



Kubernetes und Container – Aber Sicher!

Container / K8s Security

Andreas Falk



Vorstellung

Andreas Falk
Novatec Consulting



andreas.falk@novatec-gmbh.de / [@andifalk](https://www.instagram.com/andifalk)

<https://www.novatec-gmbh.de/beratung/agile-security>

Security Training for Developers

by Jim Manico

You want to learn about security from one of the world's most famous application security experts? This training is your chance and extremely rare in Europe!

Always be one step ahead of the hackers!



from March 4 to March 5, 2020

Novatec Consulting GmbH

Dieselstraße 18/1, 70771 Leinfelden-Echterdingen

<https://www.novatec-gmbh.de/schulung/application-security-training-for-developers-by-jim-manico>

Agenda

1. What can go wrong
2. Application Security
3. Container Security
4. Kubernetes Security
5. Kubernetes Secrets

What can go wrong?

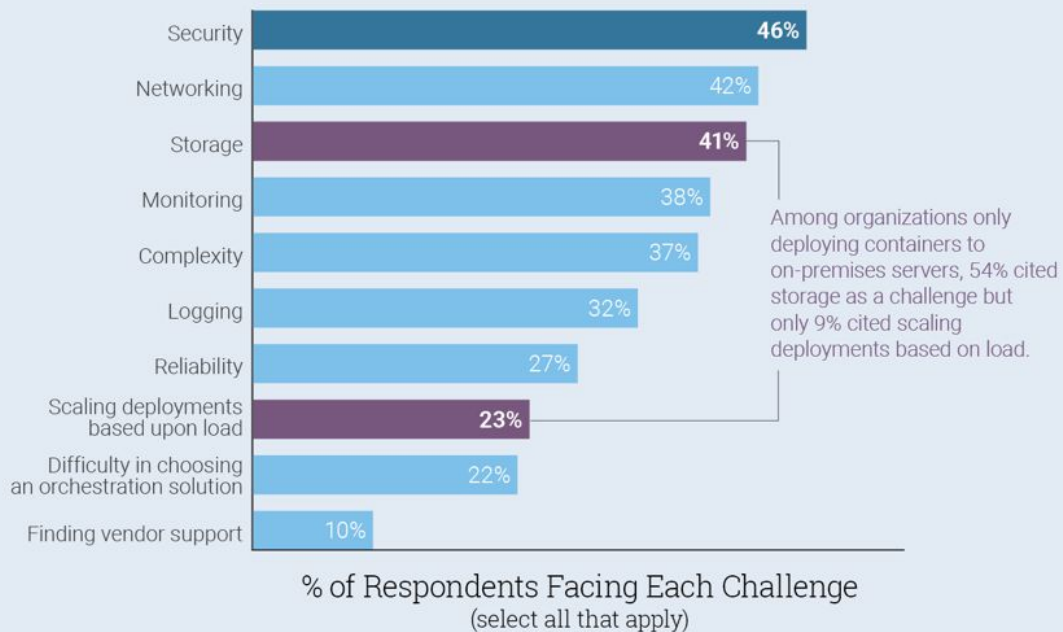
Introduction



Top Challenges in Kubernetes

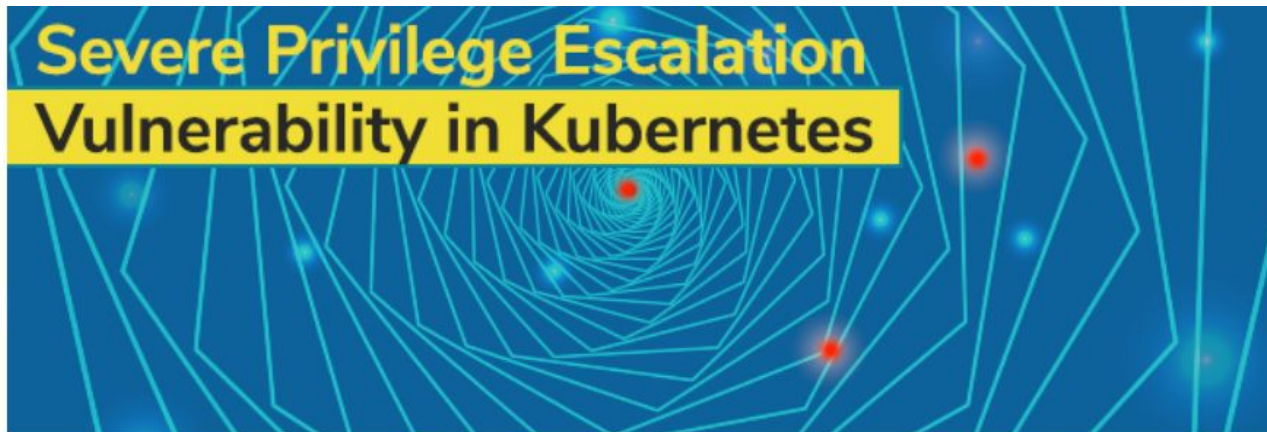
Source: <https://thenewstack.io>

Security is Top Challenge for Kubernetes Users



Severe Vulnerability in Kubernetes

Source: <https://blog.aquasec.com>



Ariel Shuper • December 06, 2018

Severe Privilege Escalation Vulnerability in Kubernetes (CVE-2018-1002105)

Earlier this week, a [severe vulnerability in Kubernetes](#) (CVE-2018-1002105) was disclosed that allows an unauthenticated user to perform privilege escalation and gain full admin privileges on a cluster. The CVE was given the high severity score of 9.8 (out of 10) and it affects all Kubernetes versions from 1.0 onwards, but fixes are available for recent versions.

Crypto Mining Via K8s Dashboard

Source: <https://blog.heptio.com>

On Securing the Kubernetes Dashboard




Joe Beda [Follow](#)

Feb 28, 2018 · 13 min read



Recently Tesla (the car company) was alerted, by security firm RedLock, that their Kubernetes infrastructure was compromised. The attackers were using Tesla's infrastructure resources to mine cryptocurrency. This type of attack has been called "cryptojacking".






The vector of attack in this case was a Kubernetes Dashboard that was exposed to the general internet with no authentication and elevated privileges. Not only this, but core AWS API keys and secrets were visible. How do you prevent this from happening to you?

Open ETCD Ports in Kubernetes (1)

 **SHODAN**

etcd port:"2379"


  Explore Downloads Reports

 Exploits  Maps  Share Search  Download Results  Create Report

TOTAL RESULTS

2,450

TOP COUNTRIES



China	1,116
United States	541
Germany	138
France	117
Singapore	70

TOP ORGANIZATIONS


Hangzhou Alibaba Advertisin...	417
Amazon.com	273
Tencent cloud computing	172
China Unicom Beijing	111
Hetzner Online GmbH	54

New Service: Keep track of what you have connected to the Internet. Check

47.52.241.38

Alibaba

Added on 2019-07-02 11:19:29 GMT

 Hong Kong

cloud

etcd

Name: etcd-hk

Version: 3.2.6

Uptime: 47h12m20.876361718s

Peers: http://10.70.10.205:2380

34.77.57.47

47.57.77.34.bc.googleusercontent.com

Halliburton Company

Added on 2019-07-02 11:05:41 GMT

 United States

etcd

Name: m3db_local

Version: 3.2.10

Uptime: 118h39m34.598205154s


Peers: http://0.0.0.0:2380

13.229.135.103

ec2-13-229-135-103.ap-southeast-1.compute.amazonaws.com

Amazon Data Services Singapore

Added on 2019-07-02 11:07:34 GMT

 Singapore, Singapore

etcd

Name: node1

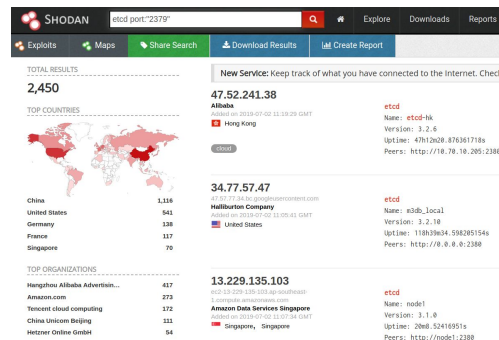
Version: 3.1.0

Uptime: 20m8.52416951s

Peers: http://node1:2380

<https://shodan.io>

Open ETCD Ports in Kubernetes (2)



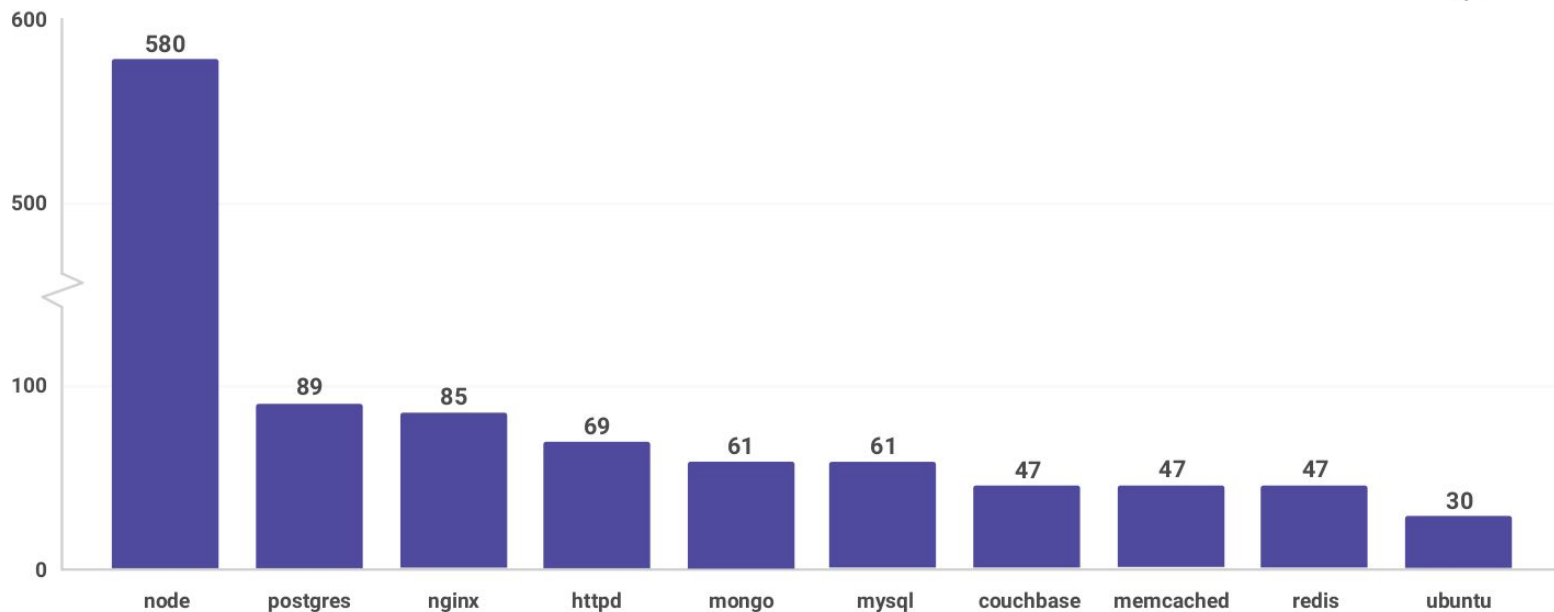
```
$ etcdctl --endpoints=http://xx.xx.xx.xx:2379  
cluster-health
```

```
member b97ee4034db41d17 is healthy: got healthy  
result  
from http://xx.xx.xx.xx:2379  
cluster is healthy
```

Vulnerable Docker Images

Source: The state of open source security report (snyk.io)

Number of OS vulnerabilities by docker image



All is Root



CZnative @ home

@pczarkowski

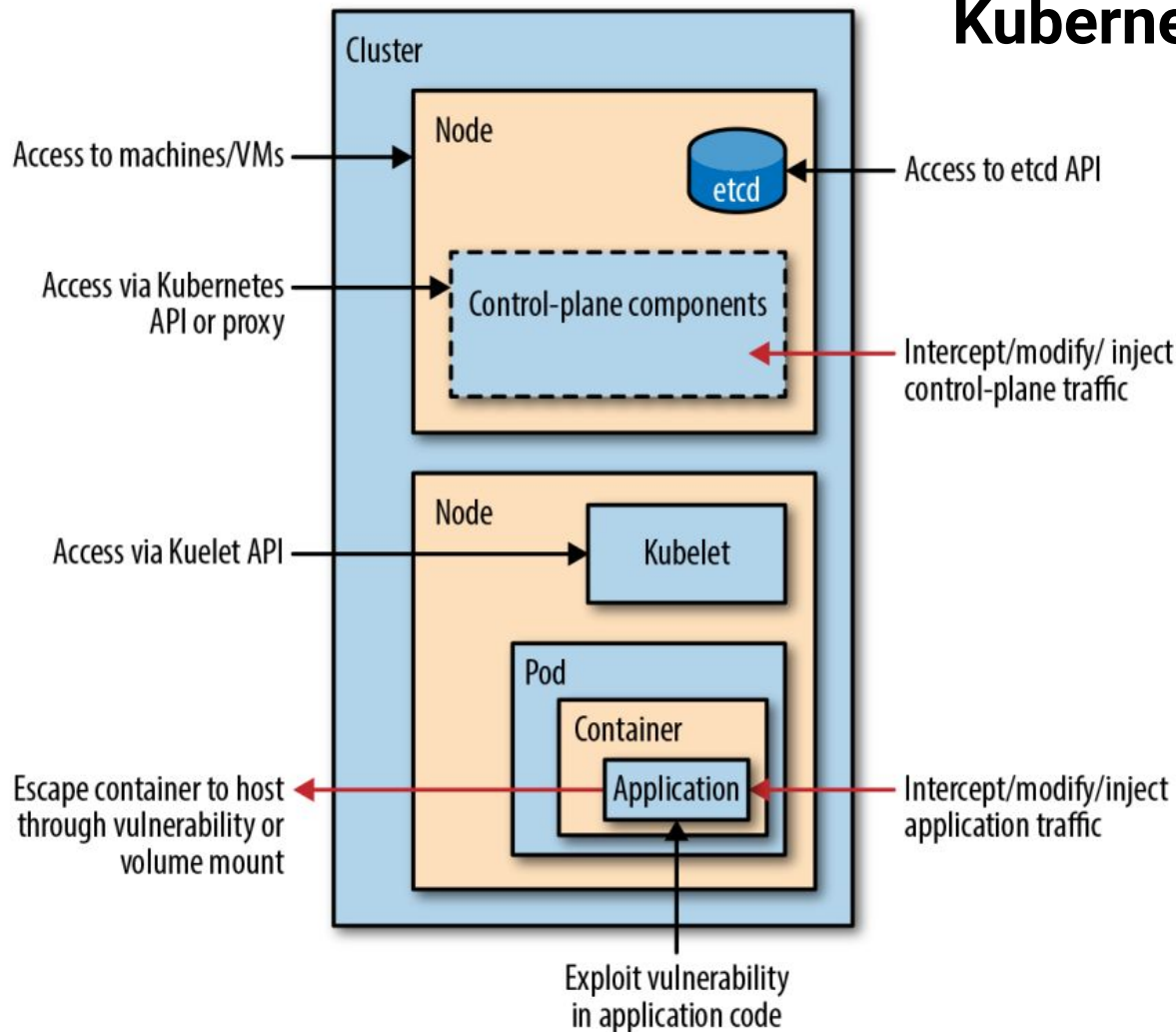
Welcome to Kubernetes where everything runs as root and the security doesn't matter!

14:22 - 8. Mai 2019

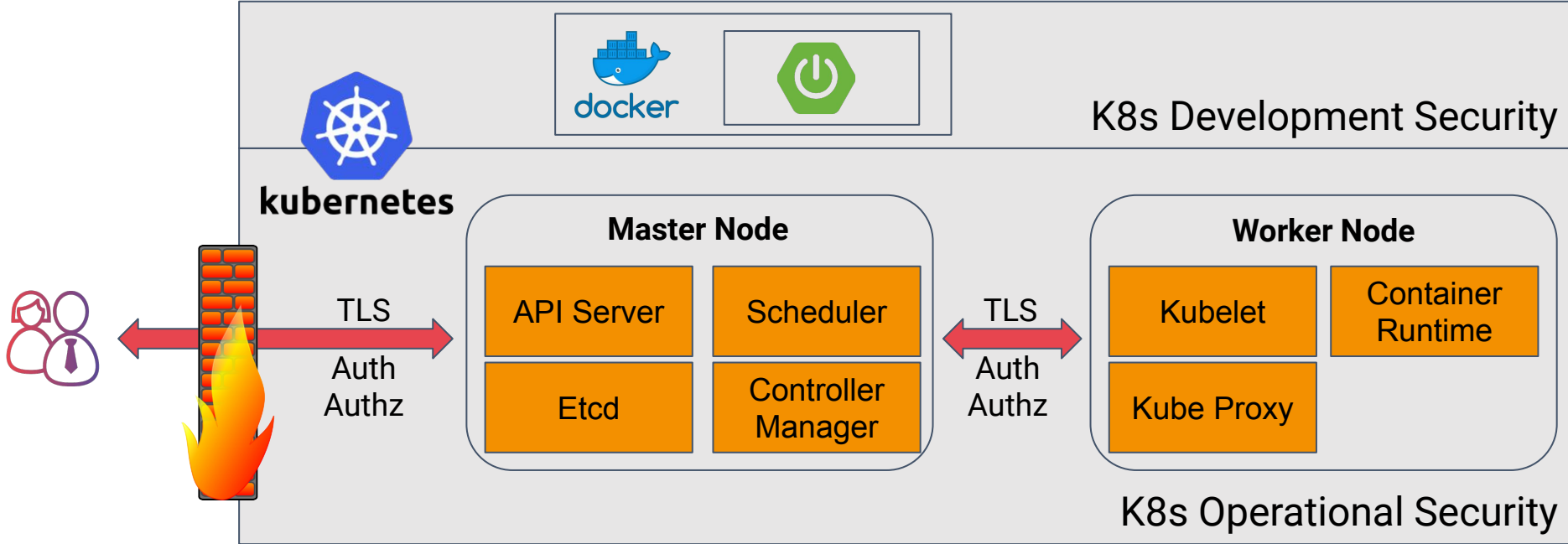
Kubernetes attack vectors

Source:

[Kubernetes Security, O'Reilly, 2018](#)



Operational / Development Kubernetes Security



<https://kubernetes.io/docs/concepts/security/overview/#the-4c-s-of-cloud-native-security>

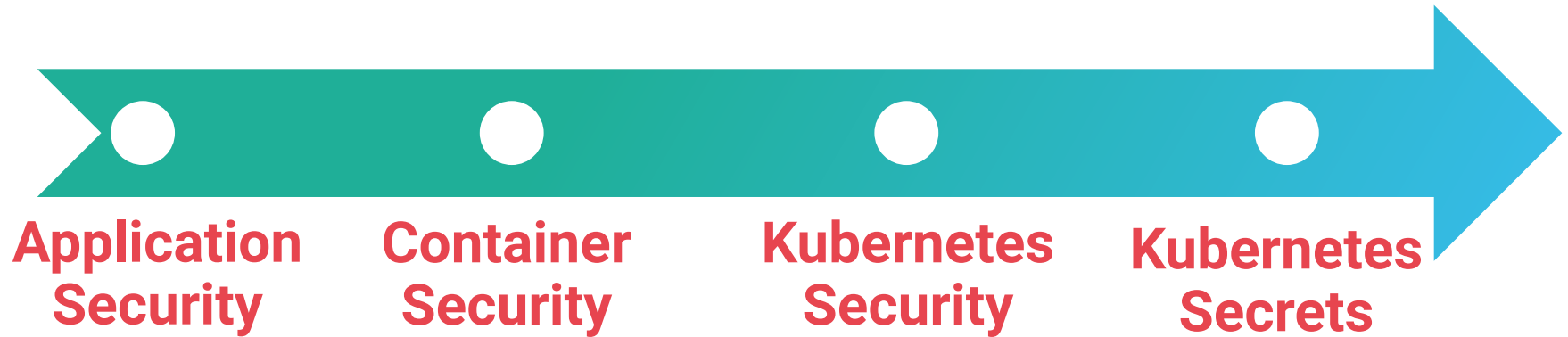
<https://learnk8s.io/production-best-practices/>



So what can we do as developers?

Application- / Docker- / K8s-Security

The Path for Secure Development on K8s



The Path for Secure Development on K8s



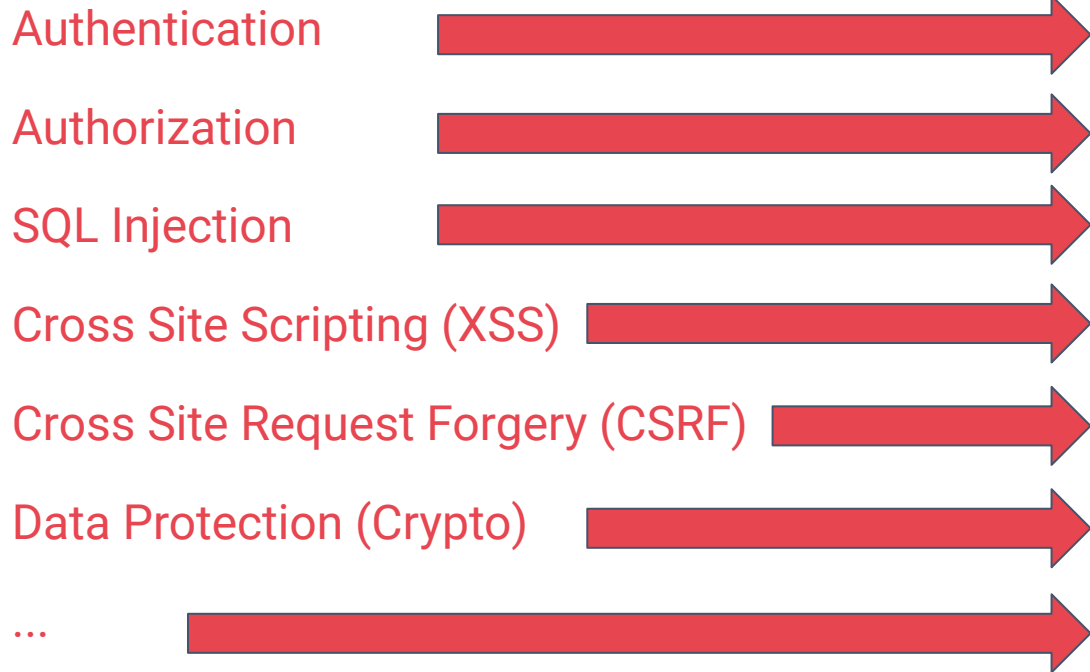
**Application
Security**

**Container
Security**

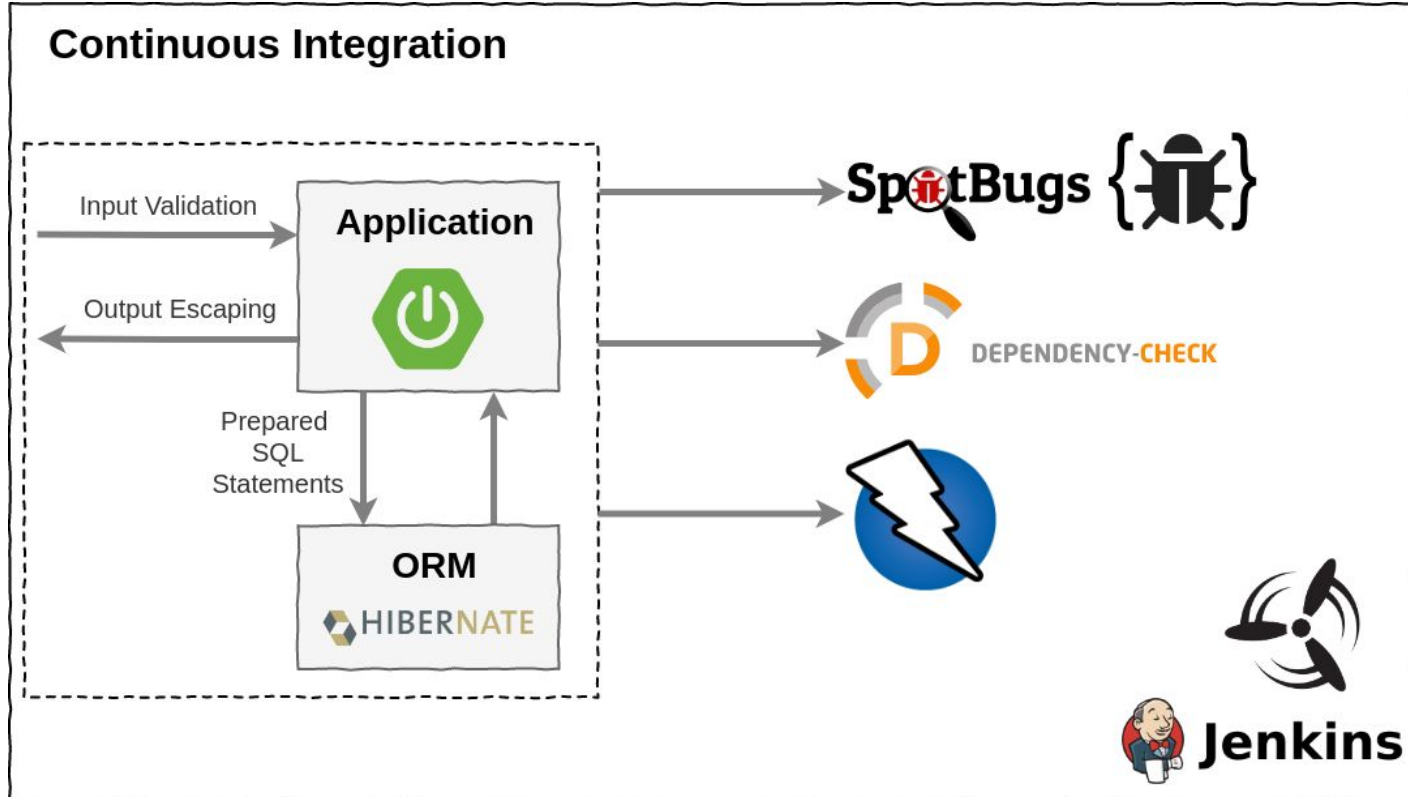
**Kubernetes
Security**

**Kubernetes
Secrets**

Application Security



Application Security





Live Demo: Show me the code

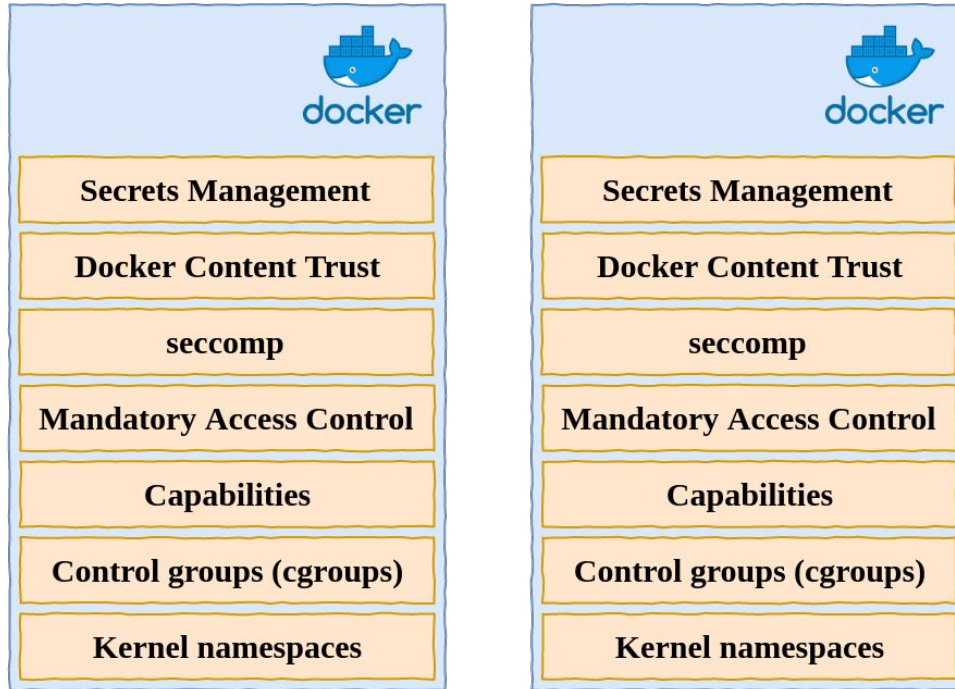
Iteration 1: Application Security

<https://github.com/andifalk/secure-development-on-kubernetes>

The Path for Secure Development on K8s



Docker Security Basics



Linux Host



Linux Kernel Namespaces

- Process ID (pid)
- Network (net)
- Filesystem/mount (mnt)
- Inter-Process Communication (ipc)
- User (user)
- UTS (hostname)

Linux Control Groups (CGroups)

- Resource Limits
 - CPU
 - Memory
 - Devices
 - Processes
 - Network

For Java this only works with container aware JDK versions as of **OpenJDK 8u192** or above

Linux Capabilities

- Break up root privileges into smaller units
 - CAP_SYS_ADMIN
 - CAP_NET_ADMIN
 - CAP_NET_BIND_SERVICE
 - CAP_CHOWN
 - ...

```
$ docker run --cap-drop=ALL --cap-add=NET_BIND_SERVICE
```

<http://man7.org/linux/man-pages/man7/capabilities.7.html>

Mandatory Access Control (MAC)

- AppArmor
- Security Enhanced Linux (SELinux)

<https://gitlab.com/apparmor/apparmor/wikis/home>
<https://github.com/SELinuxProject>

Secure Computing Mode (SecComp)

- Deny critical system calls by default
 - reboot
 - mount
 - swapon
 - ...

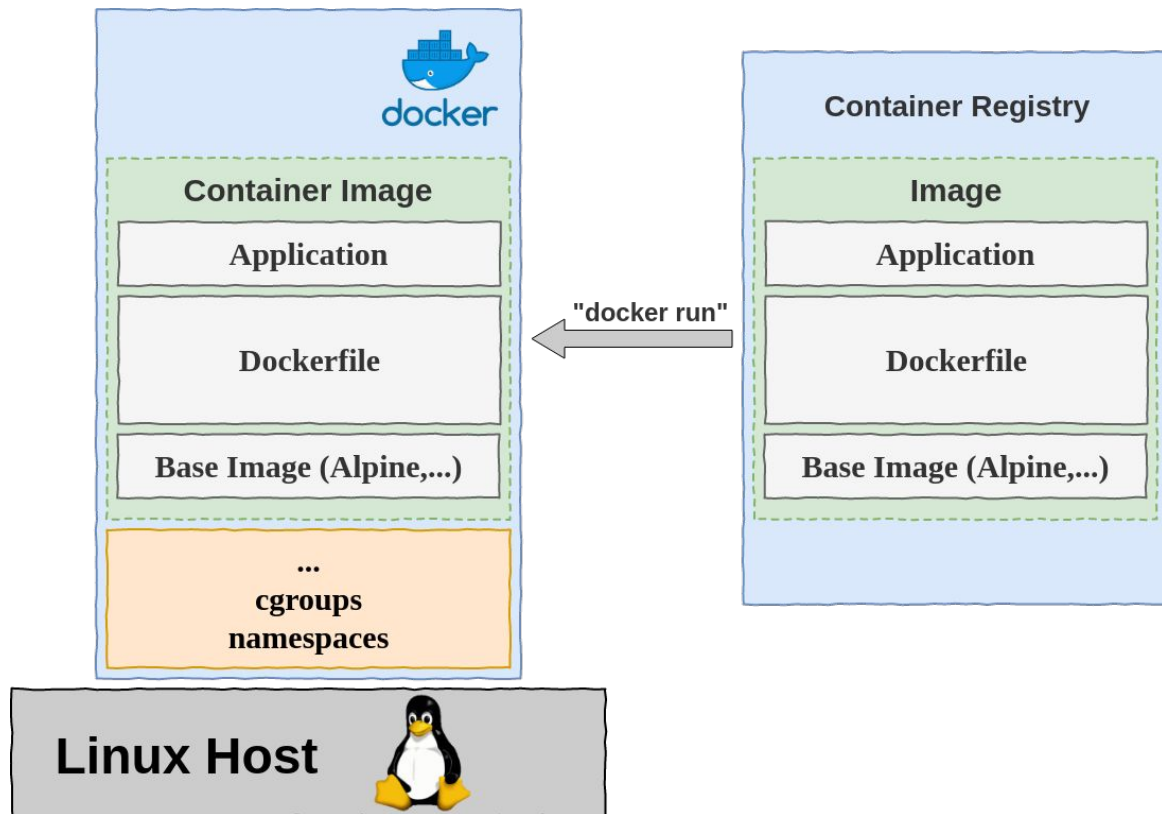
<http://man7.org/linux/man-pages/man2/seccomp.2.html>
<https://docs.docker.com/engine/security/seccomp>

OWASP Docker Top 10

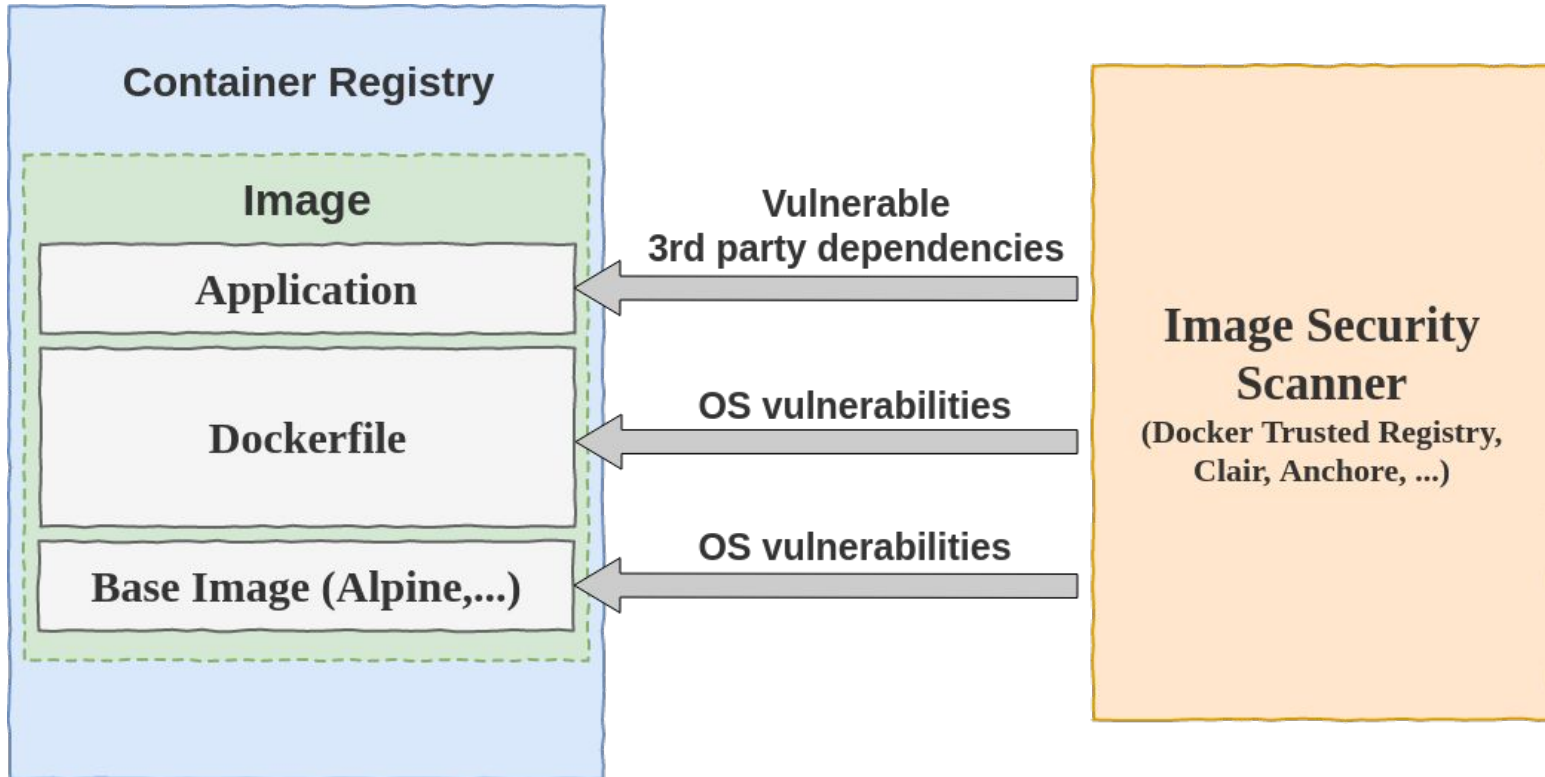
1. Secure User Mapping
2. Patch Management Strategy
3. Network Segmentation and Firewalling
4. Secure Defaults and Hardening
5. Maintain Security Contexts
6. Protect Secrets
7. Resource Protection
8. Container Image Integrity and Origin
9. Follow Immutable Paradigm
10. Logging

<https://github.com/OWASP/Docker-Security>

Docker Images



Docker Image Security



Say No To Root!

USER directive in Dockerfile

```
FROM openjdk:11-jre-slim
COPY hello-spring-kubernetes-1.0.0-SNAPSHOT.jar app.jar
EXPOSE 8080
RUN addgroup --system --gid 1002 app && adduser
      --system --uid 1002 --gid 1002 appuser
USER 1002
ENTRYPOINT java -jar /app.jar
```

<https://opensource.com/article/18/3/just-say-no-root-containers>

Say No To Root!

Use JIB and Distroless Images

```
plugins {  
    id 'com.google.cloud.tools.jib' version '...'  
}  
  
jib {  
    container {  
        user = 1002  
    }  
}
```

<https://github.com/GoogleContainerTools/jib>

Keep Being Secure

- Perform Image Scanning
 - Anchore
 - Clair
 - Trivy
- Regularly Update Base Images

<https://anchore.com/opensource/>

<https://github.com/coreos/clair>

<https://github.com/aquasecurity/trivy>



Live Demo: Show me the code

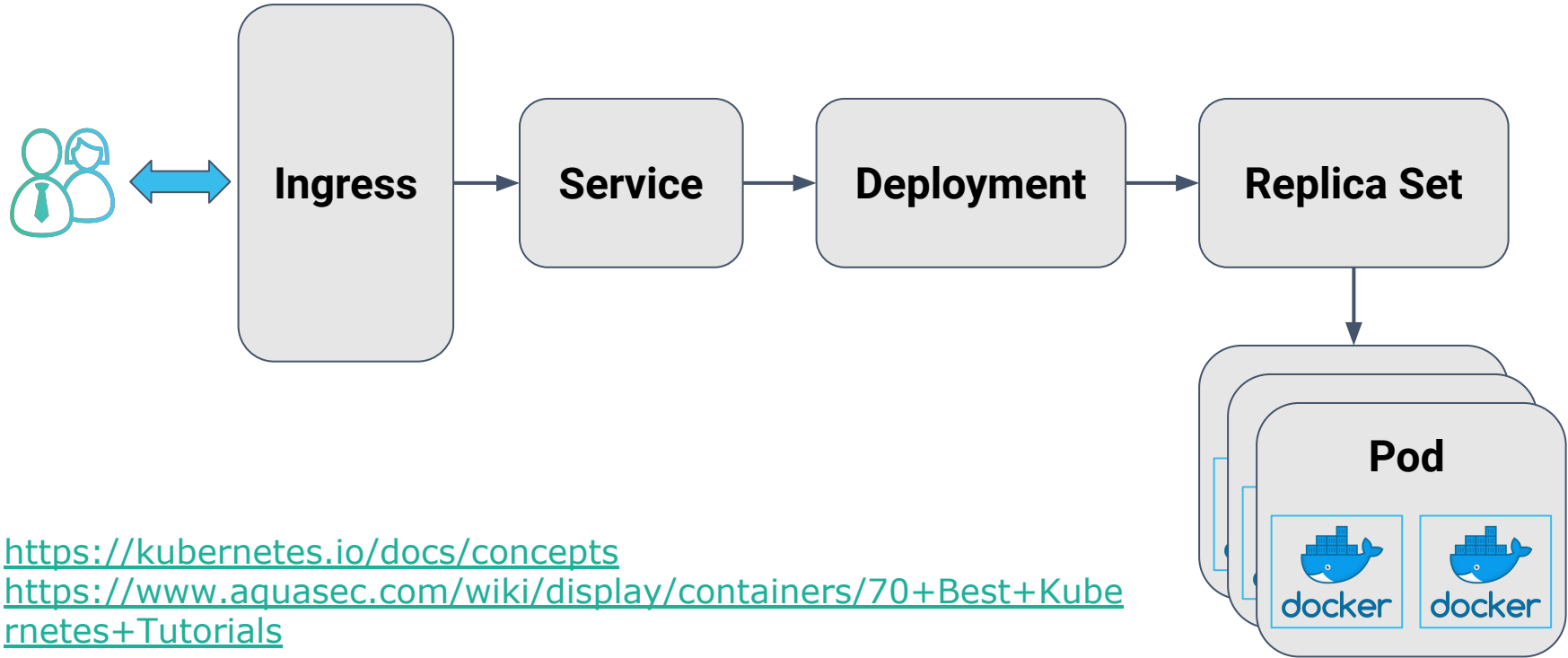
Iteration 2: Container Security

<https://github.com/andifalk/secure-development-on-kubernetes>

The Path for Secure Development on K8s



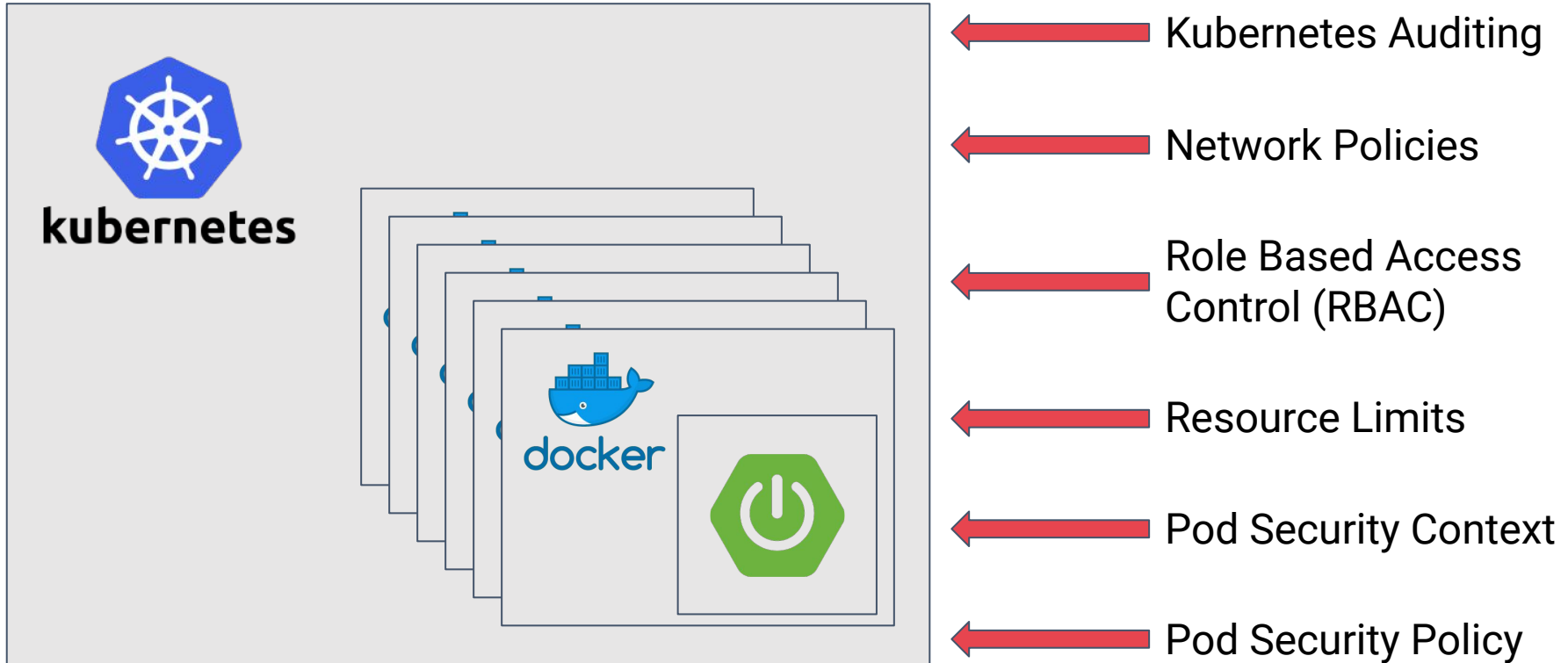
Kubernetes Basics



<https://kubernetes.io/docs/concepts>

<https://www.aquasec.com/wiki/display/containers/70+Best+Kubernetes+Tutorials>

Kubernetes Security



Resource Limits

```
spec:
  ...
  containers:
    resources:
      limits:
        cpu: "1"
        memory: "512Mi"
      requests:
        cpu: 500m
        memory: "256Mi"
    ...
```

<https://kubernetes.io/docs/tasks/configure-pod-container/assign-cpu-resource>

<https://kubernetes.io/docs/tasks/configure-pod-container/assign-memory-resource>

Pod/Container Security Context

```
spec:
  securityContext:
    runAsNonRoot: true
  containers:
    securityContext:
      allowPrivilegeEscalation: false
      privileged: false
      runAsNonRoot: true
      readOnlyRootFilesystem: true
      capabilities:
        drop:
          - ALL
```

<https://kubernetes.io/docs/tasks/configure-pod-container/security-context>

Pod Security Policy (Still In Beta!)

```
apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
  name: no-root-policy
spec:
  privileged: false
  allowPrivilegeEscalation: false
  requiredDropCapabilities:
    - ALL
  runAsUser:
    rule: 'MustRunAsNonRoot'
  ...
```

<https://kubernetes.io/docs/concepts/policy/pod-security-policy>

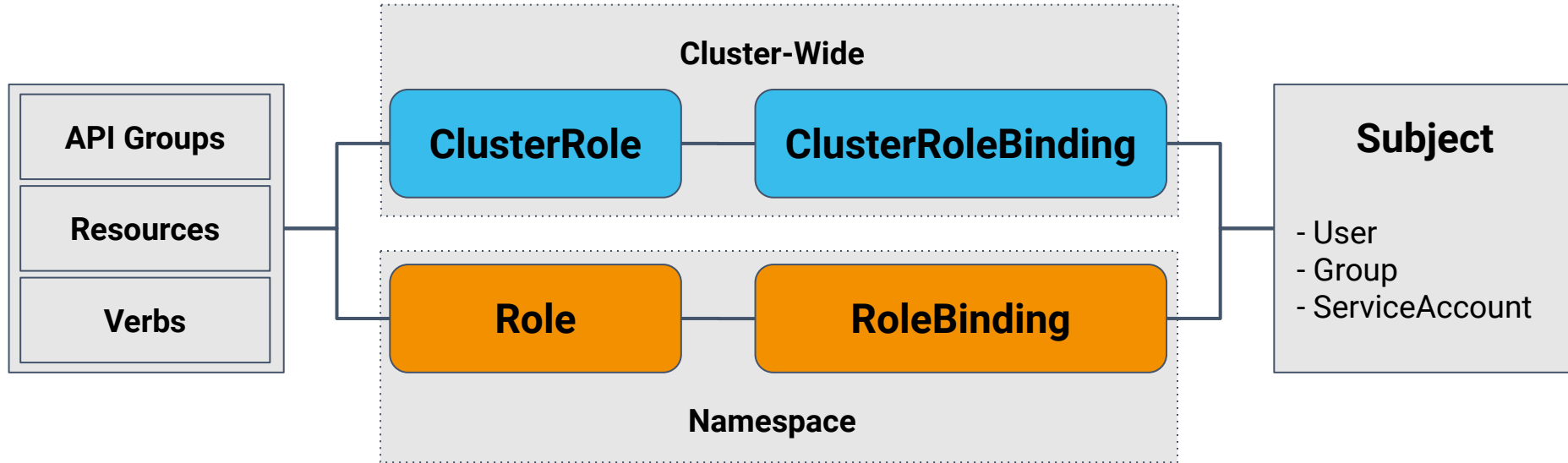
Pod Security Policy (Policy Order)

Policy order selection criteria:

1. Policies which allow the pod as-is are preferred
2. If pod must be defaulted or mutated, the first policy (ordered by name) to allow the pod is selected.

<https://kubernetes.io/docs/concepts/policy/pod-security-policy/#policy-order>
<https://kubernetes.io/docs/reference/access-authn-authz/admission-controllers>

Kubernetes Role Based Access Control (RBAC)



<https://kubernetes.io/docs/reference/access-authn-authz/rbac/>

Kubernetes Role Based Access Control (RBAC)

apiGroups	extensions, apps, policy, ...
resources	pods, deployments, configmaps, secrets, nodes, services, endpoints, podsecuritypolicies, ...
verbs	get, list, watch, create, update, patch, delete, use, ...

<https://kubernetes.io/docs/reference/access-authn-authz/rbac/>

Service Account

```
apiVersion: v1
kind: ServiceAccount
metadata:
  name: deploy-pod-security-policy
  namespace: default
```

<https://kubernetes.io/docs/concepts/policy/pod-security-policy/#authorizing-policies>

Pod Security Policy Role

```
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: no-root-policy-role
  namespace: default
rules:
  - apiGroups: ['policy']
    resources: ['podsecuritypolicies']
    verbs:     ['use']
    resourceNames:
      - no-root-policy
```

<https://kubernetes.io/docs/concepts/policy/pod-security-policy/#authorizing-policies>

Pod Security Policy Role Binding

```
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: deploy-pod-security-policy
  namespace: default
roleRef:
  kind: Role
  name: no-root-policy-role
  apiGroup: rbac.authorization.k8s.io
subjects:
- kind: ServiceAccount
  name: deploy-pod-security-policy
  namespace: default
```

Helm 3 Is Here! 😊



Ian Coldwater

@IanColdwater



Folge ich



For people who don't pay attention to the Kubernetes ecosystem: Helm 3.0 is a big deal, removing Tiller and drastically improving the security of that project. Great work, y'all!



Live Demo: Show me the code

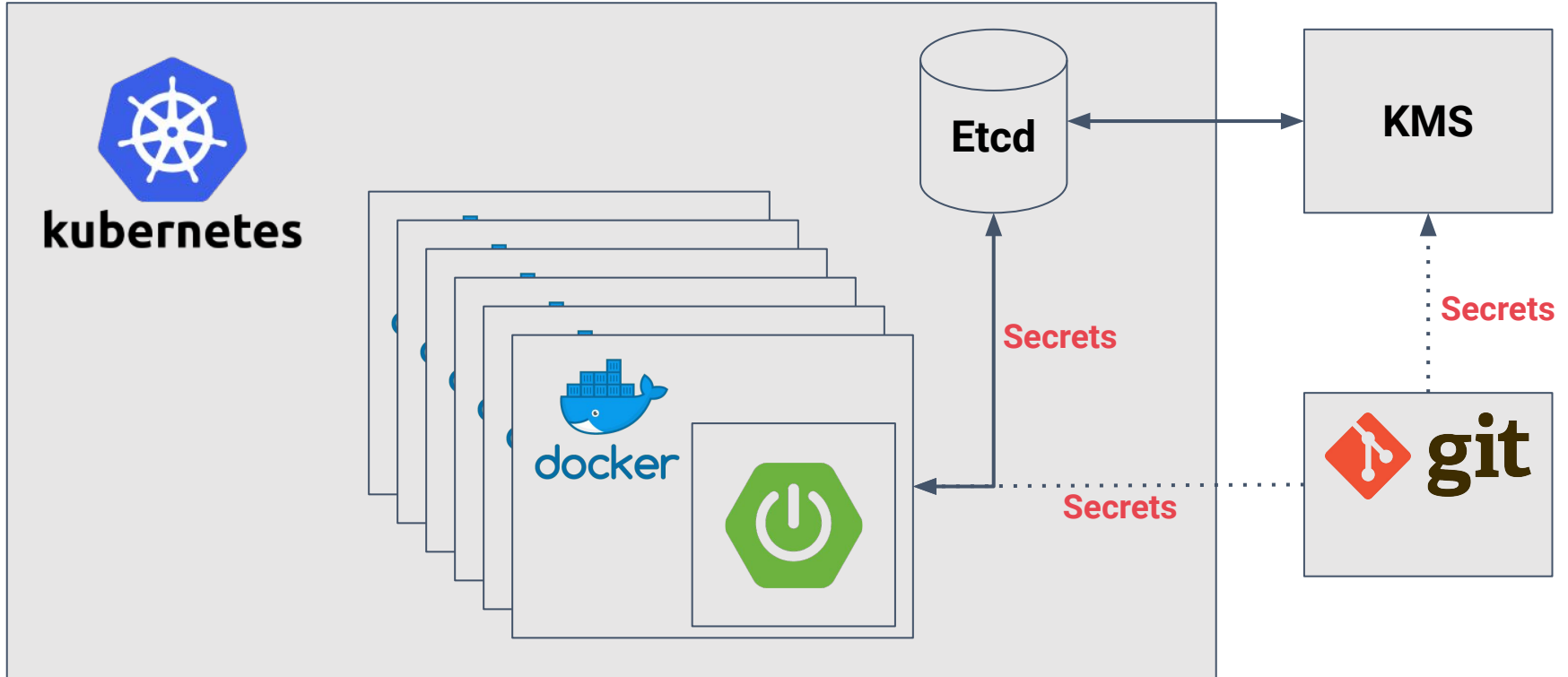
Iteration 3: Kubernetes Security

<https://github.com/andifalk/secure-development-on-kubernetes>

The Path for Secure Development on K8s



Kubernetes Secrets



Kubernetes Secrets

```
apiVersion: v1
kind: Secret
metadata:
  name: hello-spring-cloud-kubernetes
  namespace: default
type: Opaque
data:
  user.username: dXNlcmg==
  user.password: azhzX3VzZXI=
  admin.username: YWRtaW4=
  admin.password: azhzX2FkbWlu
```

<https://kubernetes.io/docs/concepts/configuration/secret>

Kubernetes Secrets - Best Practices

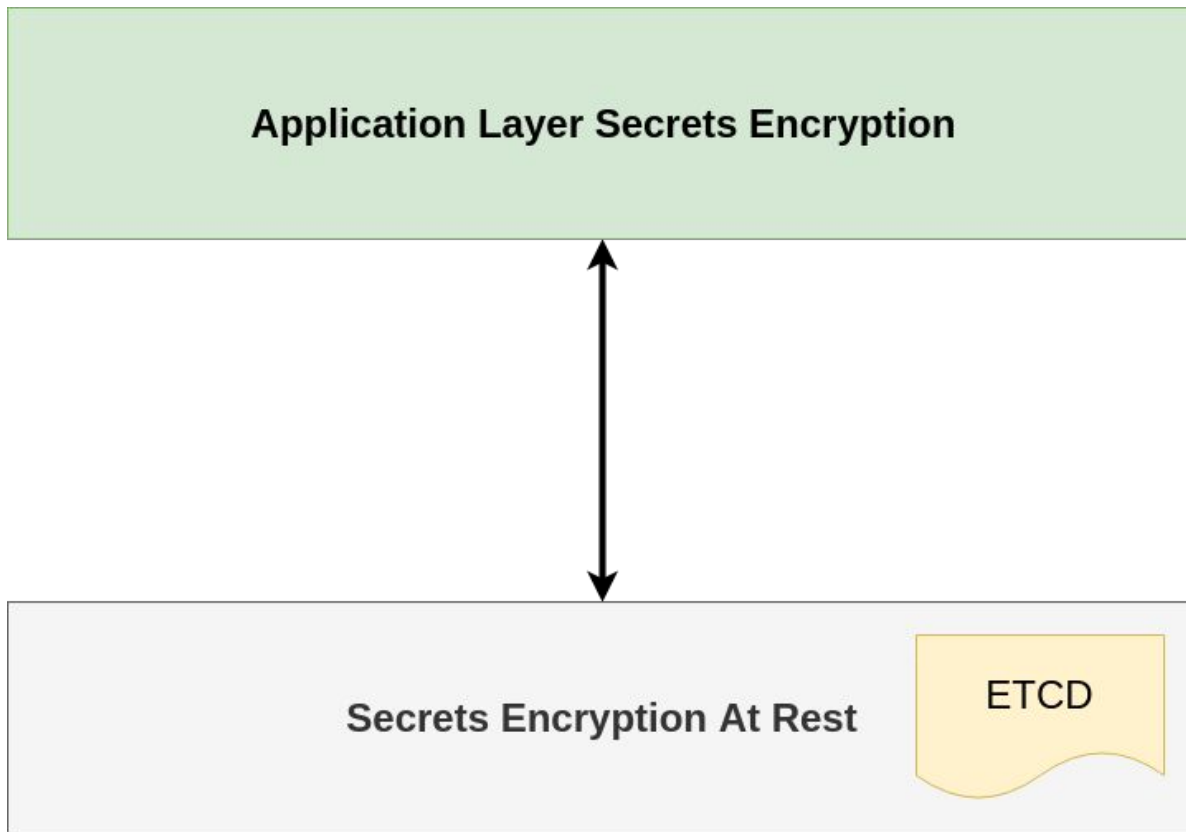
- Encrypt Secret Data at Rest
Only Base64 Encoded by Default!
- Applications interacting with secrets API should be limited using RBAC
- Mount secrets instead of ENV Mapping

<https://kubernetes.io/docs/concepts/configuration/secret/#best-practices>
<https://kubernetes.io/docs/tasks/administer-cluster/encrypt-data>

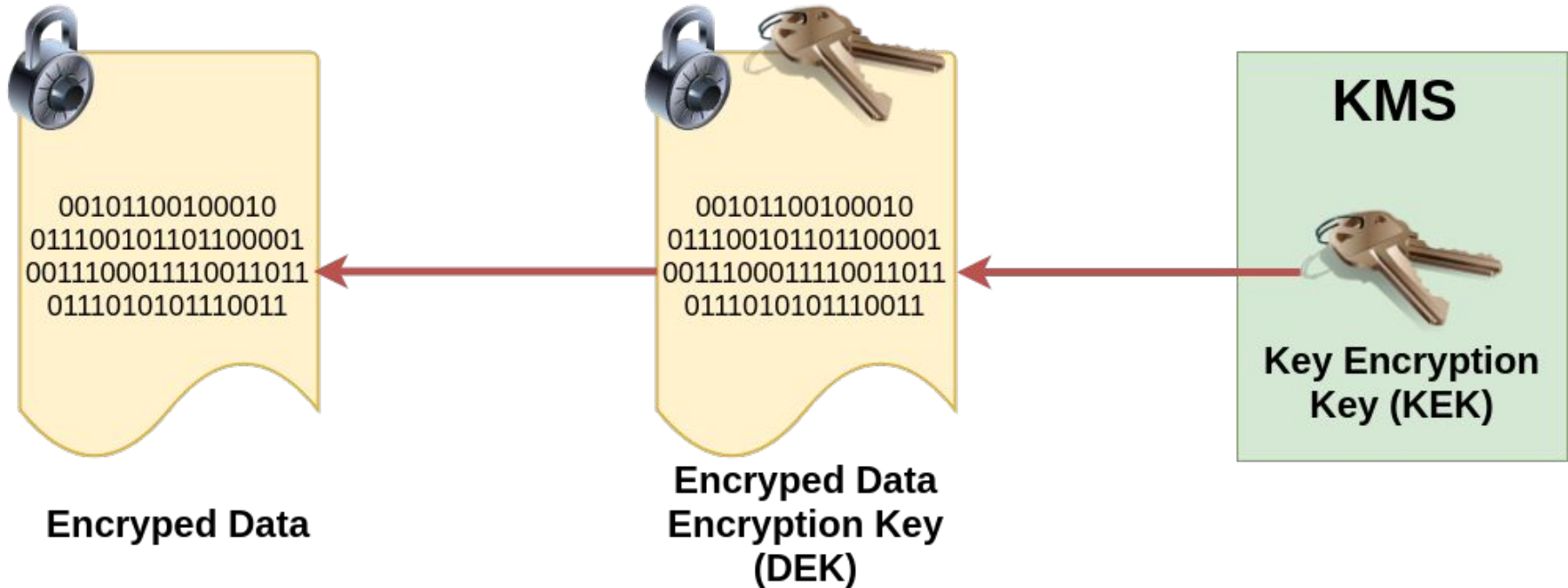
Pay Attention to Spring Boot Actuator

```
{
  "name": "applicationConfig: ...",
  "properties": {
    "greet.my-sec": {
      "value": "geheim",
      "origin": "class path resource ..."
    },
    "greet.password": {
      "value": "*****",
      "origin": "class path resource ..."
    }
  }
}
```

Encryption Layers



Envelope Encryption On Kubernetes



<https://cloud.google.com/kms/docs/envelope-encryption>
<https://kubernetes.io/docs/tasks/administer-cluster/kms-provider>

Key Management System (KMS) Cloud Providers

- Azure Key Vault (Key Vault FlexVolume)
- Google Cloud KMS
- AWS KMS
- ...

<https://github.com/Azure/kubernetes-kms>

<https://github.com/Azure/kubernetes-keyvault-flexvol>

<https://cloud.google.com/kms>

<https://aws.amazon.com/de/kms>

What about Secrets in git

- Sealed Secrets
- Helm Secrets
- Kamus
- Sops
- Hashicorp Vault

<https://learnk8s.io/kubernetes-secrets-in-git>

<https://github.com/bitnami-labs/sealed-secrets>

<https://github.com/futuresimple/helm-secrets>

<https://github.com/Soluto/kamus>

<https://github.com/mozilla/sops>

<https://www.vaultproject.io>

Summary



Summary / Key Insights

- Containers use Linux Namespaces+Caps
- Say **NO** to root on K8s
- “**Least privilege**” for service accounts
- Keep K8s up-to-date and scan for security
- Ensure your secrets are **encrypted** in K8s
- Keep K8s and container images **up-to-date**



Books and Online References

Books and Online References (1)

- [Kubernetes Security, O'Reilly, 2018, ISBN: 978-1-492-04600-4](#)
- [Cloud Native DevOps with Kubernetes, O'Reilly, 2019, ISBN: 978-1492040767](#)
- [<https://github.com/andifalk/secure-development-on-kubernetes>](#)
- [Crafty Requests: Deep Dive Into Kubernetes CVE-2018-1002105 - Ian Coldwater \(Video\)](#)
- [Ship of Fools: Shoring Up Kubernetes Security - Ian Coldwater \(Video\)](#)
- [<https://kubernetes.io/docs/concepts/security/overview/#the-4c-s-of-cloud-native-security>](#)
- [<https://kubernetes.io/docs/tasks/administer-cluster/securing-a-cluster>](#)
- [<https://opensource.com/article/18/3/just-say-no-root-containers>](#)
- [<https://github.com/GoogleContainerTools/jib>](#)
- [<https://anchore.com/opensource/>](#)
- [<https://github.com/coreos/clair>](#)
- [<https://github.com/aquasecurity/trivy>](#)
- [\[https://www.owasp.org/index.php/OWASP_Docker_Top_10\]\(https://www.owasp.org/index.php/OWASP_Docker_Top_10\)](#)

Books and Online References (2)

- <https://kubernetes.io/docs/tasks/configure-pod-container/assign-cpu-resource>
- <https://kubernetes.io/docs/tasks/configure-pod-container/assign-memory-resource>
- <https://kubernetes.io/docs/tasks/configure-pod-container/security-context>
- <https://kubernetes.io/docs/concepts/policy/pod-security-policy>
- <https://kubernetes.io/docs/reference/access-authn-authz/rbac/>
- <https://kubernetes.io/docs/concepts/configuration/secret>
- <https://kubernetes.io/docs/tasks/administer-cluster/encrypt-data>
- <https://cloud.google.com/kms/docs/envelope-encryption>
- <https://kubernetes.io/docs/tasks/administer-cluster/kms-provider>
- <https://github.com/Azure/kubernetes-kms>
- <https://cloud.google.com/kms>
- <https://aws.amazon.com/de/kms>



Andreas Falk

Managing Consultant

Mobil: +49 151 46146778

E-Mail: andreas.falk@novatec-gmbh.de

Novatec Consulting GmbH

Dieselstraße 18/1

D-70771 Leinfelden-Echterdingen

T. +49 711 22040-700

info@novatec-gmbh.de

www.novatec-gmbh.de