CPS vDRA Troubleshooting Guide, Release 18.4.0 (Restricted Release)

First Published: 2018-09-14

Last Modified: 2018-09-14

C O N T E N T S

Preface

-
-
-
-
-
-

About This Guide

This document is a part of the Cisco Policy Suite documentation set.

For information about available documentation, see the *CPS Documentation Map* for this release at Cisco.com.

Audience

This guide is best used by these readers:

- Network administrators
- Network engineers
- Network operators
- System administrators

This document assumes a general understanding of network architecture, configuration, and operations.

Additional Support

For further documentation and support:

- Contact your Cisco Systems, Inc. technical representative.
- Call the Cisco Systems, Inc. technical support number.
- Write to Cisco Systems, Inc. at support@cisco.com.

- Refer to support matrix at https://www.cisco.com/c/en/us/support/index.html and to other documents related to Cisco Policy Suite.

Conventions (all documentation)

This document uses the following conventions.

Conventions	Indication
bold font	Commands and keywords and user-entered text appear in **bold** font.
italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
[ ]	Elements in square brackets are optional.
{x	y	z }	Required alternative keywords are grouped in braces and separated by vertical bars.
[x	y	z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
courier font	Terminal sessions and information the system displays appear in courier font.
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Note Means reader take note. Notes contain helpful suggestions or references to material not covered in the manual.

Caution Means reader be careful. In this situation, you might perform an action that could result in equipment damage or loss of data.

Warning   IMPORTANT SAFETY INSTRUCTIONS.

Means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device.

SAVE THESE INSTRUCTIONS

Note   Regulatory: Provided for additional information and to comply with regulatory and customer requirements.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see What's New in Cisco Product Documentation.

To receive new and revised Cisco technical content directly to your desktop, you can subscribe to the . RSS feeds are a free service.

Important Notes

Important   Any feature or GUI functionality that is not documented may not be supported in this release or may be customer specific, and must not be used without consulting your Cisco Account representative.

RESTRICTED RELEASE

☞

Important This is a Short Term Support (STS) release with availability and use restrictions. Contact your Cisco Account or Support representatives for more information.

Troubleshooting CPS vDRA

Overview

CPS vDRA is a functional element that ensures that all Diameter sessions established over Gx, Rx interfaces and for unsolicited application reporting, the Sd interface for a certain IP-CAN session reach the same PCRF or destined PCRF when multiple and separately addressable PCRFs have been deployed in a Diameter realm.

General Troubleshooting

Run the following command in CLI to view the diagnostics status. Verify that the status of all the nodes is in passing state.

```
admin@orchestrator[master-0]# show system diagnostics status
```

Run the following command in CLI to view the docker engines status. Verify that all docker engines are in CONNECTED state.

```
admin@orchestrator[master-0]# show docker engine
```

System Maintenance Procedures

Backup Procedures

Back up CLI Configuration

Back up the CLI configuration of APP VNF and DB VNF. Then copy the backups to an external server. The following sections describe the commands for APP VNF and DB VNF.

DRA VNF

The following commands saves each configuration as a separate file in the system.

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# config
admin@orchestrator# show running-config binding | save
/data/config/binding_cli_backup
admin@orchestrator# show running-config license | save
/data/config/license_cli_backup
admin@orchestrator# show running-config network virtual-service
| save /data/config/vip_cli_backup
admin@orchestrator# show running-config alert snmp-v2-destination
| save /data/config/alert_snmp-v2_cli_backup
admin@orchestrator# show running-config alert rule | save
/data/config/alert_rule_cli_backup
admin@orchestrator# show running-config external-aaa | save
/data/config/external-aaa_cli_backup
admin@orchestrator# show running-config ntp | save
/data/config/ntp_backup
admin@orchestrator# show running-config aaa authentication users
user | save /data/config/aaa_users_backup
admin@orchestrator# show running-config nacm groups group |
save /data/config/nacm_groups_backup
```

Copy the backup of the CLI configs to an external server.

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cd /data/orchestrator/config
cps@${DRM-hostname}:~$ scp -i /home/cps/cps.pem *_backup
<user>@<external-server>:<external-folder>
```

DB VNF

The following commands saves each configuration as a separate file in the system.

```
# node: DRA DB Master
# user: cps
cps@${DBM-hostname}:~$ cli
admin@orchestrator# config
admin@orchestrator# show running-config binding |
save /data/config/database_cli_backup
admin@orchestrator# show running-config license |
save /data/config/license_cli_backup
admin@orchestrator# show running-config network
virtual-service | save /data/config/vip_cli_backup
```

```
admin@orchestrator# show running-config alert snmp-v2-destination
 | save /data/config/alert_snmp-v2_cli_backup
admin@orchestrator# show running-config alert rule |
save /data/config/alert_rule_cli_backup
admin@orchestrator# show running-config external-aaa
| save /data/config/external-aaa_cli_backup
admin@orchestrator# show running-config ntp | save
/data/config/ntp_backup
```

Copy the backup of the CLI configs to an external server.

```
# node: DRA DB Master
# user: cps
cps@${DBM-hostname}:~$ cd /data/orchestrator/config
cps@${DBM-hostname}:~$ scp -i /home/cps/cps.pem *_backup
 <user>@<external-server>:<external-folder>
```

Back up Policy Builder

Export the CPS service configuration to a single file.

1. Open DRA Central GUI : https://<master ip>/central/dra

2. Click **Import/Export** under Policy Builder.

3. Select/enter the following details:

   - Export Type

   - Export URL

   - Export File Prefix

   - Use zip file extension

4. Click **Export**.

5. Save the ZIP file.

Back up CRD

Back up the CRD data to a single file.

For more information, see .

1. Open DRA Central GUI : https://<master ip>/central/dra

2. Click **Custom Reference Data** under Custom Reference Data.

3. Select/enter the following details under **Export**:

   - Use zip file extension

4. Click **Export**.

5. Save the ZIP file.

Shutting Down CPS

Shut down DRA VNF

1. Use the following command to shut down the application processes in DRA VNF:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# system stop
```

2. Run the following command to verify that the system status running is "false".

```
admin@orchestrator# show system status
```

3. Use the following command to verify that only the infrastructure items are running:

```
admin@orchestrator# show scheduling status
```

Shut down DB VNF

1. Use the following command to shut down the application processes in DRA DB VNF:

```
# node: DRA DB Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# system stop
```

2. Run the following command to verify that the system status running is "false".

```
admin@orchestrator# show system status
```

3. Use the following command to verify that only the infrastructure items are running:

```
admin@orchestrator# show scheduling status
```

Starting up CPS

Use the following commands to start up the system after a maintaince window is completed and the VMs are powered on.

Start up DRA VNF

Use the following command to start the application processes in DRA VNF:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# system start
```

Start DB VNF

Use the following command to start the application processes in DRA DB VNF:

```
# node: DRA DB Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# system start
```

Post Power up VM Health Check

Perform a health check on both VNFs after the maintenance window is complete and the VMs are powered on. For more information, see .

Diameter Troubleshooting and Connections

For messages belonging to particular interface, CPS vDRA should be ready to make diameter connection on the configured application port. As CPS vDRA acts as a server, it should be listening on ports for different applications to accept any incoming diameter requests for the application.

If you are facing problems making diameter connections, check for the following configuration:

DRA Plug-in Configuration in DRA Policy Builder (PB)

Figure 1: DRA Endpoints

Dra Endpoints

*Vm Host Name	*Ip Address	*Base Port	*Realm	*Fqdn	*Enabled	*Application
lb01	80.80.80.10	3868	dra.cisco.com	dra	✓	Gx
lb01	80.80.80.10	4868	gx-dra2.cisco.com	gx-dra2	✓	Gx

Step 1 Check status of application base port on active policy director (lb). It should be listening to diameter connections externally on VIP and internally to Policy Servers (QNS).

```
[root@lb01 ~]# netstat -na | grep 3868
tcp        0      0 10.77.207.100:3868          0.0.0.0:*                   LISTEN
tcp        0      0 ::ffff:80.80.80.10:3868     :::*                       LISTEN
```

Step 2 Check `haproxy-diameter.cfg` file for proper entries:

For and configuration, the entries should be as follows:

```
[root@lb01 ~]# cat /etc/haproxy/haproxy-diameter.cfg
global
  daemon
  nbproc         1           # number of processing cores
  stats socket /tmp/haproxy-diameter
defaults
  timeout client    60000ms       # maximum inactivity time on the client side
  timeout server   180000ms       # maximum inactivity time on the server side
  timeout connect    5000ms       # maximum time to wait for a connection attempt to a server to
succeed
log            127.0.0.1       local1 err
listen diameter-int1
       bind 10.77.207.100:3868
       mode tcp
       option tcpka
       balance leastconn
       server lb01-A lb01:3868 check
       server lb01-B lb01:3869 check
       server lb01-C lb01:3870 check
listen diameter-int2
```

```
        bind 10.77.207.100:4868
        mode tcp
        option tcpka
        balance leastconn
        server lb01-A lb01:4868 check
        server lb01-B lb01:4869 check
        server lb01-C lb01:4870 check
listen stats_proxy_diameter lbvip01:5540
        mode http
        option httpclose
        option abortonclose
        # enable web-stats
        stats enable
        stats uri /haproxy-diam?stats
        #stats auth        haproxy:cisco123
        stats refresh     60s
        stats hide-version
```

Step 3   Listen for diameter traffic by logging into lb01 and lb02 and execute the following command:

```
tcpdump -i any port 3868 -s 0 -vv
```

Troubleshooting Basics

Troubleshooting CPS vDRA consists of these types of basic tasks:

- Gathering Information

- Collecting Logs

- Running Traces

Diameter Error Codes and Scenarios

Table 1: Diameter Error Codes and Scenarios

Result-Code	Result-Code Value	Description
Informational		
DIAMETER_MULTI_ROUND_AUTH	1001	Subsequent messages triggered by client shall also used in Authentication and to get access of required resources. Generally used in Diameter NAS.
Success		
DIAMETER_SUCCESS	2001	Request processed Successfully.
DIAMETER_LIMITED_SUCCESS	2002	Request is processed but some more processing is required by Server to provide access to user.

Result-Code	Result-Code Value	Description
Protocol Errors [E-bit set]		
DIAMETER_COMMAND _UNSUPPORTED	3001	Server returns it if Diameter Command-Code is un-recognized by server.
DIAMETER_UNABLE _TO_DELIVER	3002	Message cannot be delivered because there is no Host with Diameter URI present in Destination-Host AVP in associated Realm.
DIAMETER_REALM_NOT _SERVED	3003	Intended Realm is not recognized.
DIAMETER_TOO_BUSY	3004	Shall return by server only when server unable to provide requested service, where all the pre-requisites are also met. Client should also send the request to alternate peer.
DIAMETER_LOOP_DETECTED	3005	-
DIAMETER_REDIRECT _INDICATION	3006	In Response from Redirect Agent.
DIAMETER_APPLICATION _UNSUPPORTED	3007	-
DIAMETER_INVALID_HDR_BITS	3008	It is sent when a request is received with invalid bits combination for considered command-code in DIAMETER Header structure. For example, Marking Proxy-Bit in CER message.
DIAMETER_INVALID_AVP_BITS	3009	It is sent when a request is received with invalid flag bits in an AVP.
DIAMETER_UNKNOWN_PEER	3010	A DIAMETER server can be configured whether it shall accept DIAMETER connection from all nodes or only from specific nodes. If it is configured to accept connection from specific nodes and receives CER from message from any node other than specified.
Transient Failures [Could not satisfy request at this moment]		
DIAMETER_AUTHENTICATION _REJECTED	4001	Returned by Server, most likely because of invalid password.

Result-Code	Result-Code Value	Description
DIAMETER_OUT_OF_SPACE	4002	Returned by node, when it receives accounting information but unable to store it because of lack of memory.
ELECTION_LOST	4003	Peer determines that it has lost election by comparing Origin-Host value received in CER with its own DIAMETER IDENTITY and found that received DIAMETER IDENTITY is higher.
Permanent Failures [To inform peer, request is failed, should not be attempted again]		
DIAMETER_AVP _UNSUPPORTED	5001	AVP marked with Mandatory Bit, but peer does not support it.
DIAMETER_UNKNOWN _SESSION_ID	5002	-
DIAMETER_AUTHORIZATION _REJECTED	5003	User can not be authorized. For example, Comes in AIA on s6a interface.
DIAMETER_INVALID_AVP_VALUE	5004	-
DIAMETER_MISSING_AVP	5005	Mandatory AVP in request message is missing.
DIAMETER_RESOURCES _EXCEEDED	5006	A request was received that cannot be authorized because the user has already expended allowed resources. An example of this error condition is a user that is restricted to one dial-up PPP port, attempts to establish a second PPP connection.
DIAMETER_CONTRADICTING _AVPS	5007	Server has identified that AVPs are present that are contradictory to each other.
DIAMETER_AVP_NOT_ALLOWED	5008	Message is received by node (Server) that contain AVP must not be present.
DIAMETER_AVP_OCCURS _TOO_MANY_TIMES	5009	If message contains the a AVP number of times that exceeds permitted occurrence of AVP in message definition.
DIAMETER_NO_COMMON _APPLICATION	5010	In response of CER if no common application supported between the peers.
DIAMETER_UNSUPPORTED _VERSION	5011	Self explanatory.

Result-Code	Result-Code Value	Description
DIAMETER_UNABLE _TO_COMPLY	5012	Message rejected because of unspecified reasons.
DIAMETER_INVALID_BIT _IN_HEADER	5013	When an unrecognized bit in the Diameter header is set to one.
DIAMETER_INVALID _AVP_LENGTH	5014	Self explanatory.
DIAMETER_INVALID _MESSAGE_LENGTH	5015	Self explanatory.
DIAMETER_INVALID_AVP _BIT_COMBO	5016	For example, marking AVP to Mandatory while message definition doesn't say so.
DIAMETER_NO_COMMON _SECURITY	5017	In response of CER if no common security mechanism supported between the peers.

Policy DRA Error Codes

Non-compliant Diameter requests are checked for errors in routing AVP and P-bits. The following table describes the error codes and the reasons for errors in Diameter requests:

Table 2: Policy DRA Error Codes

Policy DRA Error String	Error Code	Sub-code	Description
No application route found	3002	001	Route List Availability Status is "Unavailable"
Timeout triggered	3002	002	Timeout triggered
No peer group	3002	003	No peer group
Session DB Error	3002	004	Session DB Error
Binding DB Error	3002	005	Binding DB Error
No key for binding lookup	3002	006	No key for binding lookup
Binding not found	3002	007	Binding not found
Message loop detected	3002	008	Message loop detected
Parsing exception with message	3002	009	Parsing exception with message
CRD DB Error	3002	010	CRD DB Error

Policy DRA Error String	Error Code	Sub-code	Description
Retries exceeded	3002	011	Retries exceeded
No peer route	3002	012	No peer routing rule found for a Realm-only or non-peer Destination-Host
P-bit not set	3002	013	P-bit in the Request message is set to "0"
Missing Origin-Host AVP	5005	014	Mandatory Origin-Host AVP missing
Missing Origin-Realm AVP	5005	015	Mandatory Origin-Realm AVP missing
Missing Destination-Realm AVP	5005	016	Mandatory Destination-Realm AVP missing
No avp found in request for SLF lookup type	3002	101	No avp found in request for SLF lookup type
SLF DB Error	3002	102	SLF DB Error
SLF credential not found in DB	3002	103	SLF credential not found in DB
SLF Destination type not found in DB	3002	104	SLF Destination type not found in DB
Destination not found in SLF Mapping Table	3002	105	Destination not found in SLF Mapping Table

Default HTTP Error Codes

You can configure the HTTP response error code (such as 4xx, 5xx) corresponding to each vDRA Rest API JSON error response code for the GET binding (for example imsi, imsiApn, msisdn, msisdnApn, ipv4, ipv6). For more information about the CRD, see the *CPS vDRA Configuration Guide*.

If you do not configure the Rest API HTTP Error Code in the CRD, vDRA uses the default HTTP error codes for GET binding Rest API.

The following table lists the default HTTP error codes:

Table 3: Default HTTP Error Codes

vDRA Rest API Error Code	HTTP Error Code	HTTP Reason-Phrase
1001 (INTERNAL_ERROR)	500	Internal Server Error
2014 (DATA_NOT_FOUND)	404	Not Found

vDRA Rest API Error Code	HTTP Error Code	HTTP Reason-Phrase
2019 (INVALID_API_FORMAT)	400	Bad Request

Debug ping / ping6

Run the following commands to check ping connectivity from the VM to other nodes using IPv4 and IPv6:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# debug ping <wtc2b1fdrd02v> -n <IPv4 address>
admin@orchestrator# debug ping6 <wtc2b1fdrd02v> -n <IPv6 address>
```

Where:

- -n:

Debug traceroute

Run the following commands to check traceroute connectivity from the VM to other nodes:

IPv4:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# debug traceroute <VMHOST> <IPv4address>
```

IPv6:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# debug traceroute <VMHOST> -6 <IPv6address>
```

Debug tcpdump

Use the following command to get packet capture from the VM. Specify interface and port details to avoid big packet capture files.

If you use the `-i any` option, you may see the same packet twice: once as it traverses the VMs interface, and again when it traverses the Docker container's virtual interface.

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# debug tcpdump wtc2b1fdrd01v test.pcap
60s -s 0 -i ens162 port 3868
admin@orchestrator# debug packet-capture gather directory test_debug
admin@orchestrator# debug packet-capture purge
```

You can download the packer capture file from : https://<master ip>/orchestrator/downloads/ after logging in to https://<master ip>/

After you download the file, delete the packet capture files to clean up the disk space.

Monitor Application Logs

Use the following commands to monitor application logs :

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# monitor log application
```

Debug Tech to Capture Logs

Run the following command to capture SVN, CRD, logs, and save it at http://<master ip>/orchestrator/downloads/debug/tech/:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# debug tech
```

Monitor Container Logs

Use the following command to monitor specific container logs:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# monitor log container <container-name>
```

Monitor Orchestrator Logs

Use the following command to monitor orchestrator logs during an upgrade/downgrade:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# monitor log container orchestrator
| include AUDIT
```

If the CLI is not accessible or is giving errors when executing commands, use the following command from the master VM for more information:

```
cps@${DRM-hostname}:~$ docker logs orchestrator
```

Change CLI User Password

If you know the existing password, use the following steps to change the user password in CLI:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# aaa authentication users user fpasapi change-password
Value for 'old-password' (<string>): *****
Value for 'new-password' (<string>): ********
Value for 'confirm-password' (<string>): ********
```

If you do not know the password, use the following commands to reset the password:

```
# node: DRA Master
# user: cps
```

```
cps@${DRM-hostname}:~$ cli
admin@orchestrator# config
admin@orchestrator(config)# aaa authentication users user fpasapi gid 100
 uid 9000 homedir "" ssh_keydir "" password <password>
admin@orchestrator(config-user-apiuser)# commit
Commit complete.
admin@orchestrator(config-user-apiuser)# end
```

Restart Docker Container

If the commands `show docker service` or `system diagnostics` show errors, check the docker service for any unhealthy processes. If there are unhealthy processes, use the command `monitor container logs` to view logs and then restart the docker container.

```
Action
# node: DRA Master / DB Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show docker service | tab | exclude HEALTHY
admin@orchestrator# show system diagnostics | tab | exclude passing
# container-name is unhealthy process container id.
admin@orchestrator# docker restart container-id <container-name>
```

Check DNS Config

Check the VMs dnsmasq file to verify whether the DNS entries are present; if not, perform the following steps:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cat /data/dnsmasq/etc/dnsmasq.conf
# If DNS entries are missing, perform the following steps:
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show running-config network dns |
save /data/config/dns_cli_backup
admin@orchestrator# config
admin@orchestrator(config)# no network dns
admin@orchestrator(config)# commit
admin@orchestrator(config)# end
admin@orchestrator# config
admin@orchestrator(config)# load merge /data/config/dns_cli_backup
admin@orchestrator(config)# commit
admin@orchestrator(config)# end
admin@orchestrator# exit
cps@${DRM-hostname}:~$ cat /data/dnsmasq/etc/dnsmasq.conf
```

Redeploy Master VM

When the master VM is deleted or redeployed for some reason, you must make it part of the existing cluster. workaround to make it part of the cluster as described in the following steps:

```
# node: DRA Master
# user: cps
# After the master VM is redeployed, log into the master VM, and wait til
# cpsinstall is complete
cps@${DRM-hostname}:~$ journalctl -u cpsinstall.service -f
# Verify that the following log apperas: log <date time stamp> master-0
bootstrap.sh[1521]: Install script completed.
```

```
# Once cpsinstall is finished; execute the following
# commands on the master VM in the order specified.
cps@${DRM-hostname}:~$ docker stop $(docker ps -a -q)
cps@${DRM-hostname}:~$ docker rm $(docker ps -a -q)
cps@${DRM-hostname}:~$ weave launch-router --ipalloc-init consensus=3
cps@${DRM-hostname}:~$ sudo rm -rf /data/orchestrator
cps@${DRM-hostname}:~$ sudo rm /var/cps/bootstrap-status
cps@${DRM-hostname}:~$ sudo /root/bootstrap.sh
cps@${DRM-hostname}:~$ ssh-keygen -f "/home/cps/.ssh/known_hosts" -R
[localhost]:2024
```

Remove MongoDB Replica Set Member

Perform the following steps to remove a replica set member from MongoDB.

⚠

Caution The command `no database cluster` deletes the configuration completely, so ensure the information is correct.

```
# node: DRA Master
# user: cps
cps@${DBM-hostname}:~$ cli
cps@${DBM-hostname}:~$ config
admin@orchestrator(config)# no database cluster binding shard
binding-shard-1 shard-server fn6-1a1bs1k
admin@orchestrator(config)# commit
admin@orchestrator(config)# end
#connect to the replica set primary member container to remove the node, take a note
#of port of the replica set
cps@${DBM-hostname}:~$ docker connect mongo-s104
root@mongo-s104:/# mongo --port 27033
rs-binding-shard-1:PRIMARY> rs.status()
#Note the name of the member from rs status output and then input it to
#rs.remove to remove the member
rs-binding-shard-1:PRIMARY> rs.remove
("[2606:ae00:2001:2420:8000::9]:27034")
```

Clean the Database

Perform the following steps if you want to clean the database and recreate a fresh database.

⚠

Warning All the data will be lost.

```
# node: DRA Master
# user: cps
cps@${DBM-hostname}:~$ cli
# Stop all the application process:
cps@${DBM-hostname}:~$ system stop
# Wait for some time till all the application proceses stop.
# You can check the process using the commands:
# show scheduling status and show system status
# Repeat the following steps in all the database nodes
cps@${DBM-hostname}:~$ rm -rf /data/configdb/*
cps@${DBM-hostname}:~$ rm -rf /data/db/*
cps@${DBM-hostname}:~$ rm -rf /mmapv1-tmpfs-<port>/*
cps@${DBM-hostname}:~$ cli
```

```
# Restart the system:
cps@${DBM-hostname}:~$ system start
```

Reset the CLI Configuration

Perform the following steps to reset the CLI configuration:

⚠️

Caution   The complete configuration will be reset.

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ docker exec -it orchestrator bash
cps@${DRM-hostname}:~$ /var/confd/bin/confd_load -D -m -l
/data/cdb//*.xml
```

Policy DRA Logger Levels

Policy DRA Application logs are available for debugging purposes.

Note that turning on logs in a production system can have a substantial impact on the system performance and is not recommended.

Enabling and Disabling Logs

Use the orchestrator CLI to enable and disable application logs.

```
admin@orchestrator# logger set ?
Possible completions:
  <logger name>

admin@orchestrator# logger set com.broadhop.dra.service ?
Possible completions:
  debug  error  info  off  trace  warn

admin@orchestrator# logger clear com.broadhop.dra.service ?
Possible completions:
  |  <cr>
```

View Log Levels

The different log levels in the order of increasing details in the log are:

- Error (error logs)

- Warn (error and warning logs)

- Info

- Debug

- Trace (all logs)

The default log level is warn.

Use the following orchestrator CLI command to view the current log levels set on a per application module basis.

```
admin@orchestrator# show logger level
Logger                          Current Level
-------------------------------------------------
com.broadhop.dra.service            warn
dra.trace                 warn
org.jdiameter               warn
```

View Logs

To view application logs continuously similar to the `tail -f` command, use the following command:

```
"monitor log application"
```

To view application logs that were previously collected in a consolidated log file (similar to the `more` command), use the following command:

```
show log application
```

Common Loggers

The following table describes the different loggers and their default log level:

Table 4: Common Loggers

Logger Name	Description	Default Log Level
com.broadhop.dra.service	Policy DRA application logs. This displays logs from various modules of the Policy DRA system.	warn
dra.trace	Policy DRA audit logs. This displays a summary of the Diameter message request and response processing.	warn
org.jdiameter	jDiameter module logs. This displays logs from various modules of the jDiameter module.	warn
com.broadhop.dra.session. expiration.service	Checks and deletes stale sessions.	warn
com.broadhop.dra.service .stack	vDRA stack-related logs to enable debugging at stack level. To be used with org.jdiameter log level.	warn
com.broadhop.dra.service. bindings.impl	This logger provides logs about the binding and API handling operations managed by the Worker.	warn

Logger Name	Description	Default Log Level
com.mongodb	Logging related to MongoDB library as the Worker invokes MongoDB API for database operations.	warn
com.broadhop.dra.service .routing	vDRA routing-related messages to debug issues in routing.	warn
com.broadhop.dra.service .control	vDRA logs related to control messaging.	warn

Common Troubleshooting Steps

CPS vDRA Logs

Step 1   Use the following command in CLI to view the consolidated application logs.

```
admin@orchestrator[master-0]# show log application
```

Step 2   Use the following command in CLI to view the consolidated engine logs.

```
admin@orchestrator[master-0]# show log engine
```

Counters and Statistics

Check for statistics generated at pcrfclient01/02 in `/var/broadhop/stats` and counters in beans at jmx terminal.

System Health Checks

View System Status

Use the following command to view the system status and verify whether the system is running, or if any upgrade or downgrade is in progress, and whether it is 100% deployed.

APP VNF

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show system status
```

DB VNF

```
# node: DRA DB Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show system status
```

If system is not 100% deployed, use the following command to view the current scheduling status: `system scheduling status`

View System Diagnostics

Use the following command to view the system disgnostics and debug failed processes.

APP VNF

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show system diagnostics | tab | exclude passing
```

DB VNF

```
# node: DRA DB Master
# user: cps
cps@ ${DBM-hostname}:~$ cli
admin@orchestrator# show system software | tab
admin@orchestrator# show system diagnostics | tab | exclude passing
```

You can monitor the log of the container using the command: `monitor container logs`

Check System Scheduling Status

Use the following command to verify the installer scheduler status. The scheduler must reach `haproxy-int-api 1 500` and all states indicate running.

APP VNF

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show scheduling status
```

DB VNF

```
# node: DRA DB Master
# user: cps
cps@${DBM-hostname}:~$ cli
admin@orchestrator# show scheduling status
```

Check Docker Engine

Use the following commands to check the docker engine:

- `show docker engine | tab`: Check docker engine connected status to verify whether all VM engines are connected.

- `show running-config docker | tab`: Check the running configuration of the docker to verify whether all VMs are registered to the Master VM correctly and whether all VMs are shown with internal IP and scheduling slots.

APP VNF

Command:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show docker engine | tab
```

Command:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show running-config docker | tab
```

DB VNF

Command:

```
# node: DRA DB Master
# user: cps
cps@${DBM-hostname}:~$ cli
admin@orchestrator# show docker engine | tab
```

Command:

```
# node: DRA DB Master
# user: cps
cps@${DBM-hostname}:~$ cli
admin@orchestrator# show running-config docker | tab
```

Check Docker Service

Use the following commands to check the docker service:

- `show docker service | tab`: to verify whether all the docker services are running.

- `show docker service | tab | exclude HEALTHY` : to view unhealthy docker services.

APP VNF

Command:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show docker service | tabb
```

Command:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show docker service | tab | exclude HEALTHY
```

DB VNF

Command:

```
# node: DRA DB Master
# user: cps
cps@${DBM-hostname}:~$ cli
admin@orchestrator# show docker service | tab
```

Command:

```
# node: DRA DB Master
# user: cps
cps@${DBM-hostname}:~$ cli
admin@orchestrator# show docker service | tab | exclude HEALTHY
```

View Alert Status

Check the alert status in both VNFs and verify that there are no issues.

APP VNF

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show alert status | tab | include firing
```

DB VNF

```
# node: DB Master
# user: cps
cps@${DBM-hostname}:~$ cli
admin@orchestrator# show alert status | tab | include firing
```

Troubleshooting Application

Call Failures

In case of call failures, check the Peer Connection, Binding Monitoring, Peer Errors, Error Result Code in Central GUI as described:

1. Log into the Central GUI as admin.

    In the Peer Monitoring, filter by the host where call failures are observed.

    If there is any problem with connection; that peer is not listed in Active Peer Endpoints screen and is listed in Inactive peers.

Figure 2: Peer Monitoring - Active Peer Endpoints

Figure 3: Peer Monitoring - Inactive Peer Endpoints
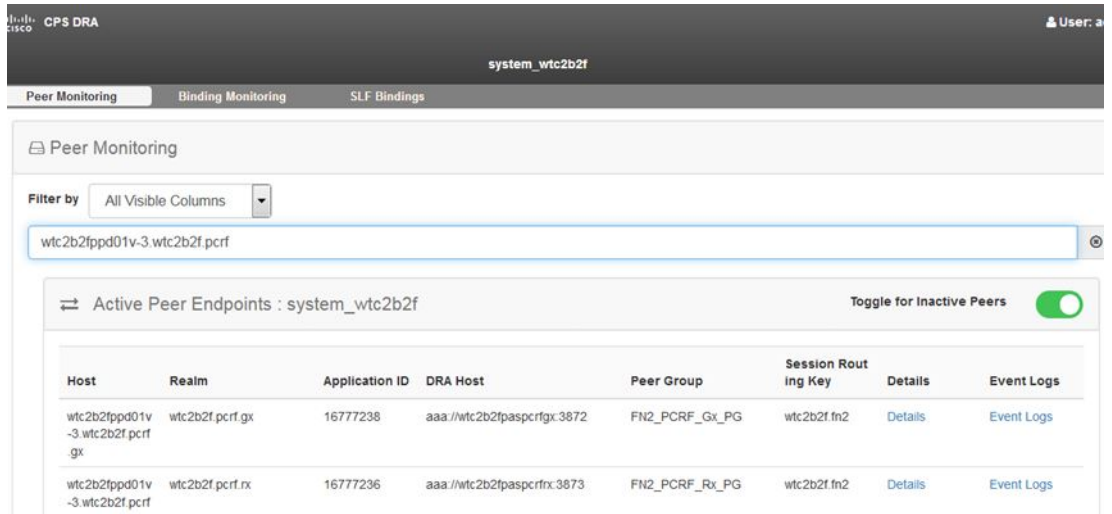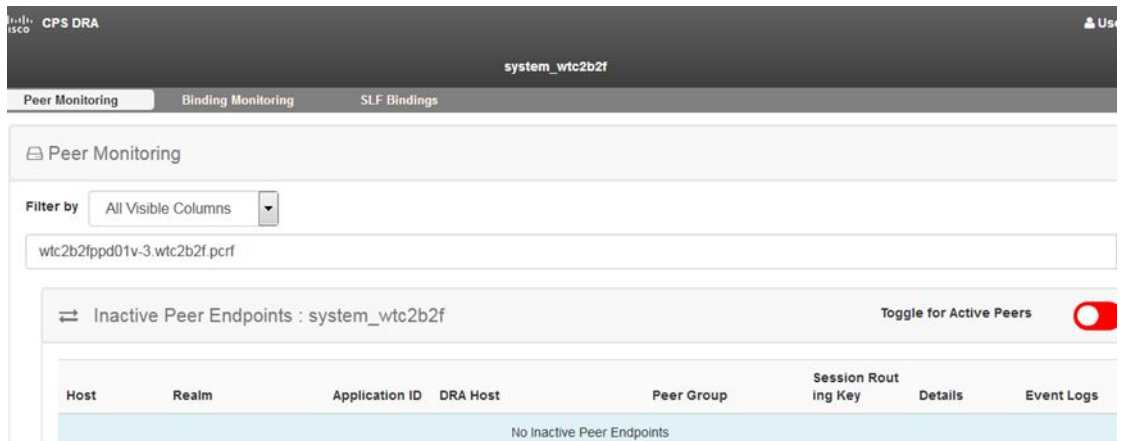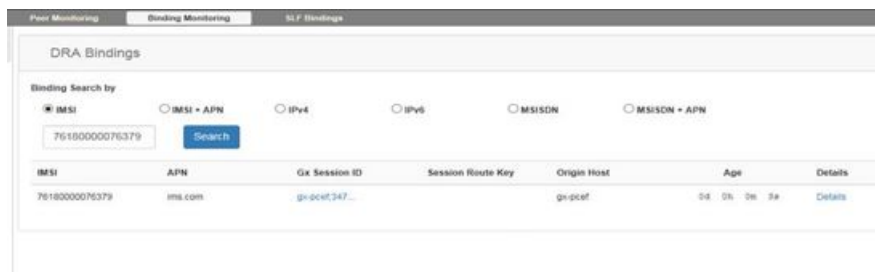

2. Check if the bindings are getting created. Filter the results for the imsiApn/ msisdnApn / ipv4/ ipv6 binding for which binding has to be retrieved.

Figure 4: DRA Bindings

3. Log into Central GUI/Grafana as admin and go to the **Home** > **Application Summary**.

Check for specific errors in Grafana. The errors indicate the exact result code received from peer.

Figure 5: Application Summary

4. Log into Central GUI/Customer Reference Data as admin.

Check for the descriptions of specific errors from customer reference data so that necessary action can be taken.

Figure 6: Error Result Code Profile

Relay Failure Between Two vDRA Instances

Use the following command to check traceroute connectivity from the VM to other nodes:

```
# node: DRA Director VM
# user: cps
cps@${drd-hostname}:~$ ping6 <Relay hostname configured in Policy Builder>
```

If there is any issue with the other vDRA, ping6 results in "timeouts."

Monitoring Exceptions

Use the following command to monitor exceptions in Redis or database:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# monitor log application | include Exception
```
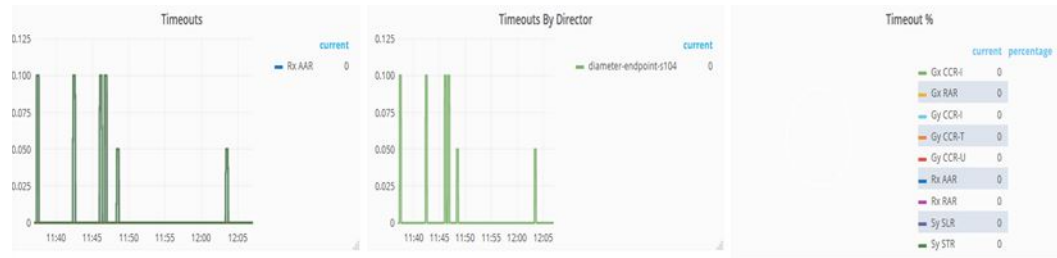
Monitoring Performance

To check if there are any performance issues with vDRA, log into Central GUI as admin and check the System Health.

Monitor for any timeouts, spikes or decrease in TPS for database response times, peer response timeouts, average response timeouts.

Figure 7: System Health

Figure 8: Timeouts
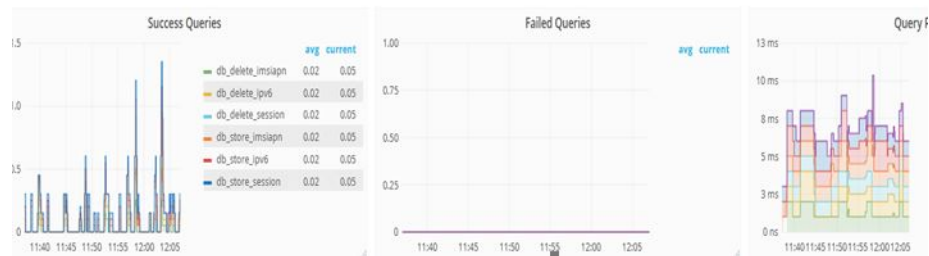

Figure 9: Database Queries

Figure 10: Peer Details
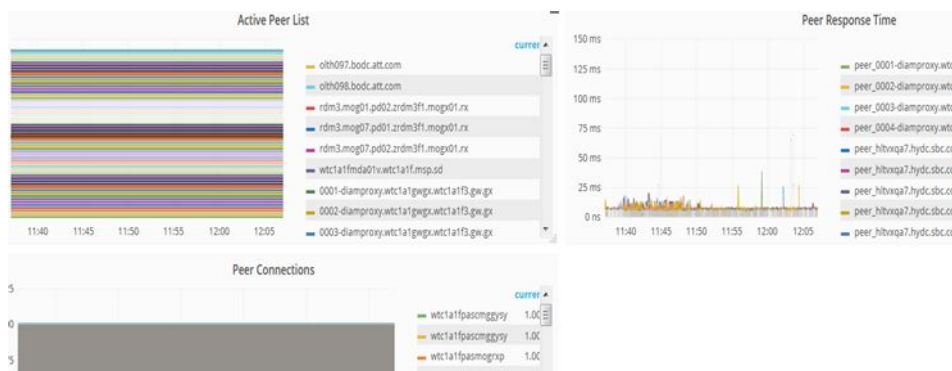
Check Alerts

Use the following command to check for alerts and any issues with peer connections, low memory, low disk, or link failures.

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show alert status | tab | include firing
```

Frequently Encountered Troubles in CPS vDRA

Redis Not Working

Step 1   Check redis status by executing the following command:

```
[root@lb01 ~]# service redis status
redis-server (pid  22511) is running...
```

Step 2   Try starting redis process by executing the following command:

```
[root@lb01 ~]# service redis start
```

Step 3   Check the following entries in /etc/broadhop/draTopology.ini file at policy directors (lb) and Policy Servers (QNS) for redis connecting on ports 6379, 6380, 6381, 6382:

```
[root@lb02 ~]# cat /etc/broadhop/draTopology.ini
dra.redis.qserver.1=lb02:6379
dra.redis.qserver.2=lb02:6380
dra.redis.qserver.3=lb02:6381
dra.redis.qserver.4=lb02:6382
dra.redis.qserver.4=lb02:6383
dra.local-control-plane.redis.1=lb02:6379
dra.mongodb.binding.db.ipv6.uri=mongodb://sessionmgr01:27718
dra.mongodb.binding.db.ipv4.uri=mongodb://sessionmgr01:27718
dra.mongodb.binding.db.imsiapn.uri=mongodb://sessionmgr01:27718
dra.mongodb.pcap.uri=mongodb://sessionmgr01:27718
dra.mongodb.binding.db.session.uri=mongodb://sessionmgr01:27718
[root@lb02 ~]# cat /etc/broadhop/redisTopology.ini
```

```
dra.redis.qserver.1=lb02:6379
dra.redis.qserver.2=lb02:6380
dra.redis.qserver.3=lb02:6381
dra.redis.qserver.4=lb02:6382
dra.local-control-plane.redis.1=lb02:6379
```

Step 4   Redis process on active policy director (lb) should be established with all Policy Servers (QNS) as shown below:

```
[root@lb01 ~]# netstat -na | grep 6379
tcp        0      0 0.0.0.0:6379              0.0.0.0:*                 LISTEN
tcp        0      0 80.80.80.10:6379         80.80.80.10:37400         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:38020         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:38034         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:37390         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38207         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.16:50597         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.14:35703         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.14:35711         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38188         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:37375         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38174         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38229         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38211         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.14:35709         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.16:50590         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:38032         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38172         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.16:50605         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:38204         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:38213         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:38223         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:38044         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38187         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38205         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:38211         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:37672         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.14:35710         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.17:59833         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:37388         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:37389         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:37662         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.17:59824         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.16:50596         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38210         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.15:49162         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38231         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38230         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.15:49159         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38152         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:37659         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38208         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.17:59832         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.15:49161         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38206         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:38212         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:38033         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:37650         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.15:49160         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38155         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.10:37660         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.17:59831         ESTABLISHED
tcp        0      0 80.80.80.10:6379         80.80.80.11:38186         ESTABLISHED
```

```
tcp       0      0 :::6379                    :::*                      LISTEN
tcp       0      0 ::ffff:80.80.80.10:37660   ::ffff:80.80.80.10:6379   ESTABLISHED
```

Gx Bindings not happening on Mongo

Step 1   Check if the binding's exceptions are coming in `consolidated-qns.log` file.

Step 2   Check for the entry `-DdraBindingTier=true` in `qns.conf` file on all Policy Servers (QNS).

Step 3   Check for the entries in `/etc/broadhop/draTopology.ini` file.

```
dra.redis.qserver.1=lb02:6379
dra.redis.qserver.2=lb02:6380
dra.redis.qserver.3=lb02:6381
dra.redis.qserver.4=lb02:6382
dra.redis.qserver.4=lb02:6383
dra.local-control-plane.redis.1=lb02:6379
dra.mongodb.binding.db.ipv6.uri=mongodb://sessionmgr01:27718
dra.mongodb.binding.db.ipv4.uri=mongodb://sessionmgr01:27718
dra.mongodb.binding.db.imsiapn.uri=mongodb://sessionmgr01:27718
dra.mongodb.pcap.uri=mongodb://sessionmgr01:27718
dra.mongodb.binding.db.session.uri=mongodb://sessionmgr01:27718
```

For example, make sure if the primary binding server is 27718 only as per above example.

Step 4   Check for the Binding Keys entries in binding key type profile and the application attached to the profile.

Rx Call Failing at CPS vDRA

Step 1   Check for the Binding key Retriever for Rx Profile.

Step 2   Check if the Gx Binding is available for that Binding key.

Step 3   Check the `consolidated-qns.log` file if CPS vDRA is able to retrieve SRK from the bindings.

Step 4   Check for any exception in `consolidated-qns.log` file during binding retrieval.

Step 5   If Rx peer is available for the same SRK at CPS vDRA, CPS vDRA should forward the Rx message to that peer.

Step 6   Check the connection for that peer and proper entries in Peer Group, Peer Routing, Peer Group Peer and Rx_Routing for Rx New session rules.

CPS vDRA Forwarding Message to Wrong Peer

Step 1   Check the Control Center configuration in Gx_Routing for new session rules. Gx routing should have the AVP defined on the basis of which, one wants to route the traffic.

Step 2   Check whether the Control Center configuration for the Peer is bonded to correct Peer Group.

Step 3   Check whether the Peer Group is assigned to correct Peer Route and Dynamic AVPs are properly aligned with Peer Route in Gx New Session Rules.

Step 4   Diameter Connection with the desired Destination Peer should be established with CPS vDRA.

PCRF Generated Messages not Reaching CPS vDRA

Step 1   Make sure PCRF has the correct entry of CPS vDRA as next hop.

Figure 11: Next Hop Routes

Next Hop Routing

*Next Hop Routes

*Next Hop Realm	*Next Hop Hosts	*Application Id	*Destination Realms P	*Destination Hosts Pa
cisco.v-pas-gx.com	cisco.v-pas	16777238	cisco.v-epc-gx.com	cisco.v-epc

Next Hop definition is mandatory in PCRF to forward the messages to CPS vDRA generated by PCRF itself.

For example, Gx-RAR, Sd-TSR

Step 2   Wild Card Entry not supported in Next Hop Routing configuration.

Issues in Reaching Ports and Setup IPs

Step 1   Check firewall is running or not.

Step 2   Make sure the firewall configuration is OK.

a) To check if this is the problem, then stop the firewall.

```
/etc/init.d/iptables stop
```

PB and CRD Inaccessible

Policy Builder and CRD are inaccessible when there are multiple route entries on the master node.

This issue occurs only on OpenStack setups.

OpenStack Neutron configures multiple default routes, if the gateway is also present in the interfaces static configuration.

For example, when configuring multiple interfaces on any VM, set "gateway" for only one interface, preferably public interface.

```
# public network
auto ens160
iface ens160 inet static
address x.x.x.60
netmask 255.255.255.0
gateway x.x.x.1
```

```
# private network
auto ens192
iface ens192 inet static
address y.y.y.155
netmask 255.255.255.0
```

Workaround

Run the following command to delete the default route to the internal network.

```
sudo route del default gw <internal network gateway IP>
```

For example: `sudo route del default gw y.y.y.1`

If the default route is not present for public network, run the following command:

```
ip route add default via <public network gateway IP>
```

For example: `ip route add default via x.x.x.1`

Central GUI Returns 503 Service Unavailable Error

After rebooting the master and control VMs, if the Central GUI returns 503 service unavailable error, perform the following steps:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# docker restart container-id haproxy-common-s101
```

Clear the browser cache and check the UI again.

Mog API Call Failure

If the MOG API calls fails internittently with an unauthroized message in a DRA director, then run the following commands to restart the container:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show network ips | include mogAPI
admin@orchestrator# show docker service | tab | include drd02v | include haproxy-common-s
admin@orchestrator# docker restart container-id haproxy-common-s10x
```

DRA Configuration API Call Returns NULL

If the DRA configuration API call returns null, restart the Policy Builder container as shown:

```
# node: DRA Master
# user: cps
cps@${DRM-hostname}:~$ cli
admin@orchestrator# show docker service | tab |
include drc01v | include policy-builder-s
admin@orchestrator# docker restart container-id
policy-builder-s10x
```