PENTESTERRA
SEE THE REAL RISK

# Vulnerability Management

Automatic exploit verification, safely — proof instead of a CVSS score.

VULNERABILITIES
VERIFIED EXPLOITS
LESS NOISE
REAL SECURITY

FIND
VERIFY
ELIMINATE RISK
STAY AHEAD

**Every finding gets an actual exploit run against it before it reaches your team.**
No manual triage backlog of maybes.


## Verification Chain

Escalating from trusted sources to custom verification — automatic and safe.

CISA KEV — Known exploited vulnerabilities
→ Metasploit — Exploitation modules
→ ExploitDB — Public exploits and PoCs
→ Custom Scripts — Non-destructive verification
→ GitHub PoCs — Community proofs
→ Generated PoC — AI-assisted (last resort)

## Finding Lifecycle

Clear, immutable progression — no silent downgrades.

Unconfirmed — Initial discovery
→ Potential — Exploit match found
→ Detected — Confirmed vulnerable
→ Verified — Exploit successfully run
→ Exploited — Active risk, proof attached

## What it does

- Verification chain, escalating: CISA KEV → Metasploit modules → ExploitDB → custom non-destructive scripts → GitHub PoCs → generated PoC
- Non-destructive by default — no data deletion, no crash-causing payloads, no exfiltration; isolated scopes; optional human approval gate
- Machine status per finding that never silently downgrades
- Full audit trail: who marked what a false positive/accepted risk, when, why
- Agentless — external perimeter + internal LAN + cloud/hybrid
- Findings feed everywhere else: pentest scope, Attack Chain, DevGuard, compliance
- Verified findings auto-create tickets (Jira integration) with reproduction steps

## Why Pentesterra

- "Verified" here means a literally executed exploit, not a heuristic confidence score
- Triage does not lose history — most VM tools let status quietly roll back on re-scan, Pentesterra does not
- Automatic proof-based verification reduces noise before findings reach your team

> *Less noise. More real security.*
> — PENTESTERRA

REAL EXPLOITS
REAL CONTEXT
REAL INSIGHTS
REAL ADVANTAGE

External Perimeter · Internal LAN · Cloud / Hybrid · Attack Chain Integration · DevGuard Correlation · Compliance Reporting · Jira Ticketing · Human Approval Gate

## Prioritize what is truly exploitable.

Explore Vulnerability Management →

REAL SECURITY
FOR A MORE
RESILIENT TOMORROW

PENTESTERRA

VULNERABILITY MANAGEMENT

# Automatic Exploit Verification, Safely

Every finding gets an actual exploit run against it before it reaches your team. No manual triage backlog of maybes - proof instead of a CVSS score.

**0 Agents**
Nothing installed on the assets you scan

**In + Out**
External perimeter and internal LAN

**Reused**
Feeds pentest, attack chains, and reporting

WHY NOW

**Your last scan produced 1,200 findings, and your team has time to seriously investigate about twenty of them.** A CVSS-sorted list treats a false positive and a live RCE as the same kind of red text. The real risk gets lost in the noise until it's exploited.

## HOW IT WORKS

### Agentless, External and Internal

Nothing to install on the assets you scan. Distributed scanner nodes cover the external perimeter from outside and the internal perimeter from inside the network.

- → No endpoint agents, no host software to deploy or maintain
- → External perimeter, internal LAN, cloud, and hybrid topologies
- → Distributed scanner nodes, including on-prem and air-gapped
- → Authenticated and unauthenticated scan modes

### Automatic Exploit Verification, Safely

The killer feature: every finding can be verified automatically by actually running an exploit against it. Non-destructive by default, with staging and production kept as isolated scopes.

- → Verification chain: CISA KEV → Metasploit → ExploitDB → custom scripts → GitHub PoCs → generated PoC
- → No crashes, no data loss, no data exfiltration
- → Optional human-in-the-loop approval before active exploitation
- → Handles WAF, non-standard ports, and masked banners - every result has a PoC

### Triage That Holds Up

Each vulnerability is a record with a machine status - unconfirmed, potential, detected, verified, exploited - that never silently downgrades.

- → Per-finding status with full history and delta between scans
- → Exclusion log: who, when, and why, revocable by any reviewer
- → Verified findings auto-create Jira tickets with reproduction steps

### The Data Is Reused Everywhere

Scan results are not a dead-end report. They set pentest scope, feed the attack chain engine, and carry business-impact context into reporting.

- → Feeds pentest scope, likely tech, and attack paths
- → Becomes nodes in cross-domain attack chains
- → Correlates with DevGuard code and supply chain findings

## FREQUENTLY ASKED QUESTIONS

### How is this different from a plain vulnerability scanner?

A plain scanner produces a list ranked by CVSS. Pentesterra can automatically verify each finding by running a real exploit against it, keeps a stable triage status per finding, and reuses the results in the pentest, attack chain, and reporting modules instead of leaving them in a silo.

### Is automatic exploit verification safe to run on production?

Yes. Verification is non-destructive by default - no data deletion, no denial-of-service payloads, and nothing that reads or exfiltrates confidential data. Each scan has an explicitly defined scope, staging and production are isolated, and a human-in-the-loop mode can require analyst approval before any active exploitation step.

### How does vulnerability prioritization work?

Prioritization is risk-based rather than severity-based. A finding is ranked by whether an exploit actually worked against it, what the affected asset does for the business, and whether it opens a path in a wider attack chain - not by CVSS alone.

### Where do the scan results go?

Into every other module. They set penetration testing scope, become nodes in attack chains, correlate with DevGuard code and supply chain findings, and feed compliance and business-impact reporting.

**Cut through the noise - see what's really exploitable**

Agentless scanning, verified findings, no manual triage backlog.

**info@pentesterra.com**
pentesterra.com/vulnerability-management