

RSA® Conference 2019

San Francisco | March 4–8 | Moscone Center



SESSION ID: CSV-W02

Protecting the Cloud with the Power of Cloud



Jay Kelath

Head of Product Security

Pranav Patel

Lead Security Engineer

#RSAC

What are we going to talk about...?

DevSecOps

- How we “SaaS-ified” our on-prem security tools with Docker and DevSecOps
- Scaling security with help of Cloud enables Security automation.

Self-healing Cloud Insecurities

- Discuss various insecurities in Cloud environments
- How our Cloud based tooling helps reduce risk
- How Automation and Self-healing works in our Cloud environment

Open Source Technologies : Takeaway -- start your DevSecOps journey from Day 1

- Open source tools enables security
 - Dow Jones Hammer
- Embedding into Pipeline
 - Project Bravos

Quick Flashback....

- Traditional Security tools
- Manual Reviews
- Lack of visibility and scalability

Challenges

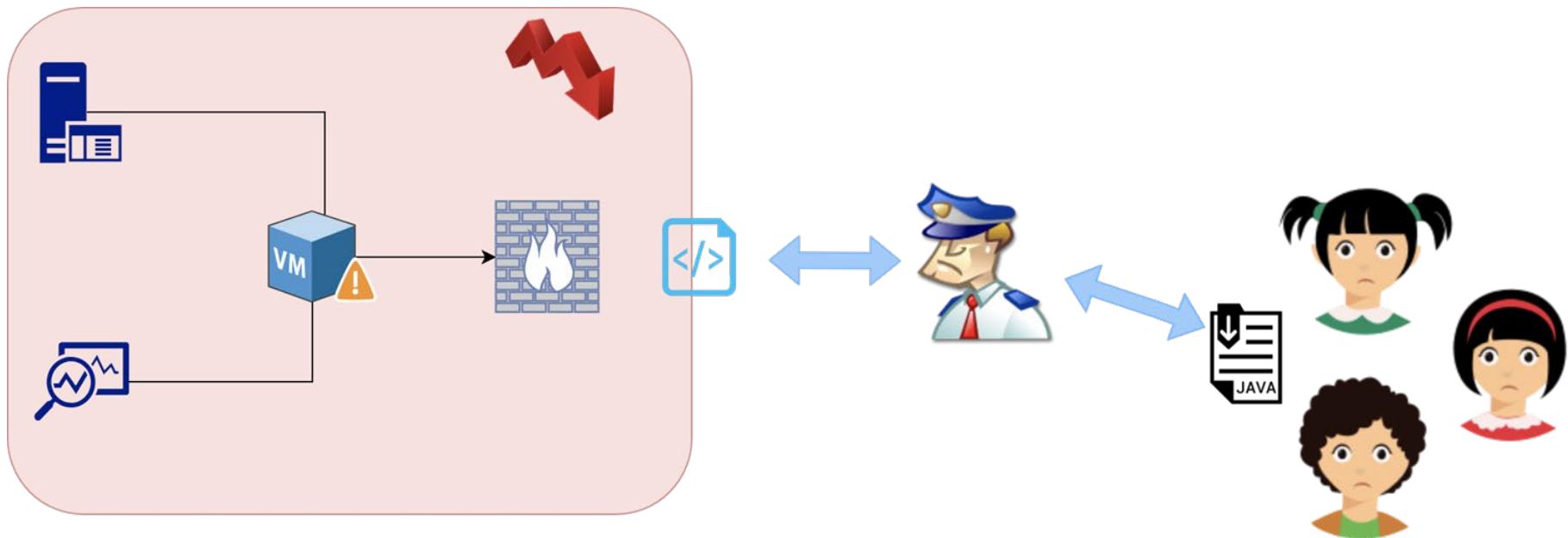
Legacy

Technology Sprawl

People

Process

Traditional Security



RSA® Conference 2019

“Don't Let a Good Crisis Go to Waste”

A large, abstract graphic in the background features a dense web of thin, light blue lines connecting numerous small, semi-transparent blue dots. These lines and dots are concentrated in several distinct, curved bands that sweep across the frame from left to right, creating a sense of motion and connectivity.

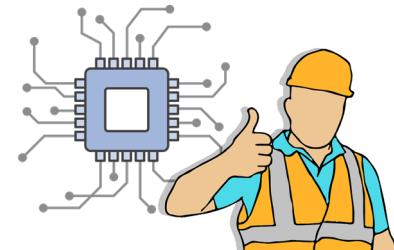
DevSecOps

Solve a specific problem in an automated manner with well defined People, Process, Technology actions and extensive, actionable reporting

Setting priorities right...

Risk vs Reward

End Goal: Reduce Risk



Think about....

Technology

API Driven

Scalable, Tunable

False Positives

KISS

Process

Use Existing Process

Feedback Loop

High Quality Report

Optimize

People

Support Model

Build Trust

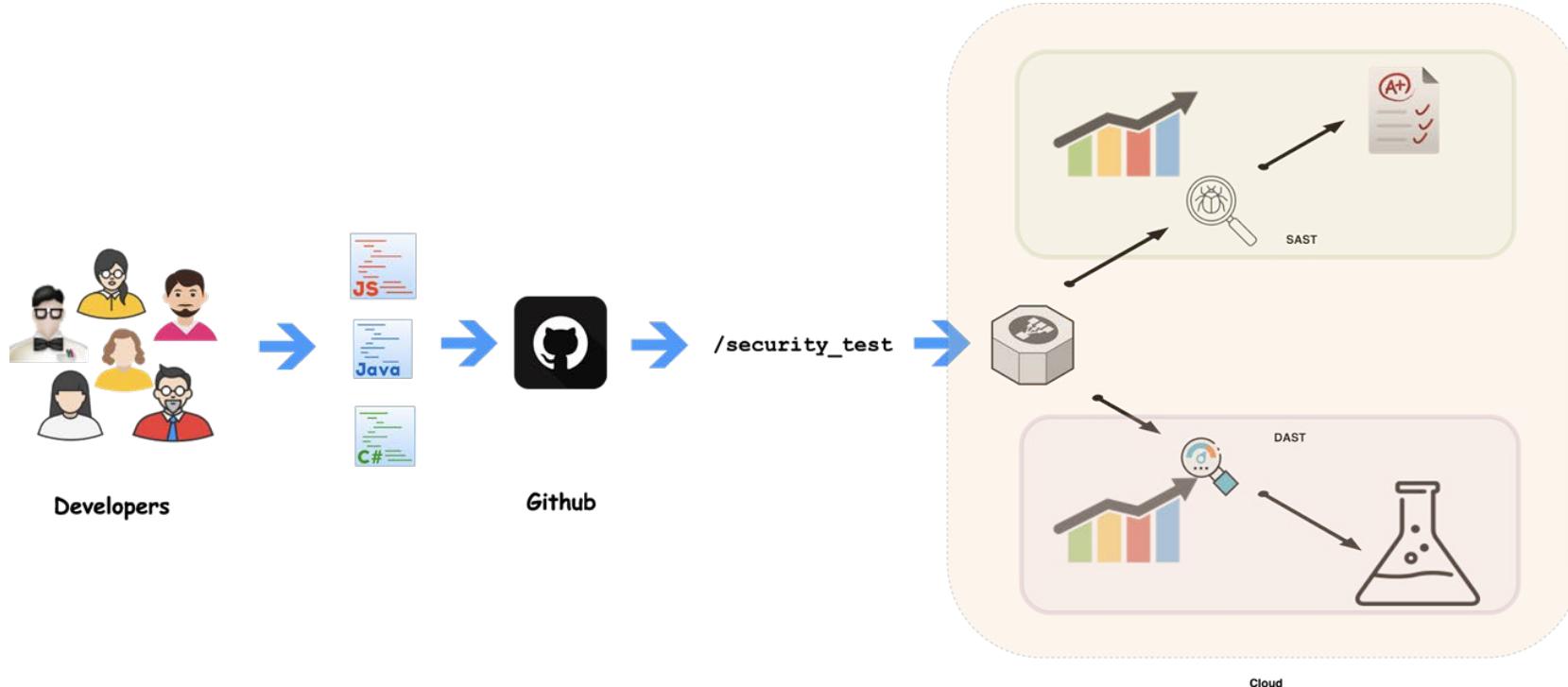
Developer support

Continuous Security



<https://memegenerator.net/instance/82149927/saltbae-roast-security>

DevSecOps way....



Dow Jones Hammer

Open Sourced : <https://github.com/dowjones/hammer>



Cloud migration journey

- Agile
- Success story out of migration
- “Lift & Shift” applications
- Cost management
- “Fail forward fast...”
- “Act now, apologize later....”
- “Move Fast Break Things....”

LETS MOVE TO CLOUD



memegenerator.net

<https://memegenerator.net/instance/80795081>

Security ??

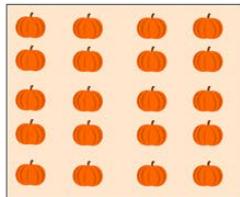
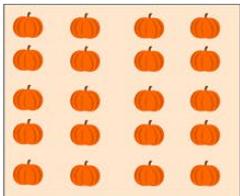
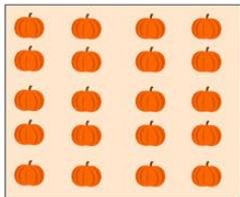
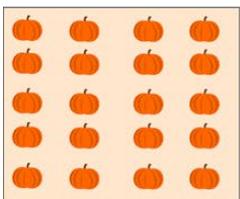
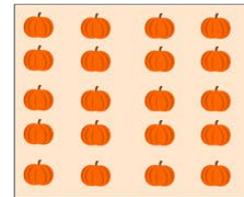
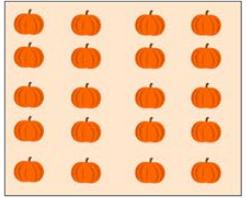
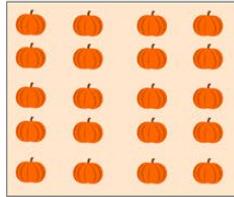
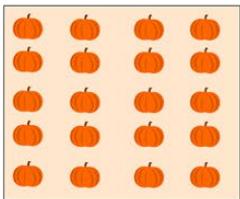
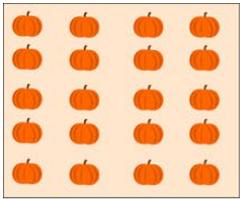
- Legacy cloud (...already?)
- Traditional security tools do not measure up
- Multi-Account visibility and controls
- Shared Responsibility Model
- Change in Landscape

“While moving fast and breaking things,
We forget to fix things!!!”



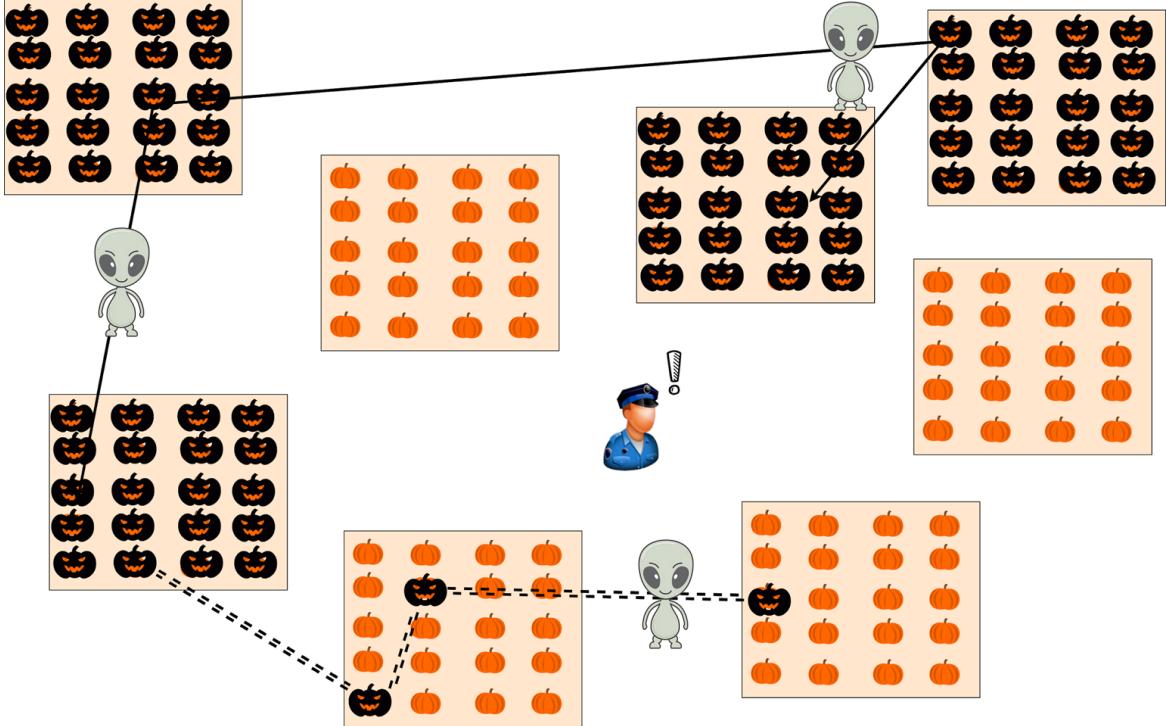
<https://memegenerator.net/instance/80794879>

Multi-account sprawl

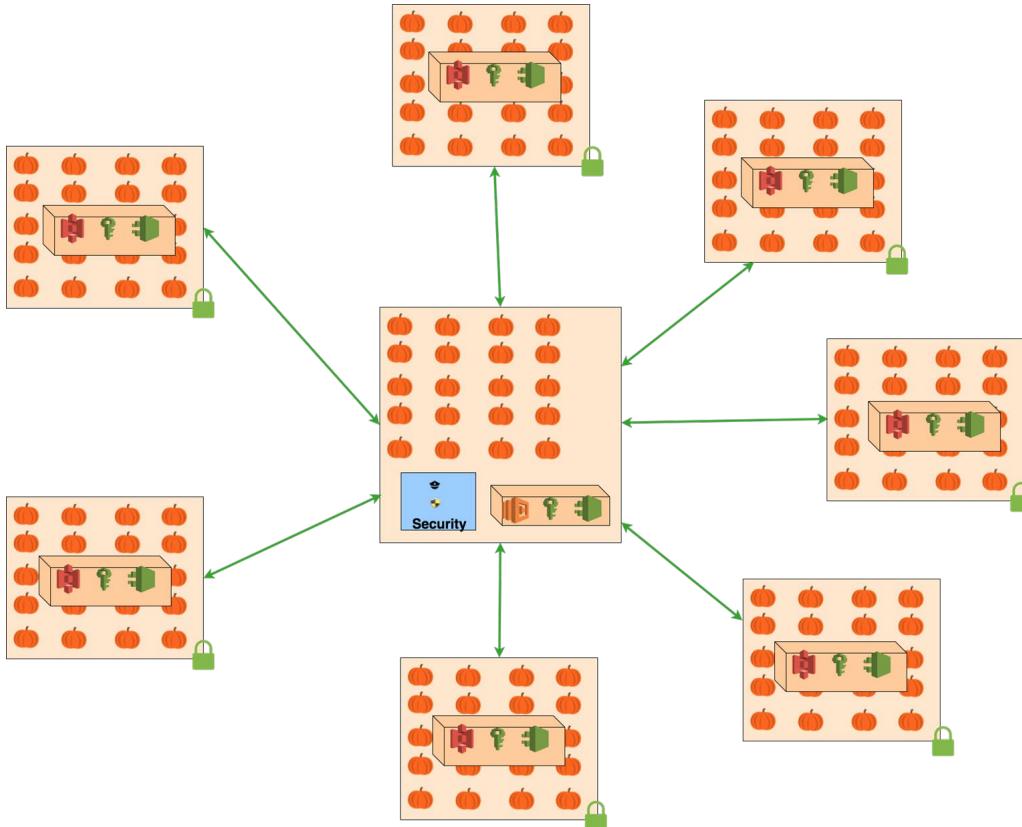


Multi-account sprawl

- Lack of visibility
- Scalability
- “Someone is looking into it...”

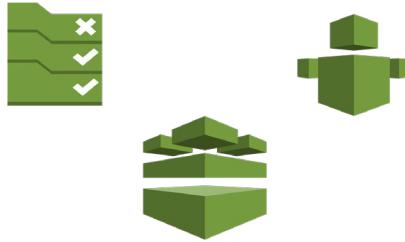


Multi-account growth : Ideal State



Security: Defense-in-depth

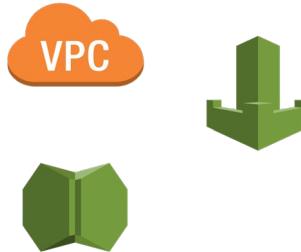
Detective Controls



Proactive Controls

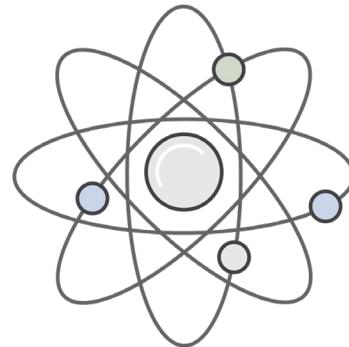
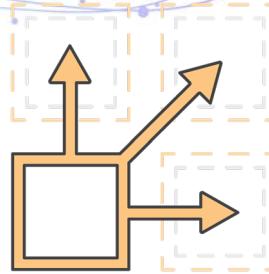
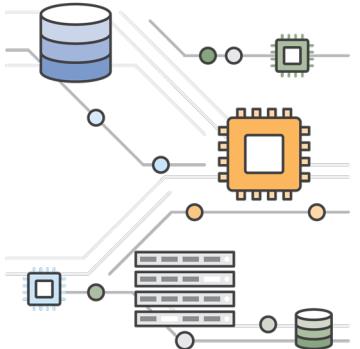


Reactive Controls



Our Solution

- Automate
- Scalable
- Self-service
- Auditable



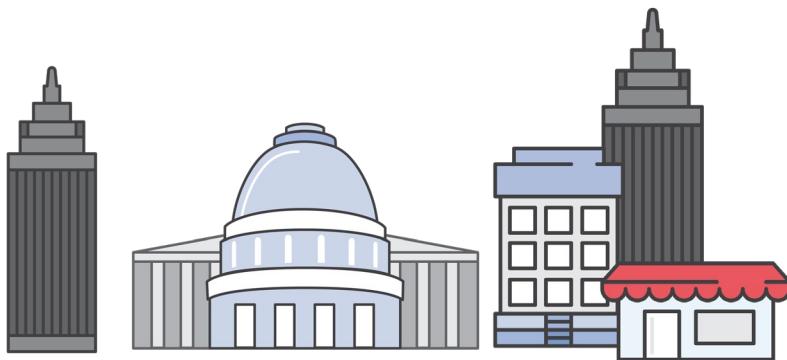
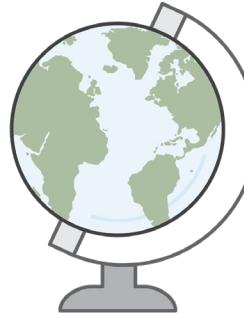
Hammer... why?

- Cloud Infrastructure visibility
- Easily pinpoint MY product's security issues
- Tailored reporting, save analysis time
- “Auto-fix” misconfigurations with ability to rollback



Consumer

- Multi-account customers
- Decentralized development organizations
- Multiple business units



Hammer: What Does it Solve?



Public Instances with Admin IAM Policies
Exposed EC2 Instances
Docker on EC2
ECS



Exposed RDS instances
Unencrypted RDS
Public RDS Snapshots

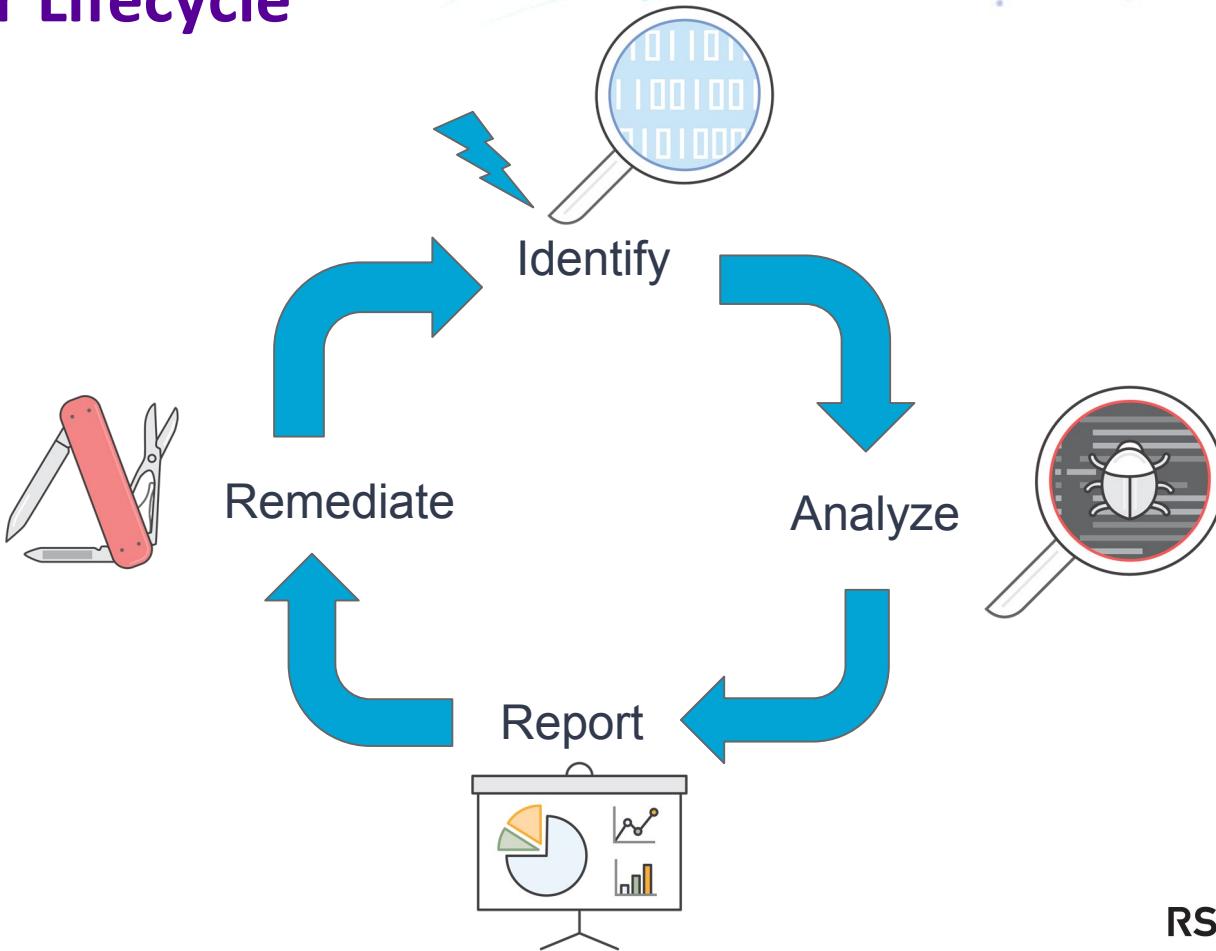


Public S3 buckets
Unencrypted S3 buckets
Public S3 bucket Policy

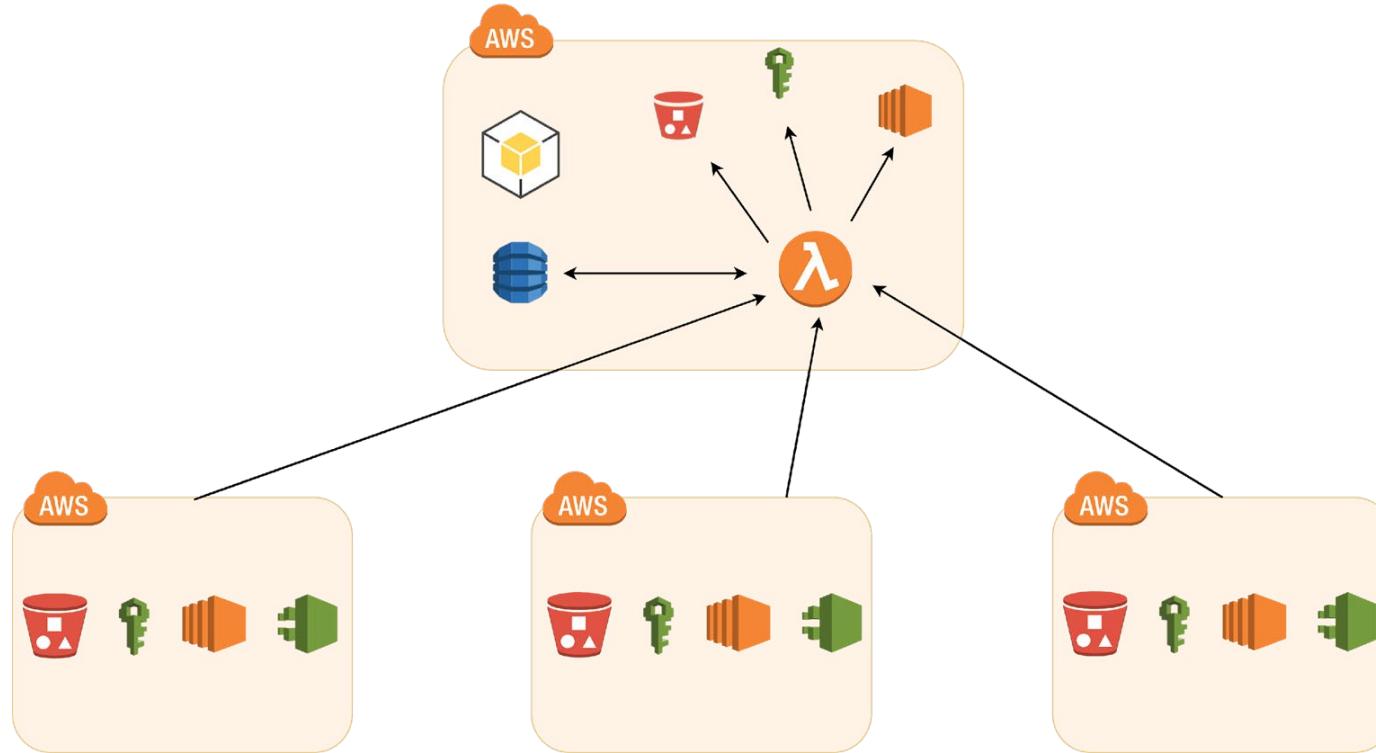


Unused IAM Keys
Stale IAM Keys (not rotated)

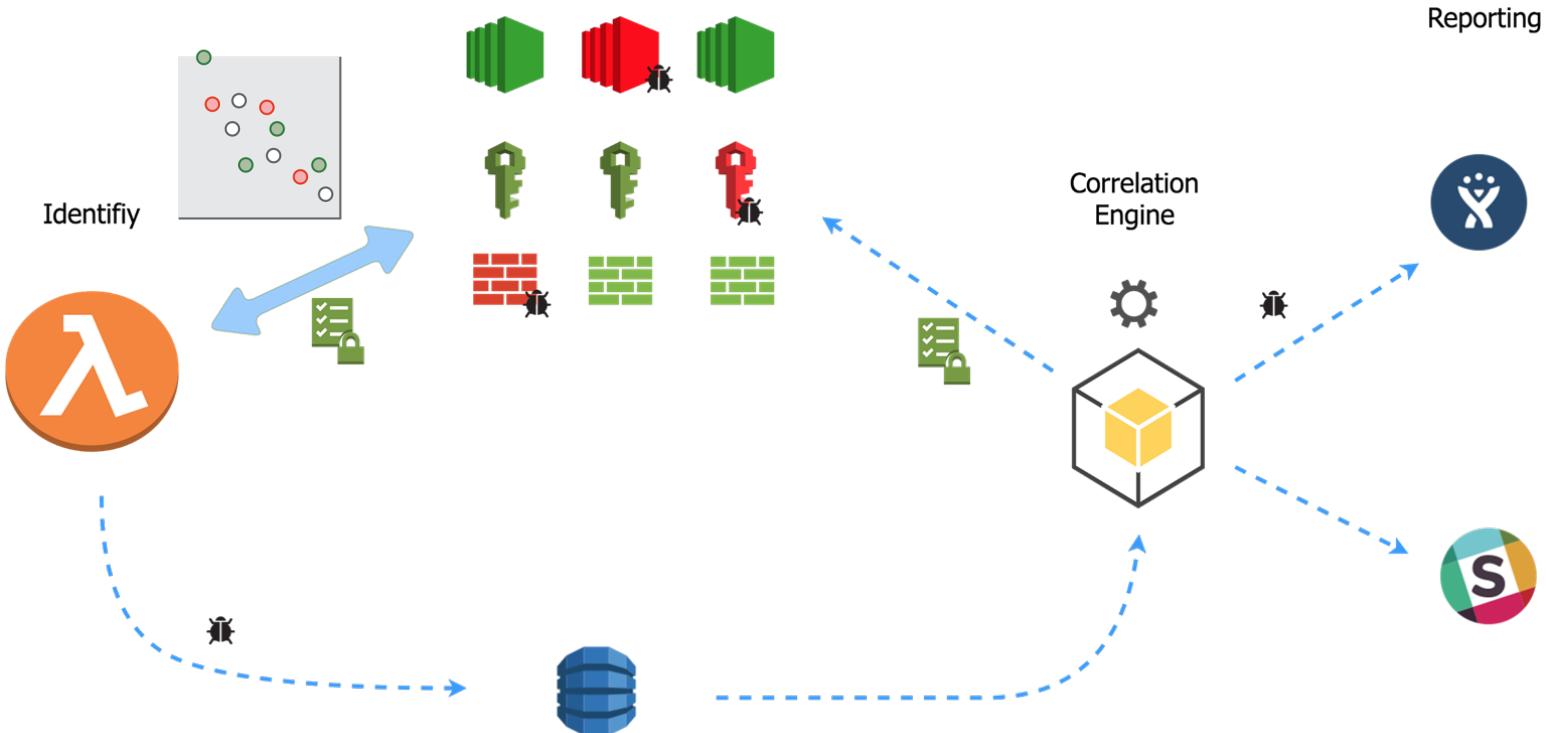
Hammer Lifecycle



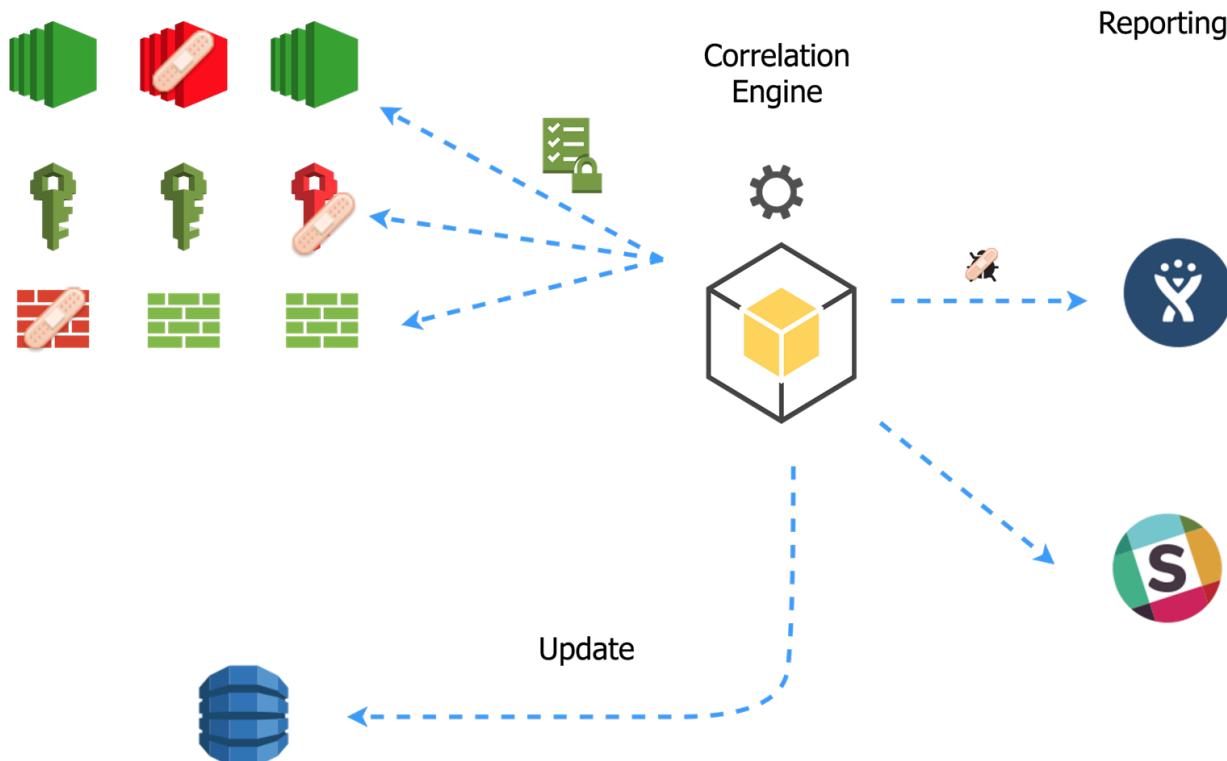
Architecture



Hammer - how does it work?



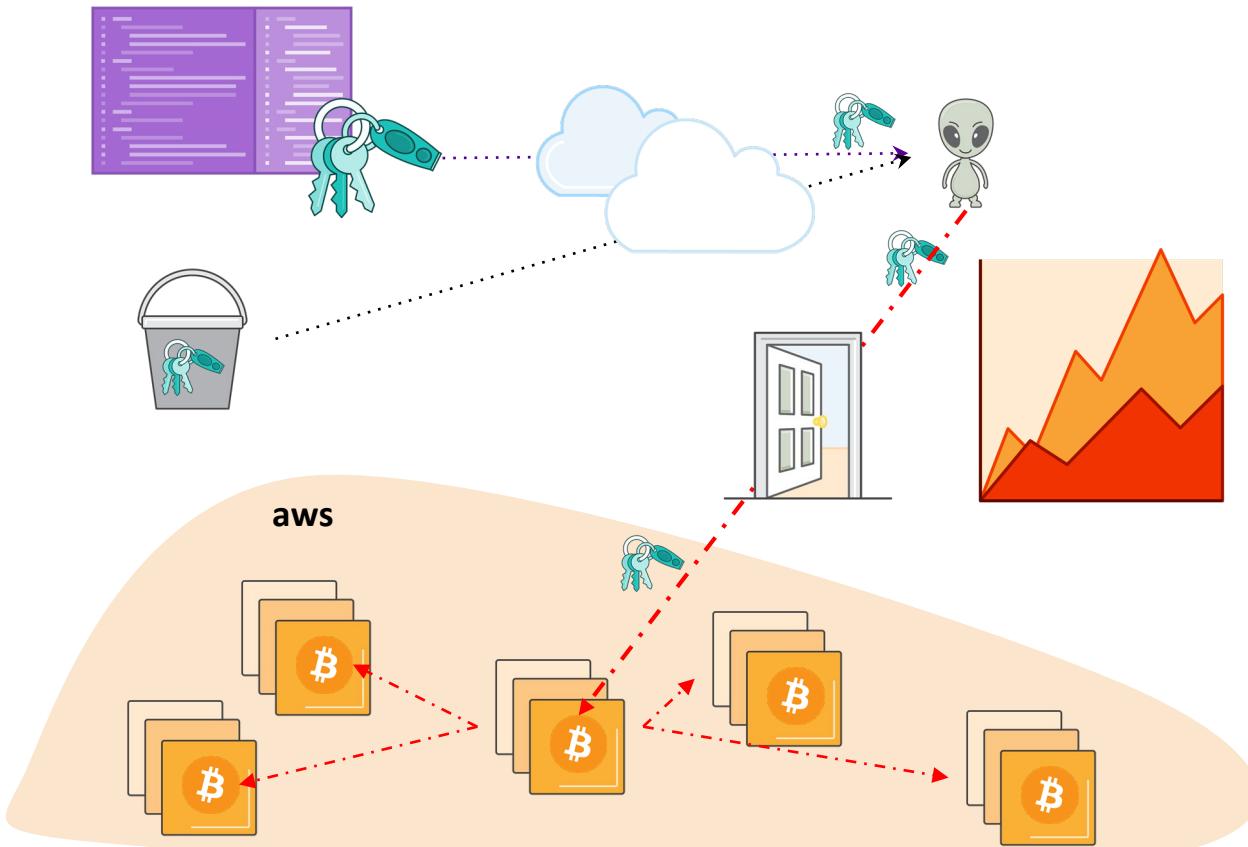
Hammer - /auto-fix



Case Study 1: Protection from bitcoin miners.....

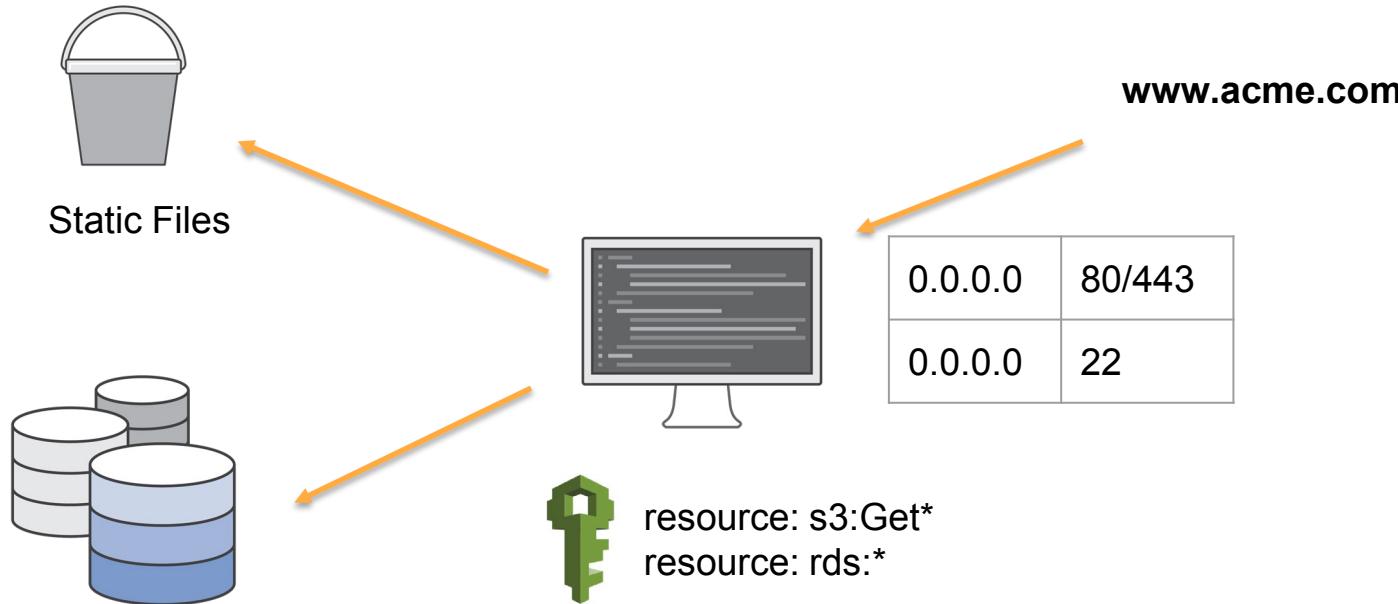
- Stale & Exposed keys in Code
- Stale & Exposed keys in public bucket
- “Exes” with Keys

Action :: Just Deactivate unused Keys



Case Study 2: Exposed Instances

Action : Lock down Non-Web Ports to Private
DMZ



Case Study: Example 1

SECVULN / SECVULN-7775

Intranet open security group 'int_p' in 'le_ec2' in 'hpdhj-stack-AWSEBSe'

Ec2 Instances:

Instance ID	State	Private Ip Address	Public Ip Address	Owner	Business unit	Product	Component	Subnet
i-033C1000000000064a	stopped	10.201.00.117	-	Sl...@dowjones.com	djs...	tsvc	othr	private
i-070200000000001028	running	10.201.70.77	34.22...	St...@dowjones.com	djs...	tsvc	othr	private

Instance Role Unsafe Policies:

Instance Id	Role Name	Policy Name	Unsafe actions
-------------	-----------	-------------	----------------

Risk Rating: High

Description

Security group has EC2 instances in private subnets and allows access from some definite public ip address, and make them vulnerable to blind injection based on the internet incoming, traffic will reach instance via an IGW.

From Port	To Port	Protocol	CIDR	Registrant
22	22	tcp	172.0.0.0/8	AT&T Internet Services (SIS-80)

Threat: Open access within the network not only provides unrestricted access to other servers but increases the risk of hacking, denial-of-service attacks, loss of data if attacker gains access to the services within the network. CY4L

Risk: High

DOW JONES

Auto-Remediation Date: 2018-10-02

Case Study : Example 2



hammer APP 12:55 PM

Discovered Intranet open security group 'riv... in
'DJLZ - **AWS Account Name/ Number** 45318' account [
(<https://is tickets.dowjones.com/Prod-123> 23)

22/tcp

0.0.0.0/0



hammer APP 1:55 PM

Closing resolved security group
0e6f3ffb6cf568cb0' issue in 'DJL **AWS Account Name/ Number**
account, 'us-east-1' region
(<https://tickets.dowjones.com/Prod-123>)

hammer-prod - Nov 28th, 2018 View in channel



hammer APP 11:56 AM

Security group 'cc-ssh / sg-cc9ac9a9' issue is changed in 'DJ -
AWS Name/ AWS Account Number 9' account, 'us-east-1'
region (<https://i ticket.dowjones.com/Product-1244>)

22/tcp

172.0.0.0/8 [AT&T Corp. (AC-3280)]

RSA®Conference2019

Apply : From day 1

<https://github.com/dowjones/hammer>

Hammer : Service Matrix

Services	Identification	Reporting	Remediation
Public S3 (ACL)	√	√	√
Public S3 (bucket policy)	√	√	√
Inactive-IAM-keys (older than 180 days)	√	√	√
IAM-key-rotation (older than 2 years)	√	√	[Roadmap]
Insecure Services (database, rdp, ssh, telnet ports)	√	√	√
Cloud Trail Disabled	√	√	n/a
Unencrypted EBS Volumes (Compliance Accounts)	√	√	n/a
Public EBS Snapshots	√	√	√

Hammer : Service Matrix

Services	Identification	Reporting	Remediation
[RDS]Public Snapshots	√	√	√
[RDS]Security Groups	√	√	√
[ECS] Security Groups	√	√	√
[SQS] Insecure config	√	√	√
Cleanup - Inactive [IAM] keys	√	√	√
[IAM] Privilege Escalation (IAMadmin roles)	√	√	√
Public AMIs	√	√	√

Lessons Learnt

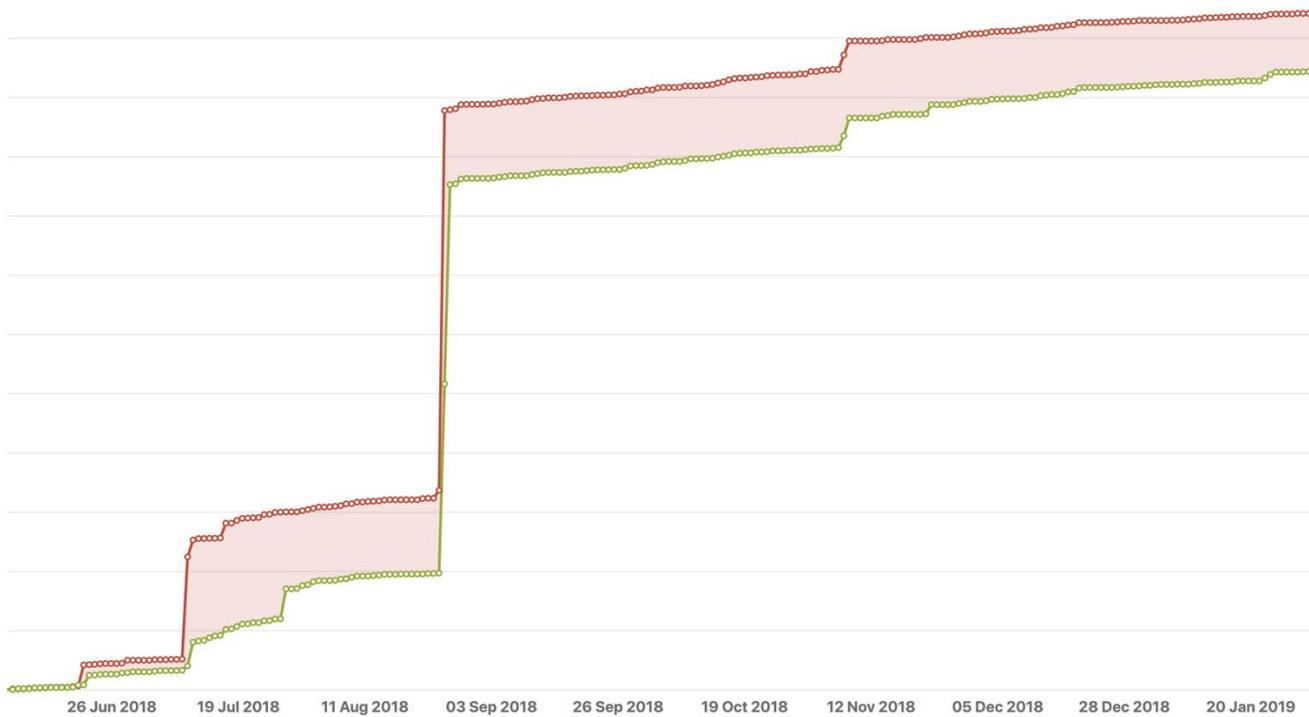
What worked..??

- Scales over 100+ AWS Accounts
- Integrates to Dev ecosystem
- Easy to plug and play
- Tonnes of insecurities fixed
- Low to minimal Impact

What did not work..??

- Event Driven
- AWS Lambda execution limits
- Server elements
- Limited security checks

Key Trends....



Scaling security and embedding into pipeline

<https://github.com/dowjones/sast>

Project Braavos : CICD Integration

Sensitive information in code
Continuous Security

Static code analysis

Risky function calls

Open Source Vulnerability Checks

Security test cases

Dynamic Security Tests

OWASP Example

Dev & AppSec Tool Integration



klocwork
a Rogue Wave Company



VERACODE



VERACODE

Gitrob



RAPTOR



VERACODE



Chef Audit Mode



Code



Manage



Store



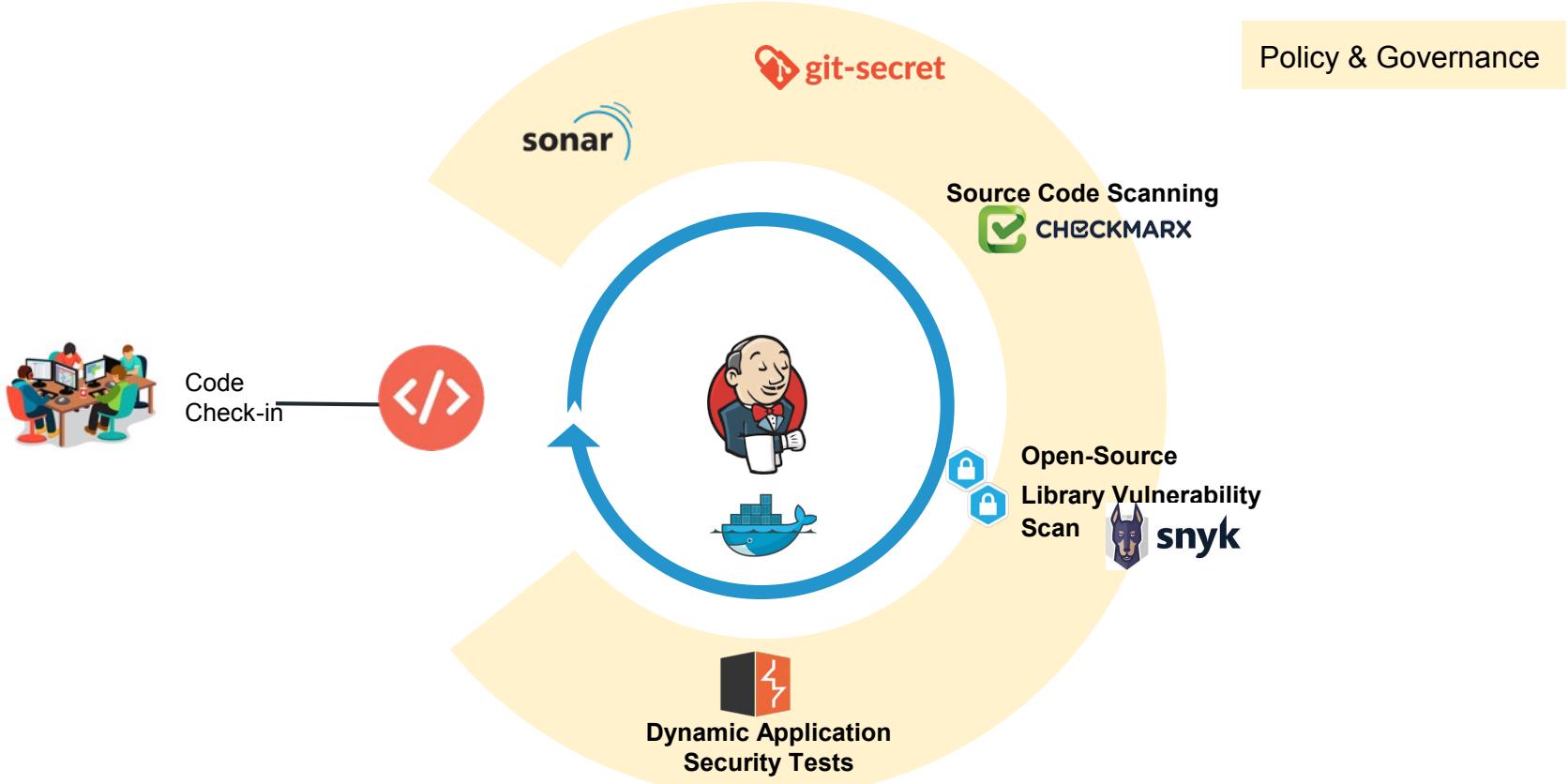
Build



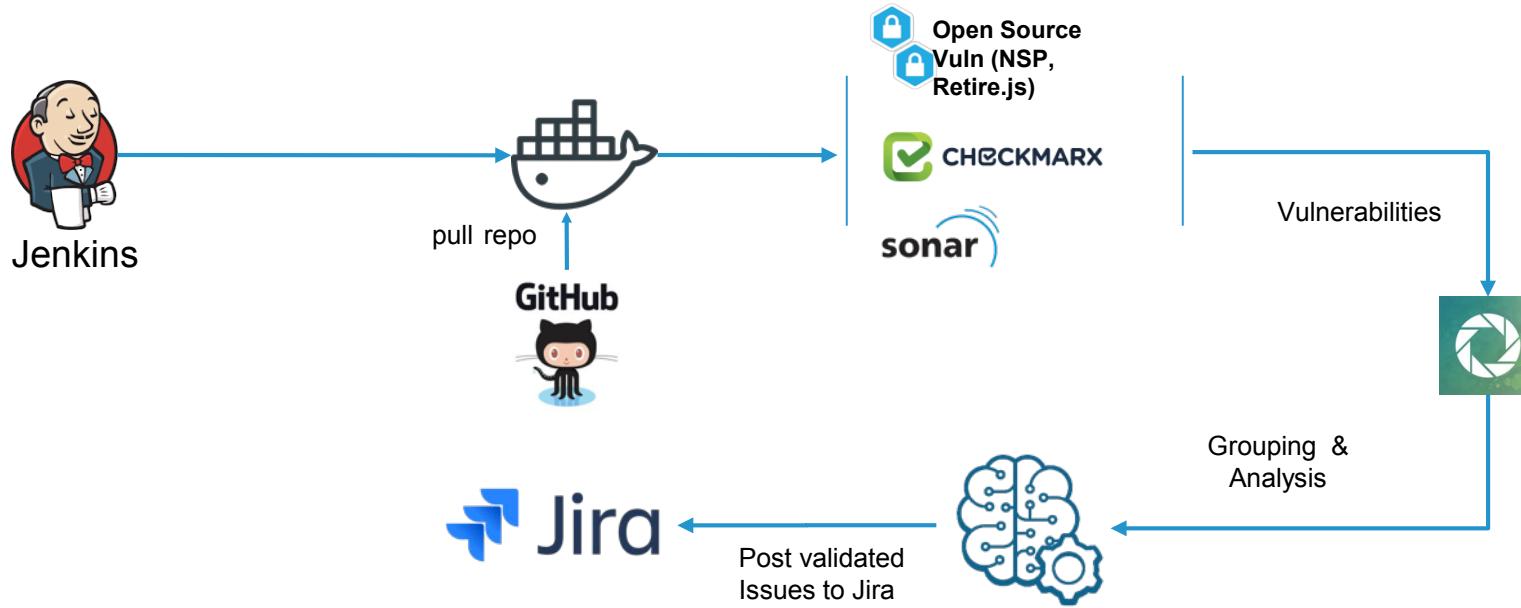
Deploy

Ref: https://www.owasp.org/index.php/OWASP_AppSec_Pipeline#tab=Pipeline_Tools

Our Solution : project bravos

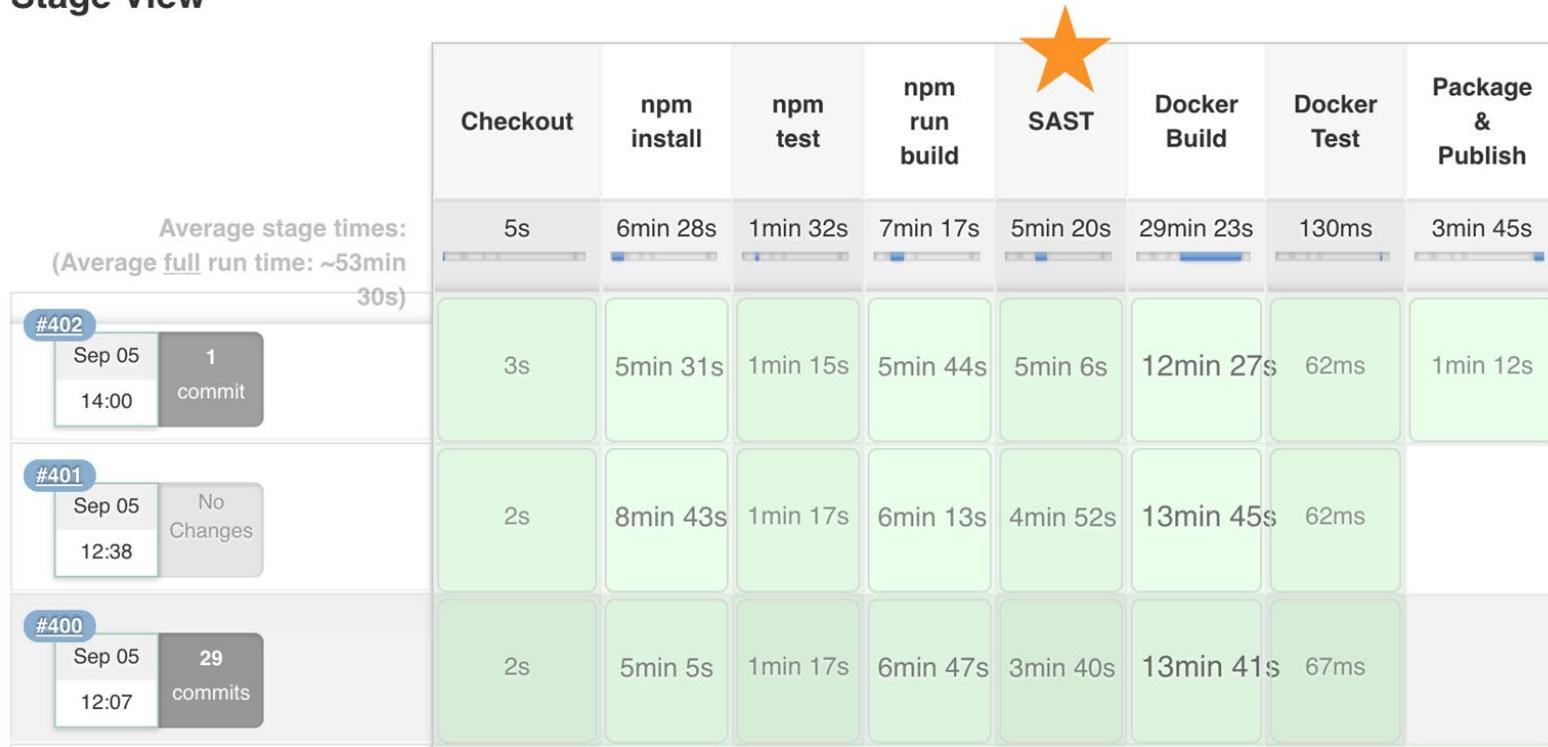


/continuous_security



/continuous_security: Jenkins Pipeline

Stage View



/continuous_security: sast(static code security testing)



checkout project	Check Checkmarx	Check NPM	Check Secrets	Check Retire.Js	Check SonarQube	publish reports
6min 3s	2min 34s	23s	2s	56s	3s	12s
6min 44s	1min 52s	21s	2s	56s	28ms	13s

Lessons Learnt

Easy to Integrate

Tune! Tune! Tune!

Tech needs to match the Tool

Incremental Scans FTW

More accessible to Devs

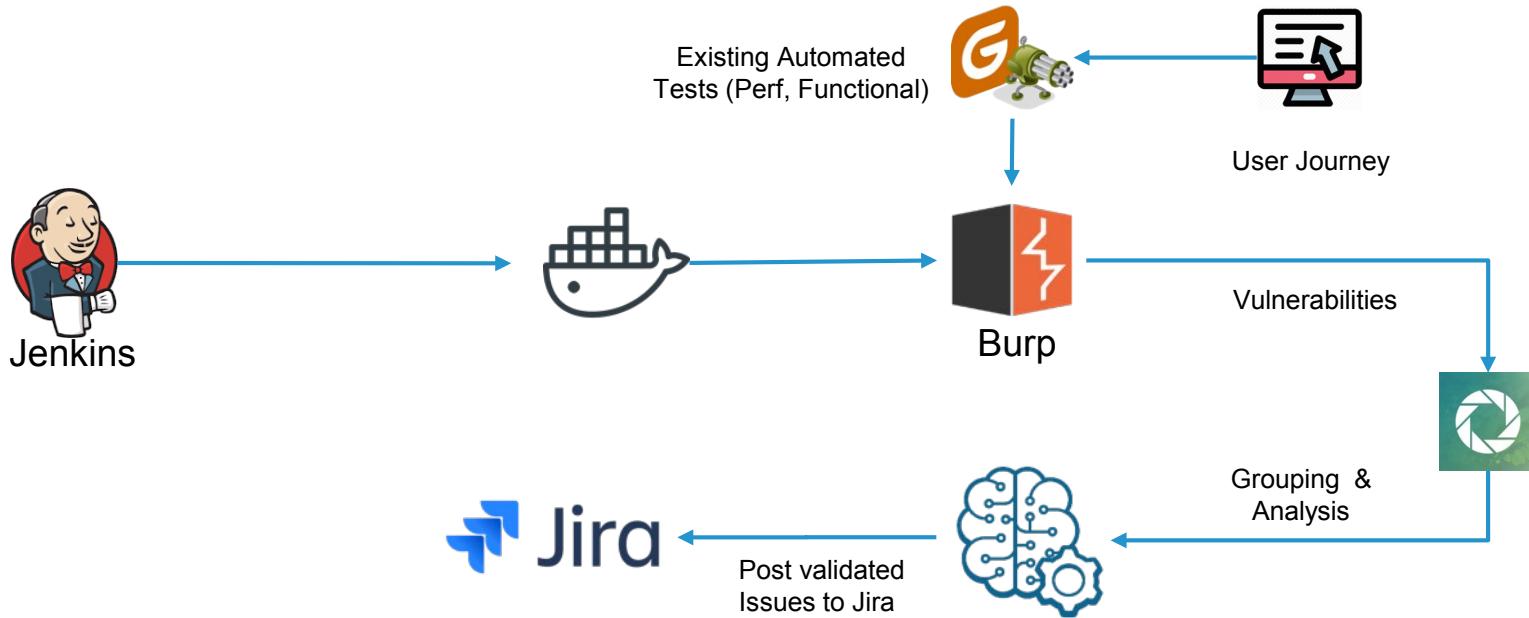
False Positives

Execution Time

Limited Tech Coverage

“Glue code” for Integration

Dynamic Scanning in Pipeline



Lessons Learnt

Existing Test Cases FTW

Focus on your Top 5

Deploy a “Security Test Stack”

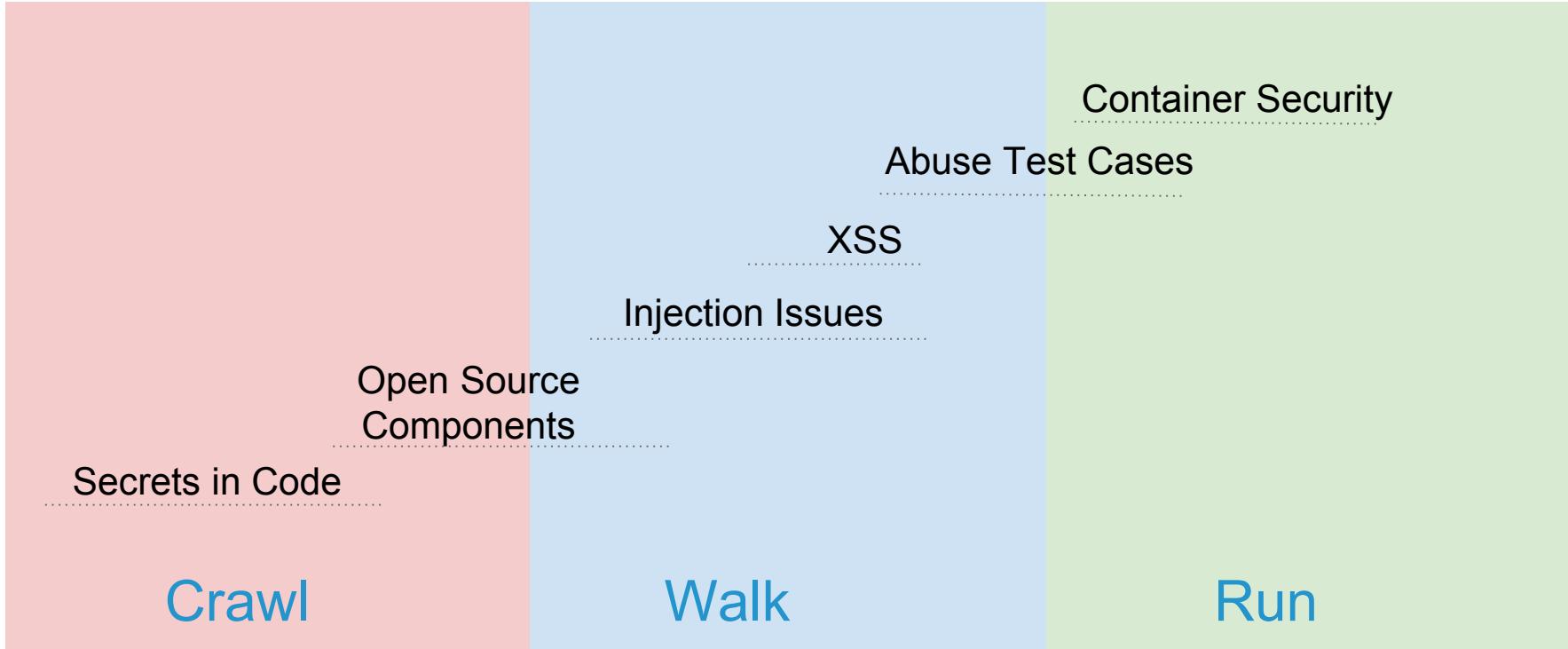
Poor Integration Options

Execution Time

More hand-holding needed

Breaking the build : Myth?

Reporting! Reporting! Reporting!



Case Study : Example 1

T	Summary	Labels	P	Status	Resolution	Created	Reporter	Due
0	Secrets exposed in /tmp/code/app/app.settings.js	GitSecrets Severity_Critical sast	✗	CLOSED	Fixed	09/24/2018	product_security	09/26/2018
0	Secrets exposed in ajaxfunctions.js	GitSecrets sast	✗	CLOSED	Won't Fix	11/06/2018	product_security	11/08/2018
0	Secrets exposed in visible.min.js	GitSecrets sast	✗	CLOSED	Not An Issue	10/10/2018	product_security	10/12/2018
0	Secrets exposed in productsscript.js	GitSecrets sast	✗	CLOSED	Not An Issue	10/10/2018	product_security	10/12/2018
0	Secrets exposed in functions.min.js	GitSecrets sast	✗	CLOSED	Not An Issue	10/10/2018	product_security	10/12/2018
0	Secrets exposed in /app/app.settings.js	GitSecrets sast	✗	RESOLVED	Done	08/17/2018	product_security	08/21/2018
0	Secrets exposed in /app/app.settings.js	GitSecrets sast	✗	RESOLVED	Done	08/17/2018	product_security	08/21/2018
0	Secrets exposed in /app/app.settings.js	GitSecrets sast	✗	RESOLVED	Done	08/20/2018	product_security	08/22/2018
0	Secrets exposed in /app/app.settings.js	GitSecrets sast	✗	RESOLVED	Done	08/20/2018	product_security	08/22/2018
0	Secrets exposed in /app/app.settings.js	GitSecrets sast	✗	RESOLVED	Fixed	08/20/2018	product_security	08/22/2018
0	Secrets exposed in app.settings.js	Braavos GitSecrets sast	✗	CLOSED	Fixed	09/10/2018	product_security	09/12/2018
0	Secrets exposed in prettyify.js	Braavos GitSecrets sast	✗	CLOSED	Fixed	09/14/2018	product_security	09/18/2018

Resolution : Day 1



Security Automation
Secrets exposed in ajaxfunctions.js

Edit Comment Assign More Reopen Issue In Validation

Type: Vulnerability Status: CLOSED (View Workflow)
 Priority: Blocker
 Labels: GitSecrets sast

People
 Assignee: Assign to me
 Reporter: product_securi
 Votes: 0 Vote for this issue
 Watchers: 2 Start watching

Dates

Description
Recommendations:
 Please use secrets management solution for storing sensitive data:
 1. HashiCorp Vault
 2. Parameter Store
 3. KMS
 4. AWS Secrets Manager

Description:
 Sensitive data like passwords, tokens, secrets and private keys should not be committed in repository.
 Keep them secret and not commit in repository.

Secrets exposed in AjaxFunctions.js

Issue Severity: Critical
 Overview: Exposed secrets:
 [line 238] fact*****
 [line 350] fact*****
 References: /AjaxFunctions.js

Secrets exposed in AjaxFunctions.js

Issue Severity: Critical
 Overview: Exposed secrets:
 [line 238] fact*****
 [line 350] fact*****
 References: /AjaxFunctions.js

Implement Cloud Security : Day 1

Github : <https://github.com/dowjones/hammer>

Hammer Case Study

<https://medium.com/dowjones/introducing-dow-jones-hammer-f0121815189a>

Behind the Scenes : (Architecture design)

<https://medium.com/dowjones/behind-the-scenes-of-dow-jones-hammer-38579391f1a0>

Thank you

We're Hiring!

<https://dowjones.jobs>

Jay Kelath

Pranav Patel



<https://github.com/kelath>

<https://github.com/pranav1688>



@kelath



@pranav16

