**Steps to install master and slave Redis with Sentinel in multiple nodes**

Redis requires a gcc compiler to run.

The latest version of Redis is 6.3.0 which requires a higher gcc version (latest 9.x). The default gcc package available for RHEL7 is gcc 4.8.x which supports upto Redis 5.0.9.

**Pre-requisites:**

1. Redis setup requires Developer Tools and gcc to be installed beforehand.
2. A user with sudo privileges should be present (The user is redisops in the below steps). All installation and setup work for Redis and Sentinel should be done using this dedicated user.
3. Port 6379 and 26379 (ports on which Redis and Sentinel run) should be open for the other nodes to connect to.

In the below steps we use 1 master and 2 slave nodes.

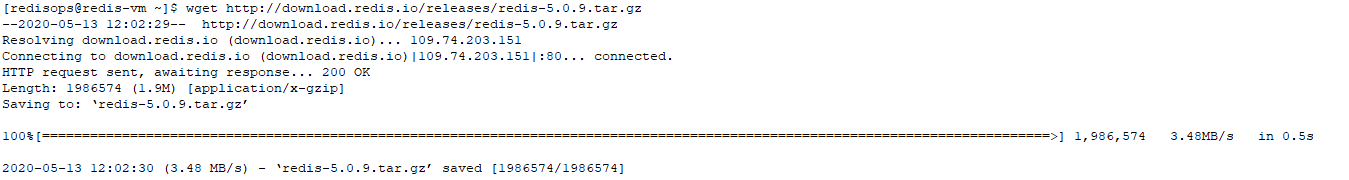
The master node will be referred to as VM1 and the slave nodes as VM2 and VM3.

**Steps:**

**In VM1, VM2, VM3**

1. Download the latest Redis tar ball. Replace <version> with the corresponding version according to the gcc version installed.

**wget** [http://download.redis.io/releases/redis-<version>.tar.gz](http://download.redis.io/releases/redis-%3cversion%3e.tar.gz)



The following steps use redis-5.0.9

Untar the downloaded tar ball.

**tar -xvzf redis-5.0.9.tar.gz**

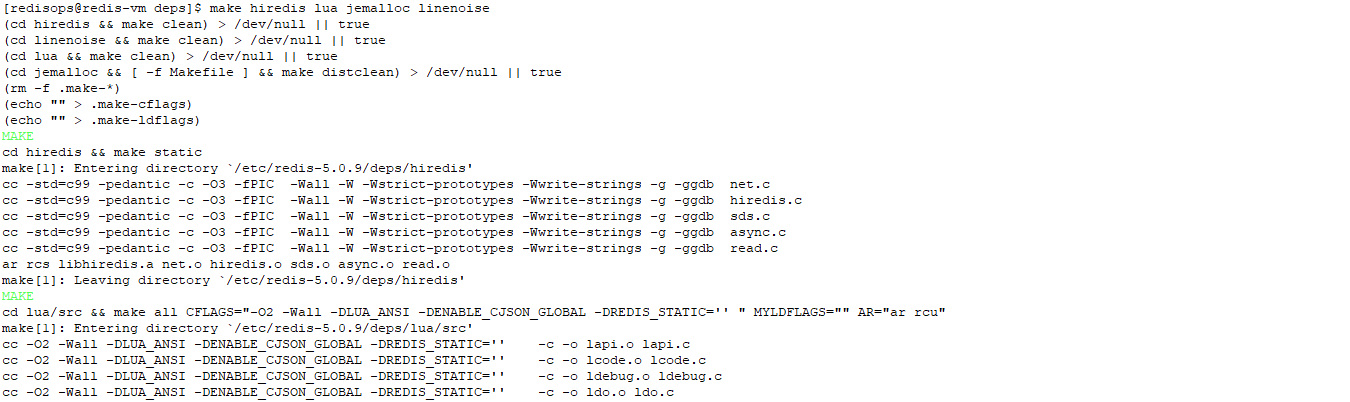
1. Move the folder to /etc

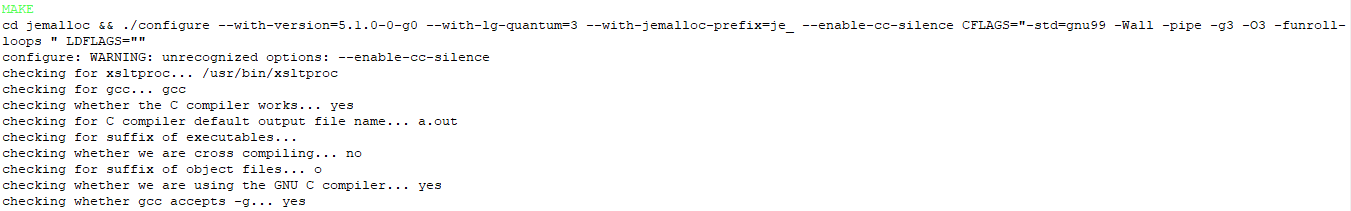
**sudo mv redis-5.0.9 /etc**

1. Compile Redis

**cd /etc/redis-5.0.9/deps/**

**make hiredis lua jemalloc linenoise**



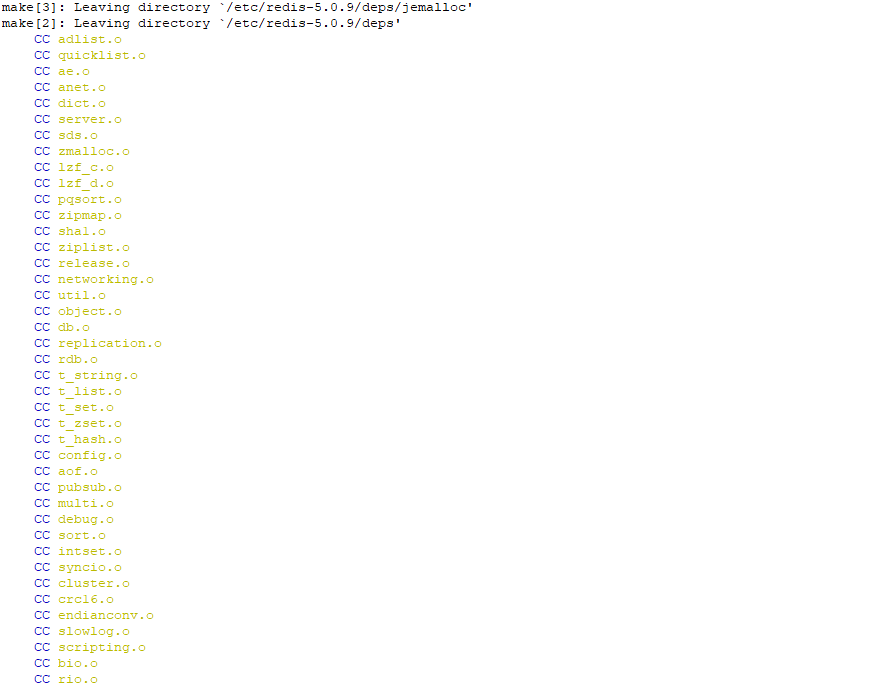




**cd ..**

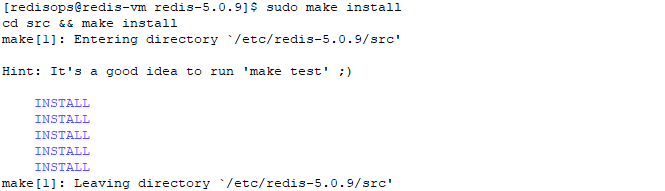
**make**





1. Send the redis server and the command line executables to their proper place in /usr/local/bin/

**sudo make install**



1. Make a log directory in /var/log and create log files for the redis and sentinel processes. Give the redisops user permission to the folder and files.

**cd /var/log**

**sudo mkdir redis**

**cd redis**

**sudo touch redis.log sentinel.log**

**cd ..**

**sudo chown -R redisops:redisops redis/**

1. Make a data directory in /var/lib/ and create directories for the redis and sentinel processes to store any data. Give the redisops user permission to the folders.

**cd /var/lib/**

**sudo mkdir redis**

**cd redis**

**sudo mkdir redis sentinel**

**cd ..**

**sudo chown -R redisops:redisops redis/**

Then return to /etc/redis-5.0.9

**cd /etc/redis-5.0.9**

**In VM1**

1. In redis.conf add/edit the following lines (making it the master initially)

Replace <host ip> with the internal/external IP address of the VM according to requirement

**vim redis.conf**

bind <host ip>

port 6379

supervised systemd

logfile "/var/log/redis/redis.log"

dbfilename dump.rdb

dir /var/lib/redis/redis

**In VM2, VM3**

1. In redis.conf add/edit the following lines (making them the slaves initially)

Replace <host ip> with the internal/external IP address of the VM according to requirement.

Replace <master ip > with the IP given in the master.

**vim redis.conf**

bind <host ip>

port 6379

supervised systemd

logfile "/var/log/redis/redis.log"

dbfilename dump.rdb

dir /var/lib/redis/redis

slaveof <master ip> 6379

slave-priority 100

**In VM1, VM2, VM3**

1. In sentinel.conf add/edit the following lines.

Replace <host ip> with the internal/external IP address of the VM according to requirement.

Replace <master ip > with the IP given in VM1.

**vim sentinel.conf**

bind <host ip>

port 26379

logfile "/var/log/redis/sentinel.log"

dir /var/lib/redis/sentinel

sentinel monitor redis-master <master ip> 6379 1

sentinel down-after-milliseconds redis-master 30000

sentinel parallel-syncs redis-master 1

sentinel failover-timeout redis-master 180000

1. Go to /etc/systemd/system/ and create 2 service files for the Redis and Sentinel processes.

redis.service, sentinel.service

**cd /etc/systemd/system/**

**sudo touch redis.service sentinel.service**

Edit the files and add the following lines in them

**sudo vim redis.service**

[Unit]

Description=Advanced key-value store

After=network.target

[Service]

User=redisops

Group=redisops

ExecStart=/usr/local/bin/redis-server /etc/redis-5.0.9/redis.conf

Restart=on-abnormal

[Install]

WantedBy=multi-user.target

**sudo vim sentinel.service**

[Unit]

Description=Advanced key-value store

After=network.target

[Service]

User=redisops

Group=redisops

ExecStart=/usr/local/bin/redis-server /etc/redis-5.0.9/sentinel.conf --sentinel

Restart=on-abnormal

[Install]

WantedBy=multi-user.target

1. Enable and start the services

**sudo systemctl enable redis**

**sudo systemctl enable sentinel**

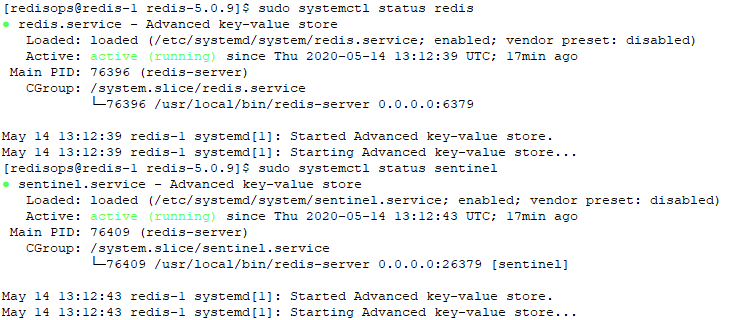
**sudo systemctl start redis**

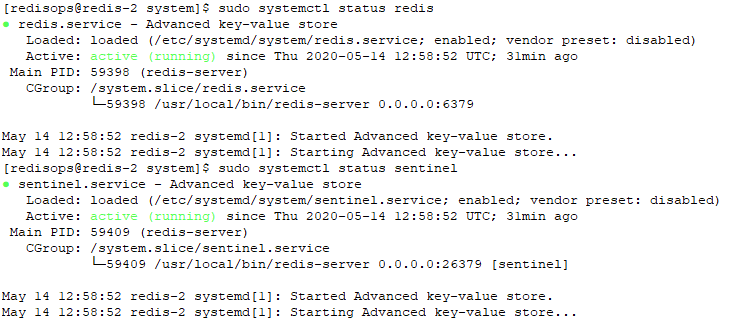
**sudo systemctl start sentinel**

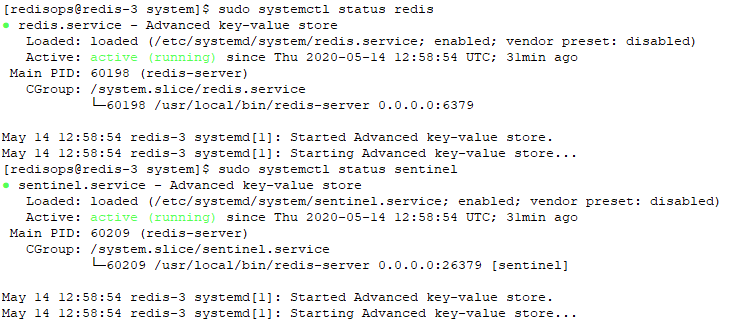
1. Check the status of the services to see if they are all running.

**sudo systemctl status redis**

**sudo systemctl status sentinel**

****

****

****

**We can check whether Redis and Sentinel have been setup properly using the following steps:**

1. Run the redis-cli in VM1, VM2, VM3 and check if values can be stored and retrieved.

Replace <ip> with the IP the redis instance is bound to.

First run the redis-cli for all the Redis processes in VM1, VM2, VM3. Enter ping and if PONG is returned then the redis processes are running fine.

**redis-cli –h <ip>**

**10.0.0.9:6379> ping**

**PONG**

**redis-cli –h <ip>**

**10.0.0.10:6379> ping**

**PONG**

**redis-cli –h <ip>**

**10.0.0.11:6379> ping**

**PONG**

**test 1-1.PNG**

**test 1-2.PNG**

**test 1-3.PNG**

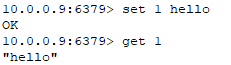
Then set and get some values in VM1 to see if data is being stored and retrieved.

**10.0.0.9:6379> set 1 hello**

**OK**

**10.0.0.9:6379> get 1**

**"hello"**

****

Then run the redis-cli for VM2, VM3 and see if the get command for the same key returns the same value. This will verify whether the replication is done properly.

**10.0.0.10:6379> get 1**

**"hello"**

**10.0.0.11:6379> get 1**

**"hello"**

**test 1-2.PNG**

**test 1-3.PNG**

1. Check in any node which process is the master process

**redis-cli -p 26379 sentinel get-master-addr-by-name redis-master**

**1) "10.0.0.9"**

**2) "6379"**

**test 2-1.PNG**

This shows the ip and port the master process is running on (This should be the IP of VM1).

Stop the master process in VM1 and run the command again to check whether any one of the slave processes of VM2 or VM3 have been promoted to master.

**sudo systemctl stop redis** (Run this in VM1 only)

**redis-cli -p 26379 sentinel get-master-addr-by-name redis-master**

**1) "10.0.0.10"**

**2) "6379"**

test 2-2.PNG