DAST COVERAGE · DATASHEET

Safeguard

Test the running application, safely, and only what you own.

Safeguard offers broad first-party DAST coverage of web applications and APIs, inside a safety kernel that keeps every request in scope.

THE PROBLEM

Some flaws only show when the application runs. Teams hesitate to test live systems because dynamic scanners can break things or point at hosts nobody owns, and a flood of repeated findings buries the few that matter.

WHAT IT DOES

Verified targets only

A target stays unverified until you prove ownership by DNS TXT record, file, meta tag or email domain. Active checks cannot run on an unverified target.

Scope and safety in code

Requests are checked against an allow-listed host and path scope, rate limited and non-destructive. Out-of-scope requests are blocked and logged.

Broad first-party coverage

Headers, injection classes, API surface and misconfiguration, run by Safeguard's own DAST engine, with API testing from OpenAPI, curl or Postman.

Findings you can act on

A host-level issue is reported once per host, not once per page, and a verified-only option keeps tentative rows out of the results.

HOW IT WORKS

1. **Register** — Add the target and its scope
2. **Verify** — Prove ownership of the target
3. **Test** — Safe, rate-limited requests
4. **Findings** — Headers, injection, configuration, API
5. **Correlate** — Linked to the SAST source sink

CONTROLS LIVE IN CODE

Verification, scope and rate limits are enforced by the scanner itself, not by a screen. The CLI refuses an unverified or out-of-scope target whatever flags you pass.

CAPABILITIES IN DETAIL

Coverage	Broad first-party DAST coverage: security headers, injection classes, misconfiguration and API surface.
Verification	DNS TXT record, uploaded file, meta tag or verified email domain.
Safety modes	Passive, safe active and authenticated safe active. A heavier lab mode is off by default and needs elevated authorization.
Guardrails	Allow-listed scope, per-target rate limits, maximum duration and benign detection markers. No destructive or denial-of-service payloads.
Authenticated scans	Test behind a login with a test account. A lapsed session on an interactive code login marks the scan partial and says so.
Result handling	Host-level issues reported once per host. Injection and parameter findings keep one finding per path and parameter.
Command line	`safeguard appsec dast` takes a target ID, base URL and scope host.

INPUTS

- A registered and verified target
- Scope hosts and paths, and a safety mode
- Credentials for a test account
- OpenAPI, Swagger, curl or Postman for API targets

OUTPUTS

- Findings with severity, rule, CWE and location
- A Hosts tested panel showing what was and was not covered
- Findings linked to source and dependency findings
- An audit trail

INTEGRATIONS AND SURFACES

Postman · OpenAPI and Swagger · Safeguard CLI · CI/CD actions · Jira · Linear · Asana · Slack · Teams

DEPLOYMENT AND DATA HANDLING

Private addresses
Local and private addresses cannot be reached from Safeguard's scanner. Scan those with a runner inside your network.

Credentials
Tokens and keys sign the scan requests and are removed from findings, inventory and reports.

Regions and hosting
SaaS in US, EU (Frankfurt) and Gov regions. Private VPC, on-prem and fully air-gapped deployments are available.

Access control
SSO (SAML 2.0, OpenID, Google, Microsoft, GitHub), roles and 2FA.

See the Safeguard platform

`safeguard.sh/platform`

Scan the code or type the address. Questions: hi@safeguard.sh

Safeguard

SOC 2 Type I and Type II audited; reports available on request (safeguard.sh/trust). Designed for FedRAMP High.