Exercise 14 – Hinted Handoff

In this exercise, you will:

- Observe how hinted-handoff keeps Apache Cassandra™ write-data consistent

Availability can affect data consistency. Hinted-handoff helps Apache Cassandra(TM) maintain consistency even when a node is down.

Steps

1) Ensure that all your nodes are up before beginning this exercise.

2) Determine which nodes are responsible for the `cassandra` partition in `videos_by_tag` by executing the following command in the terminal:

   ```
   /home/ubuntu/node1/resources/cassandra/bin/nodetool getendpoints
   killrvideo videos_by_tag 'cassandra'
   ```

3) Bring down both of those nodes. Wait for them to terminate before continuing.

4) Switch to the killrvideo keyspace:

   ```
   USE killrvideo;
   ```

5) Change your consistency level to ANY:

   ```
   CONSISTENCY ANY;
   ```

   NOTE: When a client writes with `Consistency ANY`, storing a hint is sufficient for the write to be successful. `Consistency ONE` requires that at least one successful node write - a hint is not sufficient.

6) Execute the following INSERT command:

   ```
   INSERT INTO videos_by_tag(tag, added_date, video_id, title)
   VALUES ('cassandra', '2016-2-11', uuid(), 'Cassandra, Take Me
   Home');
   ```

The write will succeed even though both replica nodes are down. The one and only node left in the cluster stores the writes as hints for the two replica nodes until they come back online.

7) You can see that the single node (i.e., the node still running) stored a hint by navigating to its `/home/ubuntu/nodeX/data/hints/` directory.

ubuntu@ds201-node1:~$ ls -l /home/ubuntu/node3/data/hints
total 8
-rw-rw-r-- 1 ubuntu ubuntu 163 Apr 11 13:10 0072bef2-87e0-4422-9a20-2a3b7244392f-1491916240152-1.hints
-rw-rw-r-- 1 ubuntu ubuntu 163 Apr 11 13:10 cb5542bc-b339-4efc-a994-ee660d510e53-1491916240156-1.hints

There are two files, each storing all the hints for a single node.

8) You may also want to try navigating to the other node hints directory and confirm that they are empty (meaning neither is storing hints).

```
ubuntu@ds201-node1:~$ ls -l /home/ubuntu/node1/data/hints
total 0
ubuntu@ds201-node1:~$ ls -l /home/ubuntu/node2/data/hints
total 0
```

9) Set your consistency level back to ONE:

CONSISTENCY ONE;

10) Try reading the new record back by executing the following SELECT command:

SELECT *
FROM killrvideo.videos_by_tag;

The query fails because even though one node is up, that node doesn't cover the entire token range itself. So, Apache Cassandra™ cannot be sure that all records will be accounted for even with a consistency level of ONE.

11) Bring one replica node back up. Wait for to startup before continuing. Investigate the `hints/` directory. You will see that one of the hint files has disappeared. This is due to this node forwarding the hints to the newly bootstrapped node.

```
ubuntu@ds201-node1:~$ ls -l /home/ubuntu/nodeX/data/hints
total 4
```

```
-rw-rw-r-- 1 ubuntu ubuntu 163 Apr 11 13:10 0072bef2-87e0-4422-
9a20-2a3b7244392f-1491916240152-1.hints
```

12) Now try the SELECT command one more time:

```
SELECT *
FROM killrvideo.videos_by_tag;
```

The query succeeds and returns our new row.

13) Now bring up the remaining offline node. Wait for it to start up before proceeding.

Again, you will notice some file activity in the `hints/` folder, and eventually the folder will clear.

```
ubuntu@ds201-node1:~$ ls -l /home/ubuntu/node3/data/hints
total 0
```

Thus, the two replica nodes now each have their hinted records.

NOTE: Pre-Apache Cassandra™ 3.0 versions stored their hints in actual tables.