Kubernetes-Metrics Server

Here's an overview of how it works:

Key Features:

1. **Resource utilization**: Metrics Server provides metrics on CPU, memory, and disk usage for pods and nodes.
2. **Scalability**: It can handle large clusters with thousands of nodes and tens of thousands of pods.
3. **Efficient storage**: Metrics are stored in a time series database (TSDB) called Prometheus, which is designed for efficient storage and querying.

Example Use Case:

Suppose you have a Kubernetes cluster with 10 nodes and 50 pods running various applications. You want to monitor the CPU utilization of each pod to ensure that it doesn't exceed a certain threshold.

Here's an example of how Metrics Server can help:

1. **Enable Metrics Server**: You enable Metrics Server in your cluster by creating a Deployment or ReplicaSet.
2. **Collect metrics**: Metrics Server collects resource metrics from the nodes and pods, including CPU utilization, memory usage, and disk I/O statistics.
3. **Expose metrics**: Metrics Server exposes these metrics as REST endpoints that can be queried using tools like kubectl.

kubectl Command:

```
kubectl get --raw /apis/metrics/v1beta1/pods | jq '.items[] | .metadata.name'
```

This command retrieves a list of all pods in the cluster and their corresponding CPU utilization metrics.

Output:

```
[
 {
   "metadata": {
     "name": "my-pod-12345"
   },
   "containers": [
     {
       "name": "my-container",
       "cpuUsageCores": 0.5,
       "memoryRssBytes": 10000000
     }
   ]
 },
 ...
]
```

In this example, the `cpuUsageCores` field represents the CPU utilization of each container in the pod.

Benefits:

1. **Easy monitoring**: Metrics Server provides easy-to-consume metrics that can be queried using tools like kubectl.
2. **Scalability**: It can handle large clusters with thousands of nodes and tens of thousands of pods.
3. **Efficient storage**: Metrics are stored in a TSDB designed for efficient storage and querying.

Overall, Metrics Server is an essential component in Kubernetes monitoring that provides valuable insights into resource utilization and helps operators optimize cluster performance.

Curated by Brajesh Kumar