Setting up a MongoDB cluster with replica set

Created by **Gary Beason**, last modified by **Michele Marques** on **Jul 23, 2015**

The MongoDB database provides replica sets to support high availability and automatic failover. Replica sets use the following types of servers.

Important
For information about setting up a secure MongoDB installation, see Configuring MongoDB for BMC MyIT and Smart IT.

Note
The MongoDB installed with BMC MyIT does not support replication. You must download and install MongoDB before installing BMC MyIT. For supported versions, see System requirements.

- **Primary server** — It stores the data, all write/read operation goes to this server from any client. One server is always primary.
- **Secondary server —** It stores backup data and stays in sync with the primary server. If the primary server is not available, then replica set elects one of these servers to become the primary server.
- **Arbiter** — It stores no data, cannot become the primary server during failover, and only participates in the election process during failover. Typically, arbiters are only required for an even number of secondary servers in order to break ties.

This topics provides the following information:

- To create the MongoDB servers and the replica set information
- To include the servers in the replica set
- Where to go from here
- Related topics

For more information, see Replication Concepts in the MongoDB documentation.

To create the MongoDB servers and the replica set information

1. Create and set up three new instances of MongoDB server as specified below.
   Do not start these servers.

server-1

Setting	Value
Port	27017
Directory path	directoryPath\rs1 Example: **M:\workspace\mongo\rs1\bin**

server-2

Setting	Value
Port	27017
Directory path	directoryPath\rs2 Example: **M:\workspace\mongo\rs2\bin**

server-3

Setting	Value
Port	27017
Directory path	directoryPath\rs3

	Example: **M:\workspace\mongo\rs3\bin**

2. In the **mongod.conf** file for each server, add the following properties:

   replSet — unique name for replica set which is given to all members

   rest — Enables rest interface for the administration web page

   Example:
   ```
   replSet=calbrors0
   rest=true
   ```

3. Start the server in any order.
   Example:
   ```
   rs1 >> mongod -f \rs1\mongod.conf
   rs2 >> mongod -f \rs2\mongod.conf
   rs3 >> mongod -f \rs3\mongod.conf
   ```

4. Continue to seting up the MongoDB servers to be part of the replica set.

To include the servers in the replica set

Though all the servers are up, they are not connected to each other and not part of replica set. In the following examples, one server will be primary, and two will be secondary servers.

1. Connect to any one of the servers using the mongo shell provided in the MongoDB binaries.
   Example: `rs1 >/rs1/bin/mongo.exe --port 27017`
2. To create the replica set configuration on one of the servers, use the `rs.initiate()` command in the mongo shell.
   This server will be the primary server.
   Example:
3. (*optional*) To check how many servers are in the replica set, use the `rs.status()` command in the mongo shell.
4. To add secondary servers, use the `rs.add("server/hostname:port")` command on each server.
   Example:
   `rs.add(“<machine/host-name>:27017”)`
5. (*optional*) To add the arbiter, use `addArb("server/hostname:port")` command.
6. Check the status with the `rs.status()` command in the mongo shell.
   If you enabled the rest parameter, you can also enter the URL **http://*machine-name*:*port*/_replSet** in a browser.
   Example: **http://calbro-pc1:27017/_replSet**

Example configuration

```
{
        "_id" : "social",
        "version" : 1,
        "members" : [
                {
                        "_id" : 0,
                        "host" : "10.170.139.27:27017"
                },
                {
                        "_id" : 1,
                        "host" : "10.170.139.28:27017"
                },
                {
                        "_id" : 2,
                        "host" : "10.170.139.29:27017"
                }
        ]
}
```

Where to go from here

Setting up ElasticSearch with MongoDB

Related topics

Installation worksheet

MongoDB installation