CYCOGNITO

TECHNOLOGY CHECKLIST

When to Use DAST for OWASP Top 10 Testing

A Practical Checklist for IT Security Practitioners

Dynamic Application Security Testing (DAST) is essential for uncovering vulnerabilities that surface during an application's runtime.

While static AST (SAST) and Software Composition Analysis (SCA) catch many issues early, DAST is critical for identifying real-world flaws tied to user input, configuration, and environment differences.

This checklist helps IT security practitioners quickly determine when DAST should be applied, ensuring better coverage across the OWASP Top 10 risks. If these items apply to your app, DAST should be considered to detect the issue effectively.

When to Use DAST

ISSUE 1

Broken Access Control

- ☑ App uses multiple roles or permission levels.
- ☑ Critical functions differ between user roles.
- ☑ Access rules are enforced both client- and server-side.

Use DAST to: Test for role bypass, IDOR, and privilege escalation.

ISSUE 2

Injection (SQL, NoSQL, OS Command)

- ☑ Dynamic queries or commands are executed.
- ☑ Forms or URL parameters interact with databases/backends.
- ☑ Logs show suspicious or malformed inputs.

Use DAST to: Probe for injections and error-based attacks.

ISSUE 3

Cross-Site Scripting (XSS)

- ☑ App reflects or stores user-supplied data.
- ☑ Dynamic content uses risky APIs (e.g., innerHTML).
- ☑ Inputs allow HTML, scripts, or special characters.

Use DAST to: Detect reflected, stored, and DOM-based XSS.

ISSUE 4

Security Misconfiguration

- ☑ Debug panels, error messages, or stack traces visible.
- ☑ Missing critical HTTP headers (CSP, HSTS, X-Frame-Options).
- ☑ Outdated or default server configs.

Use DAST to: Identify misconfigurations and missing protections.

ISSUE 5

Sensitive Data Exposure

- App processes PII, credentials, payment information.
- Non-HTTPS endpoints are exposed.
- Sensitive info leaks in URLs, logs, page content.

Use DAST to: Detect improper data handling during runtime.

ISSUE 6

Broken Authentication

- App includes login, session, or MFA workflows.
- Password policies are configurable.
- Long-lived or insecure session cookies in use.

Use DAST to: Test brute-force resistance, session hijacking.

ISSUE 7

Cross-Site Request Forgery (CSRF)

- App performs critical changes via HTTP POST.
- CSRF tokens or Origin/Referer checks are missing.
- Authentication is based mainly on cookies.

Use DAST to: Confirm CSRF protections are active and effective.

Where DAST is Partially Effective or Not Effective

ISSUE 8

Insecure Deserialization

- Serialized data objects (e.g., JSON, JWT, XML) are accepted.
- App processes encoded tokens or cookies.
- Crashes or errors from tampered inputs observed.

DAST Partially Effective: May require manual validation or SCA tools.

ISSUE 9

Server-Side Request Forgery (SSRF)

- App accepts URLs, fetches remote resources, or handles uploads.
- App behavior changes based on modified URLs or file requests.

DAST Partially Effective: Manual testing or SSRF-focused tooling also needed.

ISSUE 10

Insufficient Logging & Monitoring

- No real-time alerting for login attempts, privilege changes, or abuse.
- No SIEM integration or centralized log monitoring.

  Incidents discovered late or reactively.

External DAST Not Applicable: Focus on log management and incident detection.

How Can CyCognito Help Your Organization?

CyCognito continuously tests your exposed applications with unauthenticated black box testing from the outside-in. Our platform identifies critical OWASP-class vulnerabilities in real-world conditions, prioritizes them based on business risk, and accelerates remediation with clear, actionable guidance.

Check out our website and explore our platform with a self-guided, interactive dashboard product tour. To learn more about how CyCognito can help you identify and remediate emerging threats to your attack surface, request a customized demo.