

Lesson 07 Demo 05

Monitoring Container Logs

Objective: To view and check the container logs within a Kubernetes cluster using the `crictl` command for monitoring runtime operations

Tools required: kubeadm, kubectl, kubelet, and containerd

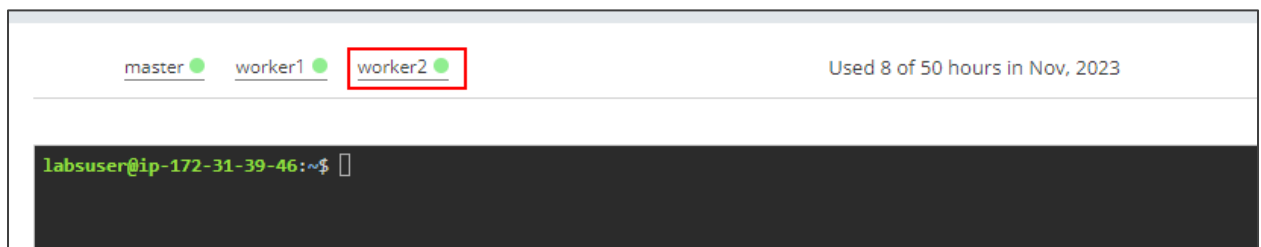
Prerequisites: A Kubernetes cluster (refer to Demo 01 from Lesson 01 for setting up a cluster)

Steps to be followed:

1. Check the container logs using the `crictl` command

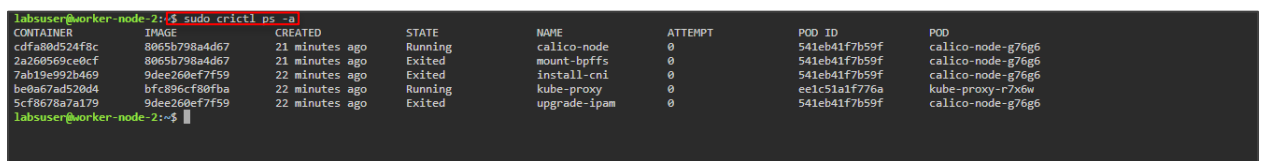
Step 1: Check the container logs using the `crictl` command

- 1.1 Navigate to worker-node-2 in the LMS dashboard



- 1.2 Fetch the container ID using the following command:

`sudo crictl ps -a`



- 1.3 Access and view the container logs for a specific container using the following command, replacing **<container-ID>** with the ID of the container that you copied in step 1.2:

sudo crictl logs <container-ID>

```
labsuser@worker-node-2:~$ sudo crictl logs cdfa80d524f8e
W1106 04:34:22.665510      8 feature_gate.go:241] Setting GA feature gate ServiceInternalTrafficPolicy=true. It will be removed in a future release.
2023-11-06 04:34:22.668 [INFO][8] startup/startup.go 432: Early log level set to info
2023-11-06 04:34:22.668 [INFO][8] startup/utils.go 126: Using NODENAME environment for node name worker-node-2.example.com
2023-11-06 04:34:22.669 [INFO][8] startup/utils.go 138: Determined node name: worker-node-2.example.com
2023-11-06 04:34:22.669 [INFO][8] startup/startup.go 94: Starting node worker-node-2.example.com with version v3.26.1
2023-11-06 04:34:23.156 [INFO][8] startup/startup.go 437: Checking datastore connection
2023-11-06 04:34:23.181 [INFO][8] startup/startup.go 461: Datastore connection verified
2023-11-06 04:34:23.181 [INFO][8] startup/startup.go 104: Datastore is ready
2023-11-06 04:34:23.585 [INFO][8] startup/startup.go 490: Initialize BGP data
2023-11-06 04:34:23.594 [INFO][8] startup/autodetection/methods.go 103: Using autodetected IPv4 address on interface ens5: 172.31.20.246/20
2023-11-06 04:34:23.594 [INFO][8] startup/startup.go 566: Node IPv4 changed, will check for conflicts
2023-11-06 04:34:23.598 [INFO][8] startup/startup.go 706: No AS number configured on node resource, using global value
2023-11-06 04:34:23.613 [INFO][8] startup/startup.go 751: found v4= in the kubeadm config map
2023-11-06 04:34:23.613 [INFO][8] startup/startup.go 755: found v6= in the kubeadm config map
2023-11-06 04:34:23.622 [INFO][8] startup/startup.go 823: Selected default IP pool is '192.168.0.0/16'
2023-11-06 04:34:23.622 [INFO][8] startup/startup.go 681: FELIX_IPV6SUPPORT is false through environment variable
2023-11-06 04:34:23.699 [INFO][8] startup/startup.go 217: Using node name: worker-node-2.example.com
2023-11-06 04:34:23.699 [INFO][8] startup/utils.go 190: Setting NetworkUnavailable to false
W1106 04:34:23.775687     36 feature_gate.go:241] Setting GA feature gate ServiceInternalTrafficPolicy=true. It will be removed in a future release.
2023-11-06 04:34:23.786 [INFO][36] tunnel-ip-allocator/env_var_loader.go 40: Found felix environment variable: "wireguardmtu"="0"
2023-11-06 04:34:23.786 [INFO][36] tunnel-ip-allocator/env_var_loader.go 40: Found felix environment variable: "vxlanmtu"="0"
2023-11-06 04:34:23.786 [INFO][36] tunnel-ip-allocator/env_var_loader.go 40: Found felix environment variable: "healthenabled"="true"
2023-11-06 04:34:23.786 [INFO][36] tunnel-ip-allocator/env_var_loader.go 40: Found felix environment variable: "defaultendpointtohostaction"="ACCEPT"
2023-11-06 04:34:23.786 [INFO][36] tunnel-ip-allocator/env_var_loader.go 40: Found felix environment variable: "ipinipmtu"="0"
2023-11-06 04:34:23.786 [INFO][36] tunnel-ip-allocator/env_var_loader.go 40: Found felix environment variable: "ipvsupport"="false"
2023-11-06 04:34:23.786 [INFO][36] tunnel-ip-allocator/config_params.go 481: Merging in config from environment variable: map[defaultendpointtohostaction:ACCEPT healthenabled:true ipinip
```

- 1.4 Retrieve the latest log entry for a specific container using the following command, replacing **<container-ID>** with the ID of the container that you copied in step 1.2:

sudo crictl logs --tail=1 <container-ID>

```
labsuser@worker-node-2:~$ sudo crictl logs --tail=1 cdfa80d524f8e
2023-11-06 05:15:25.933 [INFO][72] monitor-addresses/autodetection/methods.go 103: Using autodetected IPv4 address on interface ens5: 172.31.20.246/20
labsuser@worker-node-2:~$
```

By following these steps, you have successfully demonstrated the effective use of the `crictl` command to check container runtime operations.