Kubernetes Security Workshop

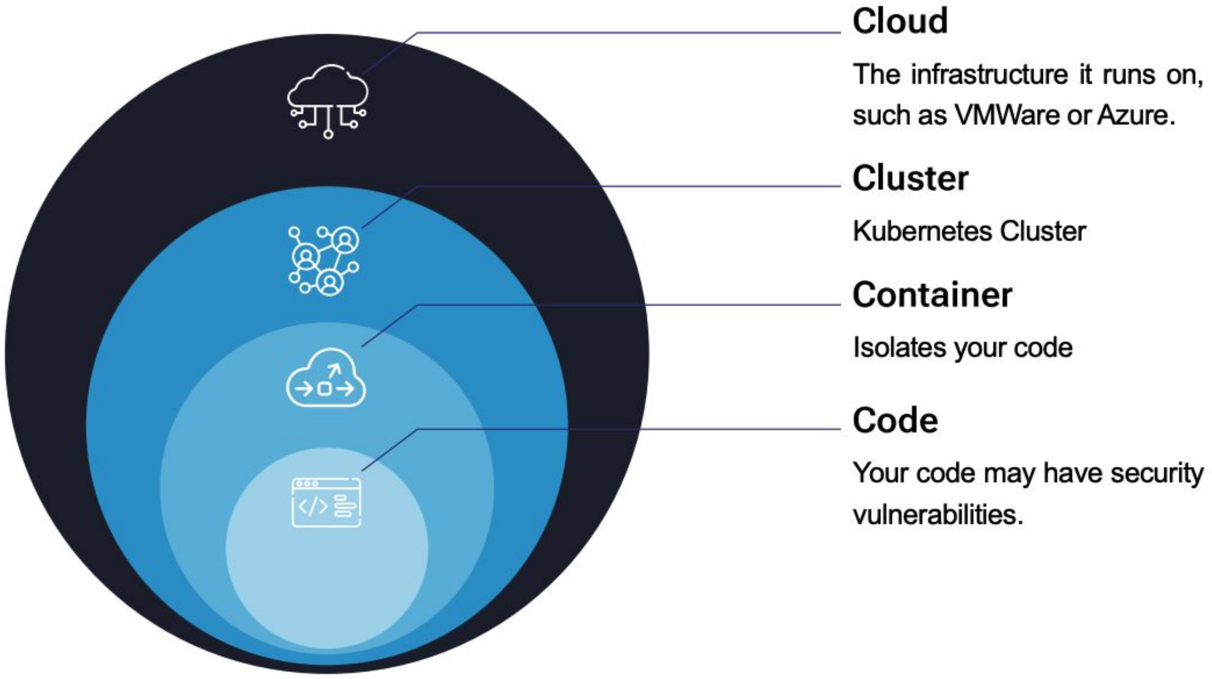
Jacob Beasley

About Your Lecturer

Jacob Beasley

- Kubernetes user since 2016
- Experience building and supporting software in just about every tech stack under the sun
- Certified Kubernetes Security Specialist and Certified Kubernetes Administrator
- Leads team behind open-source kubernetes security platform m9sweeper
- Leads a team of SREs that deploy and manage applications on Kubernetes for 20+ businesses.

4 C’s of Cloud Security



Cloud

The infrastructure it runs on, such as VMWare or Azure.

Cluster

Kubernetes Cluster

Container

Isolates your code

Code

Your code may have security vulnerabilities.

[View Lab Guide]

Lab Summary - Layers Covered

Cloud - Use VPN / Firewalls to Limit Access to Kubernetes API

Cluster - Kube-Bench, Kube-Hunter

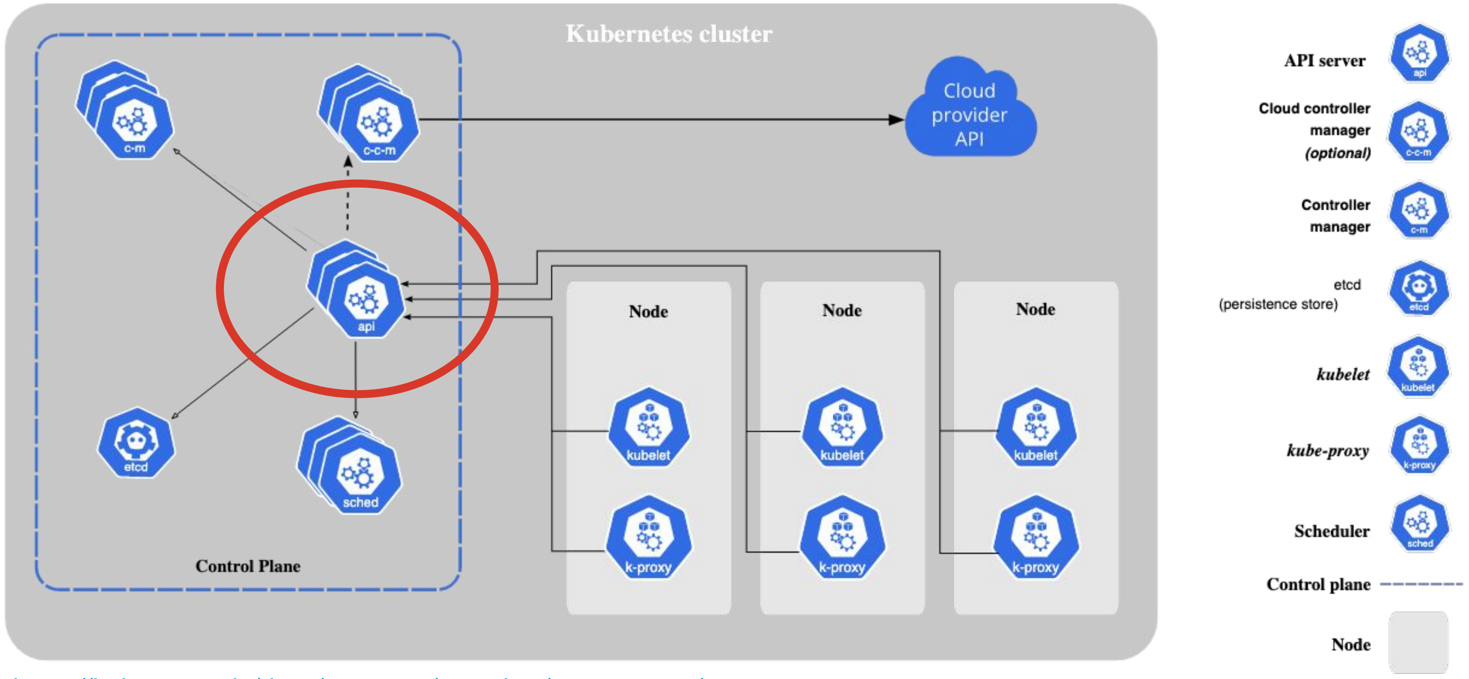
Container - OPA Gatekeeper, Kubesec

Code - Trivy, Project Falco

View Lab Guide

Module 1:
Cloud

Kubernetes Architecture



https://kubernetes.io/docs/concepts/overview/components/

Kubernetes Security Best Practices

There are a few best practices that the tools we cover cannot check:

1. **The kubernetes API should not be exposed on the internet**. Ideally it is behind a VPN or some other firewalling, and only encrypted traffic is allowed to the kubernetes API.
2. **Do not make everyone an administrator.** You should limit individual users' access using Role-Based Access Control.
3. **Access to the etcd datastore** should be strictly limited and configured to use TLS. It should also be encrypted to prevent tampering with or extracting of data.

View Lab Guide

Module 1:
Cluster

Tools to Secure Clusters

- Role Based Access Control
- Kube-Bench
- Kube-Hunter

View Lab Guide

Role Based Access Control

Cluster Roles / Roles: Define what a role can do.

Cluster Role Bindings / Role Bindings: Bind a particular user to a particular role.

Users do not actually exist in Kubernetes, but Kubernetes' role binding can refer to them. Kubernetes expects an external identity provide to identify and authenticate users.

Service Accounts allow applications to communicate with the Kubernetes API, such as if an application was to automatically create and sign SSL certificates.

View Lab Guide

Sample Role

```yaml
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  namespace: default
  name: pod-reader
rules:
- apiGroups: [""] # "" indicates the core API group
  resources: ["pods"]
  verbs: ["get", "watch", "list"]
```

View Lab Guide

Sample Role Binding

```
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: read-pods
  namespace: default
subjects:
- kind: User
  name: jane # "name" is case sensitive
  apiGroup: rbac.authorization.k8s.io
roleRef:
  kind: Role #this must be Role or ClusterRole
  name: pod-reader
  apiGroup: rbac.authorization.k8s.io
```

View Lab Guide

Why use tools?

Kubernetes has many components with many settings, and knowing each and every one of their settings is unrealistic for most administrators. Instead, we use tooling to coach us towards best practices.

We will setup and look at 2 great open source tools today:

kube-bench: Compares your kubernetes' settings against established best practices

kube-hunter: Attempts a pentest, either passively scanning or actively exploiting

View Lab Guide

Demo: kube-bench

View Lab Guide

Demo: kube-hunter

View Lab Guide

Container

What is a virtualization?

Virtualization allows an operating system to virtualize another operating system inside of it. A hypervisor or other such technology is used to coordinate and schedule the different virtual machines, allowing each to share a certain segment of the host's hardware as well virtualizing things like networks or file shares in an efficient way.

View Lab Guide

What is a container?

A container offers an alternative to virtualization. It runs as a process in a host machine and uses various linux kernel features to isolate the process:

cgroup - Can limit the amount of CPU or RAM a process has access to.

chroot - Changes the root directory from the actual linux root directory to a subdirectory with all of your containers' files unzipped into it.

namespacing - Uses various kinds of namespaces to ensure users, processes, networks, volume mounts, etc are isolated to just the container's own process as well as any child processes it starts.

View Lab Guide

Degrees of Isolation

Different approaches provide different degrees of isolation. In high-security settings, sometimes virtualization can be considered more secure.

Hardware Separation > Virtualization > Containerization > No Isolation

Kata containers can be used in kubernetes to run all pods as separate virtual machines (if additional security is desired). Tools like GVisor can block container escapes and other issues even if kubernetes is misconfigured.

View Lab Guide

Parts of a Container Image

Layered File System: Each command in your Dockerfile creates another tar file of of the files the command changed. When running a container image, these layers are untarred in the same order, resulting in the same file system every time.

CMD Command: The default command arguments to run upon booting up the container image.

PWD: Present working directory with which to run the CMD command.

ENV Variables: Any operating system environment variables to set, such as the default PATH to search for executable files.

Group and **User** with which to run the process.

View Lab Guide

Container Breakout

If an application is given escalated privileges, the application could break out of its container and execute commands as if it were a user in the host operating system.

For example, a simple way to do this is to run as root and execute linux kernel commands using a perl script to change your root directory.

For more details, see https://pentestmonkey.net/blog/chroot-breakout-perl

View Lab Guide

Preventing Container Breakout

1. Do not allow applications to run as root or escalate their privileges. Using a kubernetes security context can prevent against this.

```
apiVersion: v1
kind: Pod
metadata:
  name: security-context-demo
spec:
  securityContext:
    runAsUser: 1000
    runAsGroup: 3000
    fsGroup: 2000
  containers:
  - name: nginx
    image: nginx
    securityContext:
      allowPrivilegeEscalation: false
      Privileged: false
      runAsNonRoot: true
```

View Lab Guide

Preventing Container Breakout

2. Do not ever mount the host’s root folder as a host volume. If you do this, someone running as non-root might get access to the entire host filesystem, which would provide them with a plethora of options for escalating privileges.

3. Use tools like OPA, Gatekeeper, pod security policies/pod security standards (covered in module 4) to prevent someone from deploying pods with host volumes or with privileges

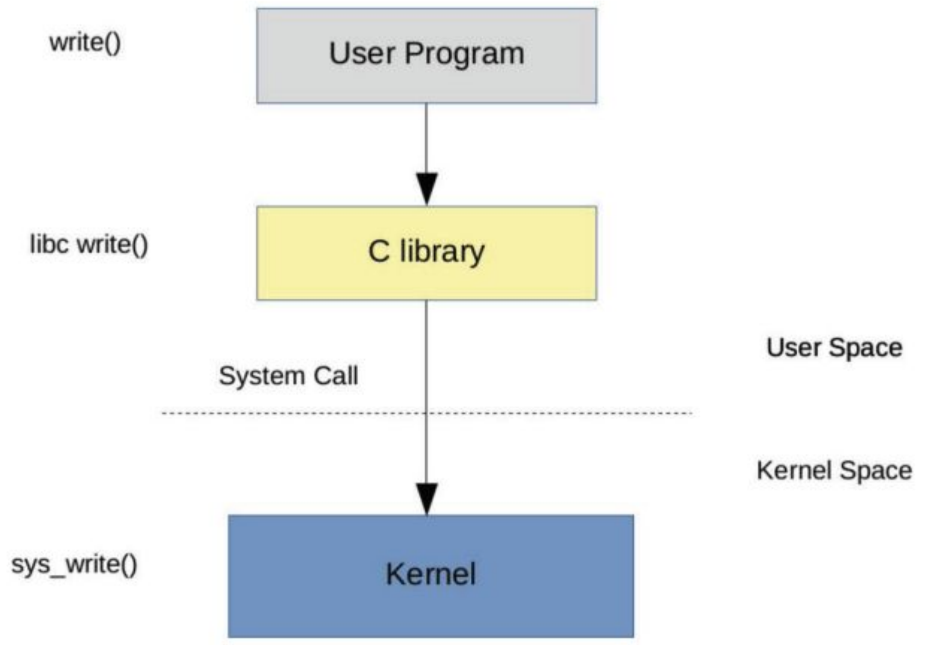
4. Limit service account’s privileges so that users cannot bypass the policies you set.

View Lab Guide

Limiting Linux Kernal Calls

You should limit which calls that applications can make to the linux kernel.

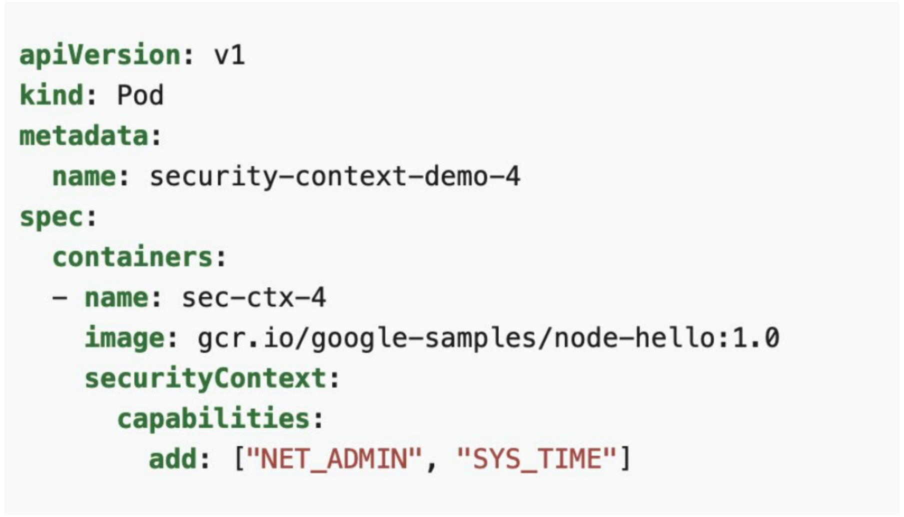
SecComp and **AppArmor** are tools that allow you to build policies and then enforce those across a number of namespaces or pods.



View Lab Guide

Limiting Linux Kernal Calls / SecComp / AppArmor

You can also limit or add capabilities using the securityContext configuration block on a pod’s container.



```yaml
apiVersion: v1
kind: Pod
metadata:
  name: security-context-demo-4
spec:
  containers:
  - name: sec-ctx-4
    image: gcr.io/google-samples/node-hello:1.0
    securityContext:
      capabilities:
        add: ["NET_ADMIN", "SYS_TIME"]
```

View Lab Guide

kubesec

kubesec will analyze the manifest of a pod and provide advice. It checks for things such as:

1. Limiting kernel privileges
2. Preventing privilege escalation
3. Limiting access to the file system
4. Preventing running as root
5. Limiting CPU / RAM
6. Mounting Host Storage

View Lab Guide

Pod Security Admissions

Pod Security Standards defines 3 isolation levels:

Privileged: Unrestricted. Allows for privilege escalation.

Baseline: Minimally restricted. Prevents privilege escalations.

Restricted: Heavily restricted, following pod hardening best practices.

View Lab Guide

Pod Security Admissions

If you set the pod security admission labels on a namespace, kubernetes will enforce certain fields be set in the securityContext of all pods created in that namespace.

```
apiVersion: v1
kind: Namespace
metadata:
  name: default
  labels:
    pod-security.kubernetes.io/enforce: restricted
```

View Lab Guide

Lab: Pod Security Admission

1. Enable pod security admissions for default namespace.

```
kubectl edit namespace default

Add label:

pod-security.kubernetes.io/enforce: restricted
```

2. Remove and re-create your nginx pod. It should not be allowed to be created.
3. Clean up: Remove the label from your namespace so it does not affect future labs.

View Lab Guide

Network Policies

Kubernetes Network Policies allow you to limit network traffic to or from pods. By default, things are basically wide open, but with network policies you can specify as wide or narrow a policy as you need to.

Pod Selectors: Determines what pods this policy applies to.

Ingress: Restricts traffic from other applications **into** your pod.

Egress: Restricts traffic from your pod **out to** other applications.

View Lab Guide

Sample Network Policy: Deny All Ingress Traffic

```
---
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: default-deny-ingress
spec:
  podSelector: {}
  policyTypes:
  - Ingress
```

View Lab Guide

Sample Network Policy: Allow All Ingress Traffic

```
---
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: allow-all-ingress
spec:
  podSelector: {}
  ingress:
  - {}
  policyTypes:
  - Ingress
```

View Lab Guide

Sample Network Policy: Deny All Egress Traffic

```yaml
---
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: default-deny-egress
spec:
  podSelector: {}
  policyTypes:
  - Egress
```

View Lab Guide

Sample Network Policy: Allow All Egress Traffic

```
---
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: allow-all-egress
spec:
  podSelector: {}
  egress:
  - {}
  policyTypes:
  - Egress
```

View Lab Guide

Shortcomings of Built-in Features

1. Limits and rules are, for the most part, namespace-wide.
2. Not all conceivable rules can be setup.
3. Pod security standards are far less granular and may not be appropriate for all workloads.

View Lab Guide

Extending Kubernetes: OPA and Gatekeeper

- OPA and Gatekeeper allow you to write scripts to be executed every time any kubernetes object is created or updated.
- These scripts can validate the object meets policy standards, or even mutate the object.

View Lab Guide

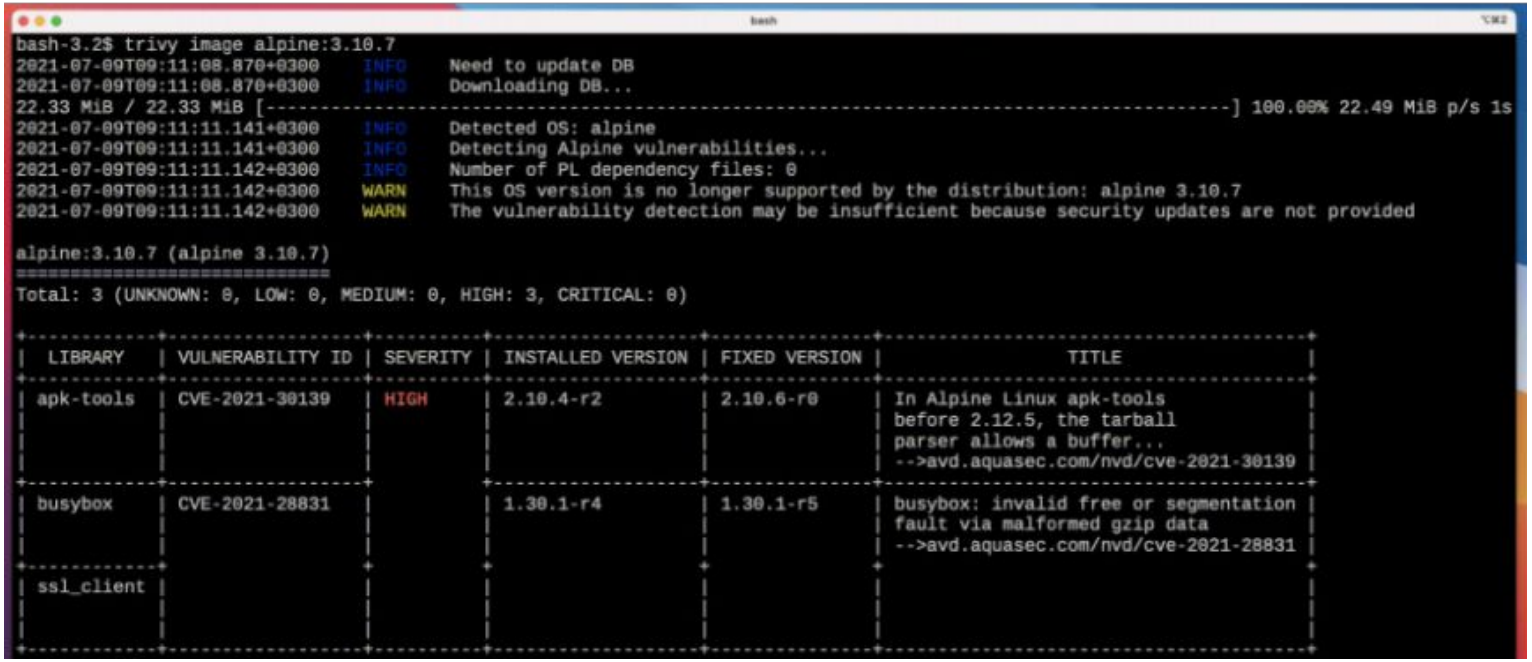
Lab: Gatekeeper

1. Install Gatekeeper Constraints to require containers to have images using M9sweeper
2. Attempt to re-deploy your nginx pod (delete and re-create) and see how it will now be blocked

View Lab Guide

Code

CVE Scanning with Trivy



```
bash-3.2$ trivy image alpine:3.10.7
2021-07-09T09:11:08.870+0300    INFO    Need to update DB
2021-07-09T09:11:08.870+0300    INFO    Downloading DB...
22.33 MiB / 22.33 MiB [-----------------------------------------------------------------------------] 100.00% 22.49 MiB p/s 1s
2021-07-09T09:11:11.141+0300    INFO    Detected OS: alpine
2021-07-09T09:11:11.141+0300    INFO    Detecting Alpine vulnerabilities...
2021-07-09T09:11:11.142+0300    INFO    Number of PL dependency files: 0
2021-07-09T09:11:11.142+0300    WARN    This OS version is no longer supported by the distribution: alpine 3.10.7
2021-07-09T09:11:11.142+0300    WARN    The vulnerability detection may be insufficient because security updates are not provided

alpine:3.10.7 (alpine 3.10.7)
=============================
Total: 3 (UNKNOWN: 0, LOW: 0, MEDIUM: 0, HIGH: 3, CRITICAL: 0)

+------------+------------------+----------+-------------------+---------------+--------------------------------+
|  LIBRARY   | VULNERABILITY ID | SEVERITY | INSTALLED VERSION | FIXED VERSION |             TITLE              |
+------------+------------------+----------+-------------------+---------------+--------------------------------+
| apk-tools  | CVE-2021-30139   | HIGH     | 2.10.4-r2         | 2.10.6-r0     | In Alpine Linux apk-tools      |
|            |                  |          |                   |               | before 2.12.5, the tarball     |
|            |                  |          |                   |               | parser allows a buffer...      |
|            |                  |          |                   |               | -->avd.aquasec.com/nvd/cve-2021-30139 |
+------------+------------------+          +-------------------+---------------+--------------------------------+
| busybox    | CVE-2021-28831   |          | 1.30.1-r4         | 1.30.1-r5     | busybox: invalid free or segmentation |
|            |                  |          |                   |               | fault via malformed gzip data  |
|            |                  |          |                   |               | -->avd.aquasec.com/nvd/cve-2021-28831 |
+------------+                  +          +                   +               +                                +
| ssl_client |                  |          |                   |               |                                |
|            |                  |          |                   |               |                                |
+------------+------------------+----------+-------------------+---------------+--------------------------------+
```

[View Lab Guide]

Demo

CVE Scanning in M9sweeper

View Lab Guide

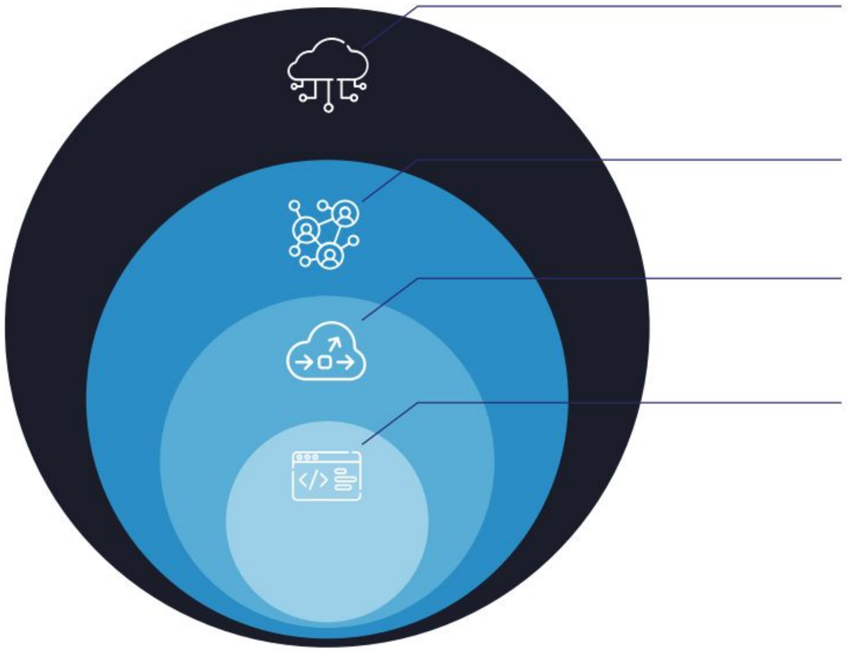
Demo: Project Falco

Project Falco: Runs and listens to alert you to applications doing suspicious things, such as opening command shells or escalating privileges

View Lab Guide

Summary

4 C’s of Cloud Security



Cloud

The infrastructure it runs on, such as VMWare or Azure.

Cluster

Kubernetes Cluster

Container

Isolates your code

Code

Your code may have security vulnerabilities.

View Lab Guide

Layers Covered

Cloud - Use VPN / Firewalls to Limit Access to Kubernetes API

Cluster - Kube-Bench, Kube-Hunter, RBAC

Container - OPA Gatekeeper, Kubesec

Code - Trivy, Project Falco

View Lab Guide

Questions?