

**ANTONINE UNIVERSITY**

**Faculty of Engineering**

**Department of Technology in Computer Science**

Computer Security Project

Task 1 Report

**Carried out by**

Lynn Haydar, 202212989

Lynn AbdelKhalek, 202311399

Technology in Computer Science Student

## Table of Contents

Project Establishment .....	3
Installing theme for our website.....	4
Preparing necessary pages.....	5
Step 1 Robust User Authentication.....	6
Install necessary plugins .....	6
Limit Log in Attempts .....	6
Captcha .....	8
User Registration.....	9
Step 2 Access Control.....	12
Download plugin .....	12
Adding sample users .....	13
Restrict Pages Based on Roles.....	16
Redirecting to login .....	17
STEP 3_ Data Encryption and SSL.....	20
Step 4 SQL Injection Prevention.....	23
Step 5 Session Management.....	23
STEP 6 File Upload Security (VERY important).....	25
STEP 7 Error Handling & Logging .....	31
Enable WordPress logging.....	31
WP Activity Log plugin .....	33
Step 8 Penetration Testing .....	34
Table of Figures .....	43

# Project Establishment

Guide: <https://www.youtube.com/watch?v=um8BtHFNJBA>

Download XAMPP, download WordPress and extract it in htdocs XAMPP,

Start Apache and MySQL servers in XAMPP.

Go to PHP my admin and create a new database

You can access now your WordPress site by this link: <http://localhost/wordpress/>

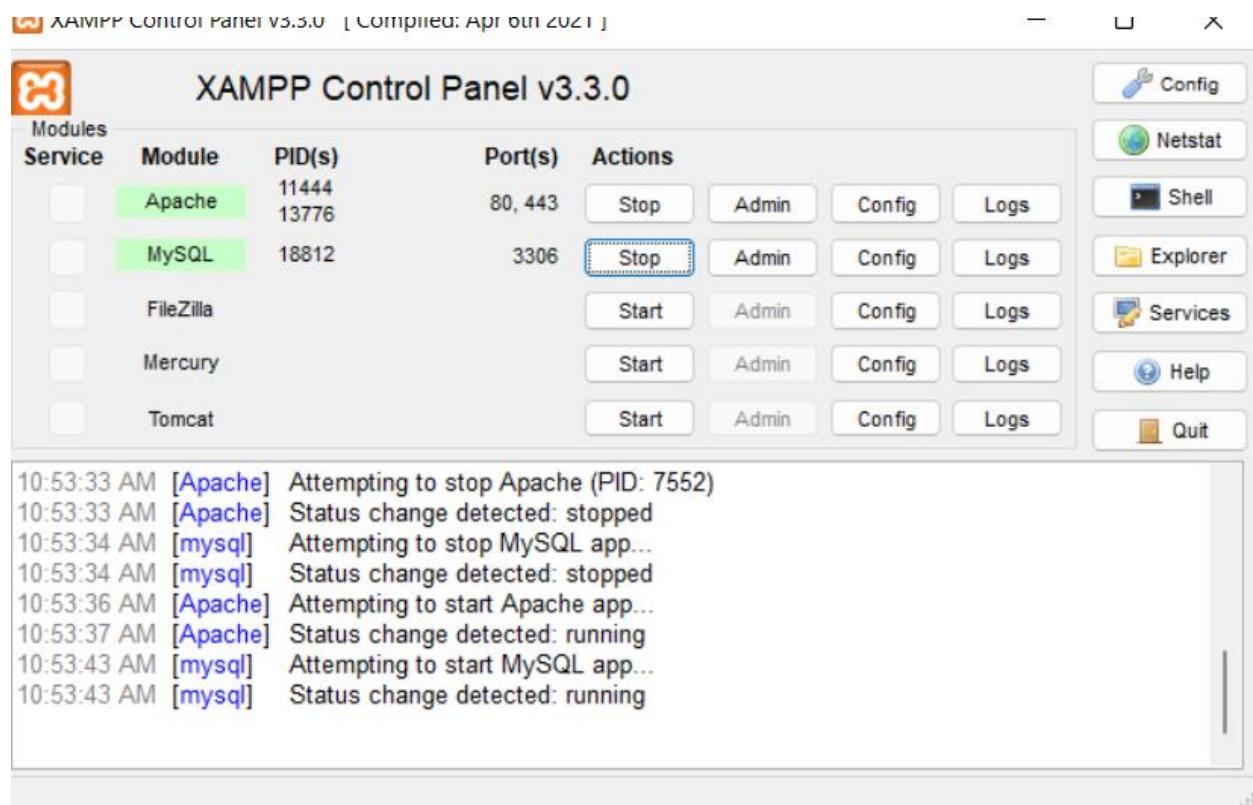


Figure 1\_starting services

Figure 2\_creating database

## Installing theme for our website

Open WordPress, go to Appearance and Themes, we chose Astra

Figure 3\_astra theme

Our website needs:

- Homepage
- Register system
- User roles

- Access control
- https: enable SSL on XAMPP
- Security Plugins
- File upload system
- Error logging
- SQL injection protection
- Penetration testing report

So we are going to build a photography website.

## Preparing necessary pages

Prepare the necessary pages for our website

Title	Author	Date	SEO Checks
About	root	Published 2025/11/25 at 10:54 am	<span>Needs Improvement</span>
Blog — Posts Page	root	Published 2025/11/25 at 10:55 am	<span>Needs Improvement</span>
Client private Gallery	root	Published 2025/11/25 at 11:03 am	
Contact	root	Published 2025/11/25 at 10:55 am	
Home — Front Page	root	Published 2025/11/25 at 10:54 am	
Photo Upload Center	root	Published 2025/11/25 at 12:13 pm	
Privacy Policy — Draft, Privacy Policy Page	root	Last Modified 2025/11/20 at 9:53 pm	
Sample Page	root	Published 2025/11/20 at 9:53 pm	
Services	root	Published 2025/11/25 at 10:55 am	
Title			

Figure 4\_pages

# Step 1 Robust User Authentication

## Install necessary plugins

Plugin	Description	Action
Anti-Malware Security and Brute-Force Firewall	This Anti-Virus/Anti-Malware plugin searches for Malware and other Virus like threats and vulnerabilities on your server and helps you remove them. It's always growing and changing to adapt to new threats so let me know if it's not working for you.	Enable auto-updates
File Upload Types by WPForms	Easily allow WordPress to accept and upload any file type extension or MIME type, including custom file types.	Enable auto-updates
Hello Dolly	This is not just a plugin, it symbolizes the hope and enthusiasm of an entire generation summed up in two words sung most famously by Louis Armstrong: Hello, Dolly. When activated you will randomly see a lyric from Hello, Dolly in the upper right of your admin screen on every page.	Enable auto-updates
Limit Login Attempts Reloaded	Block excessive login attempts and protect your site against brute force attacks. Simple, yet powerful tools to improve site performance.	Enable auto-updates
OttoKit	OttoKit helps people automate their work by integrating multiple apps and plugins, allowing them to share data and perform tasks automatically.	Enable auto-updates
Spectra	The Spectra extends the Gutenberg functionality with several unique and feature-rich blocks that help build websites faster.	Enable auto-updates
Starter Templates	Starter Templates is all in one solution for complete starter sites, single page templates, blocks & images. This plugin offers the premium library of ready templates & provides quick access to beautiful Pixabay images that can be imported in your website easily.	Enable auto-updates
SureForms	A simple yet powerful way to create modern forms for your website.	Enable auto-updates
SureRank SEO	Grow traffic of your website with SureRank — a lightweight SEO toolkit plugin for WordPress users who want better rankings without the complexity.	Enable auto-updates
User Role Editor	Change/add/delete WordPress user roles and capabilities.	Enable auto-updates
Wordfence Security	Wordfence Security - Anti-virus, Firewall and Malware Scan	Enable auto-updates
WP Activity Log	Identify WordPress security issues before they become a problem. Keep track of everything happening on your WordPress, including users activity. Similar to Linux Syslog, WP Activity Log generates an activity log with a record of everything that happens on your WordPress websites.	Enable auto-updates
Plugin	Description	Automatic Updates

Figure 5\_plugins installed

1. **Limit Login Attempts Reloaded** → prevents brute-force attacks
2. **Captcha4wp** → CAPTCHAs on login, registration, contact form

## Limit Log in Attempts

The screenshot shows the configuration interface for the "Limit Login Attempts" plugin. On the left, there's a sidebar with navigation links: Posts, Media, Pages, Comments, SureForms, OttoKit, SureRank, Spectra, Appearance, Plugins (with 1 notification), Users, and the current active app, Limit Login Attempts. Below that are links for Dashboard, Settings (Logs, Debug, Help, Free Upgrade), Tools, and Settings. The main area has tabs for "Active App" (Local (Free version)) and "Cloud App". Under "Local App", there's a "Lockout" section with settings: allowed retries (3), minutes lockout (20), lockouts increase lockout time to (4), and hours until retries are reset (24). A note explains that after a specific IP fails 4 times, a 20-minute lockout is applied, and if more failed attempts occur within 24 hours, the lockout duration is extended by 24 hours. There's also a "Trusted IP Origins" section with a field for REMOTE\_ADDR. At the bottom, there's a section titled "Why Use Our Premium Cloud App?" with icons for absorbing site load caused by attacks, getting premium support, using intelligent IP denial/unblocking technology, running auto backups of access control lists, and sync between multiple domains, along with a note about no contract.

Figure 6\_limit login attempts

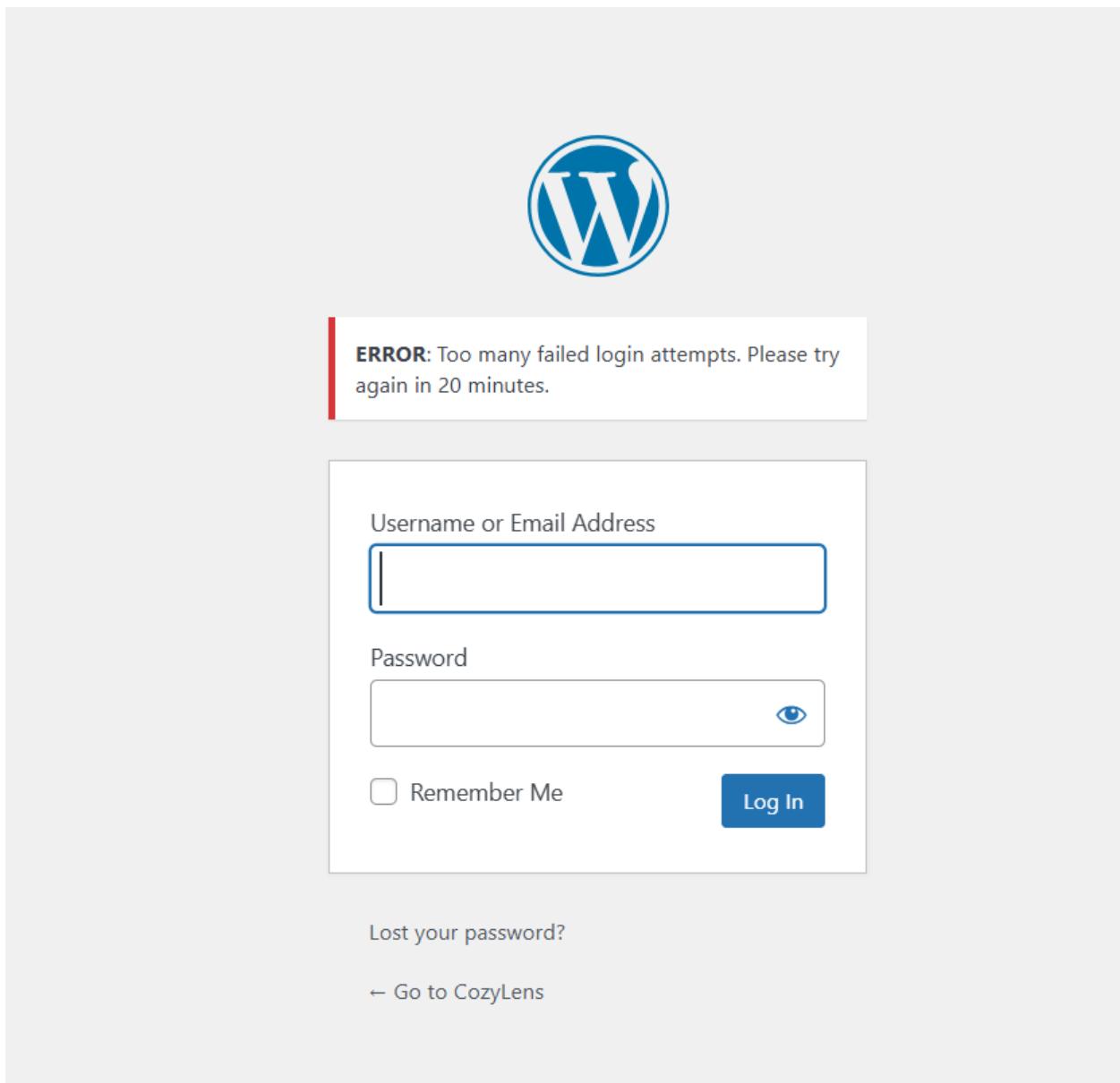


Figure 7\_locked after failed attempts

This is brute force protection

## Captcha

The screenshot shows the CAPTCHA 4WP configuration page within the WordPress admin interface. A prominent purple banner at the top announces the acquisition by WPcube, stating: "CAPTCHA 4WP has been acquired by WPcube. As part of our ongoing efforts to focus more on our core plugins, we've decided to sell CAPTCHA 4WP to WPcube. Devesh and his team are well-known for their expertise in WordPress, and we're confident that CAPTCHA 4WP will continue to thrive under their leadership." Below this, a blue button says "Read the announcement". The main content area is titled "CAPTCHA integration & configuration" and includes a message about the configuration wizard. It displays current configuration details: CAPTCHA version V2 Checkbox, Site key 6Ld-8BosAAAAAP2DnT7HqoDDVInqgaXNdd-P9oez, and Secret key 6Ld-8BosAAAAAHBVCYToX77UH\_Emi2r3TISQYqtv. There are buttons for "Reconfigure CAPTCHA integration" and "Remove CAPTCHA integration". An "Optional settings: Fine-tune CAPTCHA to your requirements" section allows users to specify sensitivity for the CAPTCHA check. To the right, a sidebar for "CAPTCHA 4WP" offers an upgrade to Premium with various features listed.

Figure 8\_Captcha 4wp

Go to <https://www.google.com/recaptcha/admin/create>

The screenshot shows the Google reCAPTCHA configuration page. It starts with a "CozyLens" header. The "reCAPTCHA type" section is set to "Challenge (v2)". The "Domains" section lists "localhost". In the "Google Cloud Platform" section, the "Project name\*" dropdown is set to "My First Project". A note below says "The selected Google Cloud project will be used. It looks like you've used Google Cloud before." At the bottom, there's a "GOOGLE CLOUD PLATFORM" link and a "Privacy + Terms" link.

Figure 9\_creating new key

Site key: 6Ld-8BosAAAAAP2DnT7HqoDDVInqgaXNdd-P9oez

Secret key: 6Ld-8BosAAAAAHBVCYToX77UH\_Emi2r3TISQYqtv

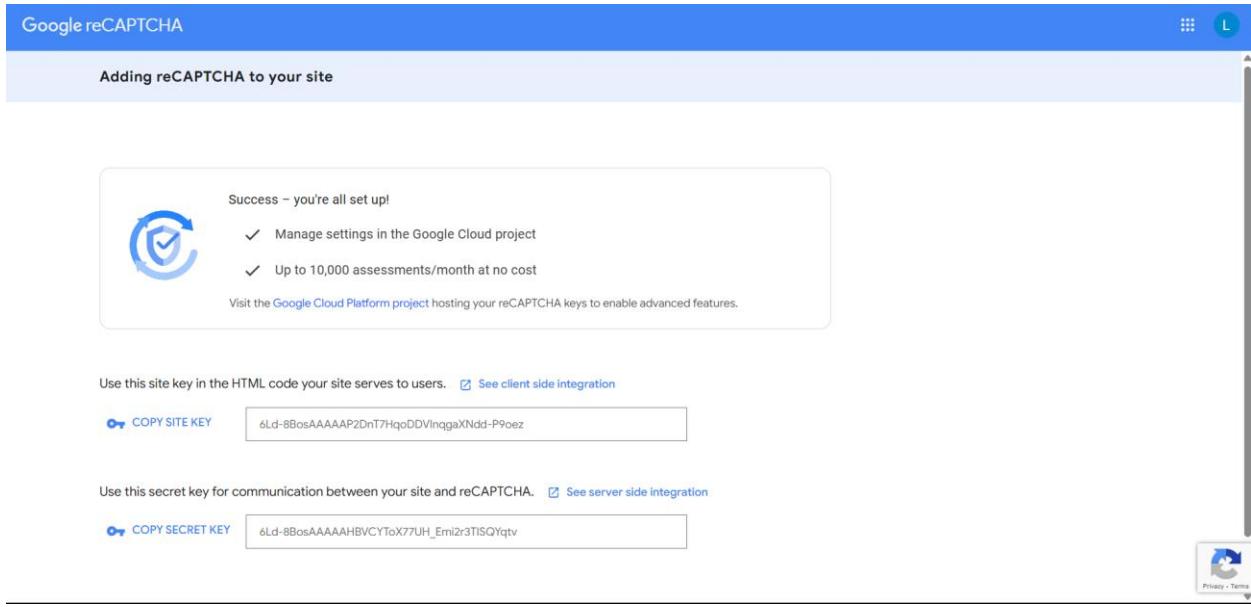


Figure 10\_Integration keys

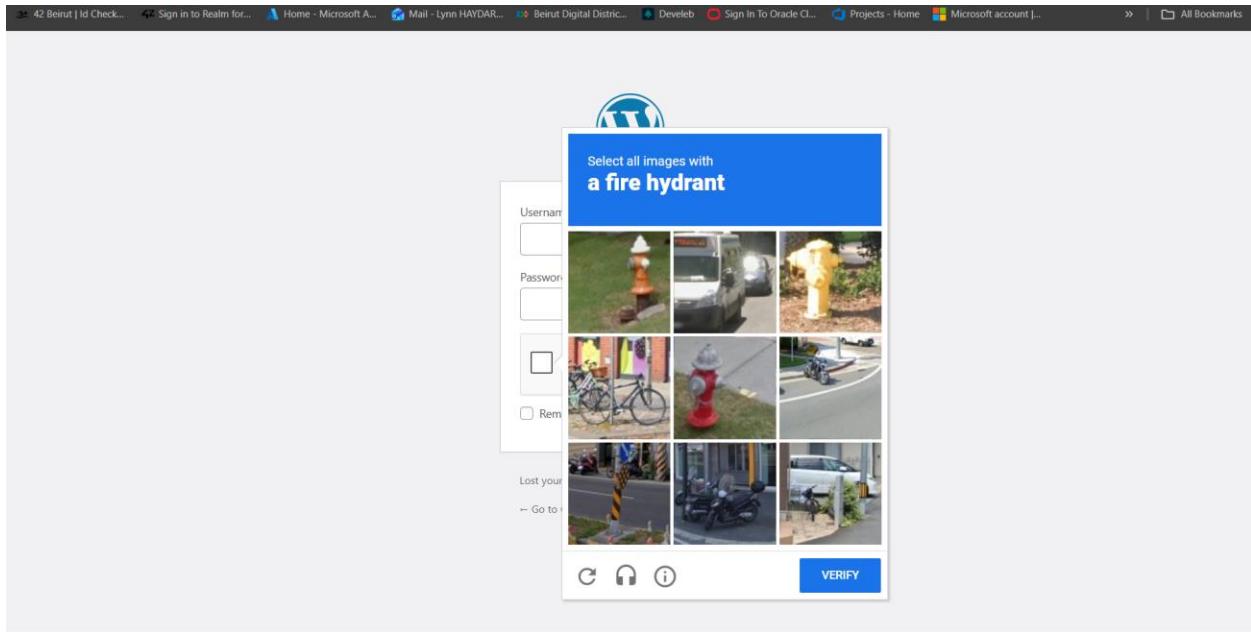


Figure 11\_checking results

This block automated login bots

## User Registration

Go to settings-> general-> check anyone can register

Tools

Settings

General

Writing

Reading

Discussion

Media

Permalinks

Privacy

Security by CleanTalk

Site Address (URL)  Enter the same address here unless you [want your site home page to be different from](#)

Administration Email Address  This address is used for admin purposes. If you change this, an email will be sent to you

Membership  Anyone can register

New User Default Role

Site Language

Figure 12 \_checking

Now download this plugin

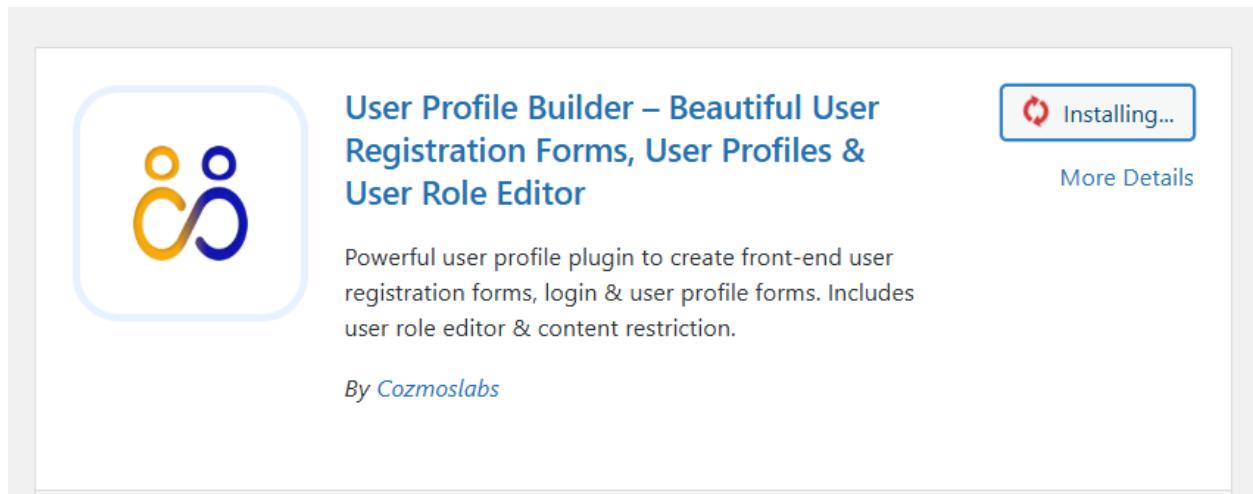


Figure 13\_downloading plugin

Test here: <https://localhost/wordpress/signip/>

**SECURITY**

**Minimum Password Length**


Enter the minimum characters the password should have. Leave empty for no minimum limit

**Minimum Password Strength**

Strong

A stronger password strength will probably force the user to not reuse passwords from other websites.

**Password Recovery Page**

None

Select the page which contains the "[wppb-recover-password]" shortcode.

UPDATE SETTINGS

**TWO-FACTOR AUTHENTICATION**

Two-Factor Authentication
Toggle
Enable the Google Authenticator functionality.

Increase the security of your user accounts with 2 Factor Authentication by upgrading to [Basic](#) or [Pro](#) versions.

**OTHER FEATURES**

^

Figure 14\_check necessary settings

Password \*

...

Minimum length of 10 characters.

The password must have a minimum strength of Strong

Very weak

Repeat Password \*

Figure 15\_123

Password \*

*Minimum length of 10 characters.*

*The password must have a minimum strength of Strong*

**Strong**

Repeat Password \*

Send these credentials via email.

**Register**

Figure 16\_strong

## Step 2 Access Control

### Download plugin

Download plugin **User Role Editor** → create custom roles

Go to users -> User Role Editor -> add role

User Role Editor

Select Role and change its capabilities: **Subscriber (subscriber)**

Show capabilities in human readable form  Show deprecated capabilities

Group (Total/Granted)	<input type="checkbox"/> Quick filter: <input type="text"/> <input type="checkbox"/> Granted Only <input type="button" value="Columns: 1"/>
All (75/2)	<input type="checkbox"/> activate_plugins <input type="checkbox"/> create_posts <input type="checkbox"/> create_users <input type="checkbox"/> delete_others_pages <input type="checkbox"/> delete_others_posts <input type="checkbox"/> delete_pages <input type="checkbox"/> delete_plugins <input type="checkbox"/> delete_posts <input type="checkbox"/> delete_private_pages <input type="checkbox"/> delete_private_posts <input type="checkbox"/> delete_published_pages <input type="checkbox"/> delete_published_posts <input type="checkbox"/> delete_themes <input type="checkbox"/> delete_users <input type="checkbox"/> edit_dashboard <input type="checkbox"/> edit_others_pages <input type="checkbox"/> edit_others_posts <input type="checkbox"/> edit_pages <input type="checkbox"/> edit_plugins <input type="checkbox"/> edit_posts <input type="checkbox"/> edit_private_pages <input type="checkbox"/> edit_private_posts

**Update**

Add Role  
Rename Role  
Add Capability  
Delete Capability

Figure 17\_add role

import

**Add New Role**

Role name (ID):

Display Role Name:

Make copy of:

delete\_users  
 edit\_dashboard  
 edit\_others\_pages  
 edit\_others\_posts  
 ...

Figure 18\_photographer role

import

**Add New Role**

Role name (ID):

Display Role Name:

Make copy of:

publish\_pages  
 publish\_posts

Figure 19\_subscriber

Photographer: upload photos

Client: access the private gallery

Now we need test accounts, so we add sample users

## Adding sample users

Go to users -> add user

## Add User

Create a brand new user and add them to this site.

Username (required)

photographer1

Email (required)

photographer1@cozylens.com

First Name

Last Name

Website

Password

Generate password

Password123

 Hide

Very weak

Confirm Password

Confirm use of weak password

Send User Notification

Send the new user an email about their account

Role

Photographer ▾

Figure 20\_adding new user with role photographer

## Add User

Create a brand new user and add them to this site.

Username (required)

client1

Email (required)

client1@cozylens.com

First Name

Last Name

Website

Password

Generate password

Password123

 Hide

Very weak

Confirm Password

Confirm use of weak password

Send User Notification

Send the new user an email about their account

Role

Client ▾

Figure 21\_adding new user with role client

Users						Add User	Screen Options ▾	Help ▾					
New user created. <a href="#">Edit user</a>						X							
All (3)   Administrator (1)   Photographer (1)   Client (1)						Search Users							
Bulk actions	▼	Apply	Change role to...	▼	Change	Grant Roles	Add role...	▼	Add	Revoke role...	▼	Revoke	3 items
<input type="checkbox"/>	Username	client1	Name	—		Email	client1@cozylens.com		Role	Client	Posts	0	
<input type="checkbox"/>	photographer1		Name	—		Email	photographer1@cozylens.com		Role	Photographer	Posts	0	
<input type="checkbox"/>	root		Name	—		Email	lynnhaydar5@gmail.com		Role	Administrator	Posts	4	

Figure 22\_users created

## Restrict Pages Based on Roles

Go to the page “Client Private Gallery” and limit access

Content Permissions (Members)

Roles	Limit access to the content to users of the selected roles.
<input type="checkbox"/> Paid Memberships	<input type="checkbox"/> Administrator <input type="checkbox"/> Author <input checked="" type="checkbox"/> Client <input type="checkbox"/> Contributor <input type="checkbox"/> Editor <input checked="" type="checkbox"/> Photographer <input type="checkbox"/> Subscriber
<input type="checkbox"/> Error Message	If no roles are selected, everyone can view the content. The author, any users who can edit the content, and users with the <code>restrict_content</code> capability can view the content regardless of role.

Figure 23\_access roles client private gallery

Do the same for the “Photo Upload Center” page

Figure 24\_roles photo upload center

## Redirecting to login

Go to Apparencies -> Theme File Editor -> functions.php -> add these functions

```

Editing Astra (active)
File: functions.php
Selected file content:
207 require_once ASTRA_THEME_DIR . 'inc/core/deprecated/deprecated-functions.php';
208
209 // Redirect unauthorized users to login page for restricted pages
210 function cozylens_redirect_unauthorized() {
211     if ( is_page('client-private-gallery') || is_page('photo-upload-center') ) {
212         if ( !is_user_logged_in() ) {
213             wp_redirect( wp_login_url() );
214             exit;
215         }
216     }
217 }
218 add_action('template_redirect', 'cozylens_redirect_unauthorized');
219
220 // Redirect clients to front-end after login
221 function cozylens_redirect_after_login($redirect_to, $request, $user) {
222     if (isset($user->roles) && is_array($user->roles)) {
223         if (in_array('client', $user->roles)) {
224             return site_url('/client-private-gallery/');
225         }
226     }
227     return $redirect_to;
228 }
229 add_filter('login_redirect', 'cozylens_redirect_after_login', 10, 3);
230
231 // Block clients from accessing admin dashboard
Documentation: Function Name... Look Up
File edited successfully.
Update File

```

Figure 25\_functions.php

```

1. // Redirect unauthorized users to login page for restricted pages
2. function cozylens_redirect_unauthorized() {
3.     if ( is_page('client-private-gallery') || is_page('photo-upload-center') ) {
4.         if ( !is_user_logged_in() ) {
5.             wp_redirect( wp_login_url() );
6.             exit;
7.         }
8.     }
9. }
10. add_action('template_redirect', 'cozylens_redirect_unauthorized');
11.
12. // Redirect clients to front-end after login
13. function cozylens_redirect_after_login($redirect_to, $request, $user) {

```

```

14.     if (isset($user->roles) && is_array($user->roles)) {
15.         if (in_array('client', $user->roles)) {
16.             return site_url('/client-private-gallery/');
17.         }
18.     }
19.     return $redirect_to;
20. }
21. add_filter('login_redirect', 'cozylens_redirect_after_login', 10, 3);
22.
23. // Block clients from accessing admin dashboard
24. function cozylens_block_wp_admin_access() {
25.     if (is_user_logged_in() && current_user_can('client') && is_admin()) {
26.         wp_redirect(site_url('/client-private-gallery/'));
27.         exit;
28.     }
29. }
30. add_action('init', 'cozylens_block_wp_admin_access');
31.
32. // Hide admin bar for clients
33. add_filter('show_admin_bar', function($show){
34.     if (current_user_can('client')) {
35.         return false;
36.     }
37.     return $show;
38. });
39.
40. // Redirect photographers to upload center after login
41. function cozylens_redirect_photographer($redirect_to, $request, $user) {
42.     if (isset($user->roles) && is_array($user->roles)) {
43.         if (in_array('photographer', $user->roles)) {
44.             return site_url('/photo-upload-center/');
45.         }
46.     }
47.     return $redirect_to;
48. }
49. add_filter('login_redirect', 'cozylens_redirect_photographer', 11, 3);
50.
51. // Block photographers from accessing wp-admin
52. function cozylens_block_photographer_admin() {
53.     if (is_user_logged_in() && current_user_can('photographer') && is_admin()) {
54.         wp_redirect(site_url('/photo-upload-center/'));
55.         exit;
56.     }
57. }
58. add_action('init', 'cozylens_block_photographer_admin');
59.

```

To test, open incognito mode, go to <http://localhost/wordpress/client-private-gallery>

You will be redirected to a login page

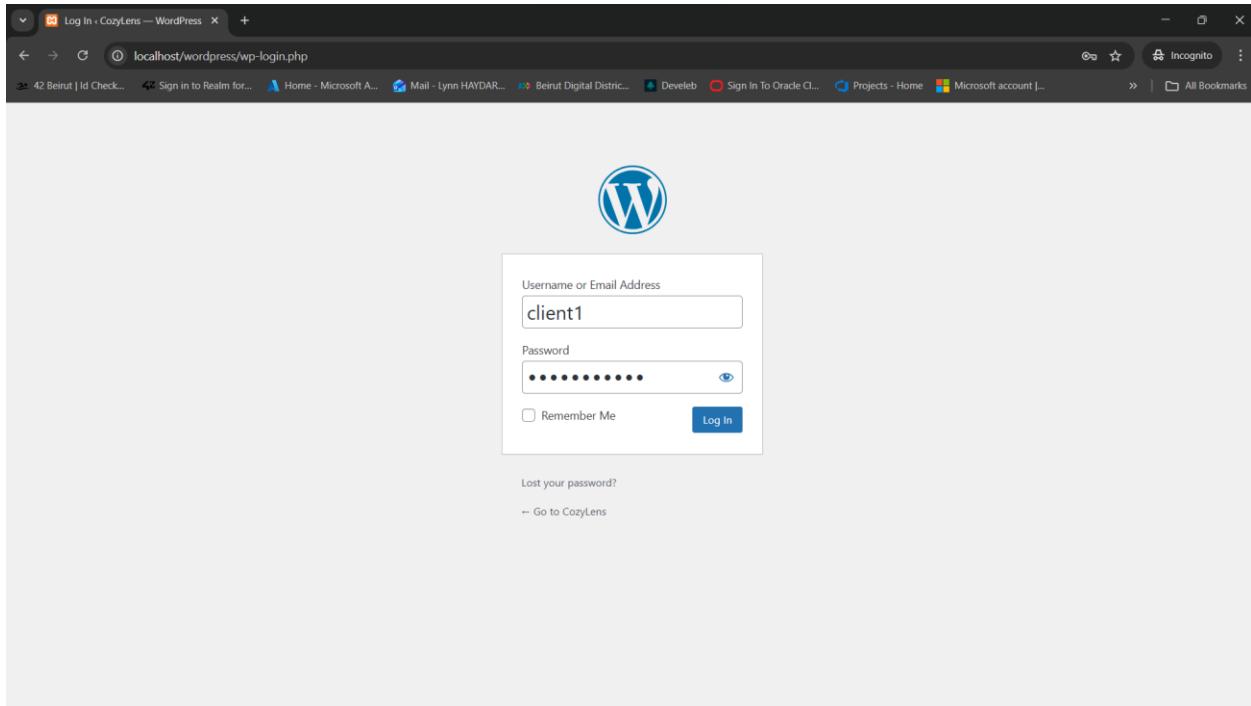


Figure 26\_redirected to login page

### Log in as client 1

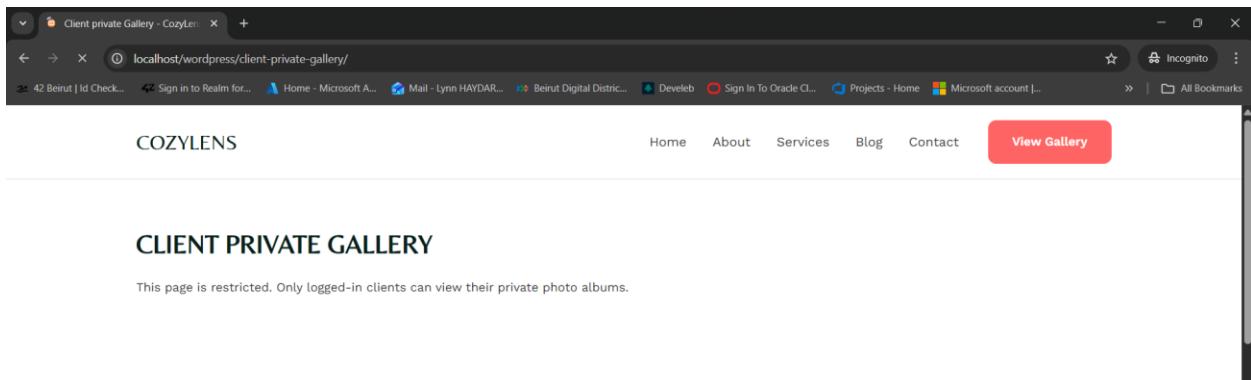


Figure 27\_log in successful

Do the same for the photographer

Go to: <http://localhost/wordpress/photo-upload-center>

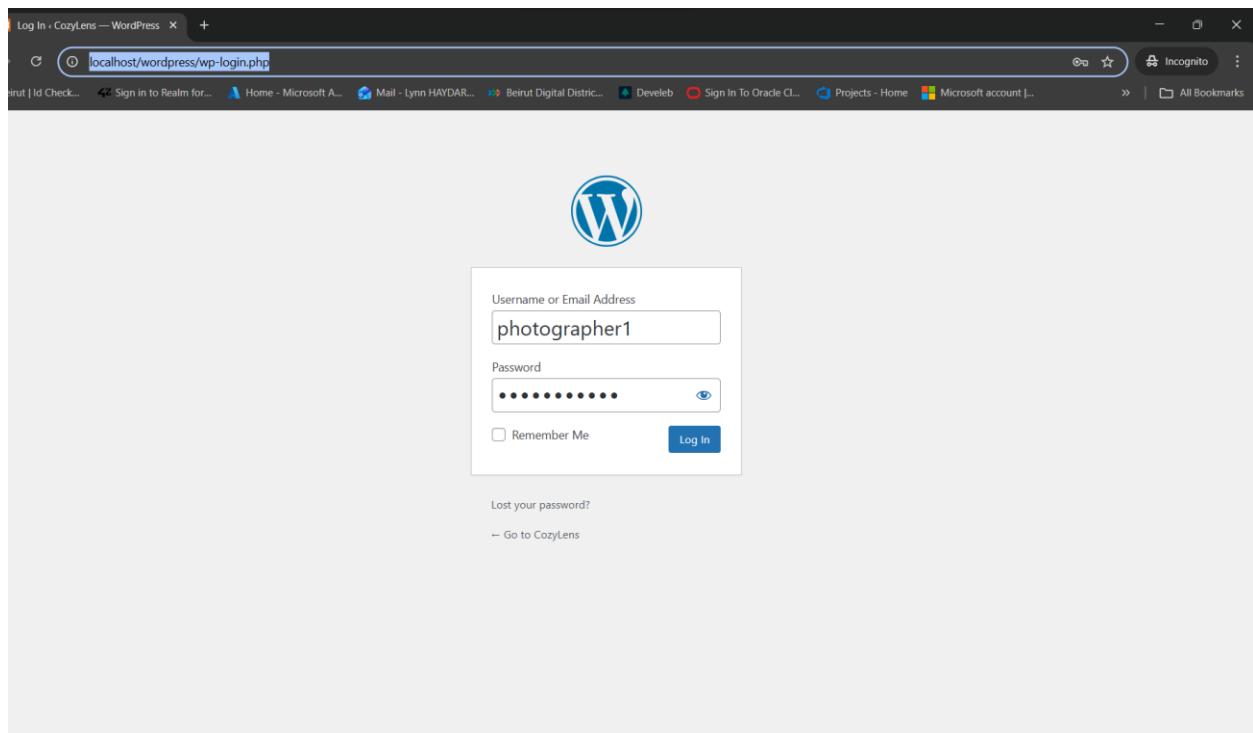


Figure 28\_redirected to login

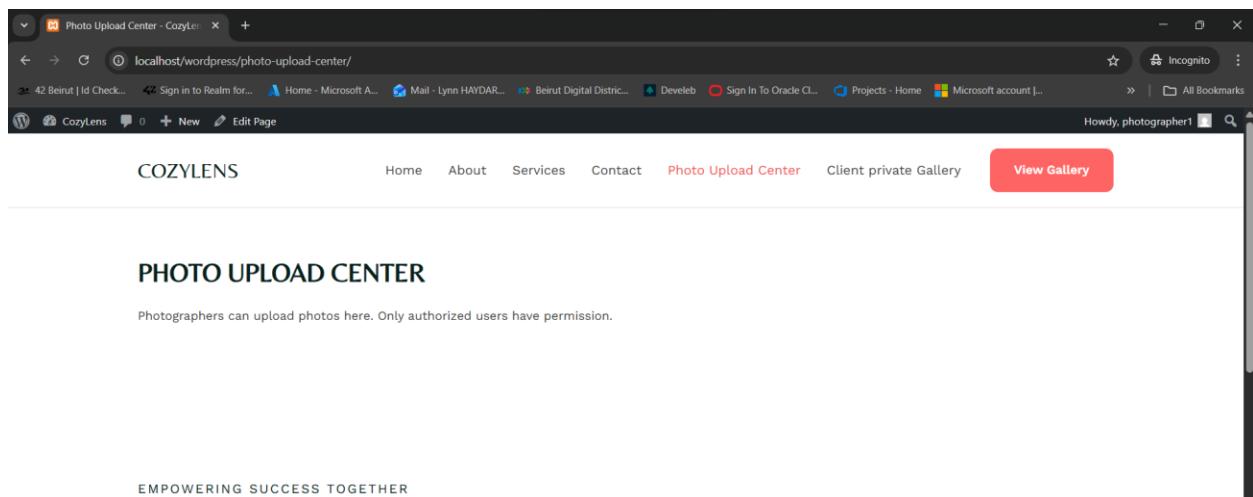


Figure 29\_log in successful

This is for the unauthorized access requirement

## STEP 3\_Data Encryption and SSL

An SSL (Secure Sockets Layer) certificate is a digital certificate that verifies a website's identity and enables an encrypted connection between a server and a browser. When a website has a

valid SSL certificate, its URL will display https:// and a padlock icon in the browser's address bar, indicating a secure and trustworthy connection.

Install plugin: “Really Simple SSL”.

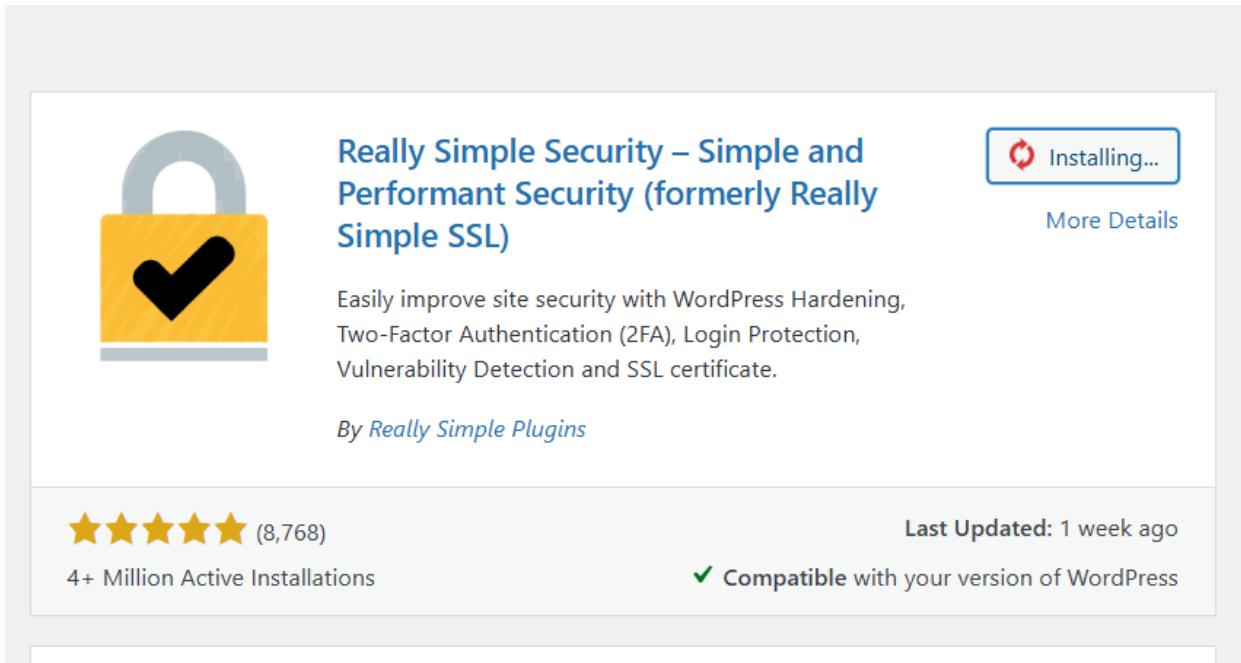


Figure 30\_installing plugin

Visit <https://localhost/wordpress>

This window appears, but it's okay it means that the plugin works

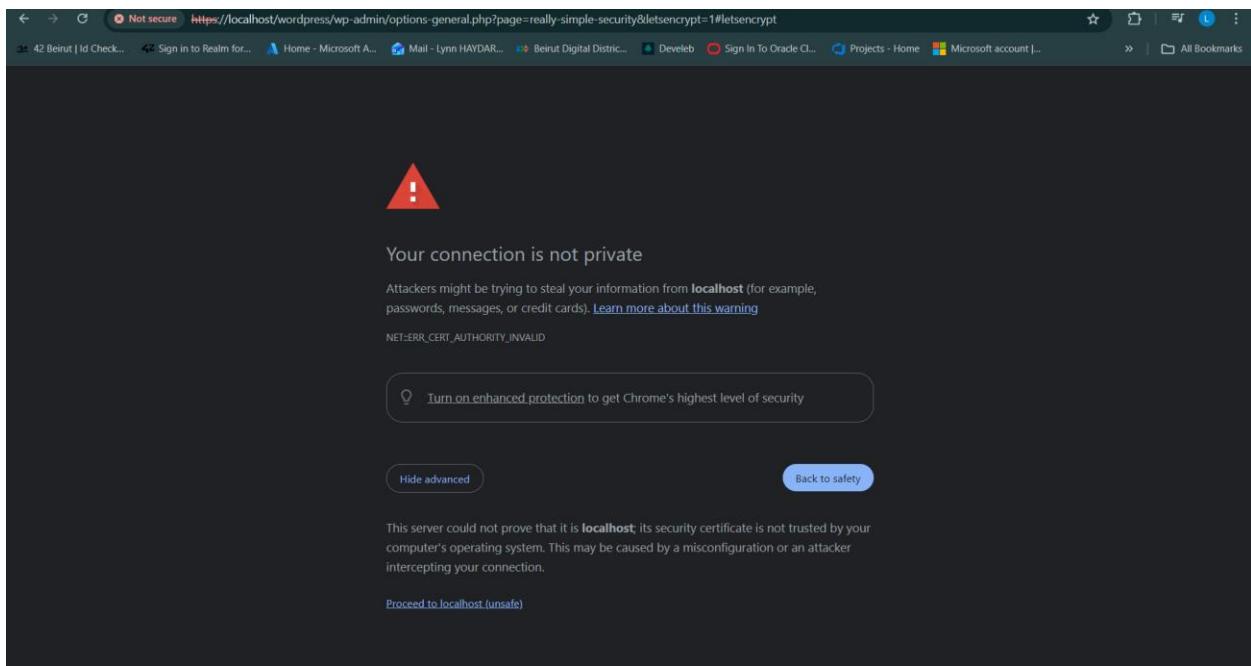


Figure 31\_ssl works



Figure 32\_https

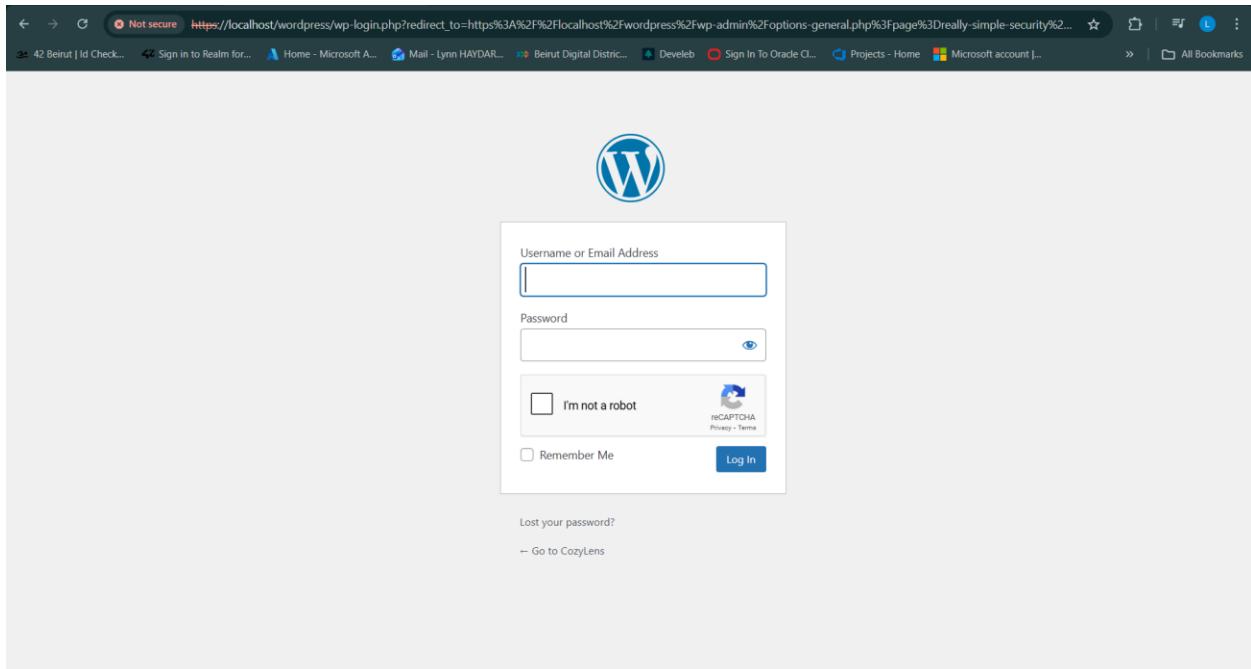
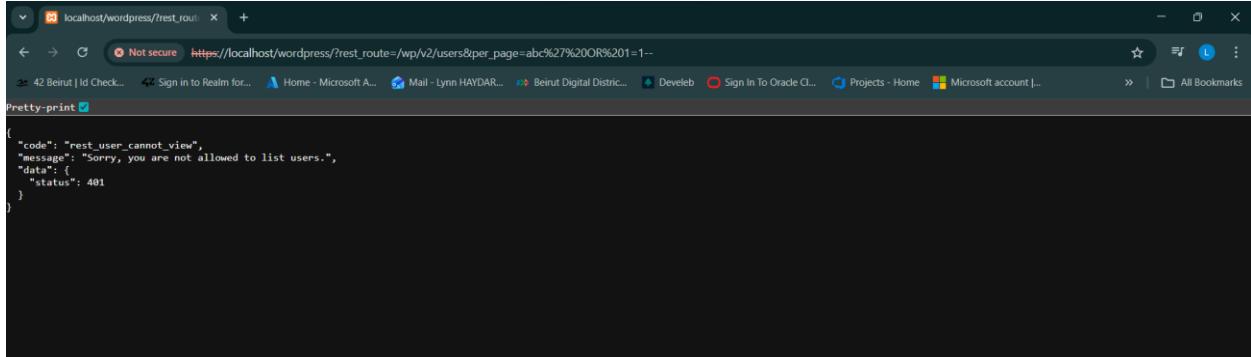


Figure 33\_proceding

## Step 4 SQL Injection Prevention

Test with this url: `http://localhost/wordpress/?rest_route=/wp/v2/users&per_page=abc' OR 1=1—`



```
{ "code": "rest_user_cannot_view", "message": "Sorry, you are not allowed to list users.", "data": { "status": 401 } }
```

Figure 34\_testing

And `http://localhost/wordpress/?p=1 UNION SELECT md5(1)—`

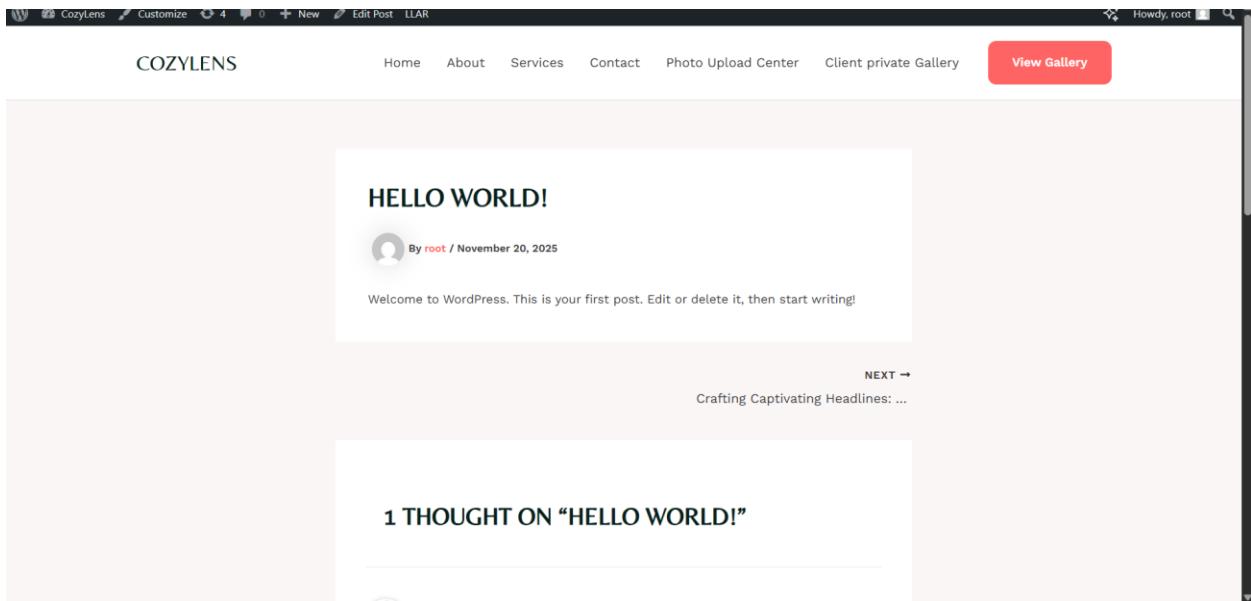


Figure 35\_test successful

Website didn't break and show sql data.

## Step 5 Session Management

Install this plugin

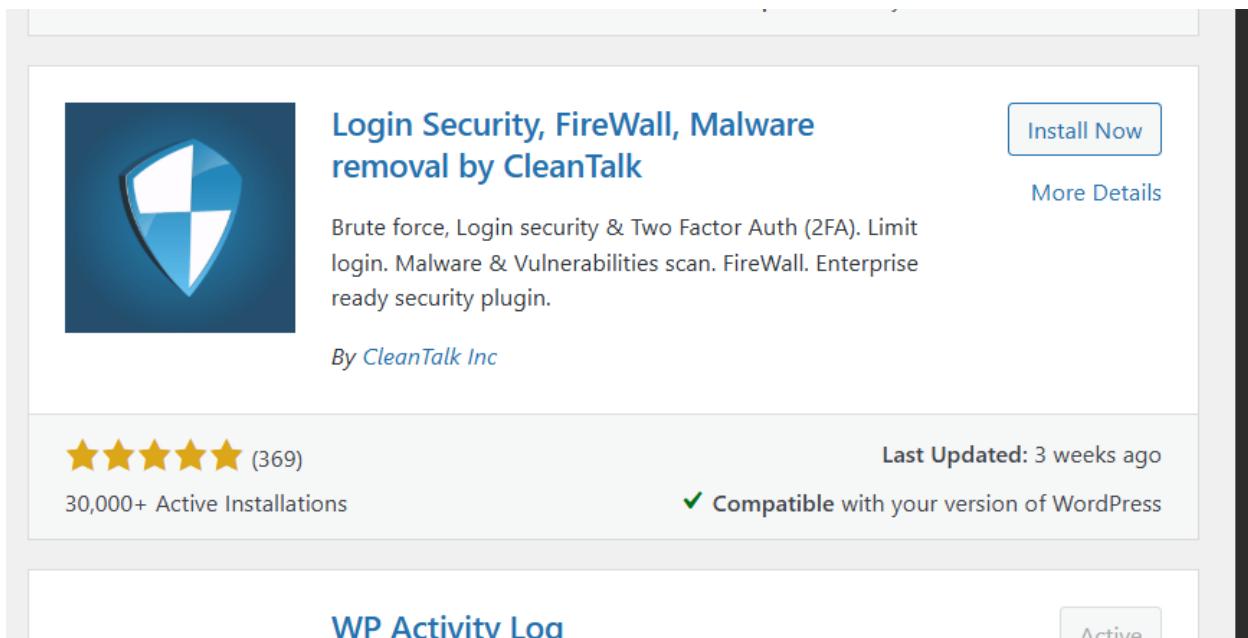


Figure 36\_installing plugin

## Get manual access key

You need only 5 minutes and your website is protected with CleanTalk Security

You will get a key to activate the plugin, installation instructions and the security plugin.

Website name is not correct. Please double check the name

Create your CleanTalk account.

lynnhaydar5@gmail.com

Cozylens.local

Your account password will be sent to your email.  
By signing up, you agree with license.

CREATE ACCOUNT

Have an account? Log In.

What happens next?  
You will get a key to activate the plugin, installation instructions and the security plugin.

Figure 37\_getting access key

And paste it

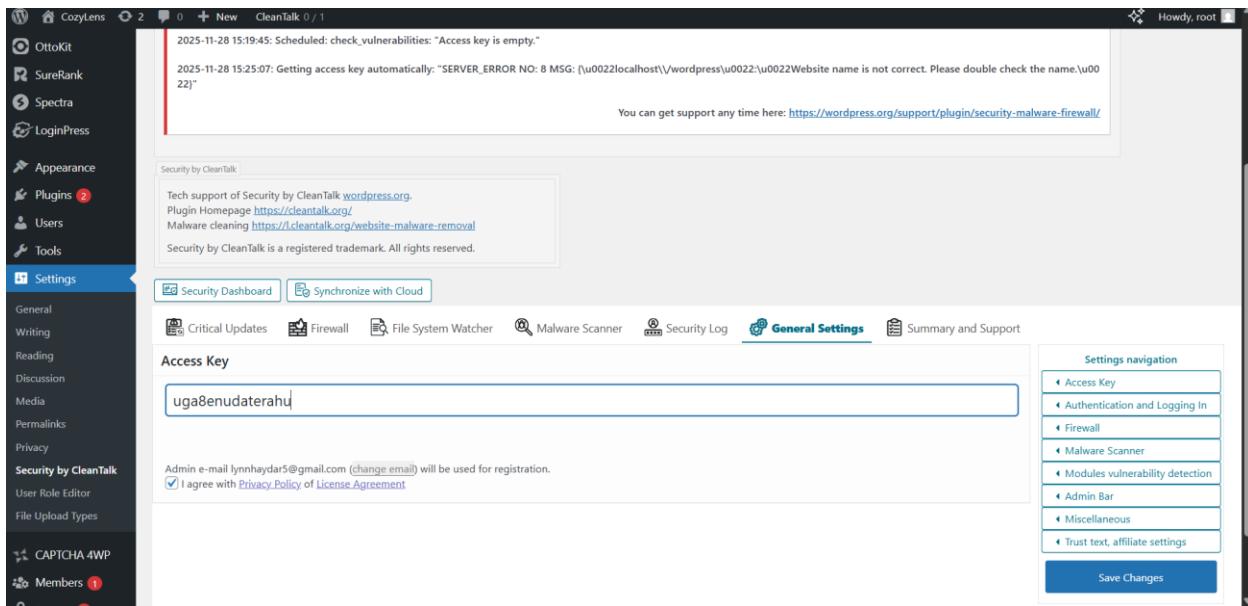


Figure 38\_pasting access key

## Modify settings and save

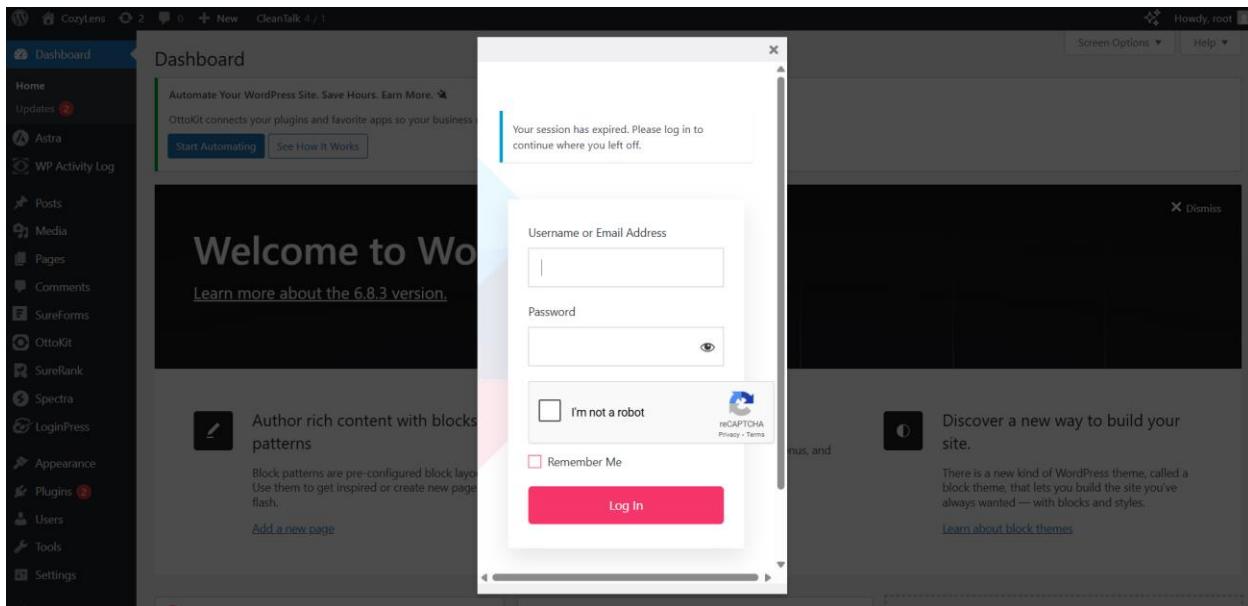


Figure 39\_session expired after one minute

## STEP 6 File Upload Security (VERY important)

Install this plugin

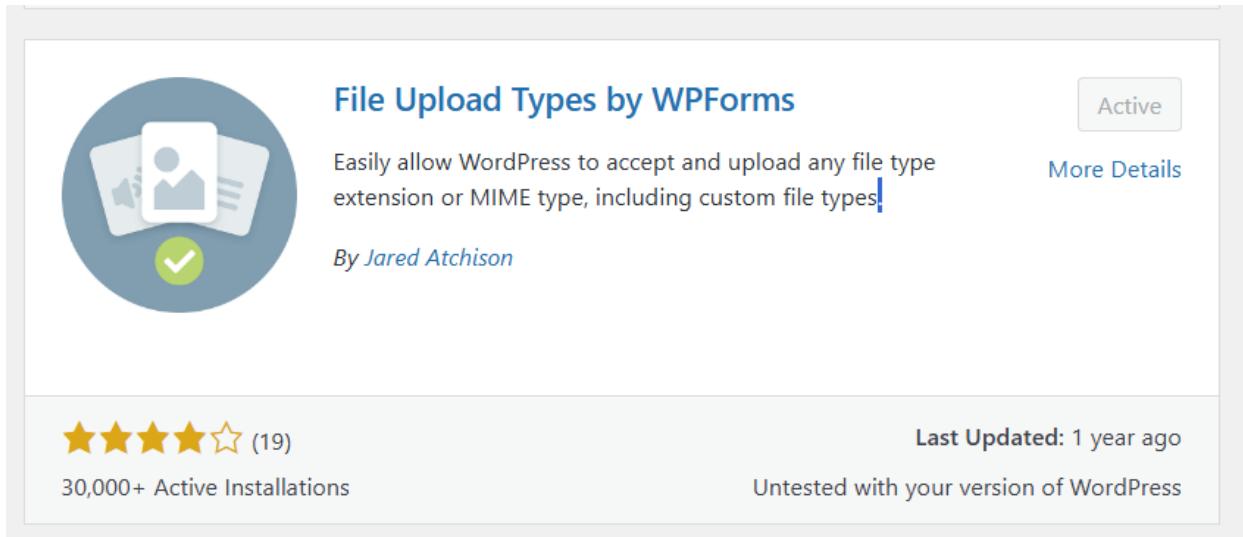


Figure 40\_file uploads plugin

We chose to enable these file types:

- jpg
- jpeg
- png
- webp
- gif (optional)
- heic (support iPhone uploads)

PNG, JPG, JPEG, GIF are already allowed by default.

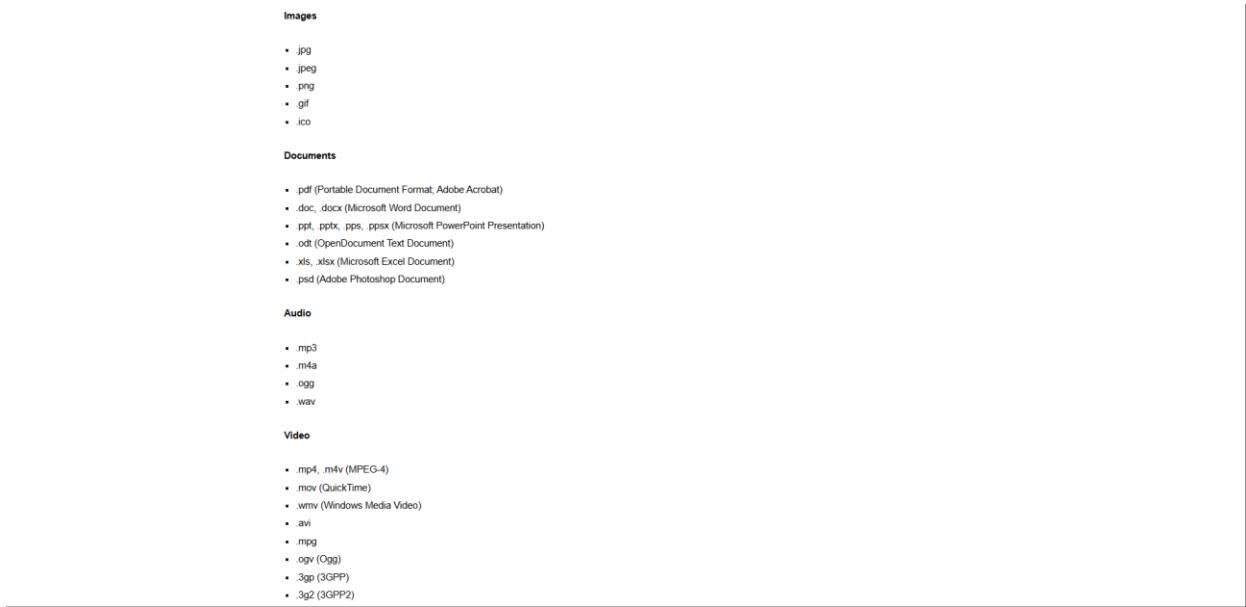


Figure 41\_allowed by WordPress

Let's Enable webp and heic

Click on add your custom file type

A screenshot of the 'Add File Upload Types' page. It shows a list of file types that can be enabled, including image/webp and webp. A note says 'Don't see what you need? No problem, add your custom file types.' A link to 'files WordPress allows by default' is also present.

Figure 42\_adding custom file type

A screenshot of the 'Save Settings' button on the configuration page. The button is blue with white text and is located at the bottom left of the configuration interface.

Figure 43\_configuring it

## Settings

Your settings have been saved.

### Add File Upload Types

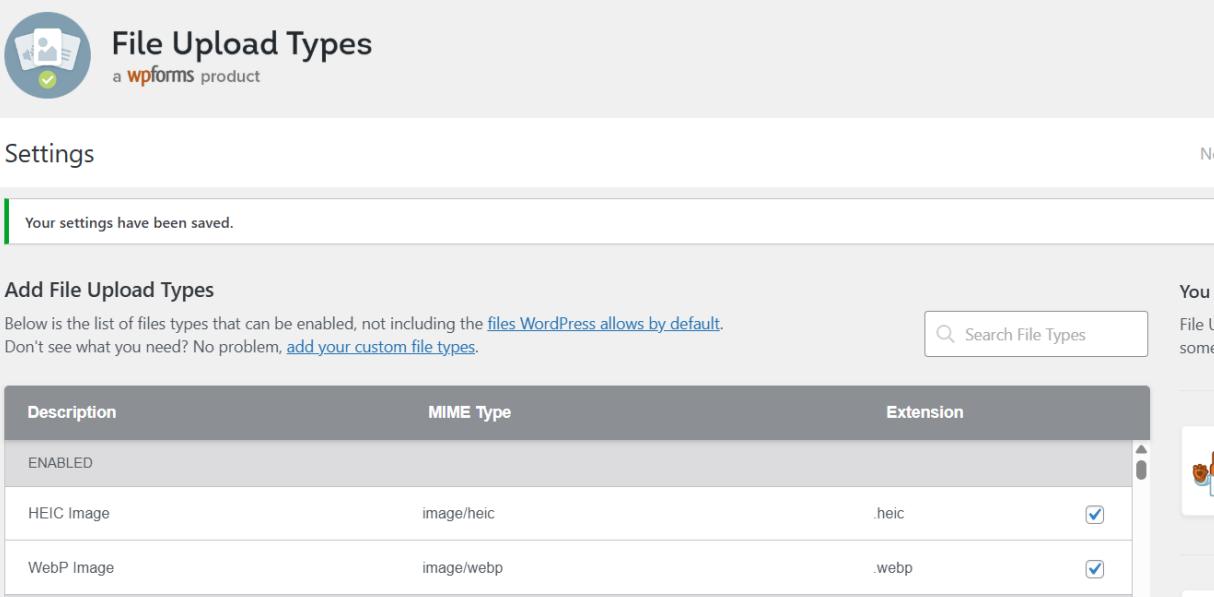
Below is the list of files types that can be enabled, not including the [files WordPress allows by default](#).  
Don't see what you need? No problem, [add your custom file types](#).

Search File Types

Description	MIME Type	Extension	
ENABLED			
WebP Image	image/webp	.webp	<input checked="" type="checkbox"/>
AVAILABLE			
3D Crossword Plugin	application/vnd.hzn-3d-crossword application/hzn-3d-crossword model/x3d+xml	.x3d	<input type="checkbox"/>

Figure 44\_save

## Same for heic



The screenshot shows the 'File Upload Types' settings page from WPForms. At the top, there's a logo and the text 'File Upload Types' followed by 'a wpforms product'. Below this is a 'Settings' header with a 'Need help?' link. A message 'Your settings have been saved.' is displayed. The main area is titled 'Add File Upload Types' with a note about allowed file types and a 'Search File Types' button. A table lists file types under 'ENABLED': HEIC Image (image/heic, .heic, checked) and WebP Image (image/webp, .webp, checked). There's also a sidebar with a cartoon bear icon.

Description	MIME Type	Extension	
ENABLED			
HEIC Image	image/heic	.heic	<input checked="" type="checkbox"/>
WebP Image	image/webp	.webp	<input checked="" type="checkbox"/>

Figure 45\_done

To test it go to apperances, theme file editor -> -> functions.php -> add this functions

```
1. 1. function cozylens_test_upload_form() {  
2. 2.     ob_start(); ?>  
3. 3.  
4. 4.     <form method="post" enctype="multipart/form-data">  
5. 5.         <label>Select a file to upload:</label><br><br>  
6. 6.         <input type="file" name="test_upload" required><br><br>  
7. 7.         <input type="submit" name="submit_test_upload" value="Upload File">  
8. 8.     </form>
```

```

9. 9.
10. 10. <?php
11. 11. if (isset($_POST["submit_test_upload"])) {
12. 12.     require_once(ABSPATH . 'wp-admin/includes/file.php');
13. 13.
14. 14.     $upload = wp_handle_upload($_FILES['test_upload'], array('test_form' => false));
15. 15.
16. 16.     if (isset($upload['error'])) {
17. 17.         echo "<p style='color:red;'><strong>UPLOAD BLOCKED:</strong> " .
esc_html($upload['error']) . "</p>";
18. 18.     } else {
19. 19.         echo "<p style='color:green;'><strong>UPLOAD SUCCESS:</strong> File uploaded to: " .
esc_url($upload['url']) . "</p>";
20. 20.     }
21. 21. }
22. 22.
23. 23. return ob_get_clean();
24. 24. }
25. 25. add_shortcode('upload_test', 'cozylens_test_upload_form');
26. 26.
27.

```

This creates a shortcode: [upload\_test]

Go to photo upload center page and add a shortcode block and paste it

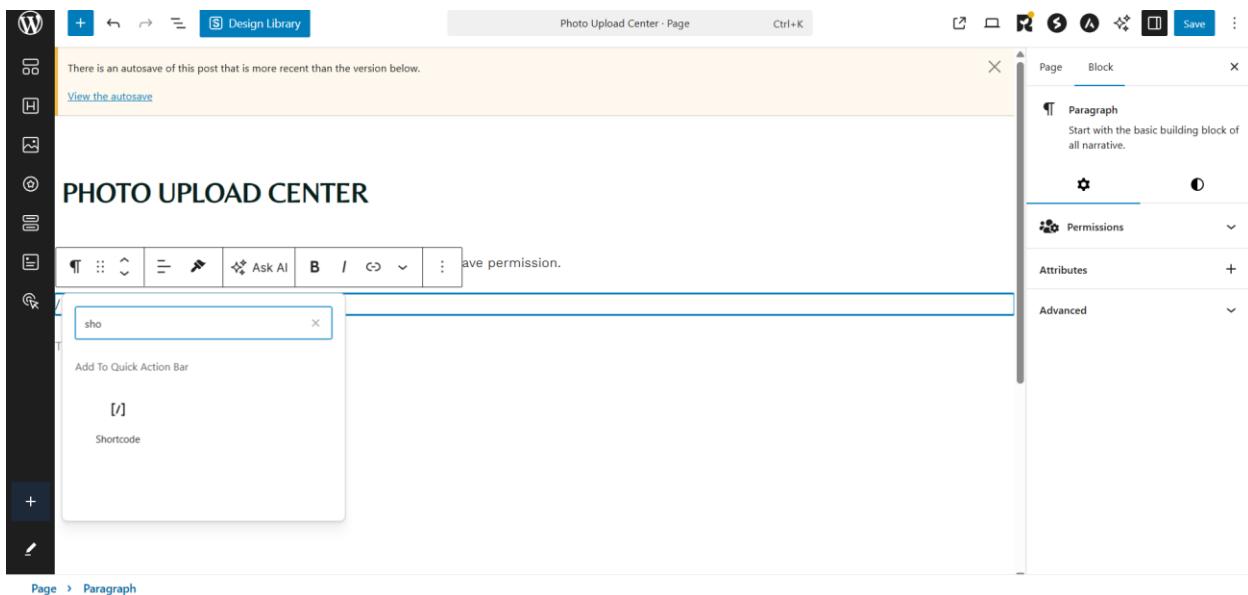


Figure 46\_shortcode block

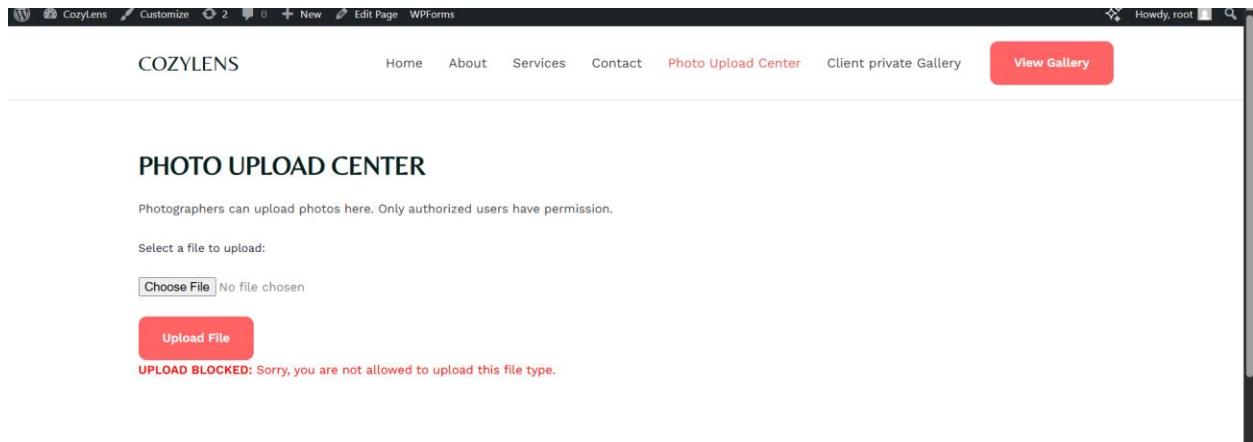


Figure 47\_testing

Now for scanning the files

Install this plugin:

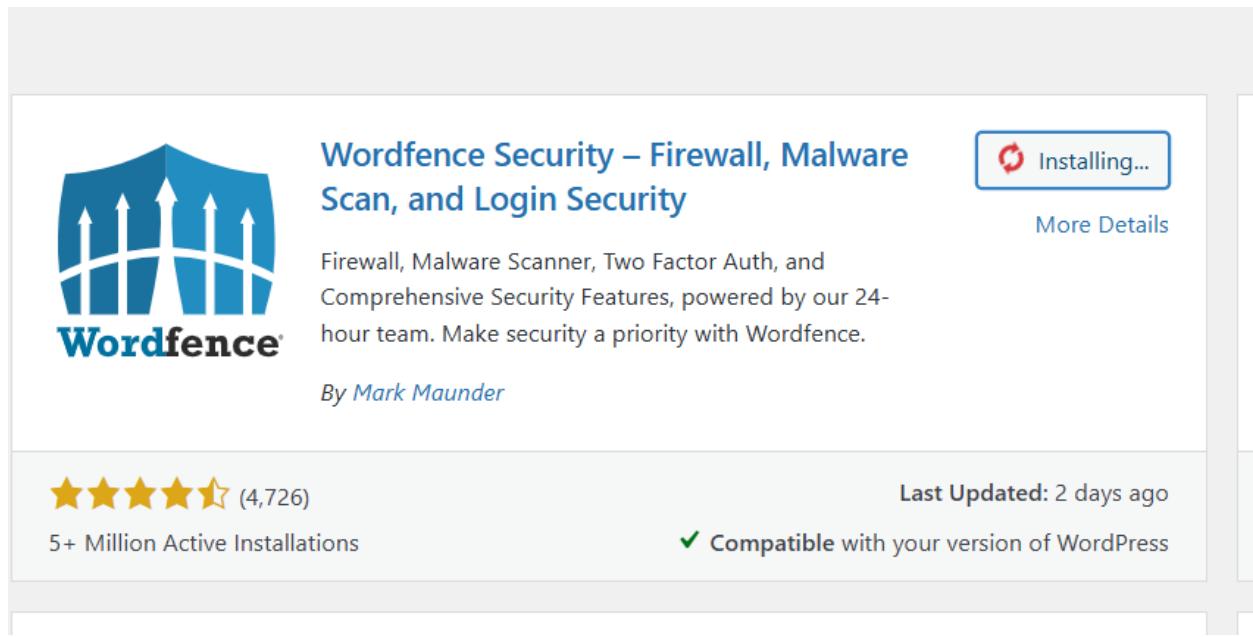


Figure 48\_wordfence security installation

Appearance  
Plugins 9  
Users  
Limit Login Attempts  
Tools  
Settings  
CAPTCHA 4WP  
Members 1  
Security 1  
Wordfence  
Dashboard 2  
Firewall  
Scan  
Tools  
Audit Log  
Login Security  
All Options  
Help  
Upgrade to Premium  
Collapse Menu

Back to Dashboard RESTORE DEFAULTS CANCEL CHANGES SAVE CHANGES

<input checked="" type="checkbox"/>	Scan for misconfigured How does Wordfence get IPs ?
<input checked="" type="checkbox"/>	Scan for publicly accessible configuration, backup, or log files ?
<input checked="" type="checkbox"/>	Scan for publicly accessible quarantined files ?
<input checked="" type="checkbox"/>	Scan core files against repository versions for changes ?
<input type="checkbox"/>	Scan theme files against repository versions for changes ?
<input type="checkbox"/>	Scan plugin files against repository versions for changes ?
<input checked="" type="checkbox"/>	Scan wp-admin and wp-includes for files not bundled with WordPress ?
<input checked="" type="checkbox"/>	Scan for signatures of known malicious files ?
<input checked="" type="checkbox"/>	Scan file contents for backdoors, trojans and suspicious code ?
<input checked="" type="checkbox"/>	Scan file contents for malicious URLs ?
<input checked="" type="checkbox"/>	Scan posts for known dangerous URLs and suspicious content ?
<input checked="" type="checkbox"/>	Scan comments for known dangerous URLs and suspicious content ?

Figure 49\_check necessary checkboxes

## STEP 7 Error Handling & Logging

WordPress already logs:

- PHP errors
- Login failures
- File upload failures
- Permission errors

But we are going to enhance it:

Enable WordPress logging

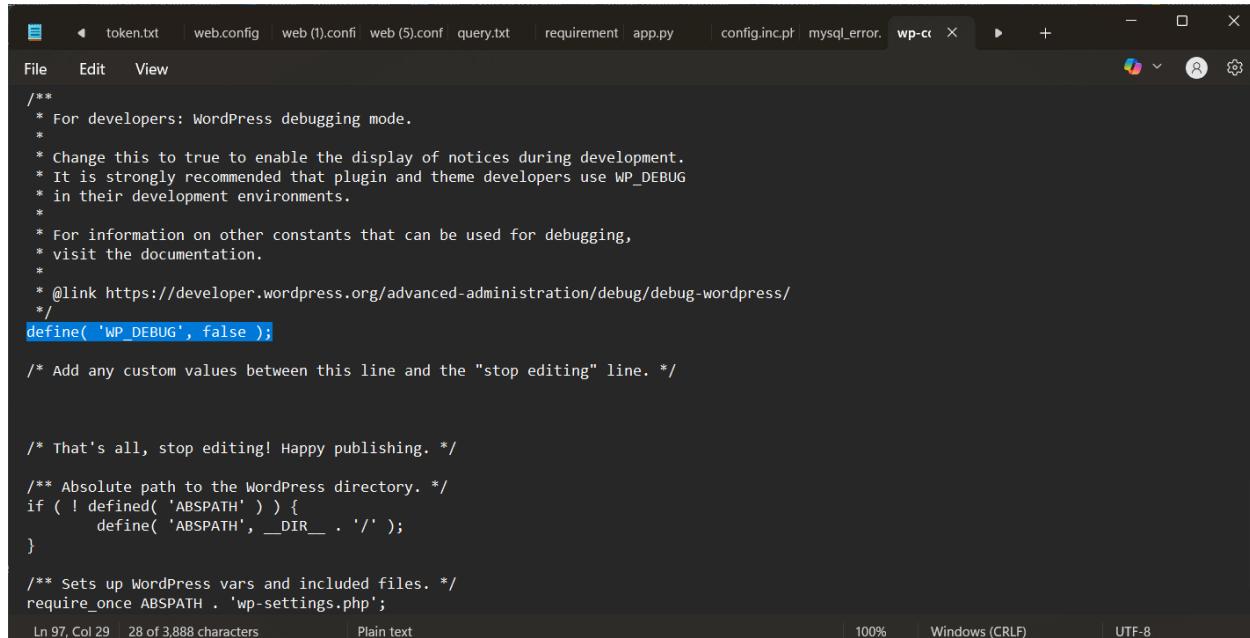
Open **wp-config.php**

Find: `define('WP_DEBUG', false);`

Replace with:

```
define('WP_DEBUG', true);
define('WP_DEBUG_LOG', true);
define('WP_DEBUG_DISPLAY', false);
```

Now errors go to: /wp-content/debug.log



```
File Edit View

/**
 * For developers: WordPress debugging mode.
 *
 * Change this to true to enable the display of notices during development.
 * It is strongly recommended that plugin and theme developers use WP_DEBUG
 * in their development environments.
 *
 * For information on other constants that can be used for debugging,
 * visit the documentation.
 *
 * @link https://developer.wordpress.org/advanced-administration/debug/debug-wordpress/
 */
define( 'WP_DEBUG', false );

/* Add any custom values between this line and the "stop editing" line. */

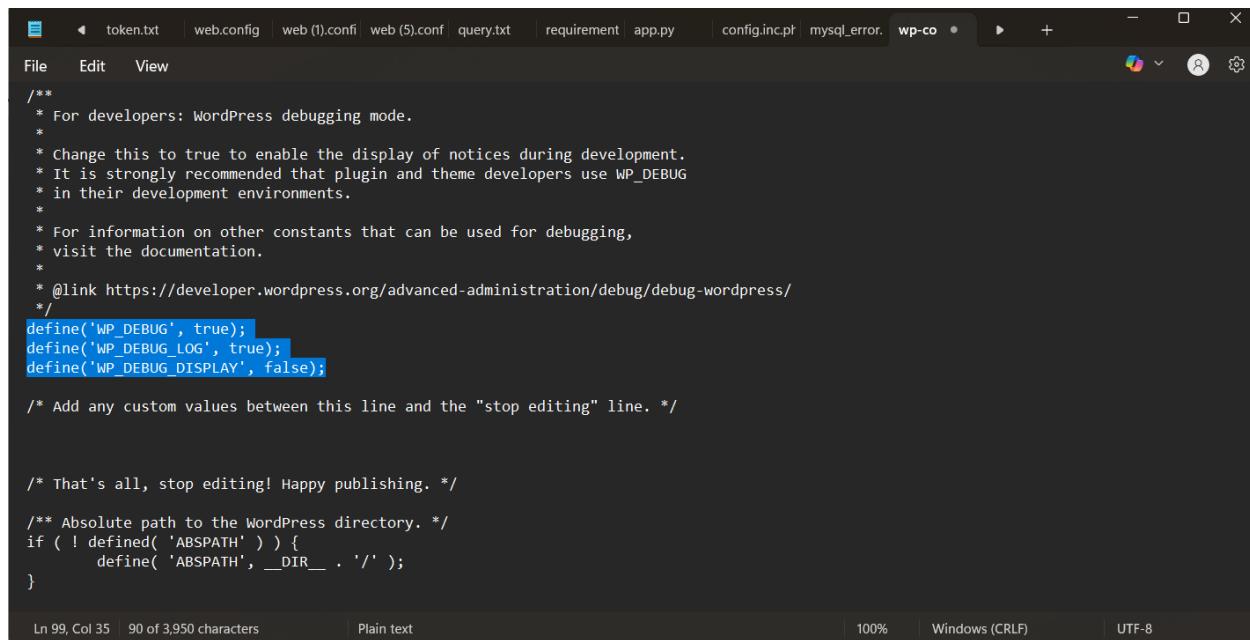
/* That's all, stop editing! Happy publishing. */

/** Absolute path to the WordPress directory. */
if ( ! defined( 'ABSPATH' ) ) {
    define( 'ABSPATH', __DIR__ . '/' );
}

/** Sets up WordPress vars and included files. */
require_onceABSPATH . 'wp-settings.php';

Ln 97, Col 29  28 of 3,888 characters | Plain text 100% Windows (CRLF) UTF-8
```

Figure 50\_find



```
File Edit View

/**
 * For developers: WordPress debugging mode.
 *
 * Change this to true to enable the display of notices during development.
 * It is strongly recommended that plugin and theme developers use WP_DEBUG
 * in their development environments.
 *
 * For information on other constants that can be used for debugging,
 * visit the documentation.
 *
 * @link https://developer.wordpress.org/advanced-administration/debug/debug-wordpress/
 */
define('WP_DEBUG', true);
define('WP_DEBUG_LOG', true);
define('WP_DEBUG_DISPLAY', false);

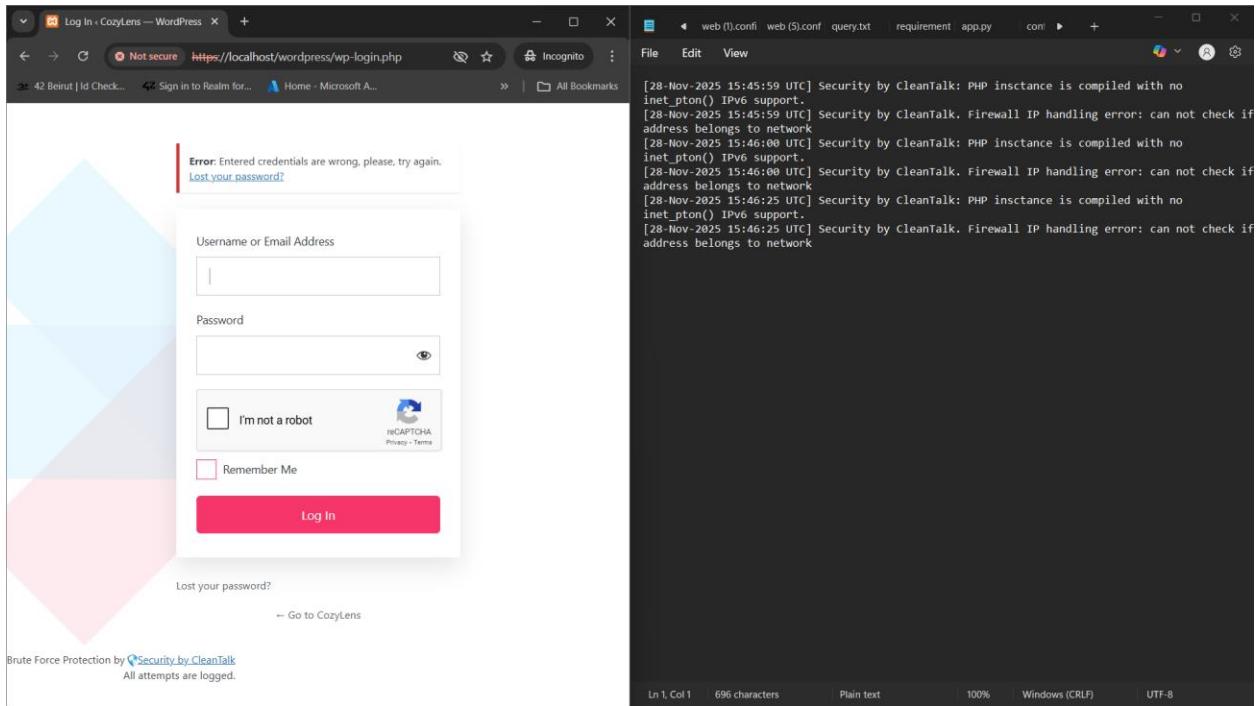
/* Add any custom values between this line and the "stop editing" line. */

/* That's all, stop editing! Happy publishing. */

/** Absolute path to the WordPress directory. */
if ( ! defined( 'ABSPATH' ) ) {
    define( 'ABSPATH', __DIR__ . '/' );
}

Ln 99, Col 35  90 of 3,950 characters | Plain text 100% Windows (CRLF) UTF-8
```

Figure 51\_replace



*Figure 52\_testing*

## WP Activity Log plugin

Already captures:

- ✓ logins
- ✓ failed logins
- ✓ role changes
- ✓ file uploads
- ✓ access to restricted pages

Using WP Activity Log, every login, logout, failed login, and role assignment is recorded

ID	Severity	Date	User	IP	Object	Event Type	Message
2101	<span style="color: orange;">i</span>	November 27, 2025 8:19:13.000 pm	root Administrator	z:1	Post	Viewed	Viewed the post Contact. Post ID: 301 Post type: page Post status: publish <a href="#">URL</a> <a href="#">View the post in editor</a>
2101	<span style="color: orange;">i</span>	November 27, 2025 8:15:30.000 pm	root Administrator	z:1	Post	Viewed	Viewed the post Home. Post ID: 296 Post type: page Post status: publish <a href="#">URL</a> <a href="#">View the post in editor</a>
2101	<span style="color: orange;">i</span>	November 27, 2025 12:30:15.000 pm	client1 Client	z:1	Post	Viewed	Viewed the post Client private Gallery. Post ID: 547 Post type: page Post status: publish <a href="#">URL</a> <a href="#">View the post in editor</a>
1000	<span style="color: green;">!</span>	November 27, 2025 12:30:13.000 pm	client1 Client	z:1	User	Login	User logged in.
1003	<span style="color: green;">!</span>	November 27, 2025 12:29:48.000 pm	Unknown User	z:1	System	Failed Login	Failed login attempt with the username lmdke - user does not exist.
2101	<span style="color: orange;">i</span>	November 27, 2025 12:26:04.000 pm	client1 Client	z:1	Post	Viewed	Viewed the post Client private Gallery. Post ID: 547 Post type: page

Figure 53\_wp login

## Step 8 Penetration Testing

Download Owasp Zap

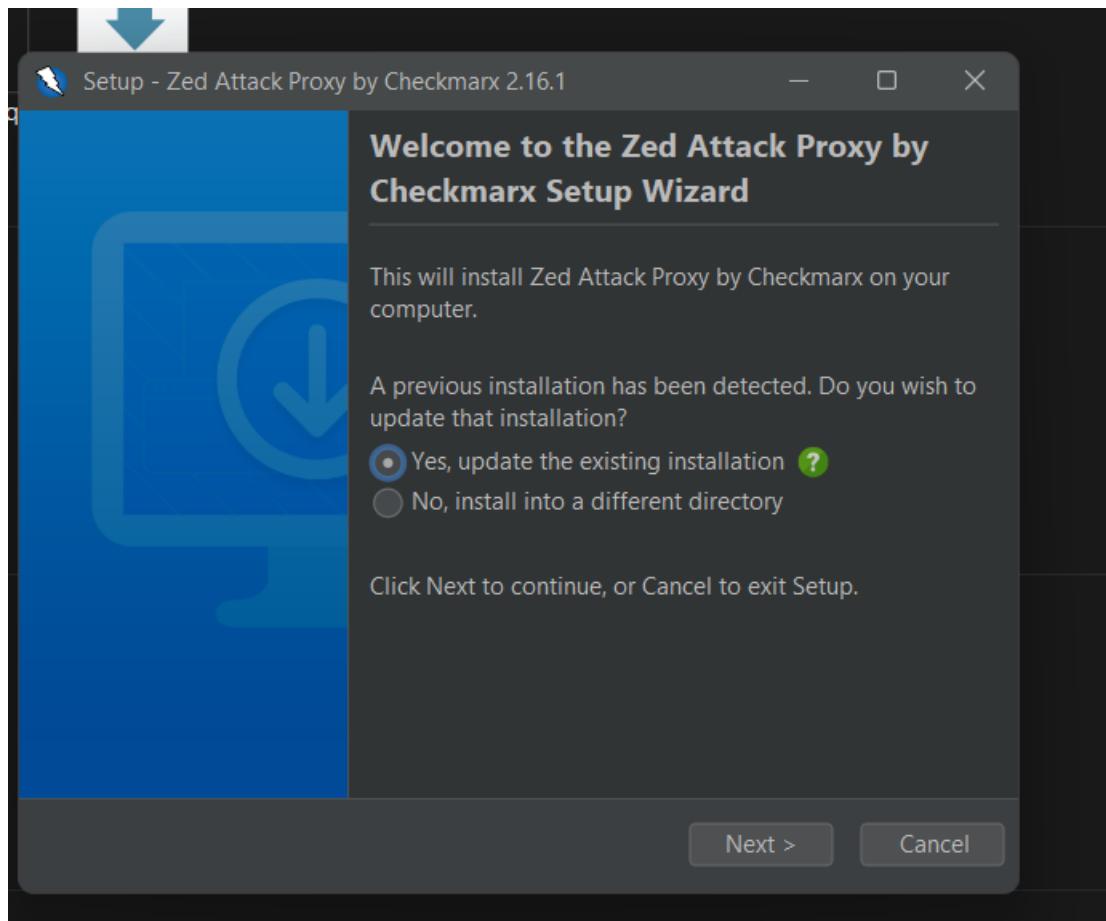


Figure 54\_downloading ZAP



Figure 55\_opening zap

Click on quickstart and automated scan

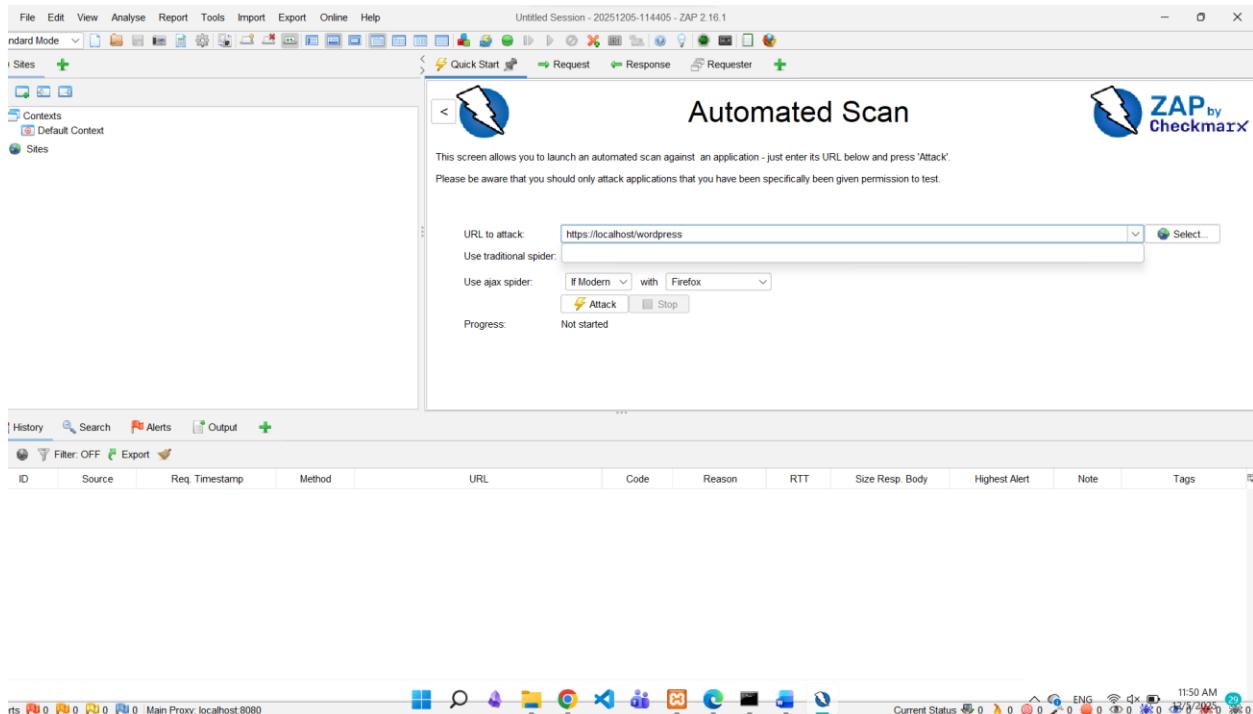


Figure 56\_ automated scan

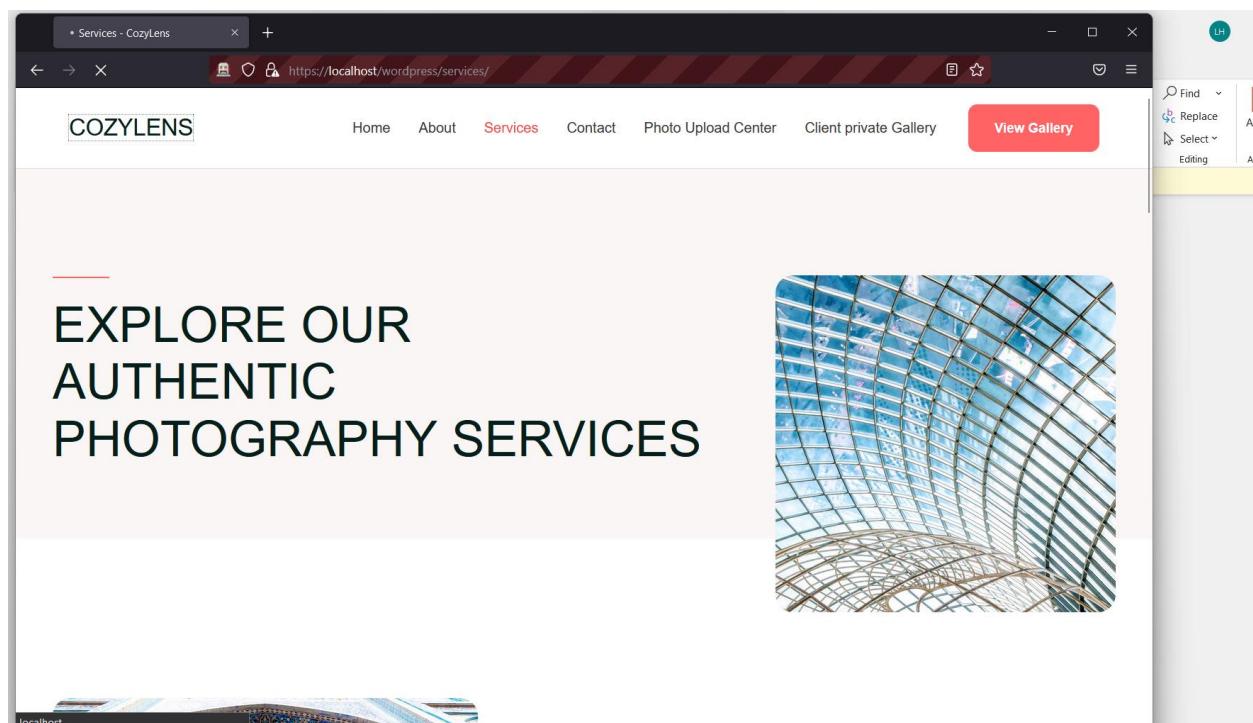


Figure 57\_testing in process

Figure 58\_attack in progress

The scanner crawled the entire WordPress installation.

Figure 59\_alerts

The most important security findings were:

## **1. Missing CSRF tokens**

Attackers could potentially submit forms on behalf of authenticated users → CSRF attack.

A CSRF token (Cross-Site Request Forgery token) is a unique, secret, random code generated by a web application and sent to a user's browser, which must be included in subsequent requests for sensitive actions (like changing passwords or making payments) to prove the request is legitimate and not from a malicious third-party site, thus preventing attackers from tricking users into performing actions they didn't intend. It works by matching the token sent with the request against a token stored server-side for that user's session, ensuring the request originates from the user.

## **2. Directory browsing & hidden file exposure**

Files like /readme.html publicly expose version info which increases attack surface.

Solution:

- 1- ZAP cannot see the internal WordPress security check happening on the server. It only reads HTML

So its logic is:

“I don't see a CSRF token in the HTML → must be vulnerable.”

But actually:

- ✓ WordPress validates the form submission
- ✓ SureForms uses WordPress nonces
- ✓ CSRF attacks cannot successfully submit the form

This is why it's called a **false positive**.

- 2- Directory browsing means someone can visit:

<http://localhost/wordpress/wp-content/uploads/>

**Index of /wordpress/wp-content/uploads/**

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
<a href="#">Parent Directory</a>	-	-	
<a href="#">2023/</a>	2025-11-28 23:16	-	
<a href="#">2024/</a>	2025-12-05 14:49	-	
<a href="#">2025/</a>	2025-12-05 15:00	-	
<a href="#">rssl/</a>	2025-12-05 14:49	-	
<a href="#">spbc_fswatcher/</a>	2025-11-28 23:16	-	
<a href="#">superank/</a>	2025-12-05 15:01	-	
<a href="#">uag-plugin/</a>	2025-12-05 14:49	-	
<a href="#">wpforms/</a>	2025-12-05 16:26	-	

*Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.0.30 Server at localhost Port 80*

3-  
Figure 60\_files

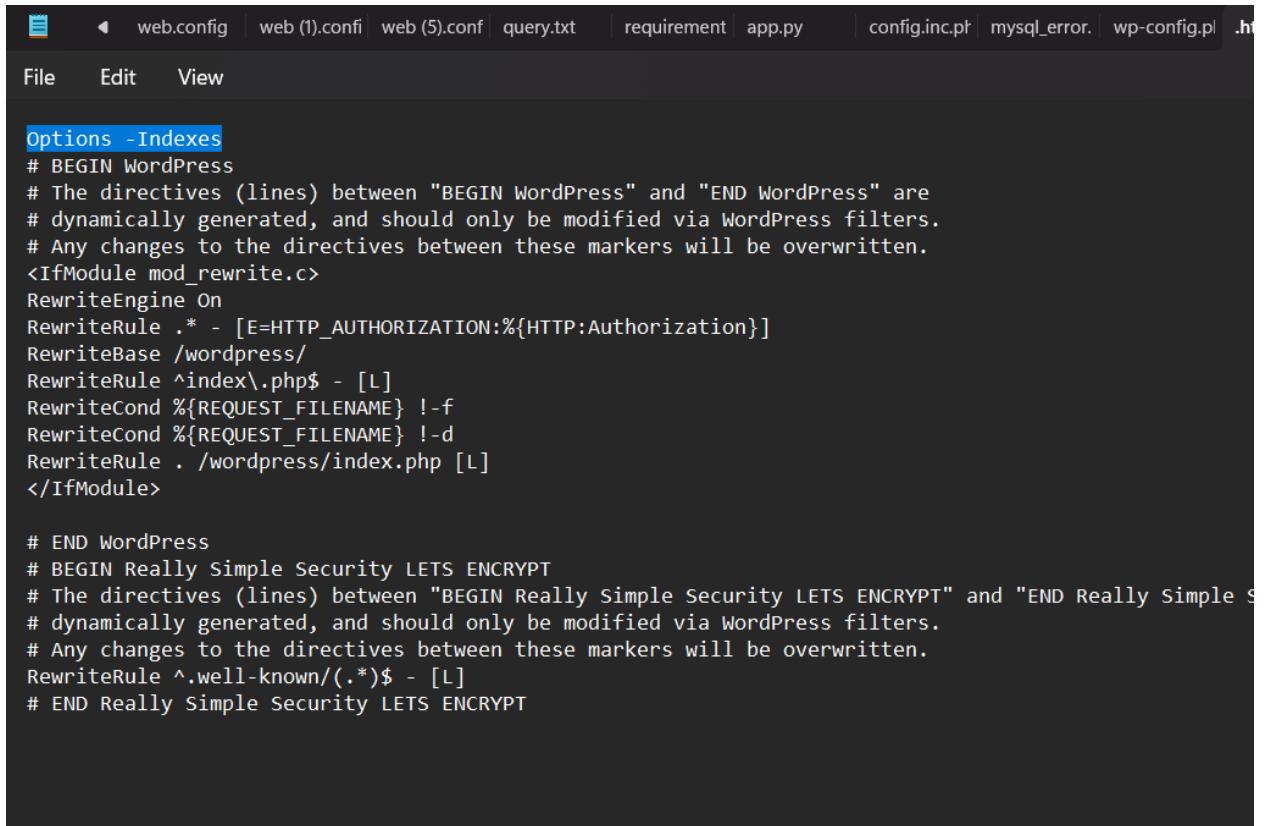
and see all files.  
This is a security risk.

Open:

xampp/htdocs/wordpress/.htaccess

Add this line **at the very top or bottom:**

**Options -Indexes**



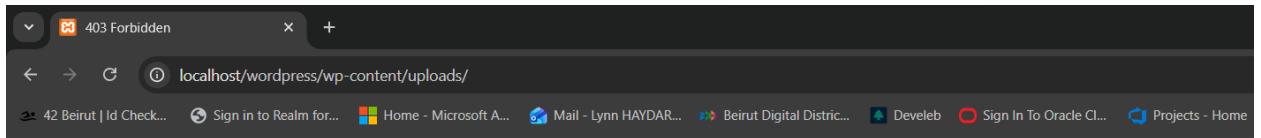
The screenshot shows a terminal window with a dark background and light-colored text. At the top, there's a navigation bar with icons and file names: web.config, web (1).confi, web (5).conf, query.txt, requirement, app.py, config.inc.php, mysql\_error, wp-config.php, and .htaccess. Below the navigation bar is a menu bar with File, Edit, and View. The main area contains Apache configuration code:

```
Options -Indexes
# BEGIN WordPress
# The directives (lines) between "BEGIN WordPress" and "END WordPress" are
# dynamically generated, and should only be modified via WordPress filters.
# Any changes to the directives between these markers will be overwritten.
<IfModule mod_rewrite.c>
RewriteEngine On
RewriteRule .* - [E=HTTP_AUTHORIZATION:{HTTP:Authorization}]
RewriteBase /wordpress/
RewriteRule ^index\.php$ - [L]
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule . /wordpress/index.php [L]
</IfModule>

# END WordPress
# BEGIN Really Simple Security LETS ENCRYPT
# The directives (lines) between "BEGIN Really Simple Security LETS ENCRYPT" and "END Really Simple Security LETS ENCRYPT" are
# dynamically generated, and should only be modified via WordPress filters.
# Any changes to the directives between these markers will be overwritten.
RewriteRule ^.well-known/(.*)$ - [L]
# END Really Simple Security LETS ENCRYPT
```

Figure 61\_adding

It tells Apache to **deny listing all directories**, even if a folder has no index.php.



*Figure 62\_it worked*

But it created another vulnerability

## Table of Figures

Figure 1_starting services .....	3
Figure 2_creating database.....	4
Figure 3_astra theme .....	4
Figure 4_pages .....	5
Figure 5_plugins installed .....	6
Figure 6_limit login attempts .....	6
Figure 7_locked after failed attempts .....	7
Figure 8_Captcha 4wp .....	8
Figure 9_creating new key .....	8
Figure 10_Integration keys.....	9
Figure 11_checking results.....	9
Figure 12_checking .....	10
Figure 13_downloading plugin .....	10
Figure 14_check necessary settings.....	11
Figure 15_123 .....	11
Figure 16_strong .....	12
Figure 17_add role .....	12
Figure 18_photographer role.....	13
Figure 19_subscriber.....	13
Figure 20_adding new user with role photographer .....	14
Figure 21_adding new user with role client .....	15
Figure 22_users created.....	16
Figure 23_access roles client private gallery.....	16
Figure 24_roles photo upload center .....	17
Figure 25_functions.php .....	17
Figure 26_redirected to login page.....	19
Figure 27_log in successful .....	19
Figure 28_redirected to login.....	20
Figure 29_log in successful .....	20
Figure 30_installing plugin .....	21
Figure 31_ssl works .....	22
Figure 32_https .....	22
Figure 33_proceding .....	22
Figure 34_testing .....	23
Figure 35_test successful .....	23
Figure 36_installing plugin .....	24

Figure 37_getting access key.....	24
Figure 38_pasting access key .....	25
Figure 39_session expired after one minute .....	25
Figure 40_file uploads plugin.....	26
Figure 41_allowed by WordPress.....	27
Figure 42_adding custom file type.....	27
Figure 43_configuring it .....	27
Figure 44_save .....	28
Figure 45_done .....	28
Figure 46_shortcode block.....	29
Figure 47_testing .....	30
Figure 48_wordfence security installation.....	30
Figure 49_check necessary checkboxes.....	31
Figure 50_find .....	32
Figure 51_replace.....	32
Figure 52_testing .....	33
Figure 53_wp login.....	34
Figure 54_downloading ZAP .....	35
Figure 55_opening zap.....	36
Figure 56_automated scan .....	37
Figure 57_testing in process .....	37
Figure 58_attack in progress .....	38
Figure 59_alerts .....	38
Figure 60_files .....	40
Figure 61_adding .....	41
Figure 62_it worked .....	42