Supplement to Orchestration Server Deployment Guide

Configure ORS with Redis Cluster

4/15/2025

Contents

- 1 Configure ORS with Redis Cluster
  - 1.1 Enable Redis
  - 1.2 TLS support for Redis Cluster connection
  - 1.3 Disable persistence and recovery of digital sessions

Configure ORS with Redis Cluster

The following configuration options have been added to the Orchestration `persistence` section to support Redis integration:

- **type** (it is recommended to set this to `redis` only after configuring the rest of the options)
- **redis-nodes**
- **username** (optional)
- **password** (optional)
- **tls-enabled** (optional)
- **tls-cert** (optional)
- **tls-key** (optional)
- **tls-cacert** (optional)
- **tls-cacertdir** (optional)
- **tls-sni** (optional)

Refer to the **persistence Section** in Application-Level Options for detailed description of the above options.

Enable Redis

- Configure the following options:
  - **type**
  - **redis-nodes**
- Set **sessionid-with-nodeid** to `true`

You must restart the ORS application for any changes in these options to take effect. Verify that the Redis connection works as expected using clients such as **RedisInsight** before enabling it with ORS.

TLS support for Redis Cluster connection

To encrypt the data communication between ORS and Redis databases, TLS support must be enabled on both ORS and Redis sides. Enabling TLS could impact the performance, so it must be taken into consideration before switching from non-TLS connection to encrypted. Refer to Redis documentation, https://redis.io/docs/management/security/ to know more about TLS support in Redis and how to enable it. Verify that the Redis TLS connection works as expected using clients such as **RedisInsight** before enabling it with ORS.

To enable ORS to connect to Redis Cluster with TLS support, specify the following ORS application options:

- **tls-enabled**
- **tls-cert**
- **tls-key**
- **tls-cacert**

Although **tls-cert** and **tls-key** are optional, if you specify one of them, you must also specify the other. Instead of specifying **tls-cacert**, you can also specify **tls-cacertdir** to the directory where certificates are stored.

> **Important**
>
> The TLS feature is currently not enabled on Windows platform due to its limited support by Redis Plus-plus library which ORS uses to communicate with Redis Cluster.

Disable persistence and recovery of digital sessions

Generally, we recommend to disable persistence and recovery of digital sessions. In case of an ORS switchover, failover, or disconnect, digital interactions are immediately placed back in queue by eServices. This is the major difference from voice interactions, Therefore, there is no need to persist and recover digital sessions. When the connection between ORS and Interaction Server is restored, interactions will continue to be processed according to the interaction queue logic. Even for chat interactions, there is no need for session recovery, as chat logic is not handled by ORS session. There is no risk of incomplete or stuck session on ORS due to restoration or if an event is missed. Persistence of digital session puts performance penalty on ORS applications and Cassandra or Redis. Also persistent data for digital sessions put extra capacity requirement for Cassandra or Redis in-memory data storage.

Therefore, persistence and recovery of digital sessions triggered by eServices interaction to Redis is disabled.