Standard Operating Procedure (SOP)

Setting Up a 6-Node Redis Cluster Using Docker on RHEL 9 Containers

Your Name

August 12, 2024

Contents

1 Introduction

This Standard Operating Procedure (SOP) details the steps to set up a 6-node Redis cluster using Docker with Red Hat Enterprise Linux (RHEL) 9 containers. The cluster configuration includes 3 master nodes and 3 replica nodes. This setup leverages Docker Desktop to manage the RHEL 9 containers.

2 Prerequisites

Before proceeding with the setup, ensure the following prerequisites are met:

- Docker Desktop installed on your host system.
- 6 Docker containers running RHEL 9.
- Basic knowledge of Docker and Linux command-line operations.
- Access to the official Redis installation packages within the RHEL 9 containers.

3 Step 1: Install Redis on Each RHEL 9 Container

3.1 Connect to Each Container

To install Redis, first connect to each RHEL 9 container via the terminal:

```
docker exec -it <container_name> /bin/bash
```

Listing 1: Connecting to a Container

Replace $<container_name> with the actual name of the container (e.g., rhel9-node-1).

3.2 Install Redis

Once inside the container, update the package manager and install Redis:

```
sudo dnf update -y
sudo dnf install -y redis
```

Listing 2: Installing Redis

3.3 Configure Redis

Edit the Redis configuration file located at /etc/redis/redis.conf:

```
sudo vi /etc/redis/redis.conf
```

Listing 3: Editing Redis Configuration

Make the following changes to enable clustering:

```
bind 0.0.0.0
protected-mode no
cluster-enabled yes
cluster-config-file nodes.conf
```

```
cluster-node-timeout 5000
appendonly yes
```

Listing 4: Redis Configuration Changes

Save the file and exit.

3.4 Start Redis

Start the Redis service inside the container:

```
sudo systemctl start redis
```

Listing 5: Starting Redis

Repeat these steps for all six RHEL 9 containers.

4 Step 2: Set Up Docker Networking

4.1 Create a Custom Docker Network

Create a custom Docker network that all Redis containers will use:

```
docker network create redis-cluster-net
```

Listing 6: Creating Docker Network

4.2 Connect Containers to the Network

Ensure each RHEL 9 container is connected to the newly created network:

```
docker network connect redis-cluster-net <container_name>
```

Listing 7: Connecting Containers to Network

Replace <container$_name$> $with the actual names of your containers$.

5 Step 3: Initialize the Redis Cluster

5.1 Obtain Container IP Addresses

Retrieve the IP addresses of the containers to configure the Redis cluster:

```
for i in ‘seq 1 6‘; do \
docker inspect -f ’{{range.NetworkSettings.Networks}}{{.IPAddress}}{{end}}’
    ↪ rhel9-node-$i; \
done
```

Listing 8: Getting Container IP Addresses

Note the IP addresses for the next step.

5.2 Create the Redis Cluster

From inside one of the containers (e.g., rhel9-node-1), use the ‘redis-cli‘ to create the cluster:

```
redis-cli --cluster create <IP1>:6379 <IP2>:6379 <IP3>:6379 <IP4>:6379 <IP5
    ↪ >:6379 <IP6>:6379 --cluster-replicas 1
```

Listing 9: Creating the Redis Cluster

Replace ‘<IP1>‘ to ‘<IP6>‘ with the IP addresses of your containers.

5.3 Verify the Cluster Setup

Verify that the Redis cluster has been properly configured:

```
redis-cli cluster nodes
```

Listing 10: Verifying the Cluster

This command should display a list of all nodes and their roles within the cluster.

6 Step 4: Post-Setup Configuration

6.1 Enable Persistence

Ensure Redis persistence is enabled by verifying the ‘appendonly‘ setting in each container:

```
redis-cli CONFIG GET appendonly
```

Listing 11: Verifying Persistence Configuration

Adjust the configuration if necessary to ensure data persistence.

6.2 Monitor the Cluster

Use the following command to monitor the health and performance of the Redis cluster:

```
redis-cli cluster info
```

Listing 12: Monitoring Cluster Health

Consider setting up automated monitoring tools to maintain cluster health.

7 Conclusion

This SOP provides a comprehensive guide to setting up a 6-node Redis cluster using Docker with RHEL 9 containers. This setup is efficient and scalable, suitable for development and production environments. Regular monitoring and maintenance are essential to ensure optimal cluster performance.

8 References

- Redis Cluster Tutorial
- Docker Documentation
- Redis Documentation