20-point *WordPress* *Security* Audit

□ Acquire SSL certificates and implement HTTPS: Obtain SSL certificates from trusted providers and ensure HTTPS is enforced throughout your website.

□ Update WordPress, plugins, and themes: Regularly check for updates to the core, plugins, and themes, and apply them as soon as possible to patch security vulnerabilities.

□ Enforce strong, unique passwords and password policies: Implement password policies, such as length, complexity, and expiration, to ensure users maintain secure credentials.

□ Enable two-factor authentication (2FA) using authenticator apps: Require 2FA for added security by integrating it within your site or using built-in WordPress functions.

□ Disable file editing through the WordPress admin panel: Add define(**'DISALLOW_FILE_EDIT', true**); to your wp-config.php file to disable the built-in file editor.

□ Disable plugin installation. By simply adding define(**'DISALLOW_FILE_MODS', true**); to your wp-config.php you will have a full control over plugin installation and update process.

□ Set correct file permissions and ownership: Ensure files have permissions set to 644 and directories to 755, with ownership assigned to the appropriate user and group.

□ Implement IP-based access control to the admin area: Configure your server to allow only specific IP addresses to access the WordPress admin area or at least change the wp-admin url.
You can do it for example by using WPS hide-login plugin

□ Disable XML-RPC if not required: If not using XML-RPC for applications like Jetpack or remote publishing, disable it to reduce potential attack vectors.

□ Configure Content Security Policy (CSP) headers: Set up CSP headers in your server configuration to prevent cross-site scripting (XSS) and other code injection attacks.

□ Audit installed plugins and themes for known vulnerabilities: Manually review source code and update history of installed plugins and themes to identify potential vulnerabilities.
You can also check WPScan

□ Change the default database table prefix during installation: During WordPress installation, select a custom database table prefix to make it more difficult for attackers to target your database.

□ Disable PHP error reporting and display: In your server's php.ini file, **set display_errors** and **display_startup_errors** to **„Off"** to prevent sensitive information exposure.

□ Use a strong, unique password for your database user:
Create a separate database user with a strong and unique password that follows best practices for password security. Also limit remote access for database connection.

☐ Adhere to secure coding principles: Follow the OWASP Secure Coding Practices and WordPress coding standards to minimize the risk of introducing vulnerabilities. You can find more details under link: owasp.org. There are also tools which will help you audit your website.

☐ Review and manage user accounts and roles: Ensure users have the least privileges necessary and revoke elevated permissions from users who no longer need them.

☐ Delete installation and upgrade files: Remove files like install.php or readme.html that may provide attackers with information about your WordPress installation.

☐ Disable directory browsing in your server configuration: Add “Options -Indexes” to your .htaccess file to prevent unauthorized users from viewing your site’s directory structure.

☐ Implement input validation and sanitization: Use WordPress functions like **esc_attr(), sanitize_text_field(),** and **wp_kses()** to validate and sanitize user input.

☐ Set up intrusion detection and monitoring: Monitor your website’s logs and use built-in server tools to detect unauthorized changes to files and potential breaches.

If you need support with your website, we’ve got you covered!
Contact us at office@osomstudio.com to schedule a free consultation.