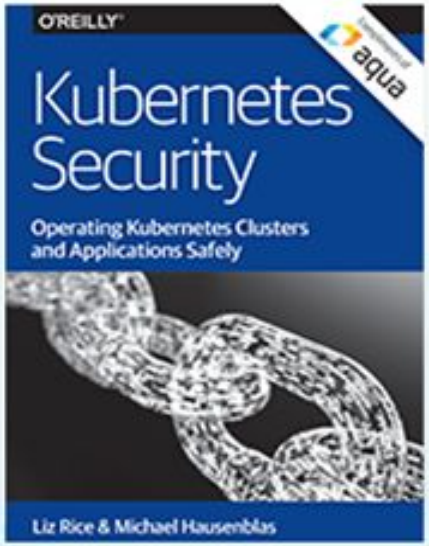
Preventative Security for Kubernetes

Liz Rice

@lizrice | @aquasecteam

Agenda

- Kubernetes configuration for security
- CIS benchmarks – testing the configuration
- Penetration testing – testing for vulnerabilities



Kubernetes Security: Operating Kubernetes Clusters and Applications Safely

A practical book that walks you through Kubernetes security features, when to use what, and how to augment those features with container image best practices and secure network communication.

Download the Book ›

Authored by Liz Rice from Aqua Security and Michael Hausenblas from Red Hat

https://info.aquasec.com/kubernetes-security

Aqua: our approach

Automate DevSecOps

- Secure the CI/CD pipeline
- “Shift left” security, fix issues early and fast
- Accelerate app delivery with security automation

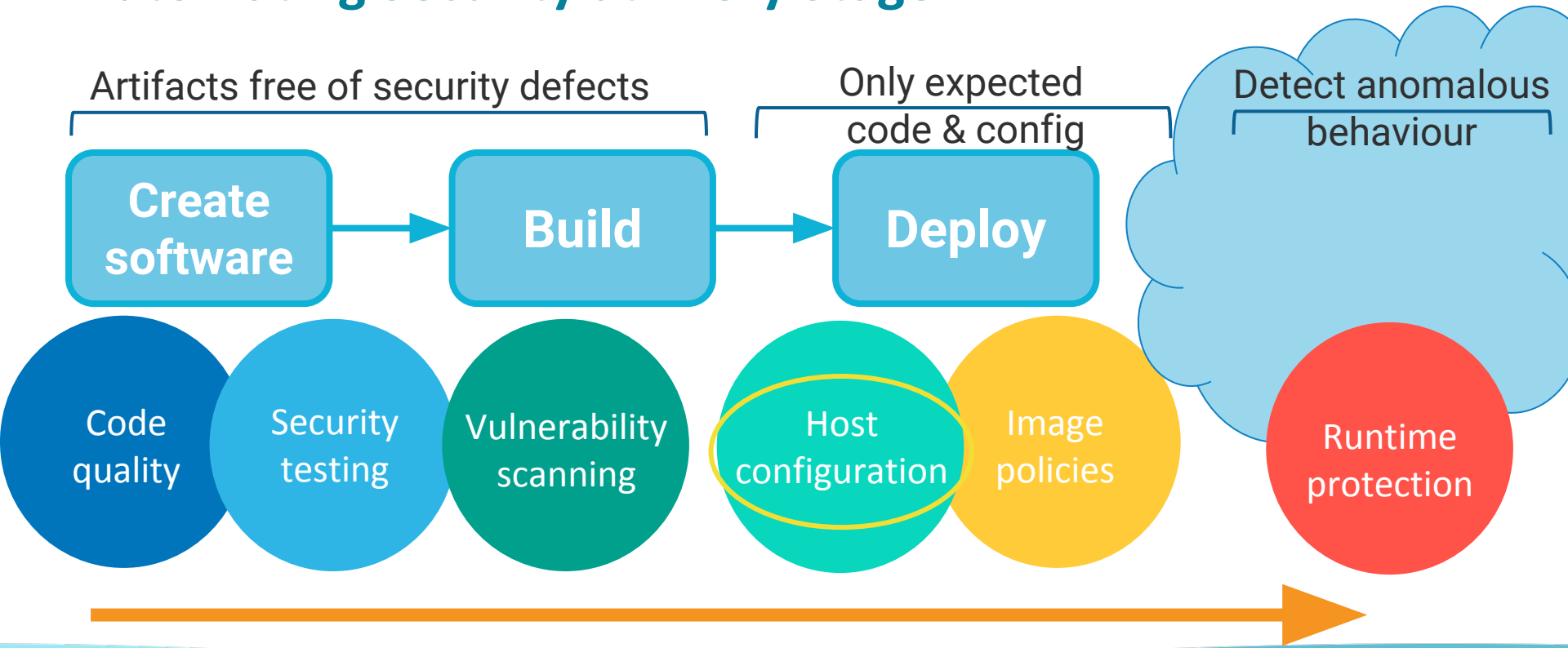
Modernize security through containers

- Enforce immutability – no patching, no drift
- Whitelist good behavior, preventing anomalies
- Prevent lateral movement

Secure once, run anywhere

- Secure apps regardless of platform, cloud, or OS
- Enable hybrid cloud and cloud migration
- Avoid cloud lock-in and security reconfiguration

Automating Security at Every Stage



Kubernetes Host Configuration

Kubernetes configuration

- Kubernetes components installed on your servers
  - Master & node components
- Many configuration settings have a security impact
  - Example: open Kubelet port = root access
- Defaults depend on the installer

CIS Kubernetes Benchmark

kube-bench

- Open source automated tests for CIS Kubernetes Benchmark
- Tests for Kubernetes Masters and Nodes
- Available as a container

`github.com/aquasecurity/kube-bench`

kube-bench

```
[INFO] 1 Master Node Security Configuration
[INFO] 1.1 API Server
[FAIL] 1.1.1 Ensure that the --allow-privileged argument is set to false (Scored)
[FAIL] 1.1.2 Ensure that the --anonymous-auth argument is set to false (Scored)
[PASS] 1.1.3 Ensure that the --basic-auth-file argument is not set (Scored)
[PASS] 1.1.4 Ensure that the --insecure-allow-any-token argument is not set (Scored)
[FAIL] 1.1.5 Ensure that the --kubelet-https argument is set to true (Scored)
[PASS] 1.1.6 Ensure that the --insecure-bind-address argument is not set (Scored)
[PASS] 1.1.7 Ensure that the --insecure-port argument is set to 0 (Scored)
[PASS] 1.1.8 Ensure that the --secure-port argument is not set to 0 (Scored)
[FAIL] 1.1.9 Ensure that the --profiling argument is set to false (Scored)
[FAIL] 1.1.10 Ensure that the --repair-malformed-updates argument is set to false (Scored)
[PASS] 1.1.11 Ensure that the admission control policy is not set to AlwaysAdmit (Scored)
[FAIL] 1.1.12 Ensure that the admission control policy is set to AlwaysPullImages (Scored)
[FAIL] 1.1.13 Ensure that the admission control policy is set to DenyEscalatingExec (Scored)
[FAIL] 1.1.14 Ensure that the admission control policy is set to SecurityContextDeny (Scored)
[PASS] 1.1.15 Ensure that the admission control policy is set to NamespaceLifecycle (Scored)
[FAIL] 1.1.16 Ensure that the --audit-log-path argument is set as appropriate (Scored)
[FAIL] 1.1.17 Ensure that the --audit-log-maxage argument is set to 30 or as appropriate (Scored)
[FAIL] 1.1.18 Ensure that the --audit-log-maxbackup argument is set to 10 or as appropriate (Scored)
[FAIL] 1.1.19 Ensure that the --audit-log-maxsize argument is set to 100 or as appropriate (Scored)
[PASS] 1.1.20 Ensure that the --authorization-mode argument is not set to AlwaysAllow (Scored)
[PASS] 1.1.21 Ensure that the --token-auth-file parameter is not set (Scored)
[FAIL] 1.1.22 Ensure that the --kubelet-certificate-authority argument is set as appropriate (Scored)
```

kube-bench

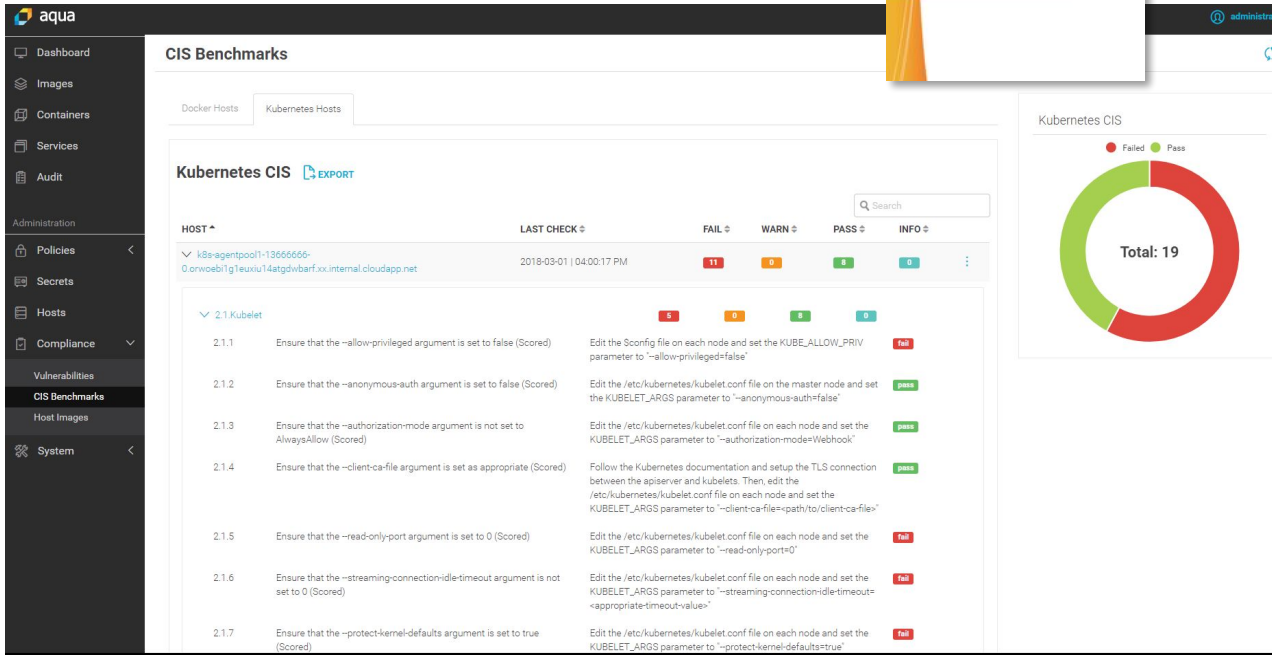
- Job configuration YAML
  - Run regularly to ensure no configuration drift
- Tests defined in YAML
  - Released code follows the CIS Benchmark
  - Modify for your own purposes

`github.com/aquasecurity/kube-bench`

kube-bench

Kubernetes & Docker CIS Benchmarks

- Built into the Aqua CSP
- Provides a scored report of the results
- Can be scheduled to run daily



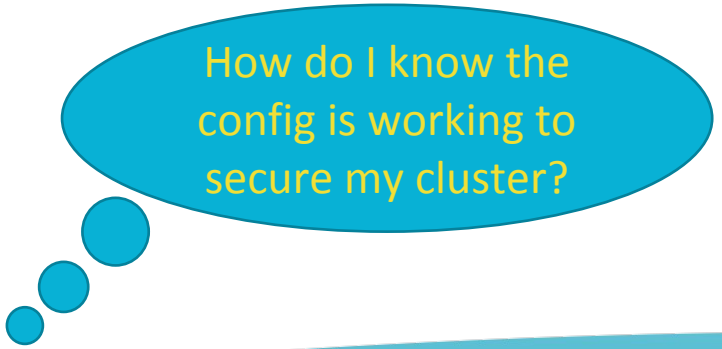
aqua

Kubernetes penetration testing

kube-hunter

- Open source penetration tests for Kubernetes
  - See what an attacker would see
  - `github.com/aquasecurity/kube-hunter`

- Online report viewer
  - `kube-hunter.aquasec.com`



kube-hunter.aquasec.com

```
Choose one of the options below:
1. Remote scanning      (scans one or more specific IPs or DNS names)
2. Subnet scanning      (scans subnets on all local network interfaces)
3. IP range scanning    (scans a given IP range)
Your choice: 1
Remotes (separated by a ','): 172.28.128.3
~ Started
~ Discovering Open Kubernetes Services...
|
| API Server:
|   type: open service
|   service: API Server
|_  host: 172.28.128.3:6443
|
| Kubelet API (readonly):
|   type: open service
|   service: Kubelet API (readonly)
|_  host: 172.28.128.3:10255
|
| Kubelet API:
|   type: open service
|   service: Kubelet API
|_  host: 172.28.128.3:10250
|
| Anonymous Authentication:
```

172.28.128.3

Node / Master

12 vulnerabilities

SEVERITY	CATEGORY	VULNERABILITY	DESCRIPTION	EVIDENCE
High	Remote Code Execution	Exposed Attaching To Container	Opens a websocket that could enable an attacker to attach to a running container	
High	Remote Code Execution	Anonymous Authentication	The kubelet is misconfigured, potentially allowing secure access to all requests on the kubelet, without the need to authenticate	
High	Remote Code Execution	Exposed Run Inside Container	An attacker could run an arbitrary command inside a container	
High	Remote Code Execution	Exposed Exec On Container	An attacker could run arbitrary commands on a container	
Medium	Information Disclosure	K8s Version Disclosure	The kubernetes version could be obtained from logs in the /metrics endpoint	v1.9.0
Medium	Information Disclosure	Exposed Pods	An attacker could view sensitive information about pods that are bound to a Node using the /pods endpoint	count: 9
Medium	Information Disclosure	Cluster Health Disclosure	By accessing the open /healthz handler, an attacker could get the cluster health state without authenticating	status: ok
Medium	Information Disclosure	Exposed Running Pods	Outputs a list of currently running pods, and some of their metadata, which can reveal sensitive information	9 running pods
Medium	Information Disclosure	Exposed Container Logs	Output logs from a running container are using the exposed /containerLogs endpoint	

kube-hunter with kube-bench

172.28.128.3

Node / Master

12 vulnerabilities

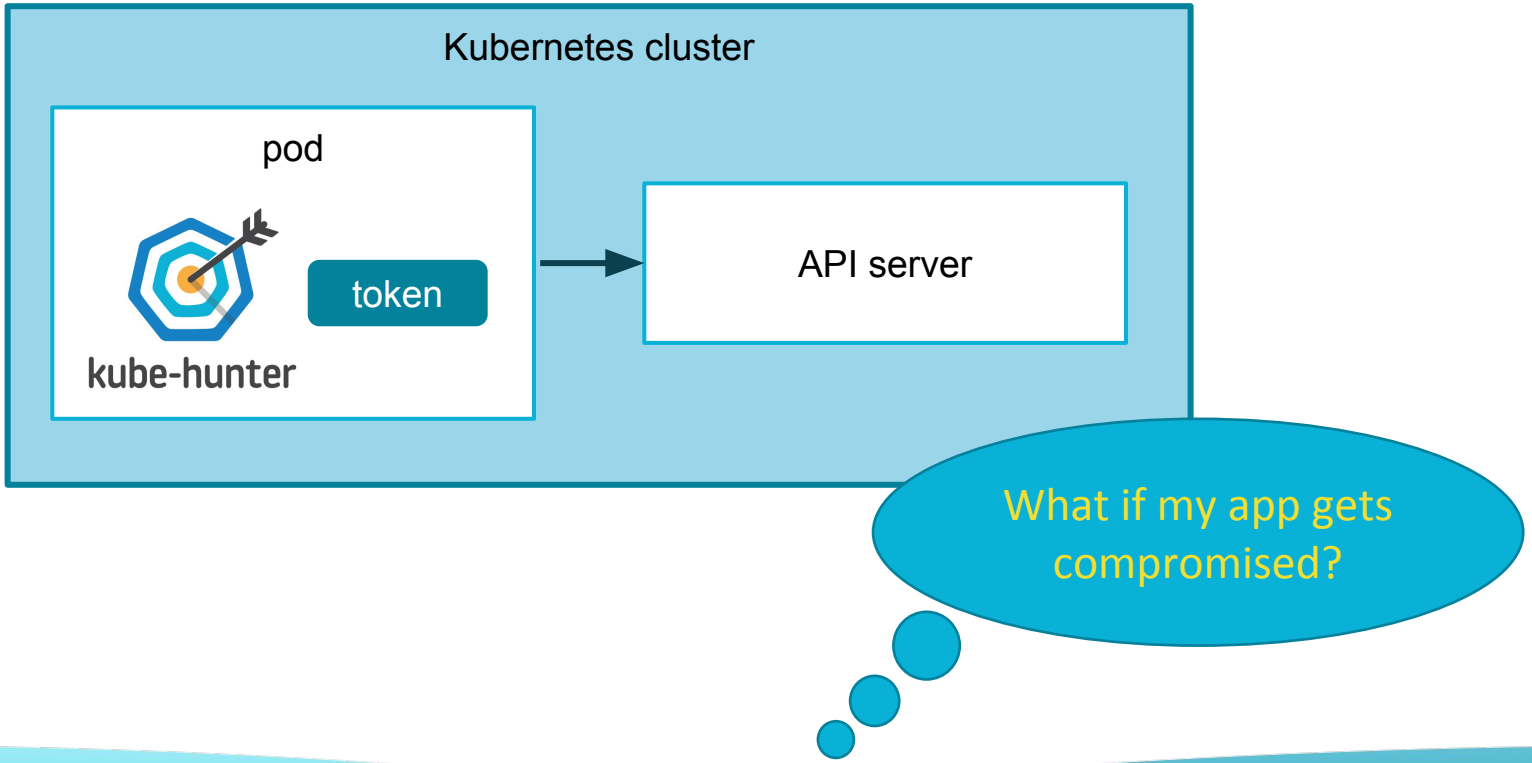
SEVERITY	CATEGORY	VULNERABILITY	DESCRIPTION	EVIDENCE
High	Remote Code Execution	Exposed Attaching To Container	Opens a websocket that could enable an attacker to attach to a running container	
High	Remote Code Execution	Anonymous Authentication	The kubelet is misconfigured, potentially allowing secure access to all requests on the kubelet, without the need to authenticate	
High	Remote Code Execution	Exposed Run Inside Container	An attacker could run an arbitrary command inside a container	
High	Remote Code Execution	Exposed Exec On Container	An attacker could run arbitrary commands on a container	
Medium	Information Disclosure	K8s Version Disclosure	The kubernetes version could be obtained from logs in the /metrics endpoint	v1.9.0
Medium	Information Disclosure	Exposed Pods	An attacker could view sensitive information about pods that are bound to a Node using the /pods endpoint	count: 9
Medium	Information Disclosure	Cluster Health Disclosure	By accessing the open /healthz handler, an attacker could get the cluster health state without authenticating	status: ok
Medium	Information Disclosure	Exposed Running Pods	Outputs a list of currently running pods, and some of their metadata, which can reveal sensitive information	9 running pods
Medium	Information Disclosure	Exposed Container Logs	Output logs from a running container are using the exposed /containerLogs endpoint	

```
[INFO] 2 Worker Node Security Configuration
[INFO] 2.1 Kubelet
[FAIL] 2.1.1 Ensure that the --allow-privileged argument is set to false (Scored)
[FAIL] 2.1.2 Ensure that the --anonymous-auth argument is set to false (Scored)
[FAIL] 2.1.3 Ensure that the --authorization-mode argument is not set to AlwaysAllow (Scored)
[PASS] 2.1.4 Ensure that the --client-ca-file argument is set as appropriate (Scored)
[FAIL] 2.1.5 Ensure that the --read-only-port argument is set to 0 (Scored)
[FAIL] 2.1.6 Ensure that the --streaming-connection-idle-timeout argument is not set to 0 (Scored)
[FAIL] 2.1.7 Ensure that the --protect-kernel-defaults argument is set to true (Scored)
[FAIL] 2.1.8 Ensure that the --make-iptables-util-chains argument is set to true (Scored)
[FAIL] 2.1.9 Ensure that the --keep-terminated-pod-volumes argument is set to false (Scored)
[PASS] 2.1.10 Ensure that the --hostname-override argument is not set (Scored)
[FAIL] 2.1.11 Ensure that the --event-qps argument is set to 0 (Scored)
[PASS] 2.1.12 Ensure that the --tls-cert-file and --tls-private-key-file arguments are set as approp
[PASS] 2.1.13 Ensure that the --cadvisor-port argument is set to 0 (Scored)
[FAIL] 2.1.14 Ensure that the RotateKubeletClientCertificate argument is set to true
[FAIL] 2.1.15 Ensure that the RotateKubeletServerCertificate argument is set to true
[INFO] 2.2 Configuration Files
[FAIL] 2.2.1 Ensure that the kubelet.conf file permissions are set to 644 or more restrictive (Score
[FAIL] 2.2.2 Ensure that the kubelet.conf file ownership is set to root:root (Scored)
[FAIL] 2.2.3 Ensure that the kubelet service file permissions are set to 644 or more restrictive (Sc
[FAIL] 2.2.4 2.2.4 Ensure that the kubelet service file ownership is set to root:root (Scored)
[FAIL] 2.2.5 Ensure that the proxy kubeconfig file permissions are set to 644 or more restrictive (S
[FAIL] 2.2.6 Ensure that the proxy kubeconfig file ownership is set to root:root (Scored)
[WARN] 2.2.7 Ensure that the certificate authorities file permissions are set to 644 or more restric
[WARN] 2.2.8 Ensure that the client certificate authorities file ownership is set to root:root
```

```
== Remediations ==
2.1.1 Edit the kubelet service file /etc/systemd/system/kubelet.service.d/10-kubeadm.conf
on each worker node and set the below parameter in KUBELET_SYSTEM_PODS_ARGS variable.
--allow-privileged=false
Based on your system, restart the kubelet service. For example:
systemctl daemon-reload
systemctl restart kubelet.service

2.1.2 Edit the kubelet service file /etc/systemd/system/kubelet.service.d/10-kubeadm.conf
on each worker node and set the below parameter in KUBELET_SYSTEM_PODS_ARGS variable.
--anonymous-auth=false
Based on your system, restart the kubelet service. For example:
systemctl daemon-reload
systemctl restart kubelet.service

2.1.3 Edit the kubelet service file /etc/systemd/system/kubelet.service.d/10-kubeadm.conf
on each worker node and set the below parameter in KUBELET_AUTHZ_ARGS variable.
--authorization-mode=Webhook
Based on your system, restart the kubelet service. For example:
systemctl daemon-reload
systemctl restart kubelet.service

2.1.5 Edit the kubelet service file /etc/systemd/system/kubelet.service.d/10-kubeadm.conf
on each worker node and set the below parameter in KUBELET_SYSTEM_PODS_ARGS variable.
--read-only-port=0
Based on your system, restart the kubelet service. For example:
systemctl daemon-reload
```

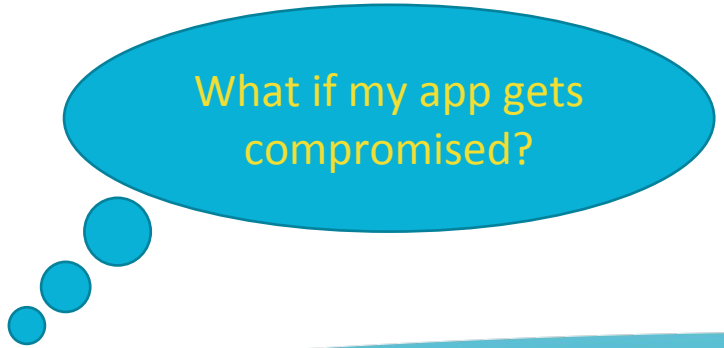
kube-hunter inside a pod

kube-hunter inside a pod



kube-hunter inside a pod

- Results depend on RBAC settings
  - and the service account you use for the pod



aqua

github.com/aquasecurity/kube-bench

github.com/aquasecurity/kube-hunter

@lizrice | @aquasecteam