RSAConference2019
Asia Pacific & Japan
Singapore | 16–18 July | Marina Bay Sands

SESSION ID: HPS-W09

The Future of AppSec is Cloud-Native

Jimmy Mesta

Founder at KSOC
CTO & Trainer at Manicode Security
@jimmesta

Jim Manico

Founder at Manicode Security
@manicode

AppSec and Cloud-Native

A Match Made in the Sky

AppSec and Cloud-Native

A Match Made in the Sky

Whether we like it or not

AppSec and Cloud-Native

A Match Made in the Sky

Whether we like it or not

Seriously…just get used to it

Hello! I'm Jimmy. 

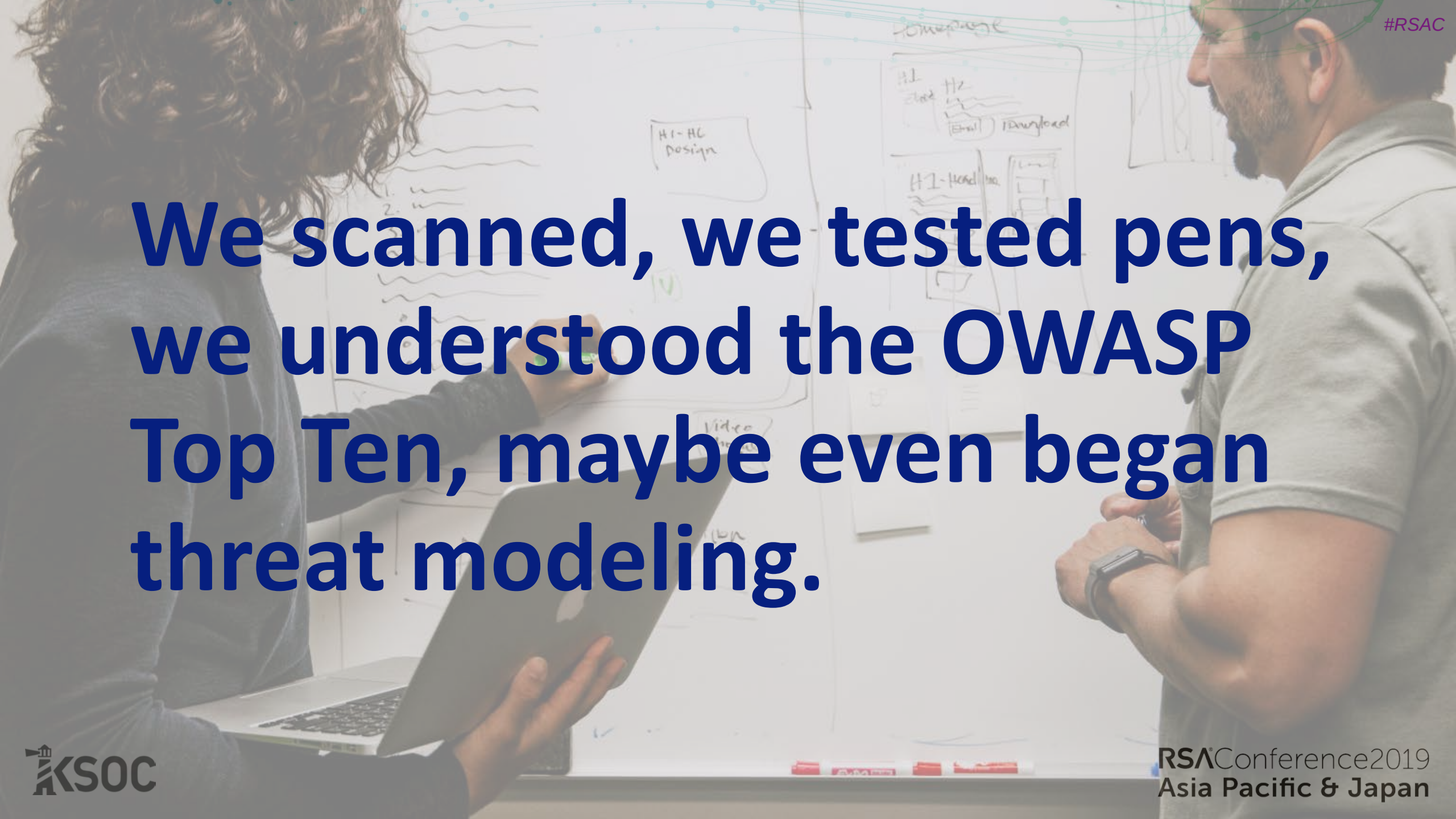
KSOC

MANICODE
SECURE CODING EDUCATION

- AppSec
- DevOps
- Cloud-Native
- CloudDevAppSecOpsNative
- Kubernetes: A Case Study
- Take Home Assignment

AppSec

Traditional AppSec relied on a monolith, guarded by a perimeter of firewalls and blinking boxes.



We scanned, we tested pens, we understood the OWASP Top Ten, maybe even began threat modeling.

We kicked off bug bounties, our pipelines started evolving and we even hired an ”AppSec” engineer.

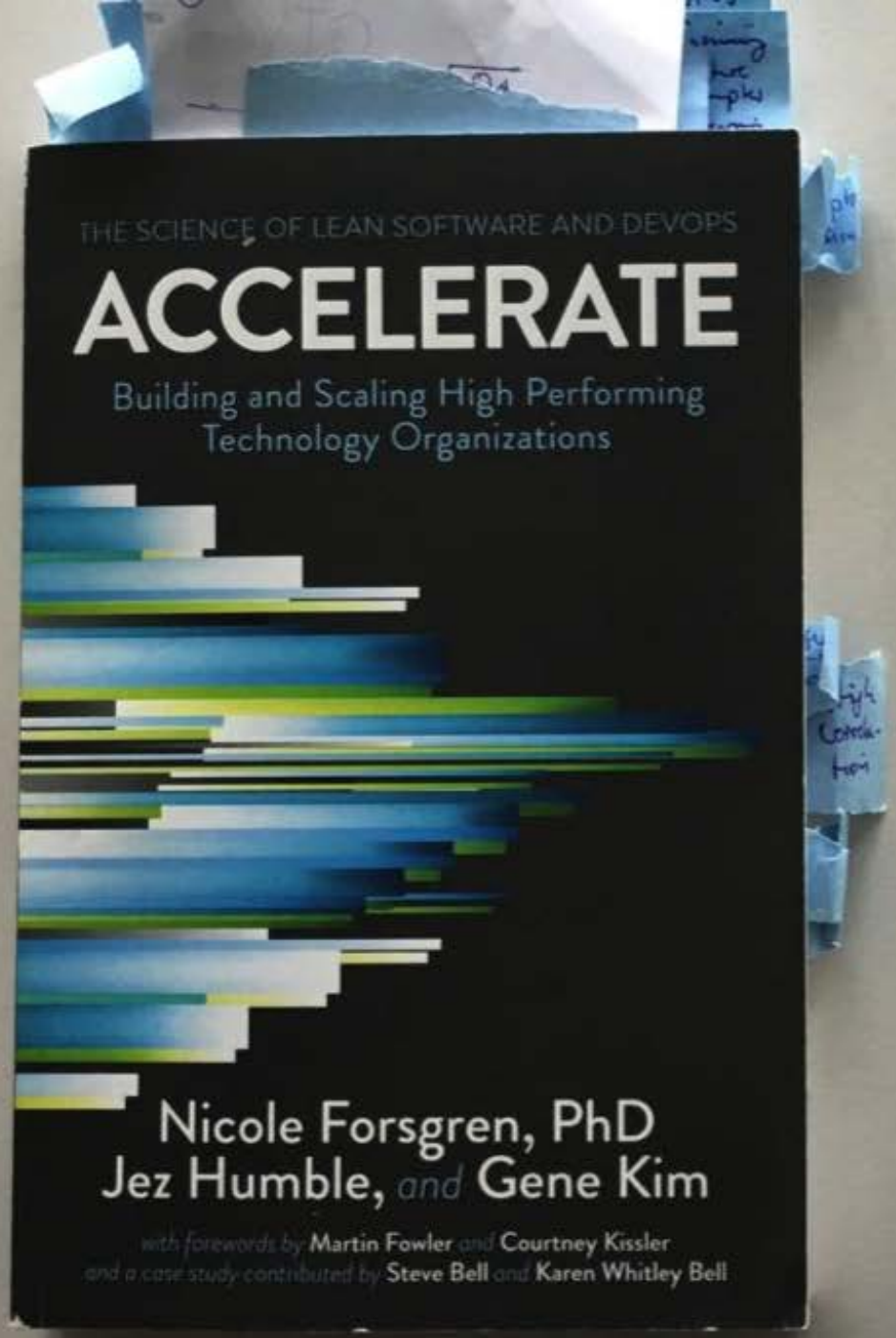
DevSecOps

The DevOps movement was a pivotal time in history for AppSec. We had to make a choice.

Do we embrace the new culture of collaboration and tooling that spans the entire SDLC or stick with what we know?

"If we have data, let's look at data. If all we have are opinions, let's go with mine."

- Jim Barksdale



Accelerate

Building and Scaling High Performing Technology Organizations

Table 2.2 Software. Delivery Performance for 2016

2016	High Performers	Medium Performers	Low Performers
Deployment Frequency	On demand (multiple deploys per day)	Between once per week and once per month	Between once per month and once every six months
Lead Time for Changes	Less than one hour	Between one week and one month	Between one month and six months
MTTR	Less than one hour	Less than one day	Less than one day*
Change Failure Rate	0-15%	31-45%	16-30%

Accelerate: Building and Scaling High Performing Technology Organizations

Table 2.2 Software. Delivery Performance for 2016

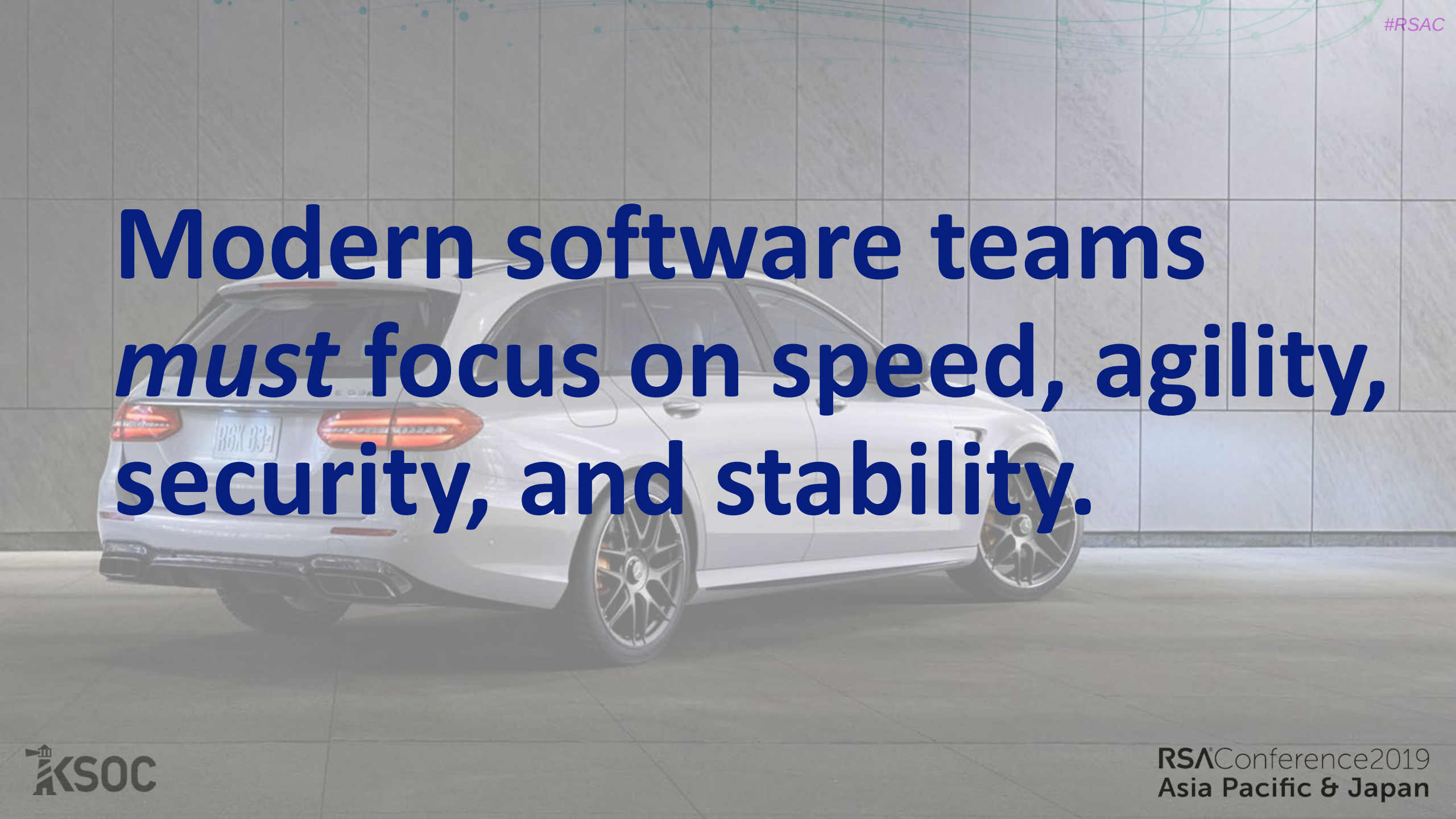
2016	High Performers	Medium Performers	Low Performers
Deployment Frequency	On demand (multiple deploys per day)	Between once per week and once per month	Between once per month and once every six months
Lead Time for Changes	Less than one hour	Between one week and one month	Between one month and six months
MTTR	Less than one hour	Less than one day	Less than one day*
Change Failure Rate	0-15%	31-45%	16-30%

Accelerate: Building and Scaling High Performing Technology Organizations

DevSecOps is the process of incorporating and enforcing meaningful security controls without slowing down deployment velocity.

DevSecOps means that we are all responsible for security.

Cloud-Native



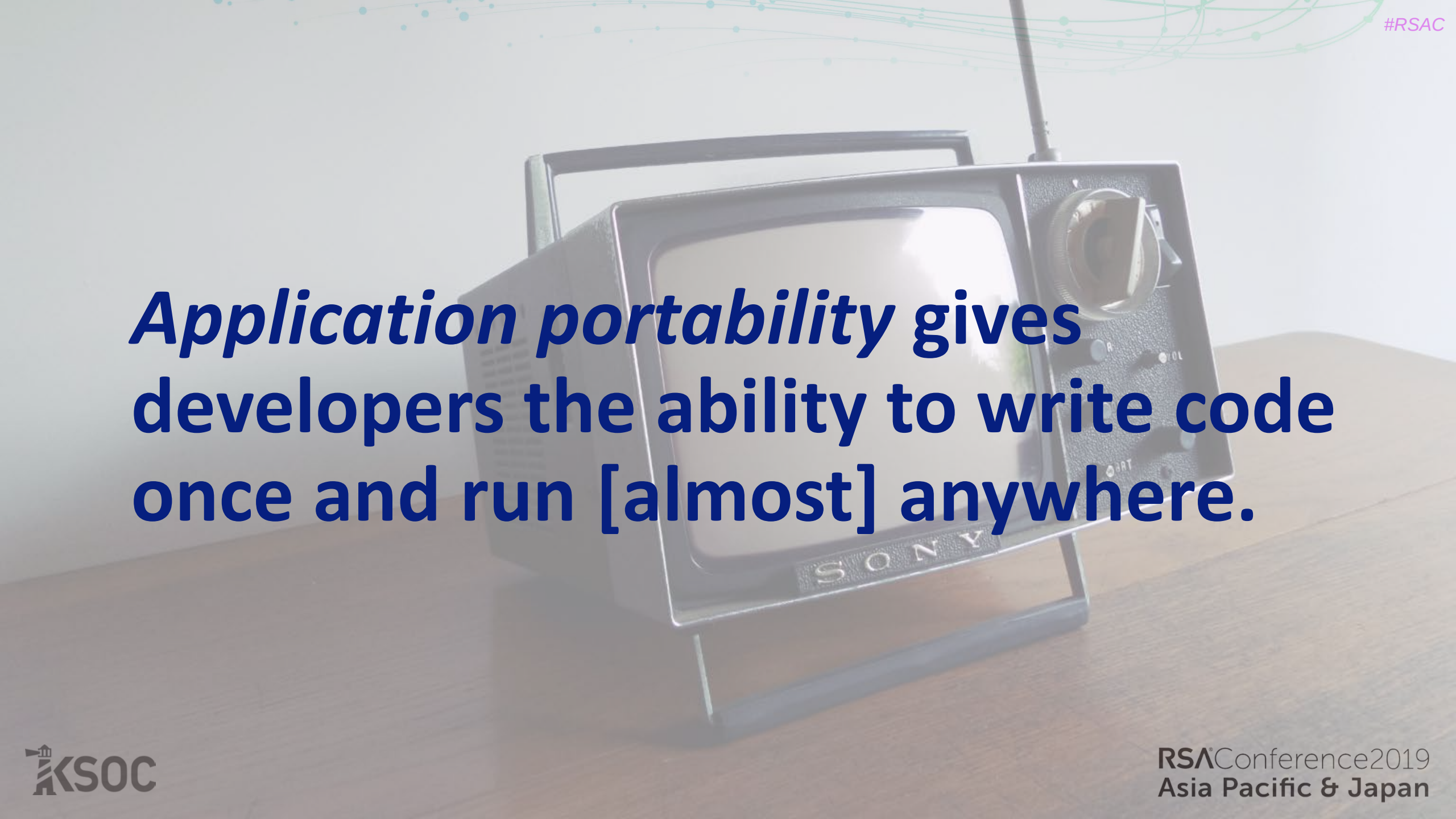
Modern software teams *must* focus on speed, agility, security, and stability.

Cloud-Native is a set of design patterns, not just a collection of tools.

Cloud-Native patterns enable organizations to deliver software at a rapid velocity with more confidence.


- Observability
- Application Portability
- Loosely Coupled Services
- Policy Driven Infrastructure
- Automated Pipelines

Observability **encompasses the log aggregation, monitoring, metric collection, to gain insight into a given system.**



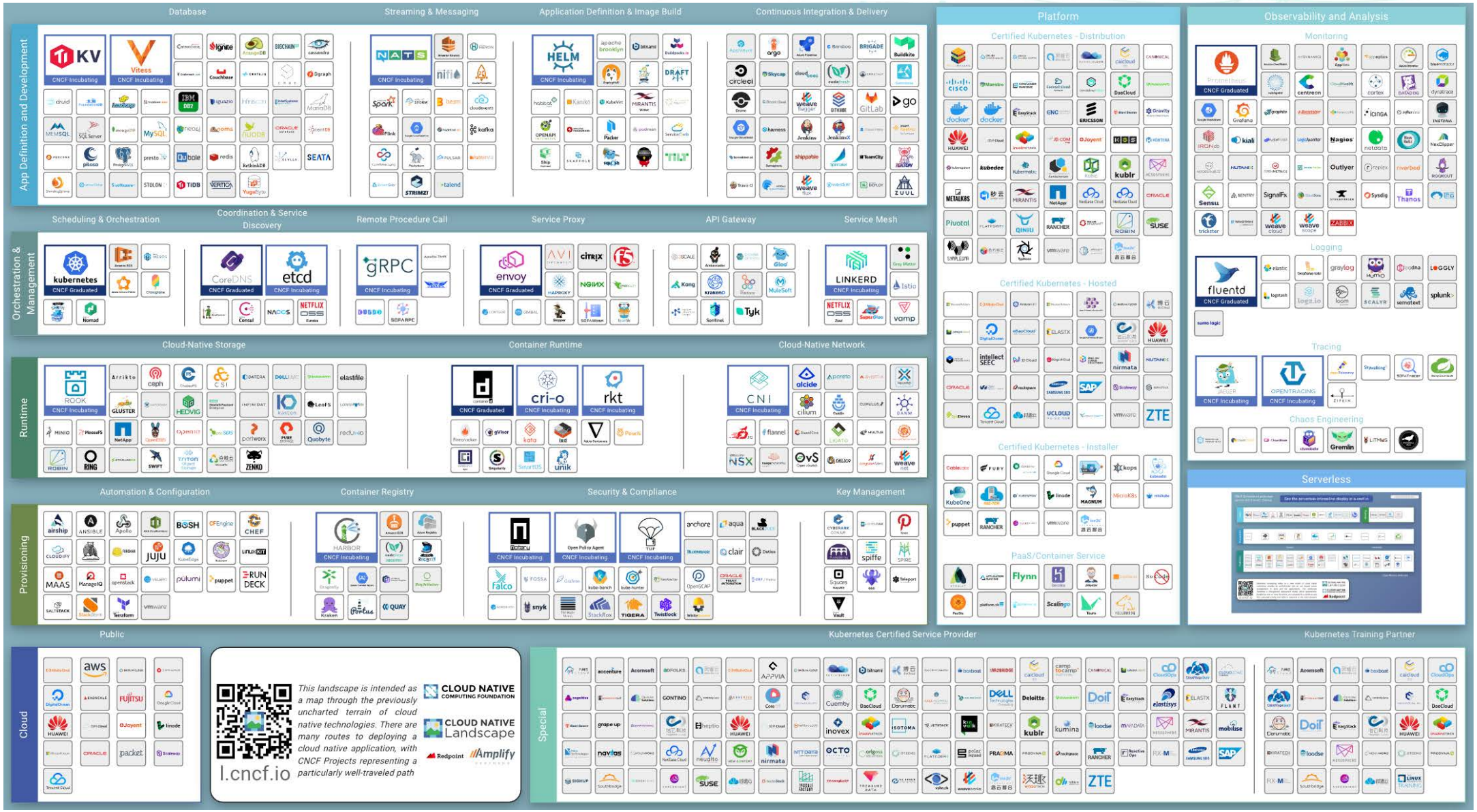
Application portability **gives developers the ability to write code once and run [almost] anywhere.**

Loosely coupled services **are often API-driven and allow applications to run, scale, and interact as an independent entity.**

Policy driven infrastructure **gives operators the ability to define the desired state of applications and underlying infrastructure using definition files.**

Automated pipelines **remove manual steps involved in shipping code from commit to production.**

The combination of cloud native tools and patterns help organizations achieve product goals.



https://landscape.cncf.io/

The Future of AppSec is Cloud-Native

The Future of AppSec is CloudDevAppSecOpsNative

Our illusion of control is gone.

Public APIs, multi-cloud environments, third-party components, containers, and cloud-native tools changed the game.

André Baptista (0xacb)

1036	-	5.58	91st	21.18	93rd
Reputation	Rank	Signal	Percentile	Impact	Percentile

393 #341876 **SSRF in Exchange leads to ROOT access in all instances**

Share:

State	Resolved (Closed)	Severity	Medium (6.9)
Disclosed	May 23, 2018 2:09pm -0700	Participants	
Reported To	Shopify	Visibility	Disclosed (Full)
Asset	https://exchangemarketplace.com/ (Domain)		
Weakness	Server-Side Request Forgery (SSRF)		
Bounty	$25,000		

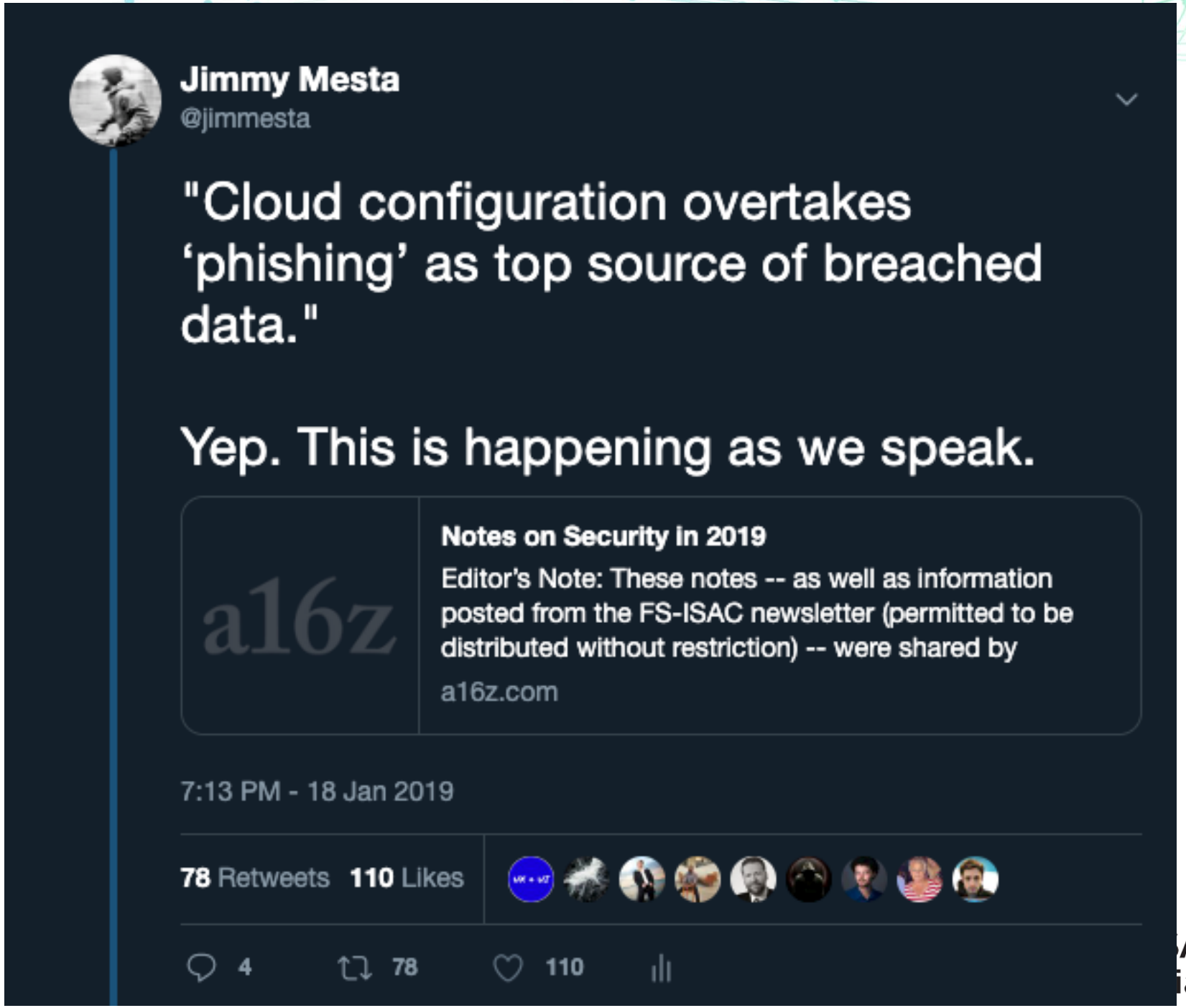
https://hackerone.com/reports/341876

”Through 2020, 80% of cloud breaches will be due to customer misconfiguration, mismanaged credentials or insider theft, not cloud provider vulnerabilities.”

Gartner

”A historic 424% jump in breaches related to misconfigured cloud infrastructure, largely due to human error.”

2017 IBM X-Force Report


Jimmy Mesta
@jimmesta
"Cloud configuration overtakes ‘phishing’ as top source of breached data."
Yep. This is happening as we speak.
a16z
Notes on Security in 2019
Editor's Note: These notes -- as well as information posted from the FS-ISAC newsletter (permitted to be distributed without restriction) -- were shared by
a16z.com
7:13 PM - 18 Jan 2019
78 Retweets 110 Likes
4
78
110

CVE-2019-9901 - Istio/Envoy Path traversal

TLDR; I found a path traversal bug in Istio's authorization policy enforcement.

https://github.com/eoftedal

Envoy Authorization Policy

```
rules:
- services: ["backend.fishy.svc.cluster.local"]
  methods: ["GET"]
  paths: ["/public/*"]
```

Path Traversal Payload

```
curl -vvvv --path-as-is  "http://backend.fishy.svc.cluster.local:8081/public/../secret/"
```

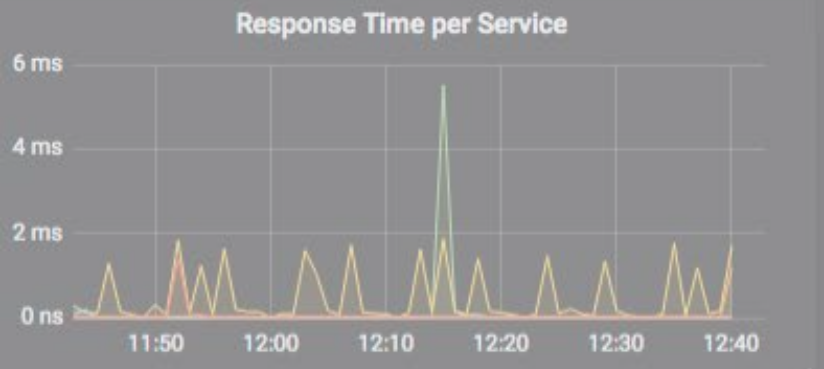
Contractor's AWS S3 server leaks data from Fortune 100 companies: Ford, Netflix, TD Bank

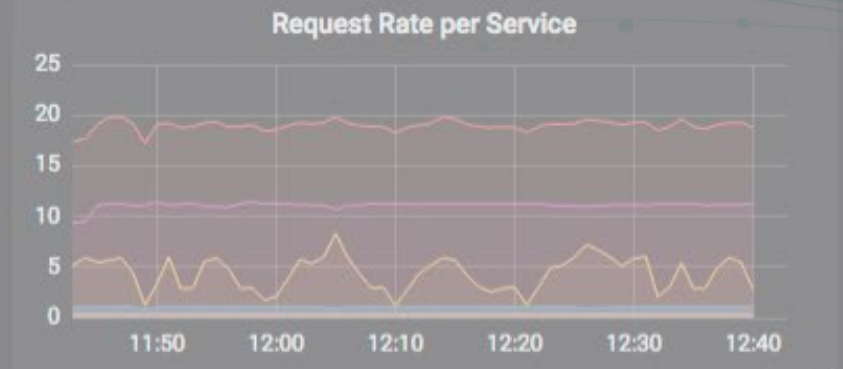
Exposed data includes passwords and private keys for production systems, employee details, sales information.

https://www.zdnet.com/article/contractors-aws-s3-server-leaks-data-from-fortune-100-companies-ford-netflix-td-bank/

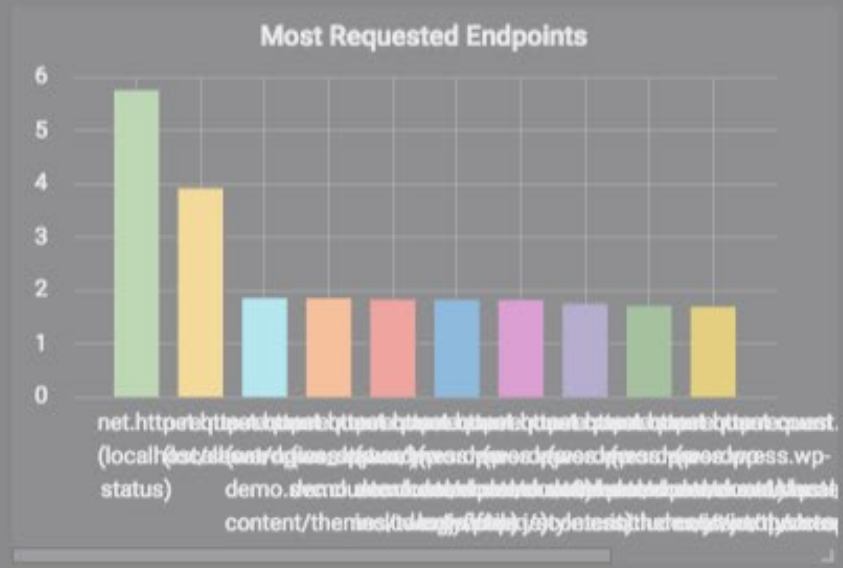
If blindly adopted, Cloud-Native patterns and tools can introduce a slew of new vulnerabilities into your infrastructure.

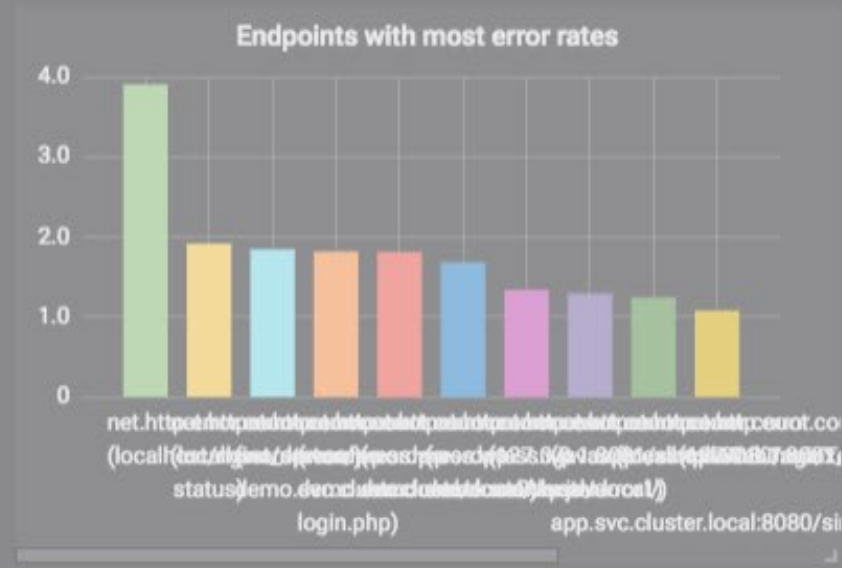
But, when properly prepared, Cloud-Native patterns can help AppSec move faster, and more safely than ever before.





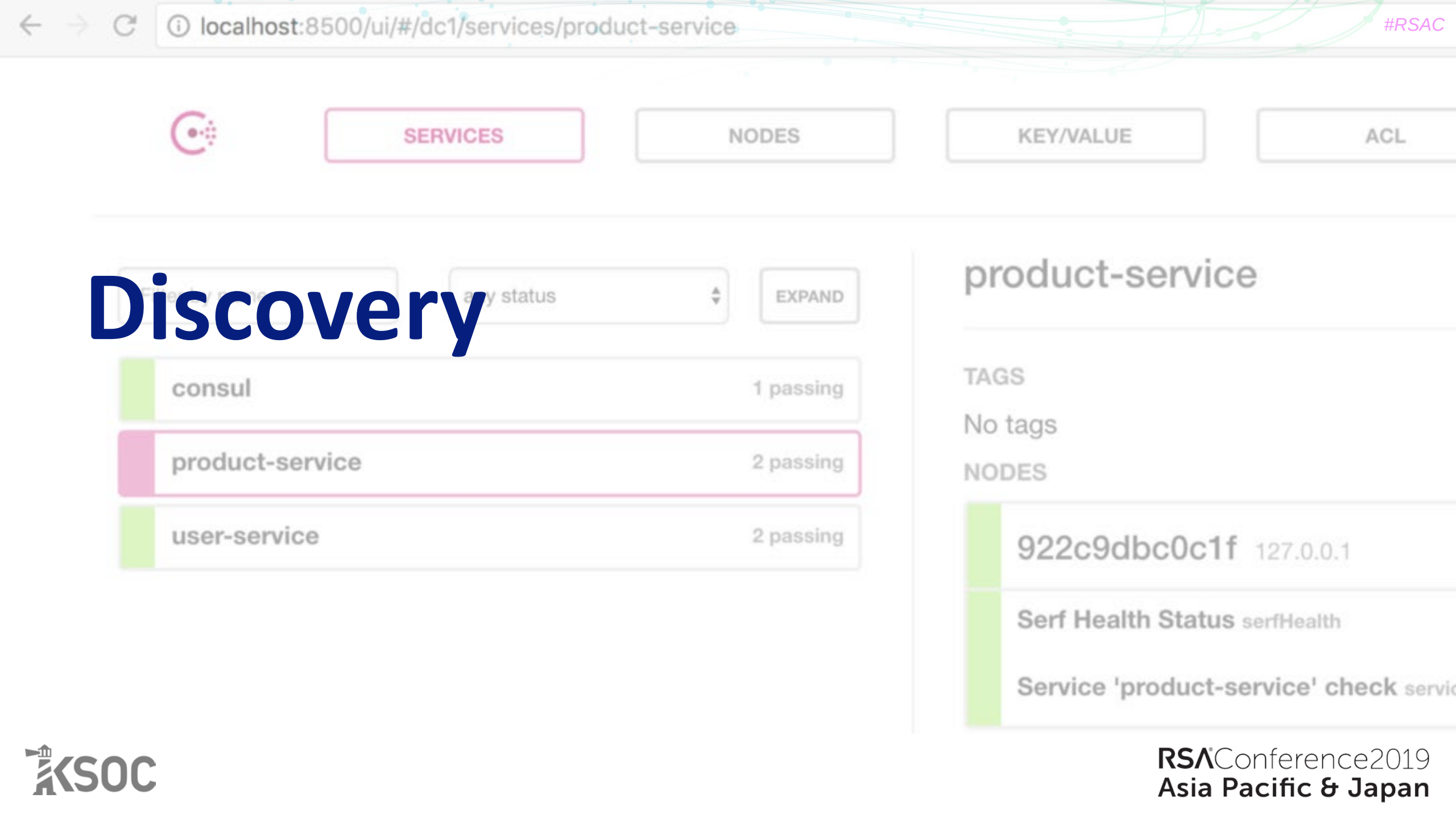
Visibility





My Services

kubernetes.service.name	net.http.request.time	net.http.request.count	net.http.error.count	net.http.statusCode
javaapp	177.43 K	18.99	10.14	200, 500
n/a	170.81 K	11.13	4.71	200
kubernetes-dashboard	21.17 K	4.33	4.33	200
kube-dns	15.29 K	0.99	0	200


localhost:8500/ui/#/dc1/services/product-service
SERVICES
NODES
KEY/VALUE
ACL
EXPAND
consul
1 passing
product-service
2 passing
user-service
2 passing
product-service
TAGS
No tags
NODES
922c9dbc0c1f
127.0.0.1
Serf Health Status
serfHealth
Service 'product-service' check

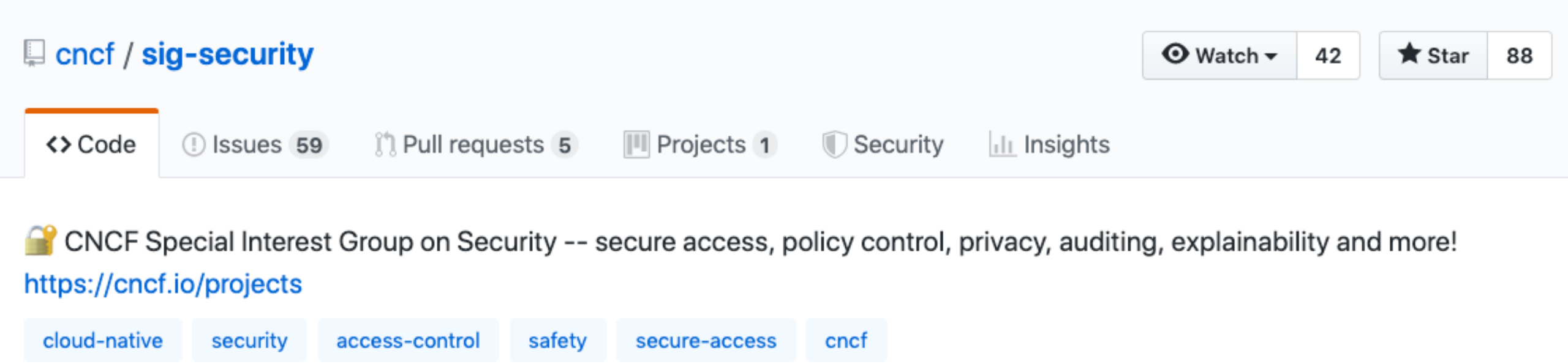
Discovery

AuthN / AuthZ

”The SPIFFE Runtime Environment forms a powerful solution for connecting, authenticating, and securing workloads in distributed environments.”

https://www.cncf.io/blog/2018/03/29/cncf-to-host-the-spiffe-project/

Community



https://github.com/cncf/sig-security

Automation and Recovery

Whitepapers

AWS Best Practices for DDoS Resiliency (June 2018)

AWS Cloud Adoption Framework: Security Perspective (June 2016)

AWS Security Best Practices (August 2016)

AWS Security Checklist

AWS Well-Architected Framework: Security Pillar (July 2018)

Introduction to AWS Security (July 2015)

Introduction to AWS Security Processes (June 2016)

Overview of AWS Security - Analytics, Mobile and Application Services (June 2016)

Overview of AWS Security - Application Services (June 2016)

Overview of AWS Security - Compute Services (June 2016)

Overview of AWS Security - Database Services (June 2016)

Overview of AWS Security - Network Services (August 2016)

Overview of AWS Security - Storage Services (June 2016)

Secure Content Delivery with CloudFront (November 2016)

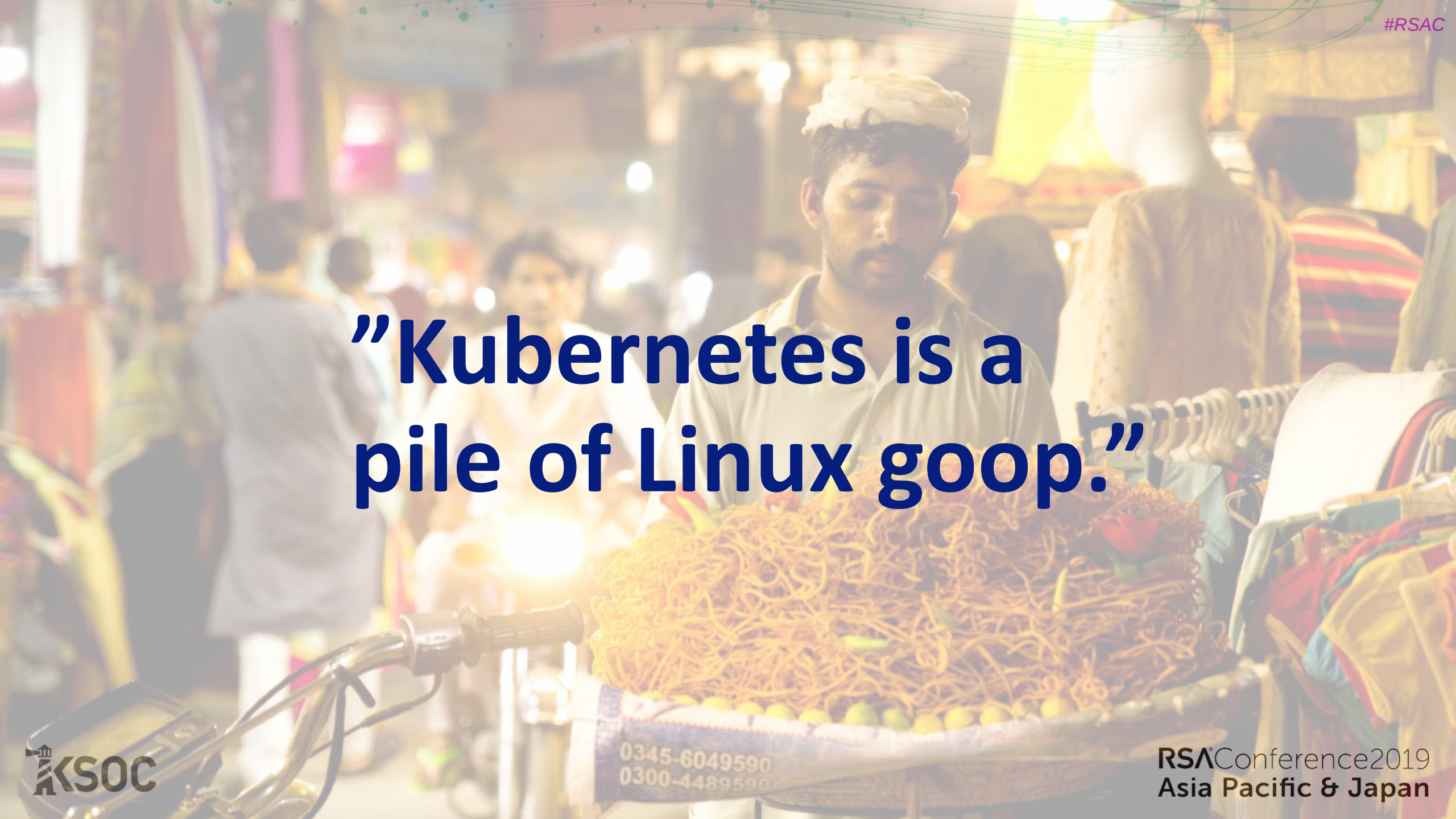
Securing Data at Rest with Encryption (November 2014)

Platform Security

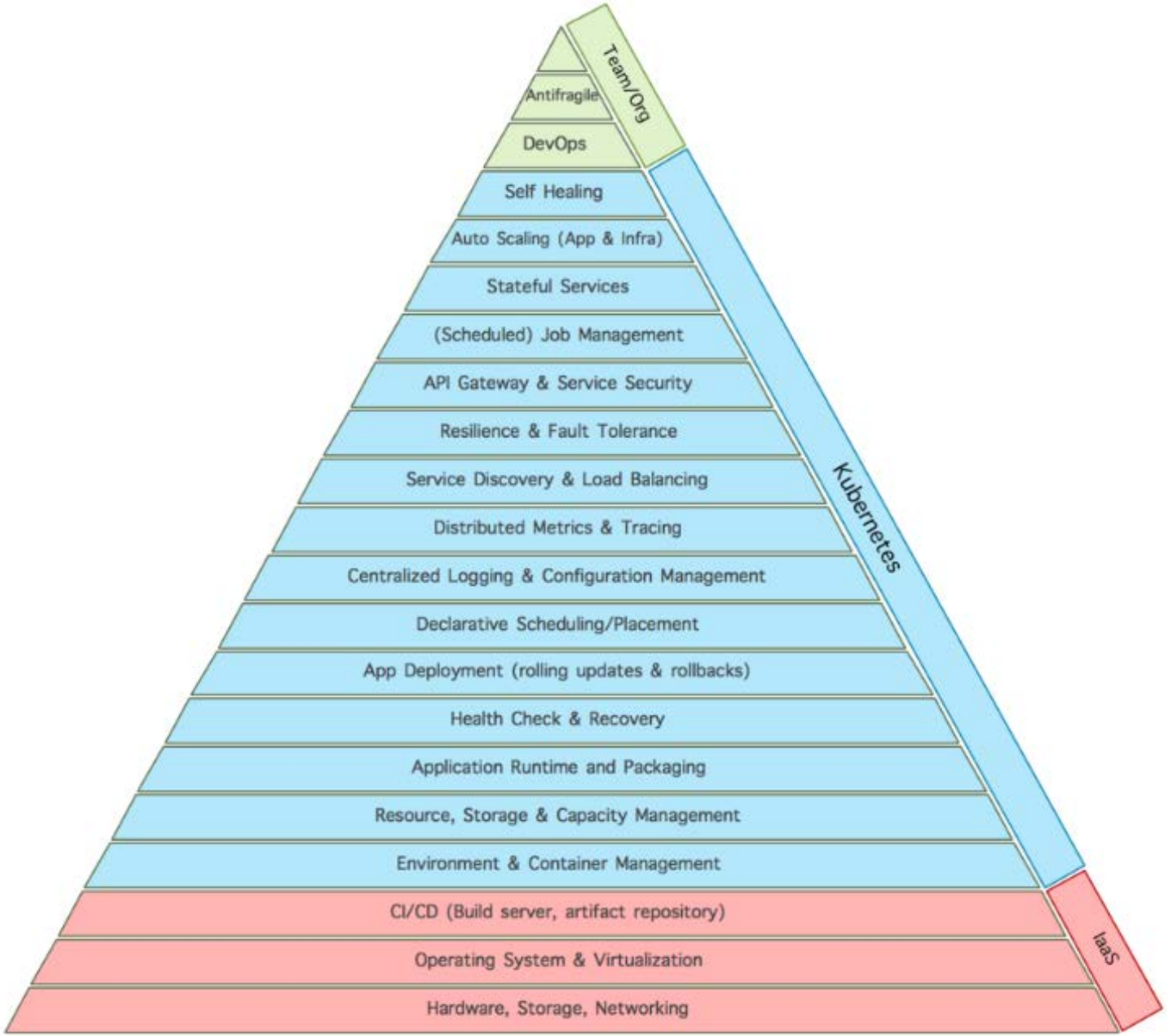

Chenxi Wang
@chenxiwang
Follow
I am seeing a quiet revolt against many security tools, b/c they do not play nicely with modern dev practices.
1:51 PM - 6 Jun 2019
117 Retweets 309 Likes
19
117
309

Kubernetes: A Case Study

Kubernetes is an open-source platform built to automate deployment, scaling, and orchestration of containers.



”Kubernetes is a pile of Linux goop.”



https://thenewstack.io/introducing-microservices-hierarchy-needs/

I thought Kubernetes was secure by default?

Name

kubernetes

Search

Config and storage > Secrets > aws-s3-credentials

Namespace

default

Overview

Workloads

- Daemon Sets
- Deployments
- Jobs
- Pods
- Replica Sets
- Replication Controllers
- Stateful Sets

Discovery and Load Balancing

- Ingresses

Details

Name: aws-s3-credentials

Namespace: default

Creation time: 2017-10-12T22:29

Type: Opaque

Data

aws-s3-access-key-id: [redacted]

aws-s3-secret-access-key: [redacted]

3 - Using Kubelet to execute arbitrary commands

It's possible to list all pods {F289460}:

```
$ kubectl --client-certificate client.crt --client-key client.pem --certificate-authority ca.crt --server

NAMESPACE                                   NAME
[redacted]                                  [redacted]                                  1/1
```

And create new pods as well:

```
$ kubectl --client-certificate client.crt --client-key client.pem --certificate-authority ca.crt --server

pod "shell-demo" created
$ kubectl --client-certificate client.crt --client-key client.pem --certificate-authority ca.crt --server

pod "shell-demo" deleted
```

How A Cryptocurrency Miner Made Its Way onto Our Internal Kubernetes Clusters

Brian Choy in JW Player Engineering Follow

Mar 19 · 10 min read

https://medium.com/jw-player-engineering/how-a-cryptocurrency-miner-made-its-way-onto-our-internal-kubernetes-clusters-9b09c4704205

Kubernetes » Kubernetes : Vulnerability Statistics

Vulnerabilities (13) CVSS Scores Report Browse all versions Possible matches for this product Related Metasploit Modules

Related OVAL Definitions : Vulnerabilities (0) Patches (0) Inventory Definitions (0) Compliance Definitions (0)

Vulnerability Feeds & Widgets

Vulnerability Trends Over Time

Year	# of Vulnerabilities	DoS	Code Execution	Overflow	Memory Corruption	Sql Injection	XSS	Directory Traversal	Http Response Splitting	Bypass something	Gain Information	Gain Privileges	CSRF	File Inclusion	# of exploits
2016	3										1	1			
2017	2										1				
2018	3									1					
2019	5	1													
Total	13	1								1	2	1			
% Of All		7.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	7.7	15.4	7.7	0.0	0.0	

Kubernetes Engine > Documentation

Security bulletins

Contents

May 31, 2019

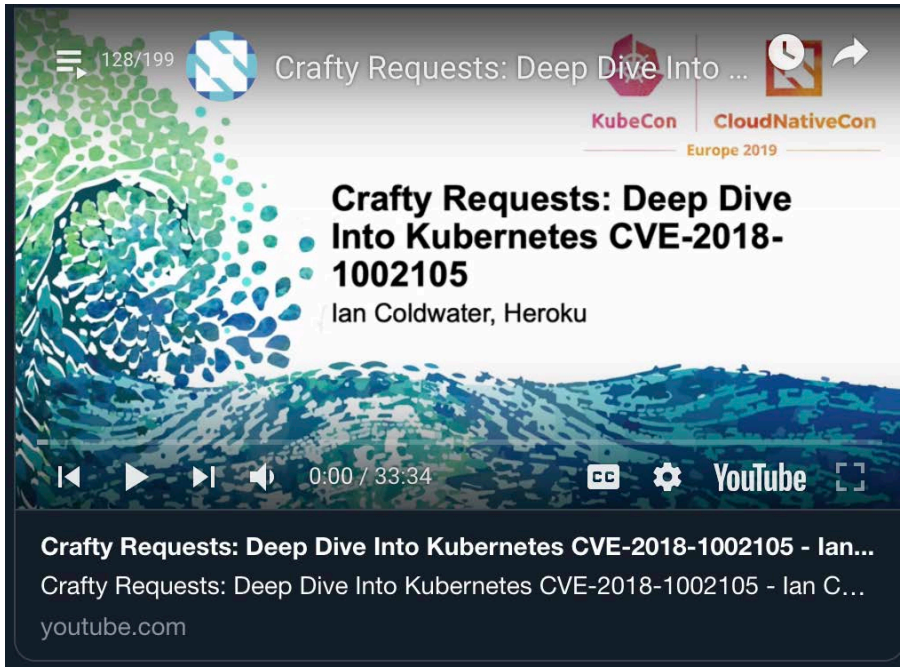
May 14, 2019

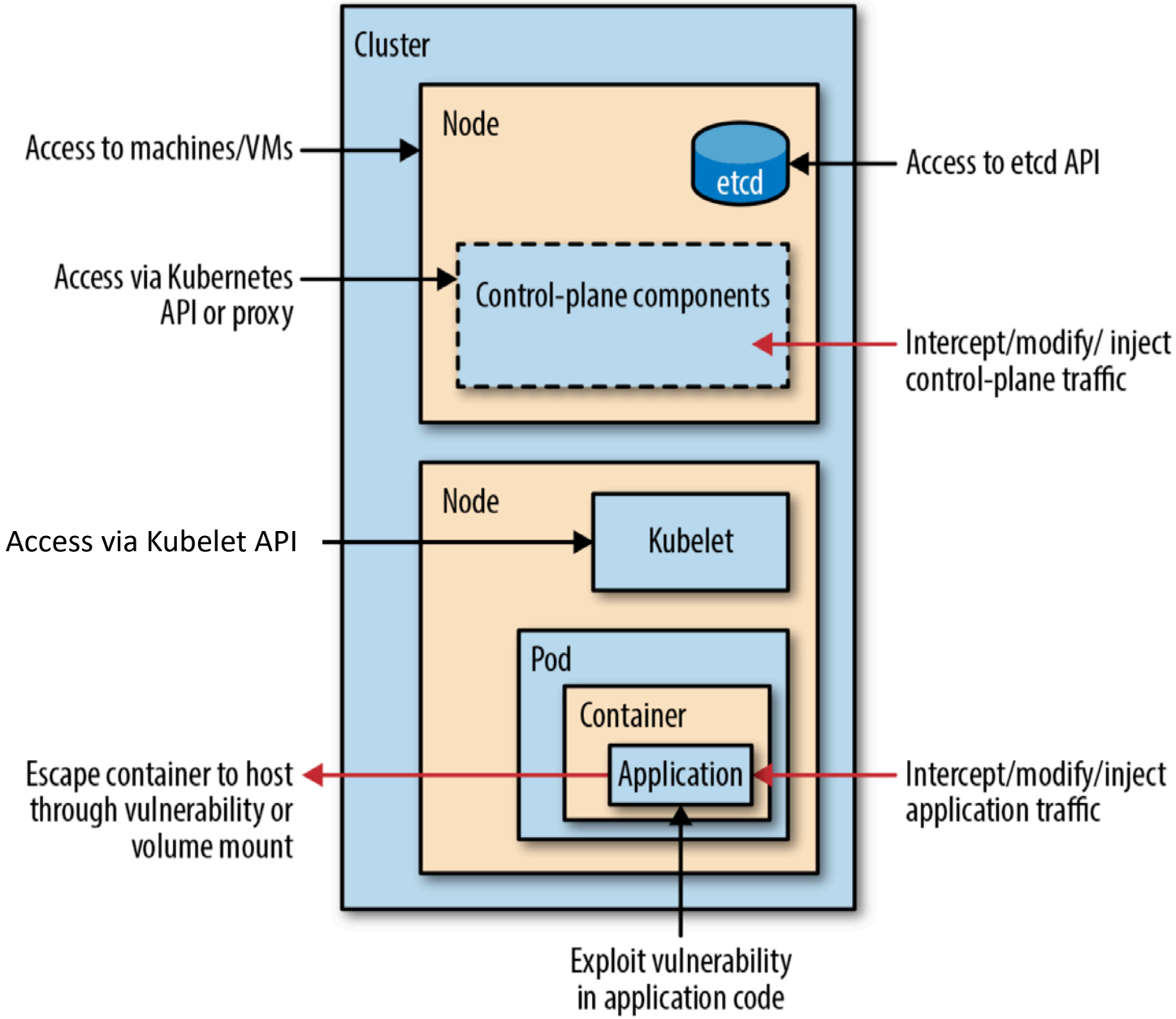
April 5, 2019

March 1, 2019

...

All security bulletins for Google Kubernetes Engine are described in this topic.





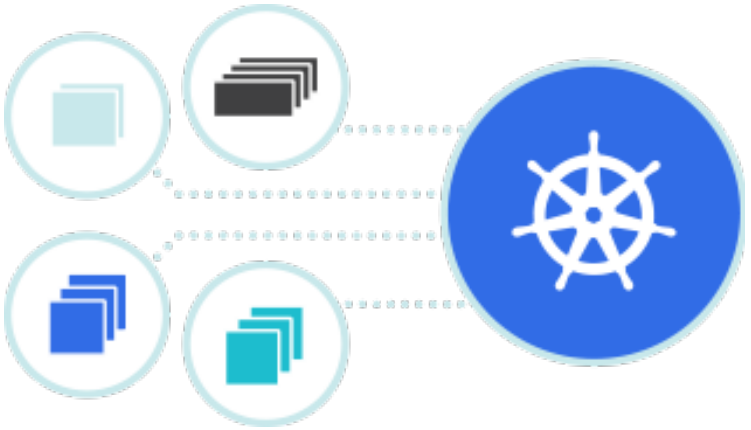
Source: Kubernetes Security - Operating Kubernetes Clusters and Applications Safely

RBAC
Container and Pod Permissions
Pod Security Policies
Dynamic Admission Control
Sandboxing
Node Protection

Role-Based Access Control (RBAC) is how we regulate access to Kubernetes resources.

Users

you@email.com
Service account



API Resources

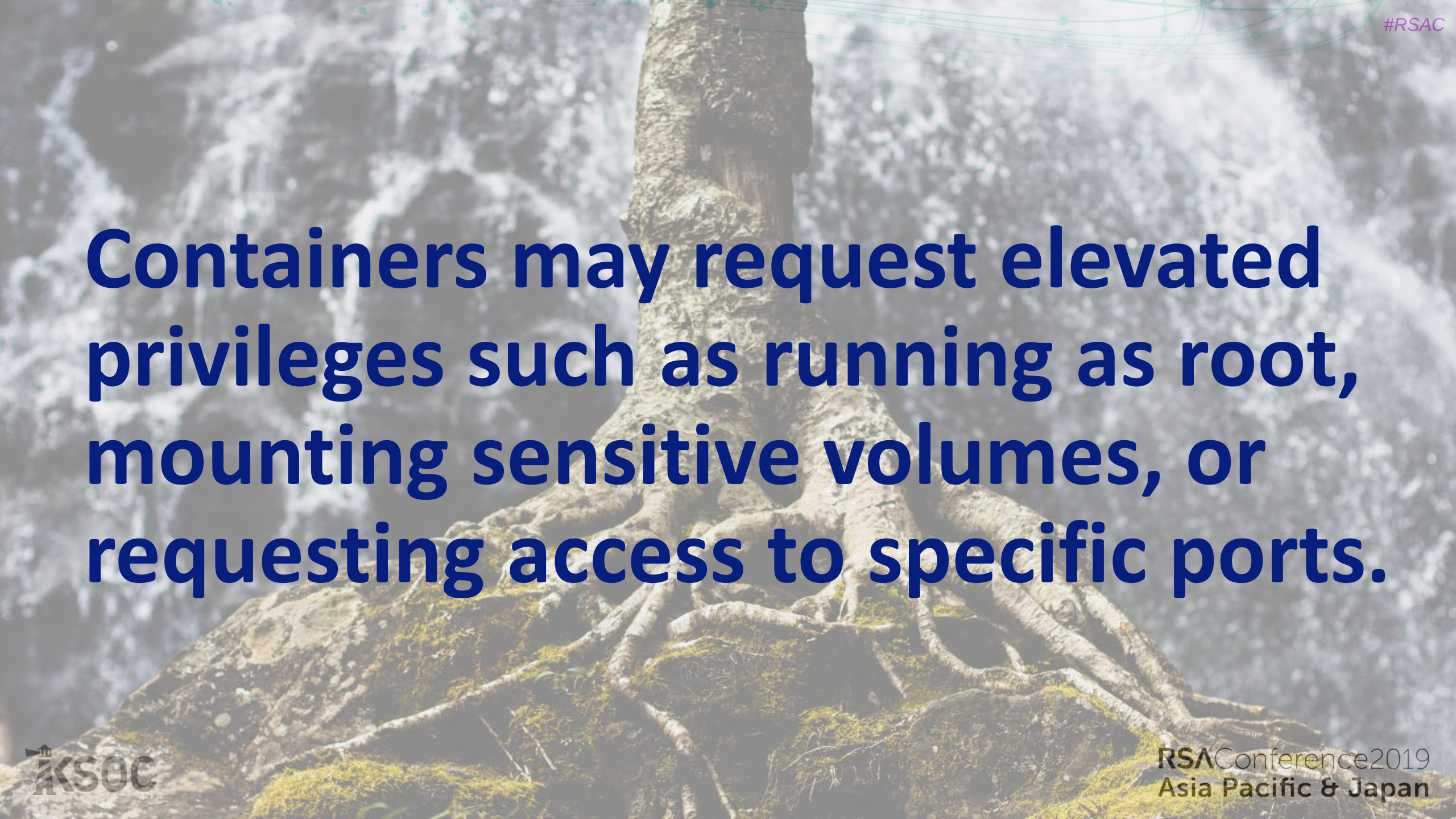
Namespaces
Pod
Service
Secrets
…

Operations

Get
List
Delete
Patch

```
kind: Role
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  namespace: development
  name: pod-reader
rules:
- apiGroups: [""]
  resources: [”pods"]
  verbs: ["get", "list"]
```

role.yaml



Containers may request elevated privileges such as running as root, mounting sensitive volumes, or requesting access to specific ports.

Pod specifications may declare to access devices on the host using `privileged` mode.

Pod Security Policies give administrators the ability to validate requests to the cluster based on security requirements.

```
apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
  name: my-psp
spec:
  privileged: false
  seLinux:
    rule: RunAsAny
  supplementalGroups:
    rule: RunAsAny
  runAsUser:
    rule: MustRunAsNonRoot
  volumes:
    - 'configMap'
    - 'emptyDir’
    - 'secret’
    - 'persistentVolumeClaim'
```

psp.yaml

Dynamic Admission Control allows teams to build custom security checks by intercepting requests to the Kubernetes API server prior to scheduling the object.

kelseyhightower / **denyenv-validating-admission-webhook**

Watch 1 | Star 90 | Fork 14

Code | Issues 0 | Pull requests 3 | Projects 0 | Wiki | Security | Insights

An Kubernetes validating admission webhook that rejects pods that use environment variables.

kubernetes serverless faas gcp-cloud-functions

https://github.com/kelseyhightower/denyenv-validating-admission-webhook

Tools such as gVisor and Kata Containers can help further isolate and sandbox containers that are running untrusted workloads inside of Kubernetes.

Remember, Kubernetes is just running servers under the hood. Our regular old OS hardening and network protections apply.

Kubernetes *can* be secure, but it is far from default.

In Summary


The future of AppSec is a deeper understanding and goes far beyond the source code.

The challenge going forward will be balancing security with deployment efficiency.

A scanner or firewall won't save you.

Take Home Assignment

Make friends

Beware of blind spots

Embrace a beginners mindset mindset

Adapt and evolve

Practice

https://github.com/ksoclabs/kube-goat

https://github.com/RhinoSecurityLabs/cloudgoat

https://www.owasp.org/index.php/OWASP_Serverless_Goat

@jimmesta
jimmy[at]ksoc.com