Troubleshoot Kubernetes Pod Not Ready in PCF

Contents

Introduction

This document describes the procedure to troubleshoot Kubernetes Pod Not Ready seen in the Policy Control Function (PCF).

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- PCF
- 5G Cloud Native Deployment Platform (CNDP)
- Dockers and Kubernetes

Components Used

The information in this document is based on these software and hardware versions:

- PCF REL_2023.01.2
- Kubernetes v1.24.6

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

In this setup, the CNDP hosts PCF.

When a Kubernetes Pod is in a 'Not Ready' state, it means that the Pod is not currently able to serve traffic because one or more of its containers are not in a ready state. This can be due to various reasons, such as containers that are still starting up, failing health checks, or encountering errors.

Problem

You see alerts on the Common Execution Environment (CEE) for Kubernetes (K8s) pods, not ready state.

```
Command:
cee# show alerts active summary summary
Example:
[pcf01/pcfapp] cee# show alerts active summary
NAME UID SEVERITY STARTS AT DURATION SOURCE SUMMARY
------------------------------------------------------------------------------------------------------------
k8s-pod-not-ready 35b143f885ec critical 06-13T08:22:05 mirror-maker-0 Pod pcf-pcf/mirror-maker-0 has be
k8s-pod-crashing-loop 990b651ad5f5 critical 04-19T22:51:08 pcf01-master-2 Pod cee-irv1bmpcf/pgpool-65fc
k8s-pod-restarting a44d31701faf minor 04-19T01:55:38 pcf01-master-2 Pod cee-irv1bmpcf/pgpool-65fc8b8d5f
k8s-deployment-replic b8f04c540905 critical 04-06T01:53:48 pcf01-master-2 Deployment cee-irv1bmpcf/pgpo
k8s-pod-not-ready cb2c8ee4a9c9 critical 04-06T01:53:48 pgpool-65fc8b8d5f-2w9 Pod cee-pcf/pgpool-65fc8b8
```

Analysis

Approach 1

After logging into a Kubernetes (K8s) node, check the alerts for a node that is currently in a 'Not Ready' status.
It is advisable to check any upgrade or maintenance process. At that time, the pod can be taken offline in order to apply updates, install new software, or perform other necessary tasks. or ongoing maintenance activities that can impact pod availability.
The mirror pods are expected to be in the mentioned state during the Site upgrade activity on the peer site (GR).

Connect to the master node and verify the mirror pod status.

```
cloud-user@pcf01-master-1:~$ kubectl get pods -A -o wide | grep mirror
NAMESPACE NAME READY STATUS RESTARTS AGE
pcf-pcf01 mirror-maker-0 0/1 Running 1 5d23h
#Post upgrade Activity mirror-maker pod status
cloud-user@pcf01-master-1:~$ kubectl get pods -A|grep mirror
pcf-pcf01 mirror-maker-0 1/1 Running 1 6d.
```

Approach 2

If you receive alerts indicating that a Kubernetes (K8s) pod is not in a ready state within the CEE, it suggests that the pod is experiencing issues and cannot be considered fully operational. This state typically implies that the pod is unable to accept traffic or fulfill its intended function.
Analyze the alerts and related information in order to understand the cause of the 'Not Ready' status. The alerts can provide details about the specific issue or trigger that led to the status change of the node.
Common reasons for a node being in a 'Not Ready' status include resource constraints, network connectivity issues, hardware failures, or configuration problems.

Step 1. Verify the pod status by using the kubectl get pods command in order to check the status of the pod and look for the pod status. If the pod is not ready, it can display a status such as 'Pending', 'CrashLoopBackOff', or 'Error'.

```
cloud-user@pcf01-master-1:~$ kubectl get pods -A -o wide | grep -v Running
NAMESPACE NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES
cee-pcf pgpool-65fc8b8d5f-2w9nq 0/1 CrashLoopBackOff 147 (117s ago) 8d xxx.xxx.xxx.xx pcf01-master-2 <no
All nodes including master-2 are in Ready state
cloud-user@pcf01-master-1:~$ kubectl get nodes
NAME STATUS ROLES AGE VERSION
Pcf01-master-1 Ready control-plane 667d v1.24.6
Pcf01-master-2 Ready control-plane 9d v1.24.6
Pcf01-master-3 Ready control-plane 9d v1.24.6
Pcf01-worker-1 Ready <none> 9d v1.24.6
Pcf01-worker-10 Ready <none> 9d v1.24.6
Pcf01-worker-11 Ready <none> 9d v1.24.6
Pcf01-worker-12 Ready <none> 9d v1.24.6
Pcf01-worker-13 Ready <none> 9d v1.24.6
Pcf01-worker-14 Ready <none> 9d v1.24.6
Pcf01-worker-15 Ready <none> 9d v1.24.6
Pcf01-worker-16 Ready <none> 9d v1.24.6
Pcf01-worker-17 Ready <none> 9d v1.24.6
Pcf01-worker-18 Ready <none> 9d v1.24.6
Pcf01-worker-19 Ready <none> 9d v1.24.6
Pcf01-worker-2 Ready <none> 9d v1.24.6
Pcf01-worker-20 Ready <none> 9d v1.24.6
Pcf01-worker-21 Ready <none> 9d v1.24.6
Pcf01-worker-22 Ready <none> 9d v1.24.6
Pcf01-worker-23 Ready <none> 9d v1.24.6
Pcf01-worker-3 Ready <none> 9d v1.24.6
Pcf01-worker-4 Ready <none> 9d v1.24.6
Pcf01-worker-5 Ready <none> 9d v1.24.6
pcf01-worker-6 Ready <none> 9d v1.24.6
pcf01-worker-7 Ready <none> 9d v1.24.6
pcf01-worker-8 Ready <none> 9d v1.24.6
pcf01-worker-9 Ready <none> 9d v1.24.6
```

Step 2. Log in to master VIP and get the pgpool pod.

```
cloud-user@pcf01-master-1:~$ kubectl get pods -A -o wide | grep -i pgpool
cee-pcf01 pgpool-65fc8b8d5f-2w9nq 0/1 Running 3173 (3m58s ago) 22d xxx.xxx.xxx.xx pcf01-master-2 <none>
cloud-user@pcf01-master-1:~$
```

Step 3. Delete the pgpool pod.

```
cloud-user@pcf01-master-1:~$ kubectl delete pod <pgpool pod name> -n cee-pcf
```

Step 4. Verify that the new pgpool pod is running fine.

```
cloud-user@pcf01-master-1:~$ kubectl get pods -A -o wide | grep -i pgpool
```

Step 5. Verify the alerts related to the pgpool pod have been cleared on the CEE ops center.

```
[pcf01/pcfapp] cee# show alerts active summary
```