WordPress and GitHub Security Policy

This policy sets minimum security rules for any developer, contractor, or administrator who works on a WordPress website and uses GitHub or other code repositories as part of that work. GitHub-related risk for WordPress sites most often comes from stolen credentials, secrets committed to repositories, and unreviewed third-party code or tools introduced into production.

1. Account security

All GitHub accounts used for company or client work must use strong, unique passwords and two-factor authentication.

All WordPress administrator and editor accounts must use strong, unique passwords and two-factor authentication wherever available.

Shared logins are not allowed for GitHub, WordPress admin, hosting panels, SFTP, SSH, database consoles, or plugin licenses.

Access must be limited to the minimum needed for the role, and access must be removed promptly when a project or contract ends.

2. Secrets and credentials

Production secrets must never be stored in Git repositories, including private repositories.

Prohibited items in Git include WordPress admin credentials, database passwords, API keys, license keys, SSH keys, SFTP credentials, SMTP passwords, and full production configuration files containing secrets.

Secrets must be stored in approved environment files, secret managers, or hosting configuration tools outside the repository.

If a secret is exposed in a repository, it must be treated as compromised and rotated immediately.

3. Code sourcing and review

No plugin, theme, snippet, deployment script, or “helper tool” from GitHub may be used in production without review by an authorized maintainer.

Public GitHub code must be checked for reputation, maintenance history, issue activity, license status, and obvious security concerns before adoption.

Proof-of-concept exploit code, “nulled” themes, cracked plugins, and unofficial admin tools are prohibited on company or client systems.

Custom code changes must be reviewed before deployment to production whenever practical.

4. WordPress hardening rules

WordPress core, themes, plugins, and server software must be kept updated on a defined schedule, with urgent security patches prioritized.

Unused plugins, themes, and dormant administrator accounts must be removed.

Only approved plugins and themes may be installed, and each must have a business owner or technical maintainer.

File permissions, backups, HTTPS, and logging must be configured according to WordPress hardening guidance and hosting best practices.

5. Deployment and change control

Production changes must be made through approved deployment workflows, not ad hoc edits from unknown machines.

Direct production access must be restricted to authorized personnel and used only when necessary.

Before major updates, a current backup must exist and restoration must be feasible.

Security plugins, WAF rules, and monitoring must not be disabled without approval.

6. Incident response

Any suspected credential theft, malware, phishing, exposed repository, suspicious plugin, or unusual WordPress login activity must be reported immediately.

After a suspected exposure, required actions include password resets, token rotation, session invalidation, and review of recent repository and admin activity.

Systems used to download unknown GitHub tools related to WordPress must be treated as potentially compromised until checked.

7. Minimum enforcement checklist

GitHub 2FA enabled for every developer.

WordPress 2FA enabled for admins and editors where supported.

No secrets in repositories.

Only approved plugins and themes in production.

Core, themes, and plugins updated regularly.

Backups verified and restorable.

Suspicious tools and phishing-style “developer utilities” banned.

This policy is intended to reduce the most common WordPress and GitHub security failures: credential theft, supply-chain compromise, unreviewed code, and weak administrative controls.