警示：实验报告如有雷同，雷同各方当次实验成绩均以 0 分计；在规定时间内未上交实验报告的，不得以其他方式补交，当次成绩按 0 分计；实验报告文件以 PDF 格式提交。

院系	计算机学院	班级	软工 3 班	学号	18342069	姓名	罗炜乐
完成日期： 2020 年 12 月 25 日							

网络扫描实验

【实验目的】

1. 掌握网络扫描技术的原理。
2. 学会使用 Nmap 扫描工具。

【实验环境】

实验主机操作系统：Windows IP地址：192.168.1.160
目标机操作系统：Raspberry Pi OS IP地址：192.168.1.105
网络环境：两台主机连接同一台路由器。

【实验工具】

Nmap (Network Mapper，网络映射器) 是一款开放源代码的网络探测和安全审核的工具。其设计目标是快速地扫描大型网络，也可以扫描单个主机。Nmap 以新颖的方式使用原始 IP 报文来发现网络上的主机及其提供的服务，包括其应用程序名称和版本，这些服务运行的操作系统包括版本信息，它们使用什么类型的报文过滤器/防火墙，以及一些其它功能。虽然 Nmap 通常用于安全审核，也可以利用来做一些日常管理维护的工作，比如查看整个网络的信息，管理服务升级计划，以及监视主机和服务的运行。

【实验过程】（要有实验截图）

在实验过程中，可通过 Wireshark 捕获数据包，分析 Nmap 采用什么探测包。

1．主机发现：进行连通性监测，判断目标主机。

首先确定测试机与目标机物理连接是连通的。

① 关闭目标机的防火墙，分别命令行窗口用 Windows 命令

ping 192.168.1.105

```
C:\Users\luowle>ping  192.168.1.105

正在 Ping 192.168.1.105 具有 32 字节的数据:
来自 192.168.1.105 的回复: 字节=32 时间=5ms TTL=64
来自 192.168.1.105 的回复: 字节=32 时间=3ms TTL=64
来自 192.168.1.105 的回复: 字节=32 时间=4ms TTL=64
来自 192.168.1.105 的回复: 字节=32 时间=2ms TTL=64

192.168.1.105 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 2ms，最长 = 5ms，平均 = 3ms
```

和 Nmap 命令

nmap -sP 192.168.1.105

```
C:\Users\luowle>nmap -sP  192.168.1.105
Starting Nmap 7.91 ( https://nmap.org ) at 2020-12-25 01:07 ?D1ú±ê×?ê±??
Nmap scan report for raspberrypi.lan (192.168.1.105)
Host is up (0.0040s latency).
MAC Address: DC:A6:32:89:5D:B5 (Raspberry Pi Trading)
Nmap done: 1 IP address (1 host up) scanned in 0.78 seconds
```

进行测试，记录测试情况。简要说明测试差别。

两者都说明可以连通。但 ping 使用的是 ICMP 包展示统计信息，而 nmap -sP 则直接扫描。

② 开启目标机的防火墙，重复①，结果有什么不同？请说明原因。

开启防火墙，设置防火墙只开放 22 端口。

```
       valid_lft 42997sec preferred_lft 37597sec
    inet6 2001:250:3002:4430:b4ba:678d:66ee:9cf/64 scope global dynamic mngtmpad
dr noprefixroute
       valid_lft 2591795sec preferred_lft 604595sec
    inet6 fe80::bca5:ce04:d8c:d491/64 scope link
       valid_lft forever preferred_lft forever
3: wlan0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qle
n 1000
    link/ether dc:a6:32:89:5d:b7 brd ff:ff:ff:ff:ff:ff
pi@raspberrypi:~ $  sudo ufw enable
Firewall is active and enabled on system startup
pi@raspberrypi:~ $ sudo ufw allow 22
Rule added
Rule added (v6)
pi@raspberrypi:~ $ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
22                         ALLOW       Anywhere
22 (v6)                    ALLOW       Anywhere (v6)

pi@raspberrypi:~ $
```

```
C:\Users\luowle>ping  192.168.1.105

正在 Ping 192.168.1.105 具有 32 字节的数据:
来自 192.168.1.105 的回复: 字节=32 时间=2ms TTL=64
来自 192.168.1.105 的回复: 字节=32 时间=2ms TTL=64
来自 192.168.1.105 的回复: 字节=32 时间=2ms TTL=64
来自 192.168.1.105 的回复: 字节=32 时间=5ms TTL=64

192.168.1.105 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 2ms，最长 = 5ms，平均 = 2ms
```

```
C:\Users\luowle>nmap -sP  192.168.1.105
Starting Nmap 7.91 ( https://nmap.org ) at 2020-12-25 01:55 ?D1ú±ê×?ê±??
Nmap scan report for raspberrypi.lan (192.168.1.105)
Host is up (0.0030s latency).
MAC Address: DC:A6:32:89:5D:B5 (Raspberry Pi Trading)
Nmap done: 1 IP address (1 host up) scanned in 0.96 seconds
```

ICMP 是网络层协议，通常来说，仅仅打开防火墙不能阻止 ping 通。修改路由规则丢弃 ICMP 包。

```
  GNU nano 2.5.3                File: /etc/ufw/before.rules                Modified


# allow all on loopback
-A ufw-before-input -i lo -j ACCEPT
-A ufw-before-output -o lo -j ACCEPT

# quickly process packets for which we already have a connection
-A ufw-before-input -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A ufw-before-output -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT
-A ufw-before-forward -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT

# drop INVALID packets (logs these in loglevel medium and higher)
-A ufw-before-input -m conntrack --ctstate INVALID -j ufw-logging-deny
-A ufw-before-input -m conntrack --ctstate INVALID -j DROP

# ok icmp codes for INPUT

-A ufw-before-input -p icmp --icmp-type echo-request -j DROP

-A ufw-before-input -p icmp --icmp-type destination-unreachable -j ACCEPT
-A ufw-before-input -p icmp --icmp-type source-quench -j ACCEPT
-A ufw-before-input -p icmp --icmp-type time-exceeded -j ACCEPT
-A ufw-before-input -p icmp --icmp-type parameter-problem -j ACCEPT
-A ufw-before-input -p icmp --icmp-type echo-request -j ACCEPT

# ok icmp code for FORWARD
-A ufw-before-forward -p icmp --icmp-type destination-unreachable -j ACCEPT
-A ufw-before-forward -p icmp --icmp-type source-quench -j ACCEPT
-A ufw-before-forward -p icmp --icmp-type time-exceeded -j ACCEPT
-A ufw-before-forward -p icmp --icmp-type parameter-problem -j ACCEPT
-A ufw-before-forward -p icmp --icmp-type echo-request -j ACCEPT


^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos   ^Y Prev Page
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line ^V Next Page
```

```
C:\Users\luowle>ping  192.168.1.105

正在 Ping 192.168.1.105 具有 32 字节的数据:
请求超时。
请求超时。
请求超时。
请求超时。

192.168.1.105 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 0，丢失 = 4 (100% 丢失)，
```

```
C:\Users\luowle>nmap -sP  192.168.1.105
Starting Nmap 7.91 ( https://nmap.org ) at 2020-12-25 02:10 ?D1ú±ê×?ê±??
Nmap scan report for raspberrypi.lan (192.168.1.105)
Host is up (0.0040s latency).
MAC Address: DC:A6:32:89:5D:B5 (Raspberry Pi Trading)
Nmap done: 1 IP address (1 host up) scanned in 0.98 seconds
```

③ 测试结果不连通，但实际上是物理连通的，什么原因？

防火墙丢弃了这些输入的 ICMP 包。

2．对目标主机进行 TCP 端口扫描

① 使用常规扫描方式

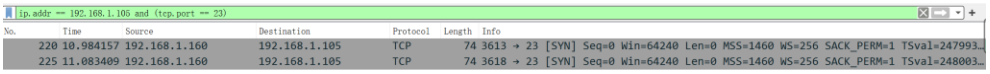
Nmap -sT 192.168.1.105

```
C:\Users\luowle>nmap -sT  192.168.1.105
Starting Nmap 7.91 ( https://nmap.org ) at 2020-12-25 01:39 ?D1ú±ê×?ê±??
Nmap scan report for raspberrypi.lan (192.168.1.105)
Host is up (0.0044s latency).
Not shown: 999 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh
MAC Address: DC:A6:32:89:5D:B5 (Raspberry Pi Trading)

Nmap done: 1 IP address (1 host up) scanned in 42.55 seconds
```

ip.addr == 192.168.1.105 and (tcp.port == 22)

No.	Time	Source	Destination	Protocol	Length	Info
230	11.182598	192.168.1.160	192.168.1.105	TCP	74	3622 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 TSval=2…
232	11.186275	192.168.1.105	192.168.1.160	TCP	74	22 → 3622 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSv…
233	11.186340	192.168.1.160	192.168.1.105	TCP	66	3622 → 22 [ACK] Seq=1 Ack=1 Win=131584 Len=0 TSval=2480141 TSecr=174378115
235	11.186536	192.168.1.160	192.168.1.105	TCP	54	3622 → 22 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
450	12.498718	192.168.1.160	192.168.1.105	TCP	74	3692 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 TSval=2…
453	12.502268	192.168.1.105	192.168.1.160	TCP	74	22 → 3692 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSv…
454	12.502351	192.168.1.160	192.168.1.105	TCP	66	3692 → 22 [ACK] Seq=1 Ack=1 Win=131584 Len=0 TSval=2481457 TSecr=174379431
456	12.502932	192.168.1.160	192.168.1.105	TCP	54	3692 → 22 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
522	13.813158	192.168.1.160	192.168.1.105	TCP	74	3758 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 TSval=2…
524	13.816528	192.168.1.105	192.168.1.160	TCP	74	22 → 3758 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSv…
525	13.816613	192.168.1.160	192.168.1.105	TCP	66	3758 → 22 [ACK] Seq=1 Ack=1 Win=131584 Len=0 TSval=2482771 TSecr=174380746
527	13.816886	192.168.1.160	192.168.1.105	TCP	54	3758 → 22 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
591	15.128483	192.168.1.160	192.168.1.105	TCP	74	3824 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 TSval=2…
593	15.130879	192.168.1.105	192.168.1.160	TCP	74	22 → 3824 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM=1 TSv…
594	15.130948	192.168.1.160	192.168.1.105	TCP	66	3824 → 22 [ACK] Seq=1 Ack=1 Win=131584 Len=0 TSval=2484086 TSecr=174382060
596	15.132480	192.168.1.160	192.168.1.105	TCP	54	3824 → 22 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
674	16.442194	192.168.1.160	192.168.1.105	TCP	74	3890 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1 TSval=2…



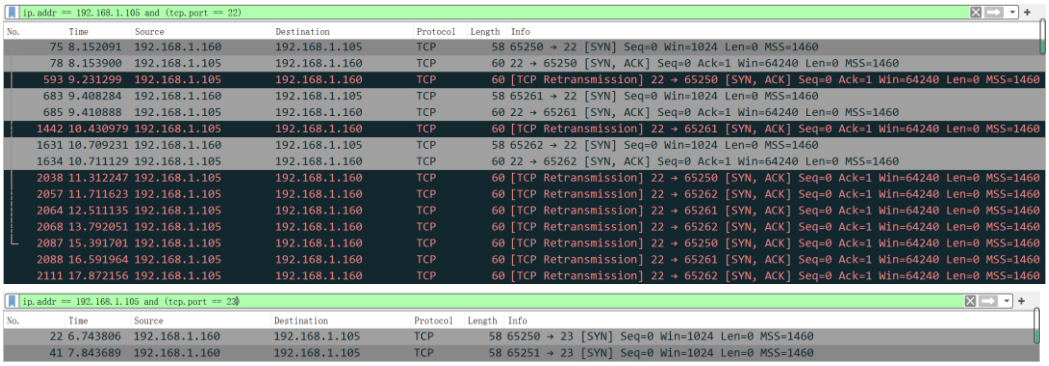
ssh 服务的 22 端口开放了

② 使用 SYN 半扫描方式

Nmap -sS 192.168.1.105

```
C:\Users\luowle>nmap -sS 192.168.1.105
Starting Nmap 7.91 ( https://nmap.org ) at 2020-12-25 01:48 ?D1ú±ê×?ê±??
Nmap scan report for raspberrypi.lan (192.168.1.105)
Host is up (0.0040s latency).
Not shown: 999 filtered ports
PORT   STATE SERVICE
22/tcp open  ssh
MAC Address: DC:A6:32:89:5D:B5 (Raspberry Pi Trading)

Nmap done: 1 IP address (1 host up) scanned in 5.70 seconds
```



ssh 服务的 22 端口开放了。

③ 比较上述两次扫描结果差异、扫描所花费的时间。并进行解释。

扫描的结果是一样的，都和我预期中的一样。但是使用常规扫描方式的时间比使用 SYN 半扫描长的多。通过抓包可以看出来，常规扫描方式需要建立整个 TCP 连接，而 SYN 半扫描接收到了 ACK 则认为该端口开放。因此，两个方式所耗费的时间有显著的差异。

【实验体会】

1. 在本次实验中，我学会了 Linux 的 ufw 防火墙和 nmap 的基本用法，同时也回顾了使用 WireShark 抓包。使用 nmap，可以获得目标主机的许多信息，比如说操作系统类型，端口开启情况。我们可以通过这些信息判断网络安全情况。

2. 在不安全的网络中（比如说公网），开启防火墙是很有必要的，否则某些端口可能成为安全漏洞。一般情况下，把防火墙开启。

3. 在开启防火墙后，我还是能 ping 通目标机，原因是防火墙默认没有将网络层的 ICMP 包屏蔽，查阅资料后让我加深了对计算机网络的认识。

4. 辨析了常规扫描和 SYN 半扫描的不同之处，让我知道了判断端口是否开放只需要一个 SYN 包探测就够了，而不一定需要建立整个连接。通常来说，SYN 半扫描的速度比常规扫描快得多。