

Resolve Kubernetes Pods Show in Not Ready State after Site Isolation

Contents

[Introduction](#)

[Problem](#)

[Solution](#)

[Initial Pod and System Verification](#)

[Restart of Affected Pods](#)

[Verify Pods and System Status After Restart](#)

Introduction

This document describes recovery steps when the Cisco Smart Install (SMI) pod gets into the not ready state due to Kubernetes bug <https://github.com/kubernetes/kubernetes/issues/82346>.

Problem

After site isolation, Converged Ethernet (CEE) reported the Processing Error Alarm in the CEE. The system ready status is below 100%.

```
[sitelapp/pod1] cee# show alerts active
alerts active k8s-deployment-replica-mismatch f89d8d09389c
state active
severity critical
type "Processing Error Alarm"
startsAt 2021-05-27T08:38:58.703Z
source sitelapp-smi-cluster-policy-oam2
labels [ "component: kube-state-metrics" "deployment: prometheus-scrapeconfigs-synch"
"exported_namespace: cee-pod1" "instance: 192.0.2.37:8080" "job: kubernetes-pods" "namespace:
cee-pod1" "pod: kube-state-metrics-6c476f7494-tqkrc" "pod_template_hash: 6c476f7494" "release:
cee-pod1-cnat-monitoring" ]
annotations [ "summary: Deployment cee-pod1/prometheus-scrapeconfigs-synch has not matched the
expected number of replicas for longer than 2 minutes." ]
```

```
[sitelapp/pod1] cee# show system status
system status deployed true
system status percent-ready 92.68
```

```
ubuntu@sitelapp-smi-cluster-policy-mas01:~$ kubectl get rs -n cee-pod1 | grep scrape
NAME DESIRED CURRENT READY AGE
prometheus-scrapeconfigs-synch-ccd454f76 1 1 0 395d
prometheus-scrapeconfigs-synch-f5544b4f8 0 0 0 408d
```

Solution

The site isolation is a trigger for the bug <https://github.com/kubernetes/kubernetes/issues/82346>. The workaround to have these pods in **Ready** state is to restart the affected pods. The fix is included in upcoming CEE releases.

Initial Pod and System Verification

Log in to CEE CLI and check system status.

```
ssh -p 2024 admin@`kubect1 get svc -A | grep " ops-center-cee" | awk '{print $4}'`
```

```
show alerts active
show system status
```

Restart of Affected Pods

Log in to the primary node, on the primary, run these commands. And identify daemonsets and replica sets that have not all members in **Ready** state.

```
kubect1 get daemonsets -A
kubect1 get rs -A | grep -v '0 0 0'
```

Copy and paste these commands in the notepad and replace all cee-xyz, with the cee namespace on the site.

```
kubect1 describe pods core-retriever -n cee-xyz | egrep "^Name:|False" | grep -B1 False
kubect1 describe pods calico-node -n kube-system | egrep "^Name:|False" | grep -B1 False
kubect1 describe pods csi-cinder-nodeplugin -n kube-system | egrep "^Name:|False" | grep -B1 False
kubect1 describe pods maintainer -n kube-system | egrep "^Name:|False" | grep -B1 False
kubect1 describe pods kube-proxy -n kube-system | egrep "^Name:|False" | grep -B1 False
kubect1 describe pods path-provisioner -n cee-xyz | egrep "^Name:|False" | grep -B1 False
kubect1 describe pods logs-retriever -n cee-xyz | egrep "^Name:|False" | grep -B1 False
kubect1 describe pods node-exporter -n cee-xyz | egrep "^Name:|False" | grep -B1 False
kubect1 describe pods keepalived -n smi-vips | egrep "^Name:|False" | grep -B1 False
kubect1 describe pods prometheus-scrapeconfigs-synch -n cee-xyz | egrep "^Name:|False" | grep -B1 False
```

Execute the commands and collect the result output. In the result, output identifies the pod names with the corresponding namespace that require a restart.

Restart all affected pods from the list obtained previously when you issue these commands (replace pod name and namespace accordingly).

```
kubect1 delete pods core-retriever-abcde -n cee-xyz
kubect1 delete pods core-retriever-abcde -n cee-xyz
```

...

Verify that the pods are up and running without any issue.

```
kubect1 get pods -A
```

Verify Pods and System Status After Restart

Execute commands:

```
kubect1 get daemonsets -A
```

```
kubectl get rs -A | grep -v '0 0 0'
```

Confirm that daemonsets and replica sets show all members in **Ready** state.

Log in to CEE CLI and confirm that no active alerts and system status must be at 100%.

```
ssh -p 2024 admin@`kubectl get svc -A | grep " ops-center-cee" | awk '{print $4}'`
```

```
show alerts active
```

```
show system status
```