RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

BACK TO BASICS

DEVSECOPS

The **DevSecOps** methodology aims to include security measures throughout the application development and production process. When using this approach, the following security best practices should be considered:

→ **Create and keep up to date a map of the applications used**, including system rights, permission and roles, installation and operating secrets, flow matrices, developers' roles (review, validation, rights on environments, etc.), referents with overall technical and business knowledge.

→ **Perform a global risk analysis**, taking into account the attack vectors of developers' workstations, subcontractors, the CI/CD* (Continuous Integration/Continuous Deployment) chain and the technologies used (e.g. cloud).

→ **Consider that actions carried out by the production CI/CD are administrative actions.** It is recommended to dedicate an administration workstation to the production CI/CD, to apply the principle of least privilege, to generate tokens on demand, as well as to log and supervise the CI/CD.

→ **Manage secrets securely.** A separate secret manager should be used for each environment (e.g. non-production, production). In addition, ensure through continuous scans that there is not secret or sensitive data leaked in the source code, spread within your log management system(s), or displayed in any code repositories.

→ **Manage dependencies rigorously** by minimizing and evaluating them, and by applying security patches before deployment.

→ **Include automated security audit and compliance checks in the CI/CD**, namely non-regression tests (to avoid new vulnerabilities), segregation between user profiles, static and dynamic analysis tests, along with IaC (Infrastructure as Code) compliance tests.

→ **Secure production deployment of applications** by maintaining end-to-end source code integrity, as well as by signing and verifying artifact version tag signatures.

→ **Implement multi-factor authentication** for accessing repositories and signing commits.

→ **Separate development and production CI/CD infrastructures. Do not expose them directly on the Internet.**

→ **Destroy and build CI/CD infrastructure regularly**. Do not store persistent data on it.

→ **Take into account the confidentiality needs** of the CI/CD infrastructure (e.g. localization, testing of source codes in public SaaS).

→ **Enforce secure development and coding rules** within teams.

→ **Apply hardening rules and policies on the operating systems hosting the deployed application** (please refer to ANSSI's Configuration recommendations of a GNU/Linux system).

(*) The CI/CD chain includes several tools, for example: orchestrator, source code repositories, automated tests, secrets manager, deployment tools, artifacts, etc.

V1.0 (02/24)

www.cyber.gouv.fr / conseil.technique@ssi.gouv.fr