

《信息安全综合实践》实验报告

实验名称： 信息收集实验

姓名： 黎锦灏 学号： 518021910771 邮箱： ljh2000@sjtu.edu.cn 实验时长： 75 分钟

一、实验目的

1. 熟悉 TCP/IP 协议、Ping 命令基本概念
2. 学习 nmap、SuperScan 扫描方式及其原理
3. 掌握在 Windows 和 Linux 环境下相关网络命令的使用
4. 了解 traceroute 追踪路由信息的原理及用法
5. 了解常见的信息收集位置和方法

二、实验环境

操作系统：

操作机：Windows_7 (administrator/123456)

目标机：Linux_Kali/Linux_Ubuntu(root/123456)

相关软件： Nmap、SuperScan、GitHack

三、实验内容

序	内容	实验步骤(需截图)
1)	主机存活性探测实验	Ping 命令
2)		使用 Nmap 进行多种方式的探测
3)		使用 SuperScan 扫描网段内的存活主机
4)		使用 Fping 实现网段内的存活主机探测
5)		使用 TRACERT 追踪路由信息
6)	手工信息收集实验 1	Robots.txt 信息收集
7)		网站备份压缩文件
8)		Git 导致文件泄露
9)	手工信息收集实验 2 (选做，无加分)	phpinfo 信息泄露
10)		CMS 指纹识别
11)		配置文件泄露

四、实验过程截图 (30 分)

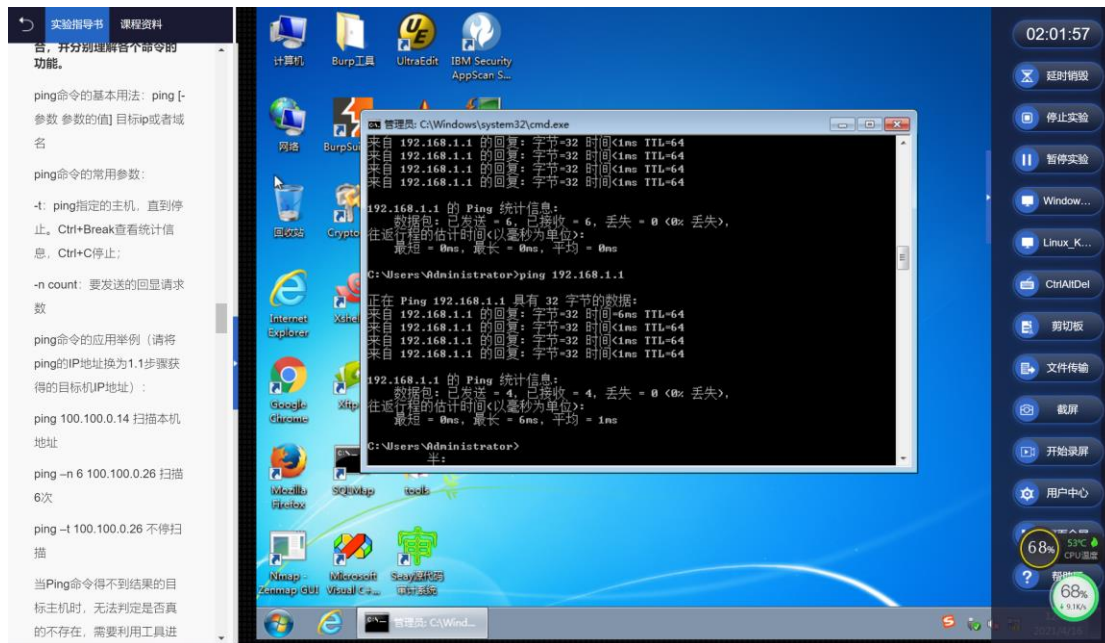
注：将实验步骤中每个步骤（例：步骤 2：使用 Nmap 进行多种方式的探测）你认为的关键信息截图保存，每个步骤不超过 3 张图片，并用简短的话描述实验所得的结果。

1. Ping、Nmap、SuperScan、Fping、Traceroute 工具的信息收集结果。

实验一：主机存活性探测实验

步骤 1：Ping 命令（Windows 环境下）

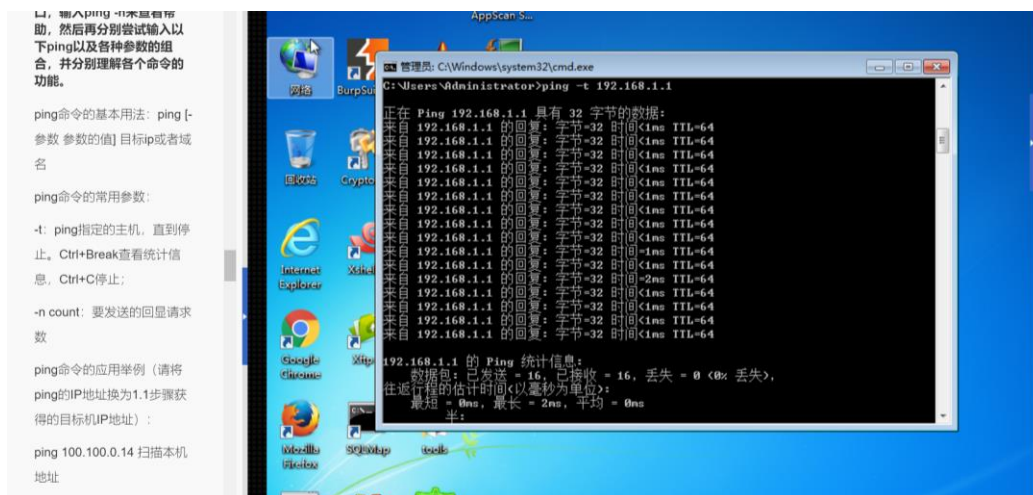
打开目标机，查看 ip 地址为 192.168.1.1。然后测试 Ping 命令的使用，输入 Ping 命令：`ping 192.168.1.1`，扫描目标机地址，可以看到目标 IP 地址发生了四个回复包，如下图所示：



下面测试-n 参数，输入 Ping 命令：`ping -n 6 192.168.1.1`，功能是扫描目标机地址 6 次，收到来自目标地址的 6 个回复包，说明可以达到目标地址，如下图所示：



接着测试-t 参数，输入 Ping 命令：`ping -t 192.168.1.1`，功能是不停扫描目标机地址，直到键盘键入 Ctrl+C 停止，如下图所示：



步骤 2：使用 Nmap 进行多种方式的探测

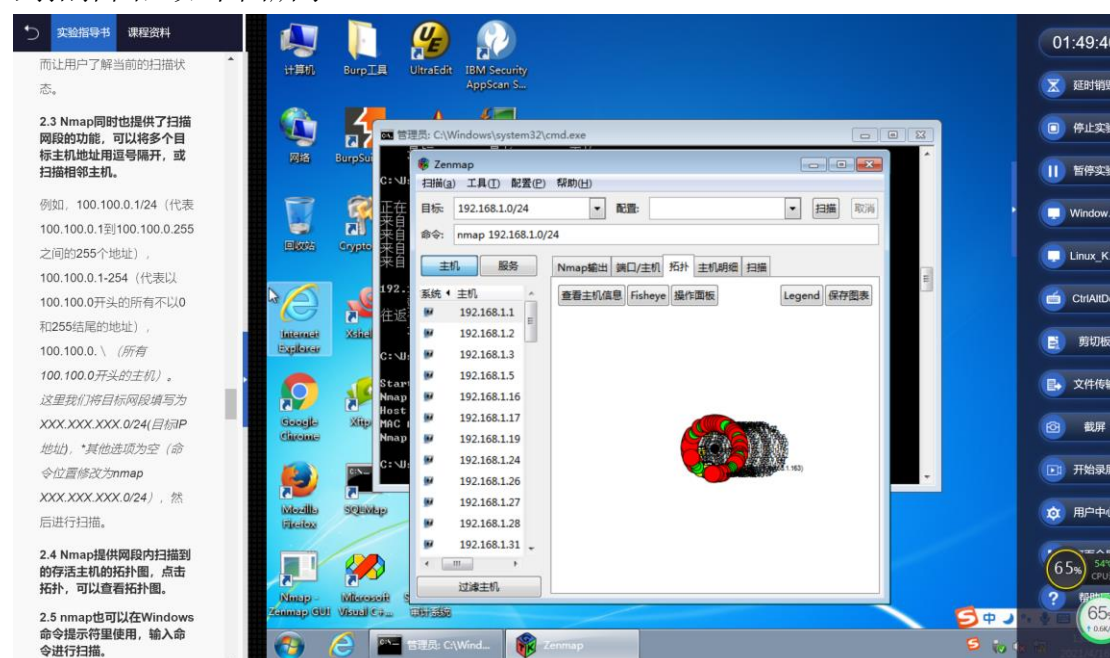
在桌面上找到 Nmap 的快捷方式，双击运行 Nmap 软件，在目标栏输入目标机 IP 地址：192.168.1.1，使用默认配置：nmap -T4 -A -v 192.168.1.1，对目标主机进行扫描，如下图所示：



Nmap 同时也提供扫描网段的功能，输入命令：**nmap -T4 -A -v 192.168.1/24**，即 192.168.1.1 到 192.168.1.255 之间的这 255 个地址，然后进行扫描，如下图所示：



Nmap 提供网段内扫描到的存活主机的拓扑图，点击界面的拓扑键，可以看到拓扑图，如下图所示：



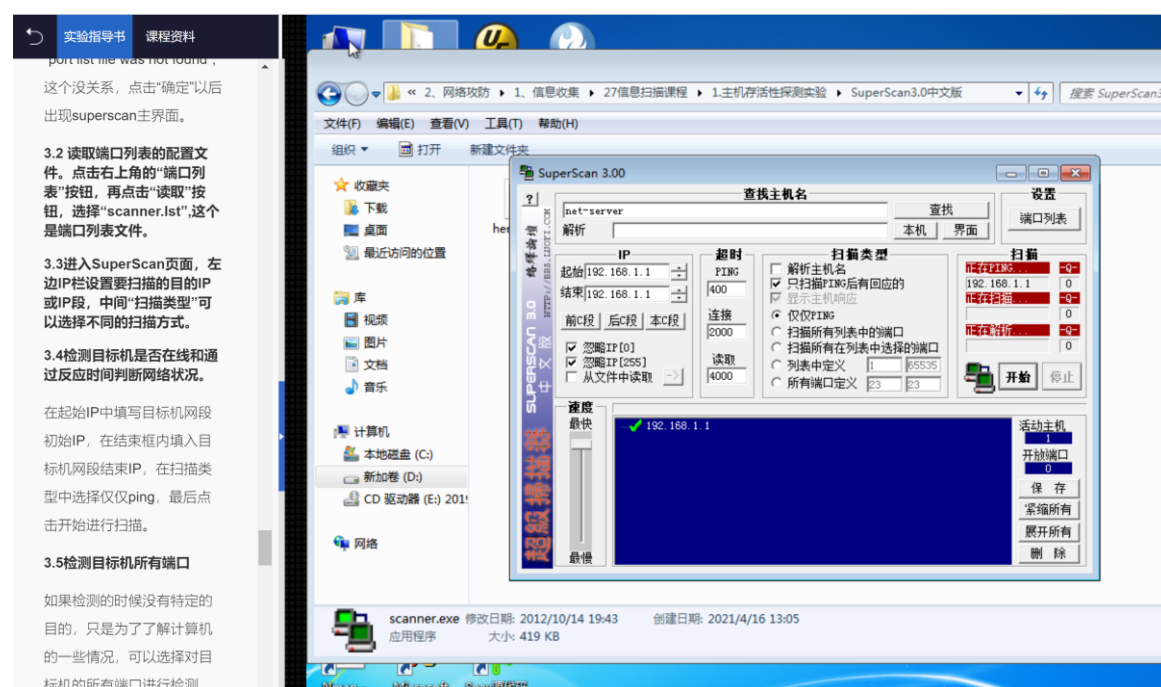
Nmap 同样可以通过 Windows 命令提示符使用，输入命令：**nmap -sn 192.168.1.1** 可以进行扫描。

步骤 3: 使用 Superscan 扫描网段内的存活主机

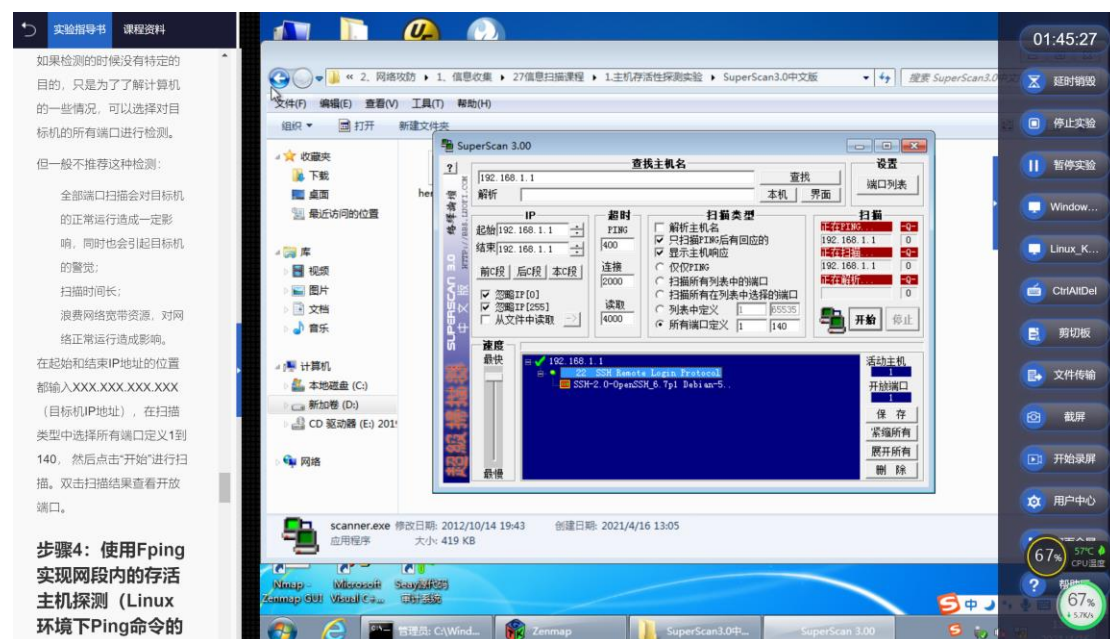
双击 scanner.exe 后运行 Superscan，读取端口列表的配置文件，点击右上角的“端口列表”按钮，选择“scanner.lst”端口列表文件，然后在左侧 IP 栏设置扫描的 IP（这里，输入目标机地址 192.168.1.100），如下图所示：



为检测目标机是否在线和通过反应时间判断网络状况，可以在扫描类型中选择“仅仅 PING”，点击开始扫描，如下图所示：



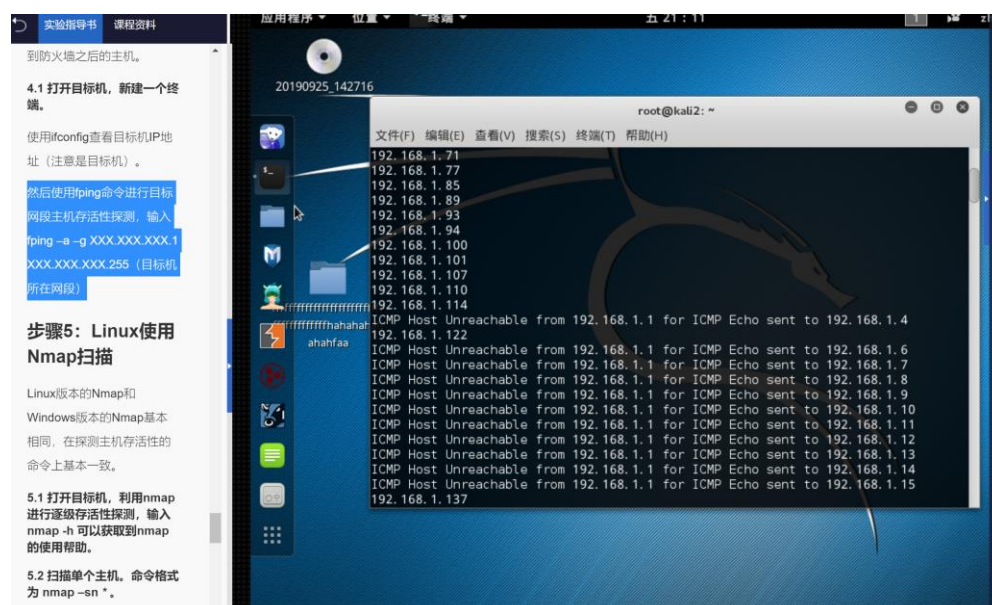
在扫描类型中选择所有端口定义：1 到 140，然后点击开始扫描，双击扫描结果可以查看开放端口，如下图所示：

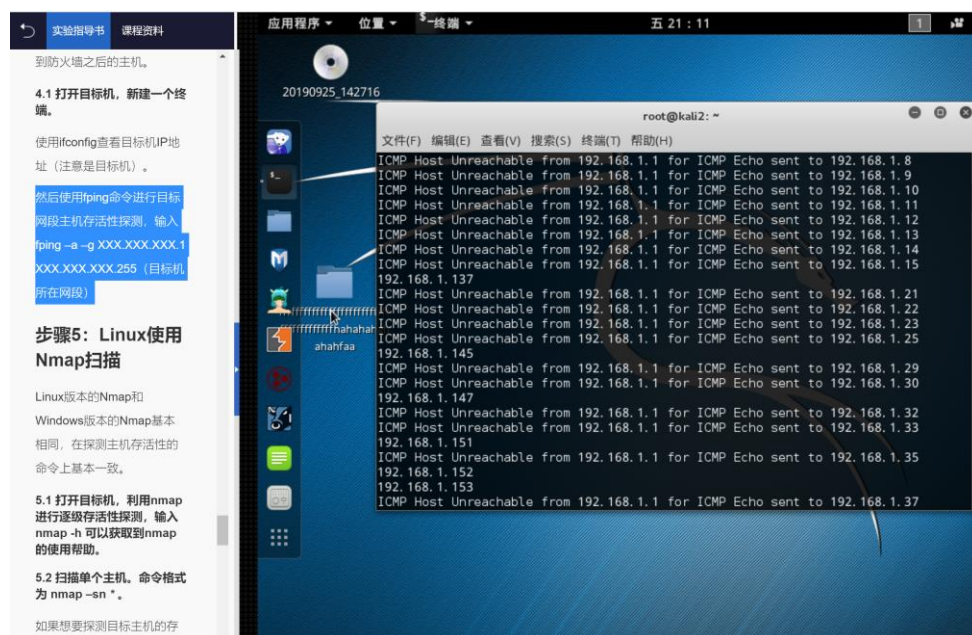


步骤 4：使用 Fping 实现网段内的存活主机探测

（Linux 环境下 Ping 命令的应用）

首先使用 `ifconfig` 命令得到 Linux 目标机 IP 地址，然后可以输入 `fping -a -g 192.168.1.1 192.168.1.255`，表示扫描 192.168.1.1 到 192.168.1.255 这个网段，进行目标网段主机存活性检测，如下图所示：

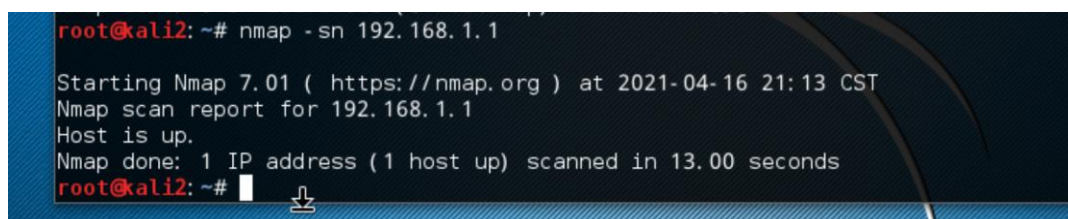




从反馈信息中可以看出，网段中哪些 IP 地址可以到达，哪些 IP 不可达，由此可以做出主机存活判断。

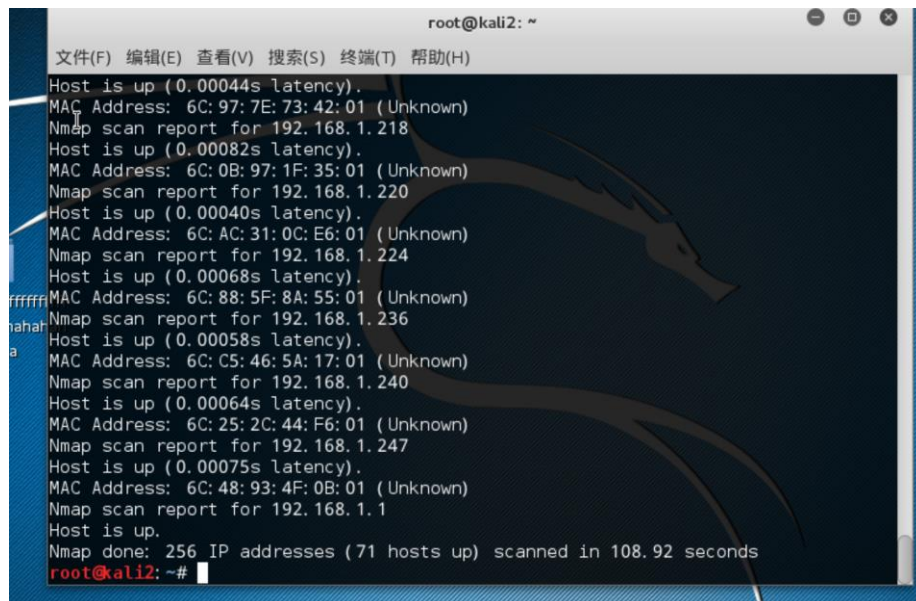
步骤 5: Linux 使用 Nmap 扫描

扫描单个主机的指令为：`nmap -sn 192.168.1.1`，如下图所示：



扫描网段的存活主机，输入指令：`nmap -sn 192.168.1.0/24`，如下图所示：



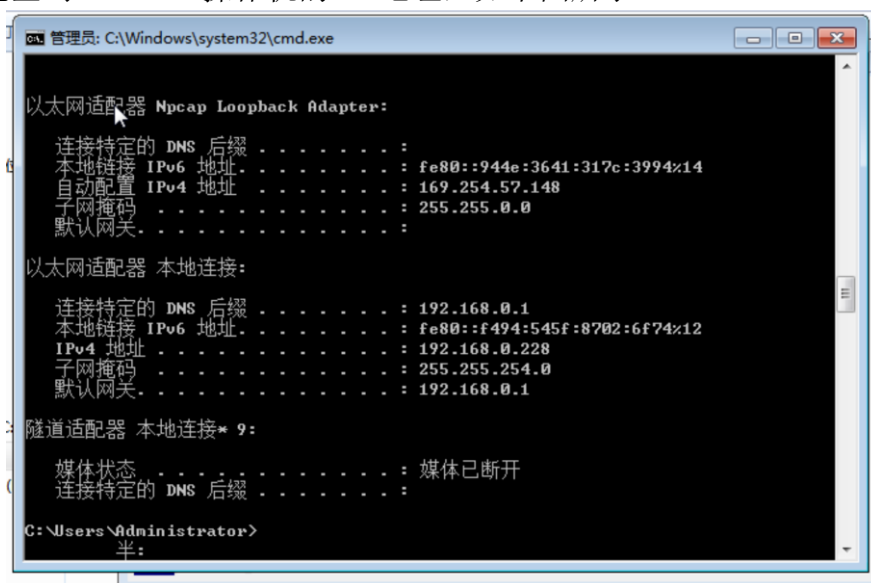


```
root@kali2: ~  
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)  
Host is up (0.00044s latency).  
MAC Address: 6C:97:7E:73:42:01 (Unknown)  
Nmap scan report for 192.168.1.218  
Host is up (0.00082s latency).  
MAC Address: 6C:0B:97:1F:35:01 (Unknown)  
Nmap scan report for 192.168.1.220  
Host is up (0.00040s latency).  
MAC Address: 6C:AC:31:0C:E6:01 (Unknown)  
Nmap scan report for 192.168.1.224  
Host is up (0.00068s latency).  
MAC Address: 6C:88:5F:8A:55:01 (Unknown)  
Nmap scan report for 192.168.1.236  
Host is up (0.00058s latency).  
MAC Address: 6C:C5:46:5A:17:01 (Unknown)  
Nmap scan report for 192.168.1.240  
Host is up (0.00064s latency).  
MAC Address: 6C:25:2C:44:F6:01 (Unknown)  
Nmap scan report for 192.168.1.247  
Host is up (0.00075s latency).  
MAC Address: 6C:48:93:4F:0B:01 (Unknown)  
Nmap scan report for 192.168.1.1  
Host is up.  
Nmap done: 256 IP addresses (71 hosts up) scanned in 108.92 seconds  
root@kali2: ~#
```

从扫描结果可以看出，当前网段下有 71 个主机存活。

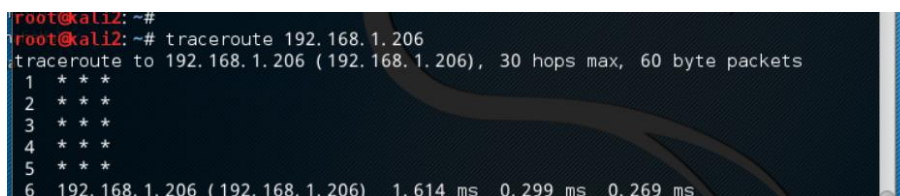
步骤 6：使用 TRACERT 追踪路由信息

首先查询 Windows 操作机的 IP 地址，如下图所示：



```
管理员: C:\Windows\system32\cmd.exe  
以太网适配器 Npcap Loopback Adapter:  
连接特定的 DNS 后缀 . . . . . :  
本地连接 IPv6 地址 . . . . . : fe80::944e:3641:317c:3994%14  
自动配置 IPv4 地址 . . . . . : 169.254.57.148  
子网掩码 . . . . . : 255.255.0.0  
默认网关 . . . . . :  
以太网适配器 本地连接:  
连接特定的 DNS 后缀 . . . . . : 192.168.0.1  
本地连接 IPv6 地址 . . . . . : fe80::f494:545f:8782:6f74%12  
IPv4 地址 . . . . . : 192.168.0.228  
子网掩码 . . . . . : 255.255.254.0  
默认网关 . . . . . : 192.168.0.1  
隧道适配器 本地连接* 9:  
媒体状态 . . . . . : 媒体已断开  
连接特定的 DNS 后缀 . . . . . :  
C:\Users\Administrator>  
半:
```

输入 traceroute 操作机 IP 地址，可以查看到达目的地的路由，如下图所示：



```
root@kali2: ~#  
root@kali2: ~# traceroute 192.168.1.206  
traceroute to 192.168.1.206 (192.168.1.206), 30 hops max, 60 byte packets  
1 * * *  
2 * * *  
3 * * *  
4 * * *  
5 * * *  
6 192.168.1.206 (192.168.1.206) 1.614 ms 0.299 ms 0.269 ms
```


2. 手工信息收集 1 结果。

实验二：手动信息收集实验 1

步骤 1: robots.txt 信息收集

首先获取目标机的 IP 地址为 192.168.0.92，如下图所示

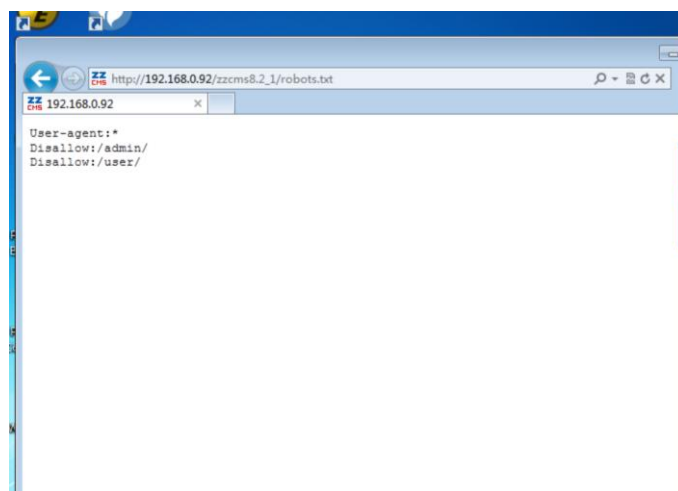
```
CentOS Linux 7 (Core)
Kernel 3.10.0-514.el7.x86_64 on an x86_64

localhost login: root
Password:
Last login: Tue Apr 13 05:04:39 on tty1
[root@localhost ~]# ifconfig
ens3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1450
    inet 192.168.0.92 netmask 255.255.254.0 broadcast 192.168.1.255
    inet6 fe80::f478:a4cc:d0ef:346d prefixlen 64 scopeid 0x20<link>
    ether 6c:e3:1f:97:af:01 txqueuelen 1000 (Ethernet)
    RX packets 46372 bytes 2872189 (2.7 MiB)
    RX errors 12692 dropped 0 overruns 0 frame 12692
    TX packets 1861 bytes 65057 (63.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

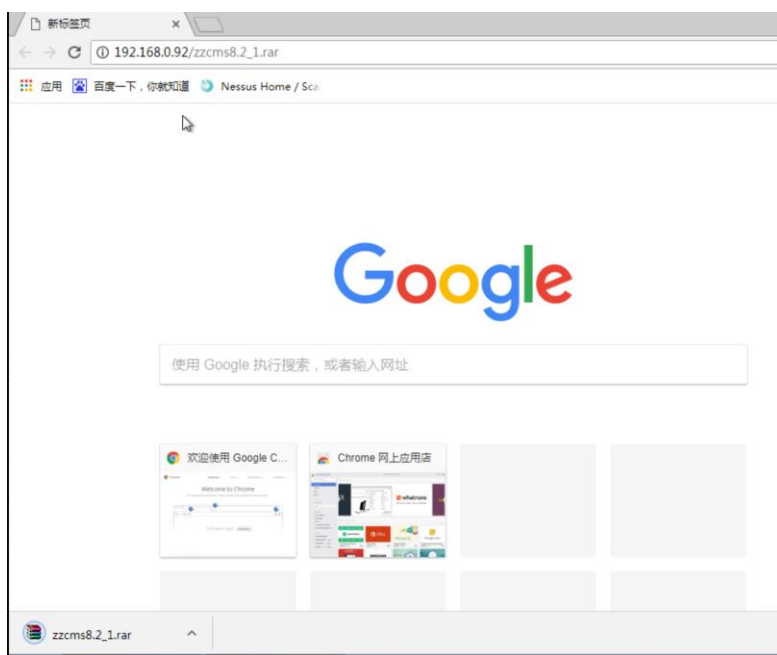
[root@localhost ~]# _
```

在本步骤中,选用的是目标机 web 服务器上 zzcms 网站的 robots.txt 文件,即访问 http://192.168.0.92/zzcms8.2_1/robots.txt, 该文件用于告诉搜索引擎,哪些页面可以去抓取,哪些页面不要抓取。访问结果如图所示,该结果说明, zzcms 限制搜索引擎访问其后台的/admin/目录和/user/目录。

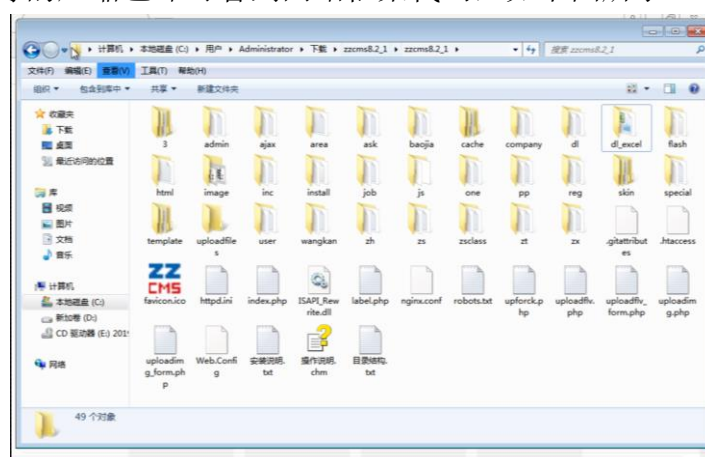


步骤 2: 网站备份压缩文件

访问 http://192.168.0.92/zzcms8.2_1.rar, 访问目标机 web 服务器上的 zzcms 网站的备份压缩包。

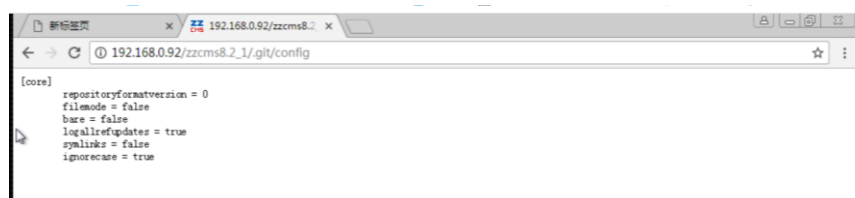


解压下载好的压缩包即可看到网站框架代码，如下图所示。



步骤 3: Git 导致文件泄露

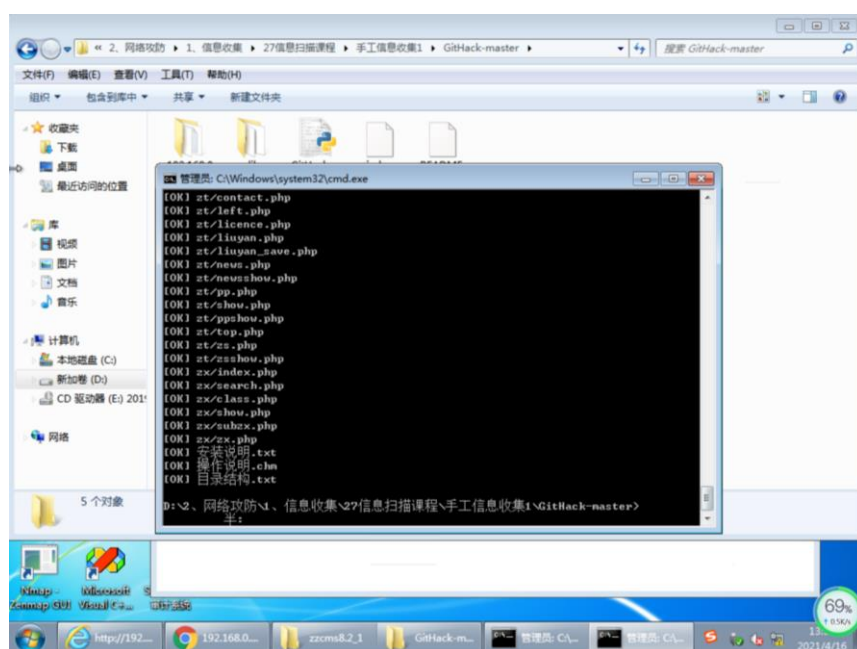
访问 http://192.168.0.92/zzcms8.2_1/.git/config, 发现未删除该文件, 由此可进行重塑网站源码, 如下图所示。



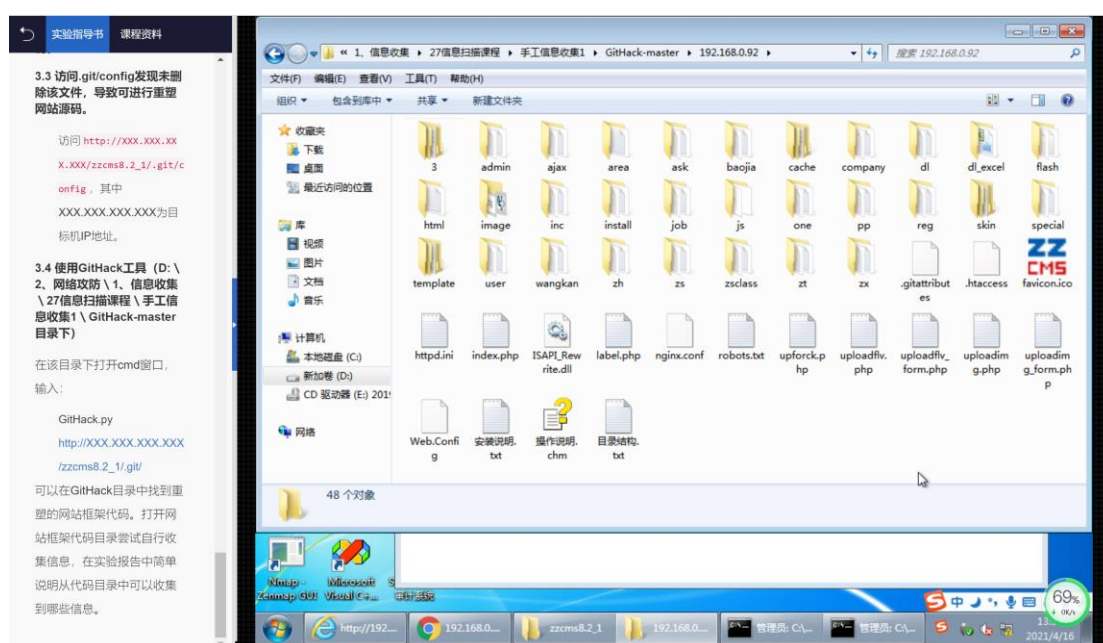
GitHack 工具可用于从 .git 文件恢复源代码。使用 GitHack 工具, 进入 GitHack-master 目录, 然后输入:

GitHack.py http://192.168.0.92/zzcms8.2_1/.git/

如下图所示：



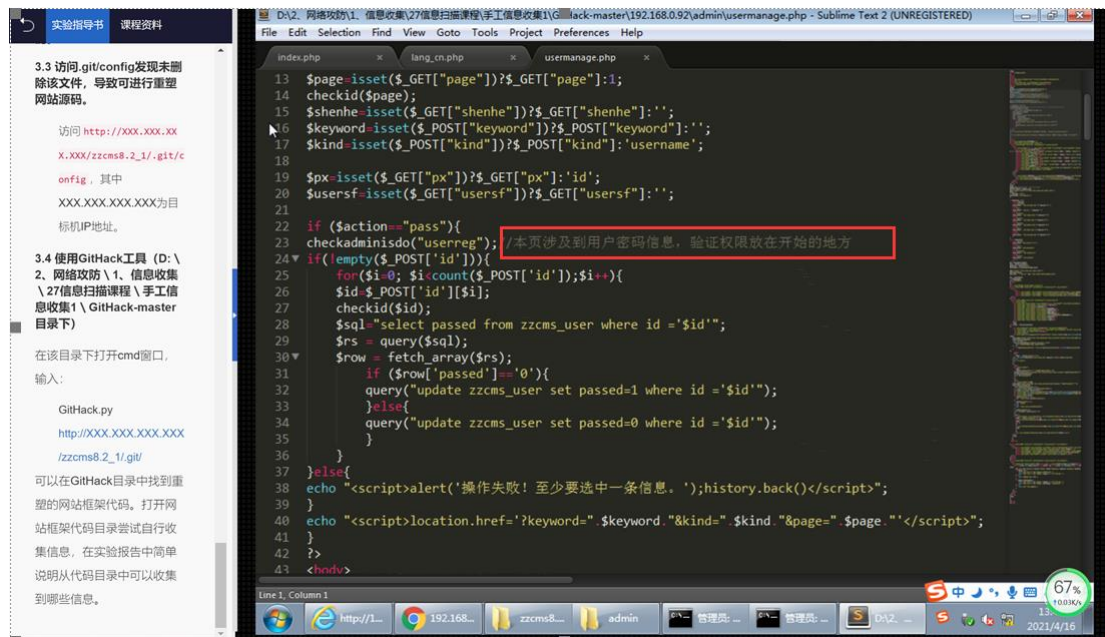
可以在 GitHack 目录中找到重塑的网站框架代码，如下图所示：



我仔细检查了代码目录，并阅读了目录下的一些文件，发现可以收集到以下信息：

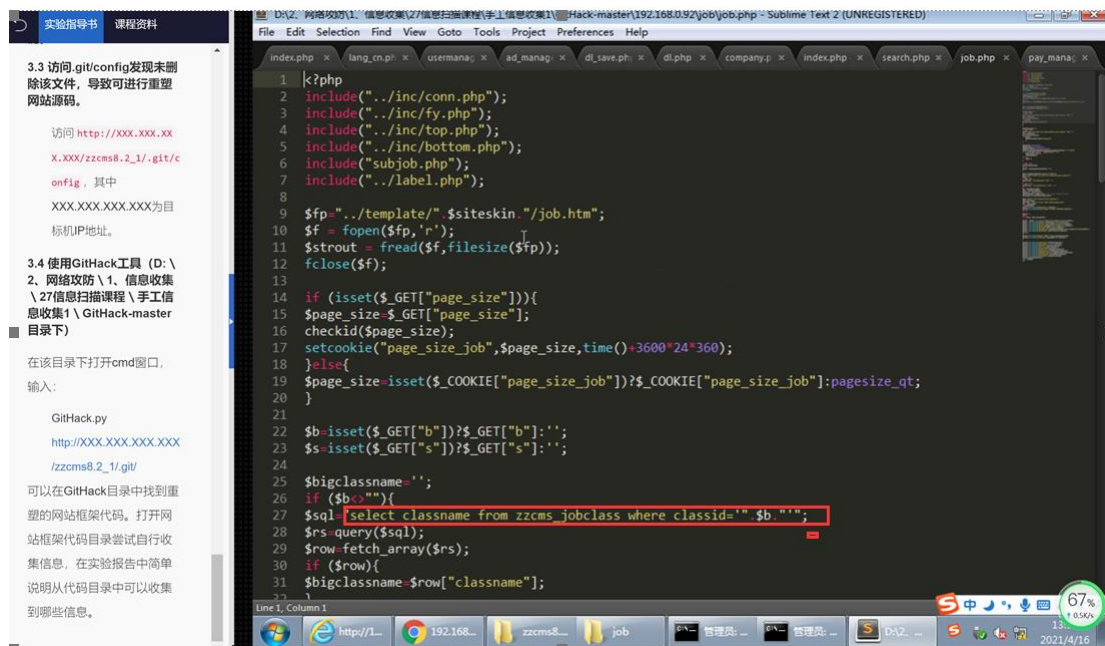
1、192.168.0.92/admin/usermanage.php

在下图所示的 php 脚本中，函数注释里写到“本页涉及到用户密码信息，验证权限放在开始的地方”，可以推测该函数存在用户密码信息和权限问题，这或许为下一步攻击提供了入口，例如可对此进行越权提取和 SQL 注入攻击。



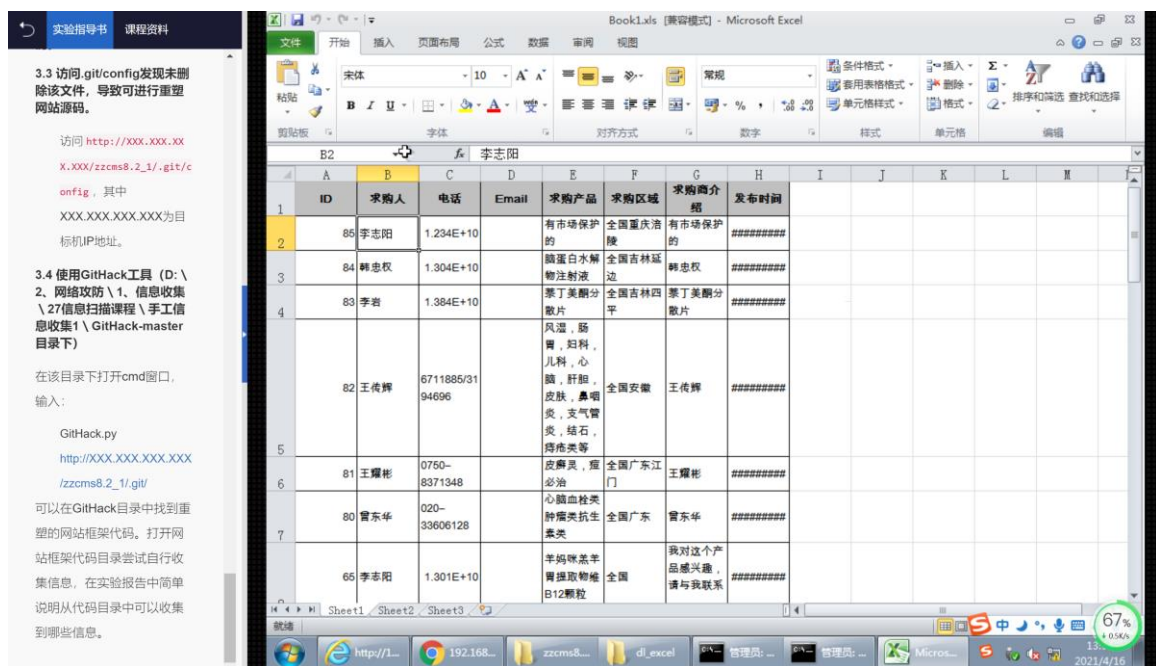
2、192.168.0.92/admin/job/job.php

在下图所示的 php 脚本中，反映了关于 job 数据库的相关信息，并且提供了标准 sql 查询语句，对于攻击者来说，可以作为 SQL 注入点。



3、192.168.0.92/dl_excel/Book1.xls

在 dl_excel 文件夹下可以看到 Book1 这个 excel 文件，打开以后能看到一个求购清单，记录了 ID、求购人、电话等隐私信息，这不仅暴露了内部求购数据的表格形式，还造成了用户的敏感信息泄露。



上面列举了收集到的一些信息，通过.git 恢复出来的源代码中还有不少类似信息泄露问题。从中我们可以发现，.git 文件的管理不慎可能导致**网站源码信息暴露**，同时可能暴露网站服务器的**网络节点信息**，从而为攻击者攻击平行的其它网站、进行渗透提供便捷。

五、分析和思考（60 分）

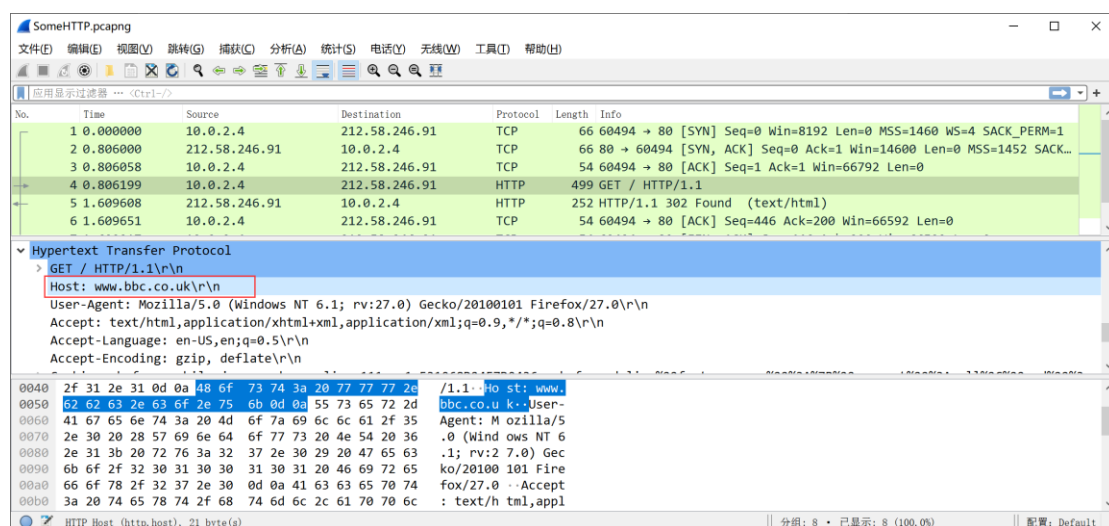
1. 使用 nmap 如何对某 IP（如 10.16.0.6）的 80、21、53 端口进行 TCP SYN 扫描？请给出对应的指令。

```
nmap -sS -p80,21,53 10.16.0.6
```

-sS 表示 TCP SYN 扫描（即半开放扫描），-p 表示相应端口。

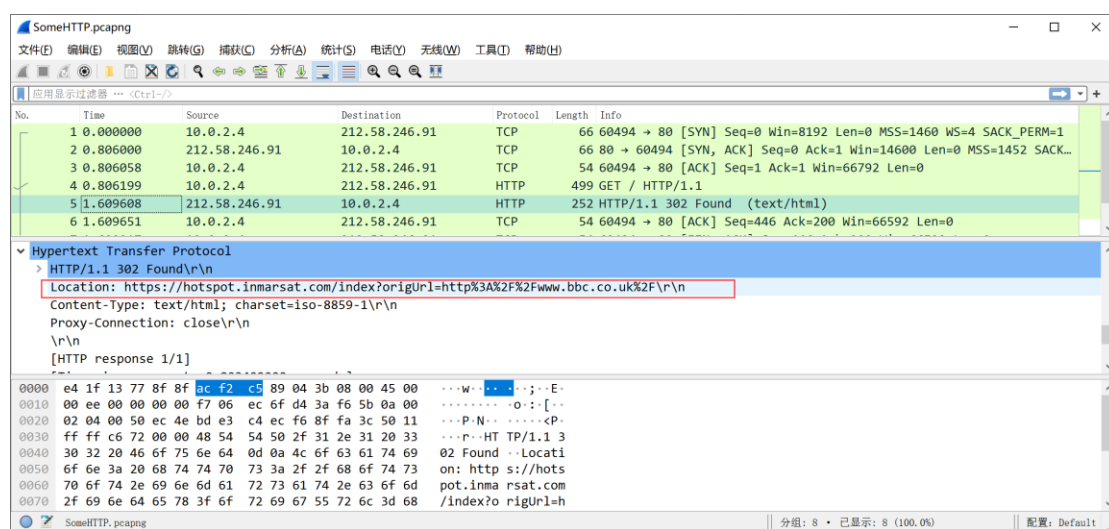
2. 通过 Wireshark 查看抓包文件 SomeHTTP.pcapng（oc 下载），分析用户访问了哪个域名？这个网站被重定向到了哪个域名（请给出 URL 解码后的答案）？

打开抓包文件后，可以看到各数据包的详细信息，从第四个数据包，即 GET 请求中可以看出，用户访问的域名为 www.bbc.co.uk/，如下图所示：



分析第五个数据包，由 Location 表项可以读出网站实际上被重定向到

<https://hotspot.inmarsat.com/index?origUrl=http://www.bbc.co.uk/>



3. 手工信息收集实验 1: 应该怎样既使用`robots.txt`的屏蔽搜索引擎访问的功能，又不泄露后台地址和隐私目录？（提示：可效仿 robots.txt 中 User-agent 的星号通配符思想）

1、在 robots.txt 中引进星号通配符，使用 “User-agent:*” 可以屏蔽搜索引擎访问某些页面，但是 robots.txt 文件本身能被任何人访问到，里面存储的后台地址和隐私目录信息就有可能泄露。

2、仿照在 User-agent 中使用的星号通配符思想，对于隐私目录的名字，我们同样可以用星号通配符来隐藏其具体信息，例如屏蔽搜索引擎访问 admin 目录且不想暴露 admin 目录名，可以采用如下方式：

User-agent:*

Disallow: /a*/

通配符部分能隐藏目录名信息，在不造成歧义的情况下起到了保护作用。

3、但是攻击者仍可以通过常用目录名来猜测隐私目录全名，因此我们可以将 admin 目录重命名为更难以猜测且具有特点个性的新目录名，例如：ljh_admin。这样可以让攻击者难以猜测隐私目录名，保护了真实的网站结构。

4. 手工信息收集实验：除实验内容外，还有哪些网站信息收集的方法？请介绍两到三种你所了解的方法并简述原理。

网站信息收集方法：

(1)、扫描网站目录、收集敏感目录文件：

扫描网站目录的常用工具有：**wwwscan**、**DirBuster**、**Sensitivefilescan**、**Spinder.py**，以 **DirBuster** 为例说明具体工作方式。

DirBuster 是一款基于 Java 编写的、专门用于探测 Web 服务器的目录和隐藏文件，而且工具界面是纯图形化的，便于使用。例如需要收集信息的目标是 <http://www.xxx.com/admin/>，为扫描指定网址，可以将请求方法设置为 “Auto Switch(HEAD and GET)”，然后设置合适的线程数值（20 到 30），选择扫描类型为 “List based brute force”，即个人字典扫描，在工具里可以选择自带的字典或自定义字典。

扫描时只需要在 URL to fuzz 中填写 “admin/{dir}”，在 {dir} 的前后可

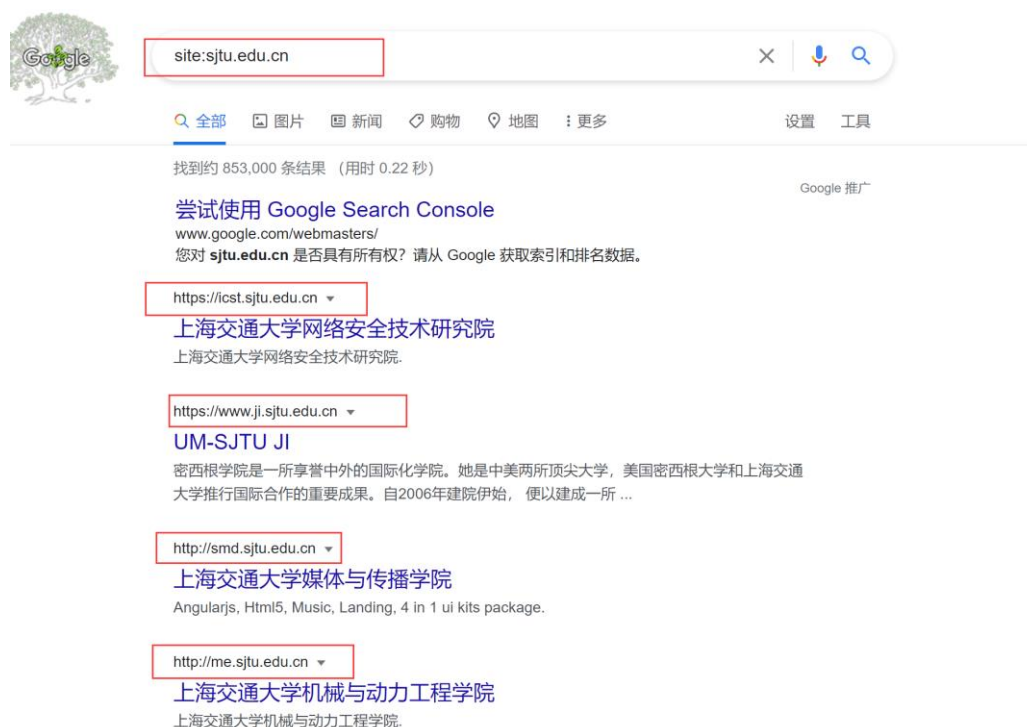
以任意拼接需要的目录或后缀，表示扫描 admin 目录下的所有 php 文件。

(2)、Google 等搜索引擎：

我们可以构造特殊的关键字语法，利用 Google 来搜索互联网上的相关敏感信息。Google 的常用语法及说明如下表所示：

关键字	说明
Site	指定域名
Inurl	URL 中存在关键字的网页
Intext	网页正文中的关键字
Filetype	指定文件类型
Intitle	网页标题中的关键字
Link	Link:baidu.com 即表示返回所有和 baidu.com 做了链接的 URL
Info	查找指定站点的一些基本信息
Cache	搜索 Google 里关于某些内容的缓存

那么我们可以根据上述语法来构造表达式收集需要的网站信息，例如我们需要搜索网页正文中含有“信息安全综合实践”并且域名后缀是 edu.cn 的网站，则我们可以在 Google 中输入：**site:sjtu.edu.cn intext:信息安全综合实践**，所得的结果即关于这门课程的一些学校网站的后台。



这样我们在 Google 搜索引擎中得到一些网站后台后,还可以相应进行收集源代码泄露、数据库文件、配置信息、未授权访问、robots.txt 等敏感信息。

(3)、绕过 CDN 查找网站真实 IP:

一般服务器为了保护真实 IP,会部署 CDN (内容分发网络),把用户经常访问的静态数据资源直接缓存到节点服务器上,当用户再次请求时,会直接分发给离用户近的节点服务器响应给用户。

这一机制使得我们无法直接探测到网站的真实 IP,而是得到**最近的目标节点的 CDN 服务器**。在实践中,我们首先 ping 目标主域名探测目标是否使用了 CDN,若使用多个地区 ping 到的 IP 结果不一致则证明存在 CDN。

为了查找真实 IP 我们需要绕过 CDN,常见方法有:

- a、**内部邮箱源**:一般邮件系统未经过 CDN 解析而是位于内部,所以可以通过查看邮件服务器域名, ping 这个域名获得目标真实 IP。
- b、**扫描网站测试文件**:例如 phpinfo、test, 查找目标网站真实 IP。
- c、**分站域名**:ping 分站的二级域名获取分站 IP,从而判断主站真实 IP。

(4)、枚举证书透明度公开日志中的子域名

证书授权机构会将每个 SSL/TLS 证书发布到公共日志中,而证书透明度是证书授权机构的一个项目。一个 SSL/TLS 证书通常包含域名、子域名和邮件地址等信息,这都是攻击者需要获得的有用信息。

因此,攻击者可以使用搜索引擎来搜索一些公开的 CT 日志,如以下网站所示:

- crt.sh: <https://crt.sh>
- censys: <https://censys.io>

然后查找某个域名所属证书,在证书里枚举子域名信息。

六、实验总结（收获和心得）（5 分）

本次实验的主要内容是信息收集，作为漏洞扫描、渗透测试之前的步骤，信息收集同样在攻击过程中起到了重要作用。我在预习 nmap 等扫描方式、撰写报告的过程中学习了各种扫描方式及其原理，在实验过程中进一步熟悉了 TCP/IP 协议和 Ping 命令。

实验分为主机存活性探测和手工信息收集两个部分。前者主要是让我熟悉了 Windows、Linux 环境下的各种扫描工具，了解了 Traceroute 追踪路由的方式和具体结果。而在手工信息收集实验中，我学习到进行 Web 渗透测试之前的信息收集方式，例如：通过 robots.txt 获取网站的隐私目录信息、利用备份信息来获取源码、通过 .git 文件还原源码。这些获取到的信息可以为攻击者下一步进行渗透攻击做铺垫，例如本次实验中得到的记录了用户信息的 excel 表格、php 脚本暴露的数据库表头项、调用的 sql 语句实例等等，都可能为后续渗透提供方向。

这次实验让我学习了渗透攻击目标网站所需的预先工作，以攻击方角度思考网站服务器的安全问题，在学到 Web 安全知识的同时也提高了安全意识，受益匪浅！

七、尚存问题或疑问、建议（5 分）

1、问题：

问题一：本实验中介绍的主机存活性探测都是基于没有部署防火墙等防护设备的情况，那么在实际应用中，如果目标系统被防火墙保护，该采取怎样的方式来探测系统内的存活主机的呢？

问题二：由于目前 Web 项目采用的开发策略一般是前后端分离，实验中通过 GitHack 工具重塑网站框架代码实质上是针对前端代码的恢复，而且实验中主要采用 PHP 脚本进行信息收集。那么对于存储在后端的数据库等信息，渗透测试中一般是如何利用前端收集到的信息进行攻击的呢？

2、 建议：

这次实验的教程完善，实验手册非常详细，在充分预习过后，我对于实验的体验非常好。有一个小小的建议就是，能不能将实验平台与环境，在课后也向同学们开放呢？大家可能在完成实验之后，还想在实验环境下做一些自己感兴趣的任務，例如自定义的渗透测试等等…