



vDRA

- [Application based Client Sharding, on page 1](#)
- [Binding Database Record Capacity Overload Handling, on page 2](#)
- [DRA Distributor, on page 4](#)
- [DRA Distributor Usability Improvements, on page 4](#)
- [In-Service Migration from MongoDB Sharding to Application Sharding, on page 6](#)
- [Linux Utilities for Troubleshooting, on page 7](#)
- [Load Balance Diameter Messages across vDRA Relay Connections, on page 8](#)
- [MongoDB Service without Authentication Detection, on page 9](#)
- [Redis Security Vulnerabilities on vDRA, on page 10](#)
- [Routing Method Order Enhancements, on page 11](#)
- [Support to Create Mongo Shard/Replica-set, on page 12](#)
- [Support to Create Prometheus Statistics Partition, on page 13](#)
- [Upgrade MongoDB Version 3.6, on page 14](#)
- [vDRA Resiliency Improvements and Database Overload Protection, on page 15](#)
- [Zone-Aware Sharding, on page 15](#)

Application based Client Sharding

Feature Summary and Revision History

Table 1: Summary Data

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	CPS vDRA Configuration Guide CPS vDRA Operations Guide

Table 2: Revision History

Revision Details	Release
First introduced:	19.4.0

Feature Description

vDRA now supports application-based client sharding to partition/distribute session/binding data across multiple database replica-sets. This feature is implemented to scale larger-scale deployments with large numbers of replica-sets.

vDRA also supports regional IPv6 bindings (per mated-pair) in addition to a national binding database storing IMSI and MSISDN binding records.

For more information, see the following:

- `binding db-read-connection-settings` section in the *CPS vDRA Operations Guide*
- `binding shard-metadata-db-connection` section in the *CPS vDRA Operations Guide*
- `database cluster db-name sharding-db name` section in the *CPS vDRA Operations Guide*
- `database cluster db-name sharding-db-seed name` section in the *CPS vDRA Operations Guide*
- `database cluster db-name multi-db-collections` section in the *CPS vDRA Operations Guide*
- *Configuring Application based Sharding* section in the *CPS vDRA Configuration Guide*.

Binding Database Record Capacity Overload Handling

Feature Summary and Revision History**Table 3: Summary Data**

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	CPS vDRA Configuration Guide CPS vDRA Operations Guide CPS vDRA SNMP and Alarms Guide

Table 4: Revision History

Revision Details	Release
First introduced	19.4.0

Feature Description

vDRA supports enforcing the maximum number of records in a binding database (active + stale) so that vDRA remains stable in the event an excessive number of bindings are required.

To make sure that the system remains stable and continues to work, there is a limit in the session and binding records in a database. If the database record limit is breached for session/bindings, vDRA rejects the call with the configurable error code. For more information about the error code configuration, see *Configure Error Result Code Profile* in the *CPS vDRA Configuration Guide*.

When session and database record limit is breached and the best effort is enabled on binding keys, vDRA forwards call to PCRF without storing best effort binding keys in database.

Using `binding db-max-record-limit` command, you can configure maximum record limit on session and bindings.

If the database is mated pair, maximum record limit values must be same on both the sites.

If the maximum record limit is not configured or if configured to zero, there will be no database record limit on that binding type. By default, there is no database record limit configured on any of the binding types.

The following new alarms have been added:

- SESSION_DB_LIMIT_EXCEEDED
- IPV6_DB_LIMIT_EXCEEDED
- IPV4_DB_LIMIT_EXCEEDED
- IMSIAPN_DB_LIMIT_EXCEEDED
- MSISDNAPN_DB_LIMIT_EXCEEDED

For more information, see the following:

- *binding db-max-record-limit* section in the *CPS vDRA Operations Guide*
- *Application Notifications* table in the *CPS vDRA SNMP and Alarms Guide*
- *Configuring Binding Database Overload* section in the *CPS vDRA Configuration Guide*

Configuration and Restrictions

If vDRA is configured to send error response, in case of database overload error response might not be sent back to PCEF for all the rejected calls.

As the record count thread runs every 10 sec, vDRA does not always have a real time record count. When applying database limit over bindings, database limit breached condition takes effect with few more records in the database than the specified maximum limit.

DRA Distributor

Feature Summary and Revision History

Table 5: Summary Data

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	Not Applicable

Table 6: Revision History

Revision Details	Release
First introduced	19.4.0

Feature Description

CPS is enhanced to support a DRA Distributor which is a traffic distributor that transparently does the following:

- Presents a virtual IP address representing a set of up to 16 application servers to application clients. This reduces the number of addresses with which the clients need to be configured.
- Routes packets from the clients that establish new TCP connections to selected application servers.
- Routes packets in existing TCP connections to the correct servers for the connection.

DRA Distributor Usability Improvements

Feature Summary and Revision History

Table 7: Summary Data

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	CPS vDRA Operations Guide

Table 8: Revision History

Revision Details	Release
First introduced:	19.4.0

Automate VIP and Netfilter Configurations

Distributor IPv4/IPv6 services require a VIP configuration in the loopback interface and Netfilter rules for each VIP in the Director VMs. Previously, you had to manually add the Distributor IPv4 and IPv6 VIPs and associated Netfilter rules to the Director VMs.

Now vDRA supports an automated mechanism to configure the Director's VIPs, ARP tables, and IPv6 tables rules. Scripts are added in two new containers on the director VMs. These containers are included in the `diameter-endpoint` module.

- Real-server-monitor
- Real-server containers

Display Output of Distributor `ipvsadm` Command

New CLI command "`show dra-distributor`" has been added to the Orchestrator container to display the output of `ipvsadm` from all Distributor VMs.

For more information, see `show dra-distributor` section in the *CPS vDRA Operations Guide*.

Health Check Single Diameter Peers TCP Port

Diameter endpoint IP address and TCP port number are monitored to determine the health of the diameter endpoint at each Director VM supporting a Distributor Service. If the diameter IP address/port status fails, the Director VM is removed from the Distributor Service until the health check passes.

Improvement in Grafana Dashboard

Two new panels, vDRA distributor statistics, and vDRA distributor VIPs are added to Grafana.

vDRA distributor statistics displays the following for each Distributor VM.

- Incoming Packet and bytes per second
- CPU Utilization
- Memory Utilization
- Connection Info
- Real Server Weights

vDRA distributor VIPs display the following information for each Distributor VM:

- Active connections per VIP
- Inactive connections per VIP

Configuration Parameter to Enable/Disable Preemption

DRA Distributor's Virtual IPs (VIPs) are managed by keepalived. Keepalived implements the Virtual Router Redundancy Protocol. VIPs have a priority of 1 – 254. The VIP with the highest priority becomes the Master. VRRP normally preempts a lower priority DRA Distributor VM when a higher priority DRA Distributor becomes available. In some circumstances it is desirable to not preempt lower priority DRA Distributor VM.

The preempt parameter, if set to false, disables preemption. By default, the preempt parameter is set to true.

Here is an example configuration:

```
network dra-distributor client
sync-id 110
sync-interface ens192
tracking-service diameter-endpoint
preempt true
preempt-delay 30
connection-timeout tcp 30
connection-timeout tcpfin 30
host 192.169.21.20
    priority 10
!
host 192.169.21.21
    priority 5
!
service Gx4
    virtual-router-id 121
    interface ens224
    service-ip 192.169.22.50
    service-port 3868
    real-server 192.169.22.13
        weight 1
    !
    real-server 192.169.22.14
        weight 1
    !
!
```

In-Service Migration from MongoDB Sharding to Application Sharding

vDRA now supports an in-service migration from vDRA running MongoDB sharding to a vDRA running Application Sharding so that there is no VoLTE calls drops and/or trigger an FCC or CRTC reportable outage.

**Restriction**

- During migration of a mated-pair, the GR resiliency is not fully available until both the sites are migrated. For example, if Site1 is migrated and all the new sessions for Site1 are getting stored in new application-sharded databases, then when Site1 goes down, the existing sessions from Site1 are not accessible from Site2 (as Site2 is still using only mongo-sharded cluster).
- vDRA does not support mixed databases with different shard-type. A DRA VNF site should not be configured where session database has mongo-based sharding and binding database has other type of sharding. The connections on all the databases (session and bindings) should be configured at the same time.
- When DRA VNF is configured with two database connections on mongo-sharded as well as application-sharded databases, then the session/binding expiration is performed on the records in the Primary DB (default primary application-sharded DB). MongoDB sharded database can be made as primary using `dra migration enable-mongo-sharded-db-as-primary-db` command

For more information, see *In-Service Migration from MongoDB Sharding to Application Sharding* chapter in the *CPS vDRA Installation Guide for VMware*.

Linux Utilities for Troubleshooting

Feature Summary and Revision History

Table 9: Summary Data

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	Not Applicable

Table 10: Revision History

Revision Details	Release
First introduced:	19.4.0

Feature Description

The following linux utilities are added as a part of vDRA base_vmdk:

- traceroute
- pstree
- dig

- nslookup
- netcat
- htop
- ipset
- ndisc6
- rdisc6
- sosreport
- sysstat utilities (iostat, mpstat, pidstat and so on)
- tcpdump (tcpdump is also available in docker-host-info container other than VM. But if you need packet captures after running "system stop", tcpdump is no longer available from the container. The tcpdump utility available in VM can be used.)

Load Balance Diameter Messages across vDRA Relay Connections

Feature Summary and Revision History

Table 11: Summary Data

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	Not Applicable

Table 12: Revision History

Revision Details	Release
First introduced:	19.4.0

Feature Description

In this release, vDRA supports load balance inter-PAS diameter requests across relay connections so that the higher TPS requests is achieved.

The following new statistics has been added:

- relay_message_total: This statistics shows the total number of messages relayed on relay channel.

Field in statistics:

- endpoint: endpoint
- relay_peer: relay peer
- app_id: Application ID
- direction: ingress/egress
- message_type: request/answer

MongoDB Service without Authentication Detection

Feature Summary and Revision History

Table 13: Summary Data

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	CPS vDRA Configuration Guide CPS vDRA Operations Guide

Table 14: Revision History

Revision Details	Release
First introduced:	19.4.0

Feature Description

vDRA now supports password based authentication for MongoDB. Encrypted password is stored in consul database in key value format.



Note Role base authentication is not supported.

For more information, see the following:

- `db-authentication set-password database mongo password` section in the *CPS vDRA Operations Guide*
- `db-authentication enable-transition-auth database mongo` section in the *CPS vDRA Operations Guide*

- `db-authentication disable-transition-auth database mongo` section in the *CPS vDRA Operations Guide*
- `db-authentication rolling-restart database mongo` section in the *CPS vDRA Operations Guide*
- `db-authentication rolling-restart-status database mongo` section in the *CPS vDRA Operations Guide*
- `db-authentication remove-password database mongo` section in the *CPS vDRA Operations Guide*
- `db-authentication change-password database mongo` section in the *CPS vDRA Operations Guide*
- `db-authentication sync-password database mongo` section in the *CPS vDRA Operations Guide*
- *Configuring MongoDB Authentication* section in the *CPS vDRA Configuration Guide*

Redis Security Vulnerabilities on vDRA

Feature Summary and Revision History

Table 15: Summary Data

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Enabled - Configuration Required
Related Changes in This Release	Not Applicable
Related Documentation	CPS vDRA Operations Guide

Table 16: Revision History

Revision Details	Release
First introduced:	19.4.0

Feature Description

In this release, Redis is upgraded to 4.0.10-alpine version which supports password based authentication. The password can be configured through the confD CLI.

After successful upgrade to this release (all DRA's in network), enable Redis authentication. It is mandatory that Redis authentication be enabled only after successful upgrade on all sites in the network.

All the sites in the network must have the same password to connect with local/global control planes/Redis servers.



Important

Password provisioning/changes must be done across all the sites in the network in one maintenance window.

For more information on Redis authentication using confD CLI, see the following sections:

- `db-authentication set-password database redis password` section in the *CPS vDRA Operations Guide*
- `db-authentication show-password database redis` section in the *CPS vDRA Operations Guide*
- `db-authentication remove-password database redis` section in the *CPS vDRA Operations Guide*

Routing Method Order Enhancements

Feature Summary and Revision History

Table 17: Summary Data

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	Not Applicable

Table 18: Revision History

Revision Details	Release
First introduced:	19.4.0

Feature Description

In this release, routing order has been changed as follows:

- Dest-Host Routing (local topology)
- Dest-Host Routing (global topology)
- SRK routing (local topology)
- SRK routing (global topology)
- Table-Driven routing (use local/global topology to find the peer)

Support to Create Mongo Shard/Replica-set

Feature Summary and Revision History

Table 19: Summary Data

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	Not Applicable

Table 20: Revision History

Revision Details	Release
First introduced:	19.4.0

Feature Description

vDRA now supports initializing of replica sets with minimum members present in NOT_INITIALIZED state. As the members from other site comes up, they automatically get added to replica set with appropriate state as Secondary or arbiter.

You need to configure database cluster configurations in each site only one time. Replica set gets initialized which results in available members with Primary, Secondary and Arbiter. Even though other members, which are not available from other sites, can still be in NO_CONNECTION state.

```
admin@orchestrator[pn-bind-master-0]# show database status | tab
```

ADDRESS	PORT	CLUSTER		TYPE	NAME	SHARD	REPLICA SET
		NAME	STATUS				
192.169.67.28	27017	server-a	PRIMARY	replica_set	binding	shard-1	rs-shard-1
192.169.67.31	27017	server-b	SECONDARY	replica_set	binding	shard-1	rs-shard-1
192.169.67.32	27017	server-c	ARBITER	replica_set	binding	shard-1	rs-shard-1
192.169.67.14	27017	server-d	NO_CONNECTION	replica_set	binding	shard-1	rs-shard-1
192.169.67.15	27017	server-d	NO_CONNECTION	replica_set	binding	shard-1	rs-shard-1



Note Make sure minimum two members are added in replica set while creating database cluster.

Support to Create Prometheus Statistics Partition

Feature Summary and Revision History

Table 21: Summary Data

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	Not Applicable

Table 22: Revision History

Revision Details	Release
First introduced:	19.4.0

Feature Description

During installation, the CPS deployer is used to create a data disk for the master, and two control VMs. Cloud-init partitions the data disk, creates a file system, and mounts the data disk. The storage layout for the Prometheus containers is hard coded in the orchestrator Prometheus module.

The new data disk layout is as follows:

- Partitions: The new data disk is 200GB and contains the following partitions.
 - /data (70 GB)
 - /stats (130 GB)
- Directory Structure

Description	Location
haproxy certificate	/data/certs/proxy.pem
Consul database	/data/consul-[123]/data
Confd network dns entries	/data/dnsmasq/etc/dnsmasq.conf
swarm.json	/data/install/swarm.json
ISOs	/data/isos/
Keepalived metadata	/data/keepalived/
Mongo admin-db	/data/mongo-admin-[abc]/

Description	Location
Mongod-node	/data/mongod-node/
Mongo-s10x	/data/mongo-s10x/
Orchestrator (confd, user config, orchestration db)	/data/orchestration/
Bulkstats	/stats/bulkstats/
Node Exporter Textfile Collector	/stats/node/
Prometheus Textfile Collector	/stats/prometheus/
Prometheus Hi-Res storage db	/stats/prometheus-hi-res/2.0/
Prometheus-hi-res-s10x metadata	/stats/prometheus-hi-res-s10x/data/
Prometheus Planning storage db	/stats/prometheus-planning/2.0/
Prometheus-planning-s10x metadata	/stats/prometheus-planning-s10x/data/
Prometheus Trending storage db	/stats/Prometheus-trending/2.0/
prometheus-trending-s10x metadata	/stats/Prometheus-trending-s10x/data/

Upgrade MongoDB Version 3.6

Feature Summary and Revision History

Table 23: Summary Data

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	Not Applicable

Table 24: Revision History

Revision Details	Release
First introduced:	19.4.0

Feature Description

vDRA supports MongoDB version 3.6.9. Currently, vDRA supports only fresh installation of MongoDB v3.6.9.

vDRA Resiliency Improvements and Database Overload Protection

Feature Summary and Revision History**Table 25: Summary Data**

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Disabled - Configuration Required
Related Changes in This Release	Not Applicable
Related Documentation	CPS vDRA Configuration Guide

Table 26: Revision History

Revision Details	Release
First introduced:	19.4.0

Feature Description

In this release, vDRA worker node supports processing of all AARs when there is a high load. There is no AAR (contains Framed-IPv6/IPv4) message drop during high load. This feature can be enabled or disabled using **A A R Priority Processing** check box through Policy Builder. vDRA worker node must have separate thread pool to execute session and binding deletions. This thread pool is configurable under **Threading Configuration** in Policy Builder.

For more information, see *DRA Feature* section in the *CPS vDRA Configuration Guide*.

Zone-Aware Sharding

Feature Summary and Revision History**Table 27: Summary Data**

Applicable Product(s) or Functional Area	vDRA
Applicable Platform(s)	Not Applicable

Default Setting	Enabled - Always-on
Related Changes in This Release	Not Applicable
Related Documentation	CPS vDRA Configuration Guide CPS vDRA Operations Guide

Table 28: Revision History

Revision Details	Release
First introduced:	19.4.0

Feature Description

vDRA now supports creating zones for IPv6 shards based on IPv6 pools. This is done so that the primary member of the replica set for an IPv6 address resides at the same physical location as the PGW assigning addresses from the IPv6 pool. This results in local writes (and reads) for the IPv6 binding database.

For more information, see the following:

- `database cluster db-name ipv6-zone-sharding` section in the *CPS vDRA Operations Guide*
- `database cluster db-name ipv6-zones-range` section in the *CPS vDRA Operations Guide*
- `database cluster db-name shard` section in the *CPS vDRA Operations Guide*
- *Configuring Zone Aware Sharding* section in the *CPS vDRA Configuration Guide*