Troubleshoot Switchover Problem on RCM Converged Core

Contents

Introduction

This document describes the basic steps to troubleshoot on Redundancy Configuration Manager (RCM) in case of a network fault event.

Background Information

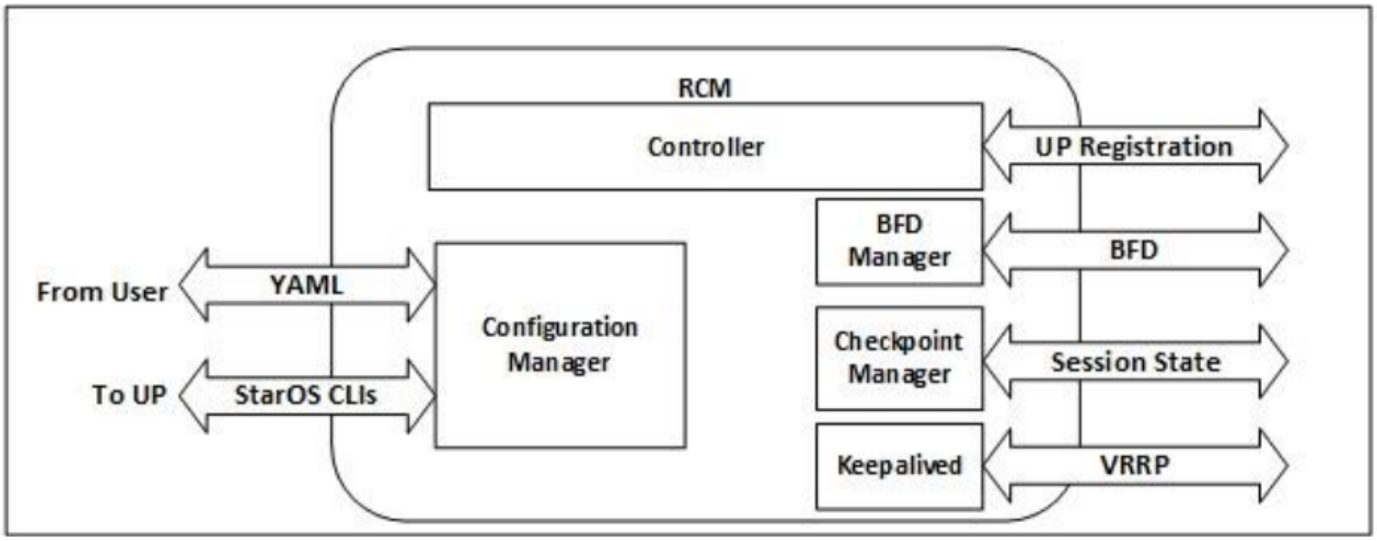
What is RCM?

The RCM is a Cisco proprietary node or network function (NF) that provides redundancy for StarOS-based User Plane Functions (UPF).

The RCM provides N:M redundancy of UPF wherein N is a number of Active UPFs and is less than 10, and M is a number of Standby UPs in the redundancy group.

Components of RCM

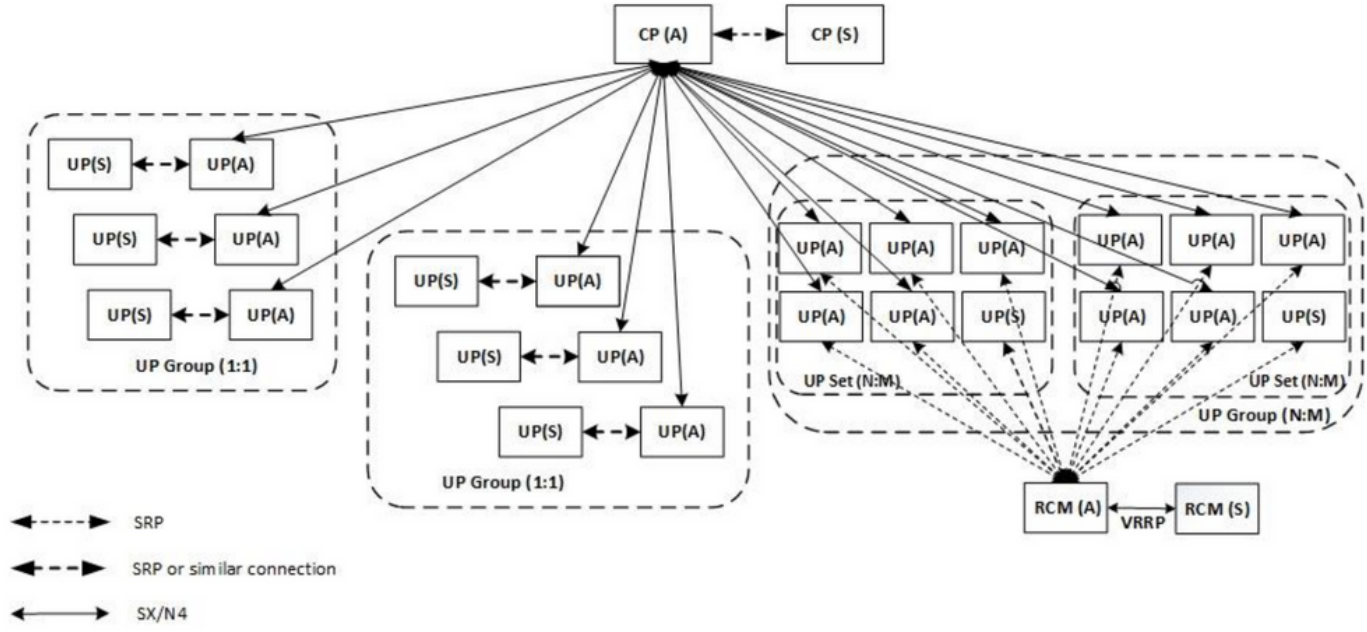
The RCM comprises components that run as pods in the RCM VM:



- Controller: It communicates event-specific decisions with all the other pods in RCM
- BFD Manager (BFDMgr): It uses the BFD protocol to identify the state of the data plane
- Configuration Manager (ConfigMgr): It loads the requested configuration to the User Planes (UPs)
- Redundancy Manager (RedMgr): It is also called the Checkpoint Manager. It stores and sends the checkpoint data to a standby UPF

- Keepalived: It communicates between Active and Standby RCM with the use of VRRP

Typical RCM Deployment Model



RCM CLI Overview

In this example, there are four RCM OPS centers. In order to confirm what RCM Kubernetes corresponds to which RCM OPS Center and RCM Common Execution Environment (CEE) you can log in to the RCM Kubernetes and list the namespaces:

```
cloud-user@up0300-aio-1-primary-1:~$ kubectl get namespace
NAME              STATUS   AGE
cee-rce31       Active   54d
default           Active   57d
istio-system      Active   57d
kube-node-lease   Active   57d
kube-public       Active   57d
kube-system       Active   57d
nginx-ingress     Active   57d
rcm-rm31       Active   54d
rcm-rm33       Active   54d
registry          Active   57d
smi-certs         Active   57d
smi-node-label    Active   57d
smi-vips          Active   57d

cloud-user@up300-aio-2-primary-1:~$ kubectl get namespace
NAME              STATUS   AGE
cee-rce32         Active   54d
default           Active   57d
istio-system      Active   57d
kube-node-lease   Active   57d
kube-public       Active   57d
kube-system       Active   57d
nginx-ingress     Active   57d
rcm-rm32          Active   54d
rcm-rm34          Active   54d
registry          Active   57d
smi-certs         Active   57d
smi-node-label    Active   57d
smi-vips          Active   57d
```

UPF Management IP Address

This IP is specific and tied to VM or UPF. It is used in initial communication between UPF and RCM, where UPF registers with RCM and RCM configures UPF and also assigns role. You can use this IP to identify UPF from RCM CLI outputs.

UPF Device Role IP

Linked to a role (active/standby):

This IP address moves as the switchover happens.

Useful CLI Commands for RCM Troubleshoot

You can review which RCM group is the UPF from RCM OPS Center. Find a sample from Cloud Native Deployment Platform (CNDP):

```
[local]UPF317# show rcm info
Redundancy Configuration Module:
-------------------------------------------------------------------------------
Context:                        rcm
Bind Address:                   10.10.9.81
Chassis State:                  Active
Session State:                  SockActive
Route-Modifier:                 32
RCM Controller Address:         10.10.9.179
```

```
RCM Controller Port:             9200
RCM Controller Connection State: Connected
Ready To Connect:                Yes
Management IP Address:           10.10.14.33
Host ID:                         UPF320
SSH IP Address:                  10.10.14.40 (Activated)
```

Note: The Host ID is not the same as the UPF hostname.

Here you can see the status on RCM OPS Center:

```
[up300-aio-2/rm34] rcm# rcm show-status
message :
{"status":[" Thu Oct 21 10:45:21 UTC 2021 : State is primary"]}
```

```
[up300-aio-2/rm34] rcm# rcm show-statistics controller
message :
{
 "keepalive_version": "65820a54450f930458c01e4049bd01f207bc6204e598f0ad3184c401174fd448",
 "keepalive_timeout": "2s",
 "num_groups": 2,
 "groups": [
   {
     "groupid": 2,
     "endpoints_configured": 7,
     "standby_configured": 1,
     "pause_switchover": false,
     "active": 6,
     "standby": 1,
     "endpoints": [
       {
         "endpoint": "10.10.9.85",
         "bfd_status": "STATE_UP",
         "upf_registered": true,
         "upf_connected": true,
         "upf_state_received": "UpfMsgState_Active",
         "bfd_state": "BFDState_UP",
         "upf_state": "UPFState_Active",
         "route_modifier": 32,
         "pool_received": true,
         "echo_received": 45359,
         "management_ip": "10.10.14.41",
         "host_id": "UPF322",
         "ssh_ip": "10.10.14.44"
       },
       {
         "endpoint": "10.10.9.86",
         "bfd_status": "STATE_UP",
         "upf_registered": true,
         "upf_connected": true,
         "upf_state_received": "UpfMsgState_Active",
         "bfd_state": "BFDState_UP",
         "upf_state": "UPFState_Active",
         "route_modifier": 32,
         "pool_received": true,
         "echo_received": 4518,
         "management_ip": "10.10.14.43",
         "host_id": "UPF317",
         "ssh_ip": "10.10.14.34"
       },
```

```
{
  "endpoint": "10.10.9.94",
  "bfd_status": "STATE_UP",
  "upf_registered": true,
  "upf_connected": true,
  "upf_state_received": "UpfMsgState_Active",
  "bfd_state": "BFDState_UP",
  "upf_state": "UPFState_Active",
  "route_modifier": 32,
  "pool_received": true,
  "echo_received": 4518,
  "management_ip": "10.10.14.59",
  "host_id": "UPF318",
  "ssh_ip": "10.10.14.36"
},
{
  "endpoint": "10.10.9.81",
  "bfd_status": "STATE_UP",
  "upf_registered": true,
  "upf_connected": true,
  "upf_state_received": "UpfMsgState_Active",
  "bfd_state": "BFDState_UP",
  "upf_state": "UPFState_Active",
  "route_modifier": 32,
  "pool_received": true,
  "echo_received": 45359,
  "management_ip": "10.10.14.33",
  "host_id": "UPF320",
  "ssh_ip": "10.10.14.40"
},
{
  "endpoint": "10.10.9.82",
  "bfd_status": "STATE_UP",
  "upf_registered": true,
  "upf_connected": true,
  "upf_state_received": "UpfMsgState_Standby",
  "bfd_state": "BFDState_UP",
  "upf_state": "UPFState_Standby",
  "route_modifier": 50,
  "pool_received": false,
  "echo_received": 4505,
  "management_ip": "10.10.14.35",
  "host_id": "",
  "ssh_ip": "10.10.14.60"
},
{
  "endpoint": "10.10.9.83",
  "bfd_status": "STATE_UP",
  "upf_registered": true,
  "upf_connected": true,
  "upf_state_received": "UpfMsgState_Active",
  "bfd_state": "BFDState_UP",
  "upf_state": "UPFState_Active",
  "route_modifier": 30,
  "pool_received": true,
  "echo_received": 4518,
  "management_ip": "10.10.14.37",
  "host_id": "UPF319",
  "ssh_ip": "10.10.14.38"
},
{
  "endpoint": "10.10.9.84",
  "bfd_status": "STATE_UP",
  "upf_registered": true,
```

```
        "upf_connected": true,
        "upf_state_received": "UpfMsgState_Active",
        "bfd_state": "BFDState_UP",
        "upf_state": "UPFState_Active",
        "route_modifier": 32,
        "pool_received": true,
        "echo_received": 4518,
        "management_ip": "10.10.14.39",
        "host_id": "UPF321",
        "ssh_ip": "10.10.14.42"
      }
    ]
  },
```

Identify Current Standby UPF From RCM OPS Center

From RCM OPS, the Center identifies the UPF in Standby with the use of the **rcm show-statistics controller** command:

```
  {
        "endpoint": "10.10.9.82",
        "bfd_status": "STATE_UP",
        "upf_registered": true,
        "upf_connected": true,
        "upf_state_received": "UpfMsgState_Standby",
        "bfd_state": "BFDState_UP",
        "upf_state": "UPFState_Standby",
        "route_modifier": 50,
        "pool_received": false,
        "echo_received": 4505,
        "management_ip": "10.10.14.35",
        "host_id": "",
       "ssh_ip": "10.10.14.60"
      },
```

Log in to UPF and check RCM information:

```
[local]UPF318# show rcm info
Saturday November 06 13:29:59 UTC 2021
Redundancy Configuration Module:
--------------------------------------------------------------------------------
Context:                          rcm
Bind Address:                     10.10.9.82
Chassis State:                    Standby
Session State:                    SockStandby
Route-Modifier:                   50
RCM Controller Address:           10.10.9.179
RCM Controller Port:              9200
RCM Controller Connection State: Connected
Ready To Connect:                 Yes
Management IP Address:            10.10.14.35
Host ID:
SSH IP Address:                   10.10.14.60 (Activated)
```

Here is the other useful information from RCM OPS Center:

```
[up300-aio-2/rm34] rcm# rcm show-statistics
Possible completions:
 bfdmgr          Show RCM BFDMgr Statistics information
 checkpointmgr   Show RCM Checkpointmgr Statistics information
```

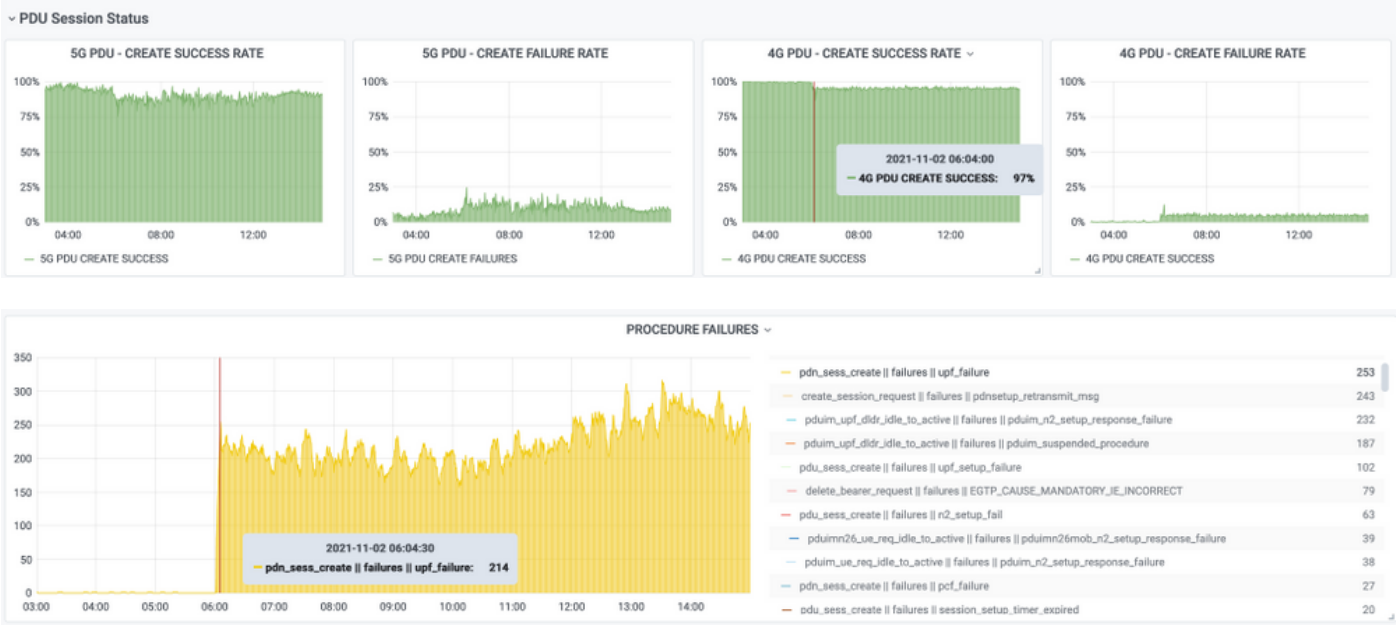
```
configmgr       Show RCM Configmgr Statistics information
controller      Show RCM Controller Statistics information
|               Output modifiers
<cr>
```

Download the RCM guide for Release 21.24.

Problem Reported by RCM Failures on CNDP PODs

The problem was reported on one of the UPFs related to alert UP_SX_SESS_ESTABLISHMENT_SR. This alert says that the Session establishment success rate on the SX interface went down under the configured threshold.

If you look at the Grafana stats, a 5G/4G degradation is observed due to disconnect reason **pdn_sess_create || failures || upf_failure**:



This confirms that the **pdn_sess_create || failures || upf_failure** were caused by UPF419:

```
[local]UPF419# show rcm info
Saturday November 06 14:01:30 UTC 2021
Redundancy Configuration Module:
--------------------------------------------------------------------------------
Context:                          rcm
Bind Address:                     10.10.11.83
Chassis State:                    Active
Session State:                    SockActive
Route-Modifier:                   30
RCM Controller Address:           10.10.11.179
RCM Controller Port:              9200
RCM Controller Connection State: Connected
Ready To Connect:                 Yes
Management IP Address:            10.10.14.165
Host ID:                          DNUD0417
SSH IP Address:                   10.10.14.162 (Activated)
```

On SMF you can check UPF configuration. In this case, you must look for the UPF N4 IP address:

```
[smf/smf2] smf# show running-config profile network-element upf node-id n4-peer-UPF417
```

```
profile network-element upf upf19
 node-id           n4-peer-UPF417
 n4-peer-address ipv4 10.10.10.17
 n4-peer-port      8805
 upf-group-profile upf-group1
 dnn-list          [ internet ]
 capacity          10
 priority          1
exit
```

Then you can perform the Grafana query to identify towards what UPF N4 address there are most failures:

Grafana Query:
sum(increase(proto_udp_res_msg_total{namespace=~"$namespace", message_name="session_establishment_res", status="no_rsp_received_tx"} [15m])) by (message_name, status, peer_info)

Label: {{message_name}} || {{status}} || {{peer_info}}

Grafana must show where failures happen. In the example, it is related to UPF419.

When you connect to the system, you can confirm the sessmgr was not set properly after the RCM switchover because many of the session managers are not in the expected 'Actv Ready' state.

```
[local]UPF419# show srp checkpoint statistics verbose
Tuesday November 02 17:24:01 UTC 2021
 smgr     state  peer    recovery  pre-alloc  chk-point rcvd   chk-point sent
 inst            conn    records   calls      full     micro   full     micro
 ----   -------  -----   -------   --------   -----    -----   -----     ----
    1      Actv Ready         0          0    1108    34001   14721   1200158
    2      Actv Ready         0          0    1086    33879   17563   1347298
    3      Actv Ready         0          0    1114    34491   15622   1222592
     4       Actv  Conn         0          0       5      923       0         0
    5      Actv Ready         0          0    1106    34406   13872   1134403
     6       Actv  Conn         0          0       5      917       0         0
     7       Actv  Conn         0          0       5      920       0         0
     8       Actv  Conn         0          0       1      905       0         0
     9       Actv  Conn         0          0       5      916       0         0
    10       Actv  Conn         0          0       5      917       0         0
   11      Actv Ready         0          0    1099    34442   13821   1167011
    12       Actv  Conn         0          0       5      916       0         0
    13       Actv  Conn         0          0       5      917       0         0
   14      Actv Ready         0          0    1085    33831   13910   1162759
   15      Actv Ready         0          0    1085    33360   13367   1081370
    16       Actv  Conn         0          0       4      921       0         0
   17      Actv Ready         0          0    1100    35009   13789   1138089
   18      Actv Ready         0          0    1092    33953   13980   1126028
    19       Actv  Conn         0          0       5      916       0         0
    20       Actv  Conn         0          0       5      918       0         0
   21      Actv Ready         0          0    1098    33521   13636   1108875
   22      Actv Ready         0          0    1090    34464   14529   1263419
```

Solution

This is related to Cisco Defect Tracking System (CDETS) CSCvz9749. The fix was integrated into 21.22.ua4.82694 and later.

Workaround

On UPF419, you must restart the session manager instances that were not in **Actv Ready** with hidden command **task kill facility sessmgr instance <>** and this resolves the situation.

```
[local]UPF419# show srp checkpoint statistics verbose
Wednesday November 03 16:44:57 UTC 2021
 smgr     state  peer    recovery  pre-alloc  chk-point rcvd   chk-point sent
 inst            conn    records   calls      full     micro   full     micro
 ----   -------  -----   -------   --------   -----    -----   -----     ----
    1      Actv Ready         0          0    1108    34001   38319   2267162
    2      Actv Ready         0          0    1086    33879   40524   2428315
    3      Actv Ready         0          0    1114    34491   39893   2335889
    4      Actv Ready         0          0       0        0   12275   1049616
    5      Actv Ready         0          0    1106    34406   37240   2172748
    6      Actv Ready         0          0       0        0   13302   1040480
    7      Actv Ready         0          0       0        0   12636   1062146
    8      Actv Ready         0          0       0        0   11446   976169
    9      Actv Ready         0          0       0        0   11647   972715
   10      Actv Ready         0          0       0        0   11131   950436
   11      Actv Ready         0          0    1099    34442   36696   2225847
   12      Actv Ready         0          0       0        0   10739   919316
   13      Actv Ready         0          0       0        0   11140   970384
   14      Actv Ready         0          0    1085    33831   37206   2226049
   15      Actv Ready         0          0    1085    33360   38135   2225816
   16      Actv Ready         0          0       0        0   11159   946364
   17      Actv Ready         0          0    1100    35009   37775   2242427
   18      Actv Ready         0          0    1092    33953   37469   2181043
   19      Actv Ready         0          0       0        0   13066   1055662
   20      Actv Ready         0          0       0        0   10441   938350
   21      Actv Ready         0          0    1098    33521   37238   2165185
   22      Actv Ready         0          0    1090    34464   38227   2399415
```

Logs to Collect in Case of UPF Failure that Causes a Switchover

> **Note**: Ensure debug logs are enabled in RCM (request approval before you enable any debug log). Refer to logging recommendations.

RCM ops-center Logging Level

```
logging level application debug
logging level transaction debug
logging level tracing off
logging name infra.config.core level application warn
logging name infra.config.core level transaction warn
logging name infra.resource_monitor.core level application warn
logging name infra.resource_monitor.core level transaction warn
```

Step by Step Data Collection

1. Summary of the issue: The problem statement must be clear. Indicate the problematic **node name/ip** so that it is easier to find the necessary information from the logs. For example, in case of a switchover issue, it is helpful if it is mentioned that IP x.x.x.x is the source UPF and

x.x.x.y is the destination UPF.

2. If there are multiple ways to reproduce the problem, mention those.
3. RCM version information: In the case of RCM VM deployment from RCM VM, cat **/etc/smi/rcm-image-versionshow helm** from the ops-center. In the case of RCM CN deployment **show helm** from the ops center.
4. RCM Tac debug CN or RCM logs at the time of the occurrence of the issue. In some cases, you can also require logs from the start when the POD had just come up.
5. Indicate which RCM is primary or backup. In the case of CN, share the information for both RCM pairs.
6. Share the running configuration from RCM ops-center from all the instances.
7. Collect the RCM SNMP traps.
8. Irrespective of switchover failure or not, it is better to collect one active UP SSD and one standby UP SSD.
9. RCM controller, configmgr, checkpoint manager, switchover, and switchover-verbose statistics commands are used to mention the exact CLI.
   rcm show-statistics controller
   rcm show-statistics configmgr
   rcm show-statistics checkpointmgr
   rcm show-statistics switchover
   rcm show-statistics switchover-verbose
10. Syslogs of UPF or RCM.
11. If the issue is related to switchover failure, a new active UPF SSD and old UPF active SSD are required. In some cases, old actives reboot due to switchover. In that case, you must reproduce the issue, and just before that you need to collect the old active UP SSD.
12. In a switchover failure case, it is also helpful to collect the vpn, sessmgr, sess-gr, and sxdemux debug logs from old and new actives at the issue reproduction.
    logging filter active facility sxdemux level debug
    logging filter active facility sessmgr level debug
    logging filter active facility sess-gr level debug
    logging filter active facility vpn level debug
13. Vpnmgr/Sessmgr cores are needed in case of error/problem in sessmgr/vpnmgr.
    The sessmgr_instance_id is the instance where problem is noticed. **vpnmgr_instance_id** is the context # of the RCM context.
    task core facility sessmgr instance <sessmgr_instance_id>
    task core facility vpnmgr instance <vpnmgr_instance_id>
14. In case of RCM HA issue, share the RCM TAC debug/pod logs from both the instances.

Related Information

- https://www.cisco.com/c/en/us/support/wireless/ultra-cloud-core-user-plane-function/products-installation-and-configuration-guides-list.html
- Technical Support & Documentation - Cisco Systems