SAMPLE AUDIT REPORT

Sample WordPress Security Audit Report

This sanitized, representative sample shows what an audit client receives: findings written for action, evidence another technical team can follow, positive controls acknowledged, and a retest status that tracks each issue to closure. Details are illustrative composites — no single client environment is described.

Prepared by G. Schad, WordPress and Linux security specialist

Published · Last reviewed

FORMAT Executive summary, findings, positive controls, retest record	EVIDENCE Sanitized request/response and configuration extracts
AUDIENCE Owner, developer, host, and administrator — ownership stated per finding	RETESTING Agreed fixes verified and the finding status updated

---

STRUCTURE

What the delivered report contains

The report is written so a non-technical owner can follow the risk while the responsible engineer can follow the fix.

- ✓ Executive summary: overall posture, the findings that matter most, and what to do first
- ✓ Scope statement: systems reviewed, access used, dates, and explicit exclusions
- ✓ Prioritized findings with severity, affected component, and sanitized evidence
- ✓ Business impact and realistic attack conditions for each finding — not raw scanner text

- ✔ Specific remediation guidance with the responsible party named
- ✔ Positive controls: what is already working and should not be changed
- ✔ Retest record: each agreed fix verified, with its final status

---

ANATOMY OF A FINDING

Every finding answers six questions

01 **Severity**

Rated by realistic exploitability and business impact in this environment — not copied from a generic CVSS feed.

02 **Component**

The exact affected element: an endpoint, plugin, account, configuration file, or server control.

03 **Evidence**

A sanitized request/response sequence, configuration extract, or screenshot that another team can reproduce.

04 **Business impact**

What the weakness allows in practice: account takeover, data exposure, downtime, or fraud — in plain language.

05 **Remediation**

The specific change, who owns it (developer, host, administrator), and any operational cautions.

06 **Retest status**

Open, resolved and verified, or risk accepted — updated after the agreed fixes are retested.

SAMPLE FINDING · WP-04

Administrative endpoint exposed without adequate access controls

Severity: High. Component: wp-admin / authentication. The administrative login endpoint was reachable from any network location, with no multi-factor authentication enforced for administrator roles and no effective rate controls on authentication attempts. Sanitized request/response evidence in the full report shows the endpoint accepting unlimited attempts without lockout or alerting.

Business impact: the configuration materially increases both the probability and the impact of credential attacks. A single phished, reused, or brute-forced administrator password becomes full site compromise, with no second factor and no alert to the owner while an attack runs.

Remediation: restrict administrative access by network location where operationally possible, enforce phishing-resistant multi-factor authentication for all administrator accounts, and add rate controls and failed-login alerting. Ownership: site administrator, with host support for network restrictions. Retest status: pending.

> **WHY IT IS WRITTEN THIS WAY**
>
> The same weakness in a scanner export reads “Login page found”. The report version argues severity from the actual configuration, states what an attacker gains, and names who fixes what — that difference is the deliverable.

MORE SAMPLE FINDINGS

Representative entries from the findings table

ID	SEVERITY	FINDING	REMEDIATION DIRECTION
WP-07	High	Abandoned plugin with a publicly known vulnerability remained installed and active.	Remove or replace the component; verify no residual data or scheduled tasks remain.
WP-09	Medium	Database backup archive stored inside the public webroot and reachable by direct URL.	Move backups outside the webroot, restrict access, and rotate the exposed database credentials.
WP-12	Medium	REST API exposed the full user list, enabling administrator username enumeration.	Restrict the users endpoint for unauthenticated requests and monitor authentication attempts.
SRV-02	Medium	PHP errors displayed full file paths and configuration detail to visitors.	Disable display_errors in production; log errors server-side with restricted access.
WP-15	Low	WordPress and server version details disclosed in headers and generator tags.	Reduce version disclosure; treat as defense-in-depth, not a primary control.

POSITIVE CONTROLS

The report also records what is already right

Listing working controls prevents a future “fix” from breaking them, and gives the owner a fair picture rather than a fear document.

- ✓ Automatic core security updates enabled and applying correctly
- ✓ TLS configuration current, with HTTP correctly redirected
- ✓ Database not remotely reachable from the public internet
- ✓ Off-site backups running on schedule — restoration test recommended and scoped

separately

RETEST RECORD

Findings are tracked to closure, not just reported

> **HONEST STATUS BEATS A CLEAN SHEET**
> A report where every line says “resolved” the day it is delivered was not retested. Open items and accepted risks are recorded as such — that record is what an insurer, client, or future responder can actually rely on.

FINDING	AGREED ACTION	RETEST RESULT
WP-04 · Admin endpoint exposure	MFA enforced, rate controls and alerting added	Resolved — verified on retest
WP-07 · Abandoned vulnerable plugin	Component removed and replaced	Resolved — verified on retest
WP-09 · Backup in webroot	Backups relocated, credentials rotated	Resolved — verified on retest
WP-12 · User enumeration	Endpoint restricted	Open — fix scheduled by client team
WP-15 · Version disclosure	Accepted as residual risk by the owner	Risk accepted — documented

AUDIT OR INCIDENT REPORT?

This sample shows prevention — the ClickFix report shows response

A security audit report like this sample documents weaknesses found before an incident, with remediation and retesting. An incident report documents an active compromise:

evidence preservation, persistence mapping, containment, eradication, and proof of recovery.

Both deliverables share the same evidence standard. To see the incident side documented at full depth across a 30-site infection, read the ClickFix report.

Read the ClickFix incident report →

See what the audit engagement includes →

COMMON QUESTIONS

Common questions about the report

Is this a real client report? −

It is a sanitized, representative sample: the structure, severity reasoning, and retest tracking match a delivered report, while the specific findings are illustrative composites so no client environment is identifiable.

Will my report look exactly like this? +

Do I receive anything besides the document? +

Can the findings be fixed and retested afterwards? +