AYI NEDJIMI Consultants

SAST, DAST, IAST : bien choisir vos outils de tests

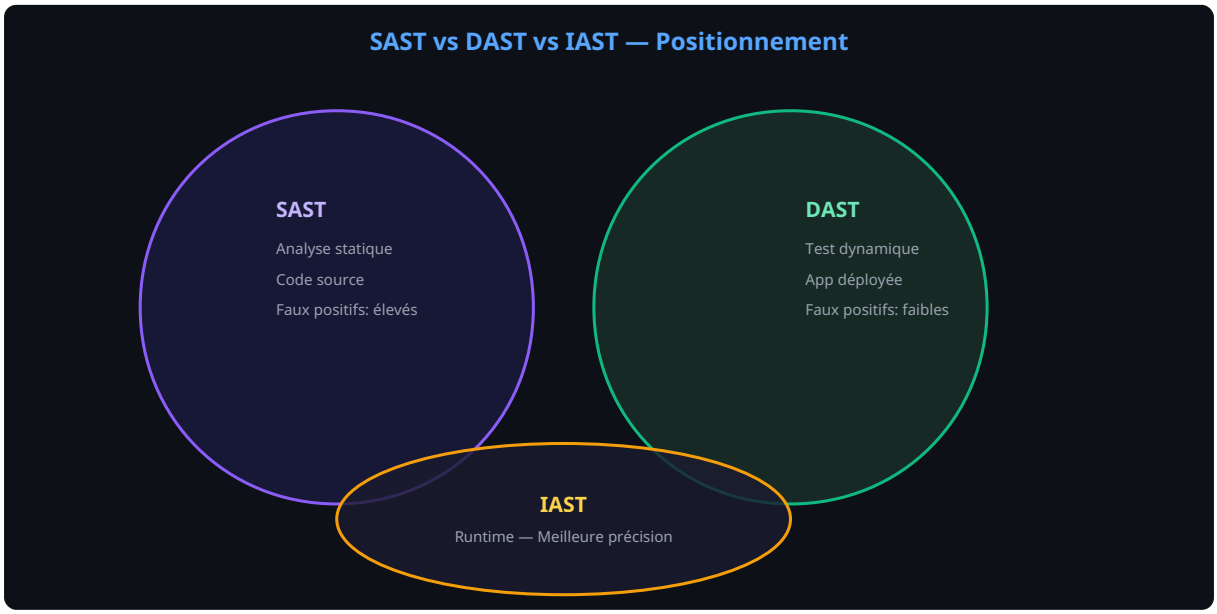
Catégorie : DevSecOps Lecture : 5 min Publié le : 12/03/2026 Auteur : Ayi NEDJIMI

Comparaison détaillée des approches SAST, DAST et IAST pour les tests de sécurité applicative. Forces, limites et stratégie de déploiement CI.

Vous avez décidé d'intégrer des tests de sécurité dans votre cycle de développement. Bonne décision. Mais face à l'offre pléthorique d'outils — Semgrep, SonarQube, Checkmarx pour le SAST, OWASP ZAP et Burp Suite pour le DAST, Contrast Security pour l'IAST — comment choisir la bonne approche ? La réponse courte : vous avez besoin des trois, mais pas de la même façon ni au même moment. Le SAST analyse votre code source sans l'exécuter, le DAST teste votre application déployée comme le ferait un attaquant, et l'IAST combine les deux en instrumentant votre runtime. Chaque méthode couvre des classes de vulnérabilités différentes, avec des taux de faux positifs et des temps d'exécution qui varient considérablement. Ce comparatif vous donne les critères concrets pour bâtir votre stratégie de tests sécurité, avec des retours terrain sur les outils les plus utilisés en 2025-2026 et des recommandations adaptées à la taille de votre équipe.

Points clés à retenir

- Le **SAST** détecte les vulnérabilités tôt mais génère beaucoup de faux positifs (20-40% selon l'outil)
- Le **DAST** trouve les problèmes de configuration et d'authentification que le SAST ne voit pas
- L'**IAST** offre le meilleur ratio précision/couverture mais nécessite un environnement d'exécution instrumenté
- Une stratégie mature combine SAST en CI, DAST en staging et IAST en QA



SAST : l'analyse statique au plus près du code

Le Static Application Security Testing scanne votre code source ou bytecode sans exécuter l'application. C'est la méthode qui intervient le plus tôt dans le cycle de développement — idéalement à chaque pull request. **Semgrep** est devenu la référence open-source : rapide (100K lignes en moins d'une minute), extensible avec des règles personnalisées. **CodeQL**, intégré à GitHub, excelle pour l'analyse de dataflow. **SonarQube** couvre 15 langages mais reste plus axé qualité que sécurité pure. **Checkmarx SAST** est la solution entreprise avec le meilleur support des frameworks Java et .NET.

Le principal reproche fait au SAST : les faux positifs. Semgrep affiche un taux d'environ 20% sur les règles OWASP Top 10, SonarQube monte à 30-35%. La parade : tuner les règles, supprimer celles qui ne s'appliquent pas à votre stack, et maintenir un fichier d'exclusions. Un SAST non tuné, c'est un outil qui finit ignoré. Pour intégrer ces outils dans votre chaîne, consultez notre guide du pipeline CI/CD sécurisé.

DAST : tester l'application comme un attaquant

Le Dynamic Application Security Testing attaque votre application déployée. Pas besoin d'accès au code source — l'outil interagit avec l'interface HTTP, envoie des payloads malveillants et observe les réponses. **OWASP ZAP** est l'outil DAST open-source le plus utilisé au monde. Mode CLI pour l'intégration CI, mode GUI pour l'exploration manuelle. Le scan baseline prend 5-10 minutes. **Nuclei** mise sur des templates communautaires : 8000+ templates couvrant CVE, misconfigurations et expositions. **Burp Suite Professional** reste la référence pour le pentest combiné automatisé et manuel.

Le DAST excelle pour détecter les problèmes que le SAST ne voit jamais : mauvaise configuration CORS, headers de sécurité manquants, cookies sans flags Secure/HttpOnly, endpoints exposés sans authentification. En revanche, le DAST ne vous dit pas où dans le code se trouve le problème. Pour une vision complète de votre surface d'attaque, notre article sur l'attack surface management complète cette approche.

IAST : le meilleur des deux mondes

L'Interactive Application Security Testing instrumente votre application pendant son exécution. Un agent (Java agent, .NET profiler, Node.js middleware) observe les flux de données en temps réel et détecte les vulnérabilités avec le contexte du code source ET du comportement runtime. **Contrast Security** mène le marché avec des agents pour Java, .NET, Node.js, Python et Ruby. L'avantage majeur : un taux de faux positifs inférieur à 5%. L'agent voit le flux de données réel, pas une approximation statique.

Le point faible : la couverture dépend des scénarios de tests exécutés. Si votre suite de tests fonctionnels ne couvre que 60% du code, l'IAST ne verra que 60% des vulnérabilités potentielles. C'est pourquoi l'IAST complète le SAST sans le remplacer. Le coût est aussi un frein : comptez 30K euros par an minimum pour Contrast, contre zéro pour Semgrep et ZAP.

Tableau comparatif SAST, DAST et IAST

Critère	SAST	DAST	IAST
Phase du cycle	Développement	Staging / QA	QA / Pre-prod
Accès au code requis	Oui	Non	Oui (agent)
Taux de faux positifs	20-40%	5-15%	2-5%
Temps de scan moyen	3-10 min	10-30 min	Temps réel
Vulnérabilités de config	Non	Oui	Partiel
Localisation dans le code	Précise	Non	Précise
Coût open-source	Gratuit	Gratuit	Rare
Coût entreprise	5-50K/an	8-30K/an	20-80K/an

Construire une stratégie combinée efficace

La recommandation selon le guide OWASP DevSecOps Guideline est claire : combinez au minimum SAST + DAST. Voici le modèle que je recommande pour une équipe de 10-30 développeurs :

1. **SAST sur chaque PR** — Semgrep avec les règles OWASP Top 10 + règles custom. Gate bloquante sur CRITICAL.
2. **DAST hebdomadaire** — ZAP full scan sur staging, chaque lundi. Rapport envoyé dans Slack.
3. **IAST en QA** — Si votre stack le permet (Java/.NET), activez Contrast pendant les tests d'acceptance.
4. **Pentest trimestriel** — Aucun outil automatisé ne remplace un testeur humain pour la logique métier.

Cette approche couvre plus de 90% des vulnérabilités courantes sans ralentir significativement votre cadence de livraison. Pour la gestion des vulnérabilités découvertes, consultez notre guide sur le triage et la priorisation DevSecOps. Le référentiel NIST Software Quality Group fournit des métriques complémentaires.

Retour terrain : migration vers une stratégie combinée

Une équipe fintech que j'ai accompagnée utilisait uniquement Checkmarx SAST. Résultat : 1200 findings ouverts, dont 900 faux positifs, et des développeurs qui avaient appris à ignorer toutes les alertes. Nous avons procédé en trois étapes. D'abord, nettoyage des règles Checkmarx : suppression de 40% des règles non pertinentes pour leur stack Spring Boot + React. Les findings sont passés de 1200 à 350. Ensuite, ajout d'OWASP ZAP sur le staging : 15 vulnérabilités réelles découvertes dès le premier scan — des headers manquants, un endpoint admin exposé, deux redirections ouvertes. Le SAST ne les avait jamais vues.

Enfin, déploiement de Contrast en monitoring sur leur environnement de QA : 8 vulnérabilités critiques identifiées avec un contexte de code précis, remédiées en moins d'une semaine. Le taux de faux positifs global est passé de 75% à 8%. Pour la détection de secrets qui accompagne ces tests, voyez notre article sur le secrets sprawl.

Sources et références : OWASP DevSecOps · NIST

Questions fréquentes sur les tests de sécurité applicative

Peut-on se passer de DAST si on a un bon SAST ?

Non. Le SAST et le DAST couvrent des catégories de vulnérabilités différentes. Le SAST ne détecte pas les problèmes de configuration serveur, les headers manquants, les erreurs CORS ou les failles d'authentification liées au déploiement. Le DAST complète le SAST, il ne le remplace pas.

L'IAST ralentit-il l'application en production ?

L'IAST n'est pas conçu pour la production — c'est un outil de test. L'overhead de l'agent Contrast en environnement de test est d'environ 3-5% sur les temps de réponse. Pour la protection en production, regardez plutôt les solutions RASP (Runtime Application Self-Protection), qui sont optimisées pour cet usage.

Quel budget prévoir pour une PME de 20 développeurs ?

En full open-source (Semgrep + ZAP + Trivy), le budget outil est nul. Comptez 2 sprints d'un ingénieur DevOps pour la mise en place. Si vous optez pour des solutions commerciales, prévoyez 15 à 25K euros par an pour Snyk ou Checkmarx en formule équipe. L'IAST avec Contrast démarre à 30K euros par an — réservez-le pour les applications critiques.

Ayi NEDJIMI Consultants — Expert cybersécurité offensive & intelligence artificielle

ayinedjimi-consultants.fr · ayi@ayinedjimi-consultants.fr