# TCP Session Hijacking（会话劫持）

**TCP会话劫持攻击的目的是劫持一个现有的TCP连接(会话)，通过向会话中注入恶意内容，来实现攻击。**如果此连接是telnet会话，攻击者可向此会话注入恶意命令(如删除重要文件)，导致受害者执行恶意命令。

**在这个任务中，您需要演示如何劫持两台计算机之间的telnet会话。你的目标是得到telnet服务器运行来自您的恶意命令。**为简单起见，我们假设攻击者和受害者在同一局域网内。

## **4A. 使用Netwox 40进行攻击**

### Step1. 首先在攻击者主机VM A上开启wireshark，过滤器设置如下

********

攻击者只抓受害者与观察者之间通信的报文。

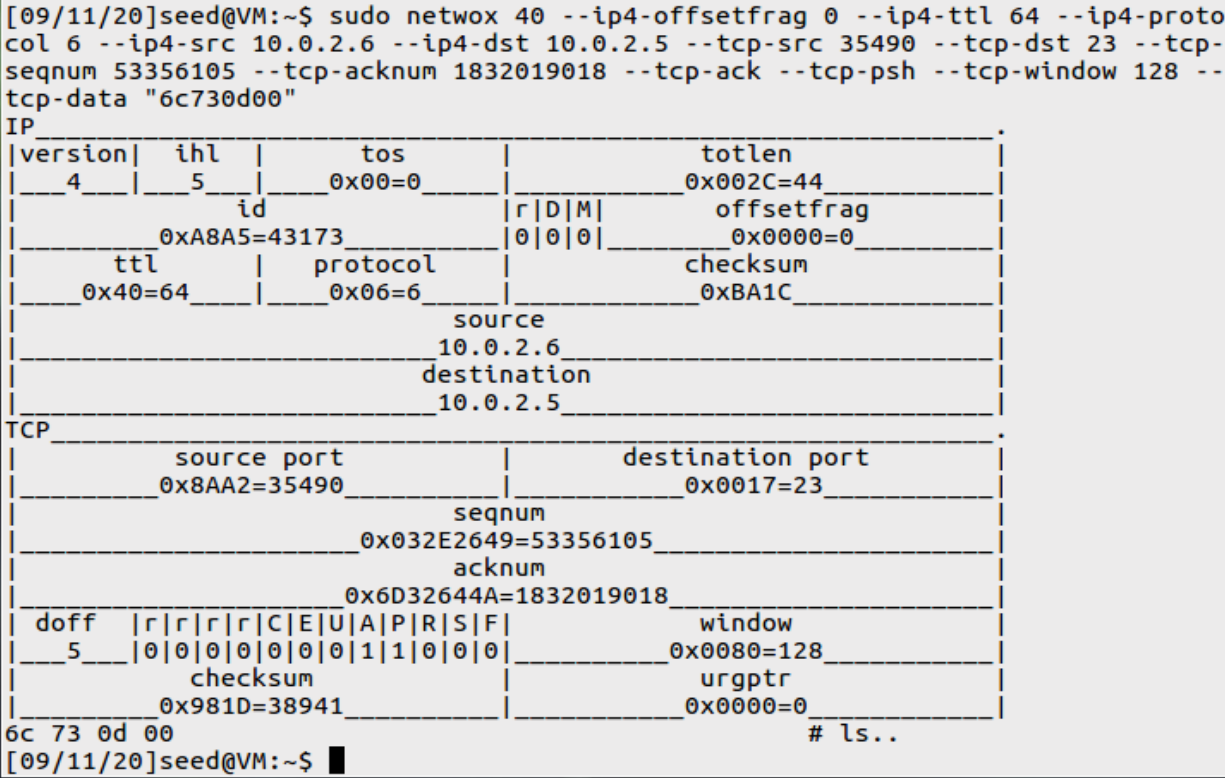
### Step2. 观察者与被害者之间建立telnet会话，同时攻击者抓包

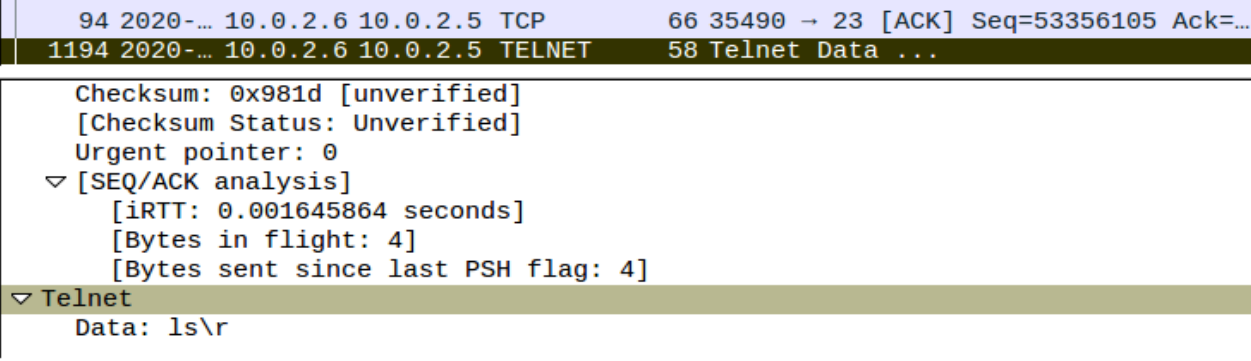


攻击者观察最后一个数据包（从观察者发往受害者）的源ip、目标ip、源端口、目的端口、sequence number和ack number，将它们填入netwox伪造报文的对应位置。

### Step3. 攻击者利用Netwox 40攻击受害者，伪造一个来自于观察者的报文发送给受害者

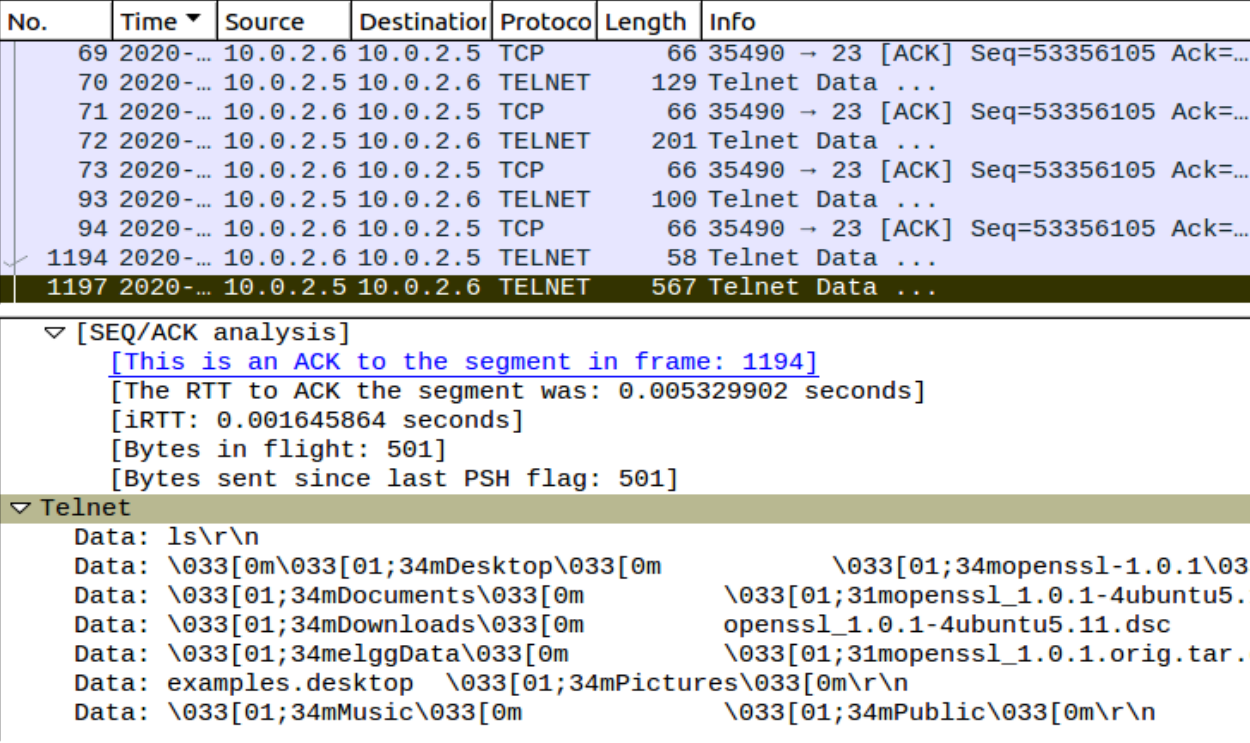
攻击者利用指令sudo netwox 40 --ip4-offsetfrag 0 --ip4-ttl 64 --ip4-protocol 6 --ip4-src 10.0.2.6 --ip4-dst 10.0.2.5 --tcp-src 35490 --tcp-dst 23 --tcp-seqnum 53356105 --tcp-acknum 1832019018 --tcp-ack --tcp-psh --tcp-window 128 --tcp-data "6c730d00"，来伪造一个从观察者发往受害者的包（从而伪装成观察者与受害者通信）。包的data部分为”6c730d00”，是指令”ls\r”的十六进制表示。





上图即为攻击者伪造的报文，可以看到，其数据部分为”ls\r”。

报文发出去后，攻击者在wireshark上抓到了受害者发给观察者（其实是假扮成观察者的攻击者）的响应报文，报文的数据部分是受害者执行指令”ls\r”的结果（输出目录下所有的文件）。

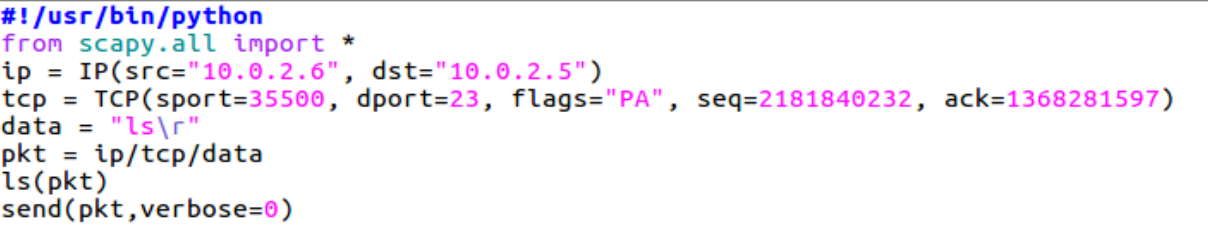


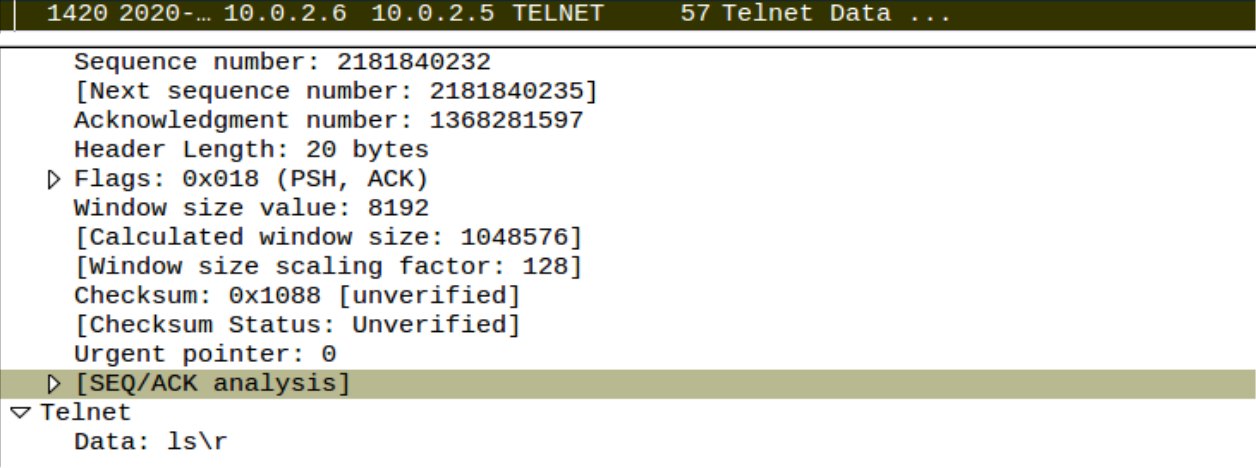
攻击者通过抓包，成功在受害者与观察者都不知道自己存在的情况下，获得了服务器当前目录下所有文件的名称，会话劫持攻击成功。

## **4B. 使用scapy进行攻击**

### Step1. 攻击者运行程序hijack.py来伪造数据包

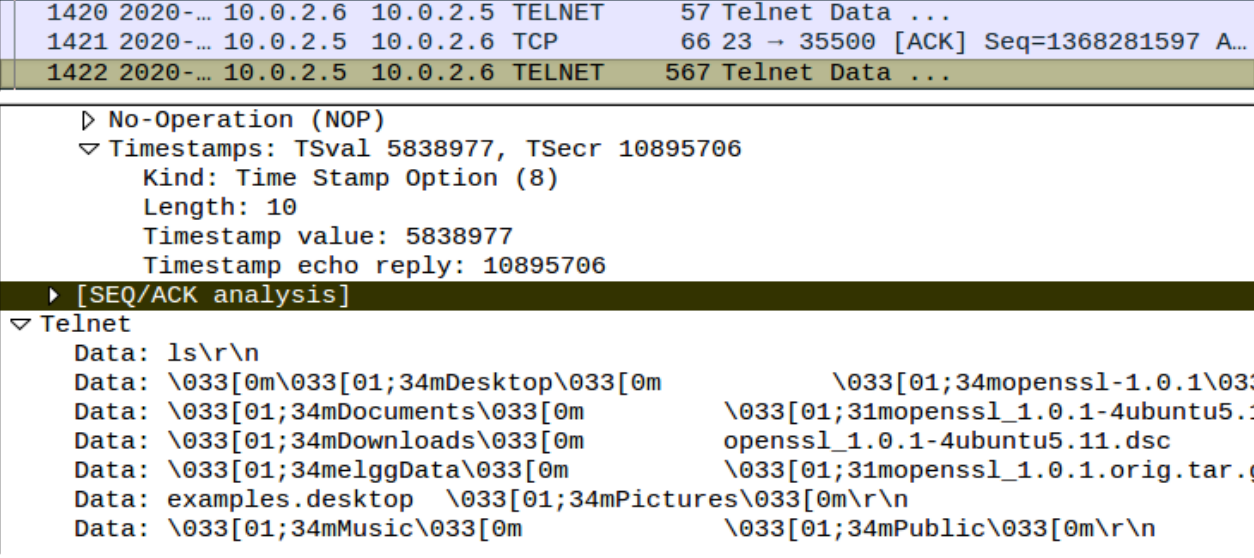
攻击者使用scapy伪造一个来自于观察者的报文，报文的数据字段为“ls\r”，将其发往受害者。





### Step2. 攻击者用wireshark抓包，获取受害者的响应报文数据内容

报文发出去后，攻击者在wireshark上抓到了受害者发给观察者（其实是假扮成观察者的攻击者）的响应报文，报文的数据部分是受害者执行指令”ls\r”的结果（输出目录下所有的文件）。



攻击者通过抓包，成功在受害者与观察者都不知道自己存在的情况下，获得了服务器当前目录下所有文件的名称，会话劫持攻击成功。