# Deploy and Configure CPC through Ops Center



## Cisco CPC architecture

The CPC Ops Center allows you to configure the CPC features such as configuring the license, CPC Engine, REST endpoint, and CDL. You can also configure the NRF components that enable the interworking of various NFs.

Policy Ops Center reuses the existing Ops Center image from mobile-cnat-infrastructure, and is accessible via the ingresses that are defined by that chart.

## Prerequisites

Base CPC configuration is a comprehensive setup that

- Ensure that all the virtual network functions (VNFs) are deployed.
- Run the SMI sync operation for the CPC Ops Center and Cloud Native Common Execution Environment (CN-CEE).

## Deploying and Accessing CPC

This section describes how to deploy and access the CPC Ops Center.

Deploying CPC involves these steps:

1. Deploying CPC
2. Accessing the CPC Ops Center

# Deploying CPC

The Subscriber Microservices Infrastructure (SMI) platform is responsible for deploying and managing the Cloud Native 5G CPC application and other network functions.

For information on how to deploy CPC Ops Center on a vCenter environment, see *Configuring the vCenter Environment* section in *Ultra Cloud Core SMI Cluster Deployer Operations Guide*.

For deploying CPC Ops Center on an OpenStack environment, see *UAME-based VNF Deployment* section in the *UAME-based 4G and 5G VNF Deployment Automation Guide, Release 6.9*.

For information on how to deploy CPC Ops Center on bare metal servers (currently Cisco UCS-C servers) environment, see *Operating the SMI Cluster Manager on Bare Metal* section in *Ultra Cloud Core Subscriber Microservices Infrastructure — Operations Guide*.

# Accessing the CPC Ops Center

This section describes how to access the CPC Ops Center.

You can access the CPC Ops Center from the console application or the Web-based CLI console. Depending upon your selection, access one of the following from the master node:

1. **CLI:**

   **ssh admin@***ops_center_pod_ip* **-p 2024**

2. **Web-based console:**

   a. Log in to the Kubernetes master node.

   b. To view the available ingress connections, use the following configuration:

      **kubectl get ingress** *namespace*

      The available ingress connections are displayed.

   c. Select the appropriate ingress from where you want to run Ops Center and open the following URL from the browser:

      **cli.***namespace***-ops-center.***ip_address***.nip.io**

# Sample configuration

**Important** The mandatory parameters are required to ensure that the critical pods such as CRD and Policy Engine are in the running state.

```
config
datastore primary-endpoint connection-settings keep-alive keep-alive-time-ms 200
datastore primary-endpoint connection-settings channel count 4
datastore primary-endpoint connection-settings timeout-ms 300
ldap replicas 2
ldap server-set USD
 search-user dn cn=sdcUser,dc=C-NTDB
```

```
 search-user password siemens
 health-check interval-ms 10000
 health-check dn cn=sdcUser,dc=C-NTDB
 health-check filter msisdn=918369110173
 health-check attributes napCustType
 initial-connections                              10
 max-connections                                  10
 retry-count                                      2
 retry-timer-ms                                   100
 max-failover-connection-age-ms                   60000
 binds-per-second                                 0.2
 number-consecutive-timeouts-for-bad-connection 2
 missing-attribute-result-code                    32
 connection 192.0.2.18 389
  priority        400
  connection-rule ROUND_ROBIN
  auto-reconnect  true
  timeout-ms      200
  bind-timeout-ms 3000
 exit
 connection 192.0.2.18 389
  priority        400
  connection-rule ROUND_ROBIN
  auto-reconnect  true
  timeout-ms      200
  bind-timeout-ms 3000
 exit
exit
//This is a mandatory parameter
db global-settings db-replica 3
//This is a mandatory parameter
db spr shard-count 1
db balance shard-count 1
debug tracing type DISABLED
debug splunk hec-url http://192.0.2.18:6066
debug splunk hec-token efd41999-fa22-1111-aca7-2222fb36c6af
debug splunk batch-interval-ms 100
debug splunk batch-count 10
debug splunk batch-size-bytes 102400
debug logging default-level error
diameter settings timeouts-ms dpa 5000
diameter application rx
 application-id   16777236
 tgpp-application true
 vendor           [ 10415 ]
exit
diameter group rx-pcf01-proto1
 mode server
 stack rx
  application rx
  bind-ip     192.0.2.18
  bind-port   3868
  fqdn        pcf-rx-server-1
  realm       pcf.rx.server.cisco.com
  node-host   pcf01-protocol01
 exit
exit
diameter group rx-pcf01-proto2
 mode server
 stack rx
  application rx
  bind-ip     192.0.2.18
  bind-port   3868
  fqdn        pcf-rx-server-2
```

```
  realm       pcf.rx.server.cisco.com
  node-host   pcf01-protocol02
 exit
exit
//This is a mandatory parameter
rest-endpoint ips    [ 192.0.2.18 192.0.2.19 192.0.2.20 ]
//This is a mandatory parameter
rest-endpoint port   9088
rest-endpoint tracing-service-name pcf-rest-endpoint
rest-endpoint replicas 2
rest-endpoint interface n28
 ip [ 192.0.2.18 ]
exit
rest-endpoint interface n7
 ip [ 192.0.2.19 ]
exit
rest-endpoint interface nnrf
 ip [ 192.0.2.20 ]
exit
advance-tuning overload-control rest global limits max-requests-per-sec 5000
advance-tuning overload-control rest global action throttle-action REJECT
advance-tuning overload-control diameter global limits max-requests-per-sec 5000
advance-tuning overload-control diameter global action throttle-action DROP
advance-tuning async-threading request-timeout-ms 300
api unified engine-group pcf01production
//This is a mandatory parameter
engine pcf01production
//This is a mandatory parameter
  replicas              2
//This is a mandatory parameter
  unified-api-replicas  1
//This is a mandatory parameter
  subversion-run-url    http://svn/repos/run
//This is a mandatory parameter
  subversion-config-url http://svn/repos/configuration
//This is a mandatory parameter
  tracing-service-name  pcf-engine
  grpc port   5005
  grpc externalIPs [ 192.0.2.21 ]
  properties com.cisco.engine.log.type
    value 2
 exit
 properties disableCommandClient
  value true
 exit
 properties ldap.retry.time.ms
  value 200
 exit
exit
label protocol-layer key smi.cisco.com/node-type
label protocol-layer value protocol
label service-layer key smi.cisco.com/node-type
label service-layer value service
label cdl-layer key smi.cisco.com/node-type
label cdl-layer value session
label oam-layer key smi.cisco.com/node-type
label oam-layer value oam
profile nf-client nf-type chf
 chf-profile chprofile
  locality locchf1
   priority 10000
   service name type nchf-spendinglimitcontrol
    endpoint-profile chf-profile
     capacity   10
```

```
     priority   10
     uri-scheme http
     version
      uri-version v1
      exit
     exit
     endpoint-name ep1
      primary ip-address ipv4 192.0.2.22
      primary ip-address port 5088
     exit
    exit
   exit
  exit
 exit
exit
profile nf-pair nf-type CHF
 nrf-discovery-group    discoveryGroup
 subscription-enabled   false
 subscription-extension 3
 locality client  pcf01
 locality preferred-server loc1
 locality geo-server loc2
exit
//This is a mandatory parameter
service-registration profile locality pcf01
//This is a mandatory parameter
service-registration profile plmn-list 100 010
//This is a mandatory parameter
exit
//This is a mandatory parameter
service-registration profile plmn-list 123 456
//This is a mandatory parameter
exit
//This is a mandatory parameter
service-registration profile snssais 1
//This is a mandatory parameter
 sd 123456
exit
group nf-mgmt group1
 nrf-mgmt-group registerGroup
 locality       pcf01
 failover sla 100
 reconnect interval 60000
exit
group nrf discovery discoveryGroup
 service type nrf nnrf-disc
  endpoint-profile discoveryProfile
   capacity   10
   priority   10
   uri-scheme http
   version
    uri-version v1
    exit
   exit
   endpoint-name ep1
    priority 1
    capacity 10
    primary ip-address ipv4 192.0.2.12
    primary ip-address port 8183
    secondary ip-address ipv4 192.0.2.12
    secondary ip-address port 8184
    tertiary ip-address ipv4 192.0.2.12
    tertiary ip-address port 8185
   exit
```

```
  exit
 exit
exit
group nrf mgmt registerGroup
 service type nrf nnrf-nfm
  endpoint-profile registerProfile
   capacity   20
   priority   10
   uri-scheme http
   version
    uri-version v1
    exit
   exit
   endpoint-name ep1
    priority 1
    capacity 10
    primary ip-address ipv4 192.0.2.12
    primary ip-address port 8183
    secondary ip-address ipv4 192.0.2.12
    secondary ip-address port 8184
    tertiary ip-address ipv4 192.0.2.12
    tertiary ip-address port 8185
   exit
  exit
 exit
exit
cdl system-id 1
cdl node-type session
cdl zookeeper data-storage-size 1
cdl zookeeper log-storage-size 1
cdl zookeeper replica 3
cdl logging default-log-level warn
cdl datastore session
 cluster-id 1
 endpoint replica 4
 index map 4
 slot replica 2
 slot map 10
 slot notification limit 10
exit
cdl kafka replica 3
cdl kafka storage 1
end
```