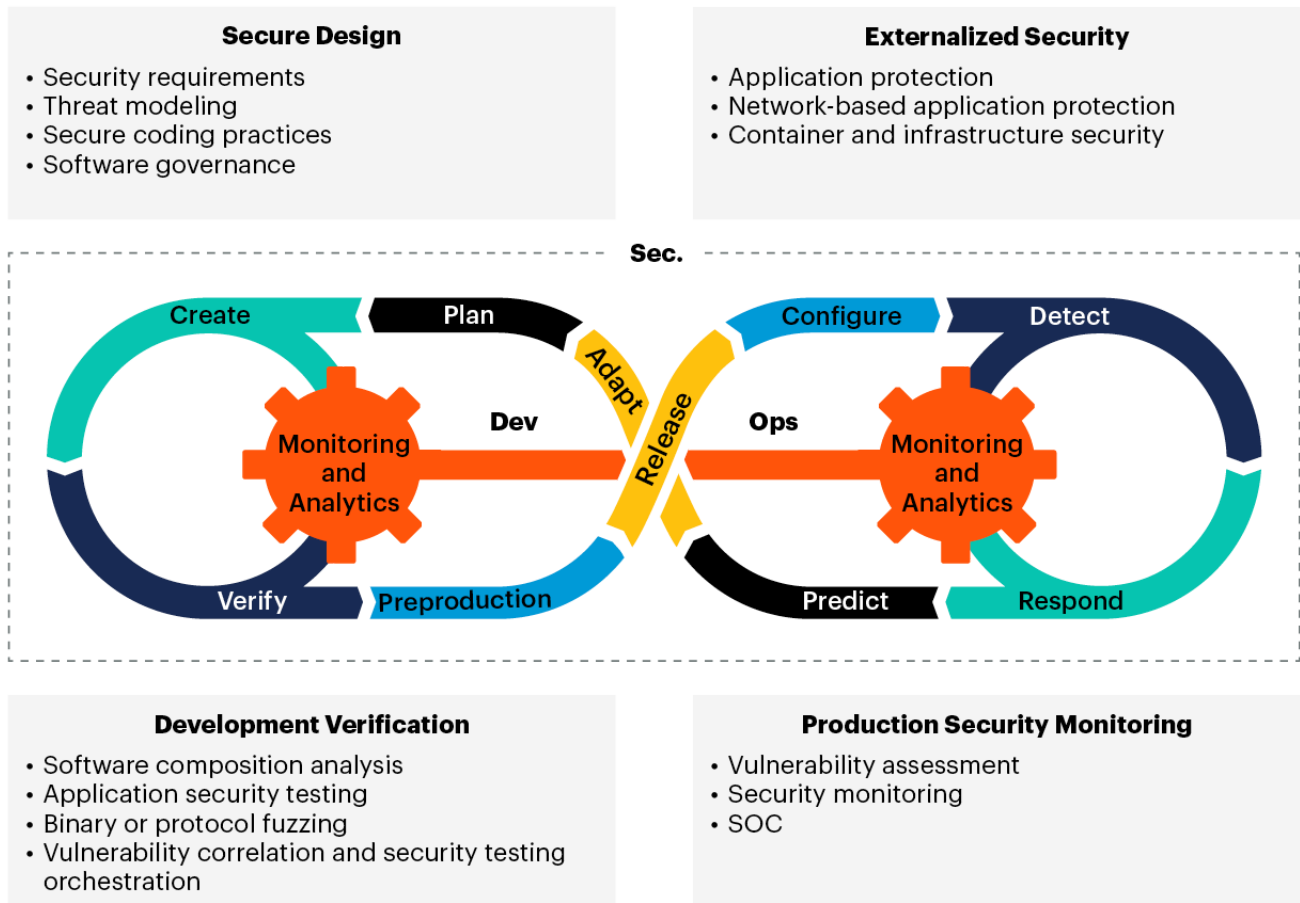


DevSecOps-Vault

Collection of roadmaps, tools, best practice, resources about DevSecOps

Gartner DevSecOps Model



Source: Gartner

729065_C

Gartner

What is DevSecOps

DevSecOps is an approach to software development that integrates security into the entire software development lifecycle. It combines the principles of DevOps with security to create a culture of security that permeates the entire development process.

The goal of DevSecOps is to ensure that security is not an afterthought but is built into the development process from the very beginning. This approach emphasizes collaboration between development, operations, and security teams to ensure that security is a shared responsibility across the organization.

DevSecOps involves the implementation of security controls throughout the development lifecycle, from design to deployment, and beyond. This includes automated security testing, continuous monitoring, vulnerability assessments, and penetration testing.

By implementing DevSecOps practices, organizations can improve their overall security posture and reduce the risk of security breaches and vulnerabilities. This approach also helps to accelerate the delivery of secure software by integrating security into the development process, rather than treating it as a separate activity.

Take a look at links for detailed explanation on DevSecOps:

- [Redhat definitions](#)
- [IBM definitions](#)
- [Synk definitions](#)
- [Synopsys definitions](#)
- [Spacelift definitions](#)

DevSecOps Security Checklist



Checklist Link

1. Design

- Development Lifecycle
 1. [SDL\(Secure Development Lifecycle\) by Microsoft](#)
 2. [OWASP's Software Assurance Maturity Model](#)
 3. [Building Security In Maturity Model \(BSIMM\)](#)
 4. [NIST's Secure Software Developerment Framework](#)
 5. [DevSecOps basics: 9 tips for shifting left \(Gitlab\)](#)
 6. [6 Ways to bring security to the speed of DevOps \(Gitlab\)](#)
- Threat Model
 1. [What is Threat Modeling / Wikipedia](#)
 2. [Threat Modeling by OWASP](#)
 3. [Application Threat Modeling by OWASP](#)
 4. [Agile Threat Modeling Toolkit](#)
 5. [OWASP Threat Dragon](#)

2. Develop

- Secure Coding
 1. [Secure coding guide by Apple](#)
 2. [Secure Coding Guidelines for Java SE](#)
 3. [Go-SCP / Go programming language secure coding practices guide](#)
 4. [Android App security best practices by Google](#)

5. Securing Rails Applications

3. Build

- SAST(Static Application Security Testing)
 1. [Scan Source Code using Static Application Security Testing \(SAST\) with SonarQube, Part 1](#)
 2. [Announcing third-party code scanning tools: static analysis & developer security training](#)

4. Test

- DAST(Dynamic Application Security Testing)
 1. [Dynamic Application Security Testing with ZAP and GitHub Actions](#)
 2. [Dynamic Application Security Testing \(DAST\) in Gitlab](#)
 3. [DAST using pdiscoveryio Nuclei \(github action\)](#)
 4. [ZAPCon 2021-Democratizing ZAP with test automation and domain specific languages](#)
- Penetration testing
 1. [Penetration Testing at DevSecOps Speed](#)

5. Deploy

- Security Hardening & Config
 1. [CIS Benchmarks](#)
 2. [DevSecOps in Kubernetes](#)
- Security Scanning
 1. [Best practices for scanning images \(docker\)](#)

6. Operate and Monitor

- RASP(Run-time Application Security Protection)
 1. [Runtime Application Self-Protection by rapid7](#)
 2. [Jumpstarting your devsecops - Pipeline with IAST and RASP](#)
- Security Patch
 1. [RASP\(Runtime Application Self-Protection\)](#)
- Security Audit
- Security Monitor
 1. [IAST\(Interactive Application Security Testing\)](#)
 2. [Metrics, Monitoring, Alerting](#)
- Security Analysis
 1. [Attack Surface Analysis Cheat Sheet by OWASP](#)

Resources

Books

Books focussed around DevSecOps, bringing the security focus up front.

- [DevOpsSec](#)
- [Docker Security - Quick Reference](#)

- [Holistic Info-Sec for Web Developers](#)
- [Securing DevOps](#)
- [The DevOps Handbook \(Section VI\)](#)

Conferences

A body of knowledge for combining DevOps and Security has been delivered via conferences and meetups. This is a short list of the venues that have dedicated a portion of their agenda to it.

- [AWS re:Inforce](#)
- [AWS re:Invent](#)
- [DevSecCon](#)
- [DevOps Connect](#)
- [DevOps Days](#)
- [Goto Conference](#)
- [IP Expo](#)
- [ISACA Ireland](#)
- [RSA Conference](#)
- [All Day DevOps](#)

Training

DevSecOps requires an appetite for learning and agility to quickly acquire new skills. We've collected these links to help you learn how to do DevSecOps with us.

Labs

Labs are hands-on learning opportunities to grow your skills in Dev, Sec, and Ops. All skills are useful and need to be grown so that you can have the empathy, knowledge and trade to operate DevSecOps style.

- [DevSecOps Bootcamp](#)
- [Exercism](#)
- [Infoseclabs](#)
- [Infrastructure Monitoring](#)
- [Pentester Lab](#)
- [Vulnhub](#)

Vulnerable Test Targets

It's important to build up knowledge by learning how to break applications left vulnerable by security mistakes. This section contains a list of vulnerable apps that can be deployed to learn what not to do. These same apps can be made safe by remediating the intentional vulnerabilities to learn how to prevent attackers from gaining access to underlying infrastructure or data.

- [Damn Vulnerable Web Application](#) (PHP/MySQL)
- [LambHack](#) (Lambda)
- [Metasploitable](#) (Linux)
- [Mutillidae](#) (PHP)
- [NodeGoat](#) (Node)

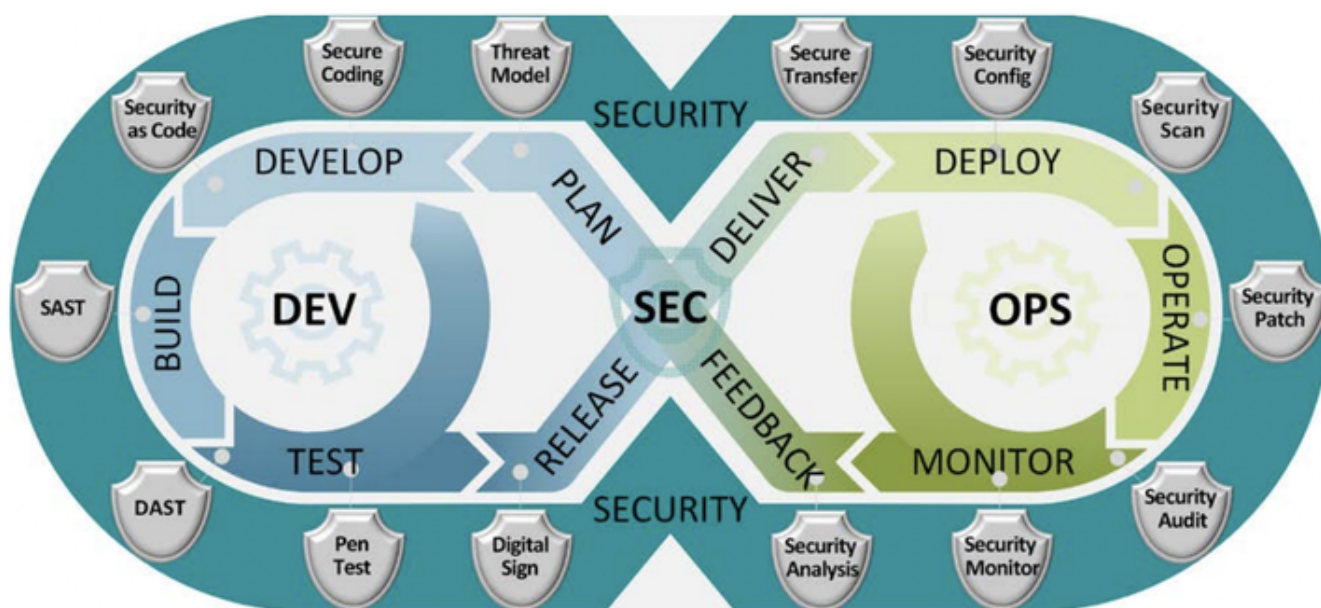
- [OWASP Damn Vulnerable Serverless Application \(DVSA\)](#) (AWS Serverless)
- [OWASP Juice Shop](#) (NodeJS/Angular)
- [RailsGoat](#) (Rails)
- [WebGoat](#) (Web App)
- [WebGoat.Net](#) (.NET)
- [WebGoatPHP](#) (PHP)

Security of CI/CD

DO NOT FORGET MANY DEVSECOPS PIPELINE SUFFER FROM SECURITY VULNERABILITIES

- Github Actions
 1. [Security hardening for GitHub Actions](#)
 2. [Github Actions Security Best Practices](#)
 3. [GitHub Actions Security Best Practices \[cheat sheet included\]](#)
- Jenkins
 1. [Securing Jenkins](#)
 2. [Securing Jenkins CI Systems by SANS](#)
 3. [DEPRECATED/chef-jenkins-hardening](#)

Tools and Links for DevSecOps



DevOps loop contains ==> Plan, Code, Build, Test, Release, Deploy, Operate, Monitor.

Pre-commit tools

DevSecOps pre-commit tools are a critical part of the software development lifecycle, particularly for organizations that prioritize security in their development processes. These tools help ensure that security considerations are integrated into every step of the development process, from coding to deployment.

Here are some reasons why DevSecOps pre-commit tools are important:

Early detection of security vulnerabilities: Pre-commit tools can identify security issues early in the development process, before code is even committed to the repository. This allows developers to address security concerns immediately, rather than discovering them later in the development process when they can be much more difficult and expensive to fix.

Consistent security standards: Pre-commit tools can enforce consistent security standards across the development team, ensuring that security best practices are followed by all developers.













Reduced risk of security breaches: By catching security issues early, pre-commit tools can help reduce the risk of security breaches and protect sensitive data.

Improved efficiency: Pre-commit tools can help developers catch security issues before they become more complex and time-consuming to fix, which can ultimately lead to more efficient development processes.

Enhanced collaboration: Pre-commit tools can help facilitate collaboration between developers and security teams by providing a common language and set of standards for discussing security issues.

Overall, DevSecOps pre-commit tools play a critical role in ensuring that security is integrated into the software development lifecycle. By catching security issues early and enforcing consistent standards, these tools help organizations build more secure and efficient software.

Name	URL	Description	Meta
git-secrets	https://github.com/awslabs/git-secrets	AWS labs tool preventing you from committing secrets to a git repository	STARS 116
git-hound	https://github.com/tillson/git-hound	Searchers secrets in git	STARS 981
goSDL	https://github.com/slackhq/goSDL	Security Development Lifecycle checklist	STARS 801
ThreatPlaybook	https://github.com/we45/ThreatPlaybook	Threat modeling as code	STARS 249
Threat Dragon	https://github.com/OWASP/threat-dragon	OWASP Threat modeling tool	STARS 496
threatspec	https://github.com/threatspec/threatspec	Threat modeling as code	STARS 269
pytm	https://github.com/izar/pytm	A Pythonic framework for threat modeling	STARS 877
Threagile	https://github.com/Threagile/threagile	A Go framework for threat modeling	STARS 441
MAL-lang	https://mal-lang.org/#what	A language to create cyber threat modeling systems for specific domains	STARS 21
Microsoft Threat modeling tool	https://docs.microsoft.com/en-us/azure/security/develop/threat-modeling-tool	Microsoft threat modeling tool	STARS 145

Name	URL	Description	Meta
Talisman	https://github.com/thoughtworks/talisman	A tool to detect and prevent secrets from getting checked in	 STARS  3.7K
SEDATED	https://github.com/OWASP/SEDATED	The SEDATED® Project (Sensitive Enterprise Data Analyzer To Eliminate Disclosure) focuses on preventing sensitive data such as user credentials and tokens from being pushed to Git.	 STARS  101
Sonarlint	https://github.com/SonarSource/sonarlint-core	Sonar linting utility for IDE	 STARS  205
DevSkim	https://github.com/microsoft/DevSkim	DevSkim is a framework of IDE extensions and language analyzers that provide inline security analysis	 STARS  781
detect-secrets	https://github.com/Yelp/detect-secrets	Detects secrets in your codebase	 STARS  2.4K
tflint	https://github.com/terraform-linters/tflint	A Pluggable Terraform Linter	 STARS  3.7K

Secrets management

#TODO: Add explanation here

****Secrets management includes managing, versioning, encryption, discovery, rotating, provisioning of passwords, certificates, configuration values and other types of secrets. ****

Secrets management is an essential component of DevSecOps, which is a methodology that integrates security practices into the software development process. Secrets management refers to the practice of managing sensitive information, such as passwords, encryption keys, API tokens, and other credentials, that are required for secure communication and authentication between software systems and services.

Effective secrets management is critical for DevSecOps for several reasons:

Protects against security breaches: Sensitive information is a prime target for attackers seeking to gain unauthorized access to systems and data. If secrets are not securely managed, it increases the risk of data breaches, which can have severe consequences for organizations.

Ensures compliance: Many regulations require the protection of sensitive data. Effective secrets management ensures that organizations are compliant with regulations like HIPAA, GDPR, and PCI-DSS.

Enables automation: DevSecOps relies on automation to streamline processes and reduce errors. Secrets management allows developers to automate the retrieval and use of sensitive information, reducing the risk




of manual errors and improving efficiency.

Facilitates collaboration: Effective secrets management allows multiple teams to access and use sensitive information securely, facilitating collaboration across teams.

Supports scalability: As organizations grow, the number of secrets that need to be managed increases. Effective secrets management supports the scalability of DevSecOps processes by enabling the secure management of a large number of secrets.

In summary, effective secrets management is essential for DevSecOps, as it protects against security breaches, ensures compliance, enables automation, facilitates collaboration, and supports scalability.

Name	URL	Description	Meta
GitLeaks	https://github.com/zricethezav/gitleaks	Gitleaks is a scanning tool for detecting hardcoded secrets	STARS 12K
ggshield	https://github.com/gitguardian/ggshield	GitGuardian shield (ggshield) is a CLI application that runs in your local environment or in a CI environment and helps you detect more than 350+ types of secrets and sensitive files.	STARS 1.3K
TruffleHog	https://github.com/trufflesecurity/truffleHog	TruffleHog is a scanning tool for detecting hardcoded secrets	STARS 11K
Hashicorp Vault	https://github.com/hashicorp/vault	Hashicorp Vault secrets management	STARS 27K
Mozilla SOPS	https://github.com/mozilla/sops	Mozilla Secrets Operations	STARS 32K
AWS secrets manager GH action	https://github.com/marketplace/actions/aws-secrets-manager-actions	AWS secrets manager docs	STARS 95
GitRob	https://github.com/michenriksen/gitrob	Gitrob is a tool to help find potentially sensitive files pushed to public repositories on Github	STARS 9.6K
git-wild-hunt	https://github.com/d1vious/git-wild-hunt	A tool to hunt for credentials in the GitHub	STARS 265
aws-vault	https://github.com/99designs/aws-vault	AWS Vault is a tool to securely store and access AWS credentials in a development environment	STARS 9.3K

Name	URL	Description	Meta
Knox	https://github.com/pinterest/knox	Knox is a service for storing and rotation of secrets, keys, and passwords used by other services	
Chef vault	https://github.com/chef/chef-vault	allows you to encrypt a Chef Data Bag Item	
Ansible vault	Ansible vault docs	Encryption/decryption utility for Ansible data files	

OSS and Dependency management


Open Source Software (OSS) and Dependency Management are two critical areas that intersect with Dependency Security Testing and Analysis in discovering supply chain attacks. Dependency management is the practice of managing and tracking the libraries, frameworks, and other components that are used to build software applications. Open Source Software (OSS) refers to software that is developed and distributed under a license that allows developers to freely modify and distribute the source code.

Dependency security testing and analysis involve assessing the security of the dependencies used in software applications to identify and mitigate any vulnerabilities that could be exploited by attackers. One important aspect of this process is creating a Software Bill of Materials (SBOM), which is a list of all the components that are used in a software application. The SBOM helps identify any vulnerabilities and potential security risks in the software supply chain.













Following dependency scanning, also known as Software Composition Analysis (SCA), is a critical part of continuous integration (CI). CI is the practice of integrating code changes into a shared repository frequently, allowing developers to detect and resolve issues early in the development process. SCA involves scanning and analyzing the source code and libraries used in the software application to identify any known vulnerabilities or potential security risks.

Data series and data trends tracking should also be a part of CI tooling. This allows developers to track changes in the dependencies and the software supply chain over time, enabling them to identify and address any security risks that may arise.

In summary, Dependency security testing and analysis is critical for discovering supply chain attacks. SBOM creation and following dependency scanning is a crucial part of continuous integration. It is essential to know what you produce and what you consume in the context of libraries and packages, and data series and data trends tracking should be part of CI tooling.

Name	URL	Description	Meta
CycloneDX	https://github.com/orgs/CycloneDX/repositories	CycloneDX format for SBOM	

Name	URL	Description	Meta
cdxgen	https://github.com/AppThreat/cdxgen	Generates CycloneDX SBOM , supports many languages and package managers.	<small>STARS</small> 179
SPDX	https://github.com/spdx/spdx-spec	SPDX format for SBOM - Software Package Data Exchange	<small>STARS</small> 218
Snyk	https://github.com/snyk/snyk	Snyk scans and monitors your projects for security vulnerabilities	<small>STARS</small> 6.4K
vulncost	https://github.com/snyk/vulncost	Security Scanner for VS Code	<small>STARS</small> 184
Dependency Combobulator	https://github.com/apiiro/combobulator	Dependency-related attacks detection and prevention through heuristics and insight engine (support multiple dependency schemes)	<small>STARS</small> 83
DependencyTrack	https://github.com/DependencyTrack/dependency-track	Dependency security tracking platform	<small>STARS</small> 1.7K
DependencyCheck	https://github.com/jeremylong/DependencyCheck	Simple dependency security scanner good for CI	<small>STARS</small> 4.8K
Retire.js	https://github.com/retirejs/retire.js/	Helps developers to detect the use of JS-library versions with known vulnerabilities	<small>STARS</small> 5.1K

Name	URL	Description	Meta
PHP security checker	https://github.com/fabpot/local-php-security-checker	Check vulnerabilities in PHP dependencies	 STARS  3K
bundler-audit	https://github.com/rubysec/bundler-audit	Patch-level verification for bundler	 STARS  2.5K
gemnasium	https://gitlab.com/gitlab-org/security-products/analyzers/gemnasium	Dependency Scanning Analyzer based on Gemnasium	
Dependabot	https://github.com/dependabot/dependabot-core	Automated dependency updates built into GitHub providing security alerts	 STARS  5.3K
Renovatebot	https://github.com/renovatebot/renovate	Automated dependency updates, patches multi-platform and multi-language	 STARS  32K
npm-check	https://www.npmjs.com/package/npm-check	Check for outdated, incorrect, and unused dependencies.	 STARS  6.4K
Security Scorecards	https://securityscorecards.dev	Checks for several security health metrics on open source libraries and provides a score (0-10) to be considered in the decision making of what libraries to use.	 STARS  3.3K

Name	URL	Description	Meta
Syft	https://github.com/anchore/syft	CLI tool and library for generating an SBOM from container images (and filesystems).	 STARS 3.5K






Supply chain specific tools

The use of libraries plays a significant role in the security of the final product, as libraries are a common target for attackers to inject malicious code. Therefore, it is crucial to carefully select and evaluate the libraries used in the supply chain to avoid any security loopholes.

CI, on the other hand, involves regularly integrating and testing changes to the codebase, which is important for detecting any potential security breaches. The text suggests that monitoring the CI process should be done inside the tasks and jobs in pipeline steps to ensure continuous security checks throughout the development process.

Furthermore, to ensure the integrity of the final product, the text recommends storing integrity checks outside the system and performing several validation runs with comparison of integrity hashes or attestation. This is important because integrity checks help verify that the codebase has not been tampered with or compromised during the development process.

Overall, the text emphasizes the importance of careful library selection, continuous integration monitoring, and integrity checks to ensure the security of the final product in the supply chain.

Name	URL	Description	Meta
Tekton chains	https://github.com/tektoncd/chains	Kubernetes Custom Resource Definition (CRD) controller that allows you to manage your supply chain security in Tekton.	 STARS 198
in-toto	https://github.com/in-toto/attestation/tree/v0.1.0/spec	An in-toto attestation is authenticated metadata about one or more software artifacts	 STARS 106
SLSA	Official GitHub link	Supply-chain Levels for Software Artifacts	 STARS 2.1K
kritis	https://github.com/grafeas/kritis	Solution for securing your software supply chain for Kubernetes apps	 STARS 646
ratify	https://github.com/deislabs/ratify	Artifact Ratification Framework	 STARS 207

SAST








Static code analysis is a method of analyzing code without actually executing it. This is in contrast to dynamic code analysis, which involves analyzing code as it runs.

Static code review tools scan the source code to identify issues such as coding errors, security vulnerabilities, and performance issues. The tools typically work by analyzing the syntax and structure of the code to identify potential problems. The analysis may involve examining variables, functions, and code flow to determine potential issues.

In the context of security, static analysis tools are often referred to as Static Application Security Testing (SAST) tools. SAST tools work by analyzing the raw code to identify potential vulnerabilities. This is in contrast to Dynamic Application Security Testing (DAST) tools, which analyze the code as it runs.

The text notes that SAST tools usually work with the raw code and not with build packages. This means that SAST tools analyze the code before it has been compiled or packaged into a final product. This is important because it allows the tools to identify potential vulnerabilities before they become part of the final product.

Overall, the text describes how static code review tools work with source code to identify potential issues and vulnerabilities. SAST tools are a specific type of static analysis tool that focuses on identifying security vulnerabilities. By working with the raw code, SAST tools can identify potential issues before they become part of the final product.

Name	URL	Description	Meta
Brakeman	https://github.com/presidentbeef/brakeman	Brakeman is a static analysis tool which checks Ruby on Rails applications for security vulnerabilities	 6.4K
Semgrep	https://semgrep.dev/	Hi-Quality Open source, works on 17+ languages	 2.5K
Bandit	https://github.com/PyCQA/bandit	Python specific SAST tool	 9K
libsast	https://github.com/ajinabraham/libsast	Generic SAST for Security Engineers. Powered by regex based pattern matcher and semantic aware semgrep	 99
ESLint	https://eslint.org/	Find and fix problems in your JavaScript code	
nodejsscan	https://github.com/ajinabraham/nodejsscan	NodeJs SAST scanner with GUI	 2.1K
FindSecurityBugs	https://find-sec-bugs.github.io/	The SpotBugs plugin for security audits of Java web applications	 2K
SonarQube community	https://github.com/SonarSource/sonarqube	Detect security issues in code review with Static Application Security Testing (SAST)	 7.4K

Name	URL	Description	Meta
gosec	https://github.com/securego/gosec	Inspects source code for security problems by scanning the Go AST.	<small>STARS</small> 6.6K
Safety	https://github.com/pyupio/safety	Checks Python dependencies for known security vulnerabilities .	<small>STARS</small> 1.5K

Note: Semgrep is free CLI tool, however some rulesets (<https://semgrep.dev/r>) are having various licences, some can be free to use and can be commercial.

OWASP curated list of SAST tools : https://owasp.org/www-community/Source_Code_Analysis_Tools

DAST





DAST tools operate by simulating an attack on an application in order to identify vulnerabilities. The tools send requests to the application and analyze the responses for any potential security issues. These requests may include inputs such as SQL injection or cross-site scripting attacks, and the tool will check if the application responds in a way that indicates a security issue.

Unlike Static Application Security Testing (SAST), which analyzes the code without executing it, DAST tools test applications while they are running and actively deployed on a server or serverless function. This allows DAST tools to identify vulnerabilities that are specific to the environment in which the application is deployed.




DAST tools rely on inputs and outputs to operate. By injecting inputs into the application and analyzing its outputs, DAST tools can identify potential security issues that may be exploitable by attackers. DAST tools can also provide feedback on the severity of the vulnerabilities found, allowing developers to prioritize and address them accordingly.

Overall, DAST tools are an important part of an application's security testing process, particularly for web applications. They operate by simulating attacks on the application, and by analyzing the responses, they can identify potential security issues that could be exploited by attackers. By relying on inputs and outputs, DAST tools can provide detailed feedback on the severity of the vulnerabilities found, allowing developers to prioritize and address them.




Name	URL	Description	Meta
Zap proxy	https://owasp.org/www-project-zap/	Zap proxy providing various docker containers for CI/CD pipeline	<small>STARS</small> 11K
Wapiti	https://github.com/wapiti-scanner/wapiti	Light pipeline ready scanning tool	<small>STARS</small> 577
Nuclei	https://github.com/projectdiscovery/nuclei	Template based security scanning tool	<small>STARS</small> 12K


Name	URL	Description	Meta
purpleteam	https://github.com/purpleteam-labs/purpleteam	CLI DAST tool incubator project	 STARS 9.6
oss-fuzz	https://github.com/google/oss-fuzz	OSS-Fuzz: Continuous Fuzzing for Open Source Software	 STARS 8.5K
nikto	https://github.com/sullo/nikto	Nikto web server scanner	 STARS 6.5K
skipfish	https://code.google.com/archive/p/skipfish/	Skipfish is an active web application security reconnaissance tool	 STARS 585




Continuous deployment security

Name	URL	Description	Meta
SecureCodeBox	https://github.com/secureCodeBox/secureCodeBox	Toolchain for continuous scanning of applications and infrastructure	 STARS 375
OpenSCAP	https://github.com/OpenSCAP/openscap	Open Source Security Compliance Solution	 STARS 1.1K
ThreatMapper	https://github.com/deepfence/ThreatMapper	ThreatMapper hunts for vulnerabilities in your production platforms, and ranks these vulnerabilities based on their risk-of-exploit.	 STARS 3.1K










Kubernetes



Name	URL	Description	Meta
KubiScan	https://github.com/cyberark/KubiScan	A tool for scanning Kubernetes cluster for risky permissions	 STARS 1.1K
Kubeaudit	https://github.com/Shopify/kubeaudit	Audit Kubernetes clusters for various different security concerns	 STARS 2.6K
Kubescape	https://github.com/armosec/kubescape	The first open-source tool for testing if Kubernetes is deployed according to the NSA-CISA and the MITRE ATT&CK®.	 STARS 8.1K

Name	URL	Description	Meta
kubesecc	https://github.com/controlplaneio/kubesecc	Security risk analysis for Kubernetes resources	 919
kube-bench	https://github.com/aquasecurity/kube-bench	Kubernetes benchmarking tool	 5.7k
kube-score	https://github.com/zegl/kube-score	Static code analysis of your Kubernetes object definitions	 2.2k
kube-hunter	https://github.com/aquasecurity/kube-hunter	Active scanner for k8s (purple)	 4.2k
Calico	https://github.com/projectcalico/calico	Calico is an open source networking and network security solution for containers	 4.4k
Krane	https://github.com/appvia/krane	Simple Kubernetes RBAC static analysis tool	 579
Starboard	https://github.com/aquasecurity/starboard	Starboard integrates security tools by outputs into Kubernetes CRDs	 1.3k
Gatekeeper	https://github.com/open-policy-agent/gatekeeper	Open policy agent gatekeeper for k8s	 3k
Inspektor-gadget	https://github.com/kinvolk/inspektor-gadget	Collection of tools (or gadgets) to debug and inspect k8s	 1.3k
kube-linter	https://github.com/stackrox/kube-linter	Static analysis for Kubernetes	 2.2k
mizu-api-traffic-viewer	https://github.com/up9inc/mizu	A simple-yet-powerful API traffic viewer for Kubernetes enabling you to view all API communication between microservices to help your debug and troubleshoot regressions.	 8.5k
HelmSnyk	https://github.com/snyk-labs/helm-snyk	The Helm plugin for Snyk provides a subcommand for testing the images.	 37
Kubewarden	https://github.com/orgs/kubewarden/repositories	Policy as code for kubernetes from SUSE.	 57





Name	URL	Description	Meta
Kubernetes-sigs BOM	https://github.com/kubernetes-sigs/bom	Kubernetes BOM generator	
Capsule	https://github.com/clastix/capsule	A multi-tenancy and policy-based framework for Kubernetes	
Badrobot	https://github.com/controlplaneio/badrobot	Badrobot is a Kubernetes Operator audit tool	
Istio	https://istio.io	Istio is a service mesh based on Envoy. Engage encryption, role-based access, and authentication across services.	

Containers

Name	URL	Description	Meta
Harbor	https://github.com/goharbor/harbor	Trusted cloud native registry project	
Anchore	https://github.com/anchore/anchore-engine	Centralized service for inspection, analysis, and certification of container images	
Clair	https://github.com/quay/clair	Docker vulnerability scanner	
Deepfence ThreatMapper	https://github.com/deepfence/ThreatMapper	Apache v2, powerful runtime vulnerability scanner for kubernetes, virtual machines and serverless.	
Docker bench	https://github.com/docker/docker-bench-security	Docker benchmarking against CIS	
Falco	https://github.com/falcosecurity/falco	Container runtime protection	
Trivy	https://github.com/aquasecurity/trivy	Comprehensive scanner for vulnerabilities in container images	
Notary	https://github.com/notaryproject/notary	Docker signing	
Cosign	https://github.com/sigstore/cosign	Container signing	






Name	URL	Description	Meta
watchtower	https://github.com/containrrr/watchtower	Updates the running version of your containerized app	
Grype	https://github.com/anchore/grype	Vulnerability scanner for container images (and also filesystems).	

Multi-Cloud

Name	URL	Description	Meta
Cloudsploit	https://github.com/aquasecurity/cloudsploit	Detection of security risks in cloud infrastructure	
ScoutSuite	https://github.com/nccgroup/ScoutSuite	NCCgroup mutlicloud scanning tool	
CloudCustodian	https://github.com/cloud-custodian/cloud-custodian/	Multicloud security analysis framework	
CloudGraph	https://github.com/cloudgraphdev/cli	GraphQL API + Security for AWS, Azure, GCP, and K8s	

AWS

AWS specific DevSecOps tooling. Tools here cover different areas like inventory management, misconfiguration scanning or IAM roles and policies review.

Name	URL	Description	Meta
Dragoneye	https://github.com/indeni/dragoneye	Dragoneye Indeni AWS scanner	
Prowler	https://github.com/toniblyx/prowler	Prowler is a command line tool that helps with AWS security assessment, auditing, hardening and incident response.	
aws-inventory	https://github.com/nccgroup/aws-inventory	Helps to discover all AWS resources created in an account	
PacBot	https://github.com/tmobile/pacbot	Policy as Code Bot (PacBot)	
Komiser	https://github.com/mlabouardy/komiser	Monitoring dashboard for costs and security	

Name	URL	Description	Meta
Cloudsplaining	https://github.com/salesforce/cloudsplaining	IAM analysis framework	 STARS 1.7K
ElectricEye	https://github.com/jonrau1/ElectricEye	Continuously monitor your AWS services for configurations	 STARS 725
Cloudmapper	https://github.com/duo-labs/cloudmapper	CloudMapper helps you analyze your Amazon Web Services (AWS) environments	 STARS 2.4K
cartography	https://github.com/lyft/cartography	Consolidates AWS infrastructure assets and the relationships between them in an intuitive graph	 STARS 2.4K
policy_sentry	https://github.com/salesforce/policy_sentry	IAM Least Privilege Policy Generator	 STARS 1.6K
AirIAM	https://github.com/bridgecrewio/AirIAM	IAM Least Privilege analyzer and Terraformer	 STARS 699
StreamAlert	https://github.com/airbnb/streamalert	AirBnB serverless, real-time data analysis framework which empowers you to ingest, analyze, and alert	 STARS 2.6K
CloudQuery	https://github.com/cloudquery/cloudquery/	AirBnB serverless, real-time data analysis framework which empowers you to ingest, analyze, and alert	 STARS 4.1K
S3Scanner	https://github.com/sa7mon/S3Scanner/	A tool to find open S3 buckets and dump their contents	 STARS 2K
aws-iam-authenticator	https://github.com/kubernetes-sigs/aws-iam-authenticator/	A tool to use AWS IAM credentials to authenticate to a Kubernetes cluster	 STARS 2K
kube2iam	https://github.com/jtblin/kube2iam/	A tool to use AWS IAM credentials to authenticate to a Kubernetes cluster	 STARS 2.6K
AWS open source security samples	Official AWS opensource repo	Collection of official AWS open-source resources	

Name	URL	Description	Meta
AWS Firewall factory	Globaldatanet FMS automation	Deploy, update, and stage your WAFs while managing them centrally via FMS	<small>STARS</small> 321
Parliment	Parliment	Parliament is an AWS IAM linting library	<small>STARS</small> 916
Yor	Yor	Adds informative and consistent tags across infrastructure-as-code frameworks such as Terraform, CloudFormation, and Serverless	<small>STARS</small> 694

Google cloud platform



GCP specific DevSecOps tooling. Tools here cover different areas like inventory management, misconfiguration scanning or IAM roles and policies review.

Name	URL	Description	Meta
Forseti	https://github.com/forseti-security/forseti-security	Complex security orchestration and scanning platform	<small>STARS</small> 1.3K

Policy as code

Policy as code is the idea of writing code in a high-level language to manage and automate policies. By representing policies as code in text files, proven software development best practices can be adopted such as version control, automated testing, and automated deployment. (Source: <https://docs.hashicorp.com/sentinel/concepts/policy-as-code>)





Name	URL	Description	Meta
Open Policy agent	https://github.com/open-policy-agent/opa	General-purpose policy engine that enables unified, context-aware policy enforcement across the entire stack	<small>STARS</small> 7.6K
Kyverno	https://github.com/kyverno/kyverno	Kyverno is a policy engine designed for Kubernetes	<small>STARS</small> 3.6K
Inspec	https://github.com/inspec/inspec	Chef InSpec is an open-source testing framework for infrastructure with a human- and machine-readable language for specifying compliance, security and policy requirements.	<small>STARS</small> 2.6K







Name	URL	Description	Meta
Cloud Formation guard	https://github.com/aws-cloudformation/cloudformation-guard	Cloud Formation policy as code	 2.1K
cnspec	https://github.com/mondoohq/cnspec	cnspec is a cloud-native and powerful Policy as Code engine to assess the security and compliance of your business-critical infrastructure. cnspec finds vulnerabilities and misconfigurations on all systems in your infrastructure including: public and private cloud environments, Kubernetes clusters, containers, container registries, servers and endpoints, SaaS products, infrastructure as code, APIs, and more.	 124

Chaos engineering

Chaos Engineering is the discipline of experimenting on a system in order to build confidence in the system's capability to withstand turbulent conditions in production.







Reading and manifestos: <https://principlesofchaos.org/>

Name	URL	Description	Meta
chaos-mesh	https://github.com/chaos-mesh/chaos-mesh	It is a cloud-native Chaos Engineering platform that orchestrates chaos on Kubernetes environments	 5.9K
Chaos monkey	https://netflix.github.io/chaosmonkey/	Chaos Monkey is responsible for randomly terminating instances in production to ensure that engineers implement their services to be resilient to instance failures.	 13K
Chaos Engine	https://thalesgroup.github.io/chaos-engine/	The Chaos Engine is a tool that is designed to intermittently destroy or degrade application resources running in cloud based infrastructure. These events are designed to occur while the appropriate resources are available to resolve the issue if the platform fails to do so on it's own.	 60
chaoskube	https://github.com/linki/chaoskube	Test how your system behaves under arbitrary pod failures.	 1.6K

Name	URL	Description	Meta
Kube-Invaders	https://github.com/lucky-sideburn/KubeInvaders	Gamified chaos engineering tool for Kubernetes	 STARS 875
kube-monkey	https://github.com/asobti/kube-monkey	Gamified chaos engineering tool for Kubernetes	 STARS 2.7k
Litmus Chaos	https://litmuschaos.io/	Litmus is an end-to-end chaos engineering platform for cloud native infrastructure and applications. Litmus is designed to orchestrate and analyze chaos in their environments.	 STARS 3.6k
Gremlin	https://github.com/gremlin/gremlin-python	Chaos engineering SaaS platform with free plan and some open source libraries	 STARS 4k
AWS FIS samples	https://github.com/aws-samples/aws-fault-injection-simulator-samples	AWS Fault injection simulator samples	 STARS 2k
CloudNuke	https://github.com/gruntwork-io/cloud-nuke	CLI tool to delete all resources in an AWS account	 STARS 2.3k

Infrastructure as code security

Scanning your infrastructure when it is only code helps shift-left the security. Many tools offer in IDE scanning and providing real-time advisory do Cloud engineers.

Name	URL	Description	Meta
KICS	https://github.com/Checkmarx/kics	Checkmarx security testing opensource for IaC	 STARS 2.5k
Checkov	https://github.com/bridgecrewio/checkov	Checkov is a static code analysis tool for infrastructure-as-code	 STARS 9.4k
tfsec	https://github.com/aquasecurity/tfsec	tfsec uses static analysis of your terraform templates to spot potential security issues. Now with terraform CDK support	 STARS 5.7k
terrascan	https://github.com/accurics/terrascan	Terrascan is a static code analyzer for Infrastructure as Code	 STARS 2.9k
cfsec	https://github.com/aquasecurity/cfsec	cfsec scans CloudFormation configuration files for security issues	 STARS 57
cfn_nag	https://github.com/stelligent/cfn_nag	Looks for insecure patterns in CloudFormation	 STARS 3.1k

Name	URL	Description	Meta
Sysdig IaC scanner action	https://github.com/sysdiglabs/cloud-iac-scanner-action	Scans your repository with Sysdig IAC Scanner and report the vulnerabilities.	<small>STARS</small> 3

Orchestration

Event driven security help to drive, automate and execute tasks for security processes. The tools here and not dedicated security tools but are helping to automate and orchestrate security tasks or are part of most modern security automation frameworks or tools.

Name	URL	Description	Meta
StackStorm	https://github.com/StackStorm/st2	Platform for integration and automation across services and tools supporting event driven security	<small>STARS</small> 5.4k
Camunda	https://github.com/camunda/camunda-bpm-platform	Workflow and process automation	<small>STARS</small> 9.2k
DefectDojo	https://github.com/DefectDojo/django-DefectDojo	Security orchestration and vulnerability management platform	<small>STARS</small> 2.7k
Faraday	https://github.com/infobyte/faraday	Security suite for Security Orchestration, vulnerability management and centralized information	<small>stars</small> 3.8k

#TODO: REFACTOR UNDER THESE SECTIONS

Monitoring

Name	URL	Description	Meta
------	-----	-------------	------