

VMware vRealize Operations Management Pack™ for Container Monitoring 1.2

vRealize Operations Manager



vmware®

You can find the most up-to-date technical documentation on the VMware website at:

<https://docs.vmware.com/>

If you have comments about this documentation, submit your feedback to

docfeedback@vmware.com

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2018 VMware, Inc. All rights reserved. [Copyright and trademark information.](#)

Contents

	vRealize Operations Management Pack for Container Monitoring 1.2	4
1	Introduction	5
2	Install the vRealize Operations Management Pack for Container Monitoring	6
	cAdvisor YAML Definition	6
3	Configure vRealize Operations Management Pack for Container Monitoring	11
4	Access Dashboards	13
	Kubernetes Overview	13
5	Metrics in vRealize Operations Management Pack for Container Monitoring	17
	Super Metrics	20
6	Alerts in vRealize Operations Management Pack for Container Monitoring	21
7	Reports in vRealize Operations Management Pack for Container Monitoring	26

vRealize Operations Management Pack for Container Monitoring 1.2

The VMware vRealize[®] Operations Management Pack[™] for Container Monitoring provides information for automating deployment and scaling operations of application containers across clusters of hosts providing container-centric infrastructure. The current version of the management pack supports monitoring Kubernetes clusters and containers deployed using the same.

Intended Audience

This information is intended for anyone who wants to install, and use VMware vRealize[®] Operations Management Pack[™] for Container Monitoring for monitoring their Kubernetes Clusters and containers deployed through Kubernetes.

Introduction

With VMware vRealize[®] Operations Management Pack[™] for Container Monitoring, Virtual Infrastructure Administrators can get complete Kubernetes topology of Namespaces, Clusters, Replica Sets, Nodes, Pods, and Containers for monitoring Kubernetes clusters. The OOTB dashboard not only provides an overview of Kubernetes eco-system but also helps in troubleshooting by highlighting the Key Performance Index and alerts for various objects pertaining to Kubernetes clusters that are monitored. This management pack extends the monitoring capability of vRealize Operations Manager to provide insights to the Kubernetes clusters to the Virtual Infrastructure administrator.

Install the vRealize Operations Management Pack for Container Monitoring

2

vRealize Operations Management Pack for Container Monitoring consists of a PAK file. You can install the vRealize Operations Management Pack for Container Monitoring to monitor cluster activity, pods, and containers.

Prerequisites

- Verify that you have the latest version of the management pack with administrator privileges.
- Install the cAdvisor DaemonSet on the cluster. Based on the Kubernetes settings, you must create a cAdvisor YAML definition. For more information, see [cAdvisor YAML Definition](#).

Procedure

- 1 Log in to the vRealize Operations Manager with administrator privileges.
- 2 From the main menu of vRealize Operations Manager, click **Administration**, and then in the left pane, click **Solutions**.
- 3 From the **Solutions** page, click the plus sign.
- 4 Browse to locate the temporary folder and select the PAK file.
- 5 Click **Upload**.
The upload might take several minutes.
- 6 Read and accept the EULA, and click **Next**.
- 7 When the vRealize Operations Management Pack for Container Monitoring is installed, click **Finish**.

cAdvisor YAML Definition

Before you install vRealize Operations Management Pack for Container Monitoring, you must deploy the cAdvisor DaemonSet on the cluster. Based on the Kubernetes settings, you must create a cAdvisor YAML definition.

Here are a few points to consider when you create a cAdvisor YAML definition:

- Containers running on *hostPort* must be accessible on your cluster. For example, the sample YAML definition on *hostPort* given below has port 31194 as the *hostPort*. So, the cluster must allow a connection on port 31194.

If the containers running on *hostPort* are not accessible, verify with *hostNetwork*. A sample YAML definition on *hostNetwork* specific to PKS is provided below.

- The docker path configured in the volume must be correct.

Note The docker path can be different based on your settings.

- All the nodes must have sufficient CPU and memory to run DaemonSets.
- You must use the *hostPort* defined in the YAML definition as the cAdvisor port when you create an adapter instance.

Sample cAdvisor YAML Definition on HostPort

```
apiVersion: apps/v1 # apps/v1beta2 in Kube 1.8, extensions/v1beta1 in Kube < 1.8
kind: DaemonSet
metadata:
  name: cadvisor
  namespace: kube-system
  labels:
    app: cadvisor
  annotations:
    seccomp.security.alpha.kubernetes.io/pod: 'docker/default'
spec:
  selector:
    matchLabels:
      app: cadvisor
  template:
    metadata:
      labels:
        app: cadvisor
        version: v0.31.0
    spec:
      tolerations:
        - key: node-role.kubernetes.io/master
          effect: NoSchedule
      containers:
        - name: cadvisor
          image: google/cadvisor:v0.31.0
          resources:
            requests:
              memory: 250Mi
              cpu: 250m
            limits:
              cpu: 400m
          volumeMounts:
            - name: rootfs
```

```

    mountPath: /rootfs
    readOnly: true
  - name: var-run
    mountPath: /var/run
    readOnly: true
  - name: sys
    mountPath: /sys
    readOnly: true
  - name: docker
    mountPath: /var/lib/docker #Mounting Docker volume
    readOnly: true
  - name: disk
    mountPath: /dev/disk
    readOnly: true
ports:
  - name: http
    containerPort: 8080 #Port exposed
    hostPort : 31194 #Host's port - Port to expose your cAdvisor DaemonSet on each node
    protocol: TCP
automountServiceAccountToken: false
terminationGracePeriodSeconds: 30
volumes:
  - name: rootfs
    hostPath:
      path: /
  - name: var-run
    hostPath:
      path: /var/run
  - name: sys
    hostPath:
      path: /sys
  - name: docker
    hostPath:
      path: /var/lib/docker #Docker path in Host System
  - name: disk
    hostPath:
      path: /dev/disk

```

Sample cAdvisor YAML Definition on HostNetwork

```

apiVersion: apps/v1beta2 # apps/v1beta2 in Kube 1.8, extensions/v1beta1 in Kube < 1.8
kind: DaemonSet
metadata:
  name: vrops-cadvisor
  namespace: kube-system
  labels:
    app: vrops-cadvisor
spec:
  selector:
    matchLabels:
      name: vrops-cadvisor
  template:
    metadata:

```



```

labels:
  name: vrops-cadvisor
  version: v0.31.0
spec:
  tolerations:
  - key: node-role.kubernetes.io/master
    effect: NoSchedule
  hostNetwork: true
  containers:
  - name: vrops-cadvisor
    image: google/cadvisor:v0.31.0
    imagePullPolicy: Always
    volumeMounts:
    - name: rootfs
      mountPath: /rootfs
      readOnly: true
    - name: var-run
      mountPath: /var/run
      readOnly: false
    - name: sys
      mountPath: /sys
      readOnly: true
    - name: docker
      mountPath: /var/lib/docker #Mounting Docker volume
      readOnly: true
    - name: docker-sock
      mountPath: /var/run/docker.sock
      readOnly: true
    - name: containerd-sock
      mountPath: /var/run/containerd.sock
      readOnly: true
    - name: disk
      mountPath: /dev/disk
      readOnly: true
  ports:
  - name: http
    containerPort: 31194 #Port exposed
    hostPort: 31194 #Host's port - Port to expose your cAdvisor DaemonSet on each node
    protocol: TCP
  securityContext:
    capabilities:
      drop:
      - ALL
      add:
      - NET_BIND_SERVICE
  args:
  - --port=31194
  - --profiling
  - --housekeeping_interval=1s
  terminationGracePeriodSeconds: 30
  volumes:
  - name: rootfs
    hostPath:
      path: /
  - name: var-run

```

```
hostPath:
  path: /var/run
- name: sys
  hostPath:
    path: /sys
- name: docker
  hostPath:
    path: /var/vcap/store/docker/docker #Docker path in Host System
- name: docker-sock
  hostPath:
    path: /var/vcap/sys/run/docker/docker.sock
- name: containerd-sock
  hostPath:
    path: /var/run/docker/containerd/docker-containerd.sock
- name: disk
  hostPath:
    path: /dev/disk
```

Configure vRealize Operations Management Pack for Container Monitoring

3

You can configure vRealize Operations Management Pack for Container Monitoring on vRealize Operations Manager after you install the solution.

Prerequisites

Verify that you have installed vRealize Operations Management Pack for Container Monitoring.

Procedure

- 1 From the main menu of vRealize Operations Manager, click **Administration**, and then in the left pane, click **Solutions**.
- 2 From the Solutions list, select **VMware vRealize Operations Management Pack for Container Monitoring**.
- 3 Click the **Configure** icon to edit an object.
- 4 Enter the display name of the adapter.
- 5 Enter the http URL of the Kubernetes master node in the **Master URL** text box.
- 6 Select either **Kubelet** or **DaemonSet** as the cAdvisor Service. You can select a cAdvisor service running inside the Kubelet or deployed externally as a DaemonSet.

Note By default, some Kubernetes deployments might have cAdvisor service disabled on Kubelet. In such a situation, the cAdvisor service must be enabled on Kubelet OR a standalone cAdvisor service must be deployed as a DaemonSet.

- 7 Enter the port number if cAdvisor is running as a **DaemonSet**.

8 Enter the **Credential** details of the Master URL.

- a Click the **Add New** icon.
- b Select the authentication to connect to the Kubernetes API Server.
vRealize Operations Management Pack for Container Monitoring currently supports basic, client certificate, and token authentication.

Table 3-1. Authentication Types

Authentication	Description
Basic Auth	Uses HTTP basic authentication to authenticate API requests through authentication plugins.
Client Certification Auth	Uses client certificates to authenticate API requests through authentication plugins.
Token Auth	Uses bearer tokens to authenticate API requests through authentication plugins.

For more information, see [Kubernetes Authentication](#).

9 Under **Advanced Settings**

- a Select the collector that is used to manage the adapter processes.
- b If the Kubernetes cluster is running on vCenter Server and the same server is monitored by the vCenter Adapter instance, you can view a link from the Kubernetes node to the vSphere VM. To view the link, enter the IP address of the vCenter Server instance.
- c If you want to monitor Java Process, then enable this option.
- d If you want to delete the non-existent objects for a defined period, then select the time frame from the drop-down menu.

Note The object deletion schedule is applicable to the Container Monitoring management pack only, and is over and above the global setting object deletion policy.

- e Click **Save Settings**.

10 Click **Close**.

4

Access Dashboards

You can use the dashboards to view and troubleshoot objects in your Kubernetes cluster eco-system that are monitored by vRealize Operations Management Pack for Container Monitoring.

Prerequisites

Verify that you have installed vRealize Operations Management Pack for Container Monitoring solution. See vRealize Operations Management Pack for Container Monitoring for information on installing.

Procedure

- 1 To access the dashboards, from the main menu of VMware vRealize Operations Manager, click **Dashboards**.
- 2 From the dashboard list, select **Kubernetes Overview**.

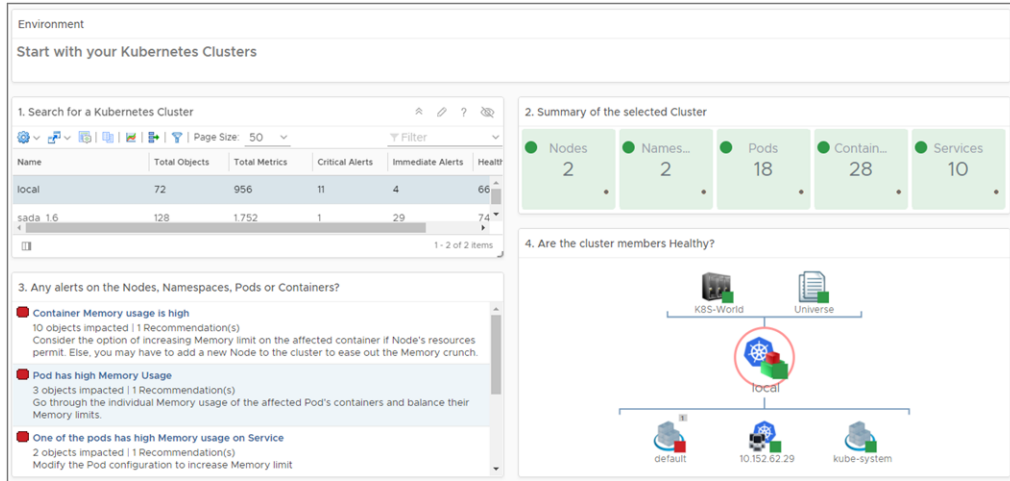
Kubernetes Overview

The overview dashboard provides an overall representation of Kubernetes environment, nodes, pods, and containers. The overview provides information of the overall health status of clusters, nodes, and pods with their respective historical trend and metric chart.

Kubernetes Overview - Environment

The Kubernetes overview environment widget provides an overall view of Kubernetes adapter instances, its associated objects information, alerts, and health status of objects.

Figure 4-1. Kubernetes Overview - Environment



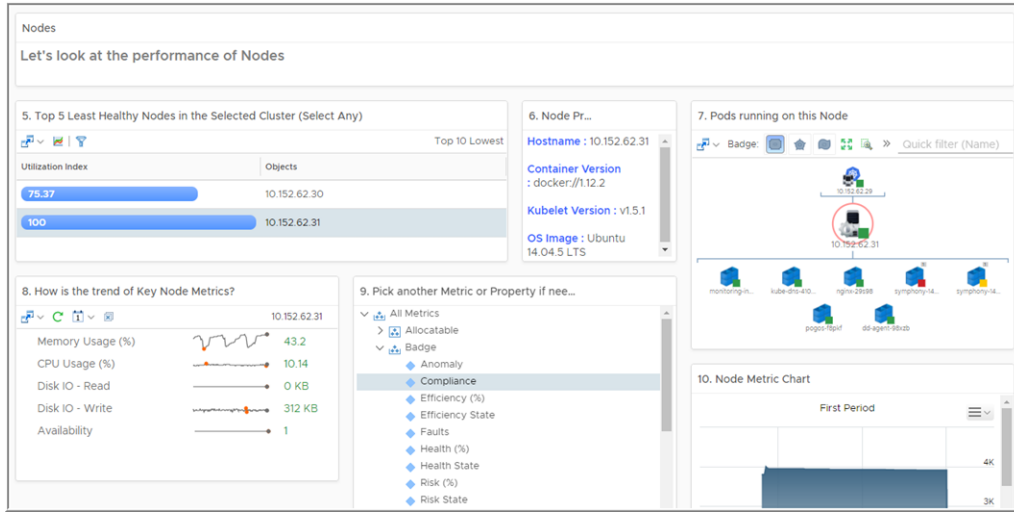
Widget Name	Description
Search for a Kubernetes Cluster	This widget displays only Kubernetes instances but not all objects types. You can retrieve the total metrics from the instances that are listed under this widget.
Summary of the Selected Cluster	This widget displays the total number of nodes, namespaces, pods, containers and services within the Kubernetes cluster.
Any Alerts on the Nodes, Namespaces, Pods or Containers	This widget displays all the immediate and critical alerts within a cluster of nodes, namespaces, pods or containers. When you select a object type from the Search for a Kubernetes Cluster widget, the corresponding alerts that are only immediate and critical gets populated.
Are the cluster members healthy?	This widget provides a hierarchical view of object relationship of a Kubernetes cluster.

Note The **Total Objects** column in the **Search for a Kubernetes Cluster** widget does not match with the sum of the objects in the **Summary of the Selected Cluster** widget. This is because the value of the total objects includes the total count of replica sets and Java container processes.

Kubernetes Overview - Nodes

The Kubernetes overview nodes widget provides detailed set of information of nodes, node properties, health status, metrics, and hierarchical representation of pod relationship.

Figure 4-2. Kubernetes Overview - Nodes

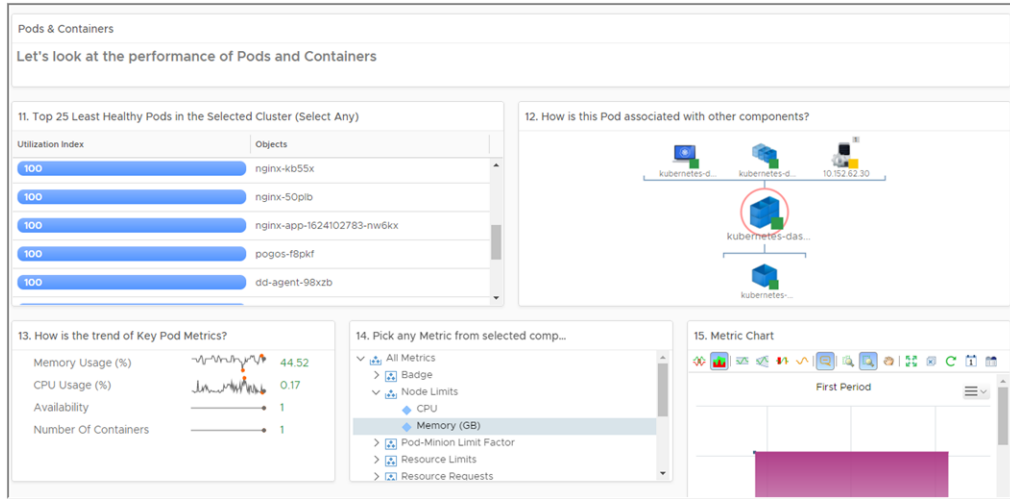


Widget Name	Description
Top 5 Least Healthy Nodes in the Selected Cluster	This widget displays the top 5 least healthy nodes in the selected cluster.
Node Properties	This widget displays respective node properties of the node that is selected in the Top 5 Least Healthy Nodes in the Selected Cluster widget.
Pods running on this Node	This widget provides a hierarchical view of pods and its relationship on a selected cluster.
How is the trend of Key Node Metrics	This widget displays the important metrics key metrics and its usage trend.
Pick another Metric or Property if needed	This widget lists all the metrics and properties for a selected node. This widget populates metrics and property of a node when it is selected from the Top 5 Least Healthy Nodes in the Selected Cluster widget.
Node Metric Chart	This widget provides a chart with the metric information of a metric or a property that is selected from the previous widget.

Kubernetes Overview - Pods and Container

The Kubernetes Overview pods and container widget provides detailed set of information of pods health status, hierarchical representation of pod relationship, metrics and so on.

Figure 4-3. Kubernetes Overview - Pods and Containers



Widget Name	Description
Top 25 Least Healthy Pods in the Selected Cluster	This widget displays the Top 5 least healthy pods in the selected cluster.
How is the Pod associated with other components	This widget provides a hierarchical view of pods and its relationship with other components on a selected cluster.
How is the trend of Key Pod metrics?	This widget displays the important key metrics and its usage trend.
Pick any Metric from the selected component	This widget lists all the metrics and properties for a selected pod. This widget populates metrics and property of a pod when it is selected from the How is this Pod Associated with other components widget.
Metric Chart	This widget provides a chart with the metric information of a metric or a property that is selected from the previous widget.

Metrics in vRealize Operations Management Pack for Container Monitoring

5

The Kubernetes adapter instance collects metrics for the objects that are available in the kubernetes container.

Table 5-1. Metrics in vRealize Operations Management Pack for Container Monitoring

Object Type	Group	Name	Unit	Type	Remarks
Kubernetes Minion	CPU Usage	System	cores	Metric	
Kubernetes Minion	CPU Usage	Total	cores	Metric	
Kubernetes Minion	CPU Usage	User	cores	Metric	
Kubernetes Minion	Filesystem	Free	MB	Metric	
Kubernetes Minion	Filesystem	Total	MB	Metric	
Kubernetes Minion	Filesystem	Used	MB	Metric	
Kubernetes Minion	Memory	Cache	MB	Metric	
Kubernetes Minion	Memory	Rss	MB	Metric	Resident Set Size --> Resident Memory
Kubernetes Minion	Memory	Usage	MB	Metric	
Kubernetes Minion	Memory	Working Set	MB	Metric	
Kubernetes Minion	Network Interface	Read	KB	Metric	
Kubernetes Minion	Network Interface	Read Dropped	Count	Metric	
Kubernetes Minion	Network Interface	Read Errors	Count	Metric	
Kubernetes Minion	Network Interface	Read Packets	Count	Metric	
Kubernetes Minion	Network Interface	Write	KB	Metric	
Kubernetes Minion	Network Interface	Write Dropped	Count	Metric	
Kubernetes Minion	Network Interface	Write Errors	Count	Metric	
Kubernetes Minion	Network Interface	Write Packets	Count	Metric	
Kubernetes Minion	Capacity	Memory	KB	Property	
Kubernetes Minion	Capacity	No of CPUs	Count	Property	
Kubernetes Minion	Capacity	No of Pods	Count	Property	
Kubernetes Minion	Node Info	Container Runtime Version		Property	

Table 5-1. Metrics in vRealize Operations Management Pack for Container Monitoring (Continued)

Object Type	Group	Name	Unit	Type	Remarks
Kubernetes Minion	Node Info	Kubelet Version		Property	
Kubernetes Minion	Node Info	OS Image		Property	
Kubernetes Minion	Allocatable	CPU	cores	Metric	
Kubernetes Minion	Allocatable	Memory	GB	Metric	
Kubernetes Service	Summary	Availability		Metric	
Kubernetes Pod	Summary	Availability		Metric	
Kubernetes Pod	Summary	No of Containers	Count	Metric	
Kubernetes Pod	Node Limits	CPU	cores	Metric	CPU limit set for the node on which a pod is running
Kubernetes Pod	Node Limits	Memory	GB	Metric	Memory limit set for the node on which a pod is running
Kubernetes Pod	Resource Limits	CPU	cores	Metric	Sum of CPU limit set for all containers under a pod
Kubernetes Pod	Resource Limits	Memory	MB	Metric	Sum of Memory limit set for all containers under a pod
Kubernetes Pod	Pod-Minion Limit Factor	CPU		Metric	Multiplier value used to find out how much %age of Node's CPU is used by a Pod
Kubernetes Pod	Pod-Minion Limit Factor	Memory		Metric	Multiplier value used to find out how much %age of Node's Memory is used by a Pod
Kubernetes Pod	Parent Node Name			Property	Name of a pod's parent node
Kubernetes Pod	Labels			Property	List of labels added to a pod which are dynamic
Kubernetes Pod	Resource Requests	CPU	cores	Metric	
Kubernetes Pod	Resource Requests	Memory	KB	Metric	
Container	CPU Usage	System	cores	Metric	
Container	CPU Usage	Total	cores	Metric	
Container	CPU Usage	User	cores	Metric	
Container	CPU	CPU contention	%	Metric	CPU usage % added for those containers that have limits specified
Container	CPU	Demand	Mhz	Metric	Unable to get these metrics from Cadvisor.
Container	CPU	I/O Wait	%	Metric	
Container	CPU	Swap wait	%	Metric	
Container	CPU	Co-stop	%	Metric	

Table 5-1. Metrics in vRealize Operations Management Pack for Container Monitoring (Continued)

Object Type	Group	Name	Unit	Type	Remarks
Container	Disk	Aggregate all instances	Read Latency	Metric	
Container	Filesystem	Base Usage	MB	Metric	
Container	Filesystem	Total Usage	MB	Metric	
Container	Memory	Cache	MB	Metric	
Container	Memory	Rss	MB	Metric	Resident Set Size --> Resident Memory
Container	Memory	Usage	MB	Metric	
Container	Memory	Working Set	MB	Metric	
Container	Memory	Contention	%	Metric	Memory usage % added for those containers that have limits specified
Container	Network Interface	Read	KB	Metric	Adapter has the capability to represent these metrics if and only if Cadvisor provides it.
Container	Network Interface	Read Dropped	Count	Metric	
Container	Network Interface	Read Errors	Count	Metric	
Container	Network Interface	Read Packets	Count	Metric	
Container	Network Interface	Write	KB	Metric	
Container	Network Interface	Write Dropped	Count	Metric	
Container	Network Interface	Write Errors	Count	Metric	
Container	Network Interface	Write Packets	Count	Metric	
Container	Network I/O	Workload	%	Metric	
Container	Disk IO	Async	KB	Metric	
Container	Disk IO	Sync	KB	Metric	
Container	Disk IO	Read	KB	Metric	
Container	Disk IO	Write	KB	Metric	
Container	Summary	Restart Count		Metric	
Container	Summary	Availability		Metric	
Container	Node Limits	CPU	cores	Metric	CPU limit set for the node on which a container is running
Container	Node Limits	Memory	GB	Metric	Memory limit set for the node on which a container is running
Container	Resource Limits	CPU	cores	Metric	CPU limit of a container set by the admin - 0 if not set
Container	Resource Limits	Memory	MB	Metric	Memory limit of a container set by the admin - 0 if not set
Container		Image		Property	

Table 5-1. Metrics in vRealize Operations Management Pack for Container Monitoring (Continued)

Object Type	Group	Name	Unit	Type	Remarks
Container	Resource Limits	CPU		Property	Based on meeting, pods should have CPU and MEM limits
Container	Resource Limits	Memory		Property	
Kubernetes Namespace	Events	Warning, Scheduled, Modified etc	Count	Metric	Number of Events per namespace segregated by Event Type

Super Metrics

The Kubernetes adapter instance also collects super metrics for the objects that are available in the kubernetes container.

Super Metric	Applied on Object	Enabled by Default
Memory Usage %	Pod	Yes
Memory Usage (MB)	Pod	No
CPU Usage %	Pod	Yes
CPU Usage (cores)	Pod	No
Memory Requests	Node	Yes
CPU Requests	Node	Yes
Max Pod Memory Usage %	Namespace, Service	Yes
Max Pod CPU Usage %	Namespace, Service	Yes

Alerts in vRealize Operations Management Pack for Container Monitoring

6

The Kubernetes objects raise alerts and alert symptom for instances that are available in the cluster.

Table 6-1. Alerts in vRealize Operations Management Pack for Container Monitoring

Alert Definition	Symptoms	Severity	Recommendation
Container CPU limit is set to unlimited	Container CPU limit is not defined	Info	A container running without CPU limit may claim all of Node's resources. Modify your Pod configuration with a CPU limit on the affected container. A quick glimpse at the CPU usage trend can help you set the limit.
Container CPU usage is high	Container CPU usage is higher than 90% Container CPU usage is higher than 80% Container CPU usage is higher than 70%	Critical Immediate Warning	Consider the option of increasing CPU limit on the affected container if Node's resources permit. Else, you may have to add a new Node to the cluster to ease out the CPU crunch.
Container Memory limit is set to unlimited	Container Memory limit is not defined	Info	A container running without Memory limit may claim all of Node's resources. Modify your Pod configuration with a Memory limit on the affected container. A quick glimpse at the Memory usage trend can help you set the limit.
Container is not available	Container is not available	Immediate	Redeploy the Pod and make sure it goes to Ready state
Container Memory usage is high	Container Process CPU usage is higher than 90% Container Process CPU usage is higher than 80% Container Process CPU usage is higher than 70%	Critical Immediate Warning	Consider the option of increasing Memory limit on the affected container if Node's resources permit. Else, you may have to add a new Node to the cluster to ease out the Memory crunch.
Container Process has high Memory Usage	Container Process Memory usage is higher than 90% Container Process Memory usage is higher than 80% Container Process Memory usage is higher than 70%	Critical Immediate Warning	Consider increasing CPU limit of the container.

Table 6-1. Alerts in vRealize Operations Management Pack for Container Monitoring (Continued)

Alert Definition	Symptoms	Severity	Recommendation
Container Process has high Memory Usage	Node Process Memory usage is higher than 90% Node Process Memory usage is higher than 80% Node Process Memory usage is higher than 70%	Critical Immediate Warning	Consider increasing Memory limit of the container.
Master Node is not available	Master Node is not available	Immediate	Ensure that the Master Node is reachable and API server is up and running.
Namespace is not Available	Namespace is not Available	Immediate	Check if the namespace has been deleted
Node has high CPU Usage	Node CPU usage is higher than 90% Node CPU Memory usage is higher than 80% Node CPU Memory usage is higher than 70% Node CPU Memory usage is higher than 60%	Critical Immediate Warning Info	Consider increasing CPU resource of the Node OR add a new Node to the cluster
Node is not available	Node is not available	Immediate	Verify if the Node is reachable and in Ready state
Node has high Memory Usage	Node Memory usage is higher than 90% NodeMemory usage is higher than 80% NodeMemory usage is higher than 70% NodeMemory usage is higher than 60%	Critical Immediate Warning Info	Consider increasing Memory resource of the Node OR add a new new Node to the cluster

Table 6-1. Alerts in vRealize Operations Management Pack for Container Monitoring (Continued)

Alert Definition	Symptoms	Severity	Recommendation
One of the Pods has highest CPU usage on Namespace	Pod with highest utilization on namespace has CPU usage higher than 90%	Critical	Consider modifying the affected Pod configurations to increase CPU limits
	Pod with highest utilization on namespace has CPU usage higher than 80%	Immediate	
	Pod with highest utilization on namespace has CPU usage higher than 70%	Warning	
	Descendant pod object (OR Operation)	Info	
	Pod memory usage is higher than 70%		
	Pod memory usage is higher than 80%		
	Pod memory usage is higher than 90%		
One of the pods has high Memory usage on Namespace	Pod with highest utilization on namespace has memory usage higher than 90%	Critical	Consider modifying the affected Pod configurations to increase Memory limits
	Pod with highest utilization on namespace has memory usage higher than 80%	Immediate	
	Pod with highest utilization on namespace has memory usage higher than 70%	Warning	
	Descendant pod object (OR Operation)	Info	
	Pod CPU Usage is higher than 70%		
	Pod CPU Usage is higher than 80%		
	Pod CPU Usage is higher than 90%		

Table 6-1. Alerts in vRealize Operations Management Pack for Container Monitoring (Continued)

Alert Definition	Symptoms	Severity	Recommendation
One of the pods has high CPU usage on Service	Pod with highest utilization on Service has CPU usage higher than 70% Pod with highest utilization on Service has CPU usage higher than 80% Pod with highest utilization on Service has CPU usage higher than 90% Descendant pod object (OR Operation) Pod CPU Usage is higher than 70% Pod CPU Usage is higher than 80% Pod CPU Usage is higher than 90%	Critical Immediate Warning	Consider modifying the affected Pod configurations to increase CPU limits
One of the pods has high Memory usage on Service	Pod with highest utilization on Service has memory usage higher than 70% Pod with highest utilization on Service has memory usage higher than 80% Pod with highest utilization on Service has memory usage higher than 90% Descendant pod object (OR Operation) Pod memory usage is higher than 70% Pod memory usage is higher than 80% Pod memory usage is higher than 90%	Critical Immediate Warning	Consider modifying the affected Pod configurations to increase Memory limits
Pod has high CPU Usage	Pod CPU Usage is higher than 90% Pod CPU Usage is higher than 80% Pod CPU Usage is higher than 70% Pod CPU Usage is higher than 60%	Critical Immediate Warning Info	Go through the individual usage of the affected Pod's containers and balance their CPU limits.

Table 6-1. Alerts in vRealize Operations Management Pack for Container Monitoring (Continued)

Alert Definition	Symptoms	Severity	Recommendation
Pod has high Memory Usage	Pod memory usage is higher than 90% Pod memory usage is higher than 80% Pod memory usage is higher than 70% Pod memory usage is higher than 60%	Critical Immediate Warning Info	Go through the individual usage of the affected Pod's containers and balance their Memory limits.
Pod is not available	Pod is not available	Critical	Redeploy the Pod and make sure it goes to Ready state
ReplicaSet is not available	ReplicaSet is not available	Immediate	Make sure that the Replica Set is present.
Service is not available	Service is not available	Immediate	Make sure that the Service is present.
Sum of Resource Requests of Pods exceed Node Capacity	CPU Requests greater than node capacity Memory Requests greater than node capacity	Critical	Minimum CPU/Memory resources required to run the Pods of the affected node has exceeded Node capacity. Consider increasing Node resources OR add more Nodes to distribute the workload.

Reports in vRealize Operations Management Pack for Container Monitoring



A report is a snapshot of views. The reports provide a view of Kubernetes adapter instance objects in XSL and PDF format. The report in the Container Monitoring management pack is called as Kubernetes Adapter Instance Summary.

The report is based on the following views:

- Kubernetes adapter instance objects
- Containers with no memory limit
- Container with no CPU limit