

[SOLUTION BRIEF]

Penetration testing on autopilot.

Autonomous agents that pentest every release and prove every finding with a working exploit.

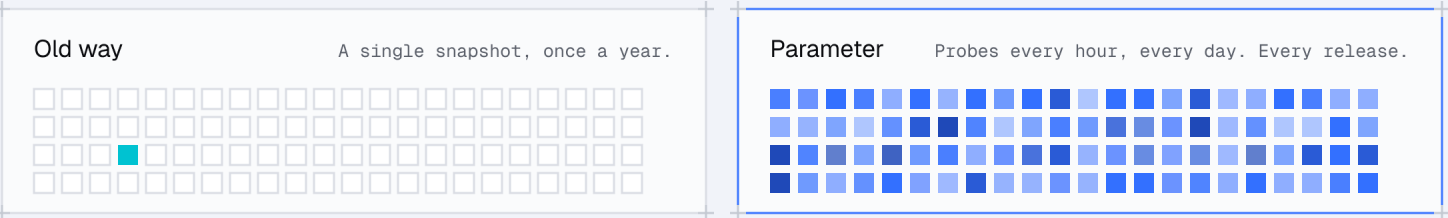
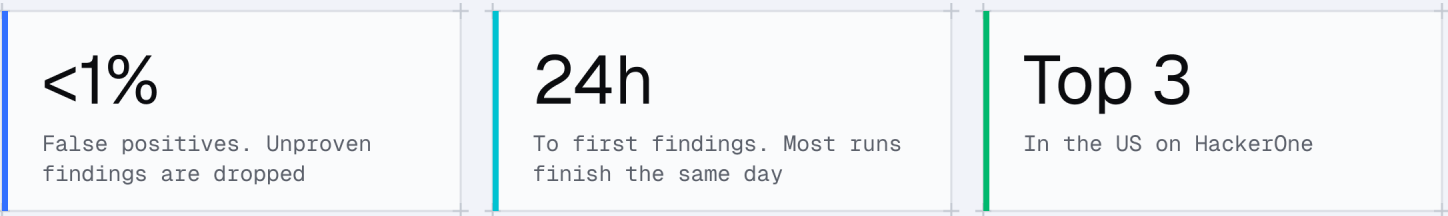
[THE PROBLEM]

A traditional pentest is a person, booked weeks out, testing a single snapshot of your app. By the time the report lands, the code has moved on. Automated scanners run more often but cannot reason about business logic, tenant boundaries or chained attack paths, and their unvalidated output is expensive to triage. Teams end up choosing between depth once a year and noise every day.

WHAT SECURITY TEAMS NEED

- 01 **Adversary-grade depth**
Testing that works real attack paths across auth, business logic and APIs, not a checklist.
- 02 **A proof for every finding**
Reproduction steps and a working exploit, so nothing unproven reaches engineering.
- 03 **Every release, same day**
Results in hours, on demand and on every deploy, with re-testing when a fix ships.

[BY THE NUMBERS]



BUILT BY THE TEAM THAT SECURED

Apple · Google · Microsoft · U.S. Department of Defense · T-Mobile · Henkel · Epic Games · AT&T · Yahoo

BACKED BY Y COMBINATOR

[HOW IT WORKS]

Probe. Exploit. Verify. Every release.

01

Probe

Agents map your real attack surface, every endpoint, parameter and auth flow, from your code, your OpenAPI specs, or just the live app.

02

Exploit

Hundreds of agents work real attack paths in parallel. They try to break expected behavior the way an attacker would, not run down a checklist.

03

Verify

Every issue is reproduced from a clean state. Unproven findings are dropped; what is left ships with impact, repro steps and a fix.

[SOLUTION ARCHITECTURE]

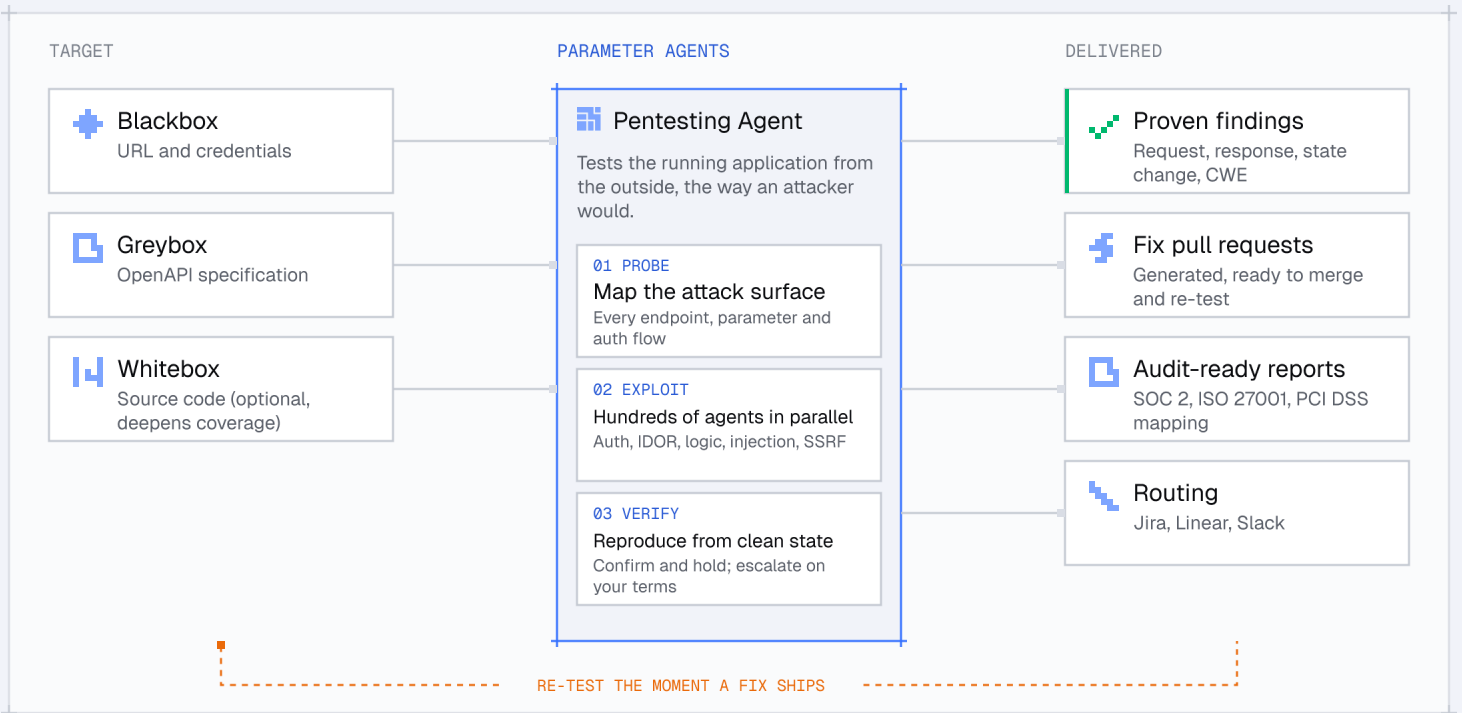


FIG. 01 Pentesting Agent flow. Targets are mapped from source, spec or the live application; parallel agents exploit; a verification stage reproduces each issue from a clean state before it is filed.

Tests from the outside, like an attacker

The agents test your running application externally. A URL and credentials are enough to start; source access is optional and only deepens coverage.

Pauses on real risk

When an agent finds something exploitable it stops and shows you the full attack analysis before going deeper. Escalate on your terms, tracked on the same finding.

Finding to fix, fully automatic

High-confidence pull requests are generated, ready to merge. Re-test the moment you ship a fix, and watch the agents work live.

[KEY CAPABILITIES]

Coverage that follows your attack surface.



Web applications

Single-page and server-rendered apps, mapped endpoint by endpoint with every parameter and state transition.



APIs

REST, GraphQL and gRPC, authenticated or not, discovered from spec, live traffic and source and re-mapped as the API changes.



Auth and access control

Sessions, OAuth, SSO and multi-tenant boundaries. Requests are replayed across roles and tenants to prove IDOR and authz gaps.



Business logic

Agents reason about expected behavior and try to break it, catching flaws no signature or rule can express.



AI and LLM applications

Prompt injection, tool abuse and data exfiltration paths in agentic and LLM-backed features.



Audit-ready reports

Executive, customer-facing, auditor and remediation reports. Each finding carries CWE, request/response proof and repro steps.

[PRODUCT]

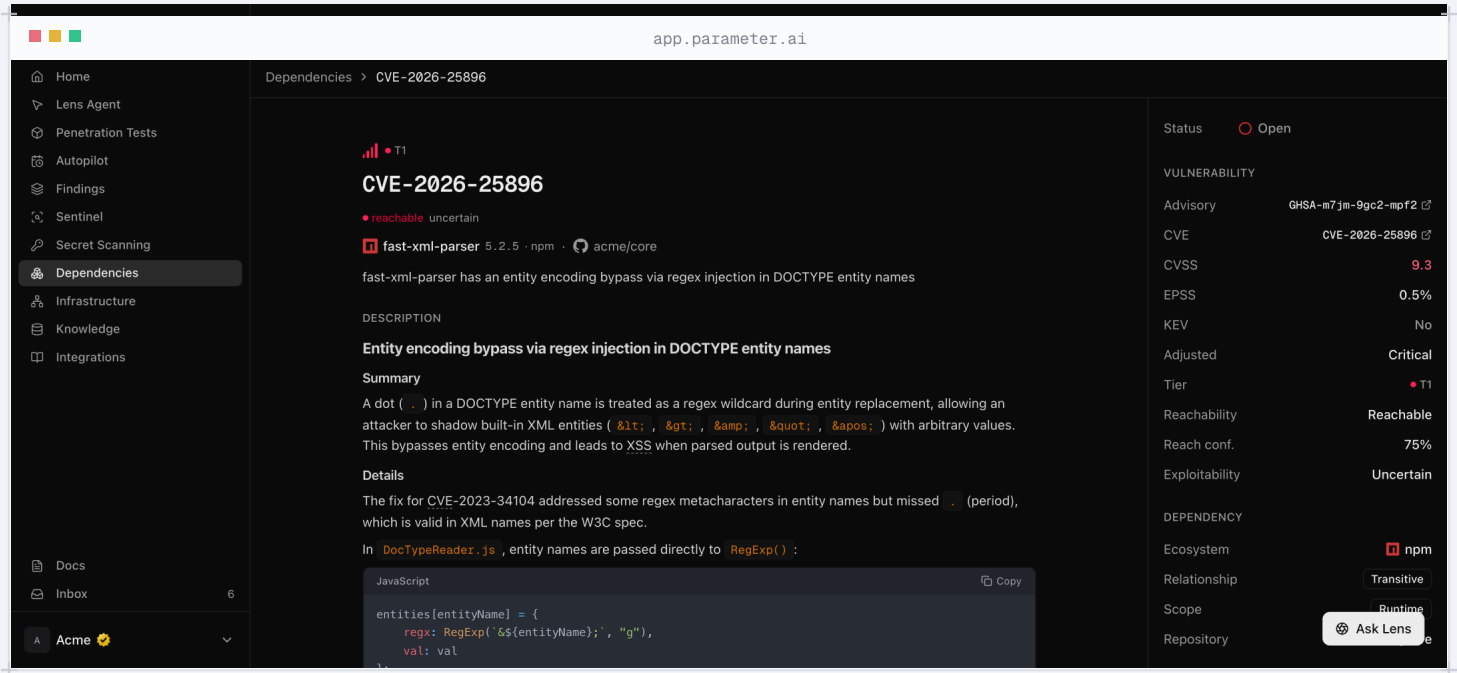


FIG. 02 A validated finding in Parameter: impact, the exact request and response that proves it, reproduction steps, and a generated fix.

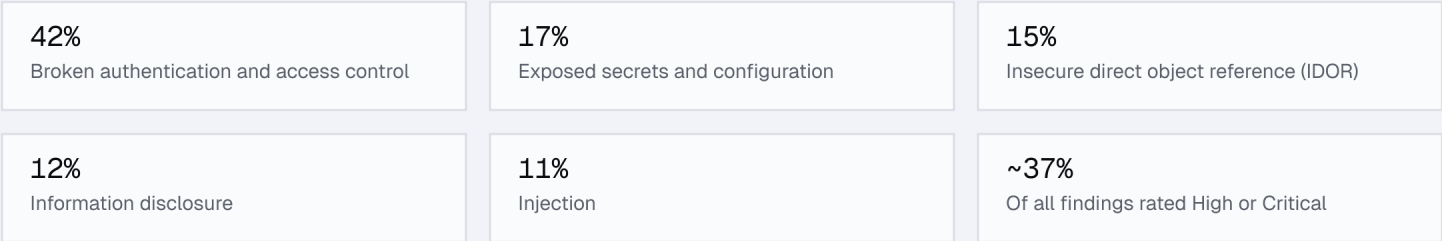
[TECHNICAL SPECIFICATIONS]

TEST MODES	Whitebox (source), greybox (OpenAPI), blackbox (URL + credentials)
TARGETS	Web apps, REST/GraphQL/gRPC APIs, auth flows, infrastructure, AI/LLM apps, Android
DEPLOYMENT	SaaS, agentless. Agents test the running application externally
CADENCE	On demand, on every release, and on every deploy (Continuous tier)
EVIDENCE	Request, response and state change captured for every finding; CWE tagged
REMEDIATION	Generated fix pull requests; automatic re-test on fix
ROUTING	Jira, Linear, Slack
COMPLIANCE MAPPING	SOC 2 CC4.1, CC7.1, CC7.2, CC7.4/7.5; ISO 27001 A.8.8; PCI DSS v4.0 Req. 11.4

SAFETY MODEL

- **Confirm and hold**
Agents prove a vulnerability exists, then stop. No chaining or escalation without your explicit go-ahead.
- **Scoped, never stray**
Agents stay inside the targets you authorize. No wandering into systems that are not in scope.
- **No destructive actions**
No dropped tables, no deleted data, no denial of service. Safe against production by design.
- **Full audit trail**
Every action an agent takes is logged and reviewable, so you can see exactly what happened.

[WHAT A TYPICAL RUN FINDS]



Distribution across a 1,200-finding sample from ~2,300 autonomous tests (parameter.ai/blog).

[WHY PARAMETER]

- ✔ **Proven, not predicted**
Every finding ships with a working proof-of-concept. Unproven findings are dropped before they reach you.
- ✔ **<1% false positives**
Validation from a clean state keeps the queue short and every entry real.
- ✔ **Top 3 in the US on HackerOne**
Offensive-security pedigree behind the agents, with independent third-party review.
- ✔ **Same-day results**
First findings within 24 hours; most runs finish the same day. No engagement to book.

[PART OF THE PARAMETER PLATFORM]

Pentesting Agent shares one queue with Sentinel, Cloud Security, Supply Chain and Secrets Detection, so a finding proven in the running app is correlated with the code, cloud resource or dependency behind it.

See first findings in 24 hours.

ABOUT PARAMETER

Parameter is an AI-powered security platform that continuously finds and validates vulnerabilities before attackers can exploit them. Autonomous agents pentest like a real adversary across your code, cloud and dependencies, with every finding proven. Formerly Hex Security, Parameter is backed by Y Combinator and built by a team that has secured Apple, Google, Microsoft, the U.S. Department of Defense, T-Mobile, Henkel, Epic Games, AT&T and Yahoo.

TRUST AND COMPLIANCE

SOC 2 Type I (Type II observation underway)
HIPAA BAA available
GDPR DPA with SCCs
Data processed in the US
parameter.ai · support@parameter.ai · app.parameter.ai