Pentest report formats. What a useful report contains.

A useful penetration test report has six sections: executive summary (10-minute read for non-technical leadership), scope and methodology (what was tested, how, and what was not), findings (one entry per weakness with title, severity, evidence, impact, and remediation), remediation roadmap (what to fix and in what order), retest results (what was verified after the customer fixed things), and appendices (raw evidence, payloads, references). A report that ranks findings only by CVSS without context, or hands the customer a scanner CSV, is not useful.

HOW IT WORKS

01 Scope and methodology: what was actually done

Auditors, customers, and procurement teams read this section first. They want evidence that the engagement was real and rigorous.

A useful section names:

- Scope. Exactly what was in scope. URLs, IPs, accounts, environments, services. Exactly what was out of scope and why.
- Approach. Black box, gray box, or white box. What information the tester started with.
- Methodology. The standard followed (OWASP WSTG, NIST 800-115, PTES, OSSTMM) and any customer-specific framework requirements.
- Activities performed. What categories of testing were done (authentication, authorisation, input validation, business logic, configuration review, and so on).
- What was not tested. Important. Time-bounded engagements have limits; naming them protects both customer and tester.

02 Findings: the technical core

Each weakness gets its own entry. A useful finding contains:

- Title. Plain language. 'Authentication bypass on admin login' not 'CVE-2023-XXXXX'.
- Severity. Critical / High / Medium / Low / Informational. CVSS score as a reference, not the only signal.

HOW TO DEFEND

- Immediate. The findings that need fixing now, regardless of cost. Usually one to five issues.
- Within the next release. The findings that need fixing in the standard development cycle.
- Within the quarter. Lower-severity issues that should be tracked but do not block other work.
- Backlog. Informational findings worth being aware of.

SOURCES

[1] NIST SP 800-115

[2] OWASP Web Security Testing Guide

[3] CVSS v3.1 Specification

[4] CWE Common Weakness Enumeration

- Affected component. The URL, endpoint, host, file, contract, or feature.
- Description. What the weakness is, in 100 to 300 words.
- Evidence. The exact request, response, payload, screenshot, or video showing the issue. Anyone with access to the report should be able to reproduce it.
- Realistic impact. What an attacker could actually do with it. Not 'attacker could read data' but 'attacker could read all customer records by enumerating IDs from 1 to N'.
- Remediation. The specific change required. Not 'use parameterized queries' but the line of code or configuration that needs to change.
- References. OWASP, CWE, CVE, vendor advisories, internal links.

Findings sorted by realistic impact, not by CVSS, are more actionable. A CVSS 7.5 that affects every customer matters more than a CVSS 9.8 that affects one admin account behind MFA.

03 Retest results: what was verified

Mature engagements include a free retest of fixes within an agreed window (often 60 to 90 days). The retest section documents:

- Which findings were verified as fixed.
- Which findings remain (with updated evidence).
- Any new findings that surfaced because of the fixes themselves (this happens more often than people expect).

A report without a retest path is incomplete. Auditors increasingly ask for it, and engineering teams need the closure.

04 Appendices: the raw material

Everything technical that does not belong in the main flow:

- Raw scanner output (curated).
- Payloads and request templates.
- Screenshots and video evidence.
- Tool versions used and configuration.

- Reference reading.

Appendices are useful when an engineer reproducing a finding needs the underlying material, or when an auditor verifying the engagement wants raw evidence.

05 What a bad pentest report looks like

Three patterns to recognise:

- Scanner CSV in a PDF. Hundreds of low-confidence findings, no curation, no business-impact context. Often signals the engagement was a vulnerability scan dressed as a pentest.
- All findings sorted by CVSS, no business context. A CVSS-only sort buries the issues that matter.
- No retest, no remediation guidance. A report that names problems without committing to verify the fix forces the customer to start over.

If the report looks like one of these, ask hard questions about what the engagement actually covered.

Get a report your engineers can act on.
securelayer7.net/learn/pentest/pentest-report-formats

Open online