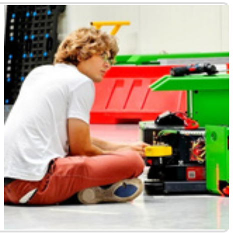
Kubernetes logging: Metrics server, containerd crio crictl usage

Check GitHub for helpful DevOps tools:

Michael Robotics

Hi, I'm Michal. I'm a Robotics Engineer and DevOps enthusiast. My mission is to create skill-learning platform that combats information overload by adhering to the set of principles: simplify, prioritize, and execute.

https://github.com/MichaelRobotics



Ask Personal AI Document assistant to learn interactively (FASTER)!

1 https://github.com/MichaelRobotics/DevOpsTools/blob/main/KubernetesMetrics.pdf

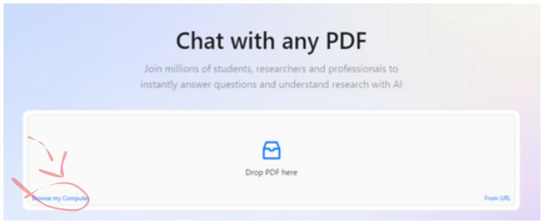
1 Download PDF

2 Go to website

Click there to go to ChatPdf website 2

3 Browse file

3



Chat with any PDF

Join millions of students, researchers and professionals to instantly answer questions and understand research with AI

4 Chat with Document

Ask questions about document! 4

Complety new to Linux and Networking?

Essential for this PDF is a thorough knowledge of networking. I highly recommend the HTB platform's networking module, which offers extensive information to help build a comprehensive understanding.

HTB - Your Cyber Performance Center

We provide a human-first platform creating and maintaining high performing cybersecurity individuals and organizations.

https://www.hackthebox.com/

What is Kubernetes?

Kubernetes is an open-source platform that automates the deployment, scaling, and management of containerized applications. It helps manage clusters of nodes running containers, ensuring efficient and reliable operation.

How Kubernetes clusters are made?

Kubernetes clusters consist of a control plane and multiple worker nodes. The control plane manages cluster operations, while worker nodes run the actual container workloads.

Why and When use Kubernetes

Kubernetes is ideal for deploying scalable, resilient, and automated containerized applications. It is used when managing multiple containers across different environments is necessary.

Example: Running a microservices-based e-commerce platform that scales up during peak hours.

System Requirements

- RAM: 2 GB per node (1 GB can work for testing but may lead to limited performance)
- 10 GB free storage
- Ubuntu

Kubernetes: Main components & packages

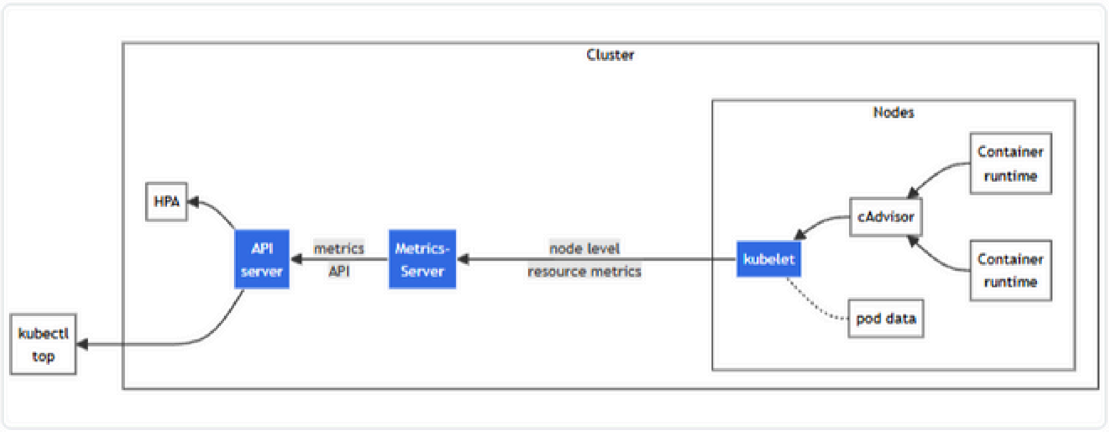
- **kube-apiserver:** Central management component that exposes the Kubernetes API; acts as the front-end for the cluster.
- **etcd:** Distributed key-value store for storing all cluster data, ensuring data consistency across nodes.
- **kube-scheduler:** Assigns pods to available nodes based on resource requirements and policies.
- **kube-controller-manager:** Manages core controllers that handle various functions like node status, replication, and endpoints.
- **kubelet:** Agent that runs on each node, responsible for managing pods and their containers.
- **kube-proxy:** Manages networking on each node, ensuring communication between pods and services within the cluster.

Kubernetes logging: Metrics server

1) What is Metrics Server?

The metrics-server collects resource metrics from kubelets and exposes them via the Metrics API in the Kubernetes API server, supporting tools like HPA, VPA, and kubectl top.
It queries nodes over HTTP for metrics and maintains a cached view of pod metadata and health, accessible via its extension API.

2) Resource metrics pipeline



- **cAdvisor**: Collects, aggregates, and exposes container metrics; included in Kubelet.
- **Kubelet**: Node agent managing container resources. Provides resource metrics via /metrics/resource and /stats endpoints.
- **Node-level Resource Metrics**: Summarized per-node stats available through the kubelet's /metrics/resource endpoint.
- **Metrics Server**: Cluster add-on that collects and aggregates metrics from kubelets. It serves the Metrics API, used by HPA, VPA, and kubectl top.
- **Metrics API**: Kubernetes API for CPU and memory metrics, enabling workload autoscaling. Requires an API extension server to function.

3) Deploy metrics server

Download

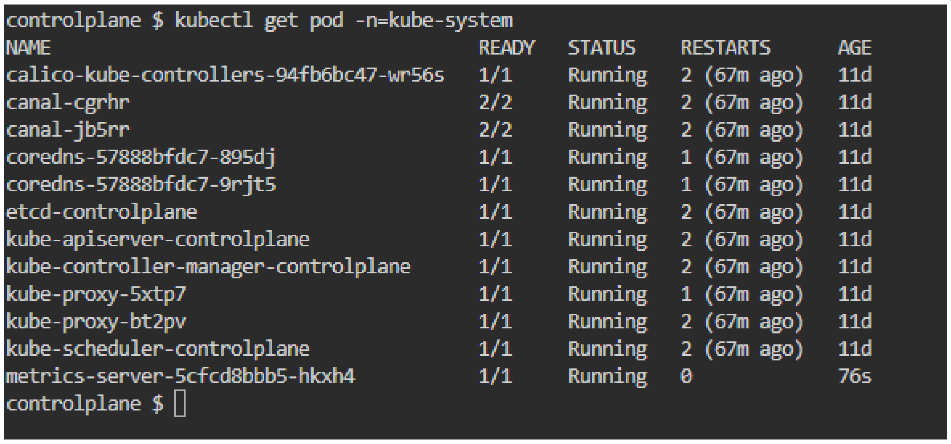
```
curl -O
https://raw.githubusercontent.com/MichaelRobotics/Kubernetes/main/KubernetesMetrics/metri
cs.yaml
```

Apply

```
kubectl apply -f metrics.yaml
```

Wait 10 seconds. Test if metrics pod is in running state & Ready

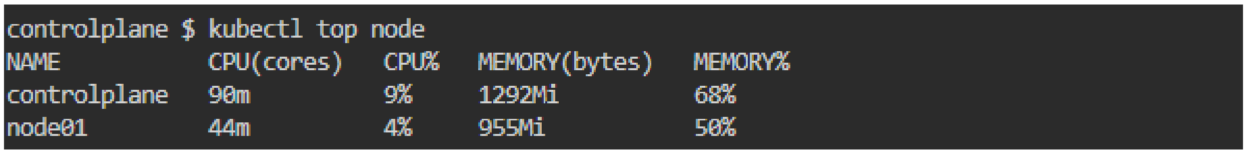
```
kubectl get pod -n=kube-system
```



```
controlplane $ kubectl get pod -n=kube-system
NAME                                      READY   STATUS    RESTARTS      AGE
calico-kube-controllers-94fb6bc47-wr56s   1/1     Running   2 (67m ago)   11d
canal-cgrhr                               2/2     Running   2 (67m ago)   11d
canal-jb5rr                               2/2     Running   2 (67m ago)   11d
coredns-57888bfdc7-895dj                  1/1     Running   1 (67m ago)   11d
coredns-57888bfdc7-9rjt5                  1/1     Running   1 (67m ago)   11d
etcd-controlplane                         1/1     Running   2 (67m ago)   11d
kube-apiserver-controlplane               1/1     Running   2 (67m ago)   11d
kube-controller-manager-controlplane      1/1     Running   2 (67m ago)   11d
kube-proxy-5xtp7                          1/1     Running   1 (67m ago)   11d
kube-proxy-bt2pv                          1/1     Running   2 (67m ago)   11d
kube-scheduler-controlplane               1/1     Running   2 (67m ago)   11d
metrics-server-5cfcd8bbb5-hkxh4           1/1     Running   0             76s
controlplane $ 
```

Check metrics

```
kubectl top node
```



```
controlplane $ kubectl top node
NAME           CPU(cores)   CPU%   MEMORY(bytes)   MEMORY%
controlplane   90m          9%     1292Mi          68%
node01         44m          4%     955Mi           50%
```

Kubernetes logging: Debugging Kubernetes nodes with crictl

1) What is crictl?

crictl is a command-line interface for CRI-compatible container runtimes. You can use it to inspect and debug container runtimes and applications on a Kubernetes node.

2) Install crictl

Download tar

```
wget https://github.com/kubernetes-sigs/cri-tools/releases/download/$VERSION/crictl-$VERSION-linux-amd64.tar.gz
```

Extract binary

```
tar -xzvf crictl-$VERSION-linux-amd64.tar.gz
```

Move binary to yout PATH

```
sudo mv crictl /usr/local/bin/
```

Verify the Installation

```
crictl --version
```

3) Basic usage

Generaly crictl is similar to docker CLI. Crictl is used primarly for Debugging Kubernetes node-level issues and managing containers and images in Kubernetes' runtime layer. Crictl can be helpful, when API server is down.

List pods

```
crictl pods
```

List pods by name:

```
crictl pods --name nginx-65899c769f-wv2gp
```

List all images:

```
crictl images
```

List images by repository:

```
crictl images nginx
```

List all containers:

```
crictl ps -a
```

Execute a command in a running container

```
crictl exec -i -t 1f73f2d81bf98 ls
```

Get a container's logs_

```
crictl logs 87d3992f84f74
```

common troubleshooting

1) Metrics Server Not Showing Metrics

Cause: Metrics Server cannot connect to kubelets or has TLS issues.
Solution: Check Metrics Server logs for errors. Ensure --kubelet-insecure-tls is set if needed. Verify network policies allow access to kubelet endpoints on port 10250.

2) Cluster Autoscaler Not Scaling Up

Cause: Container runtime misconfiguration or faulty logging setup.
Solution: Ensure the runtime service is running and check its logs. Use crictl to view container logs. Verify the logging driver configuration and restart the runtime if needed.

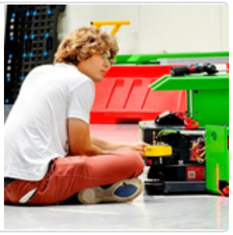
3) Karpenter Scaling Delays

Cause: Incorrect runtime endpoint configuration.
Solution: Verify the runtime socket path in the crictl config file. Ensure the runtime service is running and the socket exists.

4) Check my Kubernetes Troubleshooting series:

Michael Robotics
Hi, I'm Michal. I'm a Robotics Engineer and DevOps enthusiast. My mission is to create skill-learning platform that combats skill information overload by adhering to the set of principles: simplify, prioritize, and execute.

https://github.com/MichaelRobotics



Learn more about Kubernetes

Check Kubernetes and piyushsachdeva - great docs!

Setup a Multi Node Kubernetes Cluster

kubeadm is a tool to bootstrap the Kubernetes cluster

https://github.com/piyushsachdeva/CKA-2024/tree/main/Resources/Day27

Kubernetes Documentation

This section lists the different ways to set up and run Kubernetes

https://kubernetes.io/docs/setup/

Share, comment, DM and check GitHub for scripts & playbooks created to automate process.

Check my GitHub

Michael Robotics

Hi, I'm Michal. I'm a Robotics Engineer and DevOps enthusiast. My mission is to create skill-learning platform that combats skill information overload by adhering to the set of principles: simplify, prioritize, and execute.

https://github.com/MichaelRobotics

PS.

If you need a playbook or bash script to manage KVM on a specific Linux distribution, feel free to ask me in the comments or send a direct message!