MODUSBOX

Security Program (OSS)


Why do we have brakes on a car?

ICT SECURITY AS AN ENABLER FOR STRATEGY.

Information and communication technology (ICT) is a critical business enabler for business strategy. It enables greater efficiency, a more agile workforce and the ability to do business regardless of time or geographic boundaries.

We support Mojaloop via designing, reviewing and optimizing security controls surrounding business and technical operations.

Why talk security?

1. Hackers target to inject malicious code early in the pipeline. This way all code users are potential victims.
2. Increased need for compliance
3. Protection from reputation loss and Customer assurance – it is obviously easier to sell a secure product.
4. Cost reduction is achieved by detecting and fixing security issues during the development phases which also increases the speed of delivery.
5. ‘Secure by design’ principle is ensured by using automated security review of code, automated application and package security testing, educating, and empowering developers to use secure design patterns.

Trends - State of DevOps 2019

DevSecOps = better visibility: Developers, operations team members, and security professionals are 89% more likely to have good insight into what their colleagues are working on when their DevOps model has been in place long term
Security is also a work in progress: 50% agree that security vulnerabilities are mostly discovered by the security team after code is merged and in a test environment.
Testing is still hard: 49% of respondents encounter the most delays during the testing stage of the development lifecycle.
Security in DevOps: Sec teams are 3x more likely to discover bugs before code is merged with a good DevOps practice in place and they are 90% more likely to test between 91% and 100% of code than in an organization with early stage DevOps.
Developers aren't always on board: Nearly half of security pros surveyed (49%) said they struggle to get developers to make remediation of vulnerabilities a priority. **Why?**
Dependency scanning is the most popular at 56%, followed by cloud security (42%), container security (41%), SAST (35 %), license compliance (29%t) and DAST at 22%. All told, 12% of security teams test between 61-75% of code.

Source - 2019 GitLab Developer Report : DevSecOps

Evolution of Security Tools

Duration 2-4 weeks → 1-2 weeks → Continuous and Real-time

Penetration Testing

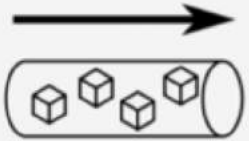
Tools
- Port Scanners
- Vulnerability Scanners
- Exploitation Tools

Audience
- Security Professionals

Secure SDLC

Tools
- Code Security Scanners
- Dynamic Security Scanners
- Vulnerability Scanners

Audience
- Security Professionals in Enterprise Security Teams

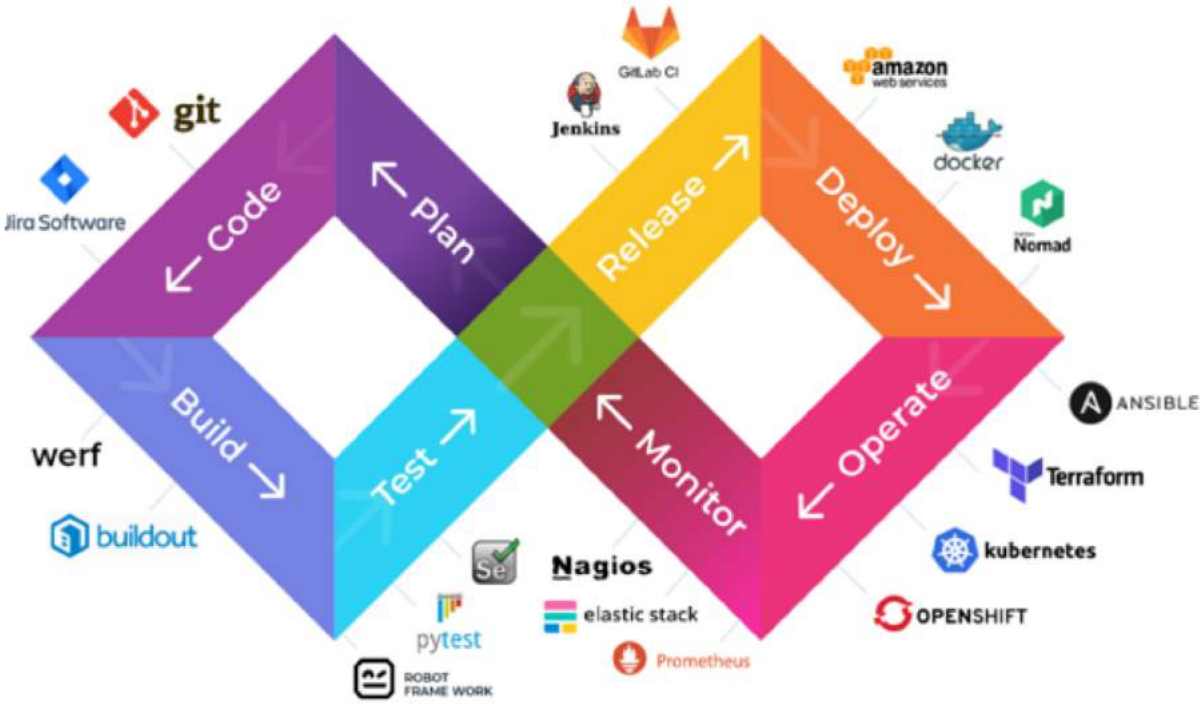


DevSecOps

Tools
- Code Security Scanners
- Interactive Security Scanners
- Runtime Application Self Protection

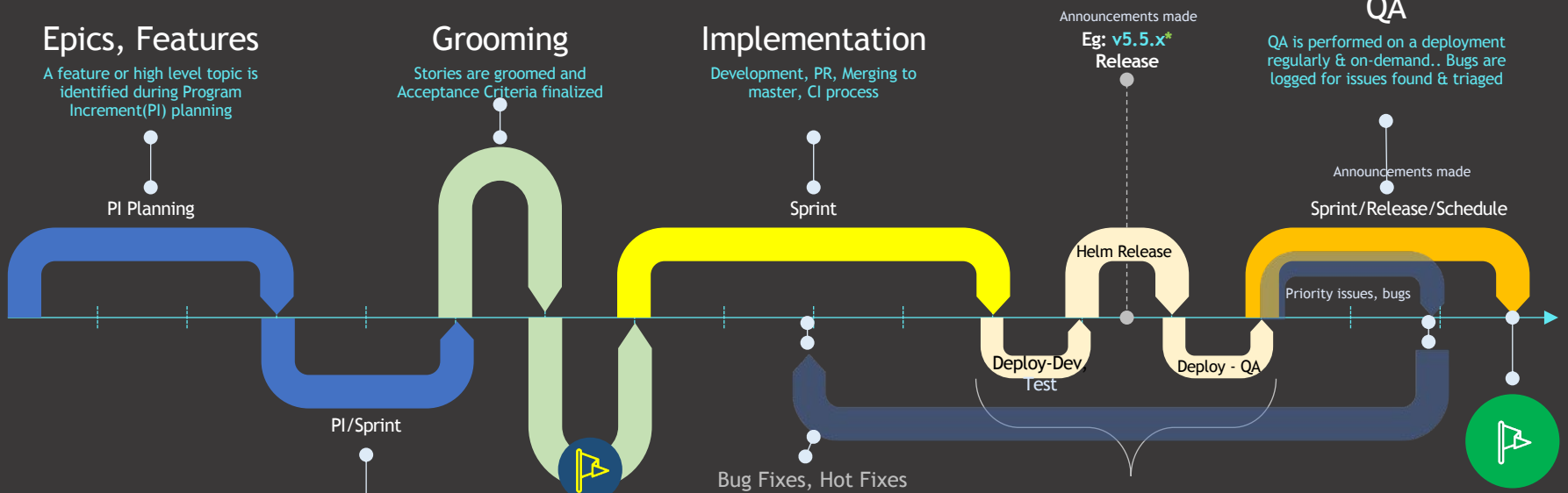
Audience
- Developers in Product Teams

Typical DevOps Cycle



Know thyself - ML OSS: Release Mechanism

REFERENCE



Note: This depicts a single iteration that happens multiple times over the course of a PI

* Inspired by SemVer; x.y.z – where 'x' is the PI number, 'x.y' is the Sprint number and 'z' is the number of release in a particular sprint

PI-6 April 2019

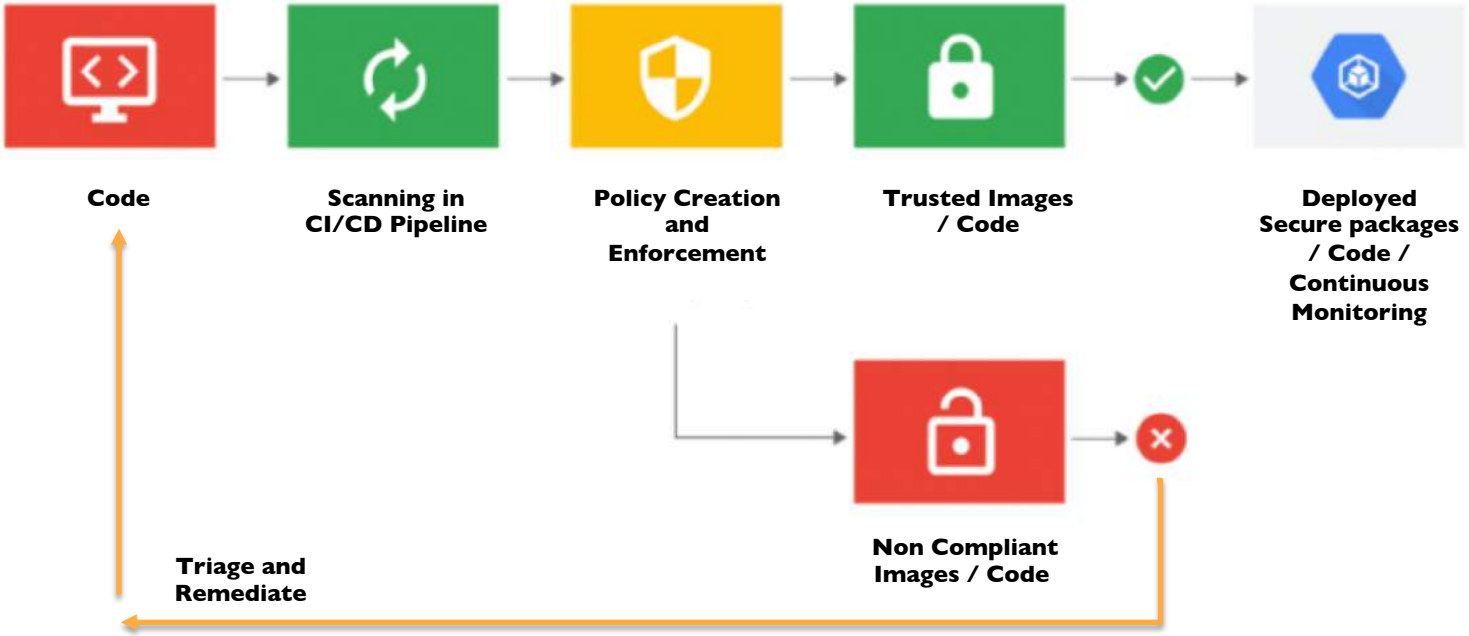
Where do we come in?

DevSecOps in Mojaloop OSS

Mojaloop DevSecOps will ensure early identification of security gaps early in the development process, ensuring the triage and immediate removal of the root cause through continuous monitoring, assessment and analysis.

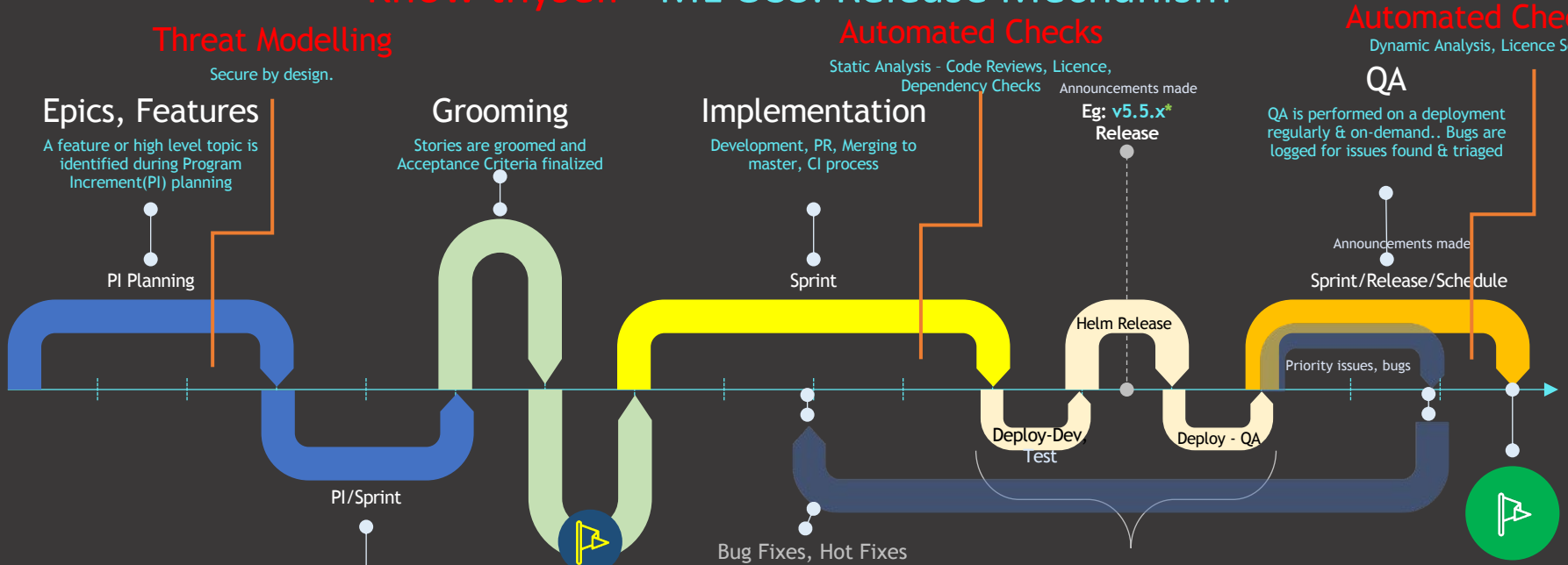
DevSecOps = DevOps + Security Component

DevSecOps – High Level Process



Know thyself - ML OSS: Release Mechanism

REFERENCE



Note: This depicts a single iteration that happens multiple times over the course of a PI

* Inspired by SemVer; x.y.z – where 'x' is the PI number, 'x.y' is the Sprint number and 'z' is the number of release in a particular sprint

PI-6 April 2019

Key Security Controls and Compliance Processes

GitHub Security (Protect Repositories)

Threat Modeling (STRIDE Model) – incorporate security principles and guidelines during design process and QA Testing

Static Analysis - Code Reviews (SAST)

Dynamic Application Testing (DAST)

- Vulnerability (VA) Scans and Assessments
- Dependency Scanning
- Container Security and Policy Compliance
- Penetration Testing

Awareness

Continuous Monitoring and Response (Operational and Security) - Post deployment

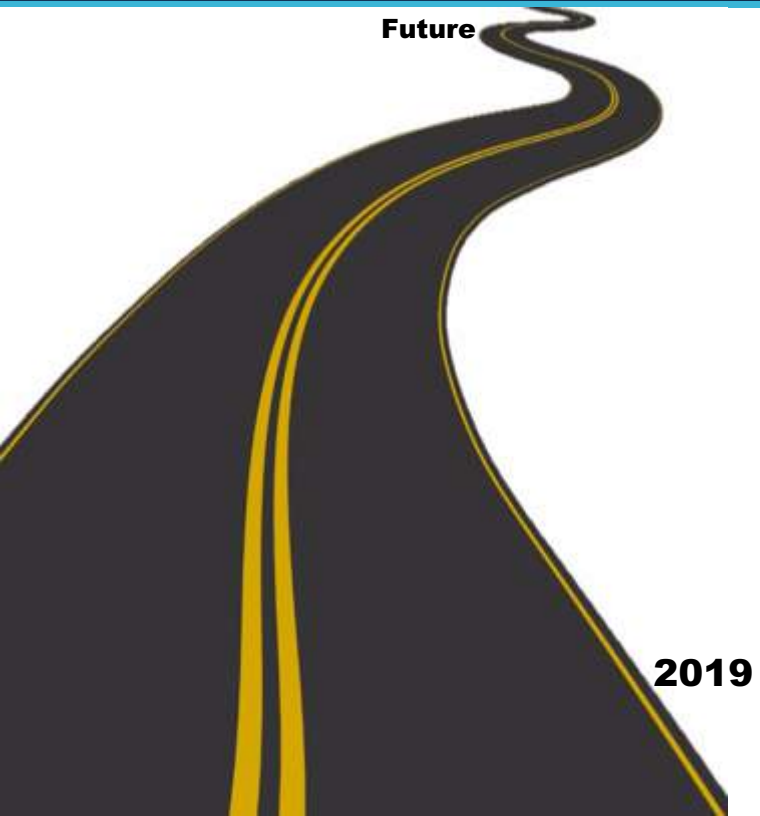
Some Questions to ask ourselves in CICD pipeline operations

MONITOR	TEST	PLAN & BUILD
1. Are our applications currently under attack? 2. What resources / services are attackers going after? 3. Are we able to automatically respond against such attacks?	1. Do our latest Changes introduce new security gaps? 2. Do any 3rd party apps and dependencies in use have known security issues? 3. Does our code contain hard coded secrets e.g. passwords, keys, tokens? 4. Are our deployments compliant against local laws, industry standards and regulations such as GDPR?	1. Do the teams know about latest successful cyber attacks in the industry? 2. Do the teams know incorporate risk mitigation in solution design? 3. Who is the dedicated security contact in the team?

OSS Controls in Place (some)

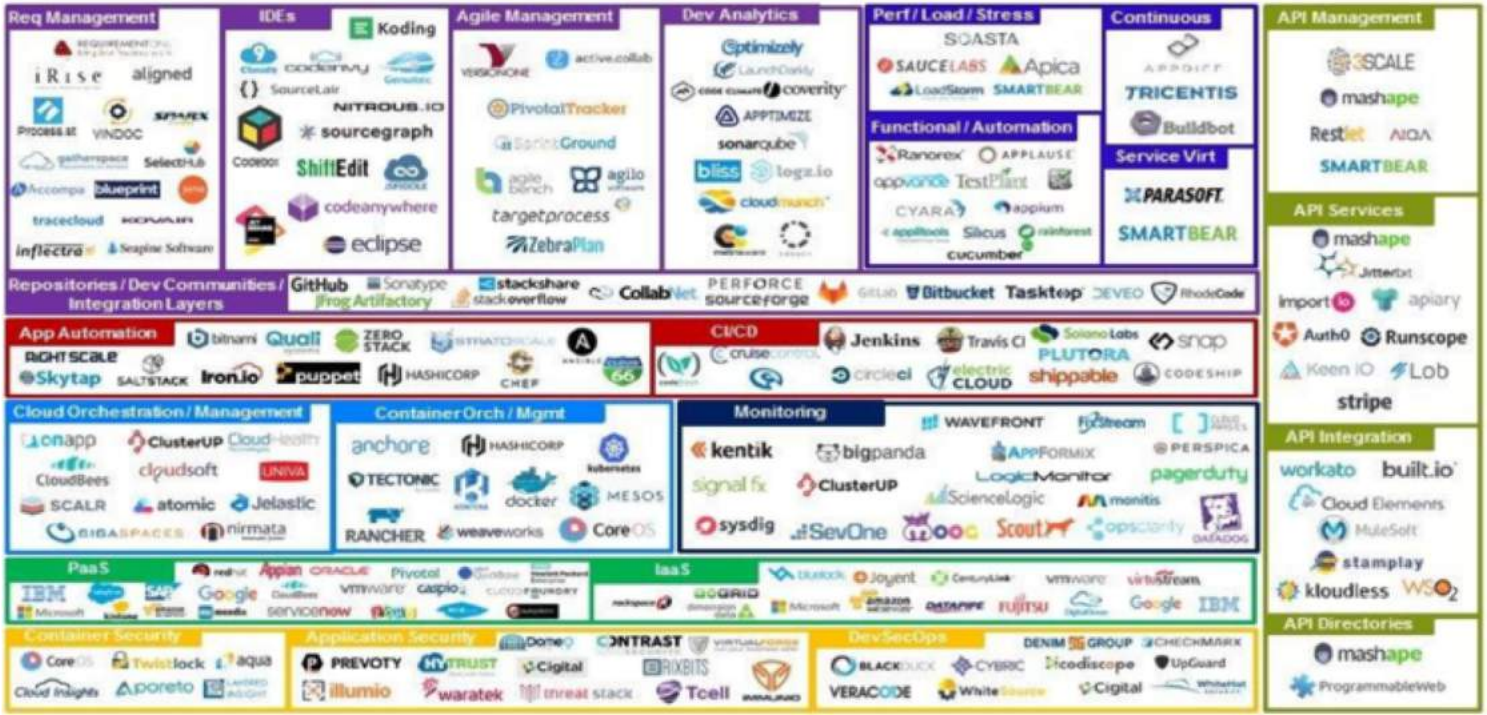
1. GitHub Security Alerts - When GitHub discovers or is notified of a new vulnerability, GitHub identify public repositories (and private repositories that have opted in to vulnerability detection) that use the affected version of the dependency, send a security alert to repository maintainers (admins), and generate an automated security fix.
2. Dependency Licence Scans – Check dependencies for license compliance.
3. Dependabot Activation - Dependabot pulls down your dependency files and looks for any outdated or insecure requirements. It then automatically installs updates to affected components.
4. GitHub Access Control – Restricted access to administration sections on GitHub.
5. Release management processes and change control.

OSS Security Roadmap

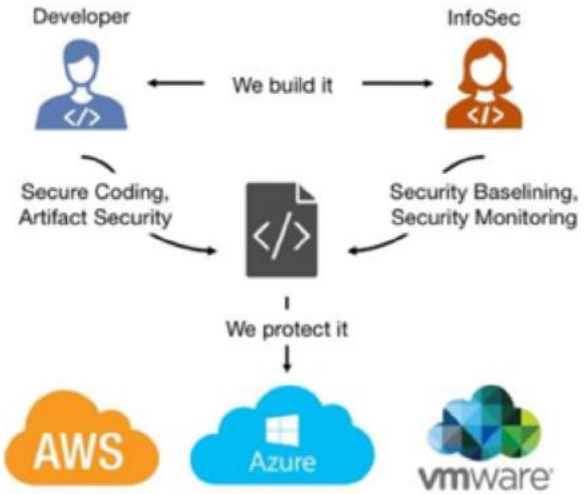


1. CI/CD Automated scanning for code and packages (tooling)
2. Document Mojaloop Security Architecture
3. Security Bug Triage and management (Operational Processes)
4. Security Best Practices documentation for OSS Code
5. Container security and compliance
6. Data Security Compliance

Finally - Some of the Tools



Questions / Feedback?



Thank You.