

CPS vDRA Troubleshooting Guide, Release 25.2.0

First Published: 2025-10-30

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/c/en/us/about/legal/trademarks.html. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2025 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface vii

About This Guide vii

Audience vii

Additional Support viii

Conventions (all documentation) viii

Communications, Services, and Additional Information ix

Important Notes x

CHAPTER 1

Troubleshooting CPS vDRA 1

Overview 1

General Troubleshooting 1

Installation Troubleshooting 1

System Maintenance Procedures 2

Backup Procedures 2

Back up CLI Configuration 2

Back up Policy Builder 3

Back up CRD 4

Shutting Down CPS 4

Shut down DRA VNF 4

Shut down DB VNF 4

Starting up CPS 5

Start up DRA VNF 5

Start DB VNF 5

Post Power up VM Health Check 5

Diameter Troubleshooting and Connections 5

DRA Plug-in Configuration in DRA Policy Builder (PB) 5

Troubleshooting Basics 6
Diameter Error Codes and Scenarios 6
Policy DRA Error Codes 9
Default HTTP Error Codes 10
Debug ping / ping6 11
Debug traceroute 11
Debug tcpdump 12
Monitoring Application Logs 12
Debug Tech to Capture Logs 12
Monitoring Container Logs 12
Monitoring Orchestrator Logs 12
Orchestrator CLI Mode Locked for All Users Due to Wrong nacm Rule Configuration 13
Change CLI User Password 14
Restart Docker Container 15
Check DNS Config 15
Redeploy Master VM 15
Remove MongoDB Replica Set Member 16
Monitoring MongoDB Logs 16
Recovering Member from Starting or Recovering State during ISSM or Rollback 16
Clean the Database 17
Reset the CLI Configuration 17
Policy DRA Logger Levels 17
Enabling and Disabling Logs 18
View Log Levels 18
View Logs 18
Change Log Level of consul-template 18
Common Loggers 19
Common Troubleshooting Steps 20
CPS vDRA Logs 20
Counters and Statistics 20
System Health Checks 20
View System Status 20
View System Diagnostics 20
Check System Scheduling Status 21

Check Docker Engine 21
Check Docker Service 22
View Alert Status 22
Troubleshooting Application 23
Call Failures 23
Relay Failure Between Two vDRA Instances 25
Monitoring Exceptions 25
Monitoring Performance 25
Check Alerts 28
Frequently Encountered Troubles in CPS vDRA 28
Redis Not Working 28
Gx Bindings not happening on Mongo 29
Rx Call Failing at CPS vDRA 30
Call Failing at CPS vDRA due to Binding 30
CPS vDRA Forwarding Message to Wrong Peer 30
PCRF Generated Messages not Reaching CPS vDRA 31
Issues in Reaching Ports and Setup IPs 31
PB and CRD Inaccessible 31
Central GUI Returns 503 Service Unavailable Error 32
Mog API Call Failure 32
DRA Configuration API Call Returns NULL 32
Removing or Correcting Incorrect Encoding Characters in Existing Configurations 33
Diameter Errors and Call Model Issues when Running Heap Dump 33
Recovery Steps when Master VM is Down 33
Call Failure Observed when Database VNF VMs are Recovering 33
Timeout Observed after Policy Builder Publish on DRA 34
No User Authentication Observed after MongoDB Auth Set on DRA VNF 34
Mongod Instance Fails to Start 34
Mongo-admin DB is not Healthy 35
Incorrect Alerts Observed on DB VNF 36
Interrupted Downgrades and Unhealthy Orchestrator DB State 36
RAR Routing Issue 39
Prometheus Database Corrupted after System Outage 39
Orchestration Container not Running after Fresh Installation 40

Orchestrator-backup not Added as Secondary Member 40
Docker Volumes Deletion 41
NTP Synchronization Behavior 42
Container not Recovering during Automation Run 42
Container Stuck during System Stop or Upgrade 44
show network ips command not Displaying all IPs 46
Wrong tmpfs Partition Names displayed in Mongo Containers 46
Binding/Endpoint Containers not Visible 48
VM Stuck in JOINING State 48
Consul Failing during DRA Downgrade or Upgrade 49
Grafana Loading Issue when Peer Number is High 50
ADMIN User Opens in Readonly Mode 51
Database IPs not Reachable 54
Shard Count Displaying Incorrect Primary 54
Missing Spikes in Longevity Report 54
Jetty Server Issue Reported in Logs 56
Scheduling Paused at cc-monitor after vmdk Redeployment 57
System Status Percent Stuck after Fresh Installation 57
SVN Error: Pristine Text not Present 58
Unreachable Peers with Redeployment 60
User Role Changes to Readonly 61
vDRA Database Troubleshooting 62
Database Operational Status 62
Validate Sharding Database Metadata 63
Validate Zone Aware Sharding Database Metadata 63
MongoDB Authentication Validation 64
Repair CLI failure due to missing .pem file for db_user authentication 65
Reconfigure the Databases 66
DB Replica Sets Reflects Bad State 69
Secnario: 1 All Sites are running with same mongo version 70
Sites running with same mongo version and prior mongo version 73
MongoDB Shard Recovery Procedure 80
Recovery Using database repair Command 82



Preface

- About This Guide, on page vii
- Audience, on page vii
- Additional Support, on page viii
- Conventions (all documentation), on page viii
- Communications, Services, and Additional Information, on page ix
- Important Notes, on page x

About This Guide

Note The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. While any existing biased terms are being substituted, exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

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.

Note The PATS/ATS, ANDSF, and MOG products have reached end of life and are not supported in this release. Any references to these products (specific or implied), their components or functions in this document are coincidental and are not supported. Full details on the end of life for these products are available at: https://www.cisco.com/c/en/us/products/wireless/policy-suite-mobile/eos-eol-notice-listing.html.

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.

Conventions	Indication
[ ]	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.

Communications, Services, and Additional Information

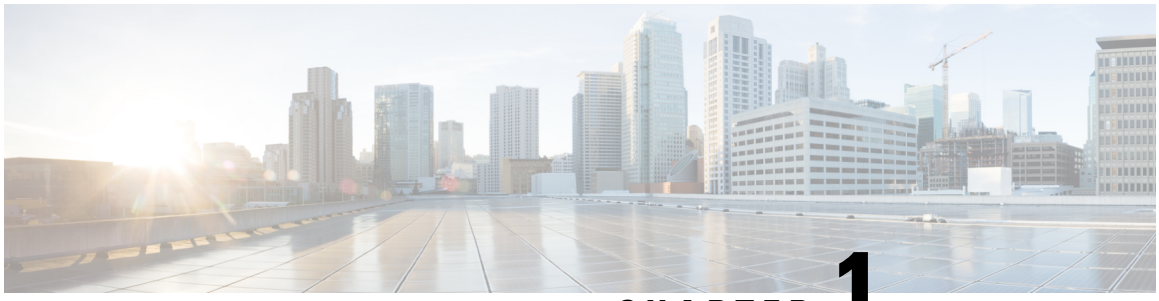
- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business results you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco DevNet.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

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.



CHAPTER 1

Troubleshooting CPS vDRA

- Overview, on page 1
- General Troubleshooting, on page 1
- Installation Troubleshooting, on page 1
- System Maintenance Procedures, on page 2
- Diameter Troubleshooting and Connections, on page 5
- Troubleshooting Basics, on page 6
- Policy DRA Logger Levels, on page 17
- Common Troubleshooting Steps, on page 20
- Troubleshooting Application, on page 23
- Frequently Encountered Troubles in CPS vDRA, on page 28
- vDRA Database Troubleshooting, on page 62

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
  ```

Installation Troubleshooting

This section describes common installation issues and its solution:

- **Issue:** The following error is displayed during vDRA install/delete/redeploy:

  ```
  sl.SSLEOFError: EOF occurred in violation of protocol (_ssl.c:645)
  ```

  Solution: Use the latest vDRA Deployer Host VMDK. This solves the issue by updating the TLS.

- **Issue**: Grafana page not loading after upgrade or installation.

  Solution: Restart grafana process with the following command docker exec grafana:

  ```
  supervisorctl restart grafana
  ```

- TCPDUMP Command failed with operation not permitted error.
  - **Issue**: tcpdump failed on ubuntu-20.04 with permission denied error even with sudo.

    ```
    Command : sudo tcpdump -i any -s 0 -w /var/broadhop/docker/DPR_Issue.pcap -W 50 -C
     100
    Conditions:
    Ubuntu-18.04 --> By default tcpdump file created with root user and which has access
     to all the folders by default
    -rw-r----- 1 root root 30713657 Jun 29 16:46 Issue.pcap00
    Ubuntu-20.04 --> By default tcpdump created with tcpdump user and which does not
    have access to all the folders due to which it was failing with
    ```

  - **Solution**: Run the run tcpdump command with target folder user name . (-Z user)

    ```
    sudo tcpdump -i any -Z root -s 0 -w /var/broadhop/docker/sample1.pcap -W 50 -C 100
     sudo
    tcpdump -i any -Z cps -s 0 -w /var/broadhop/docker/sample2.pcap -W 50 -C 100
    ```

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 System Health Checks, on page 20.

In case of resiliency event of DB VMs, sometimes database status present on that VM takes time to update. This is due to the orchestrator thread which schedules `show database status` command. You need to wait for ¾ mins. This issue doesn't always happen and is a rare one.

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)

Procedure

Step 1 Login to director VM and check the following files to find the ports that re open.

- For IPv4 endpoints, /var/broadhop/iptables/dra.iptables
- For IPv6 endpoints, /var/broadhop/iptables/dra.iptables6

Examples:

```
cps@dra1-sys04-director-1:~$ cat /var/broadhop/iptables/dra.iptables
dra,a72f412ed2d9d48386b543123f817a6bea4cc12c21b4ffaf5575681c9be5309f,-d 172.18.63.234 -p tcp
-m tcp --dport 4567 -m addrtype --dst-type LOCAL -j DNAT --to-destination 172.17.0.14:13868

cps@dra1-sys04-director-1:~$ cat /var/broadhop/iptables/dra.iptables6
dra,b76ddc032f1012c486547d5c2666fa6a3ec0082d6a502ffb2ae0d8f995434883,-d
2606:ae00:3001:8311:172:16:241:109 -p tcp -m tcp --dport 3868 -m addrtype
--dst-type LOCAL -j DNAT --to-destination [fd00:dead:beef:0:0:242:ac11:e]:13869
```

This indicates that the stack is up and running at IP and port 172.17.0.14:13868.

Step 2 Login to the Diameter endpoint container and check the port where Diameter stack is running.

If the port 3868 is configured in Policy Builder, internally in container Diameter stack runs on port 13868 (appends 1 infront of port number, this is internal port mapping). Similary for 3869, it shows diameter stack is running on 13869.

Example:

```
root@diameter-endpoint-s106:/# netstat -na | grep 3868
tcp6       0      0 :::13868                :::*                    LISTEN
```

Step 3 Listen for Diameter traffic by logging into Director VMs diameter endpoint container and execute the following command:

```
tcpdump -i any port 13868 -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.

Result-Code	Result-Code Value	Description
DIAMETER_LIMITED_SUCCESS	2002	Request is processed but some more processing is required by Server to provide access to user.
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]		

Result-Code	Result-Code Value	Description
DIAMETER_AUTHENTICATION_REJECTED	4001	Returned by Server, most likely because of invalid password.
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.

Result-Code	Result-Code Value	Description
DIAMETER_UNSUPPORTED _VERSION	5011	Self explanatory.
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	3005	008	Message loop detected

Policy DRA Error String	Error Code	Sub-code	Description
Parsing exception with message	3009	009	Parsing exception with message
CRD DB Error	3002	010	CRD DB Error
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
Binding DB Overload	3002	022	Binding record limit exceeded

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
2019 (INVALID_API_FORMAT)	400	Bad Request
2011 (Subscriber already exist with %s)	409	Duplicate subscriber
NA	401	Unauthorized
NA	403	Forbidden
NA	408	Request Time-out
NA	502	Bad Gateway
NA	503	Service Unavailable
NA	504	Gateway Time-out

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.

Monitoring 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
```

Monitoring 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>
```

Monitoring 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
```

Orchestrator CLI Mode Locked for All Users Due to Wrong nacm Rule Configuration

Issue: Orchestrator CLI does not function and timestamp when enabled does not work on the CLI.

Solution:

1. Add the *nacm* config into a xml file:

```
/var/confd/bin/confd_load -P '/nacm/rule-list[name="cfg-restrict"]' >
cfg.xml
```

```
cat cfg.xml

<config
               xmlns=http://tail-f.com/ns/config/1.0>
               <nacm
                              xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-acm">
                              <rule-list>
                                             <name>cfg-restrict</name>
                                             <group>*</group>
                                             <cmdrule

xmlns=http://tail-f.com/yang/acm>

<name>all-cfg-override-restrict</name>
                                                                    <command>load
override</command>

<access-operations>exec</access-operations>
                                                                    <action>deny</action>
                                             </cmdrule>
                                             <cmdrule

xmlns=http://tail-f.com/yang/acm>

<name>all-cfg-replace-restrict</name>
                                                                    <command>load
replace</command>

<access-operations>exec</access-operations>
                                                                    <action>deny</action>
                                             </cmdrule>
                                             <cmdrule

xmlns=http://tail-f.com/yang/acm>

<name>all-cfg-dbauth-remove-restrict</name>
                                                                    <action>deny</action>
                                             </cmdrule>
                              </rule-list>
               </nacm>
</config>
```

Modify cfg.xml as shown in the sample configuration:

```
<config
                xmlns=http://tail-f.com/ns/config/1.0>
                <nacm
                                xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-acm">
                                <rule-list>
                                                <name>cfg-restrict</name>
                                                <group>*</group>
                                                <cmdrule

xmlns=http://tail-f.com/yang/acm>

<name>all-cfg-override-restrict</name>
                                                                <command>load
override</command>

<access-operations>exec</access-operations>
                                                                <action>deny</action>
                                                </cmdrule>
                                                <cmdrule

xmlns=http://tail-f.com/yang/acm>

<name>all-cfg-replace-restrict</name>
                                                                <command>load
replace</command>

<access-operations>exec</access-operations>
                                                                <action>deny</action>
                                                </cmdrule>
                                                <cmdrule

xmlns=http://tail-f.com/yang/acm>

<name>all-cfg-dbauth-remove-restrict</name>
                                                                <command>db-authentication
 remove-password</command>

<access-operations>exec</access-operations>
                                                                <action>deny</action>
                                                </cmdrule>
                                </rule-list>
                </nacm>
</config>
```

2. Create the cfg.xml file with updated values and load the cfg.xml using the following command:

   `/var/confd/bin/confd_load -m -l cfg.xml.fix`
3. Restart the confd process inside orchestrator container:

   supervisorctl restart confd

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")
```

Monitoring MongoDB Logs

The MongoDB logs are stored under `/data/mongod-node/db` on every VM that has mongod instance running.

Recovering Member from Starting or Recovering State during ISSM or Rollback

After the ISSM or ISSM rollback, if some of the members are in the Starting or Recovering state, then perform the following steps:

- Identify the members by using show database command from CLI.
- Perform the following steps, delete the data `sharding` path, and start the members.

```
rm -rf /data/db/shardb-port/*
supervisorctl start sharding-IP*

root@mongo-s110:/# rm -rf /data/db/sharddb-27019/*
root@mongo-s110:/# rm -rf /data/db/sharddb-27018/*
root@mongo-s110:/# supervisorctl start sharding-2606-ae00-2030-2420-d-27018
sharding-2606-ae00-2030-2420-d-27018: started
root@mongo-s110:/# supervisorctl start sharding-2606-ae00-2030-2420-d-27019
sharding-2606-ae00-2030-2420-d-27019: started
```

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/mongod-node/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
```

Change Log Level of consul-template

Use the following command to change the log level of the consul-template:

```
Login to the container to change the log level of consul-template:
consul-template -log-level <level>
```

Set the log level value as debug, info, warn, and err.

Following is an example:

```
consul-template -log-level debug
```



Note The consul-template log level can be changed only on containers where the consul service is running.

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. mongo.sharding.impl	This logger provides logs about the binding and API handling operations managed by the Worker.	warn
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

Procedure

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 1: Peer Monitoring - Active Peer Endpoints

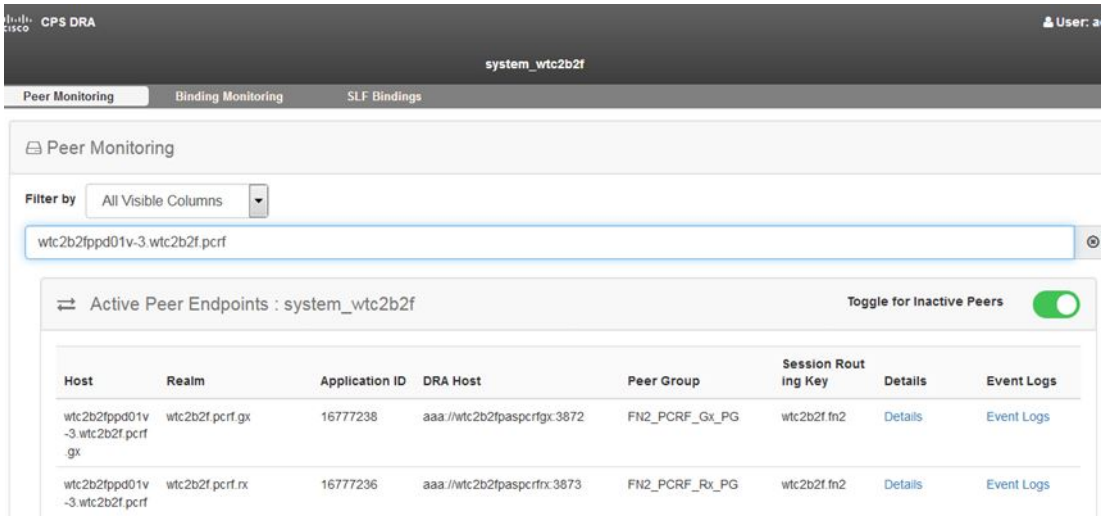
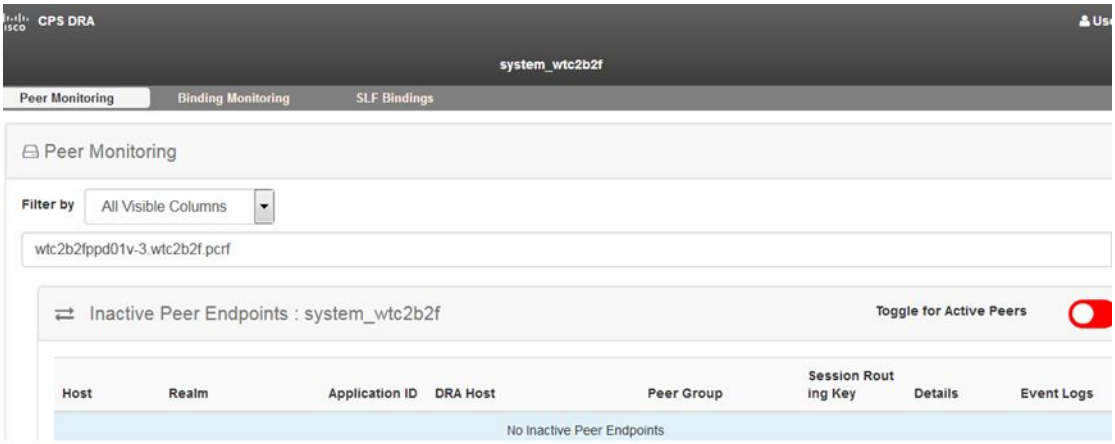
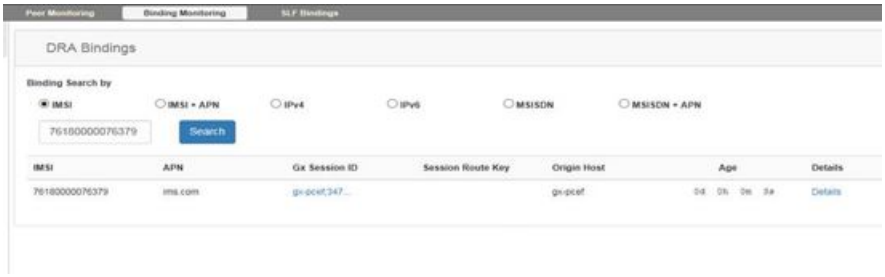


Figure 2: Peer Monitoring - Inactive Peer Endpoints



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 3: 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.

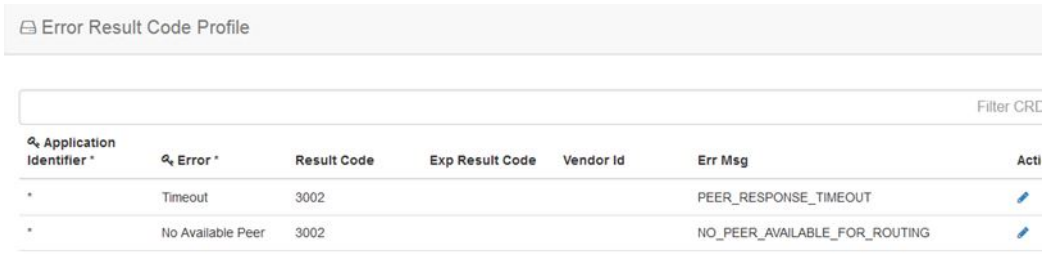
Figure 4: Application Summary



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.

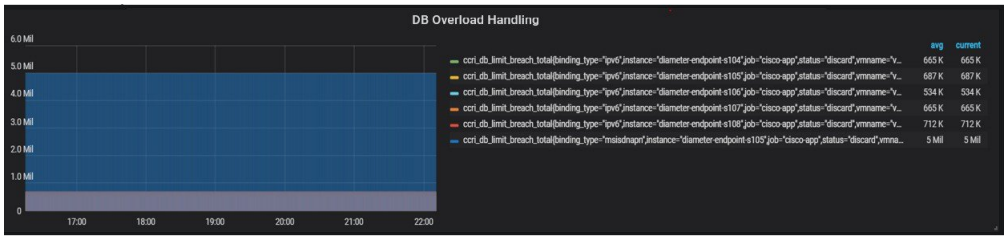
Figure 5: Error Result Code Profile



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

   Check for "discard" status in Grafana in DB Overload Handling graph. If entries are found in the graph, then check if maximum record limit has been set on database.

Figure 6: DB OVerload Handling



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

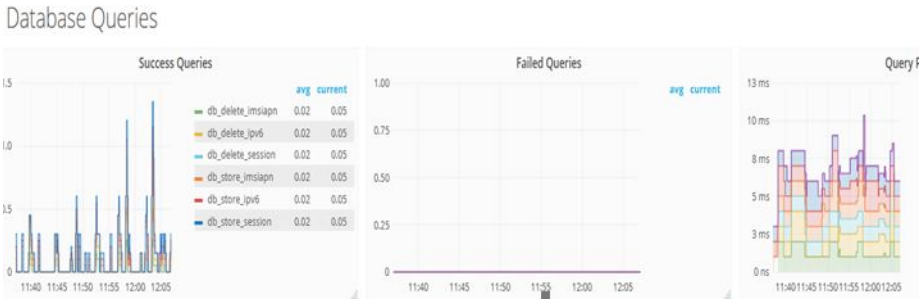
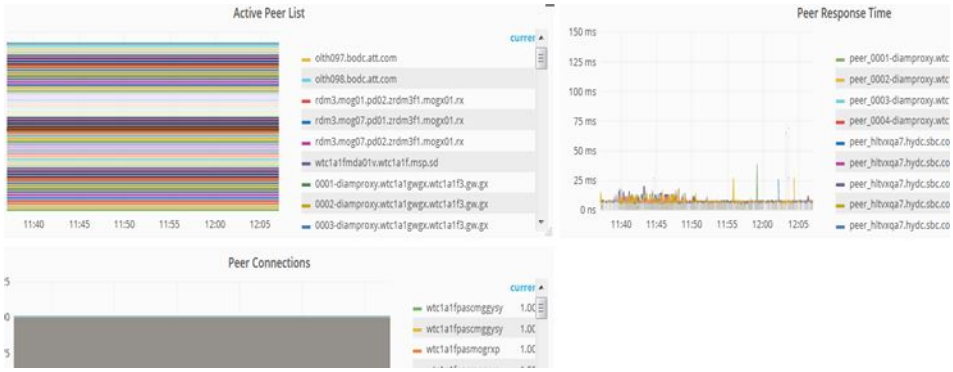
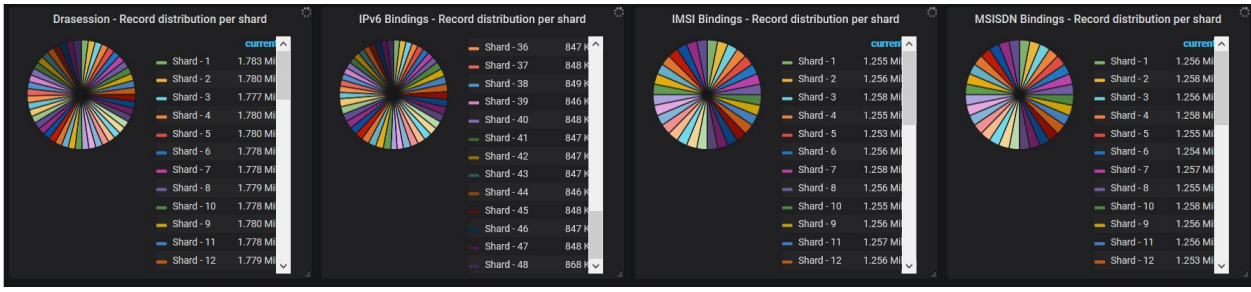


Figure 10: Peer Details



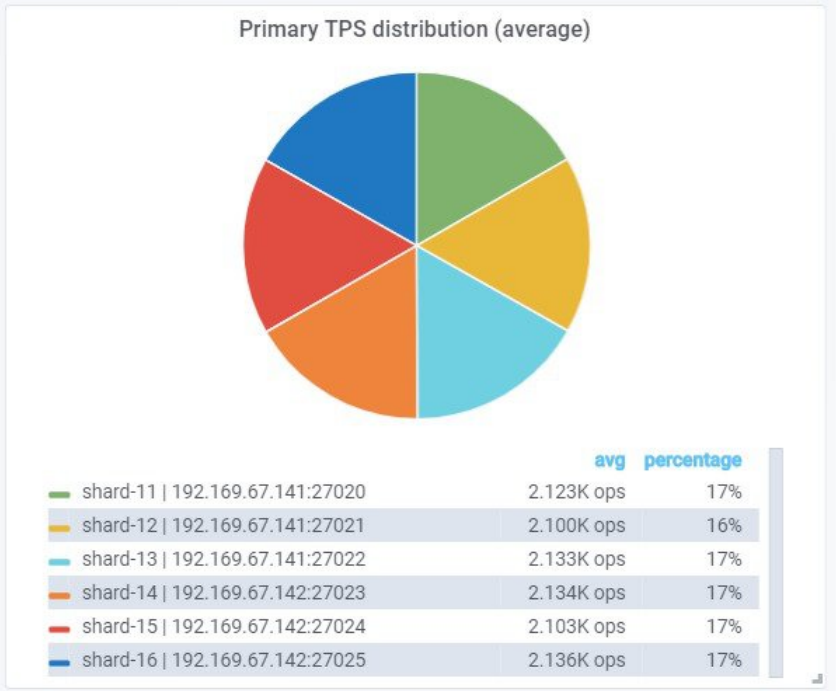
Monitor session and binding records are uniformly distributed across all the shards from ‘Application Summary’ dashboard of DRA VNF.

Figure 11: Record Distribution per Shard



Monitor Primary TPS is uniformly distributed across all the shards from 'Database Monitoring' dashboard of Binding VNF.

Figure 12: Primary TPS Distribution



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

Procedure

Step 1 Check which containers are available using the following commands:

```
admin@orchestrator[dra1-sys04-master-0]# show docker service | include control-plane | tab | exclude
 monitor
control-plane        101    control-plane        3.2.6.0   dra1-sys04-master-0    control-plane-s101
        HEALTHY  false -
control-plane        102    control-plane        3.2.6.0   dra1-sys04-control-0   control-plane-s102
        HEALTHY  false -
```

```
control-plane        103    control-plane        3.2.6.0   dra1-sys04-control-1   control-plane-s103
        HEALTHY  false -
diameter-endpoint    104    global-control-plane 3.2.6.0   dra1-sys04-director-1
global-control-plane-s104 HEALTHY  false -
diameter-endpoint    105    global-control-plane 3.2.6.0   dra1-sys04-director-2
global-control-plane-s105 HEALTHY  false -

amin@orchestrator[dra1-sys04-master-0]# show docker service | include redis | tab | exclude monitor
diameter-endpoint    104   diameter-redis-q-a    3.2.6.0   dra1-sys04-director-1
diameter-redis-q-a-s104   HEALTHY  false -
diameter-endpoint    105   diameter-redis-q-a    3.2.6.0   dra1-sys04-director-2
diameter-redis-q-a-s105   HEALTHY  false -
```

Step 2 Login in into each of the above containers.

The following example shows that the redis server is working.

```
admin@orchestrator[dra1-sys04-master-0]# docker connect control-plane-s101
/data # ps -ef
PID   USER     TIME   COMMAND
    1 redis    332:42 redis-server
```

Step 3 Check the following entries in `/etc/broadhop/draTopology.ini` file at DRA directors diameter-endpoint container.

```
root@diameter-endpoint-s104:/# cd /etc/broadhop
root@diameter-endpoint-s104:/etc/broadhop# cat draTopology.ini

dra.local-control-plane.redis.0=control-plane-s101:6379
dra.local-control-plane.redis.1=control-plane-s102:6379
dra.local-control-plane.redis.2=control-plane-s103:6379

dra.global-control-plane.redis.0=192.169.67.178:6379

root@diameter-endpoint-s104:/etc/broadhop# cat redisTopology.ini
#Generate file from consul configuration

dra.redis.qserver.0=diameter-redis-q-a-s104:6379
dra.redis.qserver.1=diameter-redis-q-a-s105:6379
```

Step 4 Verify that the global control plane is configured correctly from the CLI. For more on commands, see the *CPS vDRA Operations Guide*.

Gx Bindings not happening on Mongo

Procedure

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

Step 2 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
db.shards.metadata.ipv6.uri=mongodb://[2606:ae00:3001:8311:172:16:244:3]:27019,[2606:ae00:3001:8311:172:16:244:2a]:27019
db.shards.metadata.ipv4.uri=mongodb://[2606:ae00:3001:8311:172:16:244:4]:27019,[2606:ae00:3001:8311:172:16:244:2b]:27019
```

```
db.shards.metadata.imsiapn.uri=mongodb://[2606:ae00:3001:8311:172:16:244:4]:27019,[2606:ae00:3001:8311:172:16:244:2b]:27019
db.shards.metadata.msisdnapn.uri=mongodb://[2606:ae00:3001:8311:172:16:244:4]:27019,[2606:ae00:3001:8311:172:16:244:2b]:27019
db.shards.metadata.session.uri=mongodb://[2606:ae00:3001:8311:172:16:244:3]:27019,[2606:ae00:3001:8311:172:16:244:2a]:27019
```

For example, make sure if the primary binding server is 27019 only as per above example.

Step 3 Check for the Binding Keys entries in binding key type profile and the application attached to the profile.

Rx Call Failing at CPS vDRA

Procedure

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.

Call Failing at CPS vDRA due to Binding

Procedure

Step 1 Check the `consolidated-qns.log` file to see if there are any warn logs on MongoShardPinger class related to unreachable mongo.

Step 2 If MongoShardPinger logs are present with text containing unreachable mongos it indicates the shard is not reachable.

Step 3 Check the connection for that shard.

CPS vDRA Forwarding Message to Wrong Peer

Procedure

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.

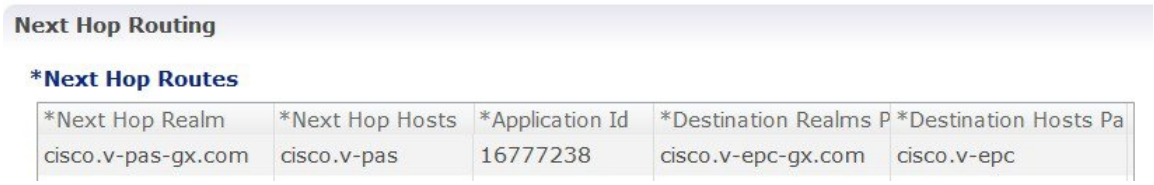
Step 4 Diameter Connection with the desired Destination Peer should be established with CPS vDRA.

PCRF Generated Messages not Reaching CPS vDRA

Procedure

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

Figure 13: 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

Procedure

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
```

Removing or Correcting Incorrect Encoding Characters in Existing Configurations

Issue: Initially, if you enter incorrect characters (encoding should be UTF-8 and not windows) as input to the expressions on the CLI (config), the subsequent configurations does not get reflected. This is because the **confd** does not accept incorrect characters in the **confd** Database and does not allow you to proceed.

Workaround: To recover, you must either remove or correct the incorrect expressions from the running configurations and then input the subsequent expressions to add the configurations.

Diameter Errors and Call Model Issues when Running Heap Dump

Issue: Messages timed out when running Heap Dump of DRA process.

Condition: Taking heap dump of director and worker process. Heap dumps taken results in full GC. This in turn causes major application pause and results in message time out.

Solution: It is recommended to take the heap dump only during Maintenance Window (MW)

Recovery Steps when Master VM is Down

Issue: When Master VM is powered ON after 12 hrs it is stuck on orchestrator container throwing following logs:

```
2019-07-01T03:40:12.858+0000 I NETWORK [conn78] end connection 127.0.0.1:33586 (4 connections
 now open)
Waiting for 5s as mongod database is not yet up. Mon Jul 1 03:40:12 UTC 2019
2019-07-01T03:40:13.469+0000 I REPL [replexec-2] Canceling priority takeover callback
2019-07-01T03:40:13.469+0000 I REPL [replexec-2] Not starting an election for a priority
takeover, since we are not electable due to: Not standing for election because member is
not currently a secondary; member is not caught up enough to the most up-to-date member to
 call for priority takeover - must be within 2 seconds (mask 0x408)
2019-07-01T03:40:15.246+0000 I REPL [replexec-2] Scheduling priority takeover at
2019-07-01T03:40:26.616+0000
2019-07-01T03:40:17.919+0000 I NETWORK [listener] connection accepted from 127.0.0.1:33596
 #79 (5 connections now open)
```

Solution: To recover the master VM, you need to execute the following commands on Master VM:

```
docker exec -it orchestrator bash
supervisorctl stop mongo
rm -rf /data/db/*
supervisorctl start mongo
```

Call Failure Observed when Database VNF VMs are Recovering

Issue: Calls failure observed when database VNF VMs are in recovery mode.

Expected Behavior: Few call failures are expected when a shard-member recovers after restart and gets elected as new primary. The following is the expected behavior when a Binding VNF recovers after failover:

- All the shards members of the database VNF do not come up at the same time. They resynchronize with the existing shard-members and transition from STARTUP2 to Secondary to Primary state is not same for all the shards.
- Two elections for each shard are possible based on the database VM recovery time. The following is the sequence:
  1. **First Election:** Database VM having shard member with second highest priority completes the resynchronization first and becomes primary.
  2. **Second Re-election:** The shard member with highest priority completes the resynchronization and becomes primary (This behavior is as per the MongoDB replica-set protocol version 1 - pv1).

Timeout Observed after Policy Builder Publish on DRA

Issue: Timeout is observed when publishing is done during load.

Solution: Policy Builder publishing during load have impact on running calls.



Note It is recommended to perform Policy Builder publishing during Maintenance Window (MW).

No User Authentication Observed after MongoDB Auth Set on DRA VNF

Issue: No users are authenticated after MongoDB auth set on DRA VNF.

Solution: After new password is set on binding VNF and DRA VNF if `there are no users authenticated` exception is observed, restart the binding container.

Mongod Instance Fails to Start

Issue: If mongod instance fails to start and displays the following error:

```
2019-08-26T07:11:48.012+0000 I -  [repl writer worker 7] Fatal assertion 16359
NamespaceNotFound: Failed to apply insert due to missing collection:
{ ts: Timestamp 1566803477000|424, t: 15, h: -4599534508984183152, v: 2, op: "i",
ns: "ipv6bindings.ipv6bindings", o: { _id: "3001:0000:0019:3dd5", ts: 1566804446889,
staleBindingExpiryTime: new Date(1566815246889), srk: "fPAS.CALIPER.PCRF4", fqdn:
"client-calipers24-gx.pcef.gx", sessionid: "ClpGx3:172.16.241.40:5024:1566562707:0021653880",

uuid: "fpas-system-22133581349" } } at src/mongo/db/repl/sync_tail.cpp 1059
2019-08-26T07:11:48.012+0000 I -        [repl writer worker 7]

***aborting after fassert() failure
```

Solution: Restart the MongoDB instances manually giving JIRA reference of mongod instance.

Mongo-admin DB is not Healthy

Issue: Mongo-admin DB is showing unhealthy status:

```
admin@orchestrator[pn-master-0]# show docker service | exclude HEAL | tab
PENALTY
MODULE INSTANCE NAME VERSION ENGINE CONTAINER ID STATE BOX MESSAGE
---------------------------------------------------------------------------------------------------------------------------------------------------
admin-db 1 mongo-admin-a 22.2.11-2023-01-31.17123.aa3e75222 pn-master-0 mongo-admin-a ABORTED
 true Exit code: 62
admin-db 1 mongo-admin-c 22.2.11-2023-01-31.17123.aa3e75222 pn-control-1 mongo-admin-c
ABORTED true Exit code: 14
subversion 1 svn 22.2.11-20230131_152801.10637 pn-control-0 svn STARTED true Pending health
 check

or in logs are showing as cps@pn-master-0:~$ docker logs mongo-admin-a

{"t":{"$date":"2023-02-06T09:04:28.825+00:00"},"s":"W", "c":"STORAGE", "id":22347,
"ctx":"initandlisten","msg":"Failed to start up WiredTiger under any compatibility version.
 This may be due to an unsupported upgrade or downgrade."}
{"t":{"$date":"2023-02-06T09:04:28.825+00:00"},"s":"F", "c":"-", "id":23089,
"ctx":"initandlisten","msg":"Fatal
assertion","attr":{"msgid":28559,"file":"src/mongo/db/storage/wiredtiger/wiredtiger_util.cpp","line":97}}
{"t":{"$date":"2023-02-06T09:04:28.825+00:00"},"s":"F", "c":"-", "id":23090,
"ctx":"initandlisten","msg":"\n\n***aborting after fassert() failure\n\n"}
{"t":{"$date":"2023-02-06T09:04:28.825+00:00"},"s":"F", "c":"CONTROL", "id":4757800,
"ctx":"initandlisten","msg":"Writing fatal message","attr":{"message":"Got signal: 6
(Aborted).\n"}}
```

Workaround:

Prerequisites: The following are prerequisites.

In DRA VNF:

- Take a backup of *pb* and *crd* data as we're cleaning up mongo admin data.
- Clean mongo-admin data on `Master/control-0/control-1` as shown below:
  - **master VM**: sudo rm -rf /data/mongo-admin-a/admin-db/*
  - **control-0**: sudo rm -rf /data/mongo-admin-b/admin-db/*
  - **control-1**: sudo rm -rf /data/mongo-admin-c/admin-db/*



Note Wait for 5 minutes and check the **show admin-database-status**. If all members are in **NO_CONNECTION** state only then execute the next step.

- (Optional) Initialize mongo on master and add members as shown below:
  - docker exec -it mongo-admin-a bash
  - mongo
  - rs.initiate()
  - rs.add("host" : "mongo-admin-b:27017")
  - rs.add("host" : "mongo-admin-c:27017")

Incorrect Alerts Observed on DB VNF

Issue: When Prometheus servers residing on Master/Control VMs are restarted, alerts are raised by the servers remaining in firing state. If the alerts are not resolved before Prometheus restart, the alerts remain in firing state forever even after issue is resolved.

For example,

```
admin@orchestrator[fPAS-site2-master-1]# show alert status | tab | include IP_NOT
IP_NOT_REACHABLE  172.26.50.114    firing    VM/VIP IP 172.26.50.114 is not reachable!
2019-08-28T14:52:49.776+00:00  -  2019-08-29T12:03:03.553+00:00
IP_NOT_REACHABLE  172.26.50.64     firing    VM/VIP IP 172.26.50.64 is not reachable!
2019-08-28T14:51:54.768+00:00  -  2019-08-29T12:03:03.548+00:00
IP_NOT_REACHABLE  172.26.50.65     firing    VM/VIP IP 172.26.50.65 is not reachable!
admin@orchestrator[fPAS-site2-master-1]# show running-config docker | tab
ID                              ADDRESS
  SCHEDULING SLOTS  MODE            ID        ADDRESS
-------------------------------------------------------------------------------------------------------------
fPAS-site2-app-persistence-db-7  http://engine-proxy-6662e4dc999a9e36f4f0ea2d0fbfcedf:2375
  [ mongo-node ]    [ mongo-node ]  internal  172.26.50.114
fPAS-site2-control-2             http://engine-proxy-89fa384df65a6c7863252a22fcbfd696:2375
  [ control-a ]     -               internal  172.26.50.65
fPAS-site2-master-1              http://engine-proxy-03b546144fadc57dc51c0fcd063375da:2375
  [ master ]        -               internal  172.26.50.64


fPAS-site2-app-persistence-db-7  CONNECTED  0
fPAS-site2-control-2             CONNECTED  0
fPAS-site2-control-3             CONNECTED  0
fPAS-site2-master-1              CONNECTED  0
```

Solution: Manually clear the alarms using CLI.

To check if this behaviour is present on a system.

1. Login to Promethues container using CLI.

   ```
   docker connect prometheus-hi-res-s<101/102/103>
   ps aux
   ```



Note Note down Prometheus start time. If it is later than “alerts” firing time, then this alert will never get cleared.

Interrupted Downgrades and Unhealthy Orchestrator DB State

Issue: Downgrade is interrupted and orchestrator mongo db replica set is showing unhealthy state.

Scenario 1:

- Verify the scenario by.
  - **Log in to orchestrator primary:** docker exec -it orchestrator bash
  - **Log in to mongo:** mongo
  - If rs.status() contains only primary, no secondaries are present as shown:

    ```
    orchestrator:PRIMARY> rs.status()

    {
    ```

```
        "set" : "orchestrator",
……
        "members" : [
                {
                        "_id" : 0,
                        "name" : "orchestrator:27017",
                        "health" : 1,
                        "state" : 1,
                        "stateStr" : "PRIMARY",
                        "uptime" : 231095,
                        "optime" : {
                                "ts" : Timestamp(1677477849, 5),
                                "t" : NumberLong(8)
                        },
                        "optimeDate" : ISODate("2023-02-27T06:04:09Z"),
        }
}
```

Solution: Reconfigure the DB with the following commands:

```
cfg=rs.config()
cfg={_id:"orchestrator", members:[ {_id:0,host:"orchestrator:27017","priority" : 9},
{_id:1,host:"orchestrator-backup-a:27017","priority" : 3},
{_id:2,host:"orchestrator-backup-b:27017","priority" : 2}]}
rs.reconfig(cfg,{force:true})
```

To verify whether any secondary DB is not able to connect successfully: Check if rs.status() contains the following messages on the primary orchestrator db.

```
"replica set IDs do not match, ours:<new replset name>,remote node's: <old replset name>"
like given below for orchestrator-backup-a

orchestrator:PRIMARY> rs.status()

{
        "set" : "orchestrator",
……
        "members" : [
                {
                        "_id" : 0,
```

```
                        "name" : "orchestrator:27017",
                        "health" : 1,
                        "state" : 1,
                        "stateStr" : "PRIMARY",
                        "uptime" : 231095,
                        "optime" : {
                                "ts" : Timestamp(1677477849, 5),
                                "t" : NumberLong(8)
                        }, {
                        "_id" : 1,
                        "name" : "orchestrator-backup-a:27017",
                        "health" : 1,
                        "state" : 8,
                        "stateStr" : "Not rechable/healthy",
                        "uptime" : 231035,
                        "optime" : {
                                "ts" : Timestamp(1677477848, 5),
                                "t" : NumberLong(8)
                        },
                        "optimeDurable" : {
                                "ts" : Timestamp(1677477848, 5),
                                "t" : NumberLong(8)
                        },
                        "lastHeartbeatMessage" : " "replica set IDs do not match, ours:
60d054b0cfe1dc9265e996f7; remote node's: 60d321b914915e255a0edd74",
                },
                        "optimeDate" : ISODate("2023-02-27T06:04:09Z"),
        }
}
```


Solution: Clean the DB for which "replica set IDs do not match" message is received in rs.status. For example, if orchestrator-backup-a message comes up, then clean the orchestrator-backup-a message as specified below:

```
login to orchestrator-backup-a container from control VM: docker exec -it
orchestrator-backup-a bash
```

```
clean db: sudo rm -rf /data/db/*
```

Scenario 2: Orchestrator mongo DB is in startup2 state.

To verify the scenario.

- **Log in to orchestrator primary**: docker exec -it orchestrator bash
- **Log in to mongo**: mongo
- Verify if the status on CLI prompt displays as startup2:

  ```
  orchestrator:STARTUP2>
  ```

Solution: Recover the Orchestratorr DB:

```
login to orchestrator primary: docker exec -it orchestrator bash

clean db: sudo rm -rf /data/db/*
```

RAR Routing Issue

Issue: When running a call-model, it is observed that all the RAR messages initiated from server-side are routed over relay connection to other sites. From logs, it is seen that the peer is detected as active but due to some reason the siteID is detected as NULL due to which the neighbor siteID is preferred to route the messages.

Condition: 4 site cluster with peer_group and peer-route configured on the basis of FQDN HOST.

Solution: Restart the diameter-endpoint and binding containers on DRA_VNF. It can be seen from the Grafana if the response time is higher than usual for PCRF RAR traffic.

Prometheus Database Corrupted after System Outage

Issue: After system restart containers are in STARTED state and not able to recover.

Conditon: The containers are in STARTED state and not able to recover after restart.

The following errors are observed in the logs in monitor log container:

```
level=error ts=2018-04-07T04:28:53.784390578Z caller=main.go:582
err="Opening storage failed unexpected end of JSON input"
level=info ts=2018-04-07T04:28:53.784418708Z caller=main.go:584
msg="See you next time!"
```

```
admin@orchestrator[site4-master-0]# show docker service | exclude HEAL | tab

                                                                                PENALTY
MODULE     INSTANCE  NAME                          VERSION                      ENGINE
      CONTAINER ID          STATE    BOX      MESSAGE
-------------------------------------------------------------------------------------------------------------------
prometheus   101   prometheus-planning   19.5.7-2020-01-30.9009.34ef765  site4-master-0
prometheus-planning-s101   STARTED  true   Pending health check
prometheus   101   prometheus-trending   19.5.7-2020-01-30.9009.34ef765  site4-master-0
prometheus-trending-s101   STARTED  false    -
prometheus   102   prometheus-hi-res     19.5.7-2020-01-30.9009.34ef765  site4-control-0
prometheus-hi-res-s102     STARTED  true   Pending health check

admin@orchestrator[site4-master-0]#
```

Solution:

1. Go to the master/control VM and execute the following command:

```
cd /stats
sudo find . -name meta.json | xargs ls -lhrt
```

2. Locate meta.json which is empty (0 file, size).

   For example:

```
-rw-r--r-- 1 root root  283 Feb  6 17:00
./prometheus-hi-res/2.0/01E0DQ3C7QAXVJ1C3M9WX24WJH/meta.json
-rw-r--r-- 1 root root    0 Feb  6 19:00
./prometheus-planning/2.0/01E0DXZ2YW04DETYFCM4P9JT2R/meta.json
-rw-r--r-- 1 root root    0 Feb  6 19:00
./prometheus-trending/2.0/01E0DXZ3H751XWBKK1AN3WX6QV/meta.json
-rw-r--r-- 1 root root    0 Feb  6 19:00
./prometheus-hi-res/2.0/01E0DXZ3FV3NR87Q0S4738HK3Q.tmp/meta.json
-rw-r--r-- 1 root root  283 Feb  7 06:29
./prometheus-hi-res/2.0/01E0F5EDDATS4K4T75P2EKE8PS/meta.json
-rw-r--r-- 1 root root  282 Feb  7 06:29
./prometheus-hi-res/2.0/01E0F5EFD9RFXC5WBC0V01WD2R/meta.json
-r
```

3. Delete the directory containing `meta.json` file.

   For example:

```
sudo rm -fr ./prometheus-trending/2.0/01E0DXZ3H751XWBKK1AN3WX6QV*
```

4. Restart the container for which `meta.json` is empty.

   For example, if the directory for prometheus-trending is deleted, restart prometheus-trending container on that VM.

Orchestration Container not Running after Fresh Installation

Issue: Sometimes orchestrator container is not running after fresh installation. As a result, user is unable to login to CLI mode for site VNF database.

Solution: If orchestrator container does not come UP on master VM, check `/var/log/install-master.log` generated on master VM.

If the log is stuck at `Starting new HTTP connection (1): 127.0.0.1` for prolonged duration then run the following commands:

```
docker stop $(docker ps -aq)
docker rm -f $(docker ps -aq)
sudo rm /var/cps/bootstrap-status
sudo /root/bootstrap.sh
```

Orchestrator-backup not Added as Secondary Member

Issue: Master db orchestrator is working as primary but orchestrator-backup-a and orchestrator-backup-b are not added as secondary members.

Solution: Follow the steps to add orchestrator-backup as the secondary member.

1. Login to orchestrator-backup-a on control-0 and cleanup db data:

- cps@control-0:~$ docker exec -it orchestrator-backup-a bash
- rm -rf /data/db/*

2. Login to orchestrator-backup-b on control-1 and cleanup db data:
   - cps@control-1:~$ docker exec -it orchestrator-backup-b bash
   - rm -rf /data/db/*
3. Login to master orchestrator db and add secondary members (orchestrator-backup-a ,orchestrator-backup-b) in orchestrator mongo db:
   - cps@master:~$ docker exec -it orchestrator bash
   - mongo (run mongo shell)
4. Run the following commands in mongo shell.

```
cfg=rs.config()

 cfg={_id:"orchestrator", members:[ {_id:0,host:"orchestrator:27017","priority" : 9},
{_id:1,host:"orchestrator-backup-a:27017","priority" : 3},
{_id:2,host:"orchestrator-backup-b:27017","priority" : 2}]}

 cfg.protocolVersion=1

 rs.reconfig(cfg,{force:true})
```

Docker Volumes Deletion

Issue: Some docker volumes are not usable when `system stop` is stuck.

Solution: When `system stop` gets stuck docker services need to be restarted for any VM. Additionally, docker volumes can be deleted manually after completion of docker restart.

Login to the VM where docker service are restarted and perform the following steps.

1. Verify the existence of docker volumes by running `docker volume ls | grep cps` command.

```
cps@site1-dra-worker-1:~$ docker volume ls | grep cps
local               cps-docker-info
local               cps-keepalived-conf
local               cps-keepalived-service-status
local               cps-keepalived-supervisord
```

2. Verify that the following containers have been stopped:
   - keepalived
   - node-exporter
   - diameter-endpoint
   - real-server

   If any of above containers is still running, execute `system stop` command to stop the container service from the respective VNF CLI.
3. Delete the volume by running `docker volume rm <volumename>` command.

Example:

```
docker volume rm cps-docker-info
docker volume rm cps-keepalived-service-status
```

4. Verify that the docker volumes are deleted by running `docker volume ls | grep cps` command.

NTP Synchronization Behavior

Issue: After system stop/upgrade/downgrade or site/VM restart, NTP related system diagnostics messages are observed.

Sample output with diagnostics messages:

```
admin@orchestrator[master-4]# show system diagnostics | tab | exclude passing
NODE CHECK ID IDX STATUS MESSAGE
--------------------------------------------------------------------------------------------
control-plane-s101 serfHealth 1 critical Agent not live or unreachable
ntpd-s102 service:ntp-client 1 warning unsynchronised
ntpd-s102 service:ntp-client 2 warning polling server every 8 s
ntpd-s103 service:ntp-client 1 warning unsynchronised
ntpd-s103 service:ntp-client 2 warning polling server every 8 s
ntpd-s104 service:ntp-client 1 warning unsynchronised
ntpd-s104 service:ntp-client 2 warning polling server every 8 s
ntpd-s105 service:ntp-client 1 warning unsynchronised
ntpd-s105 service:ntp-client 2 warning polling server every 8 s
```

Observation: It is the NTP protocol behavior that it takes time to synchronize based on the time difference between the configured NTP server and local NTP client running on containers.

As it takes some time to synchronize with the NTP server, the diagnostics messages are expected to be seen during this time period. The NTP sync issues can make diameter / binding endpoints out of sync with the connected peers, which results into call failures. If the system diagnostics are still showing up ntp-related messages, then contact your Cisco Account representative.

During this time period you can execute the following commands on the master VM:

```
#vmware-toolbox-cmd timesync status
```

If time sync status is not enabled already, you can enable time it by executing `vmware-toolbox-cmd timesync enable` command on master VM.

On the remaining VMs, time sync status must be disabled by executing `vmware-toolbox-cmd timesync disable` command.

Sample output: The system diagnostics should not contain any ntp-related messages after the NTP sychronization is complete.

Container not Recovering during Automation Run

Issue: `show system diagnostics` displays errors though the container is in HEALTHY state.

Note This issue can be observed on both DRA or Binding VNFs in any of the following scenarios:

- After VM restart
- ISO upgrade
- ISO downgrade
- Restarting of containers multiple times in short duration (for any testing)

Workaround: The following steps can be executed to confirm and resolve the issue:

1. Execute `show system diagnostics | tab | exclude passing` command to check the diagnostics status.

   Here is a sample output that displays errors in the container.

   ```
   admin@orchestrator[site2-master-0]# show system diagnostics | tab | exclude passing

   NODE              CHECK ID    IDX   STATUS       MESSAGE
   ------------------------------------------------------------------------------
   stats-relay-s102  serfHealth   1   critical  Agent not live or unreachable
   ```

2. Execute `show docker service | tab | include <containerName>` command to verify the health of the container.

   Here is a sample:

   ```
   admin@orchestrator[an-master]# show docker service | tab | include stats-relay-s102
   stats  102  stats-relay  19.4.0-xxx  an-control-0  stats-relay-s102  HEALTHY  false -
   ```

3. If Step 2 displays the state as **HEALTHY**, use the following workaround. Otherwise, diagnostics error is valid and check the container logs to find the root cause.

   a. From CLI, execute `docker connect <container-name>` to connect to a docker service and launch a bash shell running on the system.

      For example, `docker connect consul-1`

      1. Execute `consul members | grep -v alive` command.

         Here is the sample output.

         ```
         root@consul-1:/# consul members | grep -v alive
         Node                            Address         Status  Type    Build  Protocol
         DC   Segment
         stats-relay-s102.weave.local 10.45.0.32:8301  failed  client  1.0.0   2       dc1
           <default>
         ```

      2. Execute `consul force-leave stats-relay-s102.weave.local` command for all the containers which are in failed state.

   b. Execute `docker restart container-id stats-relay-s102` to restart the container.

4. Execute `show system diagnostics | tab | exclude passing` command to verify that the issue has been fixed.

Container Stuck during System Stop or Upgrade

Issue: During system stop/start or upgrade, container on a VM isn't removed by the orchestrator.

Possible Cause: There are multiple reasons for this issue. The following are some of the causes identified:

1. Sometimes docker takes time to stop container than expected. In this case, orchestrator which is trying to remove the container times out and does retry. Same situation gets repeated continuously and hence container never gets removed.
2. Sometimes during container stop, the container isn't stopped cleanly as volumes attached to it does not get deleted. In this case, the new container fails to start due to old volumes.
3. Sometimes during container stop, the container isn't stopped cleanly so if the docker proxy is exposing any ports for that container, those ports still remain in listening state. Hence when the new containers are started and docker proxy starts listening on exposed ports, it gets bind failures and thus the container doesn't get started.

Solution:

1. Execute `system abort-upgrade` command from CLI to abort the running upgrade.
2. In case of system stop/upgrade, if the container doesn't get removed, perform the following steps:
   a. Find VM from where the containers doesn't get removed.

      Example: Execute the following command from CLI.

      ```
      show scheduling status | tab | exclude RUNNING
      monitoring 106 125 application SCHEDULING false
      ```

      `monitoring 106 125 application SCHEDULING false` indicates the instance ID

      ```
      admin@orchestrator[vpas-A-dra-master-0]# show docker service | tab | include 106
      diameter-endpoint 106 diameter-endpoint 19.5.0-20200506_032527.7339
      vpas-A-dra-director-c diameter-endpoint-s106 ABORTED false
      ```

      `vpas-A-dra-director-c diameter-endpoint-s106 ABORTED false` indicates the VM name. In this example, the VM name is `vpas-A-dra-director-c`.

   b. Login to VM and use `docker ps | grep <container-name>` and timestamp of container to check if the container doesn't get removed.

   c. If the container is not removed, execute `docker rm -f <container-name>` to remove the container forcefully.

3. SSH to VM having issues and verify if the old volumes doesn't get deleted with the following steps:
   a. List all the volumes created using `docker volume ls` command.

      Example:

      ```
      cps@sk-persistence-db-1:~$ docker volume ls
      DRIVER              VOLUME NAME
      local               4b1e0622f0d774003c14eec9f17b98035445ef15f34fc055ebeb1aad572f1de3
      local               7d7d3b4b0ec4773427530b987475861106bc6d56dfc107feabce3f6c2afda875
      local               82e48b69a9b6687e54e8048aff2cc7af81c6754b1c44830f3e051e0fcfaaf380
      local               cps-docker-info
      local               cps-keepalived-conf
      local               cps-keepalived-service-status
      local               cps-keepalived-supervisord
      ```

```
local               d7b892a933a915774ac1c883c21b70bddfea002d46dfa457dbda5f1baa0af55e
local               faaeea115326b2981bd09f4f53fce157c1c6f327f7425ea27f92d5a371d8fcee
```

Execute the following commands only for name containers. For example, cps-docker-info.

b. Inspect volume using `docker inspect <volume_name>` command.

```
cps@sk-persistence-db-1:~$ docker inspect cps-docker-info
[
    {
        "CreatedAt": "2020-07-31T10:28:06Z",
        "Driver": "local",
        "Labels": {
            "com.broadhop.orchestrator.id": "test-system"
        },
        "Mountpoint": "/var/lib/docker/volumes/cps-docker-info/_data",
        "Name": "cps-docker-info",
        "Options": {
            "device": "tmpfs",
            "o": "size=64m",
            "type": "tmpfs"
        },
        "Scope": "local"
    }
]
```

c. You can delete the volumes where **labels** is null using `docker volume rm <volume_name>` command.

Example:

```
docker volume rm cps-docker-info
```

Volume not attached to any other volume will be safely deleted.

4. Reboot VMs.

```
sudo reboot
```

5. Verify whether VM is joined to master VM and all the containers are scheduled as expected using `show system status` and `show docker engine` commands from CLI.

```
admin@orchestrator[sk-master-binding-0]# show system status
system status running true
system status upgrade false
system status downgrade false
system status external-services-enabled true
system status debug false
system status percent-complete 100.0
admin@orchestrator[sk-master-binding-0]# show docker engine
ID STATUS PINGS
----------------------------------------
sk-control-binding-0 CONNECTED 0
sk-control-binding-1 CONNECTED 0
sk-master-binding-0 CONNECTED 0
sk-persistence-db-1 CONNECTED 0
sk-persistence-db-2 CONNECTED 0
sk-persistence-db-3 CONNECTED 0
sk-persistence-db-4 CONNECTED 0
```

The system status percent-complete must be 100.0. The status of all the dockers must be connected.

show network ips command not Displaying all IPs

Issue: `show network ip` command is not able to fetch all IPs information.

Condition: The issue happens when setups are migrated from older software (like, 18.2.0 or 19.4.0) where `docker-host-info-monitor` support was not present.

Possible Cause: `docker-host-info-monitor` containers are not coming up automatically.

```
show docker service | tab | include docker-host-info-monitor
```

Solution: Make sure all the VMs are running `docker-host-info-monitor` container.

If the container is not running, perform the following steps:

1. Login to the master VNF VM using `ssh` or `console`.
2. Login to orchestrator containe using `docker exec -it orchestrator bash` command.
3. Stop orchestration-engine service using `supervisorctl stop orchestration-engine` command.
4. Remove **monitoring** modules from database using:

   ```
   mongo orchestration --eval 'db.modules.remove({"_id": /^monitoring/})'
   mongo orchestration --eval 'db.services.remove({"_id": /^monitoring/})'
   ```

5. Start orchestration-engine service using `supervisorctl start orchestration-engine` command.

Wrong tmpfs Partition Names displayed in Mongo Containers

Issue: After downgrade tmpfs partitions names are not displayed correctly in mongo containers.

Scenarios:

- Downgrade from 20.2.0 release with tmpfs partition changes to tmpfs without changes
- Downgrade from 20.2.0 release to an older release

Condition: After completing downgrade, check whether `mongo-s<instance-id>` containers have unused tmpfs-partitions using the following command from CLI:

```
docker exec mongo-s1 "df -h"
```

Run the steps mentioned in the Solution if any container output shows `shard` word in tmpfs partitions names.



Note tmpfs partition names should only have port numbers.

Here is a sample output.

```
==========output from container mongo-s105===========
tmpfs           8.3T     0  8.3T   0% /mmapv1-tmpfs-rs-shard-41
tmpfs           8.3T     0  8.3T   0% /mmapv1-tmpfs-rs-shard-21
tmpfs           8.3T     0  8.3T   0% /mmapv1-tmpfs-rs-shard-42
tmpfs           8.3T     0  8.3T   0% /mmapv1-tmpfs-rs-shard-22
tmpfs           8.3T     0  8.3T   0% /mmapv1-tmpfs-rs-shard-2
tmpfs           8.3T     0  8.3T   0% /mmapv1-tmpfs-rs-shard-1
```

Solution:

1. Create a file `cleanup-tmpfs-partitions` name and add the following contents.

```
#!/bin/bash
for CONTAINER in $(/var/broadhop/cli/list-containers mongo-s); do
  if [ ${CONTAINER} == "mongo-status" ];then
    continue
  fi
  ENGINE=`/var/broadhop/cli/resolve-container-engine ${CONTAINER}`
  if [ ! -z "${ENGINE}" ]; then
    DOCKER_CONNECTION=`/var/broadhop/cli/resolve-docker-connection ${ENGINE}`
    if [ ! -z "${DOCKER_CONNECTION}" ]; then
       echo "container name... ${CONTAINER}"
       docker -H ${DOCKER_CONNECTION} cp /var/broadhop/cli/cleanupTmpfs.sh
${CONTAINER}:/usr/local/bin/
      docker -H ${DOCKER_CONNECTION} exec -d ${CONTAINER} /usr/local/bin/cleanupTmpfs.sh

    fi
  fi
done
```

2. Create a file `cleanupTmpfs.sh` name and add the following contents.

```
#!/bin/bash

count=$(df -h | grep shard | wc -l)
echo "partitions to delete.... ${count}"
while [ ${count} -ne 0 ]
do
echo "TRYING TO UNMOUNT try....${count}"
  for PARTITION in $(df -h | grep shard | tr -s ' '| cut -f 6 -d ' '); do
    echo "trying to unmount ${PARTITION}"
    umount -f ${PARTITION}
    rm -rf ${PARTITION}
  done

sleep 3
count=$(df -h | grep shard | wc -l)
done
```

3. Change the permissions for both the files.

```
chmod +x cleanup-tmpfs-partitions
chmod +x cleanupTmpfs.sh
```

4. Copy the files to an orchestrator container.

```
docker cp cleanupTmpfs.sh orchestrator:/var/broadhop/cli/
docker cp cleanup-tmpfs-partitions orchestrator:/var/broadhop/cli/
```

5. Execute `cleanup-tmpfs-partitions` command.

```
docker exec -it orchestrator /var/broadhop/cli/cleanup-tmpfs-partitions
```

6. Verify the cleanup of tmpfs-partitions by executing the following command from CLI.

```
docker exec mongo-s1 "df -h"
```

Make sure that there is no tmpfs partitions with name `shard` word present in any of the container outputs. tmpfs partition names should only have port numbers.

```
==========output from container mongo-s105===========
tmpfs            26G   49M   26G   1% /mmapv1-tmpfs-27026
tmpfs            26G   49M   26G   1% /mmapv1-tmpfs-27027
tmpfs            26G   49M   26G   1% /mmapv1-tmpfs-27036
tmpfs            26G   49M   26G   1% /mmapv1-tmpfs-27037
tmpfs            26G   17M   26G   1% /mmapv1-tmpfs-27046
```

```
tmpfs            26G   49M   26G   1% /mmapv1-tmpfs-27041
tmpfs            26G   49M   26G   1% /mmapv1-tmpfs-27042
admin@orchestrator[an-dbmaster]#
```

Binding/Endpoint Containers not Visible

Issue: Binding/endpoint containers aren't displayed when executing `show docker service` command.

Condition: Discrepancies with the blades hosting VMs.

Solution: Execute `show docker engine` command. All the VMs must be in connected state. If any VM isn't CONNECTED, make sure to bring that VM UP.

Check the blade status (by accessing through vSphere) which is hosting the impacted VMs. Make sure that the blade is up and running fine.

Refer to the following additional details:

- Orchestrator controls scheduling of the containers and also manages the high availability. If orchestrator goes down, system can't perform any activity.
- Consul is responsible for service discovery. It also shares the service details and state for health checks. If consul cluster isn't running properly, then VNF lacks these features supported by consul.
- Admin DB holds backup of Grafana, SVN. In addition, it also maintains the customer-specific reference data which is used to serve TPS processing. If admin database isn't available, it creates discrepancies in the call flows.
- Policy Builder provides GUI to add/update the policy configurations. It displays information such as peer monitoring, CRD table contents. If PB isn't working, you're not able to perform any configurations.

VM Stuck in JOINING State

Issue: VM is stuck in JOINING state.

Analysis: From CLI, execute the following commands to display VMs status:

```
admin@orchestrator[Master-hostname]# show docker engine | tab | exclude CONNECTED
an-pers-db-0    JOINING  234
an-pers-db-1    JOINING  452
```

The following are possible reasons for VM to get stuck in JOINING state.

- VM is not reachable from master VM.
- Timestamp of VM having issue is not in sync with the master VM.

Solution 1 :

1. If the VM is not reachable from master VM, reboot the VM.

   ```
   sudo reboot
   ```

2. If the timestamp of VM having issues is not in sync with the master VM, reboot the VM.

   ```
   sudo reboot
   ```

3. Verify whether the VM is moving to CONNECTED state.

Solution 2: After reboot if the VM is still in the JOINING state, peform the following steps:



Note The following steps must not be executed on master VM.

1. Remove old containers/volumes from the VM.

   ```
   docker stop $(docker ps -aq)
   docker rm -f $(docker ps -aq)
   sudo rm -rf /data/orchestrator
   sudo rm /var/cps/bootstrap-status
   ```

   For more information, refer to Docker Volumes Deletion, on page 41 section.

2. Execute bootstrapping command.

   ```
   sudo /root/bootstrap.sh
   ```

Consul Failing during DRA Downgrade or Upgrade

Issue: Consul containers fail when DRA is downgraded or upgraded using the following command:

`system downgrade version <old-dra-version>`

`system upgrade version <new-dra-version>`

Condition: Incompatible consul version was used in old or a new DRA image.

Solution:

1. Monitor consul containers using `monitor log container <consul-container-name>` command.

   Example:

   ```
   monitor log container consul-1
   .....
   monitor log container consul-7
   ```

2. Check for the following error in the consul container logs.

   ```
   Error starting agent: Failed to start Consul server: Failed to start Raft:
   failed to load any existing snapshots
   ```

3. Use the following CLI to stop the consul-server service across all the containers.

   ```
   docker exec consul- "supervisorctl stop consul-server"
   ```

4. Login to the consul container having errors using `docker connect <consul-container-name>` command.

   ```
   admin@orchestrator[site1-master-1]# docker connect consul-1
   root@consul-1:/#
   ```

5. Run `supervisorctl status` command from the consul container you logged in Step 4, on page 49.

   ```
   root@consul-1:/#supervisorctl status
   consul-server                    FATAL     Exited too quickly (process log may have
   details)
   root@consul-1:/#
   ```

6. Clear the consul state using the following commands.

```
root@consul-1:/# rm -rf /data/*
```

7. .Login using the CLI command and start the consul-server service across all consul containers.

```
docker exec consul- "supervisorctl start consul-server"
```

Repeat Step 4, on page 49 to Step 6, on page 49 for all the consul containers that have failed.

Grafana Loading Issue when Peer Number is High

Issue: When the number of peers connected to DRA is high (for example, 1500 peer connections), Grafana is slow (> 2 minutes) in rendering peer statistics. The delay in loading peer statistics increases the overall load time of the dashboard. For longer time ranges (for example, 24 hours), peer statistics charts could fail to load.

If **Traffic By Peers** panel is collapsed/minimized, dashboard loads within acceptable time.

Cause: With large number of peer connections (for example, 1500), volume of data queried to render peer statistics charts increases. This increases the time to load data and render the charts.

Solution: Load time of peer statistics charts can be improved by increasing the interval of data rendered in the charts. Duplicate the peer statistics query and create two additional queries. Change the minimum interval for the new queries to 5 mins and 10 mins respectively.

To analyze peer statistics for time ranges 1 - 6 hours, enable the query with 5 mins interval. To analyze peer statistics for time ranges > 6 hours, enable query with 10 mins interval.

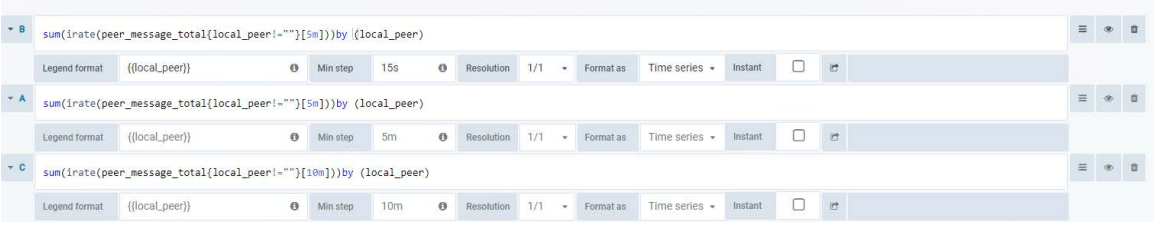
To avoid delays in loading Application Summary:

- By default, all the panels (Local Peer Traffic / Remote Peer Traffic / Active Peer List / Peer Response Time / Peer Connections) under **Application Summary** > **Traffic By Peers** are disabled.
- The panels are moved under new dashboard **Peer Traffic Monitor**.
- Recording rules (local_peer_sum_rate, remote_peer_sum_rate, active_peers_list) are added to precompute the data and serve whenever this data is queried. By default, **Trending and Planning** data is disabled for these new recording rules.

When high number of peers is configured:

- Use the default expressions to query data upto max of last 2 days.
- Use the old expressions to query data beyond the last 2 days (Trending / Planning). It is recommended to use old expressions for smaller time chunks (< 6hrs) to avoid delays in loading Grafana.

Figure 14: Grafana Queries



ADMIN User Opens in Readonly Mode

Issue: After login to DRA central, admin user opens in **Readonly** mode. Also, while importing PB in central DRA user gets error code 403.

Error has the following format:

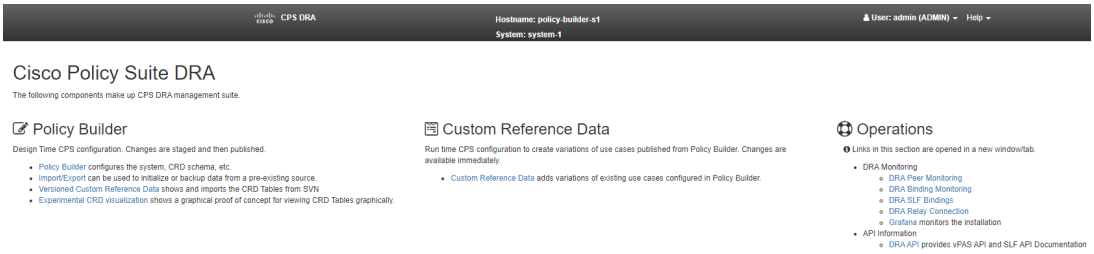
```
Status Code: 403
Response: 403 user admin is not allowed to perform post operation on the resource.
```

Cause: Admin user opens in readonly mode due to the invalid URL which was configured while creating the new repository.

Solution:

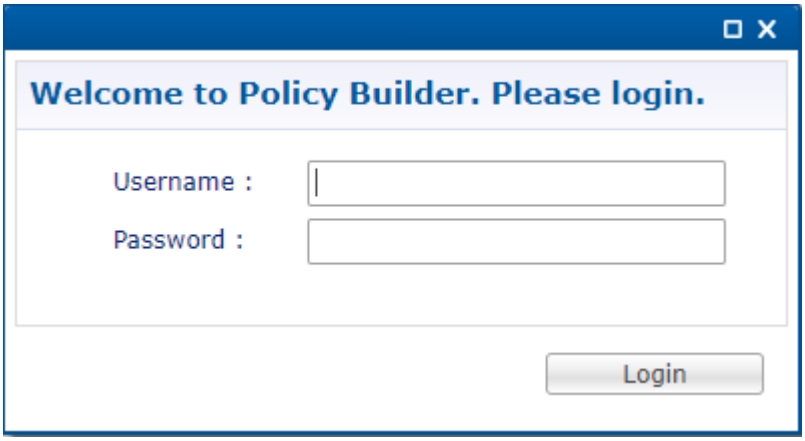
1. Login to the CPS central using `<master-ip>/central/dra/`.
2. Click **Import/Export** under Policy Builder.

   Figure 15: Import/Export

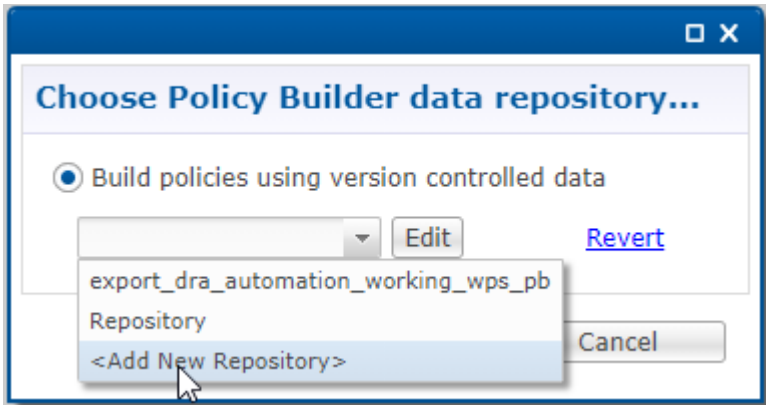
3. Click **Import** tab.
4. Click **File to Import...** and browse to the file to be imported. Once you select the file, the text field under **`Import URL: This URL will be updated/created. It is strongly suggested to import to a new URL and use Policy Builder to verify/publish.`** is updated.
5. Click **Import** to import the file to CPS Central.
6. Go to `<master-ip>pb` to open Policy Builder GUI.

Figure 16: Policy Builder - Login Screen



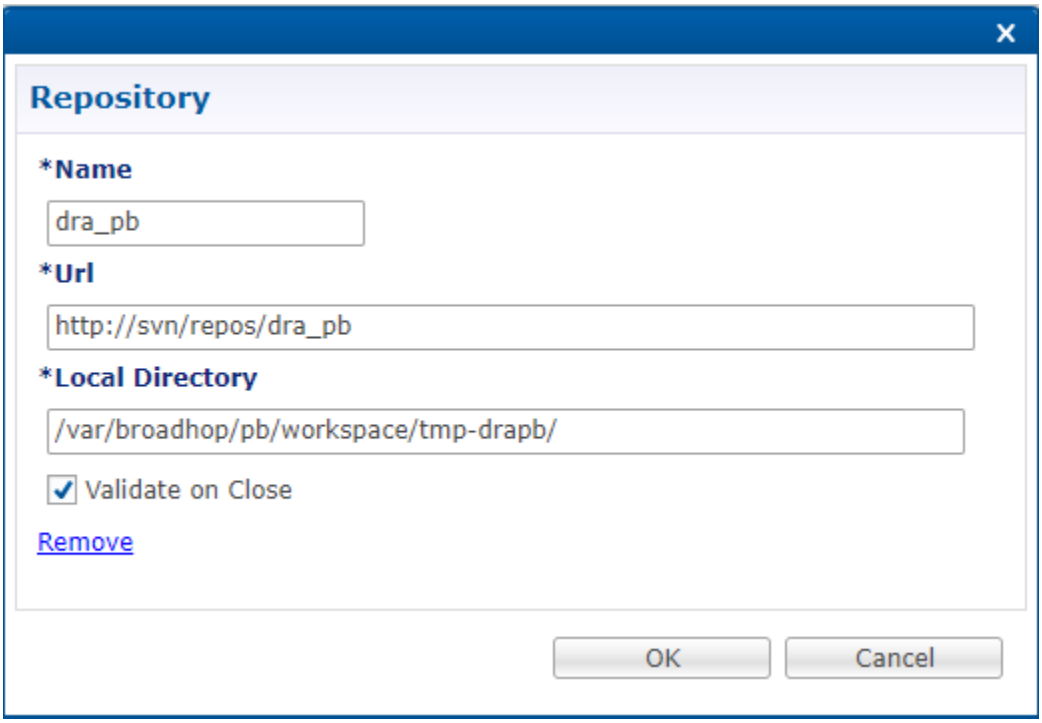
7. Enter the Username and Password and click **Login** to open **Choose Policy Builder data repository...** window.

Figure 17: Add New Repository



8. Click **<Add New Repository>** from the drop-down list to open **Repository** window.

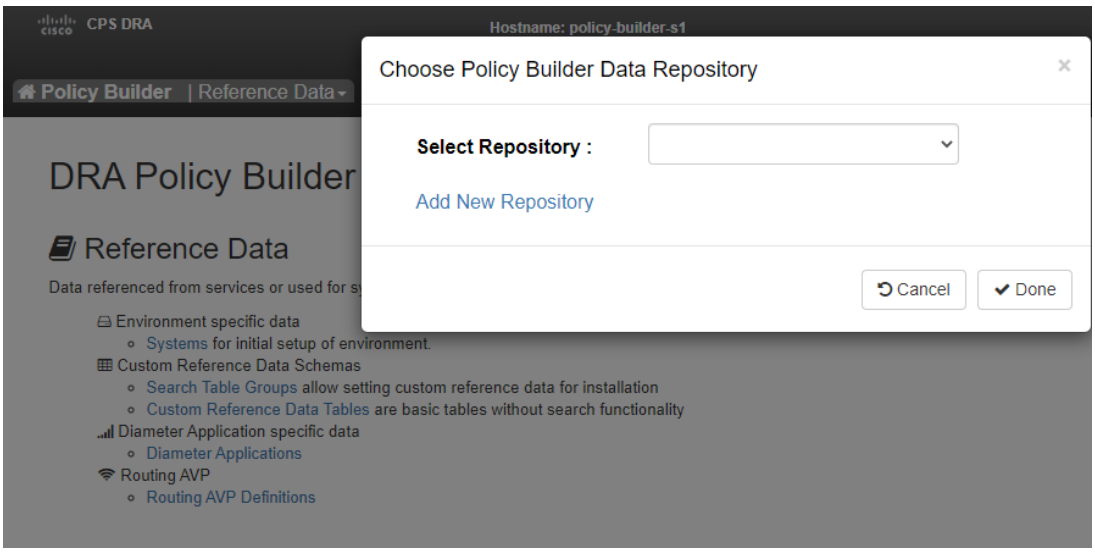
Figure 18: Repository Parameters



Note Use the correct URL while adding this. This URL must be same as in Step 4, on page 51.

9. Login to `<master-ip>/central/dra`. Click **Policy Builder** and **Choose Policy Builder Data Repository** windows opens up.

Figure 19: Choose Policy Builder Data Repository



10. Select the repository you created in Step 8, on page 52 from the **Select Repository** drop-down list and click **Done**.
11. You can now create/update/modify the configurations based on your requirements.

Database IPs not Reachable

Issue: Worker VM binding containers are not able to connect with Binding VNF database IPs.

Condition: Call running on Site1 and Site2. Powered OFF Site2. Call shifted to Site1 and the system status was OK.

Solution: Execute the following commands to solve the issue:

```
docker stop $(docker ps -aq)
docker rm -f $(docker ps -aq)
sudo rm -rf /data/orchestrator
sudo rm /var/cps/bootstrap-status
sudo /root/bootstrap.sh
```

Shard Count Displaying Incorrect Primary

Issue: Shard count is displaying more than one primary.

Condition: This happens when two primaries are encountered for the same shard in `show database status` command.

Solution: The following solution is to fix the database showing two primaries for the same shard, which is an edge scenario and is encountered when containers are started (upgrade, VM restarted or fresh installation):

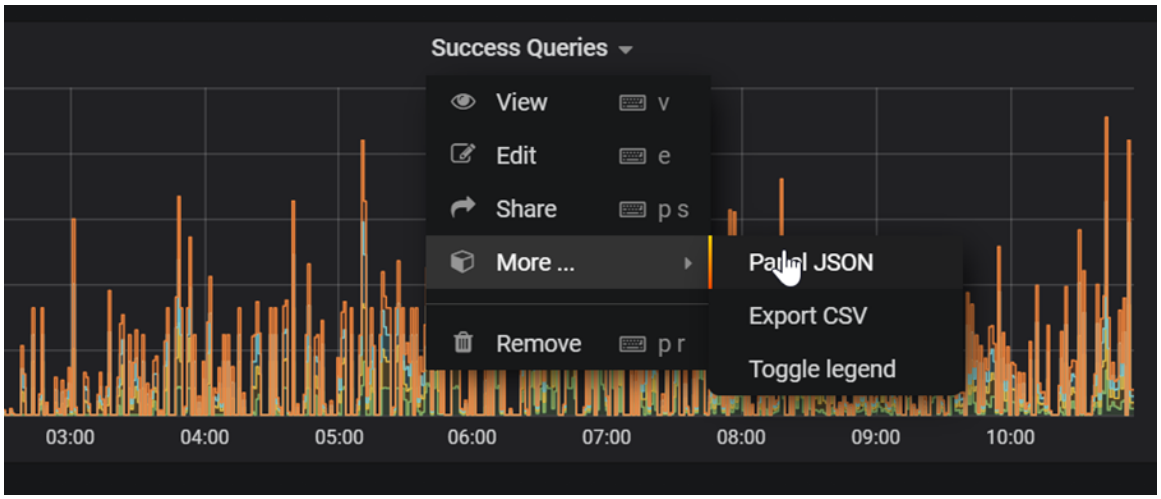
Let `shard-<ip>-<port>-MMAPv1` be the shard process running for primary in the shard, then:

```
cps@vpas-site-persistence-db-*:~$ docker exec -it mongo-s* bash
root@mongo-s106:/# supervisorctl stop shard-<ip>-<port>-MMAPv1
root@mongo-s106:/# rm -rf mmapv1-tmpfs-<port>/*
root@mongo-s106:/# supervisorctl start shard-<ip>-<port>-MMAPv1
```

Missing Spikes in Longevity Report

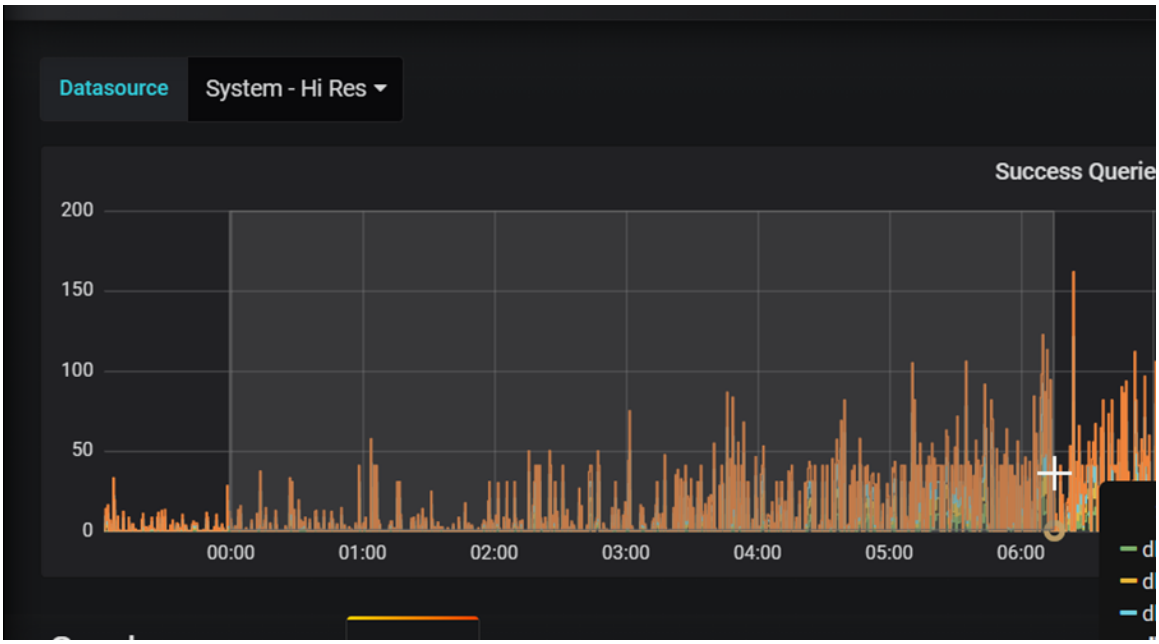
Issue: Missing spikes in longevity report for 12 hours and more.

Solution: Change panel width to maximum value (24). The following screenshots are for reference purpose.





You can get all the spikes captured for 6 hours duration. So, if you need to analyse longevity report for 12 hours or more, you can grep data by grouping in 6 hours interval. The following screenshots are for reference purpose.





Jetty Server Issue Reported in Logs

Issue: In `consolidated-qns.log` warning logs `GzipFilterCustom un-expected event occurred : Committed` is displayed followed by any URL.

Condition: Jetty server causes this exceptions. It's related to the working of the javax.servlet.http classes. This error can occur for many User Interfaces such as, API, CSS, JSS, gif, png fetching and so on.

Precautions: Close or refresh the associated DRA GUI URL page which is open in the browser for a long time.

Solution: Identify the URL which is mentioned in the logs and try to close or refresh the browser page associated with URL.

For example: `GzipFilterCustom un-expected event occurred Committed : https://10.197.97.87/proxy/dra/api/localActivePeerEndpoints`.

In this case, DRA Peer Monitoring page is open for long time, hence it is throwing java.lang.IllegalStateException. Close the page or refresh the page.

Scheduling Paused at cc-monitor after vmdk Redeployment

Issue: After fresh installation, scheduling is stuck at cc-monitor due to delay in insertion for license in mongo-admin.

Workaround: Execute the following to resolve this issue.

```
cps@vpas-A-master:~$ docker exec -it mongo-admin-a bash
root@mongo-admin-a:/# mongo
rs:PRIMARY> use sharding
switched to db sharding
rs:PRIMARY> show collections
licensedfeats
rs:PRIMARY> db.licensedfeats.insert({ "_id" : ObjectId("620a232db0f6e75a85c26697"),
"featurename" : "SP_CORE", "licatt" :
"3FC5CA0ED1FCD8DABE3B9B24078DB749B0CBAA5A686436B234B53A3BDEF7676AEF777
CD2FD215B2BDE0E9392D417F8D0F2D162A6CEA04E924D2432BCC6B006D1AF98B9A9EACAFEF721DECB4C4195C020"
 })
WriteResult({ "nInserted" : 1 })
rs:PRIMARY> db.licensedfeats.find()
{ "_id" : ObjectId("620a232db0f6e75a85c26697"), "featurename" : "SP_CORE", "licatt" :
 "3FC5CA0ED1FCD8DABE3B9B24078DB749B0CBAA5A686436B234B53A3BDEF7676AEF777
CD2FD215B2BDE0E9392D417F8D0F2D162A6CEA04E924D2432BCC6B006D1AF98B9A9EACAFEF721DECB4C4195C020"
 }
```

Login to the CLI to view the following outputs.

```
docker exec cc-monitor "supervisorctl restart app"
==========output from container cc-monitor-s102===========
app: stopped
app: started
==========output from container cc-monitor-s103===========
app: stopped
app: started
```

System Status Percent Stuck after Fresh Installation

Issue: After DRA fresh installation, not able to login in CLI or `system status percent` stuck. This issue is observed on both DRA or Binding VNFs after fresh installation.

Workaround: The following steps should be executed to resolve this issue.

APP VNF

node: DRA Master

user: cps

```
cps@${DRM-hostname}:~$ df -h
```

If `/data, /stats` directory is missing, execute the commands to resolve the issue:

```
cps@${DRM-hostname}:~$ sudo -i
cps@${DRM-hostname}:~$ rm -rf /var/lib/cloud/*
cps@${DRM-hostname}:~$ rm /var/cps/bootstrap-status
cps@${DRM-hostname}:~$ reboot
```

Check `/data`, `/stats` directory status for other DRA components (master, control). If the directory is missing, execute the following commands.

```
cps@${DRM-hostname}:~$ sudo -i
cps@${DRM-hostname}:~$ rm -rf /var/lib/cloud/*
cps@${DRM-hostname}:~$ rm /var/cps/bootstrap-status
cps@${DRM-hostname}:~$ reboot
```

DB VNF

node: DRA DB Master

user: cps

```
cps@${DRM-hostname}:~$ df -h
```

If `/data, /stats` directory is missing, execute the commands to resolve the issue:

```
cps@${DRM-hostname}:~$ sudo -i
cps@${DRM-hostname}:~$ rm -rf /var/lib/cloud/*
cps@${DRM-hostname}:~$ rm /var/cps/bootstrap-status
cps@${DRM-hostname}:~$ reboot
```

Check `/data`, `/stats` directory status for other database components (master, control). If the directory is missing, execute the following commands.

```
cps@${DRM-hostname}:~$ sudo -i
cps@${DRM-hostname}:~$ rm -rf /var/lib/cloud/*
cps@${DRM-hostname}:~$ rm /var/cps/bootstrap-status
cps@${DRM-hostname}:~$ reboot
```

SVN Error: Pristine Text not Present

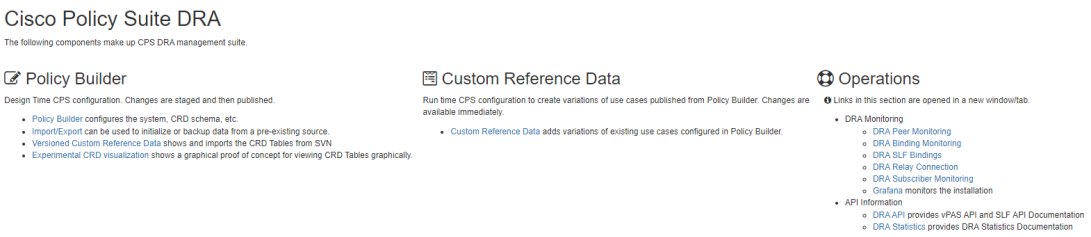
Issue: The following error may occur when **Import/Publish** operation is performed in Policy Builder.

```
Pristine text \xxxxxxxxxxxxx\xxxxxxxxxxxxx not present\n
```

Solution: During import/publish, the repository is corrupted. You can create a new repository with PB backup and publish it with the desired changes.

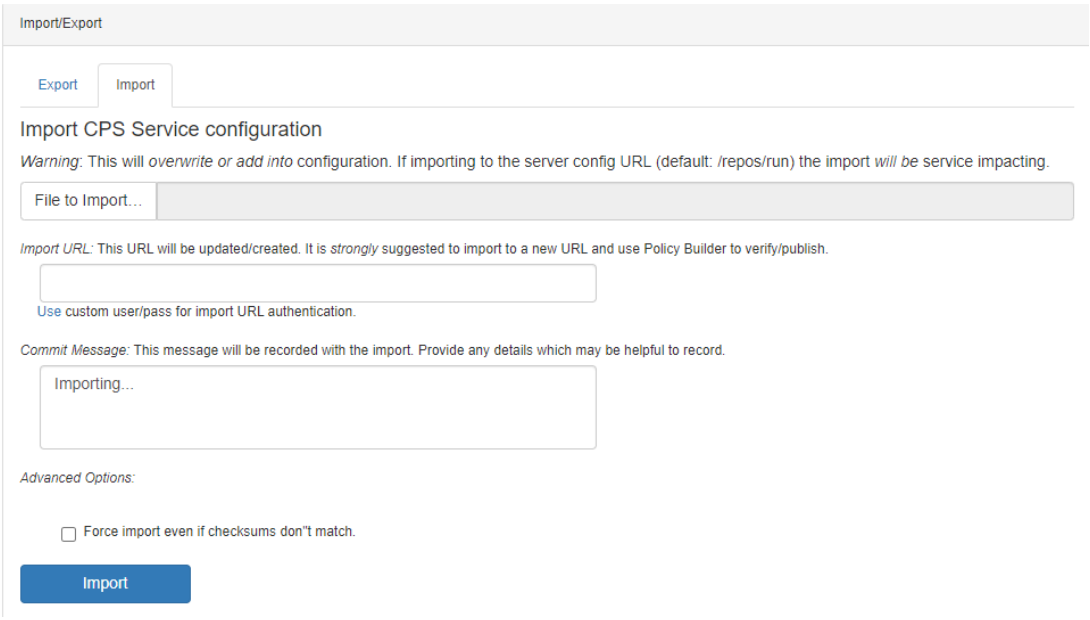
1. Log in to CPS Central using *<master-ip>/central/dra/*.

   Figure 20: Main Page

2. Click **Import/Export** under Policy Builder.
3. Select **Import** tab. Click **File to Import** and select the PB backup file to import.

Figure 21: Import/Export

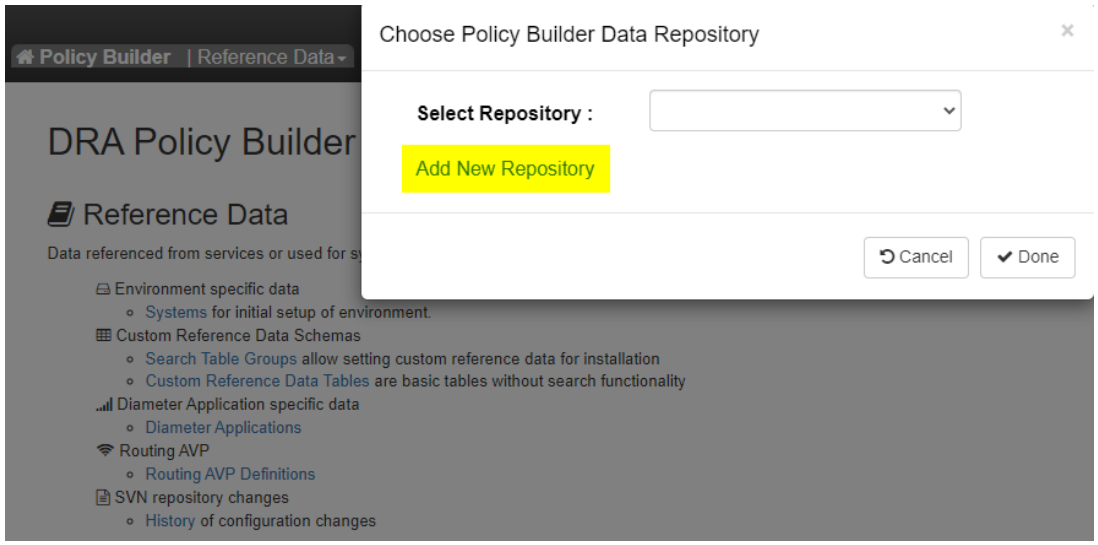


4. Enter the *Import URL* .

Note It is strongly suggested to import to a new URL and use Policy Builder to verify/publish).

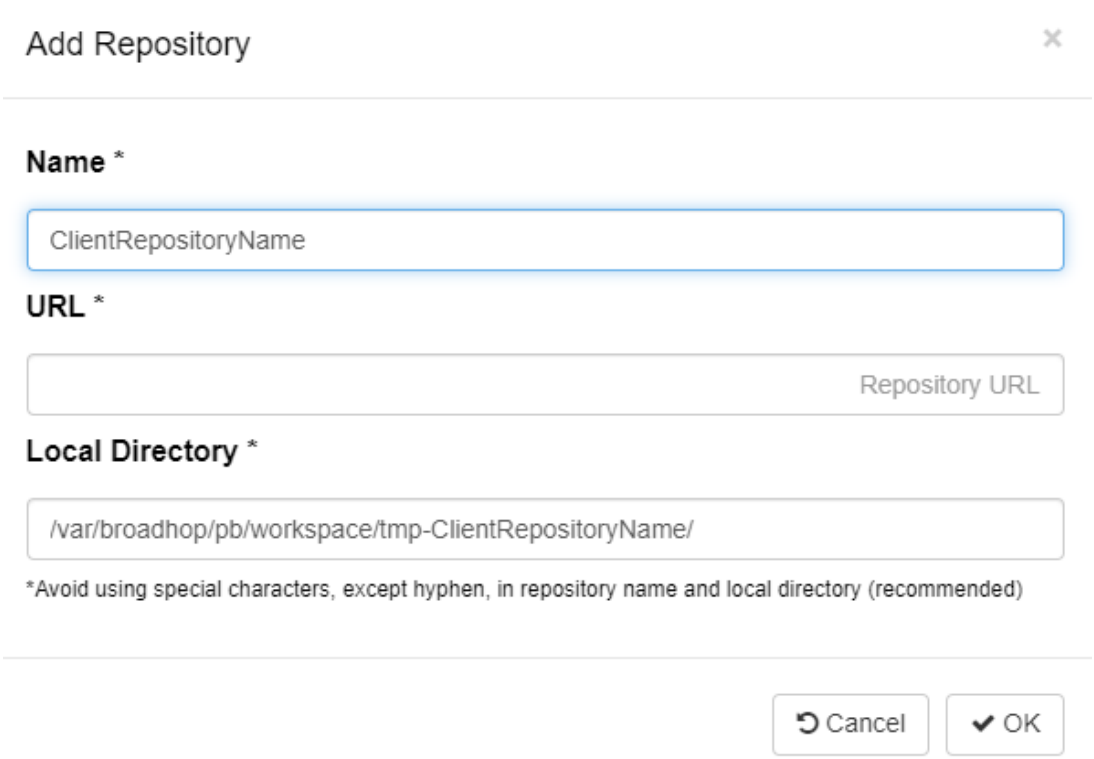
5. Click **Import** to import the file.
6. From main page, click **Policy Builder** to open **Choose Policy Builder Data Repository**.

Figure 22: Choose Policy Builder Data Repository



7. Click **Add New Repository** to open **Add Repository** window.

Figure 23: Add Repository



Enter the name of the **Repository**, **URL**, and **Local Directory**.

Note Use the correct URL. This URL should be same as added in Step 3, on page 58.

8. Log in again to CPS Central using *<master-ip>/central/dra/*. Click on Policy Builder and select your newly created repository.
9. Edit the Policy Builder configuration with the changes which were done for last corrupted repository to resolve the corrupted repository issue.
10. Save the changes and publish the updated configuration.

Unreachable Peers with Redeployment

Issue: After redeployment or power ON/OFF of VMs (also observed in resiliency tests), system intermittently comes up as not healthy and VMs can move to JOINING state. In such case, weave status is displayed as `waiting for IP range grant from peers` and `weave status ipam` command shows unreachable MACs for few VMs (same VM has 2 MACs reachable and unreachable).

Solution: After each of the above scenarios (redeployment/resiliency), if any unreachable MACs are seen on VMs, perform the following steps on the same VM:

1. Remove peers that are unreachable.

```
weave rmpeer <unreachable mac id>
```

2. Bootstrap the VM.

```
docker stop $(docker ps -aq)
docker rm -f $(docker ps -aq)
sudo rm -rf /data/orchestrator
sudo rm /var/cps/bootstrap-status
sudo /root/bootstrap.sh
```

User Role Changes to Readonly

Issue: After login to DRA Central GUI with valid admin credentials, sometimes admin role is changed to read-only.

Solution: The following workarounds should be applied in sequential manner.



Note Take backup of SVN repository before peforming any operation on SVN.

Workaround1: Check the disk space. The user in ReadOnly mode can also be due to the disk full issue on Control VMs. Free some space to recover from this problem.

Workaround2: Browser cache issue.

This workaround should be applied only if Workaround1 doesn't work.

1. Open DRA Central GUI in another browser.

   Or

   Delete the browser cache and restart the browser.

2. Login into DRA Central GUI again with valid admin credentials. You should have admin rights.

Workaround3: `.broadhopFileRepository` missing.

This workaround should be applied only if Workaround2 doesn't work.

1. Check `.broadhopFileRepository` in the client SVN repository.

   For example, current published repo URL is `https://<master-ip>/repos/configuration/ .`

2. Go to `<master-ip>/repos/configuration` in browser and check if `.broadhopFileRepository` file is missing.

   If `.broadhopFileRepository` file is missing, use the following steps to add `.broadhopFileRepository` file.

   a. Log in into SVN container using the `docker exec -it svn bash` command.

      This command works from control VM where SVN container exists.

   b. Create a blank `.broadhopFileRepository` file using the `touch .broadhopFileRepository` command.

   c. Import the file into repository using the following command.

```
svn import --username <username> --password <password> --force -m "adding broadhop
file" --no-auth-cache .broadhopFileRepository
http://svn/repos/configuration/.broadhopFileRepository
```

3. Re-login to DRA Central GUI with valid admin credentials. You should now have admin rights.

Workaround4: Delete `.broadhopFileRepository` file and add the file again.

This workaround should be applied only if Workaround1 and Workaround3 doesn't work.

1. Log in into SVN container using `docker exec -it svn bash` command.

   This command works from control VM where SVN container exists.

2. Take backup of SVN repository.

3. Considering that you are using `…/configuration` SVN URL, delete `.broadhopFileRepository` file.

   If you are not using `…/configuration` SVN URL, change URL accordingly and use the following command to delete the file.

   ```
   svn delete --username <username> --password <password> --force --no-auth-cache
   http://svn/repos/configuration/.broadhopFileRepository --message "deleting file"
   ```

4. If you have backup of `.broadhopFileRepository` file, import the file in the repository using the following command:

   ```
   svn import --username admin --password admin --force -m "adding brodhop file"
   --no-auth-cache .broadhopFileRepository
   http://svn/repos/configuration/.broadhopFileRepository
   ```

   If you don't have backup of `.broadhopFileRepository` file, create a blank file with name `.broadhopFileRepository` and import the file in the repository using the following command:

   ```
   svn import --username admin --password admin --force -m "adding brodhop file"
   --no-auth-cache .broadhopFileRepository
   http://svn/repos/configuration/.broadhopFileRepository
   ```

5. Re-login to DRA Central GUI with valid admin credentials. You should now have admin rights.

vDRA Database Troubleshooting

This section provides the information about vDRA database troubleshooting in Binding VNFs:

Note All commands under this section needs to be executed from Binding VNF CLI.

Database Operational Status

The following command provides database operational status of all database clusters configured. Execute the command in operational mode.

```
show database status | tab
```

Example:

```
admin@orchestrator[an-dbmaster]# show database status | tab
                                                                CLUSTER
ADDRESS         PORT   NAME        STATUS     TYPE         NAME     SHARD     REPLICA SET
-------------------------------------------------------------------------------------------------
192.168.11.42  27026  arbiter-21  ARBITER    replica_set  session  shard-21  rs-shard-21
192.168.11.43  27026  server-x    PRIMARY    replica_set  session  shard-21  rs-shard-21
192.168.11.44  27026  server-y    SECONDARY  replica_set  session  shard-21  rs-shard-21
192.168.11.42  27027  arbiter-22  ARBITER    replica_set  session  shard-22  rs-shard-22
192.168.11.43  27027  server-x    SECONDARY  replica_set  session  shard-22  rs-shard-22
192.168.11.44  27027  server-y    PRIMARY    replica_set  session  shard-22  rs-shard-22
192.168.11.43  27019  session     PRIMARY    shard_db     session  shdb-4    session-sharddb
192.168.11.44  27019  session     SECONDARY  shard_db     session  shdb-5    session-sharddb

admin@orchestrator[an-dbmaster]#
```

Validate Sharding Database Metadata

1. Execute the following command to find sharding database PRIMARY of particular cluster.

   ```
   show database status cluster-name session | tab | include PRIMARY | include shard_db
   ```

2. Connecting to sharding database primary member.

   Non-Mongo Auth:

   ```
   mongo --ipv6 mongodb://[2606:ae00:2001:230b::2b]:27019
   ```

   Mongo Auth:

   ```
   mongo --ipv6 mongodb://adminuser:<password>@[2606:ae00:2001:230b::2b]:27019/admin
   ```

3. After successfully connecting to sharding database primary member, execute the following step:

   For example, to validate DRA sessions sharding database metadata information:

   ```
   session-sharddb:PRIMARY> use drasessionsShardDB
   switched to db drasessionsShardDB
   session-sharddb:PRIMARY> db.shards.count()
   2
   session-sharddb:PRIMARY> db.buckets.count()
   8192
   session-sharddb:PRIMARY>
   ```

Validate Zone Aware Sharding Database Metadata

1. Execute the following command to find sharding database PRIMARY of particular cluster:

   ```
   show database status cluster-name session | tab | include PRIMARY | include shard_db
   ```

2. Connecting to sharding database primary member.

   Non-Mongo Auth:

   ```
   mongo --ipv6 mongodb://[2606:ae00:2001:230b::2b]:27019
   ```

   Mongo Auth:

   ```
   mongo --ipv6 mongodb://adminuser:<password>@[2606:ae00:2001:230b::2b]:27019/admin
   ```

3. Validate configured shard and zone mappings.

   Example:

```
use ipv6ShardDB
switched to db ipv6ShardDB
binding-sharddb:PRIMARY>
binding-sharddb:PRIMARY> db.shards.find()
```

```
binding-sharddb:PRIMARY> db.shards.find()
{ "_id" : 1, "name" : "shard-1", "hosts" : "182.22.31.13:27017,182.22.31.14:27017", "zone" : "mumbai" }
{ "_id" : 2, "name" : "shard-2", "hosts" : "182.22.31.13:27018,182.22.31.14:27018", "zone" : "pune" }
{ "_id" : 3, "name" : "shard-3", "hosts" : "182.22.31.13:27020,182.22.31.14:27020", "zone" : "hyd" }
{ "_id" : 4, "name" : "shard-4", "hosts" : "182.22.31.13:27021,182.22.31.14:27021", "zone" : "bglr" }
{ "_id" : 5, "name" : "shard-5", "hosts" : "182.22.31.13:27022,182.22.31.14:27022", "zone" : "chennai" }
{ "_id" : 6, "name" : "shard-6", "hosts" : "182.22.31.13:27023,182.22.31.14:27023", "zone" : "hyd" }
{ "_id" : 7, "name" : "shard-7", "hosts" : "182.22.31.13:27024,182.22.31.14:27024", "zone" : "bglr" }
{ "_id" : 8, "name" : "shard-8", "hosts" : "182.22.31.13:27025,182.22.31.14:27025", "zone" : "pune" }
binding-sharddb:PRIMARY>
```

4. Validate configured zones and ranges.

   Example:

```
binding-sharddb:PRIMARY> db.zoneinfo.find()
{ "_id" : 1, "name" : "r1", "start" : "2017:6000:0000:0001", "end" : "2017:6000:0000:0500", "zone" : "bglr" }
{ "_id" : 2, "name" : "r2", "start" : "2018:6000:0000:0001", "end" : "2018:6000:0000:0500", "zone" : "bglr" }
{ "_id" : 3, "name" : "r1", "start" : "2013:6000:0000:0001", "end" : "2013:6000:0000:0500", "zone" : "chennai" }
{ "_id" : 4, "name" : "r2", "start" : "2014:6000:0000:0001", "end" : "2014:6000:0000:0500", "zone" : "chennai" }
{ "_id" : 5, "name" : "r1", "start" : "2015:6000:0000:0001", "end" : "2015:6000:0000:0500", "zone" : "hyd" }
{ "_id" : 6, "name" : "r2", "start" : "2016:6000:0000:0001", "end" : "2016:6000:0000:0500", "zone" : "hyd" }
{ "_id" : 7, "name" : "r1", "start" : "2008:5000:0000:0100", "end" : "2008:5000:0000:0500", "zone" : "mumbai" }
{ "_id" : 8, "name" : "r2", "start" : "2009:5000:0000:0100", "end" : "2009:5000:0000:0500", "zone" : "mumbai" }
{ "_id" : 9, "name" : "r1", "start" : "2011:6000:0000:0001", "end" : "2011:6000:0000:0500", "zone" : "pune" }
{ "_id" : 10, "name" : "r2", "start" : "2012:6000:0000:0001", "end" : "2012:6000:0000:0500", "zone" : "pune" }
```

5. Validate buckets/shard/range mapping.

   Example:

```
binding-sharddb:PRIMARY> db.buckets.find()
{ "_id" : ObjectId("5cb9845c63ebec62ea803d42"), "bucket-id" : 1, "shard" : 4, "migration" : false, "zone" : "bglr" }
{ "_id" : ObjectId("5cb9845c63ebec62ea803d43"), "bucket-id" : 2, "shard" : 4, "migration" : false, "zone" : "bglr" }
{ "_id" : ObjectId("5cb9845c63ebec62ea803d44"), "bucket-id" : 3, "shard" : 4, "migration" : false, "zone" : "bglr" }
{ "_id" : ObjectId("5cb9845c63ebec62ea803d45"), "bucket-id" : 4, "shard" : 4, "migration" : false, "zone" : "bglr" }
{ "_id" : ObjectId("5cb9845c63ebec62ea803d46"), "bucket-id" : 5, "shard" : 4, "migration" : false, "zone" : "bglr" }
{ "_id" : ObjectId("5cb9845c63ebec62ea803d47"), "bucket-id" : 6, "shard" : 4, "migration" : false, "zone" : "bglr" }
{ "_id" : ObjectId("5cb9845c63ebec62ea803d48"), "bucket-id" : 7, "shard" : 4, "migration" : false, "zone" : "bglr" }
```

MongoDB Authentication Validation

1. Make sure the database status of all the shards and sharding database members comes up with either PRIMARY, SECONDARY or ARBITER.

```
show database status | tab
```

2. If the database status is not PRIMARY, nor SECONDARY or ARBITER, login to specific VM and check whether mongod instance is running appropriate options to enable mongo authentication or not.

   Example for working mongod instance with authentication enabled on it:

```
root     15379     1  4 03:54 ?        00:16:49 mongod --keyFile=/mongodb.key
--storageEngine mmapv1 --nojournal --noprealloc --smallfiles --ipv6 --bind_ip_all
--port 27023 --dbpath=/mmapv1-tmpfs-27023 --replSet rs-shard-13 --quiet --slowms 500
--logpath /data/db/mongo-27023.log --oplogSize 3221 --logappend --logRotate reopen
```

3. If the key file is present, but the database status is not good, check whether user exists or not for that mongod instance.

   Example:

   a. Login to VM and its mongo container (container name : mongo-s<instance>).

   b. Connect to mongod with its port.

   c. Use admin user and execute the following command:

Note The `db.getUsers()` command should display adminuser and backupuser for all the non-aribiter members.

For arbiter member users, connect to PRIMARY of that shard and execute `db.getUsers()` command.

```
session-sharddb:PRIMARY> db.getUsers()
[
        {
                "_id" : "admin.adminuser",
                "user" : "adminuser",
                "db" : "admin",
                "roles" : [
                        {
                                "role" : "root",
                                "db" : "admin"
                        }
                ]
        },
        {
                "_id" : "admin.backupuser",
                "user" : "backupuser",
                "db" : "admin",
                "roles" : [
                        {
                                "role" : "root",
                                "db" : "admin"
                        }
                ]
        }
]
```

4. If users are present, check whether mongo connection gets established manually or not by executing the following command:

   Example for Mongo Auth:

   ```
   mongo --ipv6 mongodb://adminuser:<password>@[2606:ae00:2001:230b::2b]:27019/admin
   ```

5. To validate the configurations and operational status of mongod instance, execute the following commands:

   ```
   db-authentication show-password database mongo password
   db-authentication rolling-restart-status database mongo password
   ```

Repair CLI failure due to missing .pem file for db_user authentication

Issue: The MongoDB WireTiger DB Recovery script does not work because of the shared cps pem access logic.

Solution: Create a new user "db_user" across all the sites. Modify the recovery script to use db_user as default user when logging into each VM.

Execute these steps on each virtual machine to set the password and use a *.pem* file-based authentication for db_user.

```
sudo passwd db_user
ssh-keygen -t rsa
sudo cp db_user.pem.pub /home/db_user
su db_user
mkdir -p ~/.ssh
```

```
chmod 700 ~/.ssh
sudo cat ~/db_user.pem.pub >> ~/.ssh/authorized_keys
```

Reconfigure the Databases

Here are few conditions when database reconfiguration is needed:

- All database members must be in STARTUP2 state for one or more replica sets.

  In shard replica-set, if all the data bearing members are down at the same time and arbiter is still running and once they are UP, they fail to elect new Primary and gets stuck in STARTUP2 state.

- Change in database configuration.

Database Generic fcvcheck Scenarios

Case 1: mongo 4.2.20

```
database fcvcheck4
         <set/check>set
```

Case 2: mongo 4.4.18

```
database genericfcvcheck 4.2
         <set/check>set
```

Case 3: mongo 5.0

```
database genericfcvcheck 4.4
```

The following procedural steps are applicable until CPS 22.1.0 releases.

To reconfigure the databases, perform the following steps in the following sections:



Note The following steps are required for single cluster as well as multiple clusters. In case of mated pair deployments, the steps must be performed on all the sites.

Steps to be executed on DRA VNF

1. Login to vDRA VNF CLI and run `no binding shard-metadata-db-connection` command to remove shard metadata-db-connection configurations.

   Example:

   ```
   admin@orchestrator[an-master](config)# no binding shard-metadata-db-connection
      admin@orchestrator[an-master](config)# commit
      Commit complete.
      admin@orchestrator[an-master](config)# end
      admin@orchestrator[an-master]# show running-config binding
      % No entries found.
      admin@orchestrator[an-master]#
   ```

2. Login to vDRA VNF CLI and run `db-authentication remove-password database mongo` command to remove MongoDB password.

   Example:

```
admin@orchestrator[an-master]# db-authentication remove-password database mongo
   Value for 'current-password' (<string>): ********
   admin@orchestrator[an-master]#
   admin@orchestrator[an-master]# db-authentication show-password database mongo
   result Mongo password is not configured.
   admin@orchestrator[an-master]#
```

Steps to be executed on DRA Binding VNF

1. Edit the nacm rule (config mode) to allow deleting the database configurations.

   ```
   nacm rule-list any-group rule data-base access-operations delete action permit
   ```

2. Delete database configurations (config mode).

   ```
   no database cluster <cluster name>
   ```

   where, <*cluster name*> is the name of the database cluster which has issues or need to be reconfigured.

Note As there are multiple database clusters, remove all the clusters from the configuration.

3. Stop the system and wait for all the application containers to be stopped.

   ```
   system stop
   ```

4. Verify that the application containers are stopped by running `show scheduling status | include application` command. The command should not show any containers with issues.

5. (Only for MongoDB Authentication enabled database) Disable MongoDB authentication.

   ```
   db-authentication remove-password database mongo current-password XXX
   ```

6. To delete the persistent storage and old log information, run the following command on all VMs:

   ```
   rm -rf /data/mongod-node/db/*
   rm -rf /data/mongod-node/supervisord/supervisord-mongo.conf
   ```

   Example:

   ```
   $ for dbHost in `weave status connections | grep ss-persistence-db | awk '{print $2}'
    | cut -d':' -f1`;
   do ssh -o StrictHostKeyChecking=no -i cps.pem $dbHost 'sudo /bin/rm -fr
   /data/mongod-node/supervisord/supervisord-mongo.conf'; done
   ```

   ```
   $ for dbHost in `weave status connections | grep ss-persistence-db | awk '{print $2}'
    | cut -d':' -f1`;
   do ssh -o StrictHostKeyChecking=no -i cps.pem $dbHost 'sudo /bin/rm -fr
   /data/mongod-node/db/*'; done
   ```

Note Before proceeding to next step, make sure Step 1, on page 67 to Step 6, on page 67 has been executed from all the affected/reconfigured binding VNF sites.

7. (Only when arbiter is on other site) Clear data directory for all STARTUP2 replica sets on arbiter VM.

   a. Find container where particular arbiter member is running. On the arbiter site, run the following command to find arbiter hostname.

      ```
      show network ips | tab | include <arbiter-ipaddress>
      ```

b. Find a MongoDB container name for this host.

```
show docker service | tab | include <host-name> | include mongo-s
```

c. Connect to MongoDB container by running a `docker connect mongo-s<id>` command.

d. Clean the data directory. Stop all supervisor processes on arbiter container.

```
supervisorctl stop all
```

e. Clean database directory on arbiter container.

```
rm -fr /mmapv1-tmpfs-*/*
```

f. Start all supervisor processes on arbiter container.

```
supervisorctl start all
```

8. Start the system and wait for system percentage to turn 100%.

```
system start
```

`show system status` should show system percentage as 100.

9. Apply the database configurations again (config mode).

Before proceeding to next step, make sure Step 8, on page 68 to Step 9, on page 68 has been executed from all the affected/reconfigured binding VNFs sites.

10. Verify all the database sets are UP by running `show database status | exclude "PRIMARY|SECONDARY|ARBITER"` command.

Note `show database status | exclude "PRIMARY|SECONDARY|ARBITER"` command should not show any unhealthy database member.

Before proceeding to the next step, make sure Step 9, on page 68 to Step 10, on page 68 has been executed from all the affected/reconfigured binding VNFs sites.

11. (Only for MongoDB Authentication enabled database) Enable MongoDB authentication.

For more information on enabling MongoDB authentication, see *Configuring MongoDB Authentication* section in the *CPS vDRA Configuration Guide*.

Note Before proceeding to next step, make sure that this step has been executed from all the affected/reconfigured binding VNFs sites.

Example 1: Enable MongoDB authentication with transition.

```
db-authentication set-password database mongo password XXX confirm-password XXX
db-authentication enable-transition-auth database mongo
db-authentication rolling-restart database mongo
db-authentication rolling-restart-status database mongo
Verify database member in healthy state
```

Example 2: Enable MongoDB authentication without transition.

```
db-authentication disable-transition-auth database mongo
db-authentication rolling-restart database mongo
db-authentication rolling-restart-status database mongo
```

12. Wait for all the database sets to come UP. Verify the status by running `show database status | exclude "PRIMARY|SECONDARY|ARBITER"` command.

Note `show database status | exclude "PRIMARY|SECONDARY|ARBITER"` command should not show any unhealthy database member.

13. Log in to the vDRA VNF CLI and restart the binding containers to read the latest metadata from a new configurations.

Note Before proceeding to next step, make sure that this step has been executed from all the affected/reconfigured DRA VNFs.

```
docker restart container-id <container-id>
```

Example:

```
docker restart container-id binding-s<number>
```

`binding-s`<*number*> is an example for binding container. binding-s <*number*> is the container-id. <*number*> varies according to which worker VM binding container is running.

14. Disable deleting the database configurations by editing the nacm rule (config mode).

Note Make sure that this step has been executed from all the affected/reconfigured binding VNFs sites.

```
nacm rule-list any-group rule data-base access-operations delete action deny
```

DB Replica Sets Reflects Bad State

Issue: DB replica sets reflects bad state when UCS blades are rebooted.

If blades are rebooted, then follow the below procedure for DB VNF.

- Reboot the blade one by one. After rebooting of one blade, perform the following checks before proceeding to next blade.
  - Start all VMs of the blade
  - Make sure health checks and database status is clean

Solution:

Prerequisties for reconfiguration of a database:

- Take a backup of existing database config:

```
show running-config database | save /data/config/databaseConfigBackup03-02-23.txt
```

- Reconfigure the Databases. For more information, refer the Secnario: 1 All Sites are running with same mongo version, on page 70 and Sites running with same mongo version and prior mongo version, on page 73 topics in the *CPS vDRA Troubleshooting Guide*.

Secnario: 1 All Sites are running with same mongo version

Issue: All Sites are running with same mongo version, for example mongo 4.2.20.

Following procedural steps are applicable only for CPS 22.1.1 and later releases.

To reconfigure the databases, perform the following steps in the following sections:

Note The following steps are required for single cluster and multiple clusters. In case of mated pair deployments, the steps must be performed on all the sites.

Steps to be executed on DRA VNF

1. Log in to vDRA VNF CLI and run `no binding shard-metadata-db-connection` command to remove shard metadata-db-connection configurations.

   Example:

```
admin@orchestrator[an-master](config)# no binding shard-metadata-db-connection
   admin@orchestrator[an-master](config)# commit
   Commit complete.
   admin@orchestrator[an-master](config)# end
   admin@orchestrator[an-master]# show running-config binding
   % No entries found.
   admin@orchestrator[an-master]#
```

2. Log in to vDRA VNF CLI and run `db-authentication remove-password database mongo` command to remove MongoDB password.

   Example:

```
admin@orchestrator[an-master]# db-authentication remove-password database mongo
   Value for 'current-password' (<string>): ********
   admin@orchestrator[an-master]#
   admin@orchestrator[an-master]# db-authentication show-password database mongo
   result Mongo password is not configured.
   admin@orchestrator[an-master]#
```

Steps to be executed on DRA Binding VNF

1. Edit the nacm rule (config mode) to allow deleting the database configurations.

```
nacm rule-list any-group rule data-base access-operations delete action permit
```

2. Delete database configurations (config mode).

```
no database cluster <cluster name>
```

   where, <*cluster name*> is the name of the database cluster which has issues or need to be reconfigured.

Note As there are multiple database's clusters, you need to remove all the clusters from configuration.

3. Stop the system and wait for all the application containers to be stopped.

   ```
   system stop
   ```

4. Verify that the application containers are stopped by running `show scheduling status | include application` command. The command should not show any containers with issues.

5. (Only for MongoDB Authentication enabled database) Disable MongoDB authentication.

   ```
   db-authentication remove-password database mongo current-password XXX
   ```

6. To delete the persistent storage and old log information, run the following command on all VMs:

   ```
   rm -rf /data/mongod-node/db/*
   rm -rf /data/mongod-node/supervisord/supervisord-mongo.conf
   ```

   Example:

   ```
   $ for dbHost in `weave status connections | grep ss-persistence-db | awk '{print $2}'
    | cut -d':' -f1`;
   do ssh -o StrictHostKeyChecking=no -i cps.pem $dbHost 'sudo /bin/rm -fr
   /data/mongod-node/supervisord/supervisord-mongo.conf'; done
   ```

   ```
   $ for dbHost in `weave status connections | grep ss-persistence-db | awk '{print $2}'
    | cut -d':' -f1`;
   do ssh -o StrictHostKeyChecking=no -i cps.pem $dbHost 'sudo /bin/rm -fr
   /data/mongod-node/db/*'; done
   ```

Note Before proceeding to next step, make sure Step 1, on page 70 to Step 6, on page 71 has been executed from all the affected/reconfigured binding VNFs sites.

7. (Only when arbiter is on other site) Clear data directory for all STARTUP2 replica sets on arbiter VM.

   a. Find container where particular arbiter member is running. On arbiter site, run the following command to find arbiter hostname.

      ```
      show network ips | tab | include <arbiter-ipaddress>
      ```

   b. Find MongoDB container name for this host.

      ```
      show docker service | tab | include <host-name> | include mongo-s
      ```

   c. Connect to MongoDB container by running `docker connect mongo-s<id>` command.

   d. Clean the data directory. Stop all supervisor processes on arbiter container.

      ```
      supervisorctl stop all
      ```

   e. Clean database directoy on arbiter container.

      ```
      rm -rf /data/db/wt-*/*
      ```

   f. Start all supervisor processes on arbiter container.

      ```
      supervisorctl start all
      ```

8. Start the system and wait for system percentage to turn 100%.

`system start`

`show system status` should show system percentage as 100.

9. Merge existing config with the following command:

   ```
   load merge /data/config/databaseConfigBackup03-02-23.txt
   ```

10. Apply the database configurations again (config mode).

    Before proceeding to next step, make sure Step 8, on page 71 to 9, on page 72 has been executed from all the affected/reconfigured binding VNFs sites.

11. Verify all the database sets are UP by running `show database status | exclude "PRIMARY|SECONDARY|ARBITER"` command.

Note `show database status | exclude "PRIMARY|SECONDARY|ARBITER"` command should not show any unhealthy database member.

Before proceeding to next step, make sure Step 9, on page 72 to Step 10, on page 72 has been executed from all the affected/reconfigured binding VNFs sites.

12. Set `fcvcheck` as mentioned in the Database Generic fcvcheck Scenario section.

13. (Only for MongoDB Authentication enabled database) Enable MongoDB authentication.

    For more information on enabling MongoDB authentication, see *Configuring MongoDB Authentication* section in the *CPS vDRA Configuration Guide*.

Note Before proceeding to next step, make sure this step has been executed from all the affected/reconfigured binding VNFs sites.

Example 1: Enable MongoDB authentication with transition.

```
db-authentication set-password database mongo password XXX confirm-password XXX
db-authentication enable-transition-auth database mongo
db-authentication rolling-restart database mongo
db-authentication rolling-restart-status database mongo
Verify database member in healthy state
```

Example 2: Enable MongoDB authentication without transition.

```
db-authentication disable-transition-auth database mongo
db-authentication rolling-restart database mongo
db-authentication rolling-restart-status database mongo
```

14. Wait for all the database sets to come UP. Verify the status by running `show database status | exclude "PRIMARY|SECONDARY|ARBITER"` command.

Note `show database status | exclude "PRIMARY|SECONDARY|ARBITER"` command should not show any unhealthy database member.

15. Login to vDRA VNF CLI and restart the binding containers to read the latest metadata from new configurations.



Note Before proceeding to next step, make sure this step has been executed from all the affected/reconfigured DRA VNFs.

```
docker restart container-id <container-id>
```

Example:

```
docker restart container-id binding-s<number>
```

`binding-s`<*number*> is an example for binding container. binding-s<*number*> is the container-id. <*number*> varies according to which worker VM binding container is running.

16. Disable deleting the database configurations by editing the nacm rule (config mode).



Note Make sure this step has been executed from all the affected/reconfigured binding VNFs sites.

```
nacm rule-list any-group rule data-base access-operations delete action deny
```

Sites running with same mongo version and prior mongo version

Scenario 2: SITE A, C, and D are running with same n mongo version (for example, 4.4.18) but SITE B is running with n-1 mongo version (for example 4.2.20).

The following steps are applicable only for CPS 22.1.1 and later releases.

To reconfigure the databases, perform the following steps in the following sections:



Note The following steps are required for single cluster and multiple clusters. In case of mated pair deployments, the steps must be performed on all the sites.

Steps to be executed on DRA VNF

1. Login to vDRA VNF CLI and run `no binding shard-metadata-db-connection` command to remove shard metadata-db-connection configurations.

   Example:

   ```
   admin@orchestrator[an-master](config)# no binding shard-metadata-db-connection
      admin@orchestrator[an-master](config)# commit
      Commit complete.
      admin@orchestrator[an-master](config)# end
      admin@orchestrator[an-master]# show running-config binding
      % No entries found.
      admin@orchestrator[an-master]#
   ```

2. Login to vDRA VNF CLI and run `db-authentication remove-password database mongo` command to remove MongoDB password.

   Example:

```
admin@orchestrator[an-master]# db-authentication remove-password database mongo
   Value for 'current-password' (<string>): ********
   admin@orchestrator[an-master]#
   admin@orchestrator[an-master]# db-authentication show-password database mongo
   result Mongo password is not configured.
   admin@orchestrator[an-master]#
```

Steps to be executed on DRA Binding VNF

1. Edit the nacm rule (config mode) to allow deleting the database configurations.

   ```
   nacm rule-list any-group rule data-base access-operations delete action permit
   ```

2. Delete database configurations (config mode).

   ```
   no database cluster <cluster name>
   ```

   where, <*cluster name*> is the name of the database cluster which has issues or need to be reconfigured.

Note As there are multiple database's clusters, you need to remove all the clusters from configuration.

3. Stop the system and wait for all the application containers to be stopped.

   ```
   system stop
   ```

4. Verify that the application containers are stopped by running `show scheduling status | include application` command. The command should not show any containers with issues.

5. (Only for MongoDB Authentication enabled database) Disable MongoDB authentication.

   ```
   db-authentication remove-password database mongo current-password XXX
   ```

6. To delete the persistent storage and old log information, run the following command on all VMs:

   ```
   rm -rf /data/mongod-node/db/*
   rm -rf /data/mongod-node/supervisord/supervisord-mongo.conf
   ```

   Example:

   ```
   $ for dbHost in `weave status connections | grep ss-persistence-db | awk '{print $2}'
    | cut -d':' -f1`;
   do ssh -o StrictHostKeyChecking=no -i cps.pem $dbHost 'sudo /bin/rm -fr
   /data/mongod-node/supervisord/supervisord-mongo.conf'; done
   ```

   ```
   $ for dbHost in `weave status connections | grep ss-persistence-db | awk '{print $2}'
    | cut -d':' -f1`;
   do ssh -o StrictHostKeyChecking=no -i cps.pem $dbHost 'sudo /bin/rm -fr
   /data/mongod-node/db/*'; done
   ```

Note Before proceeding to next step, make sure Step 1, on page 74 to Step 6, on page 74 has been executed from all the affected/reconfigured binding VNFs sites.

7. (Only when arbiter is on other site) Clear data directory for all STARTUP2 replica sets on arbiter VM.

   a. Find container where particular arbiter member is running. On arbiter site, run the following command to find arbiter hostname.

      ```
      show network ips | tab | include <arbiter-ipaddress>
      ```

b. Find MongoDB container name for this host.

```
show docker service | tab | include <host-name> | include mongo-s
```

c. Connect to MongoDB container by running `docker connect mongo-s<id>` command.

d. Clean the data directory. Stop all supervisor processes on arbiter container.

```
supervisorctl stop all
```

e. Clean database directoy on arbiter container. Gets updated as previous version.

```
rm -rf /data/db/wt-*/*
```

f. Start all supervisor processes on arbiter container.

```
supervisorctl start all
```

8. Start the system and wait for system percentage to turn 100%.

```
system start
```

`show system status` should show system percentage as 100.

9. Merge existing config with the following command:

```
load merge /data/config/databaseConfigBackup03-02-23.txt
```

Issue: If database `genericfcvcheck` cli is stuck like given below:

```
admin@orchestrator[fPAS-site2-master-1]# database genericfcvcheck 4.2
Press "set" to check and set fcv on primary members or
Press "check" to only check fcv on all members.
(set/check) << set
This is going to check and set required FCV on orchestrator,mongo-admin-db,shards and
 shardingDb databases.
Press yes to continue
(yes/no) << yes
Please do not kill the terminal untill FCV check completes.Kindly check
logs(Path:/var/log/broadhop/fcv.log) for more i nfo
FCV check for shard databases is in Progress......
FCV check for shard databases is Completed
FCV check for shardingDb databases is in Progress......
```

Workaround:

a. Identify primary shards where fcv needs to be set manually:

Do not kill the database `genericfcvcheck` CLI. Continue running the database `genericfcvcheck` CLI and check the `logs /var/log/broadhop/fcv.log` inside orchestrator container and identify for which replica set Primary fcvcheck update is required:

```
cps@fPAS-site2-master-1:~$ docker exec -it orchestrator bash -c "tail -f
/var/log/broadhop/fcv.log"

2023-02-22 07:15:20 INFO SHARD DATABASE FCV CHECK STARTED
2023-02-22 07:15:24 INFO Found incompatible featureCompatibilityVersion 4.4 on
app_shardCD-ipv6-1
2023-02-22 07:15:24 INFO Found incompatible featureCompatibilityVersion 4.4 on
app_shardCD-ipv6-3
2023-02-22 07:15:24 INFO Found incompatible featureCompatibilityVersion 4.4 on
app_shardCD-ipv6-2
2023-02-22 07:15:24 INFO Found incompatible featureCompatibilityVersion 4.4 on
app_shardCD-ipv6-4
2023-02-22 07:15:25 INFO Found incompatible featureCompatibilityVersion 4.4 on
app_shardCD-ipv6-7
```

```
2023-02-22 07:15:25 INFO Found incompatible featureCompatibilityVersion 4.4 on
app_shardCD-ipv6-6
2023-02-22 07:15:25 INFO Found incompatible featureCompatibilityVersion 4.4 on
app_shardCD-ipv6-8
2023-02-22 07:15:25 INFO Found incompatible featureCompatibilityVersion 4.4 on
app_shardCD-ipv6-5
2023-02-22 07:15:34 INFO Require update featureCompatibilityVersion for
app_shardCD-ipv6-1 in app-based-session-ipv6-CD cluster
2023-02-22 07:15:34 INFO Require update featureCompatibilityVersion for
app_shardCD-ipv6-3 in app-based-session-ipv6-CD cluster
2023-02-22 07:15:34 INFO Require update featureCompatibilityVersion for
app_shardCD-ipv6-2 in app-based-session-ipv6-CD cluster
2023-02-22 07:15:34 INFO Require update featureCompatibilityVersion for
app_shardCD-ipv6-4 in app-based-session-ipv6-CD cluster
2023-02-22 07:15:34 INFO Require update featureCompatibilityVersion for
app_shardCD-ipv6-7 in app-based-session-ipv6-CD cluster
2023-02-22 07:15:34 INFO Require update featureCompatibilityVersion for
app_shardCD-ipv6-6 in app-based-session-ipv6-CD cluster
2023-02-22 07:15:34 INFO Require update featureCompatibilityVersion for
app_shardCD-ipv6-8 in app-based-session-ipv6-CD cluster
2023-02-22 07:15:34 INFO Require update featureCompatibilityVersion for
app_shardCD-ipv6-5 in app-based-session-ipv6-CD cluster

As per above logs we need to set fcv on
app_shardCD-ipv6-1,app_shardCD-ipv6-2,app_shardCD-ipv6-3,app_shardCD-ipv6-4,app_shardCD-ipv6-5,
app_shardCD-ipv6-6,app_shardCD-ipv6-7,app_shardCD-ipv6-8
```

b. Get the primary ip and port of the shard:

```
admin@orchestrator[fPAS-site1-master-1]# show database status | tab | include
app_shardCD-ipv6-1 | include PRI
2606:ae00:3001:8311:172:16:244:a8 27021 sess-ipv6-DB-C1 PRIMARY replica_set
app-based-session-ipv6-CD app_shardCD-ipv6-1 rs-app_shardCD-ipv6-1
```

c. Remove unreachable nodes from config:

1. ssh from master to db VM:

   ```
   cps@fPAS-site1-master-1:~$ ssh -i cps.pem 2606:ae00:3001:8311:172:16:244:a8
   ```

2. Log in to the respective mongo container:

   ```
   cps@fPAS-site3-app-persistence-db-3:~$ docker exec -it mongo-s104 bash
   ```

3. Log in to the respective port:

   ```
   root@mongo-s104:/# mongo mongodb://adminuser:cisco123@localhost:27021/admin
   ```

4. Check rs.status() and remove unreachable members:

   ```
   1) rs-app_shardCD-ipv6-1:PRIMARY> rs.status()
   {
   "set" : "rs-app_shardCD-ipv6-1",
   ...
   },
   "members" : [
   {
   "_id" : 0,
   "name" : "[2606:ae00:3001:8311:172:16:244:a8]:27021",
   "health" : 1,
   "state" : 1,
   "stateStr" : "PRIMARY",
   "uptime" : 407
   },
   {
   "_id" : 1,
   ```

```
"name" : "[2606:ae00:3001:8311:172:16:244:57]:27021",
"health" : 1,
"state" : 7,
"stateStr" : "ARBITER",
"uptime" : 385
},
{
"_id" : 2,
"name" : "[2606:ae00:3001:8311:172:16:244:11]:27021",
"health" : 0,
"state" : 8,
"stateStr" : "(not reachable/healthy)",
"uptime" : 0,
"lastHeartbeat" : ISODate("2023-02-22T07:16:50.589Z"),
"lastHeartbeatRecv" : ISODate("1970-01-01T00:00:00Z"),
"pingMs" : NumberLong(0),
"lastHeartbeatMessage" : "remote host has incompatible wire version: Server min
 and max wire version (0,8) is incompatible with client min wire version (9,9).You
  (client) are attempting to connect to a node (server) with a binary version
with which you (client) no longer accept connections. Please upgrade the server's
 binary version.",
"syncSourceHost" : "",
"syncSourceId" : -1,
"infoMessage" : "",
"configVersion" : -1,
"configTerm" : -1
},
{
"_id" : 3,
"name" : "[2606:ae00:3001:8311:172:16:244:a9]:27021",
"health" : 1,
"state" : 2,
"stateStr" : "SECONDARY",
"uptime" : 381
},
{
"_id" : 4,
"name" : "[2606:ae00:3001:8311:172:16:244:9]:27021",
"health" : 0,
"state" : 8,
"stateStr" : "(not reachable/healthy)",
"uptime" : 0,
"lastHeartbeat" : ISODate("2023-02-22T07:16:50.588Z"),
"lastHeartbeatRecv" : ISODate("1970-01-01T00:00:00Z"),
"pingMs" : NumberLong(0),
"lastHeartbeatMessage" : "remote host has incompatible wire version: Server min
 and max wire version (0,8) is incompatible with client min wire version (9,9).You
  (client) are attempting to connect to a node (server) with a binary version
with which you (client) no longer accept connections. Please upgrade the server's
 binary version.",
"syncSourceHost" : "",
"syncSourceId" : -1,
"infoMessage" : "",
"configVersion" : -1,
"configTerm" : -1
},
{
"_id" : 5,
"name" : "[2606:ae00:3001:8311:172:16:244:b8]:27021",
"health" : 1,
"state" : 2,
"stateStr" : "SECONDARY",
"uptime" : 377,
},
```

```
{
"_id" : 6,
"name" : "[2606:ae00:3001:8311:172:16:244:b9]:27021",
"health" : 1,
"state" : 2,
"stateStr" : "SECONDARY",
"uptime" : 371,
"optime" : {
"ts" : Timestamp(1677050206, 1),
"t" : NumberLong(1)
}
],
"ok" : 1,
"$clusterTime" : {
}}

"[2606:ae00:3001:8311:172:16:244:11]:27021" and
"[2606:ae00:3001:8311:172:16:244:9]:27021" members are not able to connect.

2) Remove the members with below command

rs-app_shardCD-ipv6-1:PRIMARY>
rs.remove("[2606:ae00:3001:8311:172:16:244:11]:27021")
{
"ok" : 1,
"$clusterTime" : {
"clusterTime" : Timestamp(1677050239, 1),
"signature" : {
"hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAA="),
"keyId" : NumberLong(0)
}
},
"operationTime" : Timestamp(1677050239, 1)
}
rs-app_shardCD-ipv6-1:PRIMARY>
rs.remove("[2606:ae00:3001:8311:172:16:244:9]:27021")
{
"ok" : 1,
"$clusterTime" : {
"clusterTime" : Timestamp(1677050245, 1),
"signature" : {
"hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAA="),
"keyId" : NumberLong(0)
}
},
"operationTime" : Timestamp(1677050245, 1)
}
```

After all unreachable members are removed from *rs*, `fcv` is automatically by backend CLI. The following details appear in `fcv` logs for respective shards.

```
cps@fPAS-site2-master-1:~$ docker exec -it orchestrator bash -c "tail -f
/var/log/broadhop/fcv.log"

2023-02-22 05:35:15 INFO Successfully set featureCompatibilityVersion for
app_shardCD-ipv6-1
in app-based-session-ipv6-CD cluster
```

Repeat the step 1 and step 2 for remaining shards, then proceed for next step.

d. Rolling restart for site running on lower mongo version (n-1 mongo version(for example, 4.2.20))

```
admin@orchestrator[fPAS-site2-master-1]# db-authentication rolling-restart-parallel
 database mongo
```

```
wait for some time to complete the rolling restart and check the status with below
 command

admin@orchestrator[fPAS-site1-master-1]# db-authentication
rolling-restart-parallel-status database mongo
result Parallel rolling restart completed!! Batch 5 out of total 5 got completed
at 2023-02-23T06:52:33.449

execute below command and if output is empty then set fcv to remaining sites.

admin@orchestrator[fPAS-site1-master-1]# show database status | nomore | exclude
"PRIMA|SECON|ARBI" | tab
ADDRESS PORT NAME STATUS TYPE CLUSTER NAME SHARD REPLICA SET
```

10. Apply the database configurations again (config mode).

    Before proceeding to next step, make sure Step 8, on page 75 to Step 9, on page 75 has been executed from all the affected/reconfigured binding VNFs sites.

11. Verify all the database sets are UP by running `show database status | exclude "PRIMARY|SECONDARY|ARBITER"` command.

Note `show database status | exclude "PRIMARY|SECONDARY|ARBITER"` command should not show any unhealthy database member.

Before proceeding to next step, make sure Step 9, on page 75 to Step 10, on page 79 has been executed from all the affected/reconfigured binding VNFs sites.

12. Set `fcvcheck` as mentioned in the Database Generic fcvcheck Scenario section.

13. (Only for MongoDB Authentication enabled database) Enable MongoDB authentication.

    For more information on enabling MongoDB authentication, see *Configuring MongoDB Authentication* section in the *CPS vDRA Configuration Guide*.

Note Before proceeding to next step, make sure this step has been executed from all the affected/reconfigured binding VNFs sites.

Example 1: Enable MongoDB authentication with transition.

```
db-authentication set-password database mongo password XXX confirm-password XXX
db-authentication enable-transition-auth database mongo
db-authentication rolling-restart database mongo
db-authentication rolling-restart-status database mongo
Verify database member in healthy state
```

Example 2: Enable MongoDB authentication without transition.

```
db-authentication disable-transition-auth database mongo
db-authentication rolling-restart database mongo
db-authentication rolling-restart-status database mongo
```

14. Wait for all the database sets to come UP. Verify the status by running `show database status | exclude "PRIMARY|SECONDARY|ARBITER"` command.

Note `show database status | exclude "PRIMARY|SECONDARY|ARBITER"` command should not show any unhealthy database member.

15. Login to vDRA VNF CLI and restart the binding containers to read the latest metadata from new configurations.

Note Before proceeding to next step, make sure that this step has been executed from all the affected/reconfigured DRA VNFs.

```
docker restart container-id <container-id>
```

Example:

```
docker restart container-id binding-s<number>
```

`binding-s<number>` is an example for binding container. binding-s<*number*> is the container-id. <*number*> varies according to which worker VM binding container is running.

16. Disable deleting the database configurations by editing the nacm rule (config mode).

Note Make sure this step has been executed from all the affected/reconfigured binding VNFs sites.

```
nacm rule-list any-group rule data-base access-operations delete action deny
```

MongoDB Shard Recovery Procedure

Issue: In shard replica-set, if all the data bearing members are down at same time, they fail to elect new Primary and gets stuck in STARTUP2 state.

Run `show database status | tab` command on Binding VNF to check for shards without Primary member.

Precautions

- Each step provided in the **Solution** section needs to be executed on all the sites where shard members are involved before proceeding with the next step.
- All steps must be executed only on Binding VNF.
- If there are more than one shard in failure state, all steps need to be executed for each shard separately.

Solution

Run the following steps on all the sites involved for that shard.

1. Stop all the mongod processes involved for that shard for all the secondary and primary sites.

```
supervisorctl stop <process which stuck in STARTUP2>
```

Note Find MongoDB container where shard member is running using `supervisorctl status` command. Output grep for the port number which displays the process to be stopped.

2. Remove mmap folder for respective port.

   For example, if port number is 27030,

   ```
   rm -rf /mmapv1-tmpfs-27030/*
   ```

3. Start all the secondary member mongod processes without authentication followed by primary member.

   For example, if port number is 27030 for a particular member, then

   ```
   mongod --storageEngine mmapv1 --nojournal --noprealloc --smallfiles
   --ipv6 --bind_ip_all --port 27030 --dbpath=/mmapv1-tmpfs-27030 --replSet
   rs-shard-9 --quiet --slowms 500 --logpath /data/db/mongo-27030.log --oplogSize
   3221 --logappend --logRotate reopen &
   ```

Note Find MongoDB container where shard member is running using `supervisorctl status` command.

4. Verify that the replica-set is up with primary.

Note Find MongoDB container where shard member is running using `supervisorctl status` command. Run `mongo --port 27030` command which connects to MongoDB shell where status is displayed.

5. Set password on primary. Connect to primary member MongoDB shell and run the following commands to create user and set password.

   ```
   use admin
   db.createUser({user: "adminuser",pwd: "PASSWORD",roles:[{role: "root" , db:"admin"}]})
   db.createUser({user: "backupuser",pwd: "PASSWORD",roles:[{role: "root" , db:"admin"}]})
   ```

6. Restart the secondary mongod process using transition authentication followed by primary.

   a. Find MongoDB container where shard member is running using `supervisorctl status` command. Run the command to stop the process.

      ```
      ps -aef | grep <portnum>
      kill -SIGTERM <pid>
      > /mmapv1-tmpfs-27030/mongod.lock
      ```

   b. Run the following command to start the process with transition-auth option.

      ```
      mongod --storageEngine --transition-auth mmapv1 --nojournal
      --noprealloc --smallfiles --ipv6 --bind_ip_all --port 27030
      --dbpath=/mmapv1-tmpfs-27030 --replSet rs-shard-9 --quiet --slowms 500
      --logpath /data/db/mongo-27030.log --oplogSize 3221 --logappend
      --logRotate reopen &
      ```

7. Restart secondary keyfile and disable transition-auth followed by primary.

   a. Find MongoDB container where shard member is running using `supervisorctl status` command. Run the command to stop the process.

```
ps -aef | grep <portnum>
kill -SIGTERM <pid>
> /mmapv1-tmpfs-27030/mongod.lock
```

b. Run the following command to start the process without transition-auth option and with keyfile.

```
supervisorctl status
```

From the above command, output grep for the port number, which displays the process to be stopped.

```
supervisorctl start <pid/process name>
```

8. Run `show database status | tab` to display the shard recovery with the database status.

Recovery Using database repair Command



Attention In HA deployment, CLI needs to be run on single site.

Logs (`/var/log/broadhop/shardrecovery.log`) should be checked after executing CLI.

- **Issue 1:** In shard replica-set, if all the data bearing members are down at same time, they fail to elect new Primary and gets stuck in STARTUP2 state.

  OR

  Primary is present but single or multiple secondary members are down.

  Solution: Use `database repair <clustername> <shardname>`

  For example, to recover shard1 in binding cluster, execute `database repair binding shard1` command.

- **Issue 2:** In multiple shards, if all the data bearing members are down at same time, they fail to elect new Primary and gets stuck in STARTUP2 state.

  OR

  Primary is present but single or multiple secondary members are down.

  Solution: Use `database repair <clustername> <shardname1> <shardname2> <shardname3>`

  For example, to recover shard1, shard2, shard3, and shard4 in binding cluster, execute `database repair binding shard1 shard2 shard3 shard4` command.

- **Issue 3:** If all shards in the cluster are in bad state.

  Solution: Use `database repair <clustername> All`

  For example, to recover all shards in the binding cluster, execute `database repair binding All` command.

- If sharding database members are in STARTUP2 state.

  Solution: Use `database repair <clustername> sharddb`

  For example, to recover sharding database in the binding cluster, execute `databse repair binding shard-db` command.