

NRF Rolling Software Update

- Feature Summary and Revision History, on page 1
- Feature Description, on page 1
- How it Works, on page 2
- Monitoring and Troubleshooting, on page 3

Feature Summary and Revision History

Summary Data

Table 1: Summary Data

Applicable Product(s) or Functional Area	5G-NRF
Applicable Platform(s)	SMI
Feature Default Setting	Enabled – Always-on
Related Changes in this Release	Not Applicable
Related Documentation	Not Applicable

Revision History

Table 2: Revision History

Revision Details	Release
First introduced.	2026.01

Feature Description

This feature enables NRF to perform a rolling software update for the NRF software.

The NRF software update or in-service update procedure utilizes the K8s rolling strategy to update the pod images. In K8s rolling update strategy, the pods of a StatefulSet are updated sequentially to ensure that the ongoing process remains unaffected. Initially, a rolling update on a StatefulSet causes a single pod instance to terminate. A pod with an updated image replaces the terminated pod. This process continues until all the replicas of the StatefulSet are updated. The terminating pods exit gracefully after competing all the ongoing processes. Other in-service pods continue to receive and process the traffic to provide a seamless software update. You can control the software update process through the Ops Center CLI.

Limitations

In this release, the rolling update feature has the following limitations:

- A rollback or downgrade is not supported.

  If the rolling update fails, redeploy the application by using the previous working release through the NRF Ops Center.

- ANY release to ANY release upgrade is not supported.

How it Works

The NRF software update or in-service update procedure utilizes the K8s rolling strategy to update the pod images. NRF pods are part of StatefulSets and uses the rolling update strategy by default.

1. Rolling upgrade replaces currently running instances of the application with new instances, one by one in reverse ordinal order.
2. NRF performs a health check for the new versions before removing the older versions.
3. In K8s rolling update strategy, the pods of a StatefulSet are updated sequentially to ensure that the ongoing process remains unaffected while increasing traffic to the application.
4. The StatefulSet controller terminates each pod and waits for it to transition to Running and Ready prior to updating the next pod. The StatefulSet controller does not update the next pod until its ordinal successor is Running and Ready. Although, the controller restores any pod that fails during the update to its current version.
5. Pods that have received the update already are restored to the updated version. The rest of the pods yet to receive the update are restored to the previous version. In this way, the controller continues to keep the application healthy and the update consistent in the presence of intermittent failures.
6. If the application pods are getting upgraded, then NRF handles the traffic in the following ways:
   - If one NRF REST endpoint pod is unavailable, then it suspends handling the transactions and the NRF client must retry the messages. In such scenarios, timeouts at NRF clients are expected. NRF client must retry the message on timeout.
   - If one NRF service pod is unavailable, then it suspends handling the transactions and NRF rest endpoint pod must resend the messages to another active pod.
   - When both the NRF REST endpoint and NRF service pods are unavailable, then both the above mentioned scenarios are applicable at the same time.

Note Sometimes, when the NRF dependent pods also gets upgraded (for example, cache-pod), the dependent pods might be in an error state due to incompatibility with the older version pods. In such cases, application does not take any traffic until all the dependent pods are available.

Rolling Software Update Using Inception VM for SMI Cluster Deployer Ops Center

This section describes the procedure for updating the NRF software by using Inception virtual machine (VM) for SMI Cluster Deployer Ops Center.

1. To add a new NRF package or build entry, use the following CLI command:

```
software cnf <package or build name>
 url                                 <NRF repository url>
 password                          <repo password>
 accept-self-signed-certificate true
 sha256                             <SHA256 of the package>
 description                        <description of package (optional)>
exit
```

2. In the respective NRF Ops Center configuration, update the NRF Ops Center repository with the new NRF package or build entry.

```
clusters cluster-gr
ops-centers nrf cisco
  repository-local <package or build name>
  ingress-hostname <ingress host name>
  initial-boot-parameters use-volume-claims false
  initial-boot-parameters first-boot-password <password to be used for Ops Center>
exit
exit
```

3. Delete the old NRF package or build entry (optional).
4. To commit the change, run the following CLI command:

```
commit
```

5. To upgrade the NRF cluster, run the following CLI command:

```
clusters <cluster-name> actions sync run sync-phase opscenter
```

Note:

- The option `sync-phase opscenter` is required for Ops Centers and application upgrades.
- This procedure is also used for patch builds of NRF application pods, for example, the REST endpoint and service pods. This procedure is not meant for NRF Ops Center upgrades.

Monitoring and Troubleshooting

This section provides information regarding monitoring the NRF rolling upgrade.

To monitor the upgrade status for individual NRF StateFulsets, use the following CLI command:

```
kubectl rollout status  sts/nrf-rest-ep-nX -n <namespace>
 kubectl rollout status  sts/nrf-service-nX -n <namespace>
```

Note:

- X represents the node number.
- If there is only one replica, then the complete service is down. For this reason, it is recommended to have a minimum number of replicas as two.
- The upgrades are validated on specified paths within same branch.