

HOW TO SETUP A CASSANDRA CLUSTER



Here we show how to set up a Cassandra cluster. We will use two machines, 172.31.47.43 and 172.31.46.15. First, open these firewall ports on both:

7000
7001
7199
9042
9160
9142

Then follow [this document](#) to install Cassandra and get familiar with its basic concepts. Make sure to install Cassandra on each node.

(This article is part of our [Cassandra Guide](#). Use the right-hand menu to navigate.)

Configure Cluster Setting

There is no central master in a Cassandra cluster. Instead you just make each one aware of the others and they work together.

First we will edit `/etc/cassandra/cassandra.yaml` on both machines set the the values as shown in the table below. Don't change the cluster name yet. We will do that later.

- **seeds**—set the IP address on one machine to be the seed. It is not necessary that all machines be seeds. **Seeds** are nodes that Cassandra nodes use when you start Cassandra start to find other nodes.

- **listen_address**—the IP address for Cassandra to run.
- **endpoint_snitch**—this is used to determine where to route data and send replicas. We use the default below. There are several. The others are rack-aware, meaning they would not put a replica on the same physical storage rack as another. If you did that and the whole rack failed the data could be lost. There is even one (Ec2Snitch) designed for Amazon EC2 that can spread data across Amazon Zones.

```
machine 172.31.46.15 settings          machine 172.31.47.43 settings
endpoint_snitch: SimpleSnitch        endpoint_snitch: SimpleSnitch
```

```
- seeds: "seeds: 172.31.47.43" - seeds: "seeds: 172.31.47.43"
```

```
listen_address: 172.31.46.15    listen_address: 172.31.47.43
```

Now run on both machines:

```
sudo service cassandra start
```

Then wait a few seconds for discovery to work and then run on both machines:

```
nodetool status
```

It should show both nodes:

```
Datacenter: datacenter1
```

```
=====
```

```
Status=Up/Down
```

```
|/ State=Normal/Leaving/Joining/Moving
```

```
-- Address          Load          Tokens          Owns (effective)  Host ID
Rack
UN  172.31.46.15    245.99 KiB    256             100.0%            fb1d89bb-
cbe2-488f-b2e7-da145bd2dde7 rack1
UN  172.31.47.43    196.01 KiB    256             100.0%            472fd4f0-9bb3-48a3-a933-9c9b07f7a9f6 rack1
```

If you get any kind of error message look in `/var/log/cassandra/system.log`

Now let's change the name of the cluster from the default. Run `cqlsh` and then paste in the SQL below. Cassandra does not replicate this system change across the cluster so you have to run this on both machines.

```
UPDATE system.local SET cluster_name = 'Walker Cluster' where key='local';
```

Now edit `/etc/cassandra/cassandra.yaml` and change the cluster name to whatever you want. It should be the same on both machines:

```
cluster_name: 'Walker Cluster'
```

Then:

```
sudo service cassandra restart
```

Run this check again:

```
nodetool status
```

```
Datacenter: datacenter1
```

```
=====
```

```
Status=Up/Down
```

```
|/ State=Normal/Leaving/Joining/Moving
```

```
-- Address          Load           Tokens         Owns (effective)  Host ID
Rack
UN  172.31.46.15    312.4 KiB     256            100.0%            fb1d89bb-
cbe2-488f-b2e7-da145bd2dde7 rack1
UN  172.31.47.43    294.71 KiB    256            100.0%
472fd4f0-9bb3-48a3-a933-9c9b07f7a9f6 rack1
```

Now, following [these instructions](#) from our introduction to cassandra, let's create some data. We will see that data entered on one node is replicated to another. Paste these SQL commands into csq:

```
CREATE KEYSPACE Library
  WITH REPLICATION = { 'class' : 'SimpleStrategy', 'replication_factor' :
3 };
```

```
CREATE TABLE Library.book (
ISBN text,
copy int,
title text,
PRIMARY KEY (ISBN, copy)
);
```

```
CREATE TABLE Library.patron (
ssn int PRIMARY KEY,
checkedOut set
);
```

```
INSERT INTO Library.book (ISBN, copy, title) VALUES('1234',1, 'Bible');
INSERT INTO Library.book (ISBN, copy, title) VALUES('1234',2, 'Bible');
INSERT INTO Library.book (ISBN, copy, title) VALUES('1234',3, 'Bible');
INSERT INTO Library.book (ISBN, copy, title) VALUES('5678',1, 'Koran');
INSERT INTO Library.book (ISBN, copy, title) VALUES('5678',2, 'Koran');
```

Then logon to the opposite machine and verify that the data has been copied there:

```
select * from Library.book;
```

```
isbn | copy | title
-----+-----+-----
5678 | 1 | Koran
5678 | 2 | Koran
```

1234	1	Bible
1234	2	Bible
1234	3	Bible