

## Exercise 11 – Snitches

In this exercise, you will:

- Understand how to make Apache Cassandra™ aware of the cluster topology using Snitches.

To improve fault-tolerance, distributed systems must consider the topology of the cluster into account. Apache Cassandra™ uses this topological information to replicate data across a geographically diverse area - which increases data availability.

In this exercise we will change the snitch and rack assignments for our two nodes. We will decommission each node (remove it from the cluster), reset the node's data, etc.

### Steps

- 1) We will first clear out your current nodes.
- 2) Run the following commands to remove stop both nodes:

```
/home/ubuntu/node2/resources/cassandra/bin/nodetool stopdaemon  
/home/ubuntu/node1/resources/cassandra/bin/nodetool stopdaemon
```

These commands will take some time to complete.

- 3) For both nodes, delete all the files in the data/ folder.

```
rm -rf /home/ubuntu/node1/data  
rm -rf /home/ubuntu/node2/data
```

- 4) Edit `/home/ubuntu/node1/resources/cassandra/conf/cassandra.yaml` and find the `endpoint_snitch` setting.

NOTE: Using `endpoint_snitch` default `DseSimpleSnitch` places your node in a datacenter that is based upon work type

```
# You can use a custom Snitch by setting this to the full class name  
# of the snitch, which will be assumed to be on your classpath.  
endpoint_snitch: com.datastax.bdp.snitch.DseSimpleSnitch
```

- 5) Change the `endpoint_snitch` to `GossipingPropertyFileSnitch`.

```
# You can use a custom Snitch by setting this to the full class name
# of the snitch, which will be assumed to be on your classpath.
endpoint_snitch: GossipingPropertyFileSnitch
```

- 6) Save and close the `cassandra.yaml` file.
- 7) Make the same change to node 2's `cassandra.yaml` file as well.

- 8) Edit `/home/ubuntu/node1/resources/cassandra/conf/cassandra-rackdc.properties` file.

```
# These properties are used with GossipingPropertyFileSnitch and will
# indicate the rack and dc for this node
dc=dc1
rack=rack1
```

This is the file that the `GossipingPropertyFileSnitch` uses to determine the rack and data center this particular node belongs to.

NOTE: Racks and datacenters are purely logical assignments to Apache Cassandra™. You will want to ensure that your logical racks and data centers align with your physical failure zones.

- 9) Change the `dc` to `west-side` and the `rack` to `hakuna-matata`.

```
# These properties are used with GossipingPropertyFileSnitch and will
# indicate the rack and dc for this node
dc=west-side
rack=hakuna-matata
```

- 10) Save and close the `cassandra-rackdc.properties` file.

- 11) Edit `/home/ubuntu/node2/resources/cassandra/conf/cassandra-rackdc.properties` and change its `dc` to `east-side` and `rack` to `hakuna-matata`. Although the rack names between the two data centers are the same, each rack lives in a unique data center (`west-side` vs. `east-side`).

```
# These properties are used with GossipingPropertyFileSnitch and will
# indicate the rack and dc for this node
dc=east-side
rack=hakuna-matata
```

- 12) Start node1. Once it is up, start node2.

13) Once both nodes are up, execute:

```
/home/ubuntu/node1/bin/dsetool status
```

Notice the differing datacenter assignments now.

```
DC: east-side      Workload: Cassandra      Graph: no
=====
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Server ID      Address      Load          Owns  VNodes  Rack          Health [0,1]
UN   06-74-00-EF-3F-E8  127.0.0.2  142.47 KiB   ?    128     hakuna-matata 0.20

DC: west-side      Workload: Cassandra      Graph: no
=====
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Server ID      Address      Load          Owns  VNodes  Rack          Health [0,1]
UN   06-74-00-EF-3F-E8  127.0.0.1  117.54 KiB   ?    128     hakuna-matata 0.20
```