vDRA

Generate Logs for specific Diameter Endpoint or binding container

Feature summary and revision history

Table 1: 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 2: Revision History

Revision Details	Release
In this release, enhancement is supported to enable logs for a specifc DRA application's container (max 3 containers per module) to reduce the log size and simplify troubleshooting.	25.2.0

Enabling application logs for specific containers

Enabling application logs for specific containers is a feature that:

- reduces log flooding and out-of-order messages,
- simplifies troubleshooting for DRA applications, and
- allows granular control over logging by targeting individual or multiple containers.

In Diameter Routing Agent (DRA), enabling application logs is crucial for troubleshooting, debugging, performance monitoring, and audit and accountability of the DRA application. Previously, enabling trace, error, debug, warn, or info logs would apply to all application containers, causing excessive logging, log misses, and out-of-order messages. This feature addresses these issues by allowing you to enable application logs only for specific containers, making troubleshooting more efficient.

Limitations

The limitations are:

- Even when the debug level is enabled for specific containers, WARN and ERROR application logs are always included in the consolidated QNS logs for all containers.
- If you specify more than three containers in the command-line interface (CLI), only the first three containers are considered for logging. A message will indicate that logs are enabled only for the first three containers. Similarly, the show logger level command will display only these first three containers, and application logs will be generated in the consolidated QNS logs for only these three.

Manage loggers during system upgrades

This task allows you to manage existing loggers during system upgrades to a newer version, for example, from 25.1 to 25.2.

Use these steps to perform pre-upgrade actions and recovery (if pre-upgrade clearing was missed).

Procedure

Step 1 Perform pre-upgrade actions using these steps:

a) Before initiating the upgrade, take a backup of all currently enabled or disabled loggers.
b) Remove all existing loggers from your system.
c) After the upgrade is complete, re-configure your desired loggers using the new CLI format, which supports specifying containers.

```
admin@orchestrator[site3-dra-master0]# logger set  logger-name
logger-level container-name,container-name2,containers-name3
```

Example:

logger set com.example.app debug container1,container2

Step 2 If you upgraded from version 25.1 or earlier without clearing the existing loggers, complete these steps after the upgrade is complete:

a) Execute the following command to list loggers set with the previous format:

```
consul kv get --recurse cisco-policy/logging/loggers/
```

b) Review the output from Step a and collect the names of loggers that do not include attributes such as "instance1, instance2, instance3, level".

Example:

if the output contains `cisco-policy/logging/loggers/com.broadhop.dra:warn`, then `com.broadhop.dra` is the logger name to target.

c) For each logger name identified in Step b, run the delete command:

```
consul kv delete cisco-policy/logging/loggers/<logger-name>
```

Example:

```
consul kv delete cisco-policy/logging/loggers/com.broadhop.dra
```

d) Repeat Step c for all loggers identified in Step b.

e) Run the **consul kv get** command again to confirm that no loggers from the old format remain.

```
consul kv get --recurse cisco-policy/logging/loggers/
```

Note
If you upgrade from a version later than 25.2, all existing loggers will be automatically available in the new version.

Managing loggers during downgrades

This task allows you to manage existing loggers during system downgrades, especially those configured with container-specific settings.

This task allows you to downgrade your system to a newer version, for example, from 25.2 to 25.1.

Before you begin

Use these steps to perform pre-downgrade actions and recovery (if pre-downgrade clearing was missed).

Procedure

Step 1 Perform pre-downgrade actions using these steps:

a) Before initiating the downgrade, take a backup of all currently enabled or disabled loggers.

b) Remove all existing loggers from your system.

c) Proceed with the system downgrade to the older version.

d) After the downgrade is complete, re-configure your desired loggers using the new CLI format, which supports specifying containers.

```
admin@orchestrator[site3-dra-master0]# logger set  logger-name logger-level
```

Example:

```
logger set com.example.app info
```

Step 2 If you downgraded from version 25.2 or earlier without clearing the existing loggers, complete these steps after the downgrade is complete:

a) Execute the following command to list loggers set with the newer format:

```
consul kv get --recurse cisco-policy/logging/loggers/
```

b) Review the output from Step a and collect the names of loggers that do not include attributes such as "instance1, instance2, instance3, level".

Example:

if the output contains `cisco-policy/logging/loggers/instance1/com.broadhop.dra:binding-s107`, then `instance1/com.broadhop.dra` is the logger name to target.

c) For each logger name identified in Step b, run the delete command:

```
consul kv delete cisco-policy/logging/loggers/<instanceX>/<logger-name>
```

Example:

```
consul kv delete cisco-policy/logging/loggers/instance2/com.broadhop.dra
```

d) Repeat Step c for all loggers identified in Step b.

e) Run the **consul kv get** command again to confirm that no loggers from the old format remain.

```
consul kv get --recurse cisco-policy/logging/loggers/
```

Note
If you downgrade from a version later than 25.2, all existing loggers will be automatically available in the new version.

MFA Support for Orchestrator CLI

Feature Summary and Revision History

Table 3: Summary Data

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

Table 4: Revision History

Revision Details	Release
First introduced.	25.2.0

Feature Description

This feature enhances the security of the vDRA orchestrator system by introducing Multi-Factor Authentication (MFA) for CLI access. With this update, users attempting to access the orchestrator CLI must provide two authentication factors: a PEM file (private key) and a password. This dual-factor requirement adds an extra layer of security, reducing the risk of unauthorized CLI access.

The commands introduced are:

- **mfa-cli enable user-id** <*USER_ID*> CLI command enables the MFA for a user. To disable the MFA support for a user, use the **no mfa-cli enable user-id** <*USER_ID*> CLI command.
- **show running-config mfa-cli enable** CLI command displays MFA-enabled users.
- **cli-mfa -i user.pem -p 2024 user@localhost** CLI command helps to access the confd CLI interface for MFA users.

Configure MFA Support for Orchestrator CLI

Prerequisite:

Before using the MFA CLI commands, ensure the SSSD configuration is updated. Update the `/etc/sssd/sssd.conf` file on DRM and DRC VMs. This configuration allows GTAC users to list their IDs.

Follow this procedure to configure the MFA support for orchestrator CLI.

1. Add the alias in artifacts before VMDK upgrade or fresh installation procedure.

   ```
   alias cli-mfa="/etc/ssh-mfa"
   ```

2. Manually create the alias in Deployer VM.

   ```
   alias cli-mfa="/etc/ssh-mfa"
   ```

3. Copy the script from deployer and create alias to install the package as given in the jump server.

   ```
   alias cli-mfa="/etc/ssh-mfa"
   sudo apt-get install sshpass # For Debian/Ubuntu
   ```

4. Add the given configuration in the DRM/DBM to allow only the admin to execute the MFA enable CLI access.

   ```
   admin@orchestrator[TEST-Binding-master](config)# nacm groups group mfa user-name admin
   admin@orchestrator[TEST-Binding-master](config)# commit
   Commit complete.
   admin@orchestrator[vpas-B1-master-0](config)# nacm rule-list allow-mfa-access group [
   mfa ] rule allow-access module-name tailf-cps-orchestrator path /mfa-cli/
   access-operations create,read,update,delete,exec action permit
   admin@orchestrator[vpas-B1-master-0](config-rule-allow-access)# commit
   Commit complete.
   admin@orchestrator[vpas-B1-master-0](config)# nacm rule-list restrict-mfa-access group
   [ * ] rule restrict-access module-name tailf-cps-orchestrator path /mfa-cli/
   access-operations create,read,update,delete,exec action deny
   admin@orchestrator[vpas-B1-master-0](config-rule-restrict-access)# commit
   Commit complete.
   ```

5. Verify `external-aaa pam gid-mapping` configurations. These mappings ensure MFA-enabled users can access applications like Grafana or CPS Central by linking user roles to Group IDs (GIDs).

   ```
   admin@orchestrator[M3-vpas-A-master-0]# show running-config external-aaa pam
   external-aaa pam gid-mapping 100 admin
   !
   external-aaa pam gid-mapping 100 grafana-viewer
   ```

```
!
external-aaa pam gid-mapping 100 policy-admin
!
external-aaa pam gid-mapping 100 policy-ro
!
external-aaa pam gid-mapping 500 admin
!
external-aaa pam gid-mapping 500 grafana-admin
!
external-aaa pam gid-mapping 500 policy-admin
!
```

Limitations of MFA Support for Orchestrator CLI

- Only admin users can enable or disable MFA for any user via the CLI. Non-admin users do not have permissions to configure MFA settings.
- MFA can only be enabled for users who have access to both the Orchestrator CLI and the underlying VM (typically external LDAP/GTAC users). MFA is not supported for DRA VM users.
- Orchestrator CLI with MFA can only be accessed from designated servers: DRM, DBM, DIM, or Jump servers. Direct access from other sources is not permitted.
- The same PEM file must be used for both VM authentication and Orchestrator CLI login.
- It is not possible to restrict a user to only the orchestrator CLI without also granting them access to DRM/DBM.
- The `cli-mfa` alias must be properly configured either in artifacts before upgrade or installation to use the MFA CLI command.
- When MFA is enabled for a user, direct SSH login is denied. Access the Orchestrator CLI exclusively through the `cli-mfa` command.
- MFA users must provide both a PEM file and a password to access the CLI. If either credential (PEM file or password) is incorrect or missing, access is denied.
- MFA for CLI does not affect the ability of the user to log in to the GUI, which continues to use password-based authentication.

For more information on the CLI commands, refer to the *CPS vDRA Operation Guide*.

Support for TLS/SSL Encryption in MongoDB

Feature Summary and Revision History

Table 5: Summary Data

Applicable Product(s) or Functional Area	CPS vDRA
Applicable Platform(s)	Not Applicable
Default Setting	Disabled – Configuration Required to Enable

Related Changes in This Release	Not Applicable
Related Documentation	*CPS vDRA Operation Guide*

Table 6: Revision History

Revision Details	Release
First introduced.	25.2.0

Feature Description

This feature introduces TLS/SSL encryption for MongoDB communication within the vDRA environment.

The key aspects of this feature include:

- Securing data in transit between the DRA application and MongoDB, as well as between MongoDB replica set members.
- Utilizing X.509 certificates for authentication and enforcing strong TLS ciphers (TLS 1.2 and 1.3 support).
- Configuring TLS modes such as allowTLS, preferTLS and requireTLS for flexible deployment and migration.

  For more information on the **db-encryption mode set** and **db-encryption enable** CLI commands, refer to the *CLI Commands* chapter in the *CPS vDRA Operation Guide*.
- Importing certificates, enabling, and disabling encryption, changing TLS modes, and synchronizing settings across the cluster through CLI.

Upgrade and downgrade support with TLS encryption

This section describes upgrade, downgrade vDRA versions with TLS encryption, and its limitations.

1. Upgrade support:
   - Upgrading from vDRA version 25.2 to 26.1 or 26.2 with TLS encryption enabled is supported
   - **Prerequisities**: Ensure TLS encryption is properly configured before initiating the upgrade process.
2. Downgrade limitation:
   - Downgrading from version 25.2 to 25.1 with TLS encryption enabled is not supported.
   - TLS encryption must be **disabled** before performing this operation
   - **Recommendation**: Verify that TL.S encryption is disabled prior to downgrade to avoid errors.
3. Upgrade from 25.1 to 25.2:
   - Upgrading from version 25.1 to 25.2 with TLS encryption enabled is not supported.
   - TLS encryption must be disabled before performing this operation
   - **Recommendation**: Consider upgrading directly to a higher version (example, 26.1 or 26.2) if TLS encryption is required.

4. Downgrade from 26.1 to 25.2:
   - Downgrading from version 26.1 to 25.2 with TLS encryption enabled is supported.

Note Ensure the target environment is compatible with TLS encryption settings