MAY 2022

KUBERNETES

RAJUARAVIND S

Kubernetes

Kubernetes is an open-source container orchestration system for automating software deployment, scaling, and management. Google originally designed Kubernetes, but the Cloud Native Computing Foundation now maintains the project.

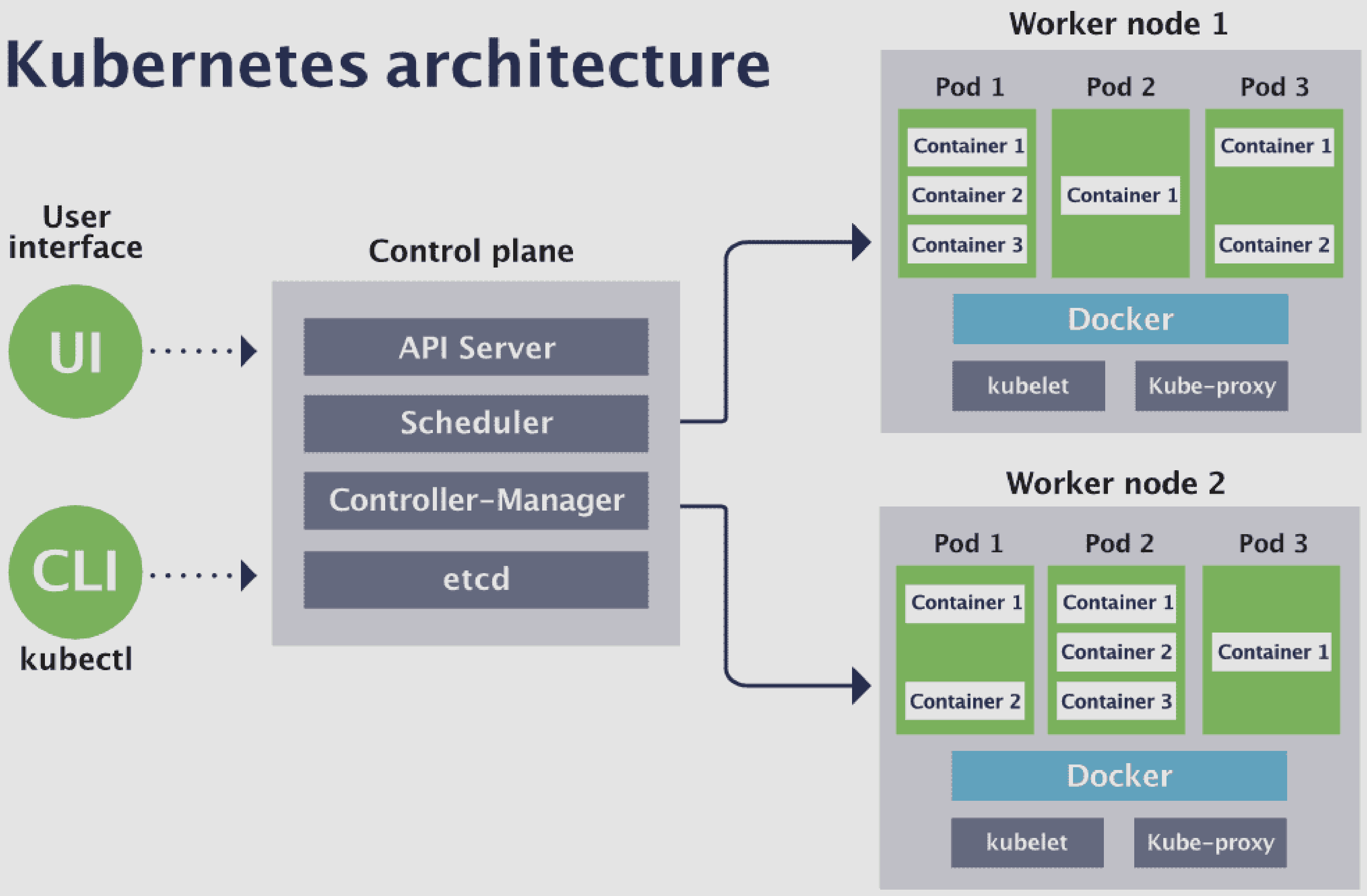
Kubernetes, often abbreviated as “K8s”, orchestrates containerized applications to run on a cluster of hosts. The K8s system automates the deployment and management of cloud native applications using on-premises infrastructure or public cloud platforms.

Why Kubernetes

- Reducing development and release timeframes.
- Optimizing IT costs.
- Increased software scalability and availability.
- Flexibility in multi-cloud environments.
- Cloud migration paths.

ARCHITECTURE

Kubernetes itself follows a client-server architecture, with a master node composed of etcd cluster, kube-apiserver, kube-controller-manager, cloud-controller-manager, scheduler. Client (worker) nodes are composed of kube-proxy and kubelet components.
A working Kubernetes deployment is called a cluster. You can visualize a Kubernetes cluster as two parts: the control plane and the compute machines, or nodes. Each node is its own Linux® environment, and could be either a physical or virtual machine. Each node runs pods, which are made up of containers



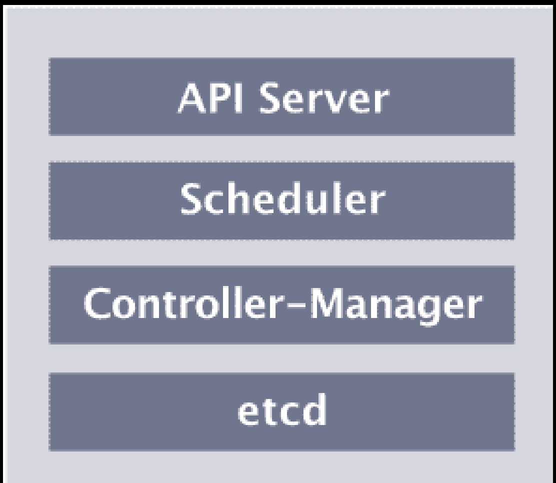
Advantage of K8s

-Self Healing
-Automated Rollbacks
-Auto scaling
-Load Balancing

ARCHITECTURE IN DETAIL

kubernetes Master has several components - API server - Scheduler - Controller_manger and etcl. Each components plays a major role in K8s Architecture. Kuber Master Controls/Manager the worker nodes in the architecture.

API server ---> Front-end for kubernetes
ETCD ----> Highly available key value store
Kube-Scheduler ---> Watches newly created Pods
Controller Manager ---> Runs Controller Porcess



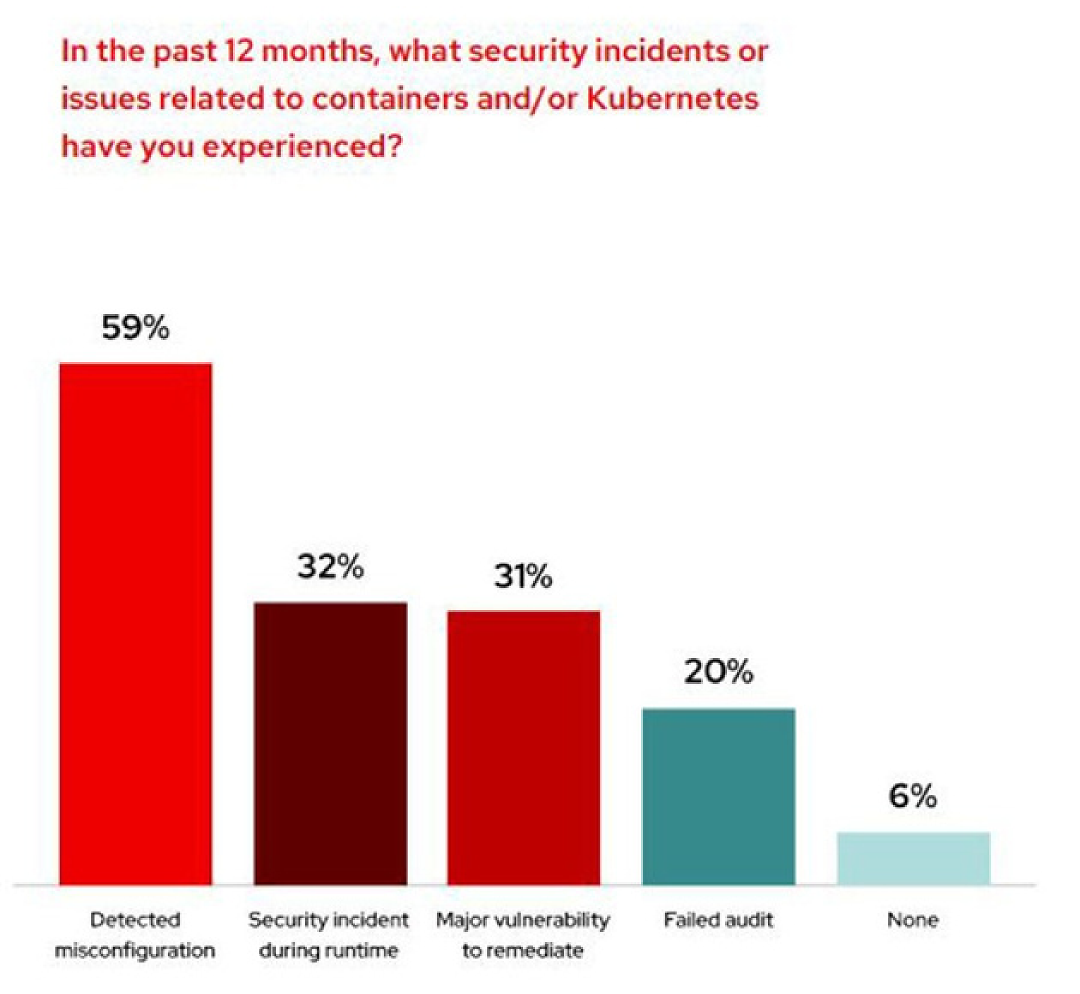
API Server

Scheduler

Controller-Manager

etcd

From my research on K8s have found that "Each Pods Communicates With any other Pod in a Cluster " . **So Monitoring Pod logs will be very Efficient in order of securing Kubernetes**



Misconfigured Kubernetes UIs were a favorite target of attackers, according to Aqua's report

Kubernetes security issues:
An examination of major attacks

Case #1: Misconfigured Docker

If you're an attacker looking for misconfigured Docker instances to exploit, it's as easy as probing open ports 2375, 2376, 2377, 4243, and 4244 on the internet. Vulnerable instances can also be located using search engines like Shodan. Organized threat actors like TeamTNT have been observed using legitimate Kubernetes monitoring tools like Weave Scope to backdoor these Docker instances.

Case #2: Malicious Docker images

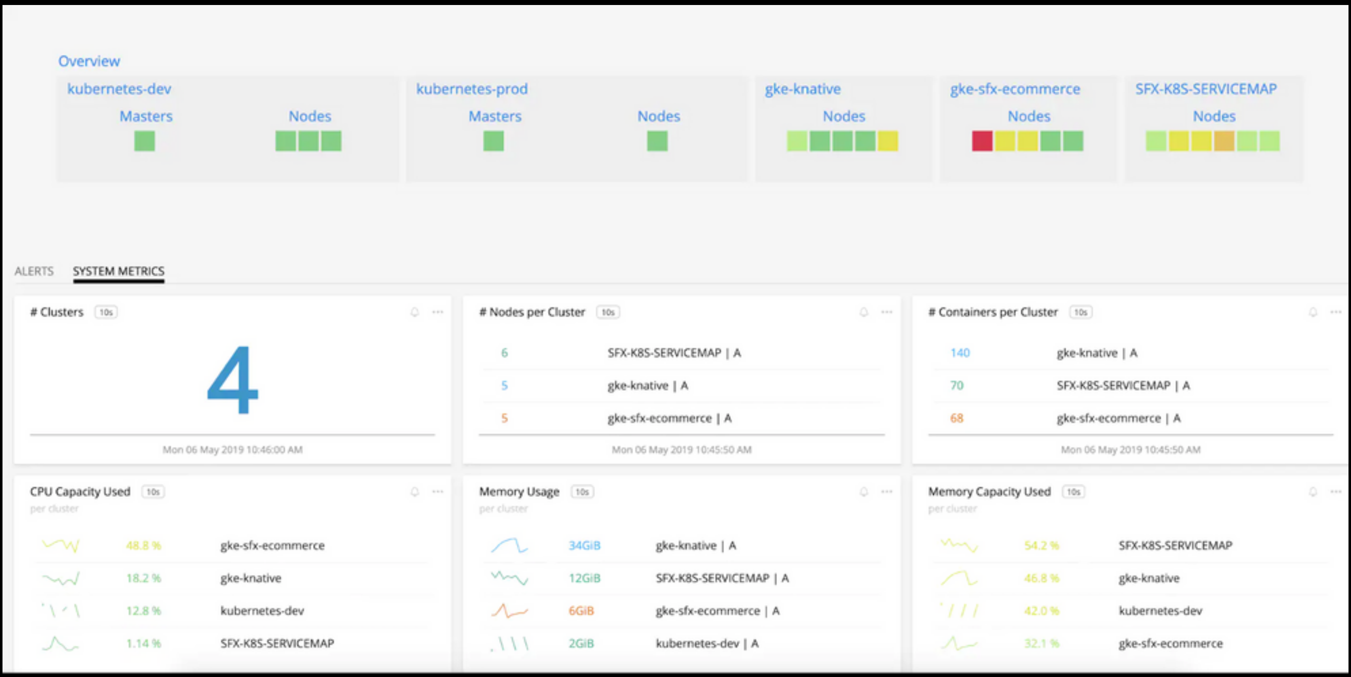
Public image repositories have become an easy tool for distributing malicious images disguised as custom configurations. An example that illustrates the reach of this attack vector is an incident with Docker hub, where a malicious image was pulled 5 million times before being removed. This underscores the need to check your base image sources, regardless of whether you are creating your own, or using images from a public repository. Forensic analysis tools like dive (as well as Docker itself) can help you check image history.

Case #3: Unintentional cluster misconfiguration

A considerable amount of trust is placed in cloud providers to configure, patch, and manage Kubernetes security on our behalf. Sometimes, however, things can fall through the cracks. Even the most highly trusted cloud services can be susceptible to configurations that unwittingly enable privilege escalation and facilitate the takeover of the cluster.

Splunk Kubernetes monitoring

The Kubernetes Overview provides a heatmap of every Kubernetes cluster you are monitoring in Splunk Infrastructure Monitoring, as well as a dashboard showing the number of clusters, nodes and containers for each, and aggregated system metrics (CPU, disk, memory, network) across all of these nodes.



Clusters 10s

4

Mon 06 May 2019 10:46:00 AM

Nodes per Cluster 10s

6	SFX-K8S-SERVICEMAP | A
5	gke-knative | A
5	gke-sfx-ecommerce | A

Mon 06 May 2019 10:45:50 AM

Containers per Cluster 10s

140	gke-knative | A
70	SFX-K8S-SERVICEMAP | A
68	gke-sfx-ecommerce | A

Mon 06 May 2019 10:45:50 AM

CPU Capacity Used 10s
per cluster

48.8 %	gke-sfx-ecommerce
18.2 %	gke-knative
12.8 %	kubernetes-dev
1.14 %	SFX-K8S-SERVICEMAP

Memory Usage 10s
per cluster

34GiB	gke-knative | A
12GiB	SFX-K8S-SERVICEMAP | A
6GiB	gke-sfx-ecommerce | A
2GiB	kubernetes-dev | A

Memory Capacity Used 10s
per cluster

54.2 %	SFX-K8S-SERVICEMAP
46.8 %	gke-knative
42.0 %	kubernetes-dev
32.1 %	gke-sfx-ecommerce

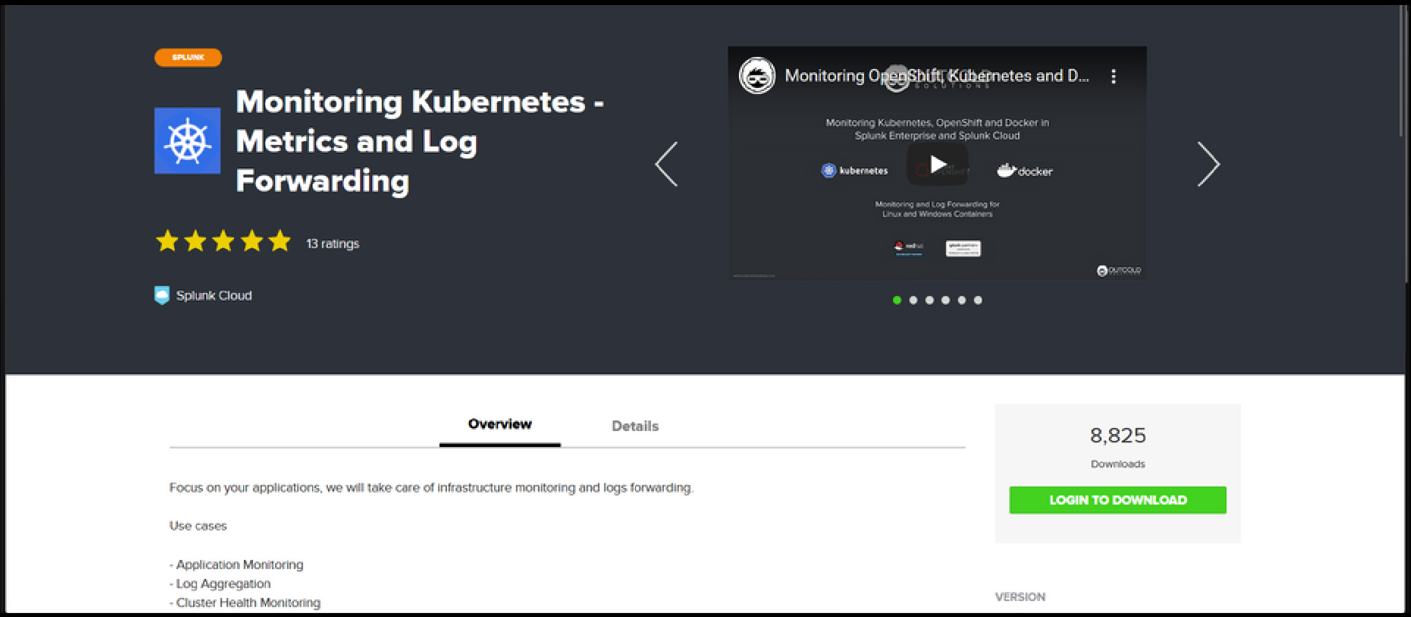
Use cases- General

- Nodes per Cluster
- Container Per Cluster
- Memory Used - Cluster
- Memory Used - Each Node
- Total Count of Kuber Master
- Total Count of Node
- Total Count of Container
- Active pods / Inactive Pods
- Top 10 Node Used
- Top 10 Container Used
- Top Nodes by memory
- Pod logs
- Top images used in a Container

Monitoring Kubernetes - Metrics and Log Forwarding

Its provide solutions for monitoring Kubernetes, OpenShift and Docker clusters in Splunk Enterprise and Splunk Cloud. With 10 minutes setup, you will get a monitoring solution, that includes log aggregation, performance and system metrics, metrics from the control plane and application metrics, a dashboard for reviewing network activity, and alerts to notify you about cluster or application performance issues.

Reference :https://splunkbase.splunk.com/app/3743/#/details



SPLUNK
Monitoring Kubernetes - Metrics and Log Forwarding
13 ratings
Splunk Cloud
Monitoring OpenShift, Kubernetes and D...
Monitoring Kubernetes, OpenShift and Docker in Splunk Enterprise and Splunk Cloud
kubernetes
docker
Monitoring and Log Forwarding for Linux and Windows Containers
Overview
Details
8,825
Downloads
LOGIN TO DOWNLOAD
VERSION

Focus on your applications, we will take care of infrastructure monitoring and logs forwarding.

Use cases

- Application Monitoring
- Log Aggregation
- Cluster Health Monitoring

Use cases

- Application Monitoring
- Log Aggregation
- Cluster Health Monitoring
- Security and Audit
- Reduce complexity and improve productivity

Detecting Kubernetes scanning activity

You can run many searches with Splunk software to detect Kubernetes scanning activity. Depending on what information you have available, you might find it useful to identify some or all of the following:

- Amazon EKS cluster scan detection
- Amazon EKS pod scan detection
- Azure Kubernetes Service scan fingerprinting
- Azure Kubernetes Service pod scan fingerprinting
- Google Kubernetes Engine cluster scan detection

Measuring impact and benefit is critical to assessing the value of detecting Kubernetes scanning activity. The following are example metrics that can be useful to monitor when implementing this use case:

- Less unauthenticated traffic to sensitive URLs: The provided detections provide an understanding of the HTTP API traffic your cluster is seeing that is unauthenticated
- Identified presence of scanning tools: Tools such as Zgrap or Nmap are usually clear indicators of suspicious activit

Required data

Kubernetes for Amazon EKS, Azure, or GCP