Secure Container Journey

Accelera

DESIGN

1 Cloud Native & secure code training

Engineers gain knowledge and awareness of Cloud Native and AppSec principles and responsibilities.

2 Secure by Design

Gather threat and abuse case models, and security requirements. Adopt reusable Secure-by-default design patterns. Identify Cloud Native deployment model and runtime policies. Branching strategy that allows for GitFlow/GitOps, Code reviews, tooling integration in pull requests.

3 Secure access to app. code & image repository

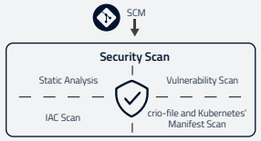
Implement least privilege RBAC and n/w isolation to code and private container repository. Infrastructure as code version controlled. Private registries white-listed in orchestration systems and deployment prevented from Docker Hub etc. Registry staging implemented.

DEVELOP

SPRINT

6 Pre-Commit analysis

Automated analysis of all code and image assembly files for vulnerabilities, secret scanning (client and server). Prevent commit if fails.



5 Dev code analysis

Developer analyses all app and IaC code for weaknesses and composition analysis. IDE integrated plugins provide remediation advice.

4 Secure access to CI service

Prevent unauthorised manipulation of the pipeline itself with RBAC and separation of duties.

SPRINT

DISTRIBUTE

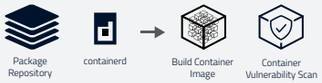
SPRINT

7 Security test

Application, container and configuration security testing, scanning for standards compliance and best practice.

8 Build image and push to registry

Packaging container images from access-restricted build platform. Vulnerability scan the image before pushing to private access controlled container registry.



9 Promote to staging

Dev compliant containerised application promoted to staging.

10 User Acceptance Test

Verify application behaviour, performance, system security testing and fit for purpose.

DEPLOY

SPRINT

13 Promote to production

Compliant containerised application promoted to production deployment.

12 Runtime compliance

Kubernetes Dynamic Admission control for prevention of non compliant deployment.

11 Preflight check

- Image signing and encryption
- Image runtime policies
- Container runtime policies
- Host vulnerability and compliance controls
- File integrity
- Process integrity
- Syscalls
- Workload, application and network security policies

SPRINT

RUNTIME

14 Policy enforcement

- Privileged container runtime enforcement
- Admission Controllers
- RBAC
- Security Policies

15 Runtime resource constraint

Linux kernel cgroup isolation providing object level resource request limits.

16 Continuous monitoring

Analyse intrusion / breach alerts and threat intelligence. Log attacks, behaviours and threats. Observability, SIEM, Reporting, Incident Management.

17 Audit and repeat

Continuous assessment and feedback loop. Improve and enforce governance.

GATES

Automated

Sign-off

www.accelera.com.au/ContainerSecurity

v1.0 - April 2021