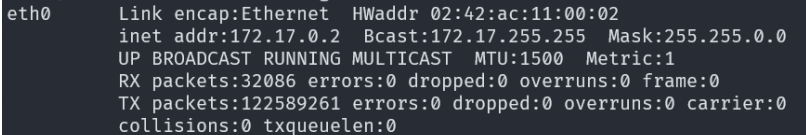
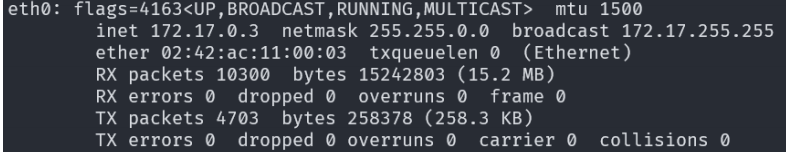
TCP/IP Attack Lab 57117219尚林灏

在实验中，设置三个虚拟机，ip地址如图

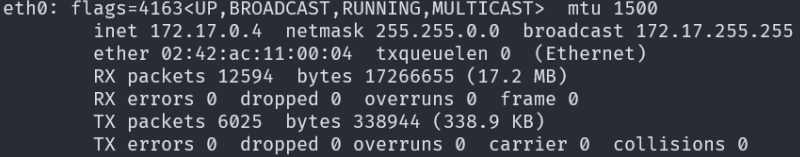
攻击者A



被攻击者B



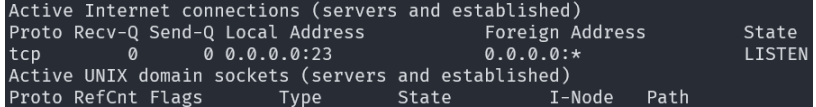
观察通信者C



Task1

A对B的端⼝发起洪泛攻击，C对B发起Telnet连接进⾏测试。在B内启动Telnet服务器，使⽤apt下载telnetd，然后使⽤ /etc/init.d/inetd 重启⽹络服务。

进⾏攻击之前，在B使⽤ netstat -na 查看当前的套接字队列：

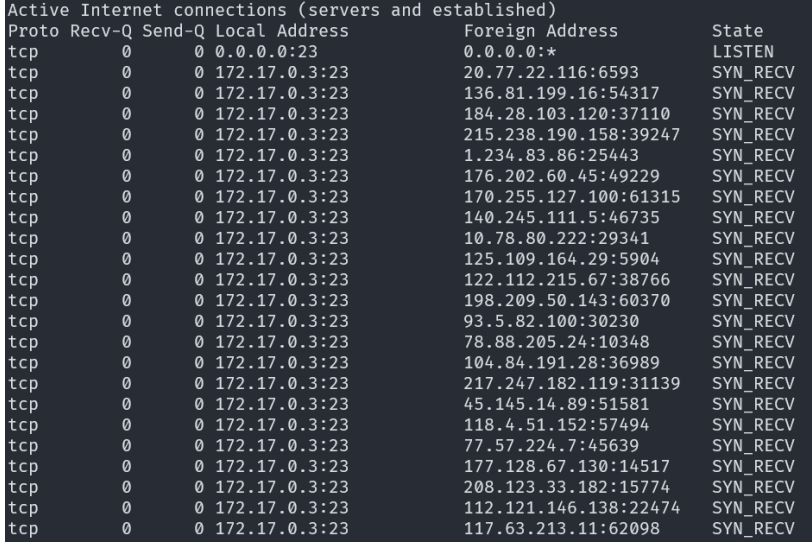


除了telnet在监听端⼝以外，没有任何套接字。此时通过C可以正常对B发起Telnet连接。

在B中关闭SYN Cookie的防御

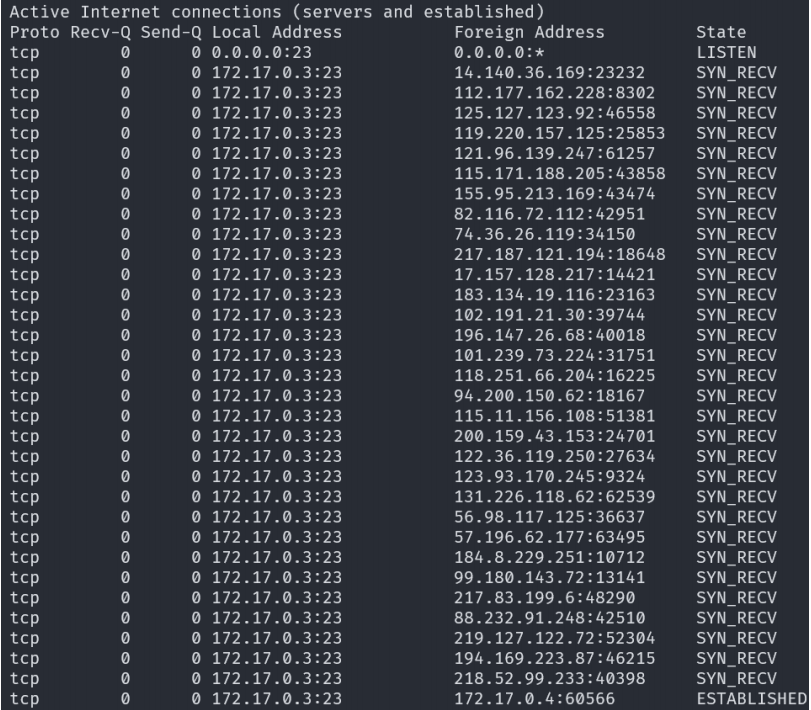
A中启动 netwox 发起SYN洪泛攻击

在B内使⽤ netstat -na 查看



发现许多状态为 SYN\_RECV，仅发出了第⼀次握⼿，没有后续TCP连接请求。此时，C再次向B发起Telnet连接请求，请求失败。

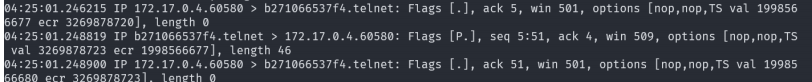
在B中重新开启SYN Cookie的防御。从A对B发起SYN洪泛攻击。接着，从C向B发起Telnet连接，连接成功。在B中再次⽤ netstat -na 查看套接字队列



有⼤量的 SYN\_RECV 状态的套接字，C发起的Telnet连接顺利建⽴。

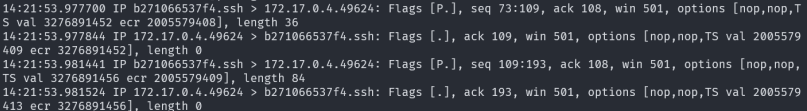
Task2

C与B建⽴ telnet 连接。通过 tcpdump 查看结果



B和C的 telnet 通信中，B的IP为 172.17.0.3 ，端⼝为23，C的IP为 172.17.0.4 ，端⼝为60580。最后⼀次通信，B的下⼀个seq值为51，C的下⼀个 seq值为4。

因此，构造脚本，在A中运⾏之后，C中的 telnet 连接中断。C与B建⽴ssh连接。通过 tcpdump 查看结果

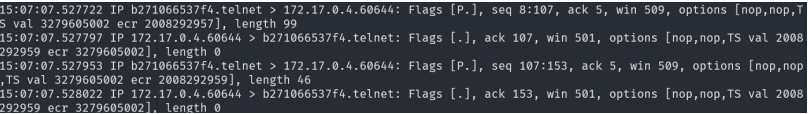


SSH是在TCP层之上的加密，可以得到正确的C的端⼝号为49624，最后⼀次通信后，B的下⼀个 seq 值为193，C的下⼀个 seq 值为108

构造脚本在A中运⾏之后，C中的 ssh 连接中断。

Task4

C与B建⽴ telnet 连接。然后通过 tcpdump 查看结果



C的端⼝为60644。最后⼀次通信后，B的下⼀个 seq 值为153，C的下⼀个 seq

值为5。因此，构造脚本为

from scapy.all import \*

ip = IP(src="172.17.0.4", dst="172.17.0.3")

tcp = TCP(sport=60644, dport=23, flags="PA", seq=5, ack=153)

payload = "touch evian"

pkt = ip/tcp/payload

ls(pkt)

send(pkt, verbose=0)

运⾏脚本后，在B的⽬录下成功查看到⽂件。