How Splunk gives actionable relief to torture testing Kubernetes across multi-cloud

Rod Soto / Jose Hernandez
Splunk

Rod Soto

Principal, Threat Research

splunk> .conf20

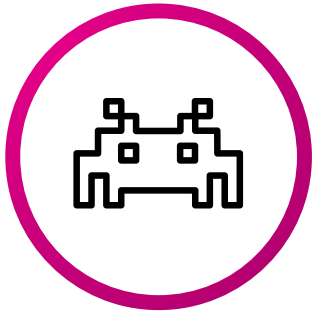
Jose Hernadez

Senior Manager, Threat Research

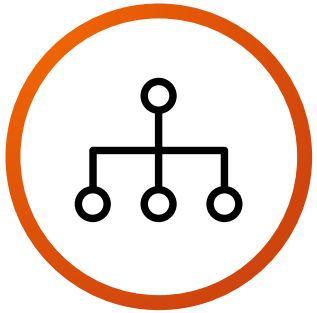
splunk> .conf20

Splunk Threat Research Team

Study Threats

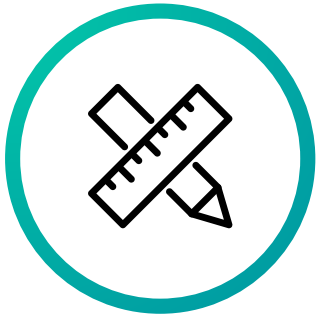


Create Datasets



Build Detections

Release Tools

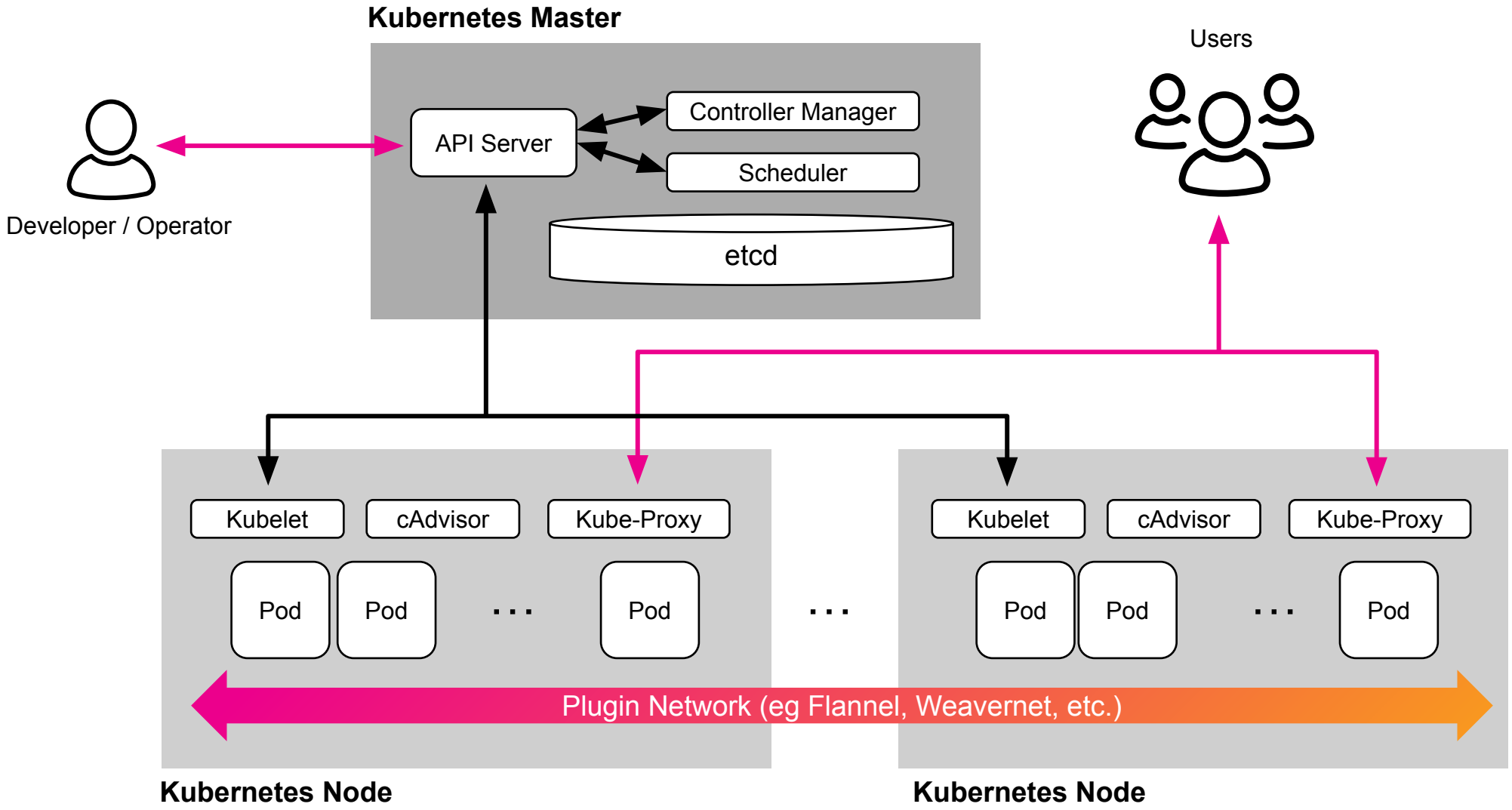


Share with Community

Agenda

1) Kubernetes
2) Architecture & Components
3) Attack surface of a K8s cluster
4) Tools for torture testing
5) Results

Kubernetes



Important Kubernetes components

Pod: A Pod is the basic execution unit of a Kubernetes application–the smallest and simplest unit in the Kubernetes object model that you create or deploy. A Pod represents processes running on your cluster*

Service: An abstract way to expose an application running on a set of Pods as a network service

Volume: is just a directory, possibly with some data in it, which is accessible to the Containers in a Pod.

Namespace: “Intended for use in environments with many users spread across multiple teams, or projects.*

Sensitive objects in a Kubernetes cluster

Configmaps: Includes things such as bind configuration files, command line arguments, environment variables, port numbers and other system components are runtime. This object is used for configuration replication and reference and should be stored in a central and protected place. Think for example of a hardcoded API key that is used by multiple applications distributed in containers.

Secrets: Stores and manages sensitive information, such as passwords, Oauth tokens and ssh keys. Unauthorized access to these files may result in cluster compromise on a multiple scale.

Accessing Kubernetes

Via API with Kubectl or REST requests

These requests go through several stages of authentication, authorization and Admission control.

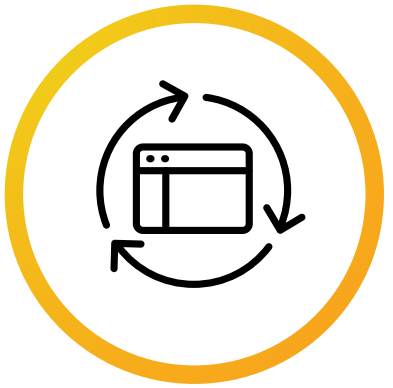
Authentication

Client Certificates, Password, Plain Tokens, Bootstrap tokens and JWT Tokens.

Authorization

ABAC, RBAC or Webhooks (Depends on provider setup)

Admission Control



Software modules that can modify or reject requests

Accessing Kubernetes - Network

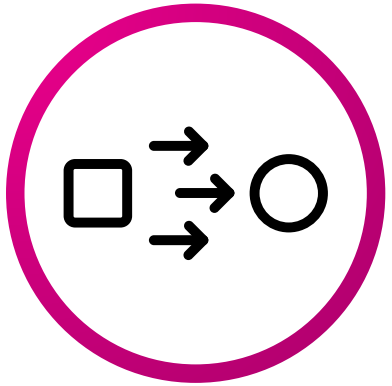
A series of ports used for Kubernetes access and functionality (TCP)

- 443 API
- 2379 etcd
- 6443 kube-apiserver
- 6666 etcd
- 8443 kube-apiserver
- 8080 kube-apiserver
- 10250 kubelet
- 10255 kubelet
- 10256 kube-proxy

Accessing Kubernetes

Special consideration must be given to the following objects of a cluster due to their sensitive nature as well.

Kubelet

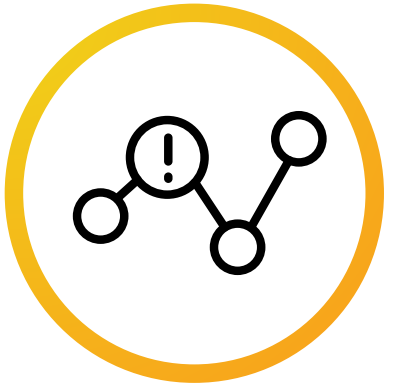


Agent that runs in all pods, may disclose sensitive information, used to execute commands, and lateral movement.

Etcd

Cluster state and configuration.

API



Unauthenticated requests may lead to command execution and cluster compromise.

Kubernetes attack surface

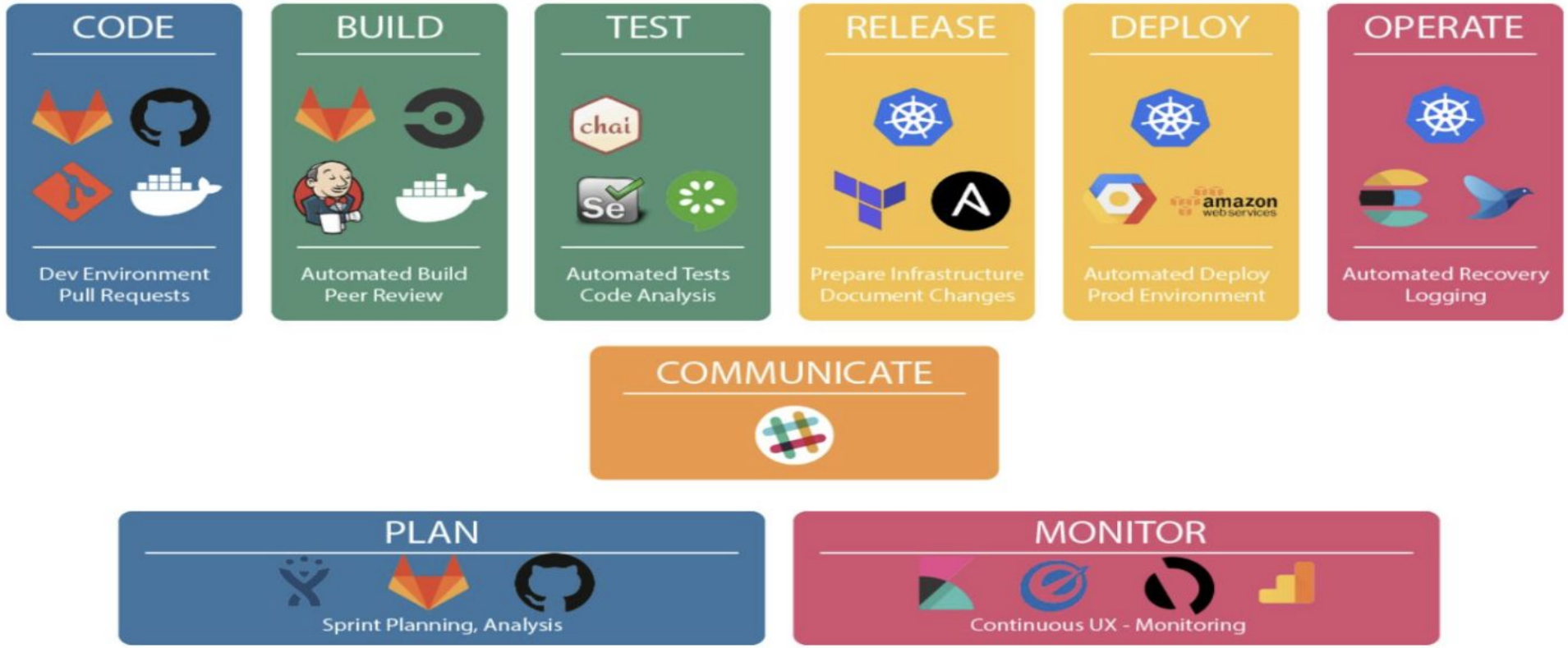
Inside Cluster

Compromised API Keys
Application vulnerabilities
K8s Platform vulnerabilities
Container Implantation
Container Escape
Running Cluster with high privilege account (root)
Privilege Abuse

Outside Cluster

Exposed application vulnerabilities
Exposed etcd
Exposed kubelet
Exposed management interface
Denial of Service
Exposed management GUI

DevOps attack surface (K8s periphery)



DevOps Pipeline Tool Overview (selection of tools, note: image/logo rights are with the respective copyright owners)

Microsoft Threat Matrix for K8s

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Impact
Using Cloud Credentials	Exec into Container	Backdoor Container	Privileged Container	Clear Container Logs	List K8S Secrets	Access the K8S API Server	Access Cloud Resources	Data Destruction
Compromised Images in Registry	bash/cmd inside container	Writable hostPath mount	Cluster-admin binding	Delete K8S events	Mount service principal	Access Kubelet API	Container service account	Resource Hijacking
Kube config file	New container	Kubernetes CronJob	hostPath mount	Pod/container name similarity	Access container service account	Network mapping	Cluster internal networking	Denial of service
Application Vulnerability	Application exploit (RCE)		Access cloud resources	Connect from Proxy server	Applications credentials in configuration files	Access Kubernetes dashboard	Applications credentials in configuration files	
Exposed Dashboard	SSH server running inside container					Instance Metadata API	Writable volume mounts on the host	
							Access Kubernetes dashboard	
							Access tiller endpoint	

Tools + Matrix, Where Do They Fit?

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Impact
Using Cloud Credentials (Git-Wild-hunt)	Exec into container (Kubectl, Kubeadmin, Docker)	Backdoor Container (CCAT)	Privileged Container	Clear Container Logs	List K8S secrets	Access the K8S API server	Access Cloud Resources	Data Destruction
Compromised Images in Registry (CCAT)	New Container (CCAT)	Writable hostPath mount (Kubecti, Docker)	Cluster-admin binding	Delete K8S events	Mount service principal	Access Kubelet API	Container service account	Resource Hijacking
Kube config file (Kube_hunter)	Application Exploit (MetaSploit)	Kubernetes CronJob (CCAT, bash)	hostPath mount	Pod/container name similarity	Access container service account	Network mapping	Cluster internal networking	Denial of service
Application Vulnerability (Trivy_Kube_ hunter)	SSH Server running inside container (THC Hydra)		Access cloud resources	Connect from Proxy server	Applications credentials in configuration files	Access Kubernetes dashboard	Applications credentials in configuration files	
Exposed Dashboard (Shodan, Kube_hunter)						Instance Metadata API	Writable volume mounts on the host	
							Access Kubernetes dashboard	
							Access tiller endpoint	

Tools + Matrix, Where Do They Fit?

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Impact
Using Cloud Credentials (Git-Wild-hunt)	Exec into container (Kubectl, Kubeadmin, Docker)	Backdoor Container (CCAT)	Privileged Container (KubiScan)	Clear Container Logs (Kubectl, Kube_hunter)	List K8S secrets (kube_hunter)	Access the K8S API server	Access Cloud Resources	Data Destruction
Compromised Images in Registry (CCAT)	New Container (CCAT)	Writable hostPath mount (Kubectl, Docker)	Cluster-Admin Binding (Kube-Audit, KubiScan)	Delete K8S events (Kubectl, Docker)	Mount service principal (kubectl)	Access Kubelet API	Container service account	Resource Hijacking
Kube config file (Kube_hunter)	Application Exploit (MestaSploit)	Kubernetes CronJob (CCAT, bash)	hostPath mount (Kube_hunter)	Pod/container name similarity (kubectl)	Access container service account (kubectl)	Network mapping	Cluster internal networking	Denial of service
Application Vulnerability (Trivy_Kube_ hunter)	SSH Server running inside container (THC Hydra)		Access Cloud Resources (Kube_hunter)	Connect from Proxy server (Kube_hunter)	Applications credentials in configuration files (kubectl)	Access Kubernetes dashboard	Applications credentials in configuration files	
Exposed Dashboard (Shodan, Kube_hunter)						Instance Metadata API	Writable volume mounts on the host	
							Access Kubernetes dashboard	
							Access tiller endpoint	

Tools + Matrix, Where Do They Fit?

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Impact
Using Cloud Credentials (Git-Wild-hunt)	Exec into container (Kubectl, Kubeadmin, Docker)	Backdoor Container (CCAT)	Privileged Container (KubiScan)	Clear Container Logs (Kubectl, Kube_hunter)	List K8S secrets (kube_hunter)	Access the K8S API server (kube_hunter)	Access Cloud Resources (kube_hunter)	Data Destruction (kubectl)
Compromised Images in Registry (CCAT)	New Container (CCAT)	Writable hostPath mount (Kubectl, Docker)	Cluster-Admin Binding (Kube-Audit, KubiScan)	Delete K8S events (Kubectl, Docker)	Mount service principal (kubectl)	Access Kubelet API (kube_hunter)	Container service account (kubectl)	Resource Hijacking (kubectl)
Kube config file (Kube_hunter)	Application Exploit (MestaSploit)	Kubernetes CronJob (CCAT, bash)	hostPath mount (Kube_hunter)	Pod/container name similarity (kubectl)	Access container service account (kubectl)	Network mapping (kube_hunter)	Cluster internal networking (kube_hunter)	Denial of service CVE-2019-11253
Application Vulnerability (Trivy_Kube_ hunter)	SSH Server running inside container (THC Hydra)		Access Cloud Resources (Kube_hunter)	Connect from Proxy server (Kube_hunter)	Applications credentials in configuration files (kubectl)	Access Kubernetes dashboard (kube_hunter)	Applications credentials in configuration files (kubectl)	
Exposed Dashboard (Shodan, Kube_hunter)						Instance Metadata API (Pacu)	Writable hostPath mount (Kubectl, Kube_hunter)	
							Access Kubernetes dashboard (kube_hunter)	
							Access tiller endpoint (kubectl)	

Torture Tools ✂️👍🍷

Starting List

Trivy
Skan
CCAT
Kubeaudit
FairwindsOps Polaris
Kubesec (controlplaneio)
Kubiscan
Kube-bench
Kube-scan
Kube-hunter
Kubei

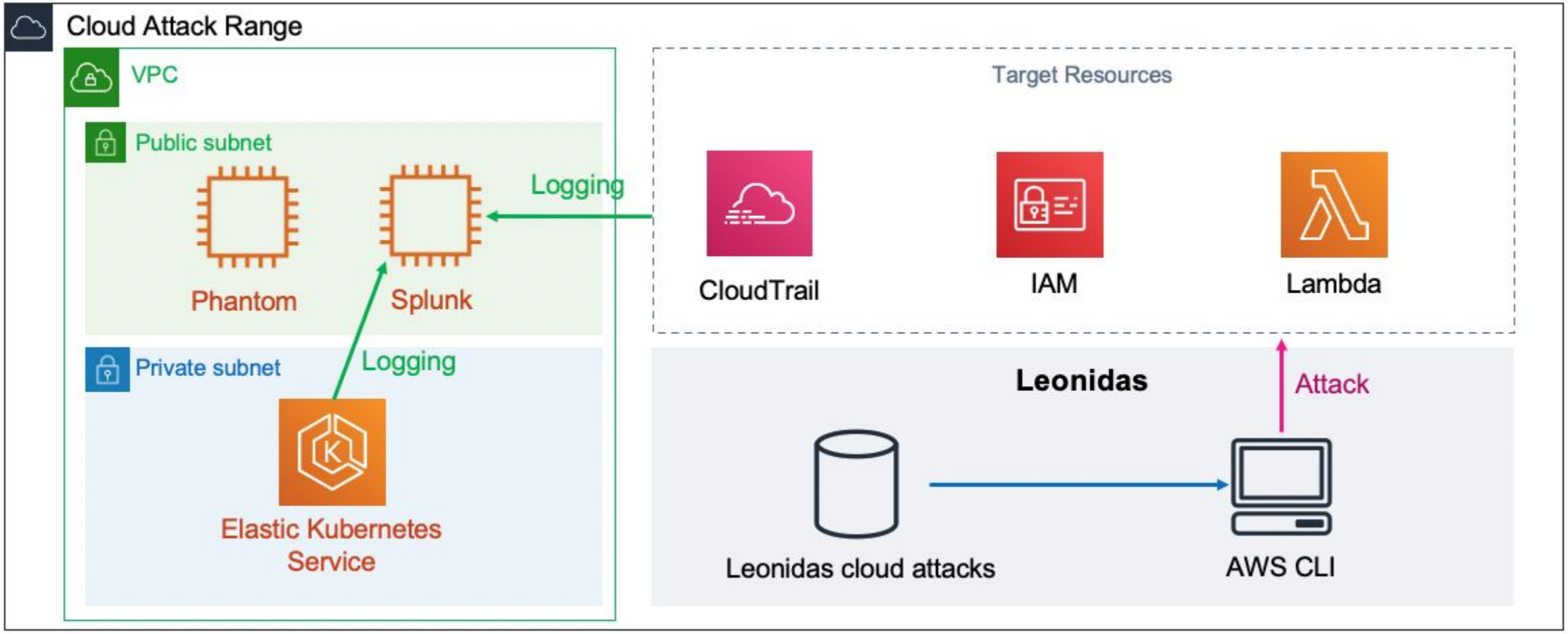
Run as pod

❌Trivy
❌Skan
❌CCAT
❌Kubeaudit
FairwindsOps Polaris
Kubesec (controlplaneio)
Kubiscan
Kube-bench
Kube-scan
Kube-hunter
Kubei

Checks for exploits 💉

❌Trivy - check manifests
❌Skan - checks manifests
❌CCAT - implantation
❌Kubeaudit - auditing
❌FairwindsOps Polaris - health/auditing
❌Kubesec (controlplaneio) - checks manifests
❌Kubiscan - auditing
❌Kube-bench - checks manifests
❌Kube-scan - auditing
✔️Kube-hunter - vulnerability scanner
✔️Kubei - vulnerability scanner

Test Platform



https://github.com/splunk/attack_range_cloud/

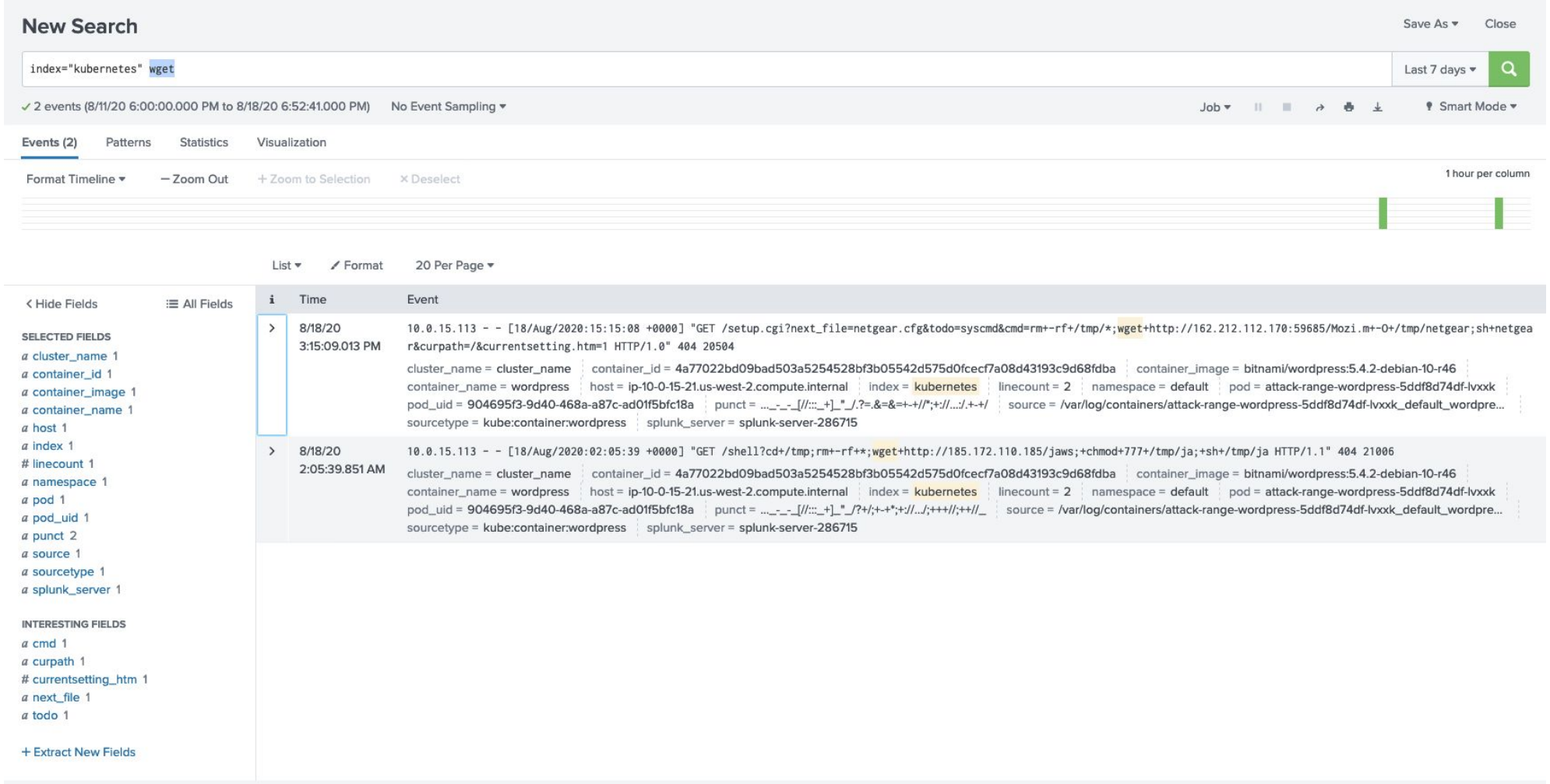
VIDEO PLACEHOLDER

Presenters:

DO NOT embed videos in your slides. Use this slide as a placeholder to cue the production team to stitch your separate MP4 file into your recorded presentation. This slide will not be shown in your final session. Please place the filename of your video below.

Video Filename:

Interesting Side effect 



splunk> .conf20

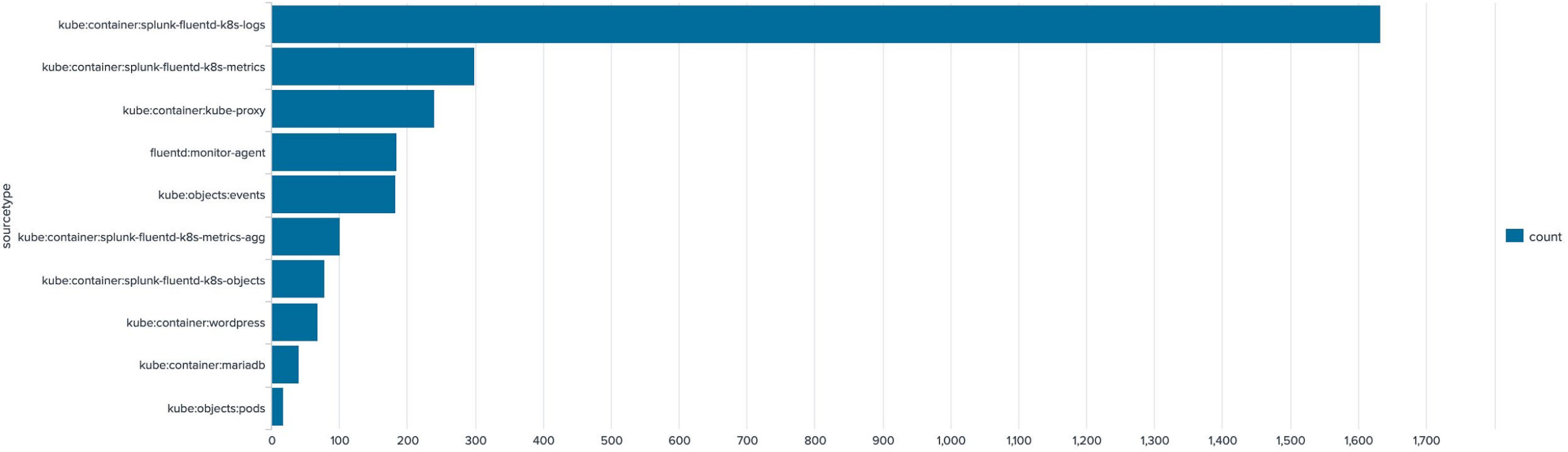
Key Takeaways

- Few k8s security tools check running cluster
- Kube hunter provided the best data
- Many tools deploy their own UI
- Not easy to get JSON out of most tools for reporting
- Most tools are only auditing
- While we did not test x-clouds this would work in GCP and Azure

Where is the data? 💻

all the logs we collected during the attacks

github.com/d1vious/



Thank You

Please provide feedback via the

SESSION SURVEY