RSAConference2018

San Francisco | April 16 – 20 | Moscone Center

NOW MATTERS

SESSION ID:

AUTOMATING SECDEVOPS WORKSHOP

Murray Goldschmidt

Chief Operating Officer
Sense of Security Pty Ltd

- Sense of Security is a leading, independent, privately owned consulting practice, founded in 2002.
- We’re celebrating our 15th birthday this year as a business.
- At SOS we are relentless at achieving positive security outcomes for all our clients.
- We do that through our hard work, knowledge, and skills that we constantly keep improving.

Agenda

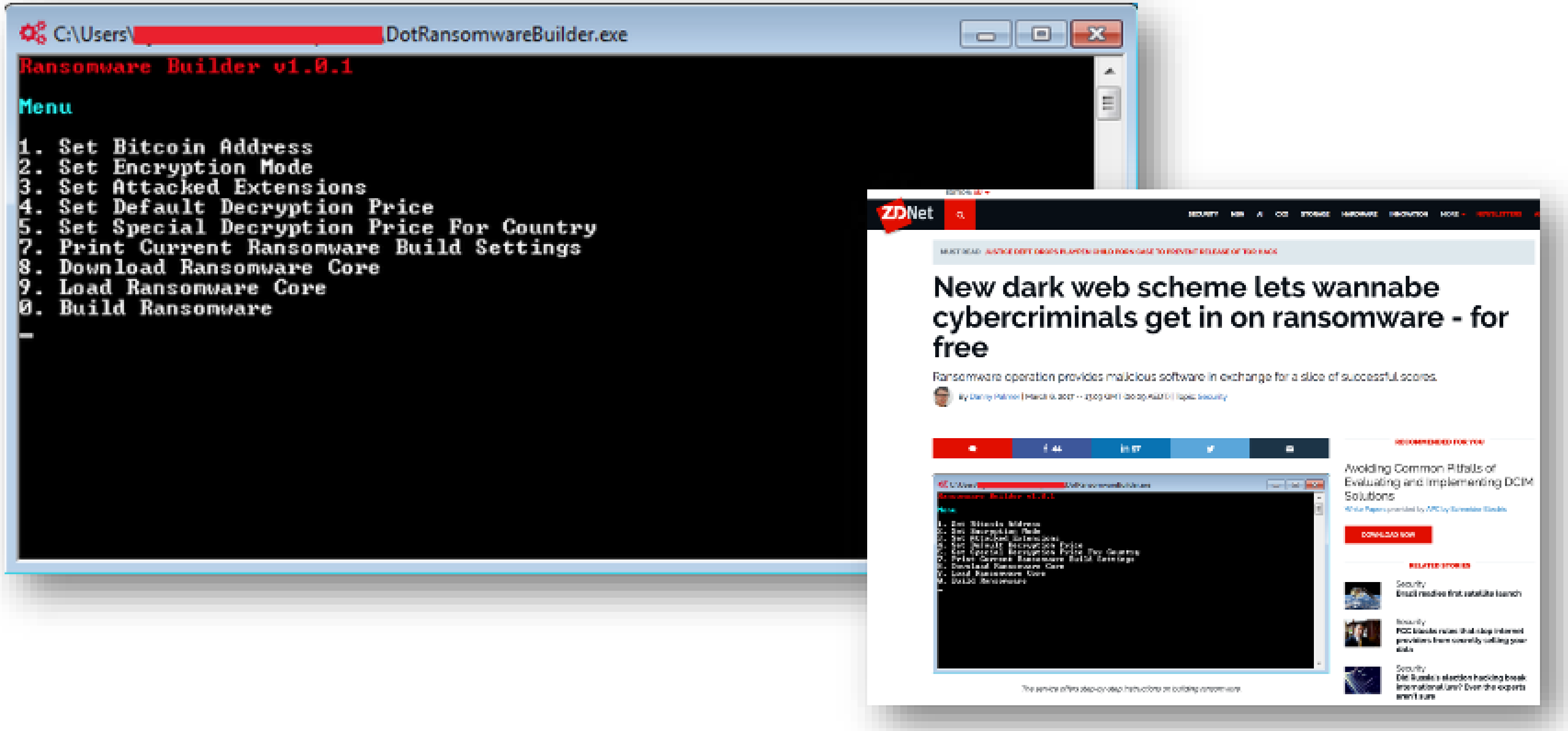
Overview	• Security in DevOps Overview – Stack Security • AWS DevOps Environment Compromise Demo • Morning Tea @ 10:30am
Coding	• Overview of a DevOps Lab Environment • Securing Custom Code • Third Party Code Issues • Static & Dynamic Code Analysis
Scanning	• Continuous Monitoring • Automating Security / Self Healing • Configuration & Infra as Code
Attacking	• Active Defense & Healing • Countermeasures for Attacks - Interactive • Question Time

Automation is Everywhere

THIS ROBOT BARISTA MAKES A DANG GOOD LATTE

Source: https://www.wired.com/2017/01/cafe-x-robot-barista/

Adversaries are using Automation



C:\Users\[redacted]\DotRansomwareBuilder.exe

```
Ransomware Builder v1.0.1

Menu

1. Set Bitcoin Address
2. Set Encryption Mode
3. Set Attacked Extensions
4. Set Default Decryption Price
5. Set Special Decryption Price For Country
7. Print Current Ransomware Build Settings
8. Download Ransomware Core
9. Load Ransomware Core
0. Build Ransomware
_
```

ZDNet

New dark web scheme lets wannabe cybercriminals get in on ransomware - for free

Ransomware operation provides malicious software in exchange for a slice of successful scores.

By Danny Palmer | March 9, 2017 -- [illegible] | Topic: Security

The service offers step-by-step instructions on building ransomware.

Avoiding Common Pitfalls of Evaluating and Implementing DCIM Solutions

Security
Brazil readies first satellite launch

Security
FCC blocks rules that stop internet providers from secretly selling your data

Security
Did Russia's election hacking break international law? Even the experts aren't sure

Source: http://www.zdnet.com/article/new-dark-web-scheme-lets-wannabe-cybercriminals-get-in-on-ransomware-for-free/

RDONER
FELIX

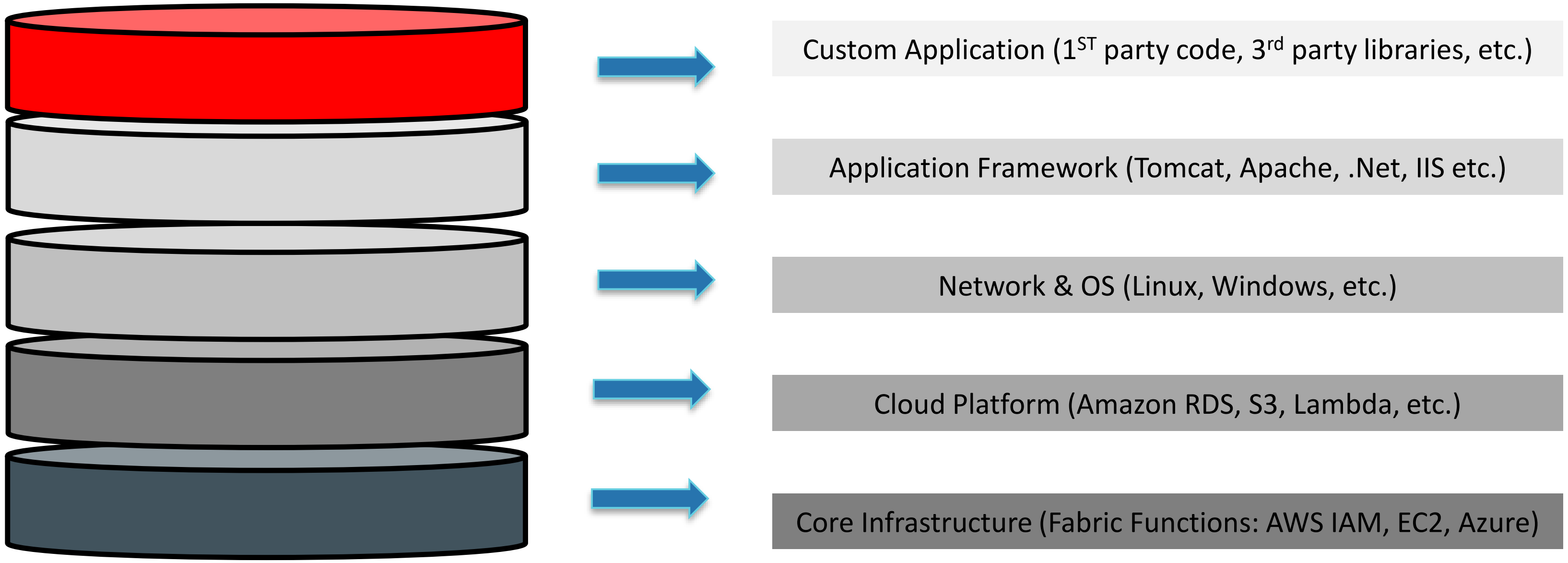
NOGA
LIU
BAJI

8

DevOps DJ’s!

DevOps DJ’s!

DevOps Coverage: Speed & Timing



Custom Application (1ST party code, 3rd party libraries, etc.)

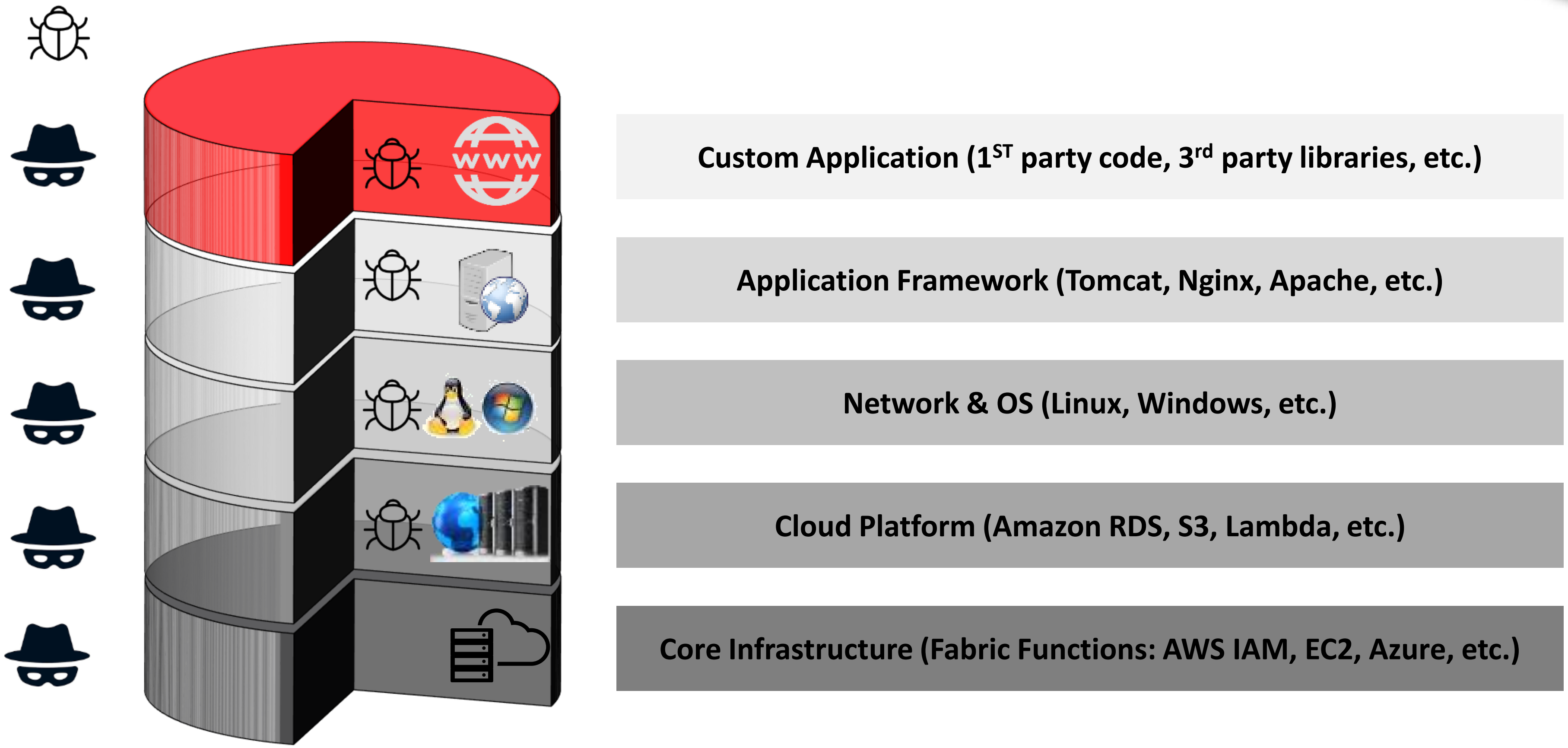
Application Framework (Tomcat, Apache, .Net, IIS etc.)

Network & OS (Linux, Windows, etc.)

Cloud Platform (Amazon RDS, S3, Lambda, etc.)

Core Infrastructure (Fabric Functions: AWS IAM, EC2, Azure)

Introducing StackSec



Custom Application (1ST party code, 3rd party libraries, etc.)

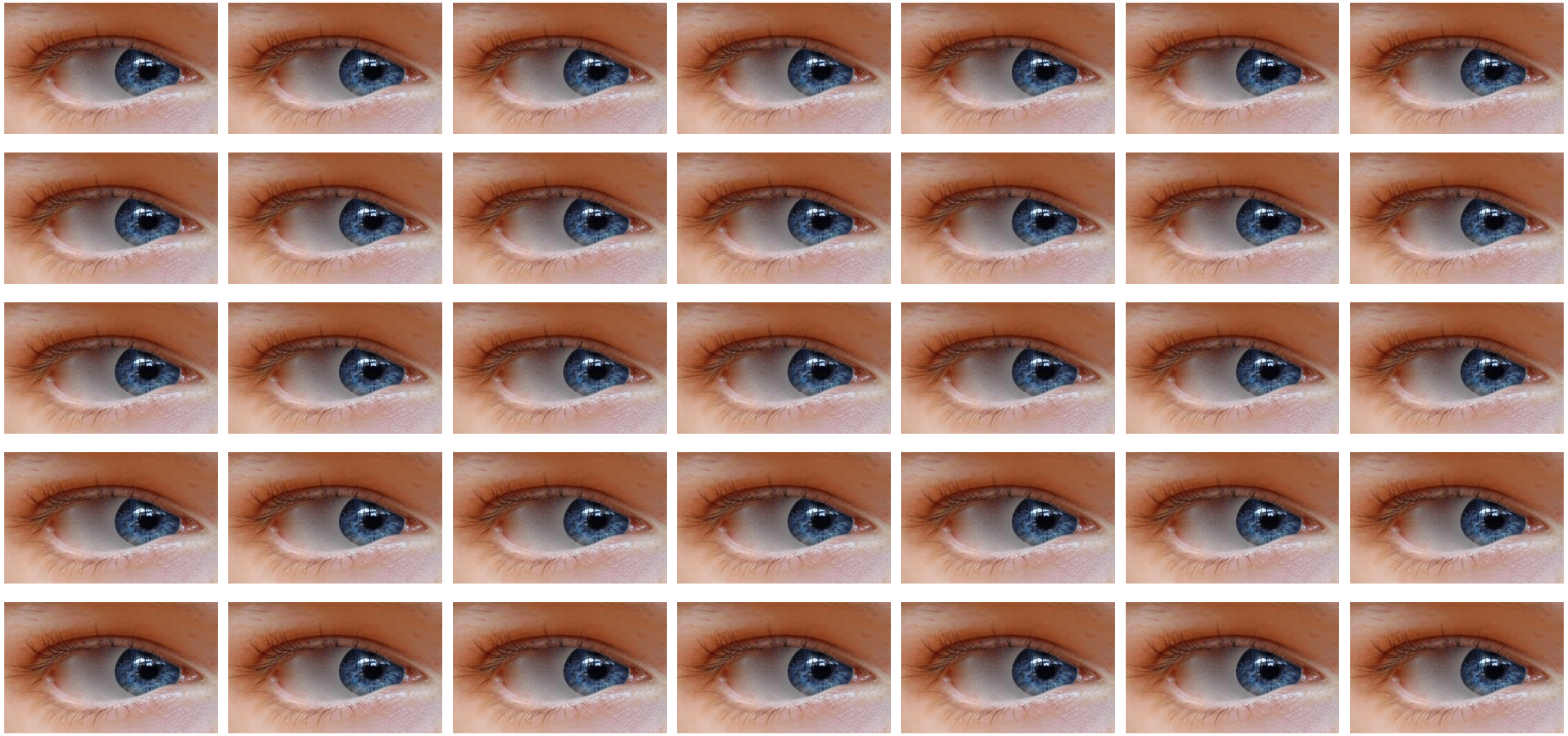
Application Framework (Tomcat, Nginx, Apache, etc.)

Network & OS (Linux, Windows, etc.)

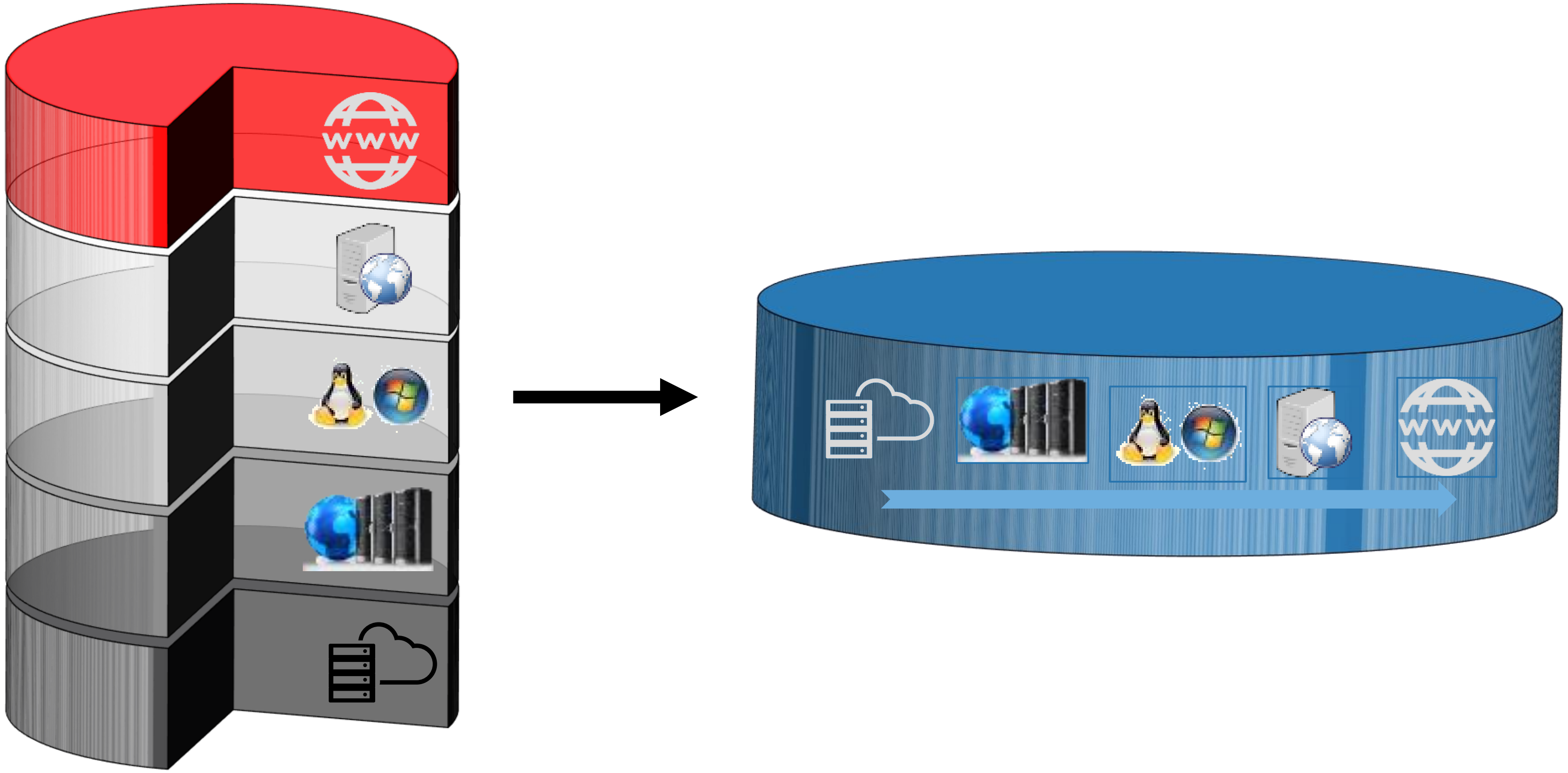
Cloud Platform (Amazon RDS, S3, Lambda, etc.)

Core Infrastructure (Fabric Functions: AWS IAM, EC2, Azure, etc.)

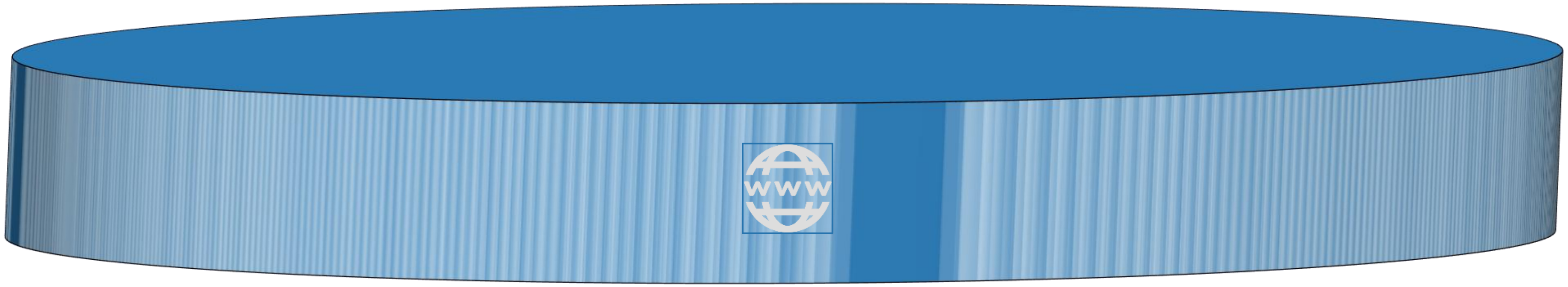
Continuous Monitoring



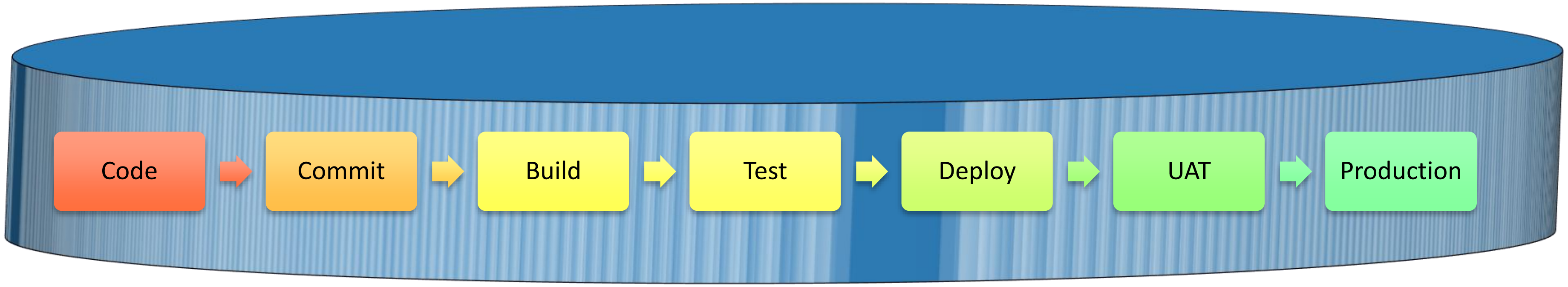
Collapse the Vertical Plane



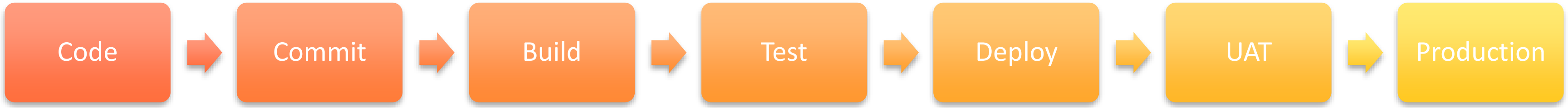
Stretch into Horizontal Plane



Produces the DevOps Pipeline



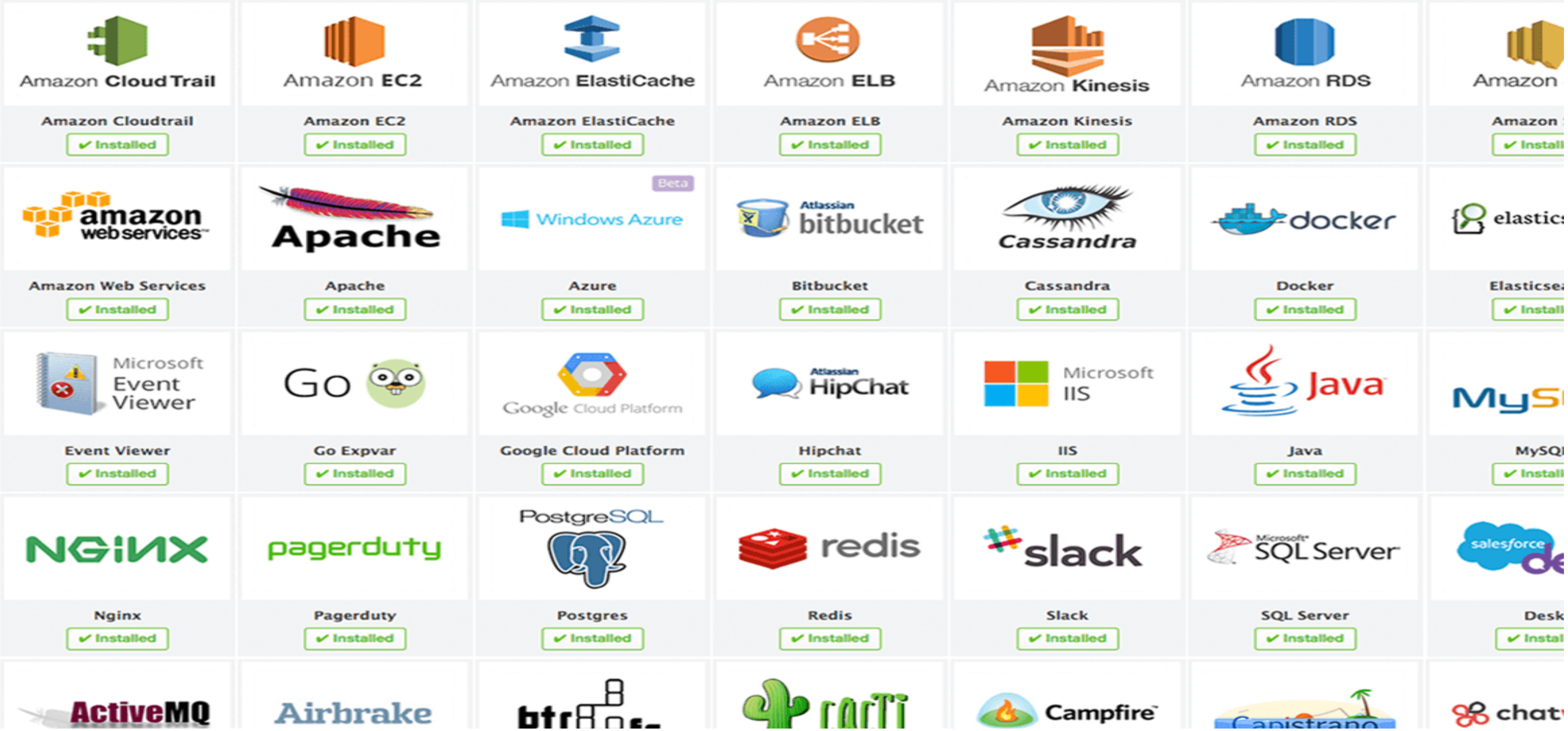
We look at a generic development pipeline…



Code → Commit → Build → Test → Deploy → UAT → Production

1. Development Environment
2. Source Code Repository
3. Build Platform (CI)
4. Deployment Process (CD)
5. Staging / Production Hosting Environment

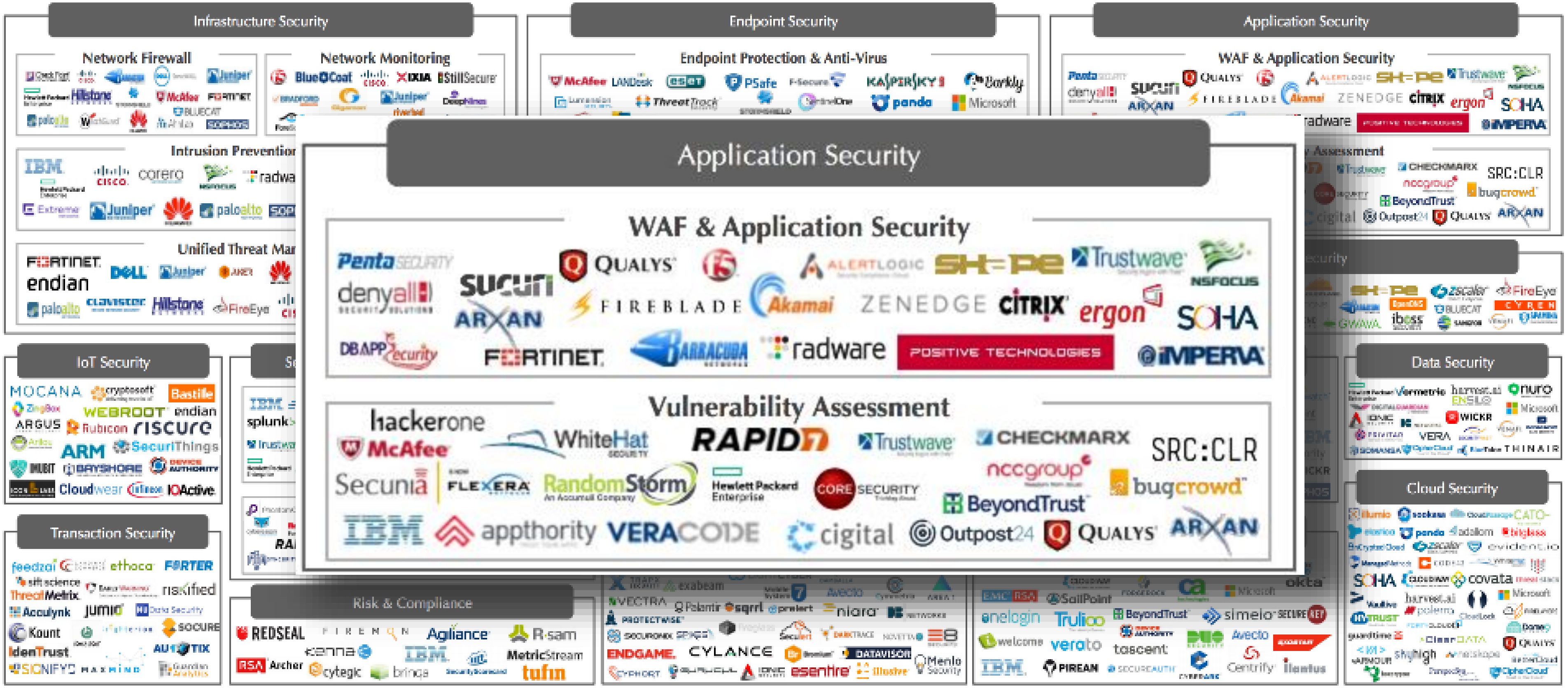
DevOps Mayhem



Amazon Cloudtrail ✔ Installed	Amazon EC2 ✔ Installed	Amazon ElastiCache ✔ Installed	Amazon ELB ✔ Installed	Amazon Kinesis ✔ Installed	Amazon RDS ✔ Installed	Amazon ✔ Install
Amazon Web Services ✔ Installed	Apache ✔ Installed	Beta Azure ✔ Installed	Bitbucket ✔ Installed	Cassandra ✔ Installed	Docker ✔ Installed	Elasticse ✔ Install
Event Viewer ✔ Installed	Go Expvar ✔ Installed	Google Cloud Platform ✔ Installed	Hipchat ✔ Installed	IIS ✔ Installed	Java ✔ Installed	MySQ ✔ Install
Nginx ✔ Installed	Pagerduty ✔ Installed	Postgres ✔ Installed	Redis ✔ Installed	Slack ✔ Installed	SQL Server ✔ Installed	Desk ✔ Instal
ActiveMQ	Airbrake			Campfire		chat

15Aug17

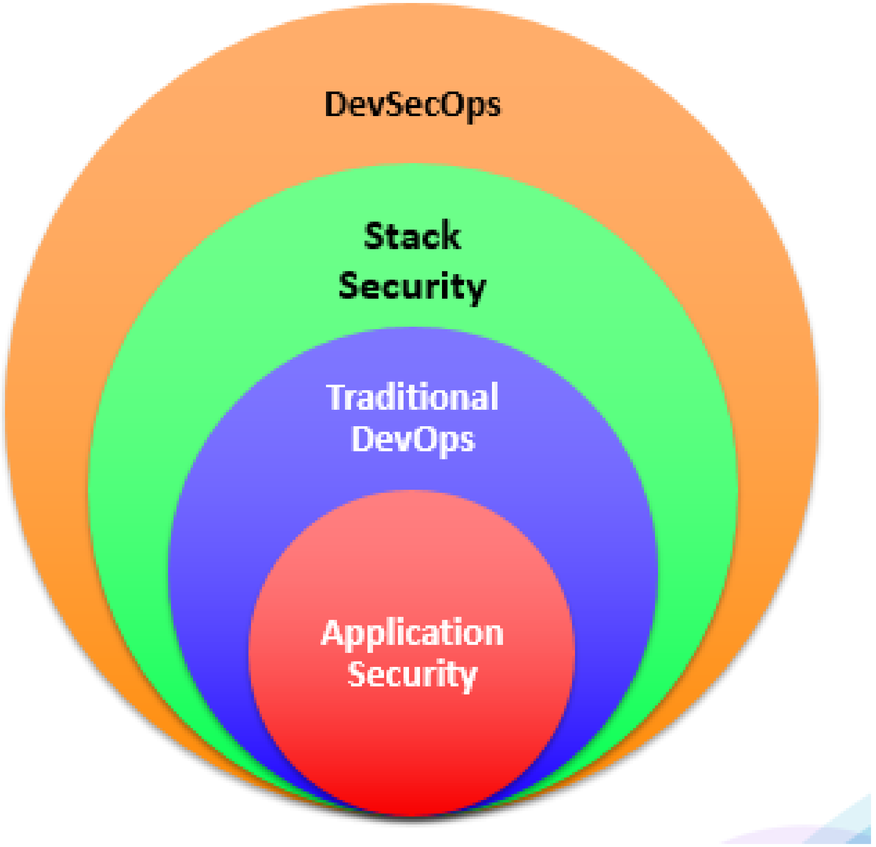
Tools, Tools & More Tools



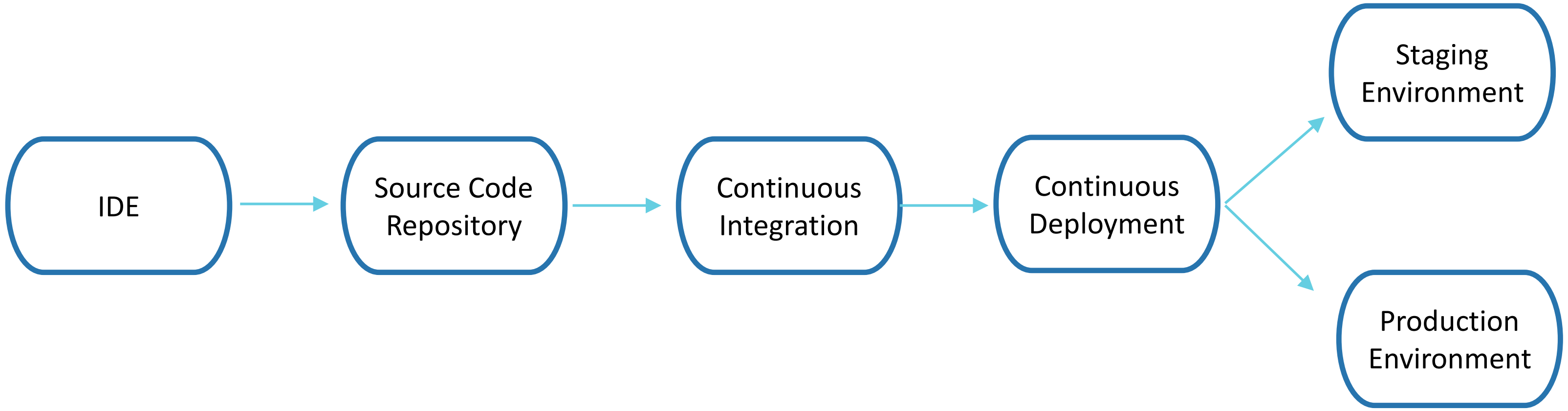
Source: Momentum Partners

Coverage Across Public, Private & Hybrid Clouds

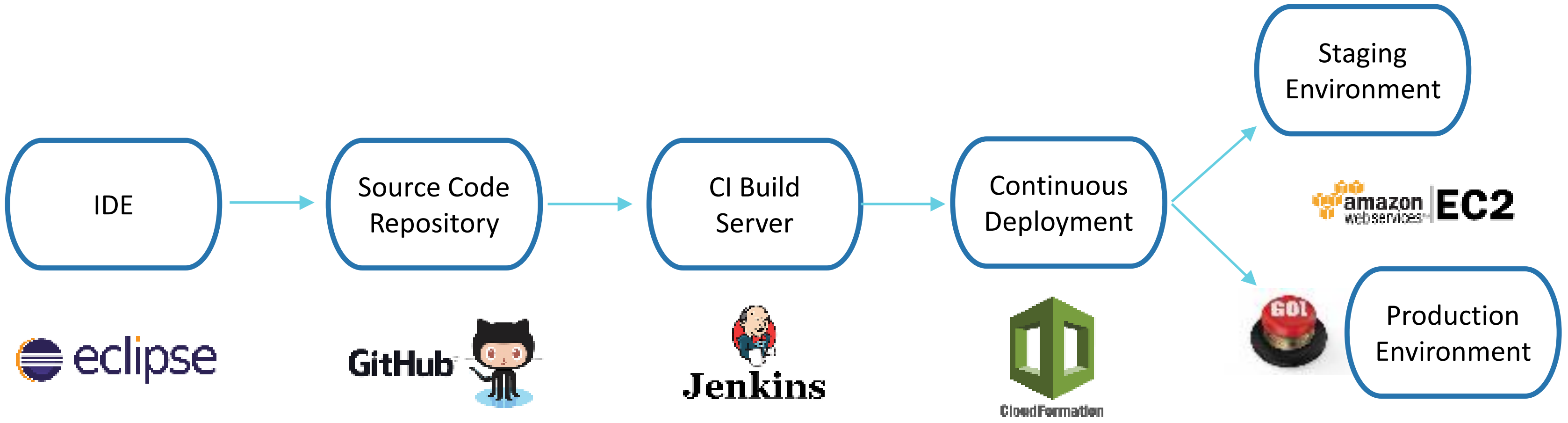
DevSecOps – Securing the Stack



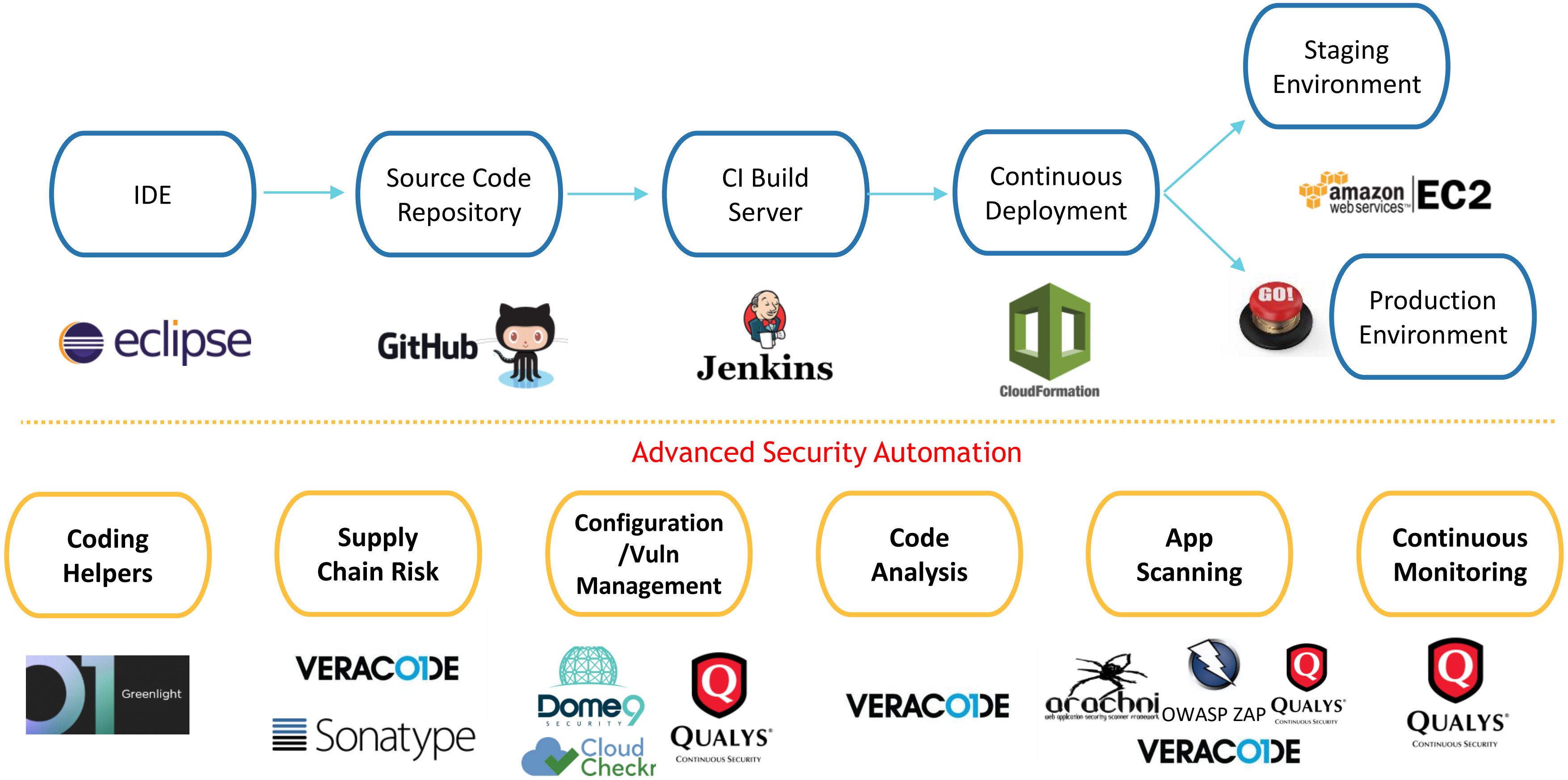
DevSecOps Lab



DevSecOps Lab



DevSecOps Lab



IDE → Source Code Repository → CI Build Server → Continuous Deployment → Staging Environment / Production Environment

Advanced Security Automation

Coding Helpers

Supply Chain Risk

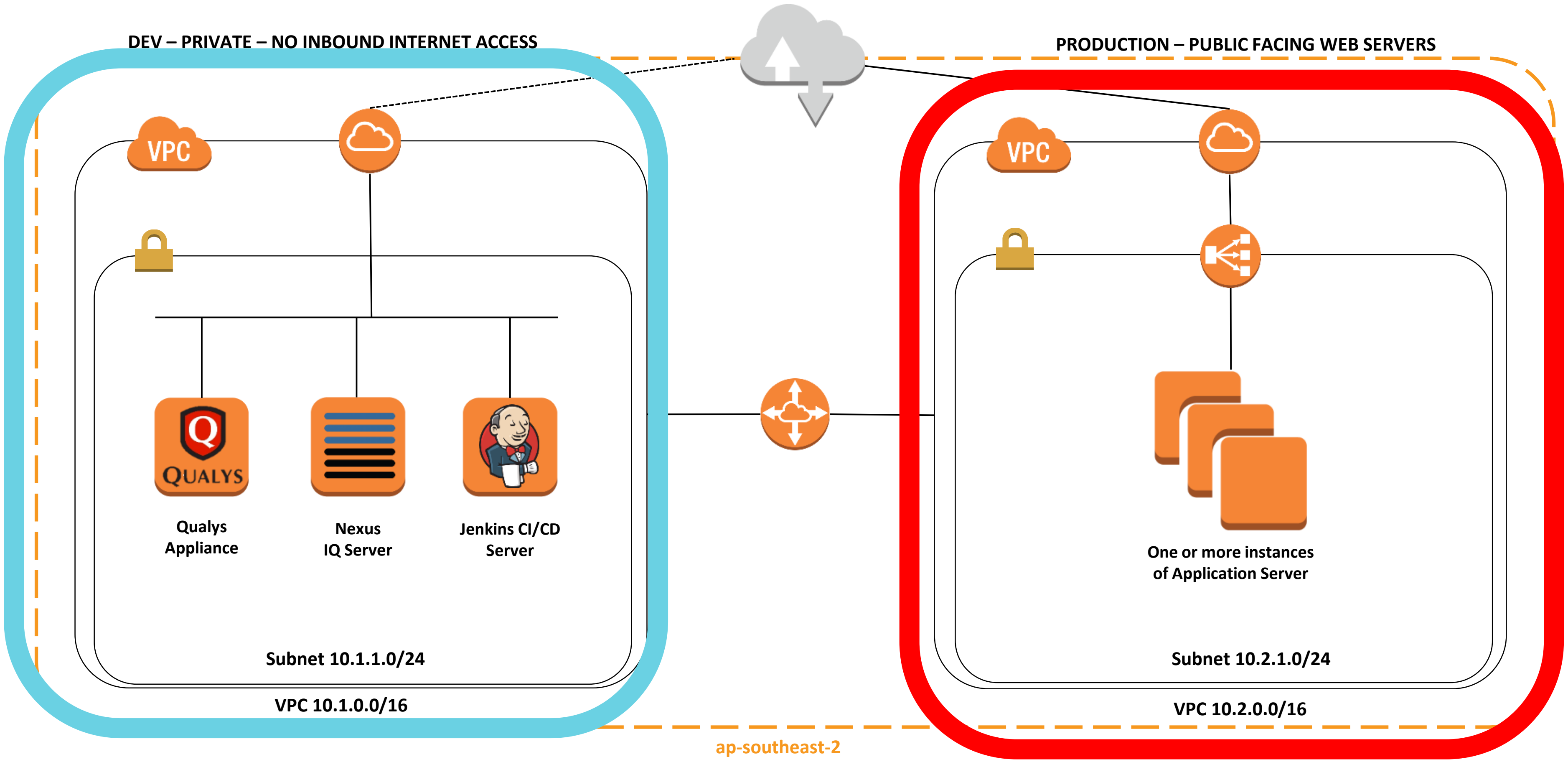
Configuration /Vuln Management

Code Analysis

App Scanning

Continuous Monitoring

Welcome to the DevSecOps Lab

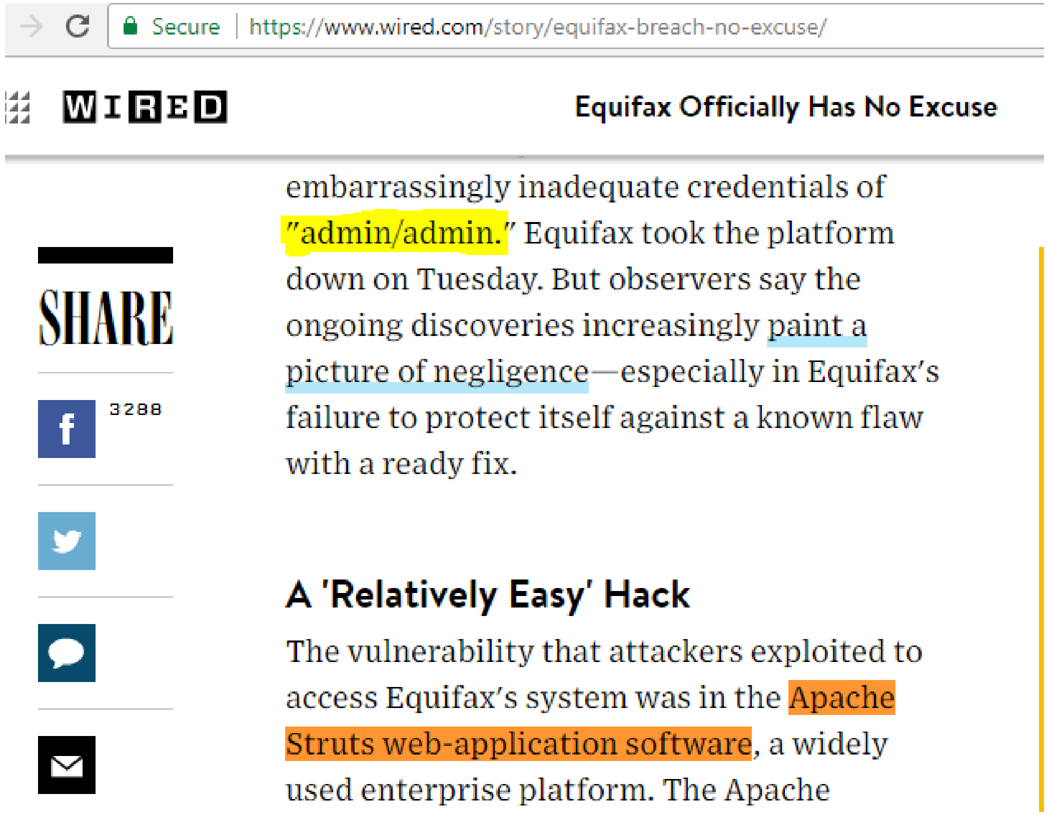


DevSecOps Lab AWS Kill Chain Attack

We played this video during the learning lab:

https://www.youtube.com/watch?v=fm4CqlxqQfs

World's Largest PII Data Breach?

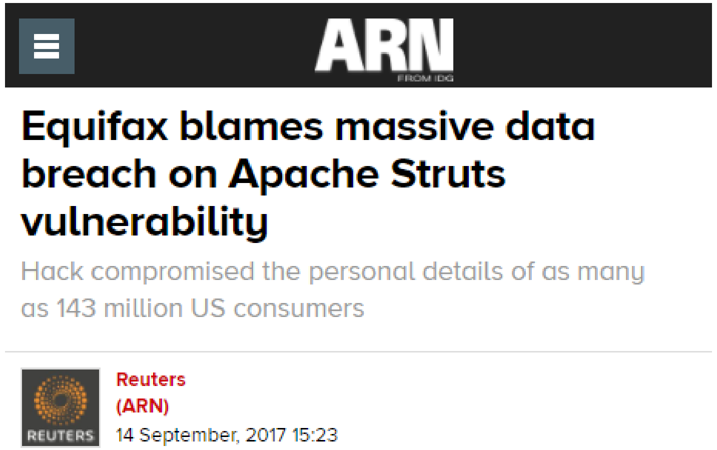


Equifax Officially Has No Excuse

embarrassingly inadequate credentials of "admin/admin." Equifax took the platform down on Tuesday. But observers say the ongoing discoveries increasingly paint a picture of negligence—especially in Equifax's failure to protect itself against a known flaw with a ready fix.

A 'Relatively Easy' Hack

The vulnerability that attackers exploited to access Equifax's system was in the Apache Struts web-application software, a widely used enterprise platform. The Apache



Equifax blames massive data breach on Apache Struts vulnerability

Hack compromised the personal details of as many as 143 million US consumers

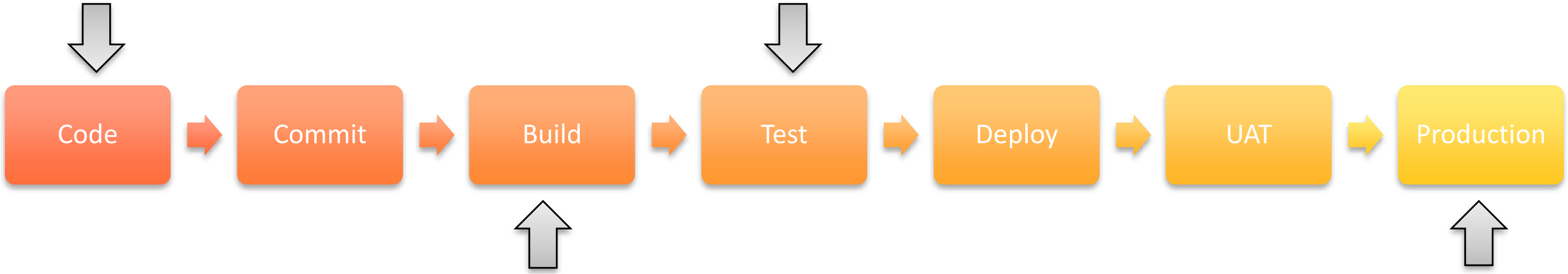
Reuters (ARN)
14 September, 2017 15:23

- Addressing the need to identify defects earlier.
- Writing and testing your in-house “first party” code.
- Testing and inspecting libraries and “third party” code.

Defense in Depth

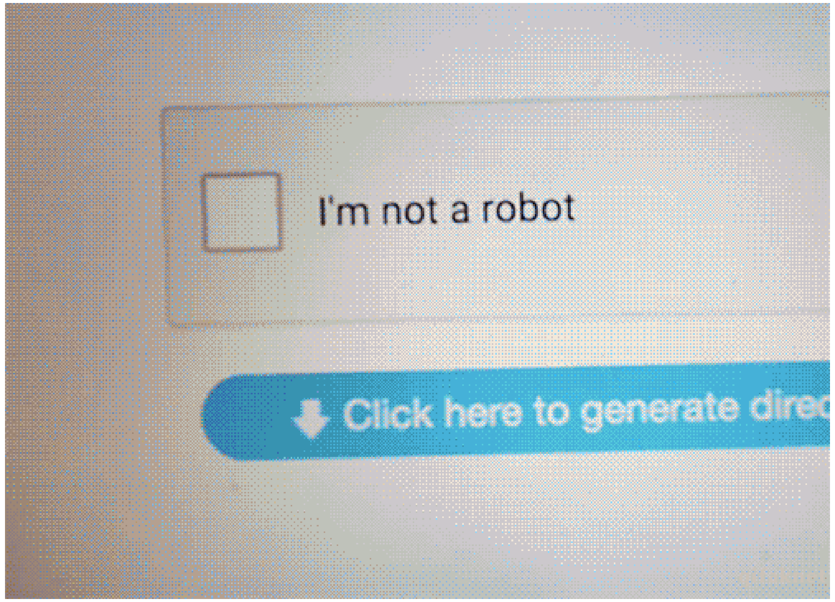
Layer #1 – The developer has an opportunity to avoid introducing a security vulnerability in their IDE.

Layer #3 – Automated dynamic scanning of the application detects the same vulnerability if it gets this far.



Layer #2 – Static code analysis triggered by the code commit action identifies the vulnerability – build fails.

Layer #4 – Continuous Monitoring & Vulnerability Management detects the exposed vulnerability. Add comprehensive Manual Pen Test.


□ I'm not a robot
Click here to generate direc

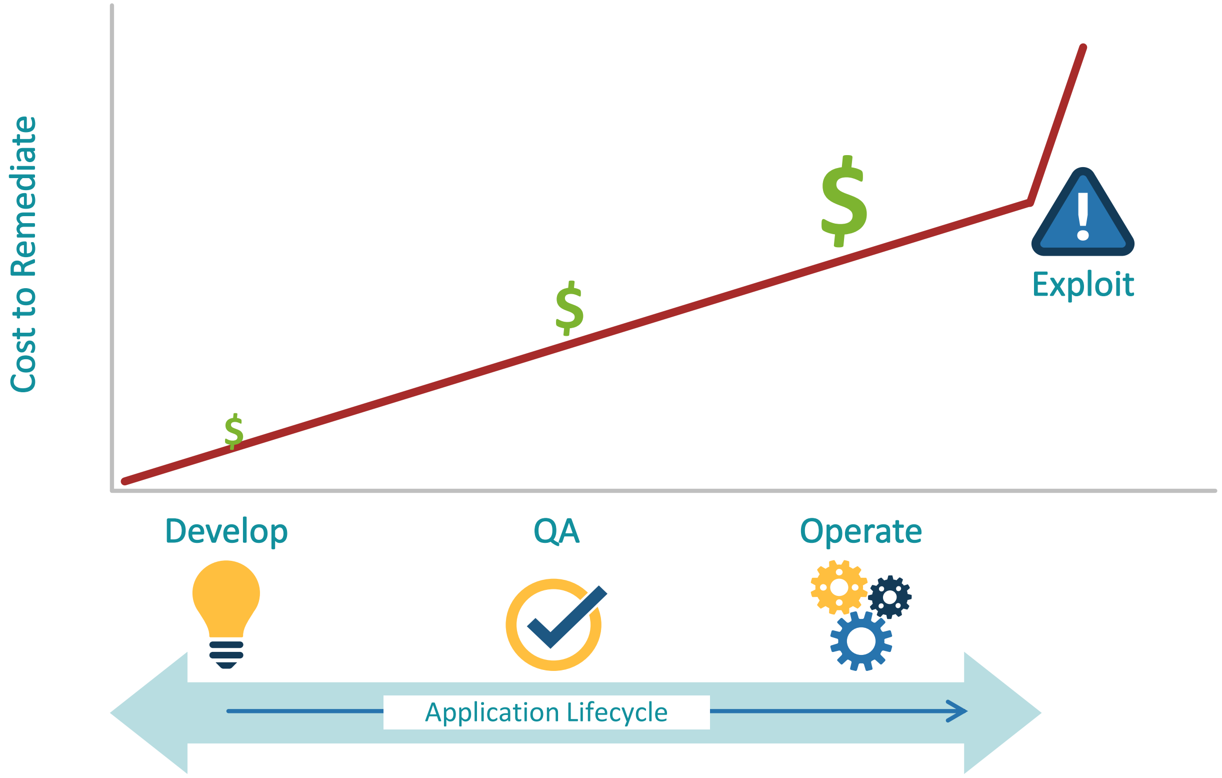
Good Quality Code – Problem Statement

Why do you need to address code quality?

- Vulnerabilities caused by coding may lead to **unacceptable risk**.
- Well written code **performs better**
  - If well understood, has less risk of being vulnerable.
  - Likely to have better bottom line results on the final application.

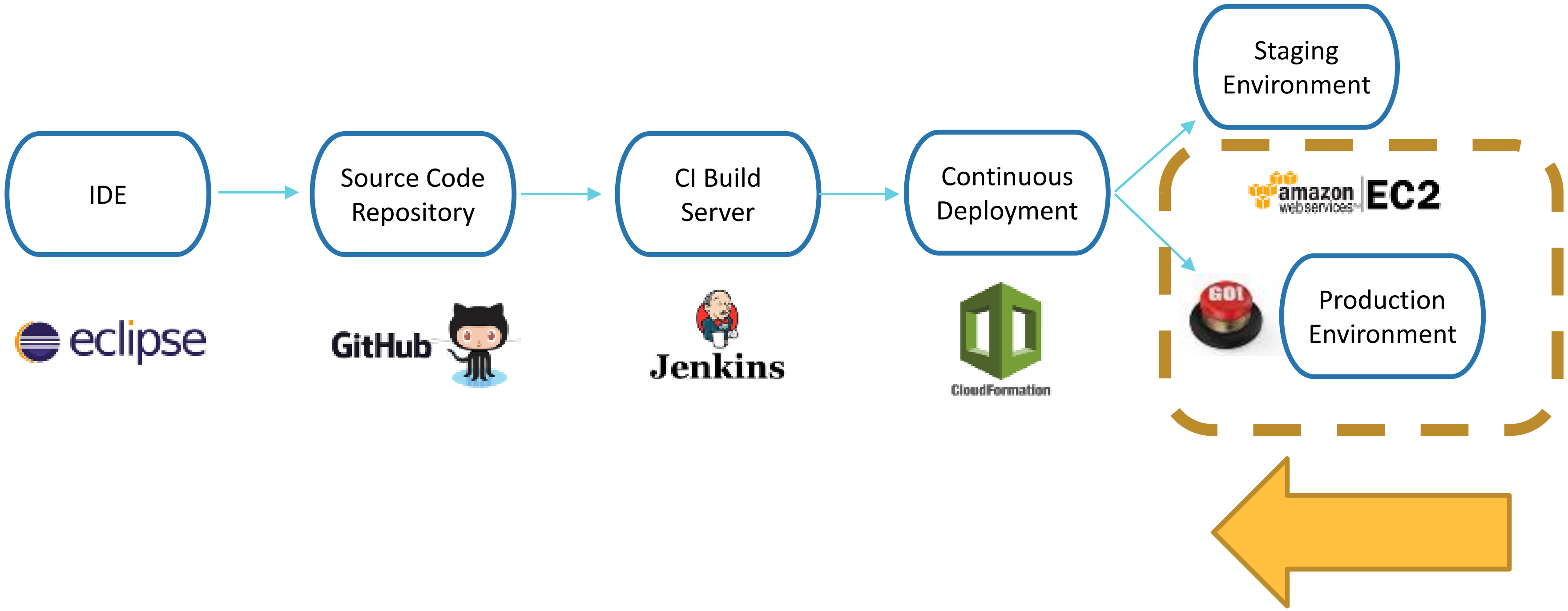
When is the best time to address coding defects?

Identify Defects As Soon As Possible

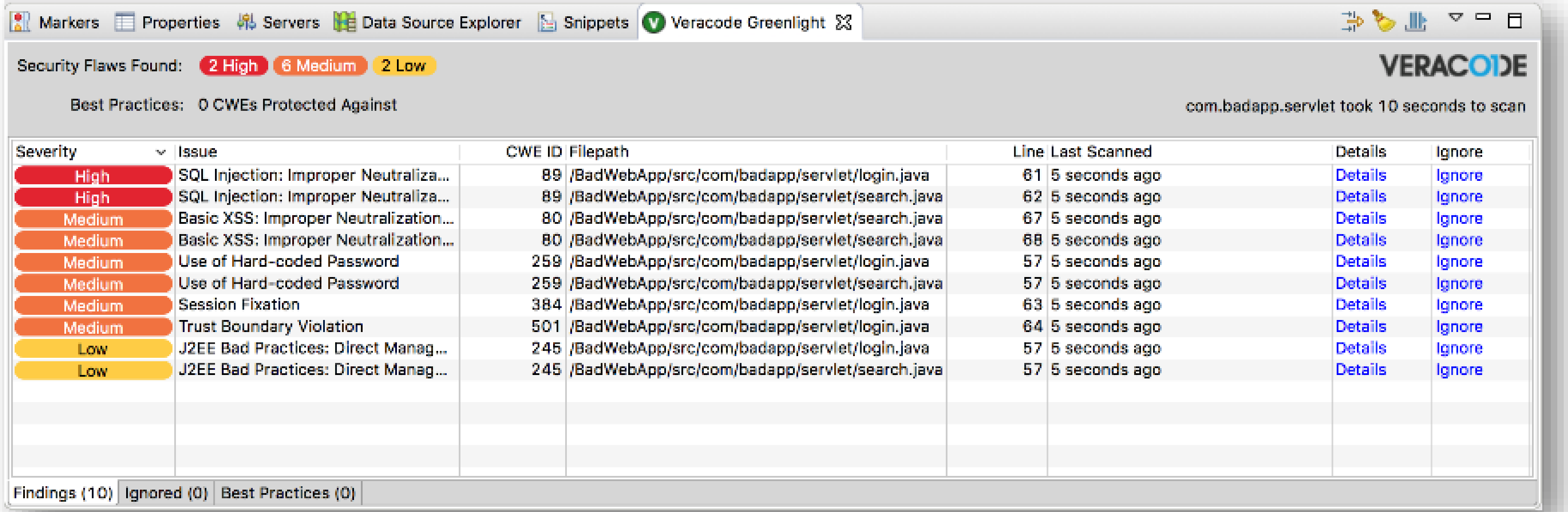


Source: Veracode

Shifting Left



Scanning Code at the IDE



Security Flaws Found: 2 High 6 Medium 2 Low

Best Practices: 0 CWEs Protected Against

com.badapp.servlet took 10 seconds to scan

Severity	Issue	CWE ID	Filepath	Line	Last Scanned	Details	Ignore
High	SQL Injection: Improper Neutraliza...	89	/BadWebApp/src/com/badapp/servlet/login.java	61	5 seconds ago	Details	Ignore
High	SQL Injection: Improper Neutraliza...	89	/BadWebApp/src/com/badapp/servlet/search.java	62	5 seconds ago	Details	Ignore
Medium	Basic XSS: Improper Neutralization...	80	/BadWebApp/src/com/badapp/servlet/search.java	67	5 seconds ago	Details	Ignore
Medium	Basic XSS: Improper Neutralization...	80	/BadWebApp/src/com/badapp/servlet/search.java	68	5 seconds ago	Details	Ignore
Medium	Use of Hard-coded Password	259	/BadWebApp/src/com/badapp/servlet/login.java	57	5 seconds ago	Details	Ignore
Medium	Use of Hard-coded Password	259	/BadWebApp/src/com/badapp/servlet/search.java	57	5 seconds ago	Details	Ignore
Medium	Session Fixation	384	/BadWebApp/src/com/badapp/servlet/login.java	63	5 seconds ago	Details	Ignore
Medium	Trust Boundary Violation	501	/BadWebApp/src/com/badapp/servlet/login.java	64	5 seconds ago	Details	Ignore
Low	J2EE Bad Practices: Direct Manag...	245	/BadWebApp/src/com/badapp/servlet/login.java	57	5 seconds ago	Details	Ignore
Low	J2EE Bad Practices: Direct Manag...	245	/BadWebApp/src/com/badapp/servlet/search.java	57	5 seconds ago	Details	Ignore

Scanning Code at the IDE

eclipse

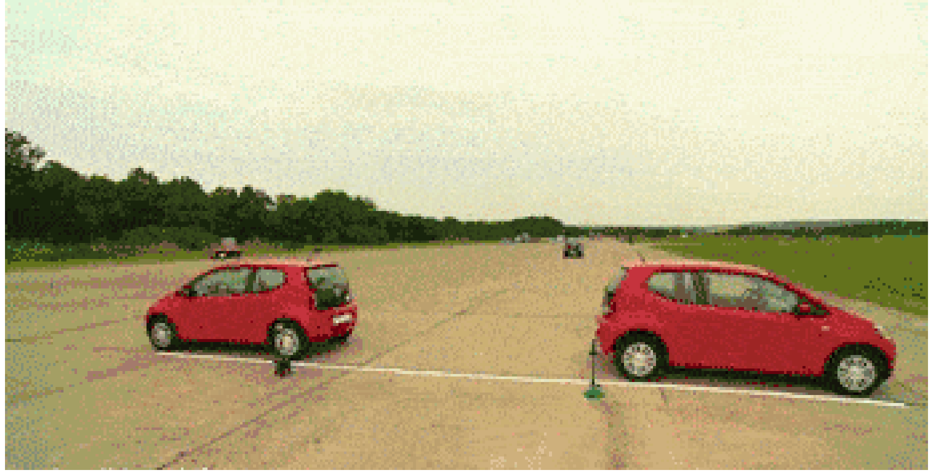
Jenkins

Automating Security at the Deploy Layer

Preventing a deployment if something fails.

```
Using Scan 1218389
Checks Failed
POST BUILD TASK : FAILURE
END OF POST BUILD TASK: 0
ESCALATE FAILED POST BUILD TASK
TO JOB STATUS
Build step ‘Post build task’
changed build result to FAILURE
Finished: FAILURE
```

Third Party Libraries



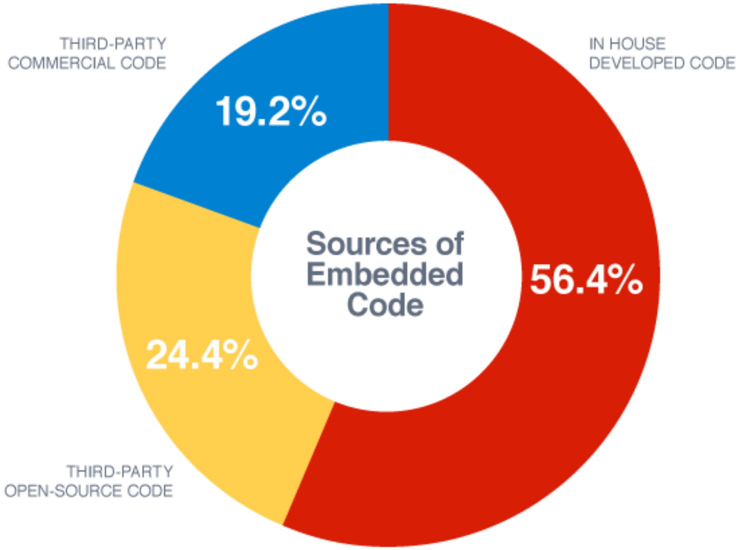
Third Party Code – Problem Statement

Why do you need to address third party library risk?

- Embedding third party code in your application has huge advantages, but comes at the risk of **latent exposure to vulnerabilities**.
- Many open source library repositories have little or no vetting of contributors, meaning **third party code cannot be trusted** blindly.
- When vulnerabilities are discovered in a shared library, it is important to **quickly identify your exposure**.

- Supply Chain Security: Identify Vulnerable Third Party Components. Automatically strengthen and secure software supply chains everywhere, and at scale

Source: https://www.sonatype.com



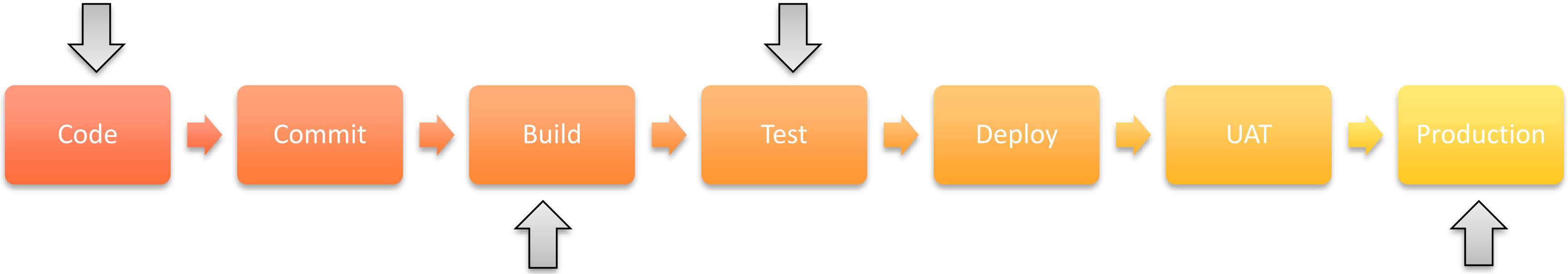
Source: https://www.grammatech.com/

Source: https://www.grammatech.com/

Defense in Depth

Layer #1 – The developer has an opportunity to avoid introducing a security vulnerability in their IDE.

Layer #3 – Automated dynamic scanning of the application detects the same vulnerability if it gets this far.



Code → Commit → Build → Test → Deploy → UAT → Production

Layer #2 – Static code analysis triggered by the code commit action identifies the vulnerability – build fails.

Layer #4 – Continuous Monitoring & Vulnerability Management detects the exposed vulnerability. Add comprehensive Manual Pen Test.

Monitoring & Self-Healing

- Cloud environments require proper configuration management.
- Visibility is key to knowing if your DevOps stack is secure.
- Self-healing is a growing trend and worth implementing.

Monitoring & Self-Healing

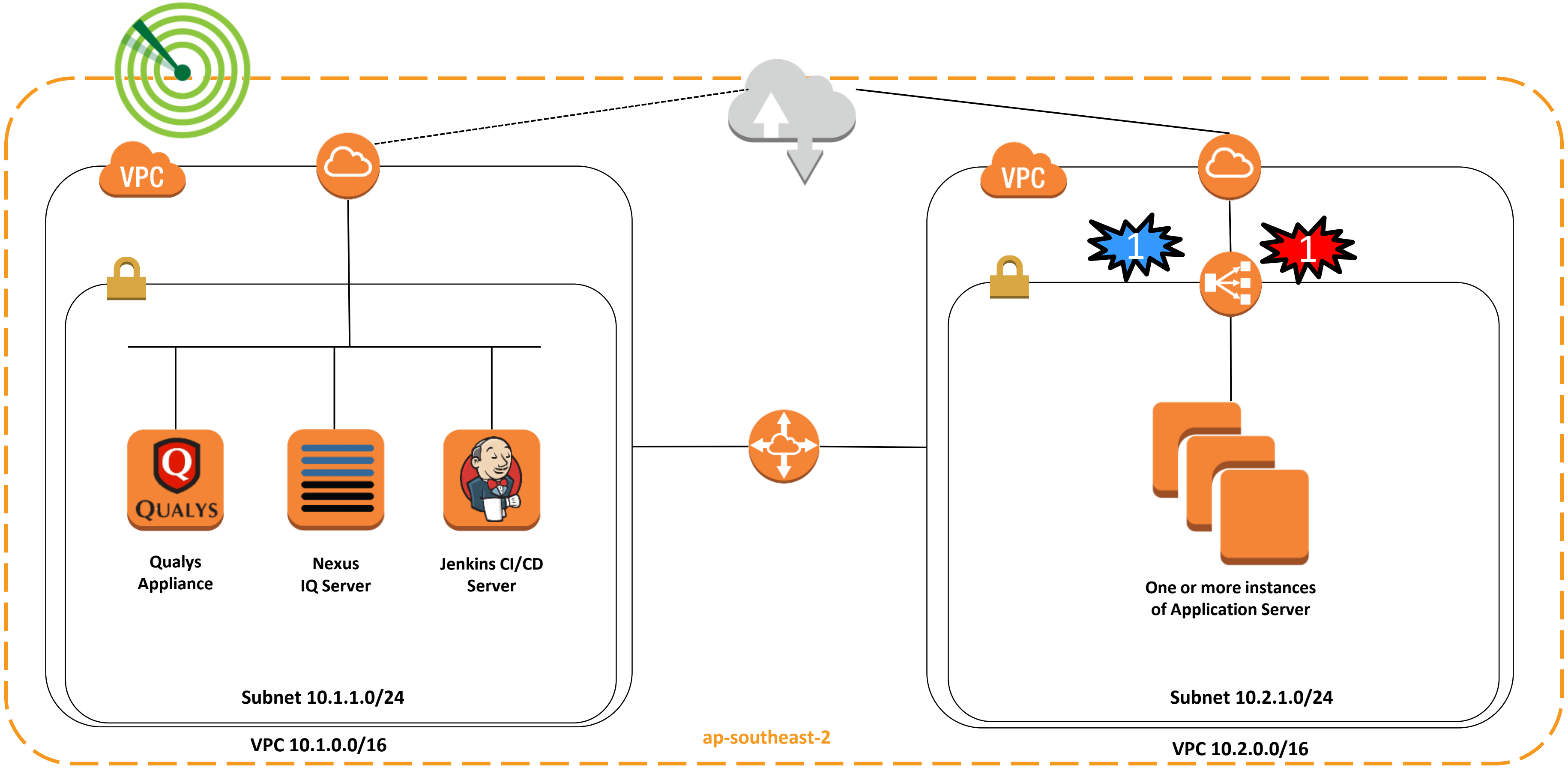
Configuration Monitoring – Problem Statement

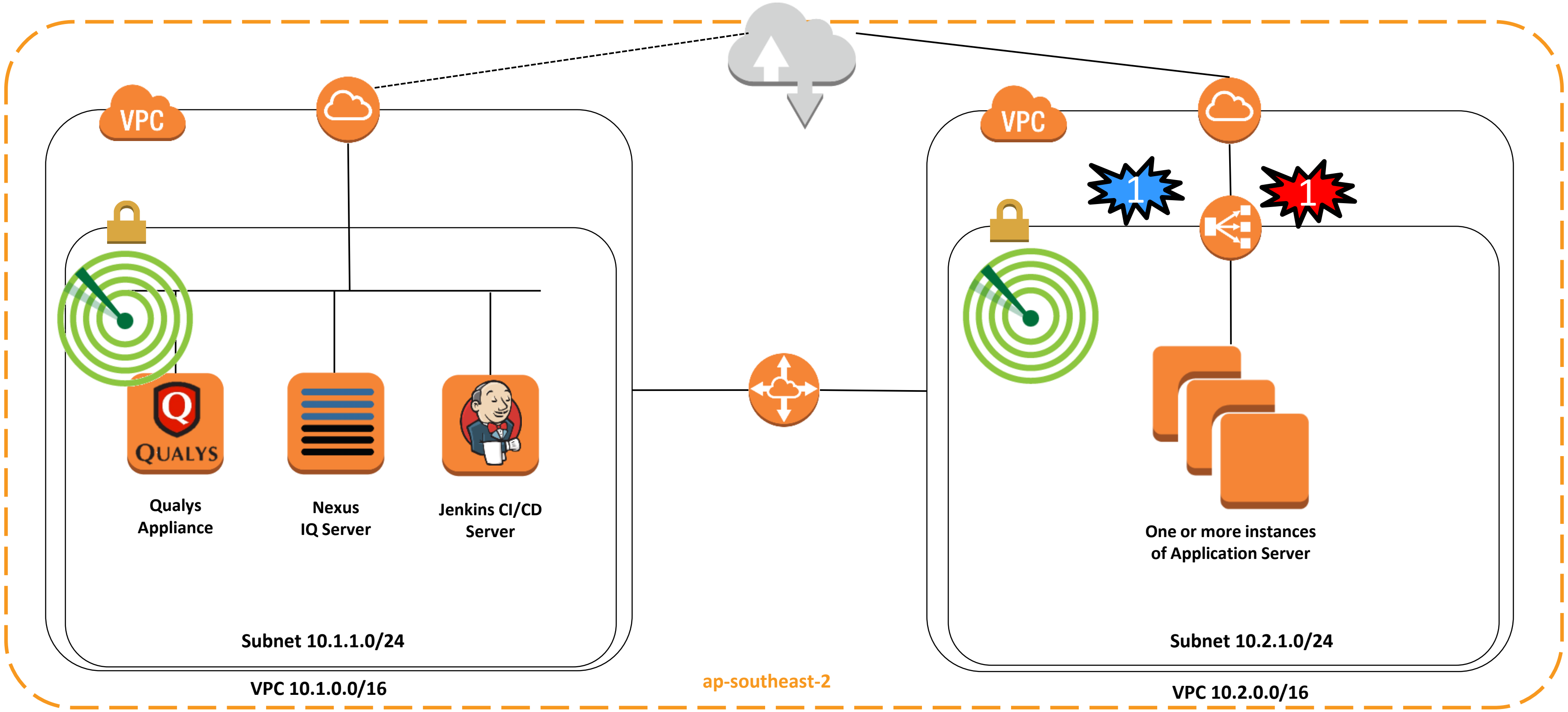
Why is your cloud environment configuration important?

- Complex environments have **complex and diverse configurations**.
- Cloud configurations **aren't always visible**, and we need that visibility to understand the real configuration.
- We **need to have assurance** that our configuration standard is being enforced and is compliant.

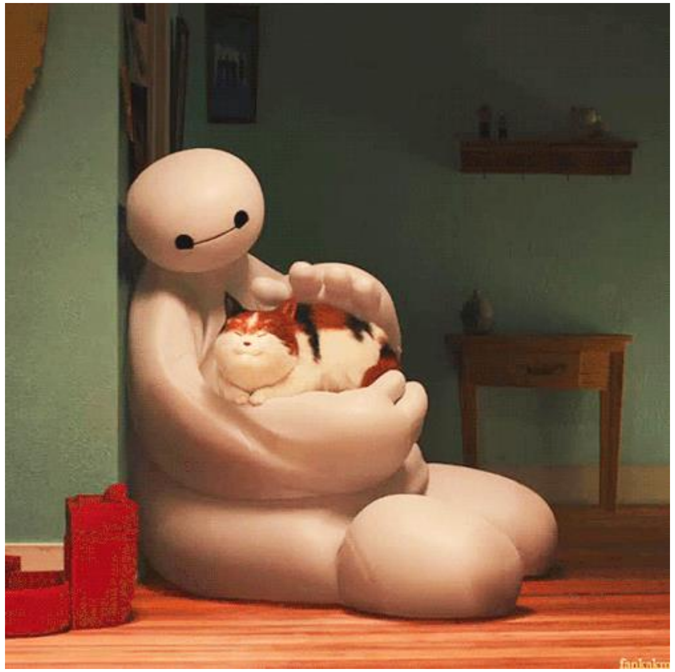
CAM 05
5/23/2008 12:10 PM
MAKE GIFS AT GIFSOUP.COM

Continuous Monitoring


VPC
Qualys Appliance
Nexus IQ Server
Jenkins CI/CD Server
Subnet 10.1.1.0/24
VPC 10.1.0.0/16
ap-southeast-2
VPC
1
1
One or more instances of Application Server
Subnet 10.2.1.0/24
VPC 10.2.0.0/16


VPC
VPC
1
1
Qualys Appliance
Nexus IQ Server
Jenkins CI/CD Server
One or more instances of Application Server
Subnet 10.1.1.0/24
Subnet 10.2.1.0/24
VPC 10.1.0.0/16
ap-southeast-2
VPC 10.2.0.0/16

Self-Healing



Self-Healing - Problem Statement

Why is **Self-Healing** important?

- Respond to changes in your environment immediately, reverting changes - malicious or accidental.
- Assurance that your stack configuration is compliant to your risk appetite at all times.
- Alert you to take action for improvement if it does detect unwanted changes (or alert of a security incident).

Event Driven Security, Self-Healing, or RASP

The techniques we’re about to look at in our lab are all known by different names:

- Event Driven Security – responding to events
- RASP – Runtime Application Self Protection
- Self-Healing – we think this describes it nicely!

There may be subtle difference in implementation, but for the large part we consider they all do the same thing.

We’re Going Serverless!

"Serverless computing solutions execute logic in environments with no visible VM or OS. Services such as Amazon Web Services Lambda are disrupting many cloud development and operational patterns. Technology and service provider product managers must prepare for the change.” - ***Gartner***

AWS Lambda

- It’s ”Serverless”
- A stateless, programmatic function that responds to events based on triggers.
- Other Platforms:
  - Microsoft Azure: “Azure Functions”
  - Google Cloud Platform: “Google Cloud Functions”

AWS Lambda

Event Driven Security / Self-Healing

To implement automated self-healing using a serverless solution we generally need a few things:

1. A well defined “event” that we can respond to (i.e. an open port, or a new user account being created)
2. A near real-time source of logging data to listen for the event.
3. Something to do if the event is triggered.

Demo

Demo Lambda locking a user out after they try to create another user account.

Or disable user without 2-factor?

Run Time Defence - WAF

Capability	Requirements
WAF's "could" mitigate this attack through Whitelisting *	**But only IF** the rules are set to whitelist valid content types or blacklist Object Graph Navigation Library (OGNL) expressions.
WAF's "could" mitigate this attack through Custom Rules **	**BUT a Custom rule reqd** to block requests that contain invalid Content-Type header values for a specific URL that accepts multipart requests conditions: request.path EQUAL "/struts2-showcase/index.action" request.header "Content-Type" NOT.EQUAL "multipart/form-data"
More Advanced WAFs "could" mitigate this attack through Zero Day Protections ***	Payload analysis on form submissions & API calls.

Run Time Defence - WAF

Capability	Requirements
More Advanced Application Firewalling – RASP ****	• Runtime application self-protection (RASP) • Built into an application • Detect and prevent real-time application attacks • “self-protecting” or reconfiguring automatically without human intervention (on conditions of threats, faults, etc.)

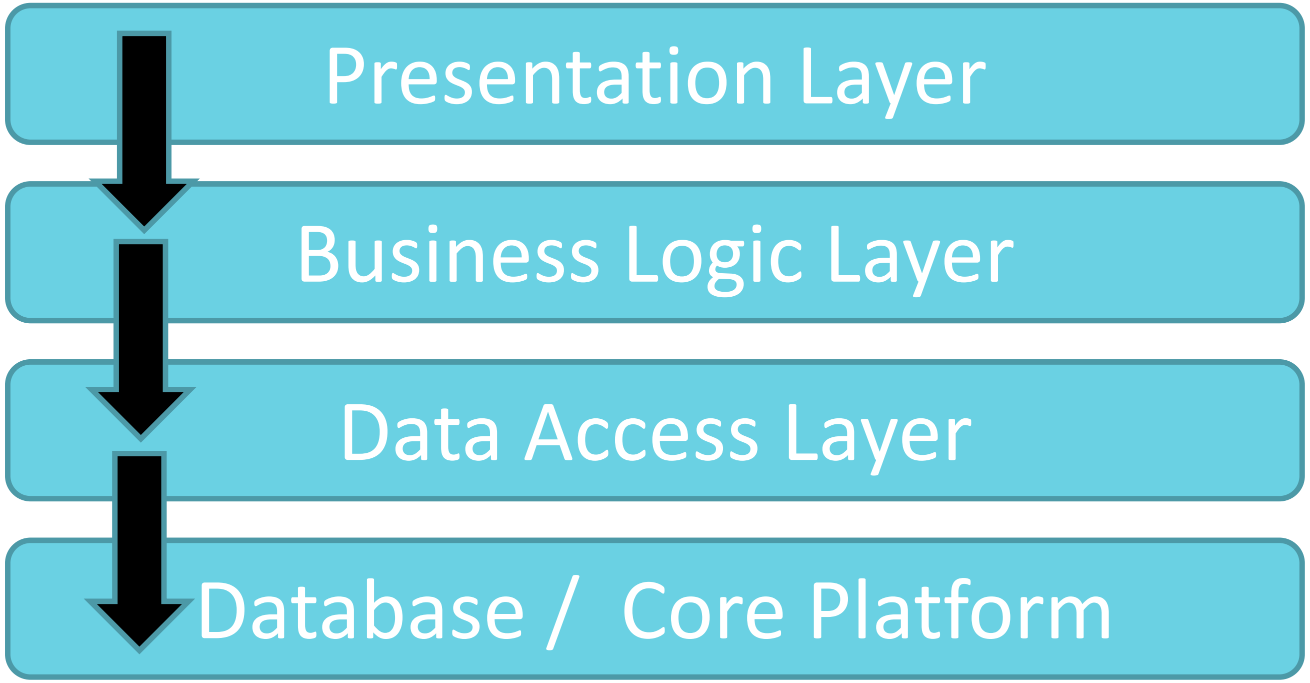
* https://blog.blackducksoftware.com/cve-2017-5638-anatomy-apache-struts-vulnerability
** https://blog.qualys.com/technology/2017/03/09/qualys-waf-2-0-protects-against-critical-apache-struts2-vulnerability-cve-2017-5638
*** https://www.imperva.com/blog/2017/09/apache-struts-rce-and-managing-app-risk/
**** https://www.veracode.com/security/runtime-application-self-protection-rasp , https://www.waratek.com/runtime-application-self-protection-rasp/

When attackers hack web apps/servers, they want to:

- Get access to sensitive data
- Remain persistent
- Access additional internal resources – Horizontal Attack

Pre Run Time Defence – Containers

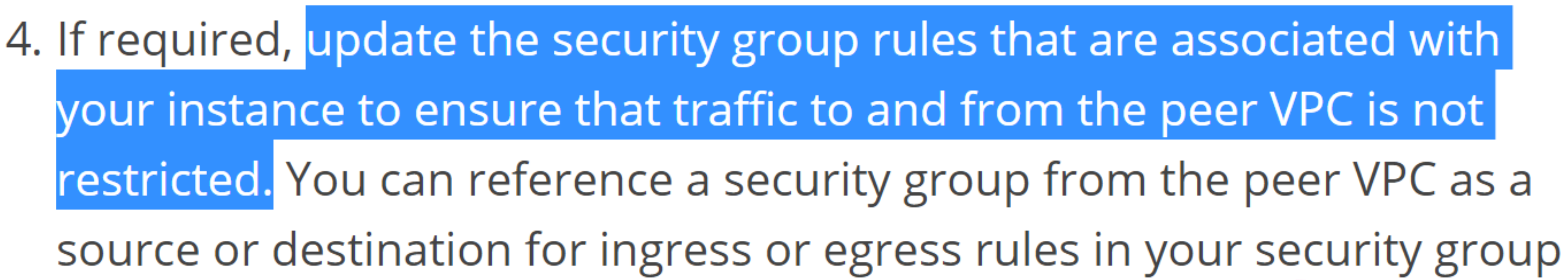
Container Attribute	Defence in Depth
TTL - Containers Don’t Live as Long as servers	Affects Persistence of Attack BUT – Permanent storage negates
Isolated from the underlying machine, and from other containers	Increasing difficulty for Pivot Attack BUT – need hardening
Fewer privileges than regular processes	Escape from a container usually involves kernel exploitation (difficult)
Container images can be scanned (before deployment) for known vulns	Quality at Source. *Prevent* images with a vulnerability from being deployed
Supports microservice architecture	Patch, update, redeploy


Presentation Layer
Business Logic Layer
Data Access Layer
Database / Core Platform

Run Time Defence – Container Firewalls

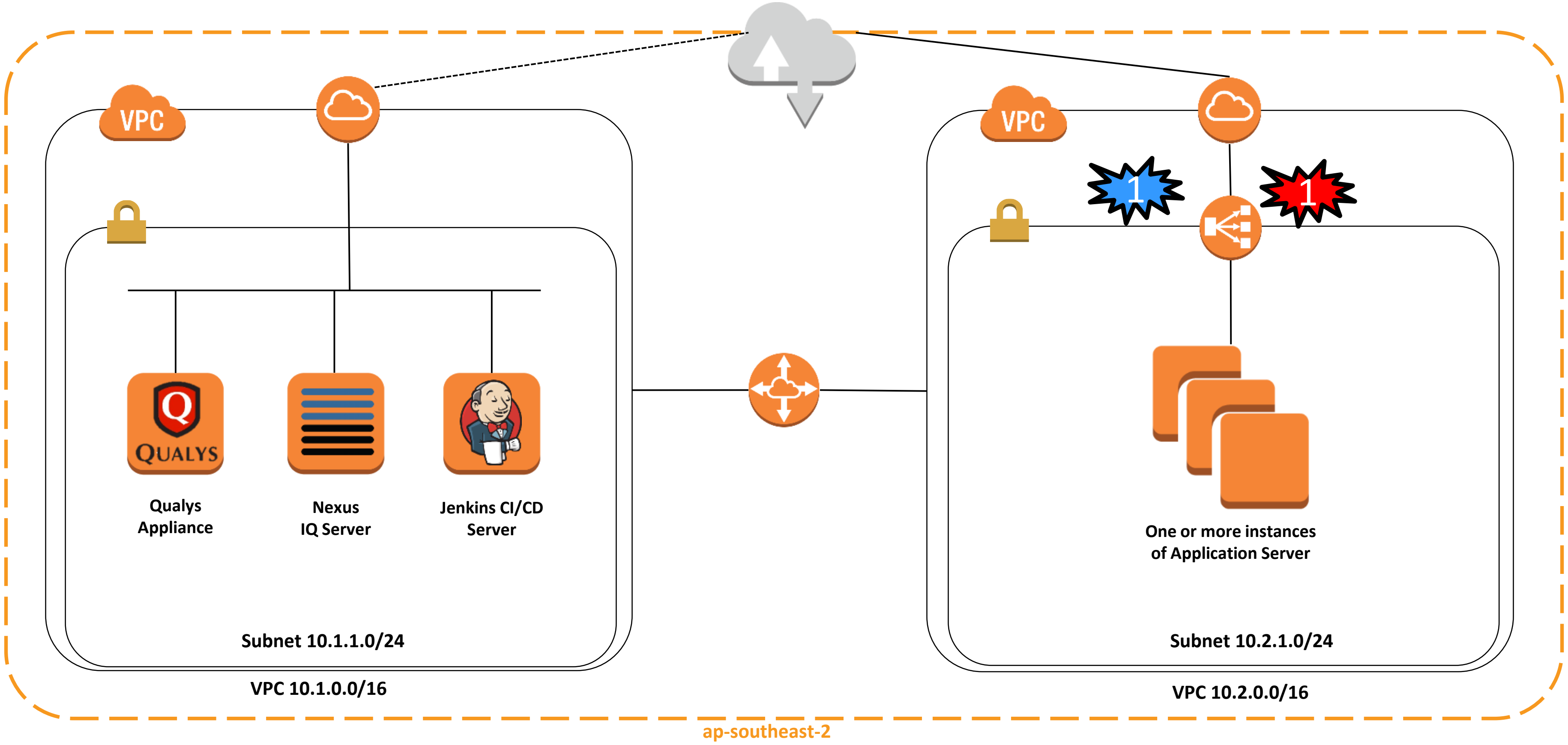
Attribute	Defence in Depth
Attack Window	• before a vulnerability is published • before a patched is available • before you can implement a corrective action
Additional Controls • Container Firewall	• application segmentation • whitelist of allowed container connections • policy for internal applications (web servers) prevent connections to external networks • prohibit direct connections to database/core

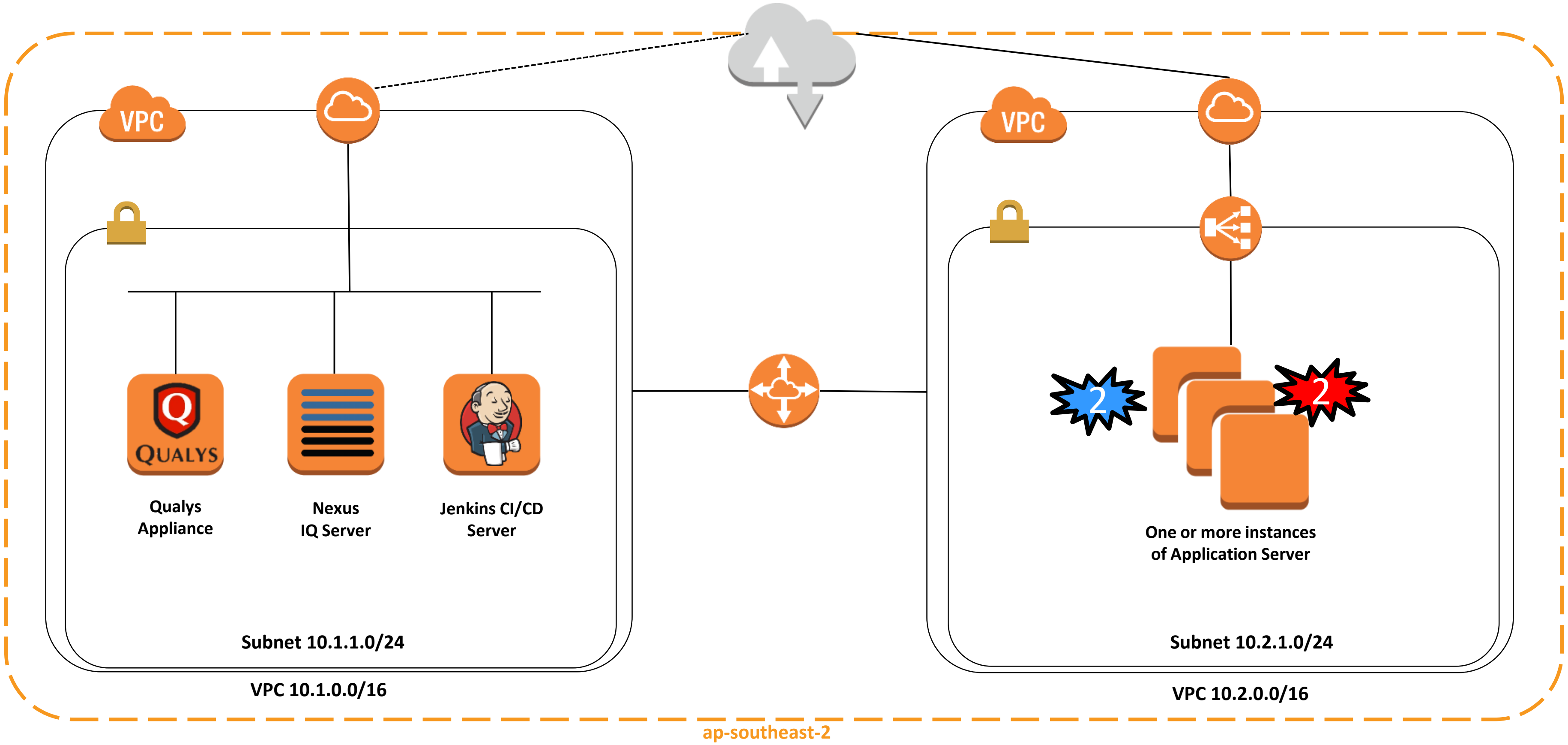
http://docs.aws.amazon.com/AmazonVPC/latest/PeeringGuide/vpc-peering-basics.html

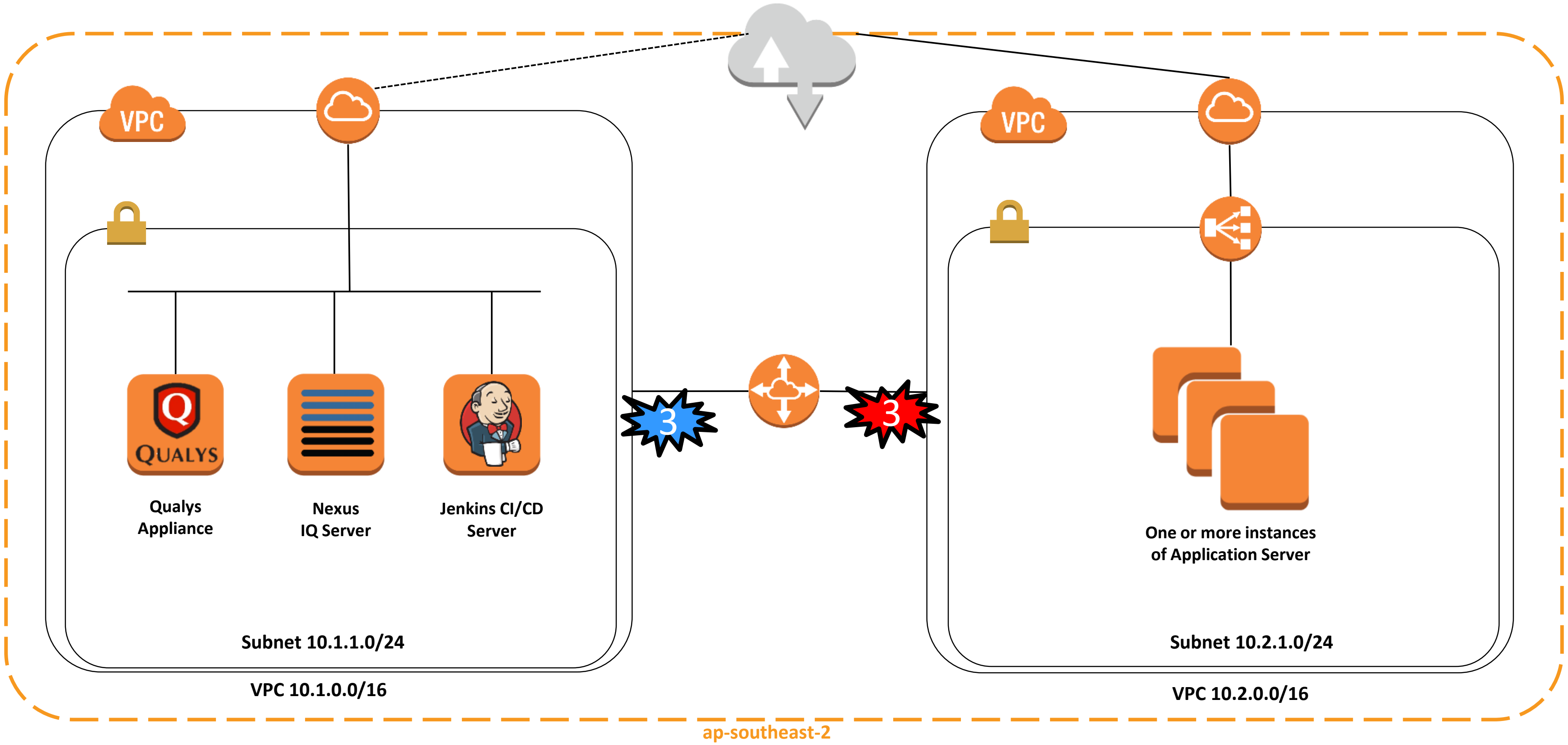


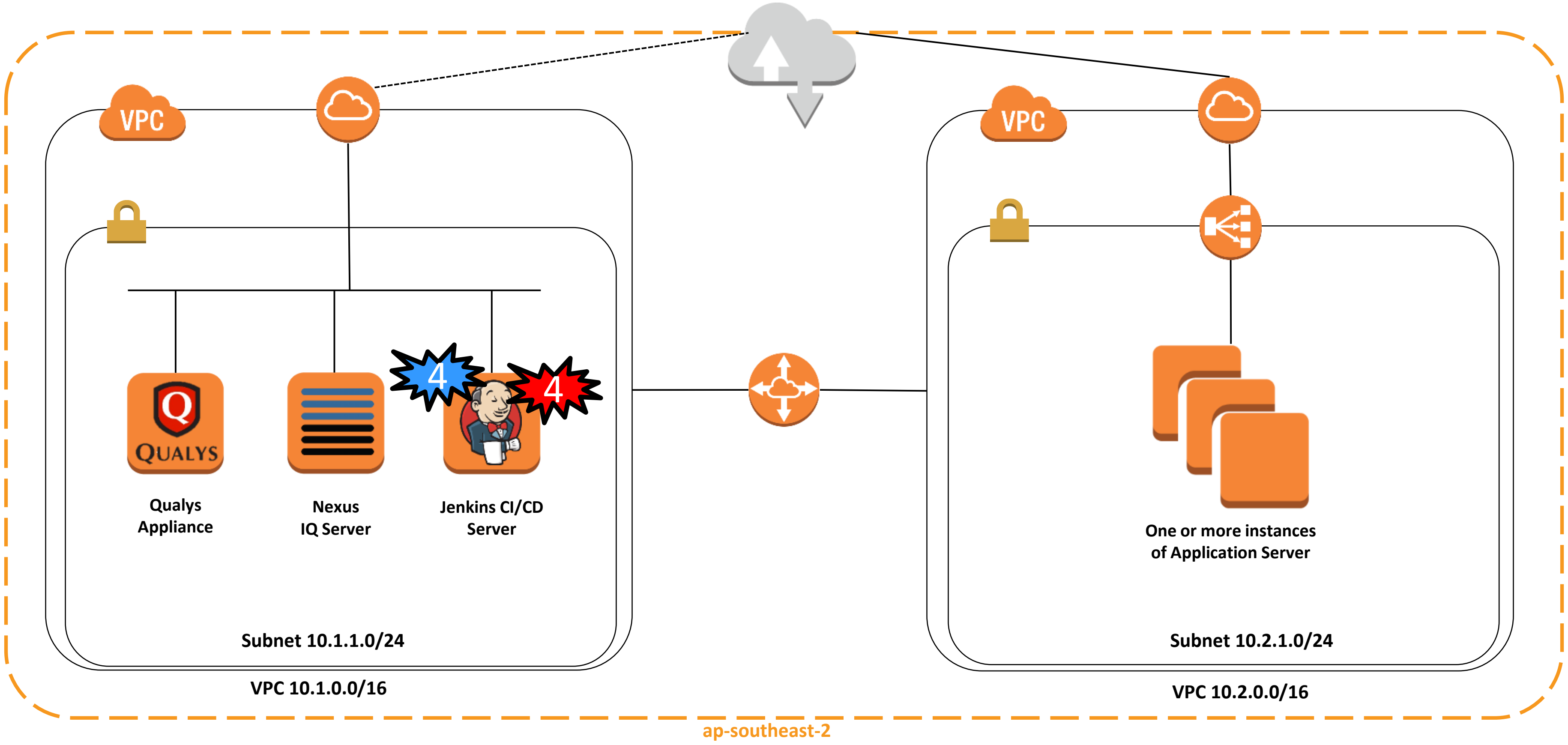
4. If required, update the security group rules that are associated with your instance to ensure that traffic to and from the peer VPC is not restricted. You can reference a security group from the peer VPC as a source or destination for ingress or egress rules in your security group rules.

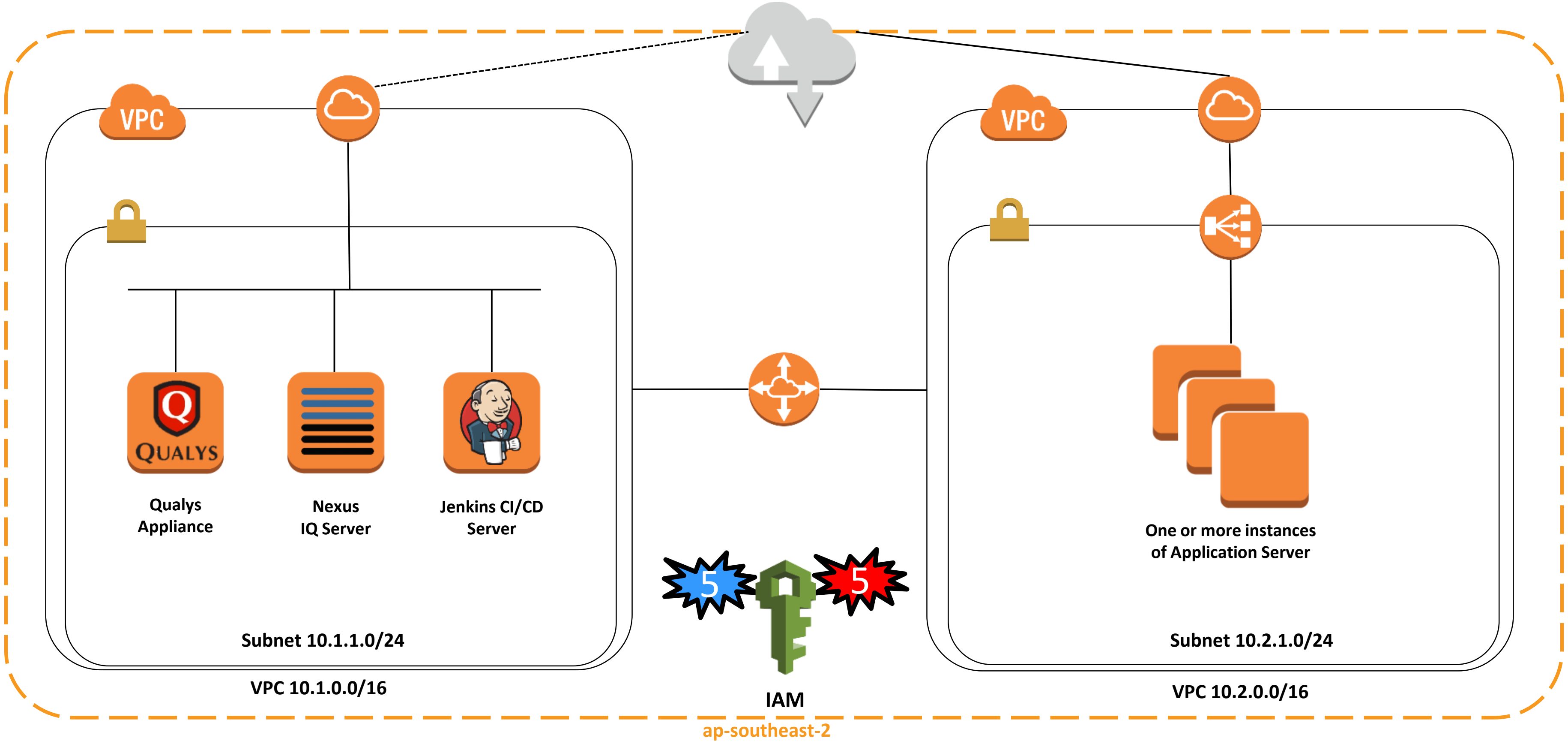
NO!

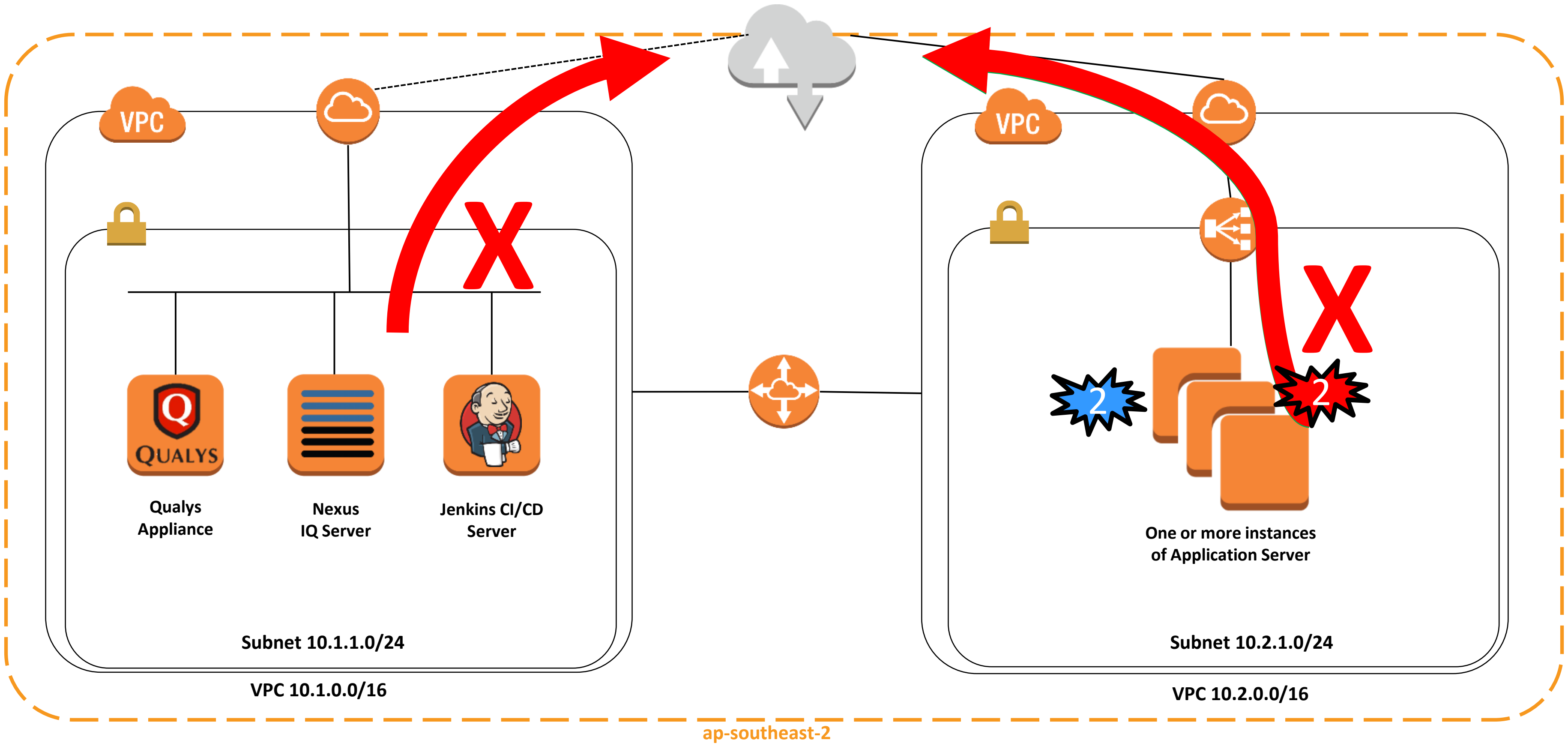

VPC
VPC
1
1
Qualys Appliance
Nexus IQ Server
Jenkins CI/CD Server
One or more instances of Application Server
Subnet 10.1.1.0/24
VPC 10.1.0.0/16
Subnet 10.2.1.0/24
VPC 10.2.0.0/16
ap-southeast-2

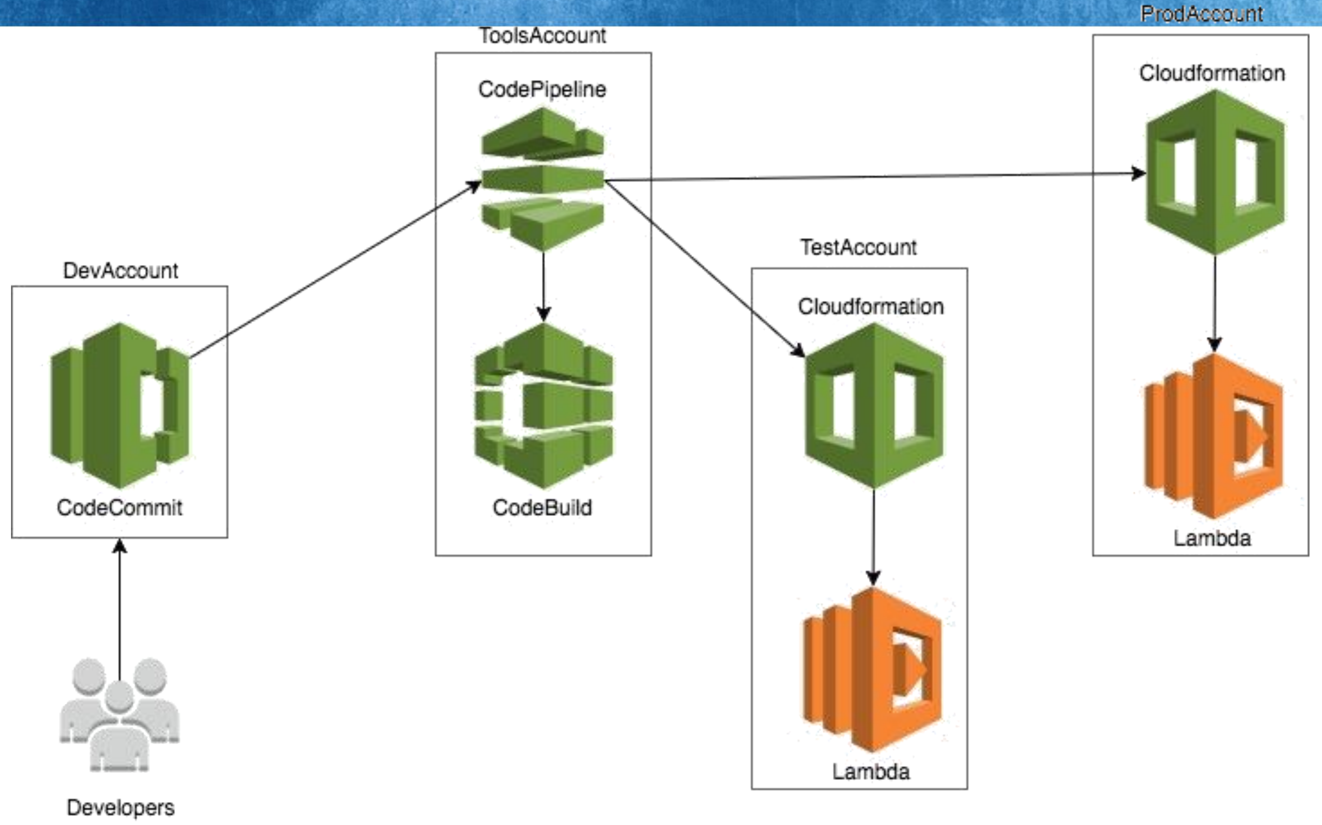

VPC
VPC
Qualys Appliance
Nexus IQ Server
Jenkins CI/CD Server
Subnet 10.1.1.0/24
VPC 10.1.0.0/16
2
2
One or more instances of Application Server
Subnet 10.2.1.0/24
VPC 10.2.0.0/16
ap-southeast-2


VPC
VPC
QUALYS
Qualys Appliance
Nexus IQ Server
Jenkins CI/CD Server
Subnet 10.1.1.0/24
VPC 10.1.0.0/16
3
3
One or more instances of Application Server
Subnet 10.2.1.0/24
VPC 10.2.0.0/16
ap-southeast-2


VPC
VPC
QUALYS
4
4
Qualys Appliance
Nexus IQ Server
Jenkins CI/CD Server
Subnet 10.1.1.0/24
VPC 10.1.0.0/16
One or more instances of Application Server
Subnet 10.2.1.0/24
VPC 10.2.0.0/16
ap-southeast-2

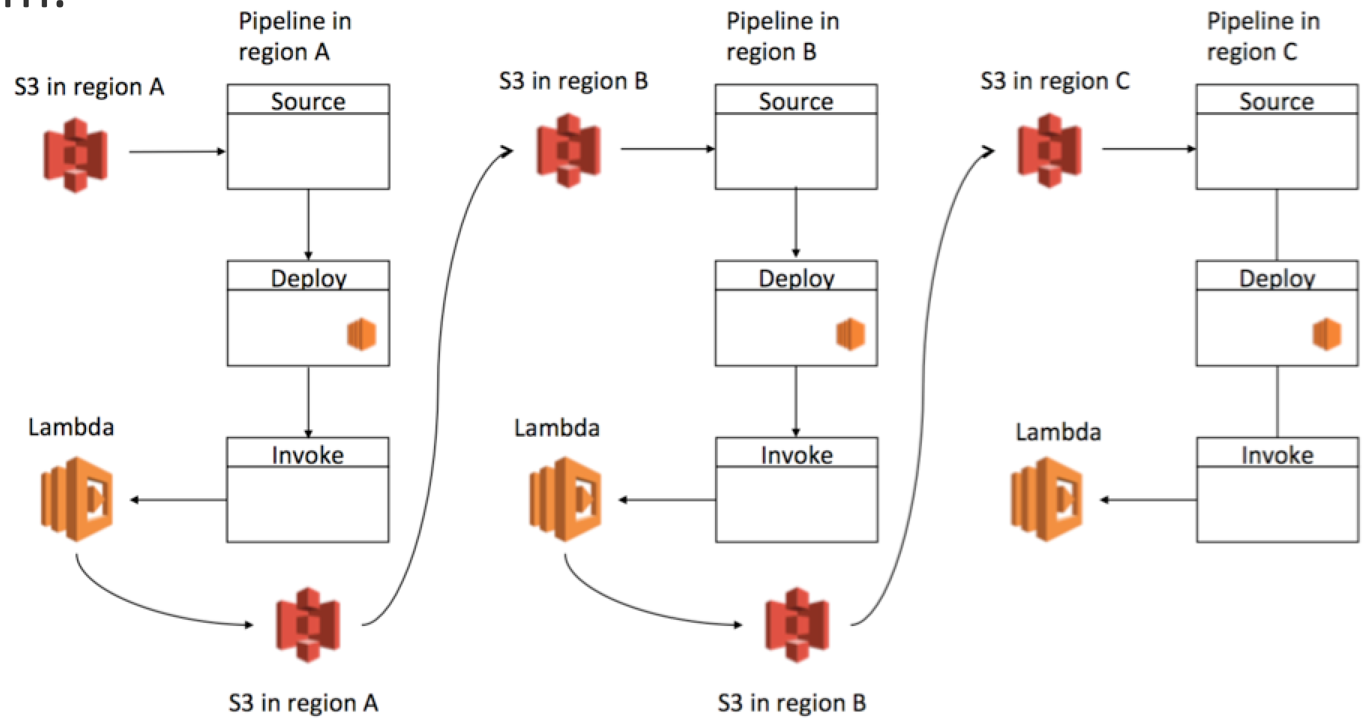

VPC
VPC
Qualys Appliance
Nexus IQ Server
Jenkins CI/CD Server
Subnet 10.1.1.0/24
VPC 10.1.0.0/16
5
5
IAM
ap-southeast-2
One or more instances of Application Server
Subnet 10.2.1.0/24
VPC 10.2.0.0/16


VPC
VPC
X
X
2
2
Qualys Appliance
Nexus IQ Server
Jenkins CI/CD Server
One or more instances of Application Server
Subnet 10.1.1.0/24
VPC 10.1.0.0/16
Subnet 10.2.1.0/24
VPC 10.2.0.0/16
ap-southeast-2



https://aws.amazon.com/blogs/devops/aws-building-a-secure-cross-account-continuous-delivery-pipeline/
https://d0.awsstatic.com/aws-answers/AWS_Multi_Account_Security_Strategy.pdf

- A successful processing of source code in all of its AWS CodePipeline stages will invoke a Lambda function as a custom action, which will copy the source code into an S3 bucket in Region B. After the source code is copied into this bucket, it will trigger a similar chain of processes into the different AWS CodePipeline stages in Region B. See the following diagram.



Time Line

Time Line

ID	Attack	Countermeasure Process	Countermeasure Technology
1	Vulnerability Identification	External Vuln Scanning Automation – extend to Continuous Monitoring	Qualys (VM + Cont Mon, WAS) Veracode (Dynamic)
1	Vulnerability Prevention (OS, Framework, Environment etc.)	Config Mgt Patch Mgt	Active: • IPS Passive: • Qualys (VM, Policy Compliance)
1	Vulnerability Prevention (First Party Code)	Security in SDLC	Active WAF RASP (e.g. Veracode) SDLC Veracode (Greenlight, Static)
1	Vulnerability Prevention (3rd Party Code)	Security in SDLC	Veracode (SCA) Sonatype

Time Line

ID	Attack	Countermeasure Process	Countermeasure Technology
2	Vulnerability Prevention (3rd Party Code)	Security in SDLC	Veracode (SCA) Sonatype
2	Shell Binding, Tools Download etc.	Restrict unsolicited outbound access	• Self-Healing / Tamper Resistance • Application Whitelisting • AWS Lambda Functions (DIY) • Dome9 Clarity Diagram • Dome9 Clarity VPC Log Review
2	Vulnerability Prevention	Configuration Management Patch Management	• IPS • Qualys (VM, Policy Compliance)
2	Vulnerability Prevention (First Party Code)	Security in SDLC	WAF RASP (e.g. Veracode) Veracode (Greenlight, Static)

Time Line

ID	Attack	Countermeasure Process	Countermeasure Technology
3	Pivot, Vuln Identification	Restrict unsolicited traffic intra-VPC, intra-Account, VPC-WAN etc.	Active Automation • Dome9 AWS Security Group Rule Tamper Resistance Visual • Dome9 Clarity Diagram • Dome9 Clarity VPC Log Review • Passive • Qualys VM + Cont Mon

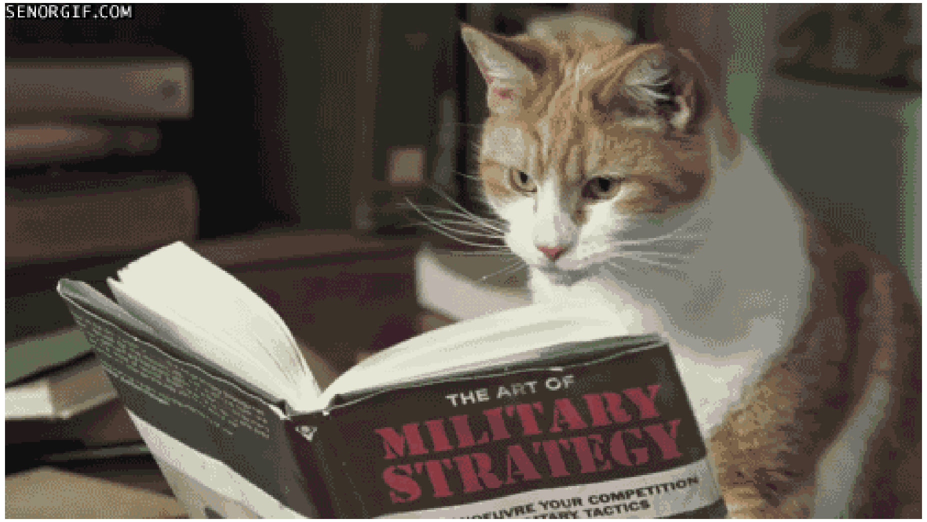
Time Line

ID	Attack	Countermeasure Process	Countermeasure Technology
4	Vulnerability Prevention (OS, Framework, Environment etc.)	As Per Previous • Depends on Vuln Type: • Config Mgt • Patch Mgt • Security in SDLC	Active: • IPS Passive: • Qualys (VM, Policy Compliance) SDLC • Veracode, Sonatype etc

Time Line

ID	Attack	Countermeasure Process	Countermeasure Technology
5	Cloud, Account Creation, Priv Escalation, Priv Abuse	Access Controls and Permissions • RBAC • Permissions on business need to know/use	Active • Dome9 IAM Protection • AWS Lambda Functions (DIY)

Time Line


SENORGIF.COM
THE ART OF
MILITARY
STRATEGY
...NOEUVRE YOUR COMPETITION
...ITARY TACTICS

Applying Security Automation in DevOps

- Look for opportunities in your SDLC to automatically identify defects earlier in the pipeline – i.e. “Shift Left”
- Examine all your security tools and investigate whether exposed API’s can be leveraged to provide automated control/feedback.
- Review your cloud based architecture for opportunities to apply automated checking of configuration and continuous monitoring.
- Remember to protect the “full stack” of tools, processes and technology in your DevOps pipeline. It’s not just about the output!

Improved DevOps Architecture Principles ….

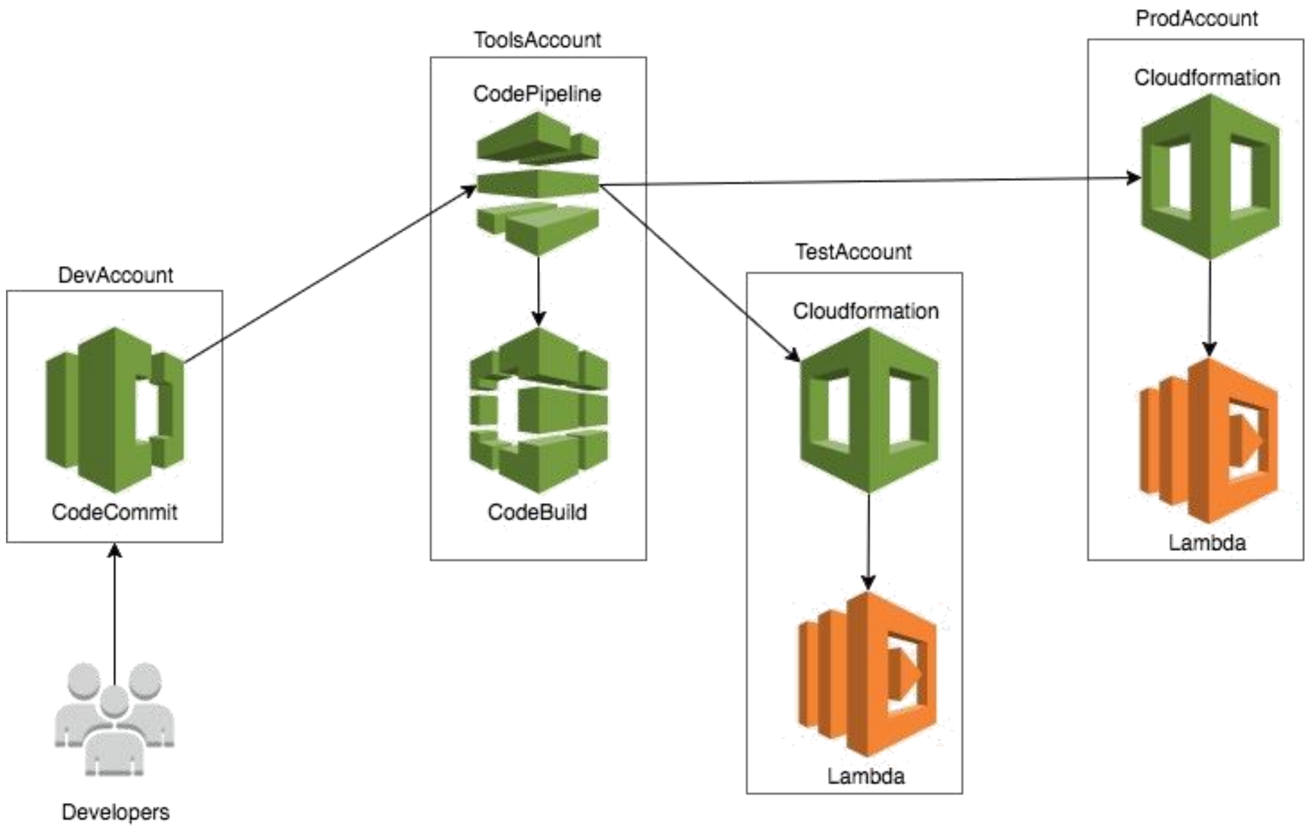
Isolation for Development,

Testing,

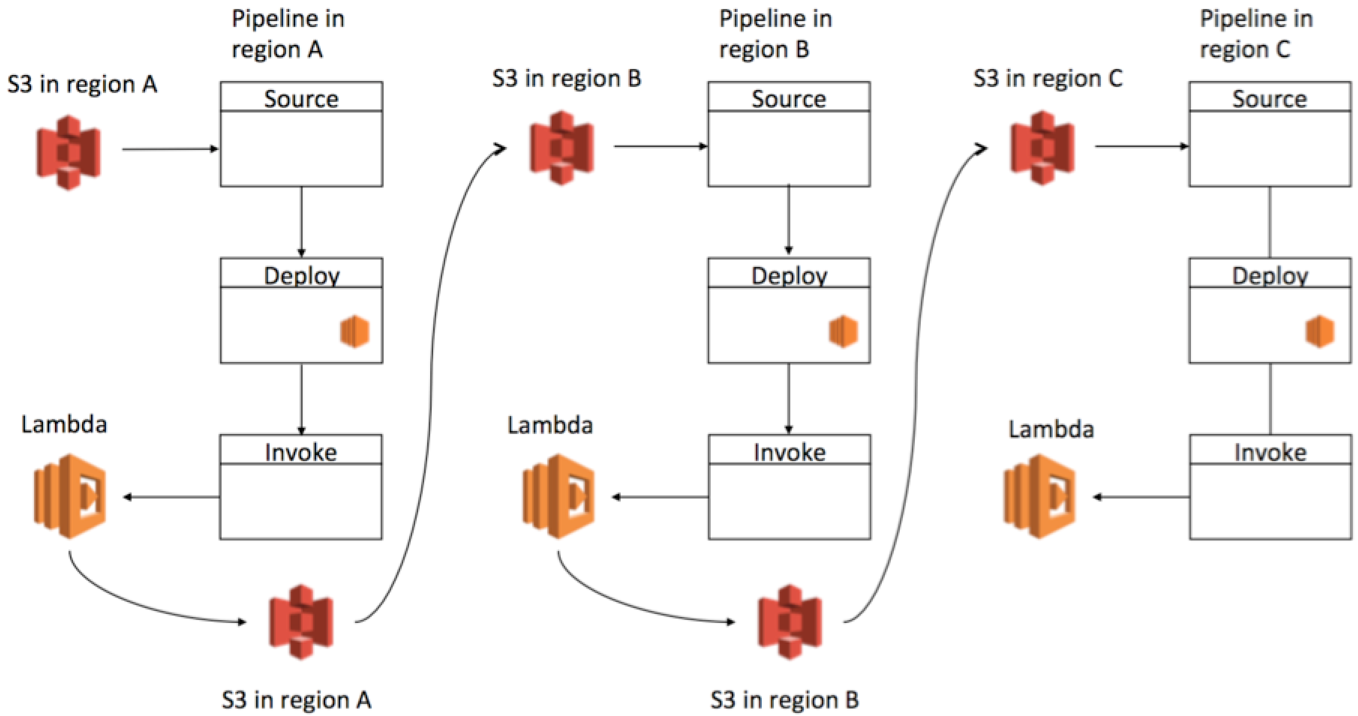
Deployment, etc

Use Multiple Accounts

Account	RBAC
DevAccount	• Developers check the code into repo • Store all the repositories as a single source of truth for application code. • Developers have full control over this account • Used as a sandbox for developers
ToolsAccount	• Central location for all the tools related to the org, incl CI/CD services • Developers have limited/read-only access in this account • Operations team has more control
TestAccount	• Applications using the CI/CD orchestration for test purposes deployed from this account • Developers & Ops team have limited/read-only access in this account
ProdAccount	• Applications using the CI/CD orchestration tested in the ToolsAccount deployed to production from this account • Developers & Ops team have limited/read-only access in this account


ToolsAccount
CodePipeline
CodeBuild
ProdAccount
Cloudformation
Lambda
DevAccount
CodeCommit
TestAccount
Cloudformation
Lambda
Developers

- A successful processing of source code in all of its AWS CodePipeline stages will invoke a Lambda function as a custom action, which will copy the source code into an S3 bucket in Region B. After the source code is copied into this bucket, it will trigger a similar chain of processes into the different AWS CodePipeline stages in Region B. See the following diagram.



Thank you

T: 1300 922 923
T: +61 (0) 2 9290 4444
F: +61 (0) 2 9290 4455
info@senseofsecurity.com.au
www.senseofsecurity.com.au