### **To monitor CPU and Memory usage effectively, metrics-server and cAdvisor can provide additional insights:**

### 

### **1. metrics-server (for live CPU/Memory stats)**

**metrics-server** collects and provides resource usage data (CPU/Memory) for **Kubernetes nodes and pods**. This data is used by kubectl top and is available to Prometheus as well.

### **How to install metrics-server:**

#### **Step 1: Install metrics-server in your Kubernetes cluster**

Run the following to deploy it:

bash

CopyEdit

kubectl apply -f https://github.com/kubernetes-sigs/metrics-server/releases/latest/download/components.yaml

#### **Step 2: Check if it's running properly**

bash

CopyEdit

kubectl get pods -n kube-system

Ensure that the metrics-server pod is running.

#### **Step 3: Verify the metrics**

You can use kubectl top to see the live stats for nodes or pods:

bash

CopyEdit

kubectl top nodes

kubectl top pods

#### **Step 4: Configure Prometheus to scrape metrics-server**

You will need to add the metrics-server endpoint to Prometheus’ scrape configuration:

In prometheus.yml, add:

yaml

CopyEdit

- job\_name: 'metrics-server'

scheme: https

kubernetes\_sd\_configs:

- role: pod

relabel\_configs:

- source\_labels: [\_\_meta\_kubernetes\_pod\_label\_app]

target\_label: job

- source\_labels: [\_\_meta\_kubernetes\_namespace]

target\_label: namespace

### **✅ 2. cAdvisor (for per-container stats)**

**cAdvisor** provides detailed resource usage stats for individual containers, such as CPU, Memory, and Network metrics.

To install **cAdvisor**, follow these steps:

#### **Step 1: Deploy cAdvisor as a DaemonSet**

Create a cadvisor.yaml file with this content:

yaml

CopyEdit

apiVersion: apps/v1

kind: DaemonSet

metadata:

name: cadvisor

namespace: kube-system

spec:

selector:

matchLabels:

k8s-app: cadvisor

template:

metadata:

labels:

k8s-app: cadvisor

spec:

containers:

- name: cadvisor

image: google/cadvisor:latest

ports:

- containerPort: 8080

resources:

requests:

cpu: 100m

memory: 100Mi

limits:

cpu: 500m

memory: 500Mi

volumeMounts:

- mountPath: /rootfs

name: rootfs

- mountPath: /var/run

name: varrun

- mountPath: /sys

name: sys

- mountPath: /var/lib/docker

name: dockers

volumes:

- name: rootfs

hostPath:

path: /

- name: varrun

hostPath:

path: /var/run

- name: sys

hostPath:

path: /sys

- name: dockers

hostPath:

path: /var/lib/docker

Deploy cAdvisor using:

bash

CopyEdit

kubectl apply -f cadvisor.yaml

#### **Step 2: Access cAdvisor metrics**

After deploying cAdvisor, the metrics should be available at:

arduino

CopyEdit

http://<node-ip>:8080/metrics

If you want to monitor this with Prometheus, you need to add a scrape job for **cAdvisor** in prometheus.yml.

Example:

yaml

CopyEdit

- job\_name: 'cadvisor'

static\_configs:

- targets: ['<node-ip>:8080']

### **✅ 3. Combine metrics from metrics-server and cAdvisor**

You can combine data from **metrics-server** (for Kubernetes node/pod resource stats) and **cAdvisor** (for per-container resource stats) by adding the appropriate scrape configurations in Prometheus.

**Prometheus Queries** will now show you CPU, memory, and other container-specific metrics from both sources. Examples include:

**For CPU/Memory usage:** prometheus  
CopyEdit  
sum(rate(container\_cpu\_usage\_seconds\_total{container!="",pod!="",namespace!="kube-system"}[5m])) by (pod, namespace)

sum(container\_memory\_usage\_bytes{container!="",pod!="",namespace!="kube-system"}) by (pod, namespace)

**For node CPU/Memory:** prometheus  
CopyEdit  
sum(node\_cpu\_seconds\_total{mode="user"}) by (instance)

* sum(node\_memory\_MemTotal\_bytes) by (instance)