HLD

stages :

- •Pré-Commit : check if the code is formatted correctly or not
- •Commit : automate static tests and analyses before allowing a commit
- •Analyse Statique : Analyze the code to detect vulnerabilities, bugs.
- •Tests Unitaires : ensure that each component functions correctly in isolation
- •Tests de Sécurité : Analyze dependencies to identify known vulnerabilities.
- •Tests Dynamique : security tests on the running application to detect vulnerabilities
- •Build and packaging : Build and package the application for deployment
- •Publier dans repository : to store, manage and distribute software artifacts
- •Build and tag image : build image with tag (latest commit of developer)

- •Pousser docker image : push this image to Docker Hub
- •Deploy : Deploy the application in the test or production environment
- •Monitoring et logging : Monitor the application and collect logs to detect performance and security problems.
- •feadback et repport : Provide feedback on test and analysis results to developers and relevant teams.

Tools :

1. **Prettier :**
It integrates with code editors, to maintain a clean and uniform codebase
2. ESLint: JavaScript
define coding rules to detect style problems and to ensure that commits respect the coding rules.
3. TSLint: TypeScript
It helps to detect errors and enforce code style conventions.
4. Commitizen
Commitizen helps developers write commit messages that conform to a standardized format**.**
5. SonarQube :
to analyze the code after each commit or pull request, providing detailed reports on the quality of the code
6. Selenium :
Used for end-to-end testing of web applications in development and test environments.
7. Jest:

Used mainly for testing JavaScript and React applications,

8. Junit:

Used to test Java applications

9. Snyk :

to scan dependencies, source code and containers for security vulnerabilities.

10. Dependency-Check:

to check the dependencies and make sure that they do not contain known vulnerabilities.

12. OWASP ZAP :

Used in test and pre-production environments to identify security vulnerabilities in web applications.

13. Maven:

Used to build and manage Java projects

14. Nexus:

manage and distribute software artifacts such as libraries, packages and containers**.**

15. Xray:

Used to scan artifacts in repositories for vulnerabilities and ensure their compliance before deployment.

16. Docker:

Used to encapsulate applications and their dependencies in containers

17. **Docker Hub:**

Used to host and distribute Docker images

18. Kubernetes :

is a container orchestration platform that automates the deployment

19. Trivy:

to scan Docker images and Kubernetes configurations before deployment.

20. ELK Stack (Elasticsearch, Logstash, Kibana)

Used for the collection, analysis and visualization of logs and metrics of applications and infrastructure.

21. Grafana:

is a platform for monitoring and visualizing metrics

22. Prometheus:

Used to monitor the performance and availability of applications and infrastructure, often in combination with Grafana for visualization.

Differences Between Tools at Each Stage

Pre-commit and Commit:

- **Prettier and ESLint**: Focus on code formatting and quality.
- **Husky and Commitizen**: Manage Git hooks and commit messages to ensure compliance with standards.

Static Analysis:

- **SonarQube**: Provides comprehensive source code analysis.
- **Fortify**: Focuses on detecting vulnerabilities in the code.
- **Veracode**: Combines static and dynamic analysis for broader security coverage.

Unit Testing:

- **Jest**: Primarily used for unit testing in JavaScript.
- **JUnit**: The standard testing framework for Java applications.

Security Testing:

- **OWASP ZAP**: Focuses on web application security testing.
- **Burp Suite**: Offers advanced features for web application security testing.

Dynamic Testing:

- **OWASP ZAP**: Performs dynamic security testing.
- **Selenium**: Used for automating user interface tests.

Build Package:

- **Maven and Gradle**: Popular build tools for Java projects, with Gradle offering additional flexibility.

Publish to Repository:

- **Nexus and JFrog Artifactory**: Repositories for managing and distributing binary artifacts, with JFrog Artifactory providing universal artifact management.

Build and Tag Image:

- **Docker**: The standard tool for creating and managing containers.
- **Buildah**: Allows for creating container images without requiring a container daemon.

Scan Docker Image:

- **Trivy, Clair, and Anchore**: All provide vulnerability scanning capabilities for Docker images, with varying features for detecting security issues.

Push Docker Images:

- **Docker Hub**: The most widely used Docker image registry.
- **GitHub Container Registry**: Integrates container image registry capabilities into GitHub.

Deployment:

- **Kubernetes**: The most widely used container orchestrator.
- **Helm**: Simplifies the management of Kubernetes applications.
- **OpenShift**: Adds additional management and security features on top of Kubernetes.

Monitoring and Logging:

- **Prometheus**: Used for metrics monitoring.
- **Grafana**: Provides advanced metrics visualization capabilities.
- **ELK Stack**: A comprehensive solution for log management.

Feedback and Reporting:

- **Jira**: Used for project management and task tracking.
- **Slack**: Facilitates team communication.
- **UpGuard**: Provides cybersecurity risk management and risk score analysis.

Code Security Tools

1. SonarQube / SonarCloud
2. Source Guard
3. Shiftleft Scan
4. checkmarx
5. Veracode Greenlight

Build Security Tools

1. Burp Suite
2. Zed Attack Proxy (ZAP)
3. ModSecurity
4. WhiteSource Bolt
5. Skipfish
6. Veracode SourceClear

Code Security Tools

1. Yelp
2. CredScan
3. Changeme
4. Secret-code-scanner
5. Veracode Greenlight

Artifactory Security Tools

1. Jfrog Xray
2. Kroll Parser
3. Archiva
4. Aqua
5. Anchore

SCA Security Tools

1. Qualys

2. Snyk
3. WhiteSource
4. Veracode
5. CheckMarx

Container Security Tools

1. Aqua Security Tools
2. Anchore Container security
3. Whitesource
4. Twistlock
5. Qualis
6. Clair

Penetration Testing Tools

1. Qualys
2. Snyk
3. WhiteSource
4. Veracode

Threat Modelling Tools

1. OWASP Threat Dragon
2. Microsoft Threat Modelling Tool 2016.
3. Threat Modeler
4. Raindance
5. Threatspec
6. PyTM

Website Vulnerability Tools

1. URL Freezer
2. SQLi Scanner
3. XSS Scanner
4. Drupal
5. Joomla