RSAConference2020

San Francisco | February 24 – 28 | Moscone Center

HUMAN ELEMENT

SESSION ID: **DSO-R07**

How to harness Dev and their native tools to accelerate DevSecOps

Cindy Blake

Sr Security Evangelist
GitLab
@cblake2000

Problem: Even Basic Application Security Testing is Hard

- Applications are a prime target of cyber attacks
- Lack of hygiene allows proven exploits to be reused
- App Sec tools are expensive and require integration of both technology and processes
- To shift left, workflows must target both dev and sec teams
- Security and developer teams lack the means to collaborate and scale across silos

RSAConference2020

Lead, follow or get out of the way!

There will always be more of them then there are of you!

3 Shifts in Software That Will Impact Security

1. **How software is composed and executed**
   a. Open Source
   b. Cloud Native and serverless
   c. Dynamic environments
2. **How software is delivered and managed**
   a. Iterative MVC, agile
   b. Policy-driven automation
   c. Everything as-code
3. **How software complies with regulatory requirements**
   a. Beyond application security testing
   b. Supply chain and SDLC integrity, auditability

Cloud Native? THENEWSTACK

1. Packaged as lightweight **containers**
2. Developed with best-of-breed languages and **frameworks**
3. Designed as loosely coupled **microservices**
4. Everything is an **API** to connect microservices
5. Architected with a clean separation of **stateless and stateful** services
6. **Abstracted** from server and operating system dependencies
7. Deployed on self-service, elastic, **cloud** infrastructure
8. Managed through agile **DevOps processes**
9. **Automated** capabilities
10. Defined, **policy-driven** resource allocation

New Attack Surfaces of Cloud Native

Three main building blocks of cloud native architecture

Containers

Hold a cloud native application's libraries and processes. They share one operating system. They make the applications portable.

Orchestrators

Direct how and where containers run.

Microservices

Apps are broken down into smaller parts, or microservices, to make them easier to scale based on load.

Is Security a Square Peg in a Round Hole of DevOps?

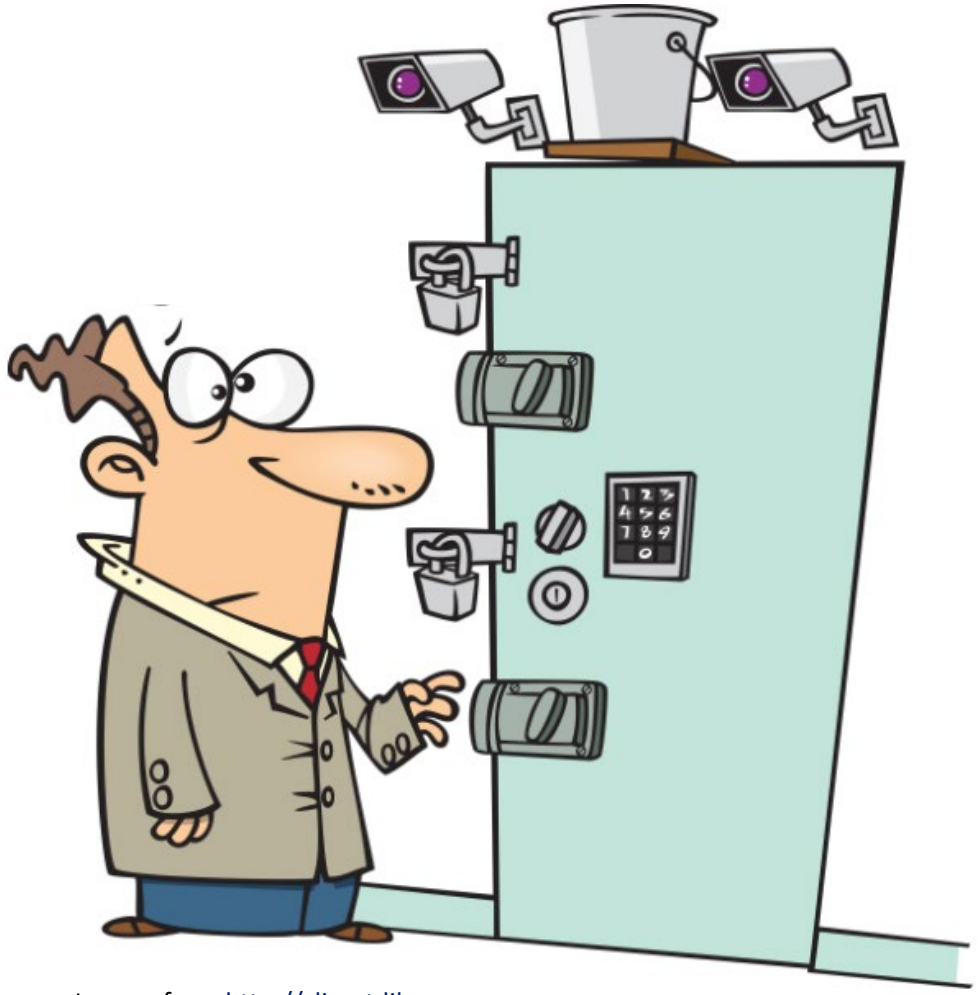
Established security tools were intended for a waterfall process at the end of SDLC and are incongruent with DevOps's

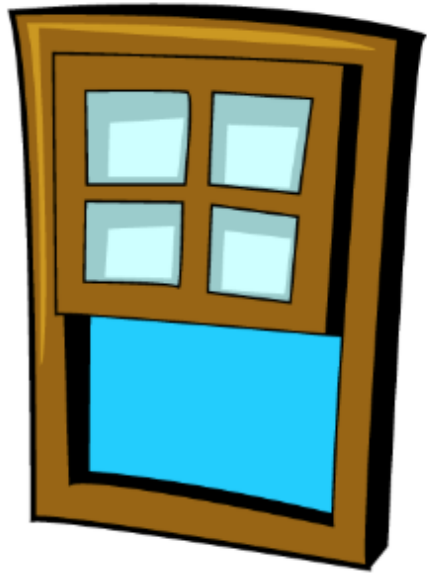
- People
- Process
- Technology

Problem: Modern Software vs Legacy App Sec

- New attack surfaces of composable application infrastructure make network security less relevant.
- The iterative development process (Agile/MVC) is incongruent with full app security scans
- Code changes faster and faster, with more open source , more APIs, and microservices (mini apps)
- DevSecOps doesn’t scale without developer enablement, automation, and exception-based security

Do You Solve for the Obvious Threats?





Images from http://clipart-library.com

RSAConference2020

“Your most important security product won’t be a security product .”

CISO of VMWare

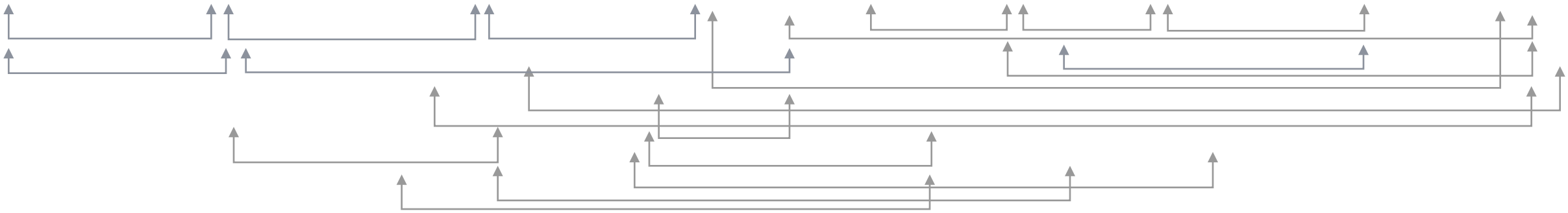
Git What?

Know your Git... and why it matters

Git
GitLab
GitHub

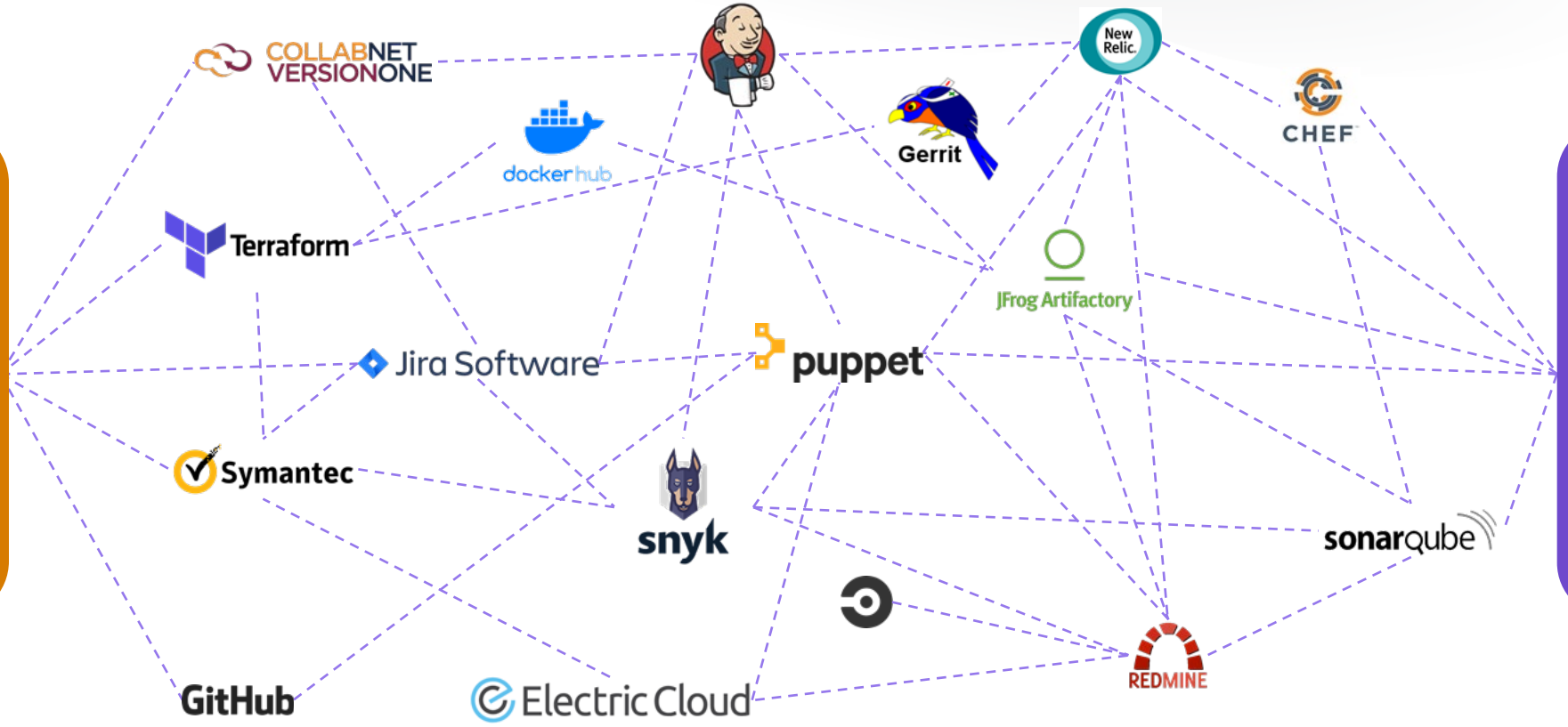
Integration Complexity of Toolchains Slows Down Teams

Manage	Plan	Create	Verify	Package	Secure	Release	Configure	Monitor	Defend
COLLABNET VERSIONONE	JIRA	GitHub	Jenkins	JFrog Artifactory	snyk		p	New Relic.	Symantec.
VERSIONONE	REDMINE flexible project management	Atlassian Bitbucket	Travis CI	Sonatype Nexus	sonarqube	XL DEPLOY	CHEF	Nagios®	FireEye



DevOps Complexity Inhibits Auditability… and Introduces More Security Risk

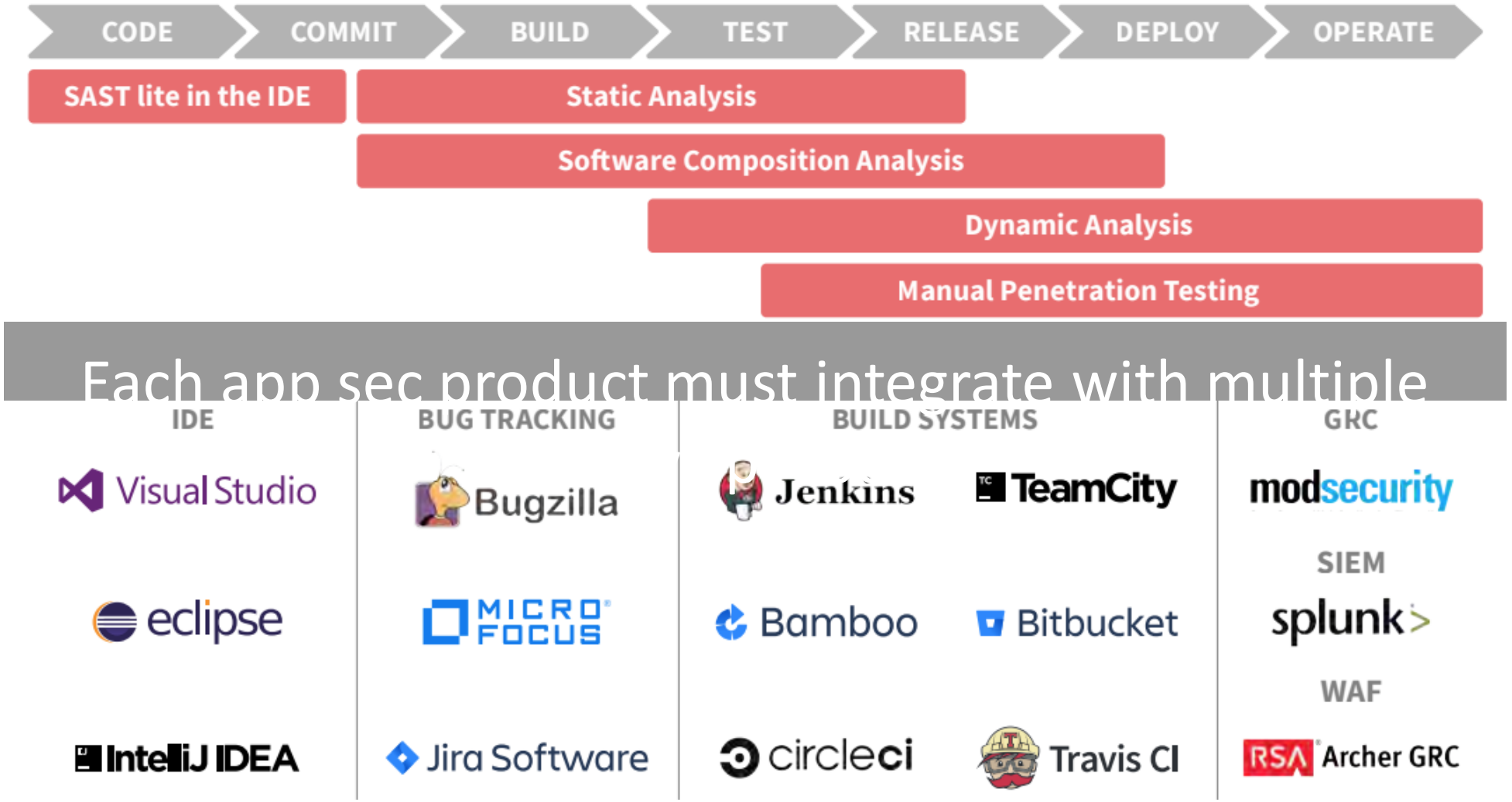
Business Problems



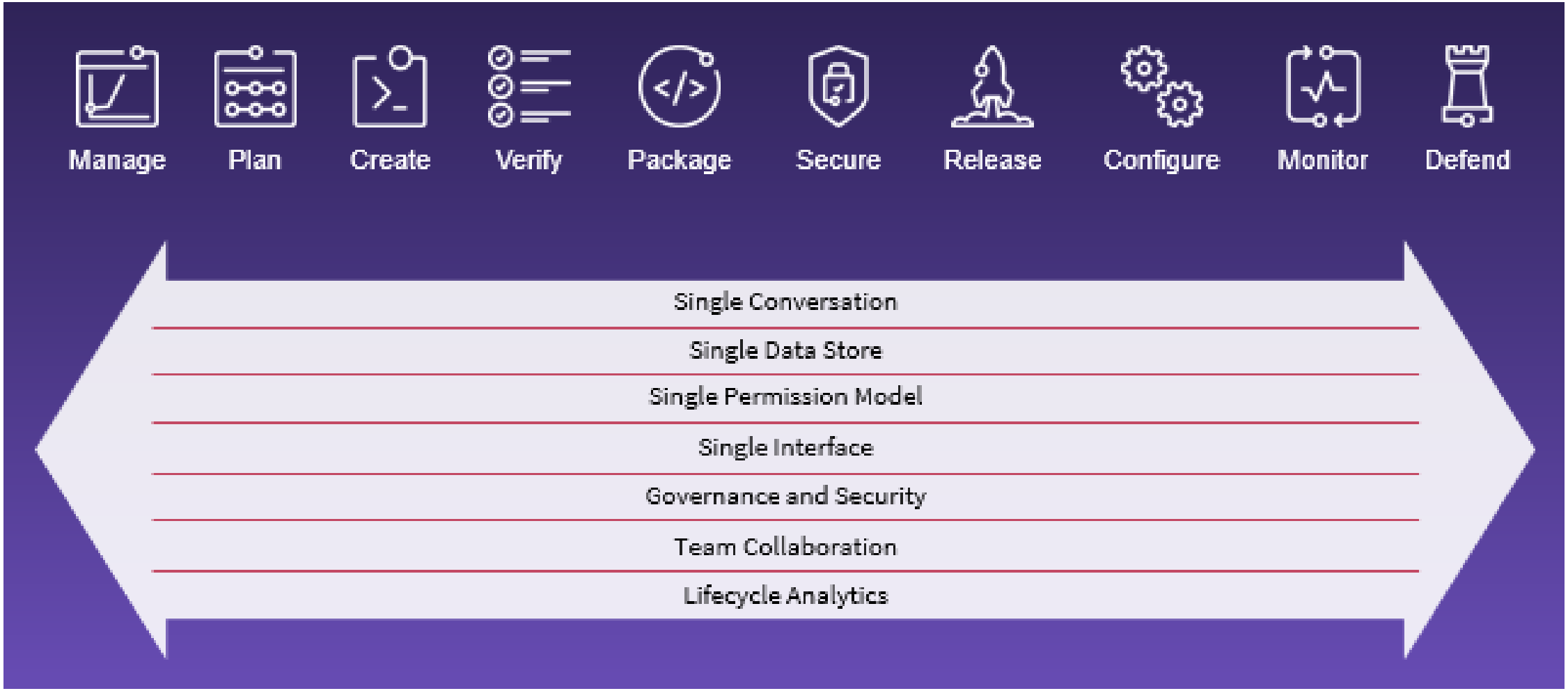
Solution Deploy-ment

- X 100s of Tools
- X Multiple Data Models
- X Complexity & Risk
- X Lack of Transparency

App Sec Silos Compound a Web of DevOps Integrations



The advantage of a Single Application for DevOps



Continuous Application Security – Embedded into CI

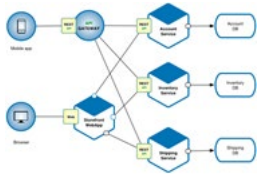
Manage | Plan | Create | Verify | Package | Secure | Release | Configure | Monitor | Defend

01100
10110
11110

Static Application Security Testing (SAST)

Container Scanning

Dependency Scanning



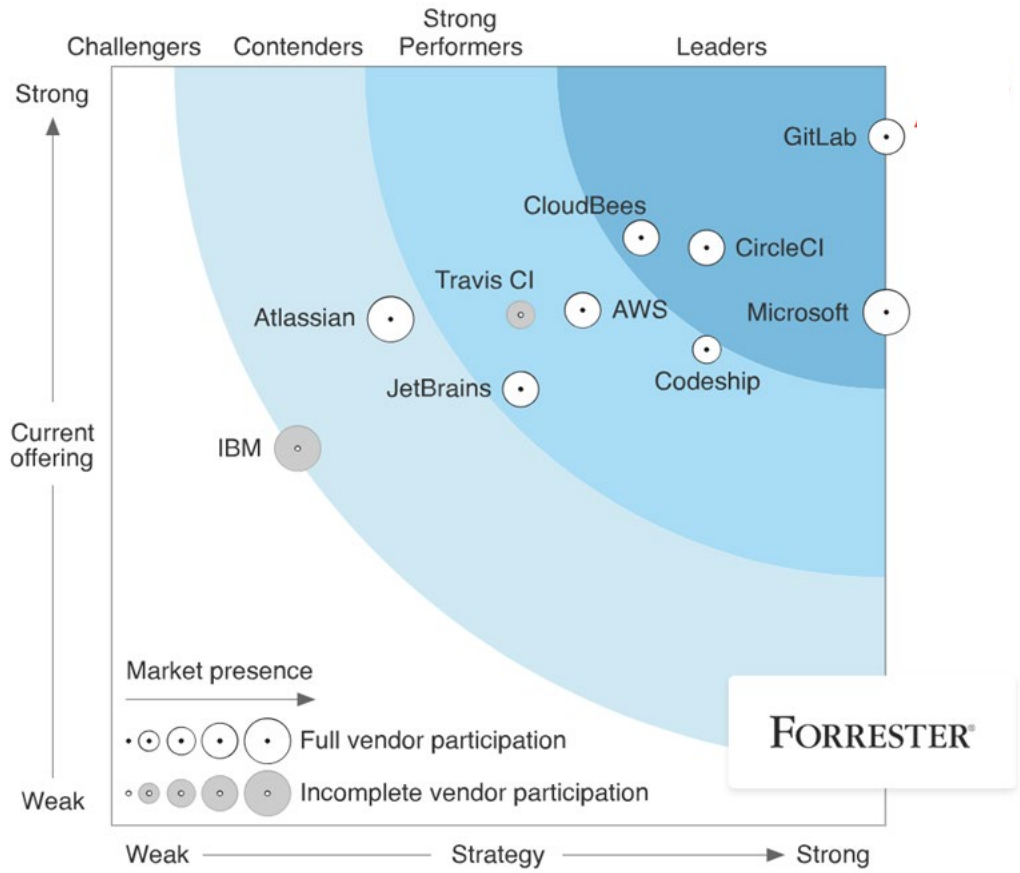
Dynamic Application Security Testing (DAST)

License Compliance

GitLab

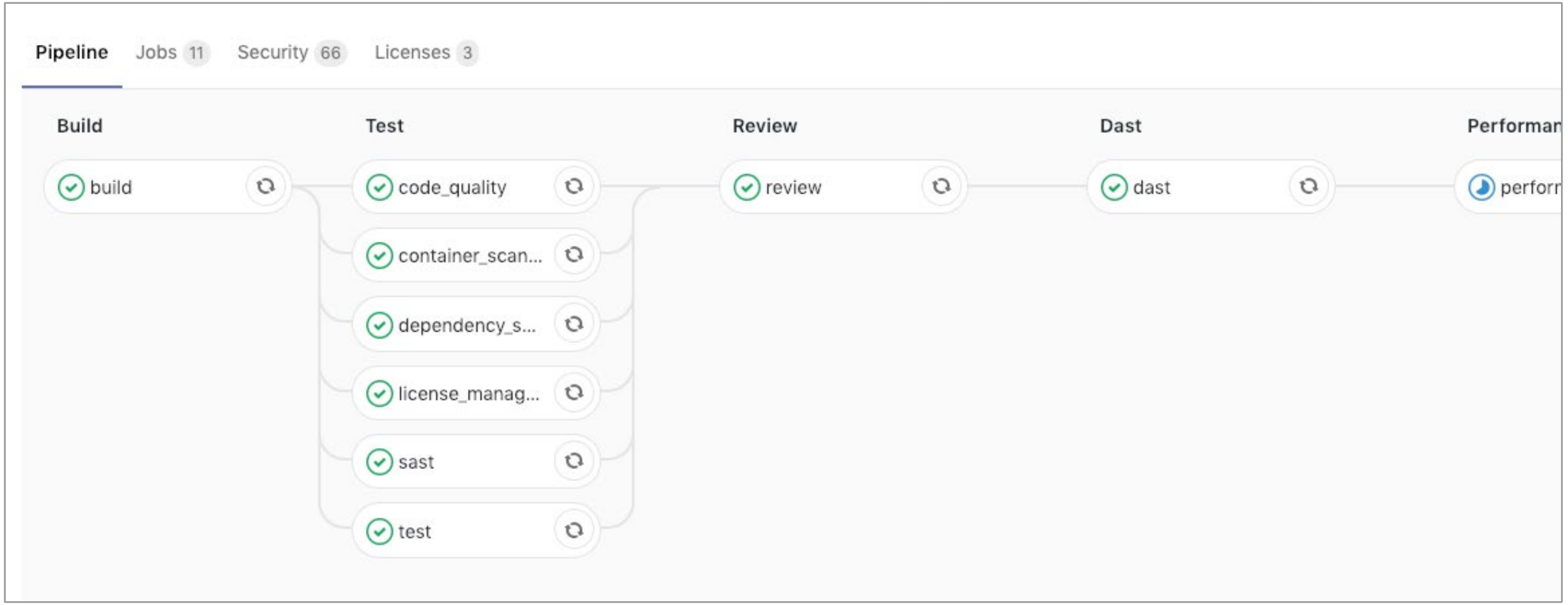
Security Testing Built into Merge Requests

Focus on leaders in Continuous Integration (CI)



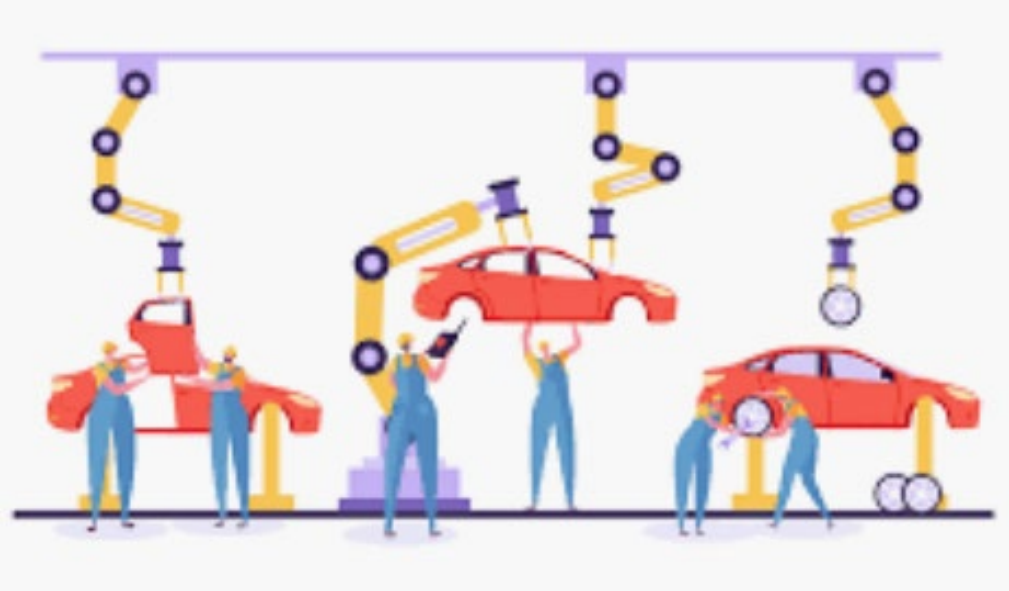
Leader in the Forrester CI Tools Wave™

Continuous Application Security = a United Workflow



What If You Could…

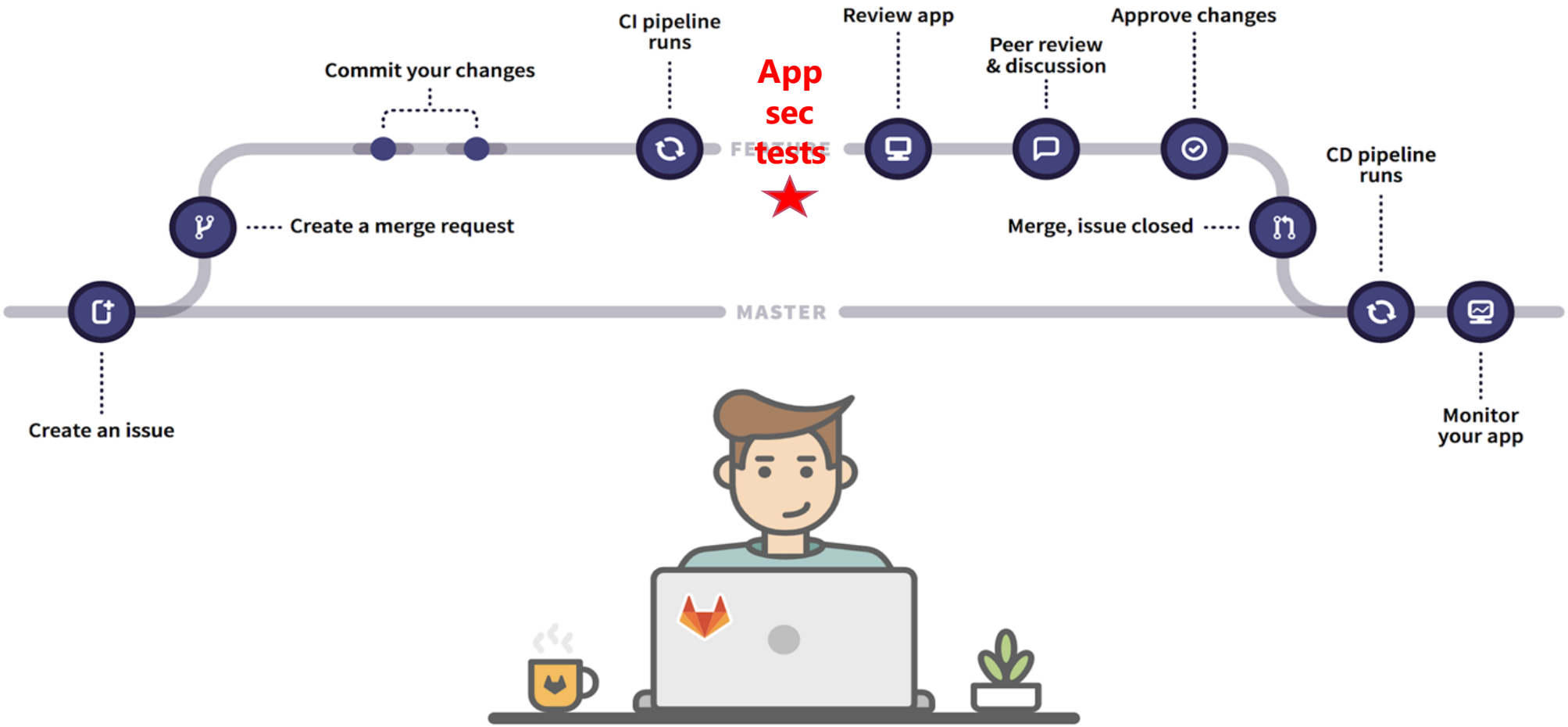
A Software Factory Approach Also Reduces Risk



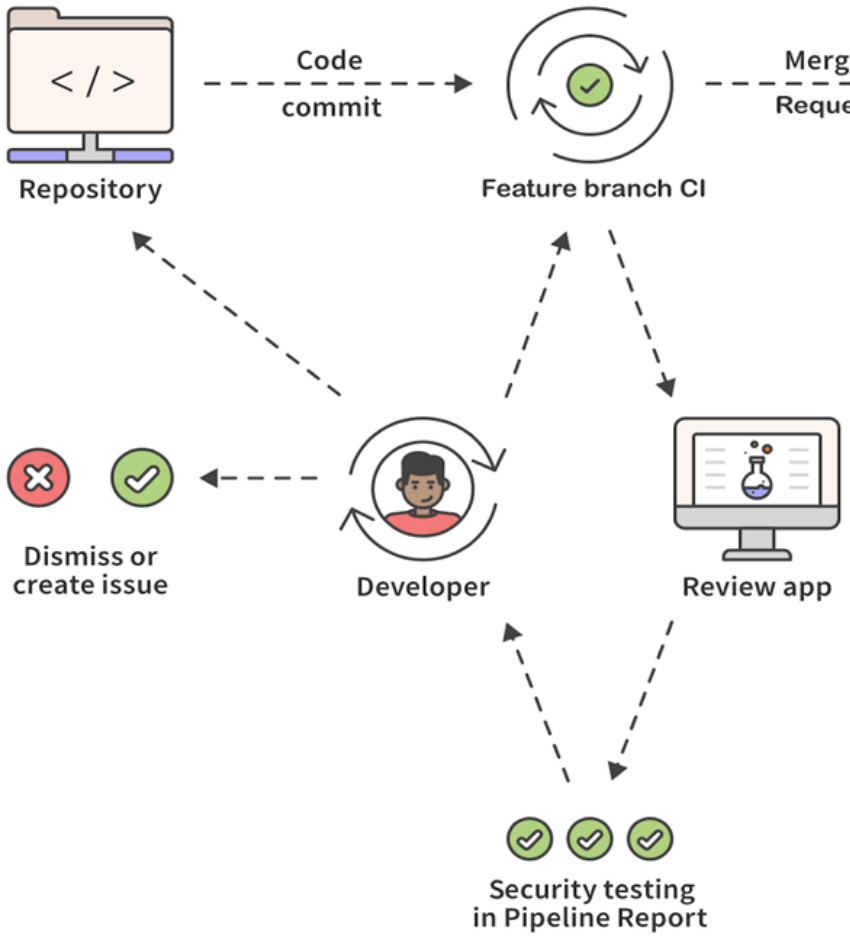
What if you dealt with each one at the point where it is introduced?

What happens when you find 10k vulnerabilities at the end of the SDLC?

Seamlessly Test for Vulnerabilities within the Developer Workflow



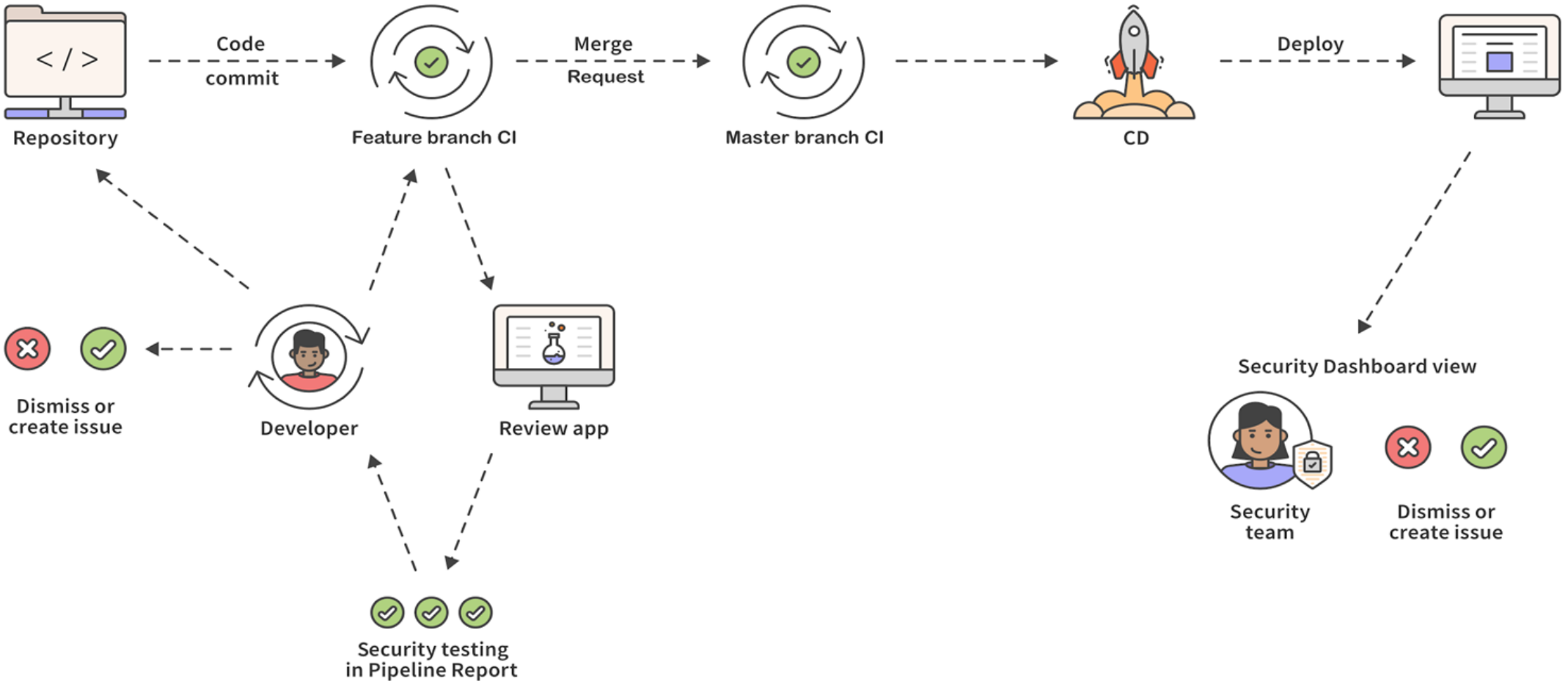
Security Scanning as Iterative as Your Development



Focus on the point of code commit

- Actionable
- Accountable
- Iterative
- Contextual

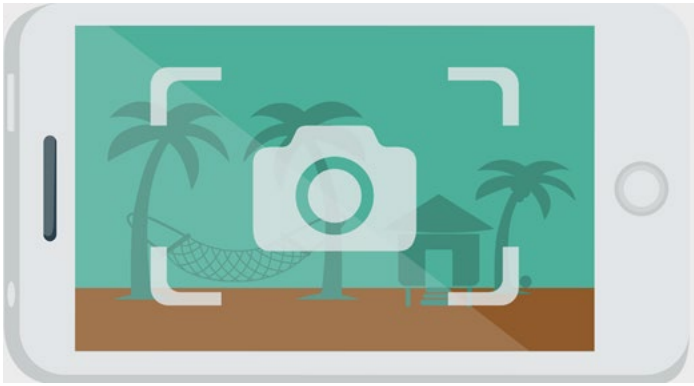
Automate, allowing security to focus on exceptions



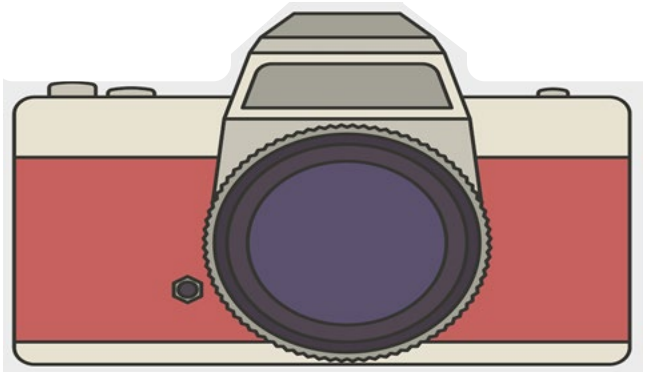
Why It Works!

- Contextual
  - Within CI/CD dev workflow - accountable person
  - MR pipeline for dev
  - Security dashboard for Security
- Congruent with DevOps processes
  - Iterative within dev, tests every code change
  - Immediate cause/effect of code changes
- Integrated with DevOps tools
  - Create issues
  - Auto remediation
  - Production feedback
- Efficient and automated
  - Eliminate work wherever possible
  - No context-switching
  - Less tracking/triaging and more value-added security

Simplicity and Integration Wins!



VS



RSAConference2020

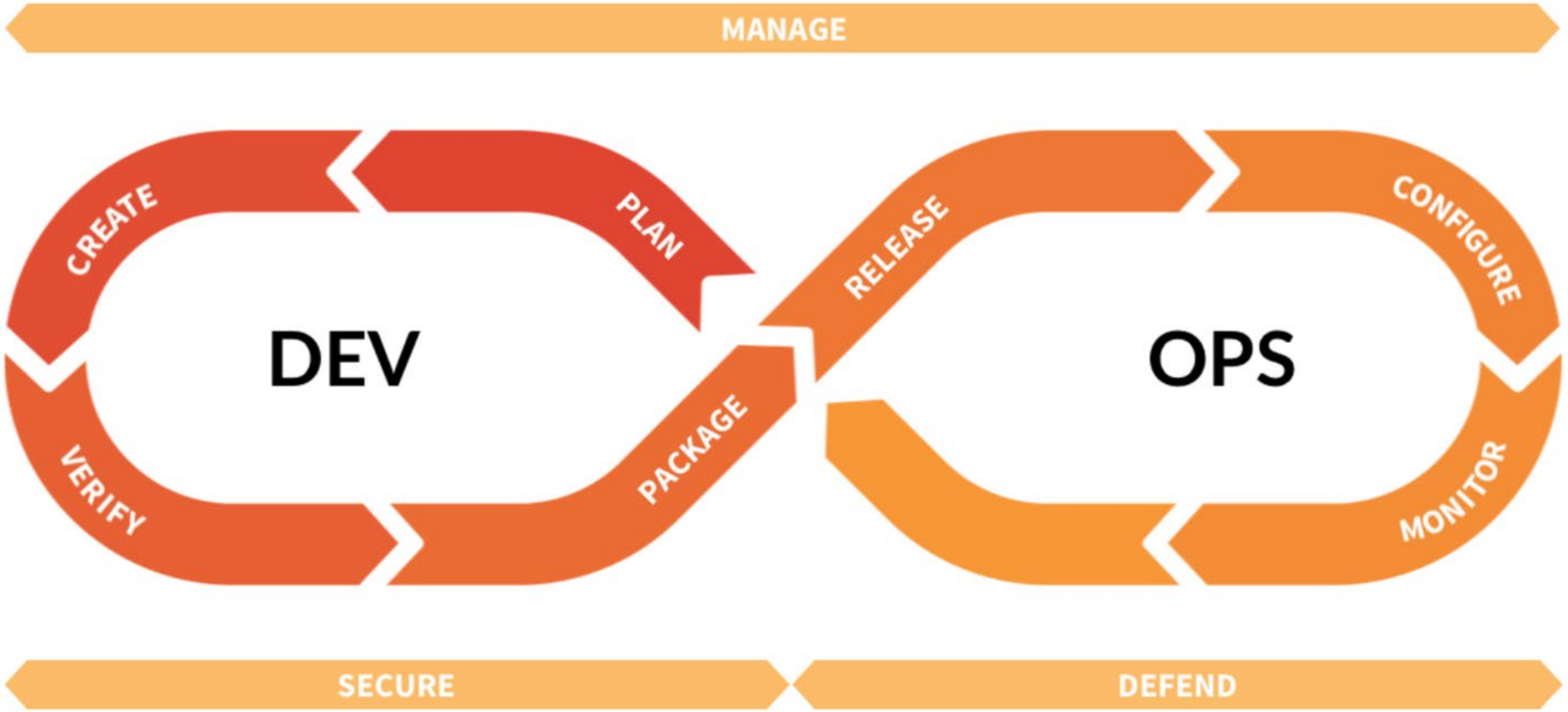
Demo

Secure the Integrity of the Application End-to-End

End-to-end application security needs to automate:

- Application security testing and remediation
- Production Application Protection
- Policy Compliance and Auditability
- SDLC Platform Security

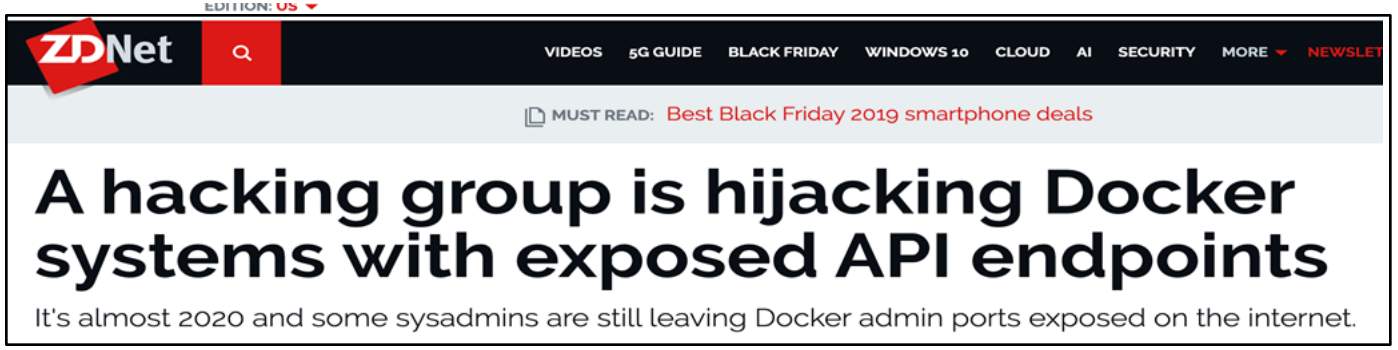
Secure and Defend Your Apps and Their Infrastructure



New Attack Surfaces Will Require New Approaches

Major Docker vulnerability disclosed in January

https://neuvector.com/docker-security/runc-docker-vulnerability/



https://redlock.io/blog/cryptojacking-tesla

Kubernetes console entry to AWS S3 storage bucket to sensitive data

Access to the company's AWS server by exploiting a misconfiguration

GitLab

RSAConference2020

Securing Next Gen Apps in Production

Additional Resources

https://lnkd.in/er8tjQg

Apply What You Have Learned Today

- Next week you should:
  - Identify Source Code Manager and CI tools used within your organization
  - Assess use of Cloud Native and Serverless tools and techniques
- In the first three months following this presentation you should:
  - Understand who is configuring cloud, containers, orchestrators
  - Define appropriate controls for these and determine gaps
  - Conduct POC for embedding security scans into CI
- Within six months you should:
  - Define tools and processes that will automate controls defined above
  - Drive an implementation project to ramp DevOps projects

RSA Conference 2020

Want to chat more?

cblake@gitlab.com

Or linkedin/in/cblake2000

Thank you!