Exercise 7c

Creating a Cassandra Cluster in EC2

Prior Knowledge
Unix Command Line Shell
Amazon AWS / EC2 Console
Previous Cassandra exercises

Learning Objectives
Demonstrate Cassandra replication and scaling

Software Requirements
- AWS, Cassandra AMI

Overall plan
We are going to start a new (medium size) instance and deploy DataStax OpsCenter on it. From there we are going to create a cluster on EC2 (OpsCenter will do all the hard work for us). Finally we will install the Cassandra stress tools on the opscenter machine and test the cluster. We can view the cluster status on the dashboard.

1. Start up the AWS Console and login.

2. Make sure you are using the eu-west (Ireland) region.

3. Check that you can see the cassandra-security-group

4. Start up a new Launch Wizard and choose the same Ubuntu 14.04 VM we've used previously.

5. Choose **t2.medium** as the instance size.

6. Click through Next repeatedly until you get to Tag instance and call it: oxcloXX-opscenter

7. **Next: Configure Security Group**

8. Choose Existing Group and then **cassandra-security-group**

9. Continue to Launch the instances and choose your own keypair (so that you can access the instances).

10. Find the IP address of the new instance and once it has started up, then SSH to the instance.

11. In the SSH session type (or cut and paste):
    curl -L http://downloads.datastax.com/community/opscenter.tar.gz | tar xz

12. Now
    cd opscenter-5.2.2/

13. By default there is no security on the opscenter. Ideally we would enable full SSL, certificates, etc. But that is overkill for this lab. On the other hand, leaving it wide open on an internet facing host is a bit mad, so we'll turn on authentication (even if its only over HTTP and not HTTPS).

    nano conf/opscenterd.conf

14. Now change the line under [authentication] from enabled=False to enabled=True

15. Start the opscenter:
    bin/opscenter

16. Now from your browser browse ip:8888 (where ip is the ipaddress of your instance).

17. Login with admin/admin
    You should see:
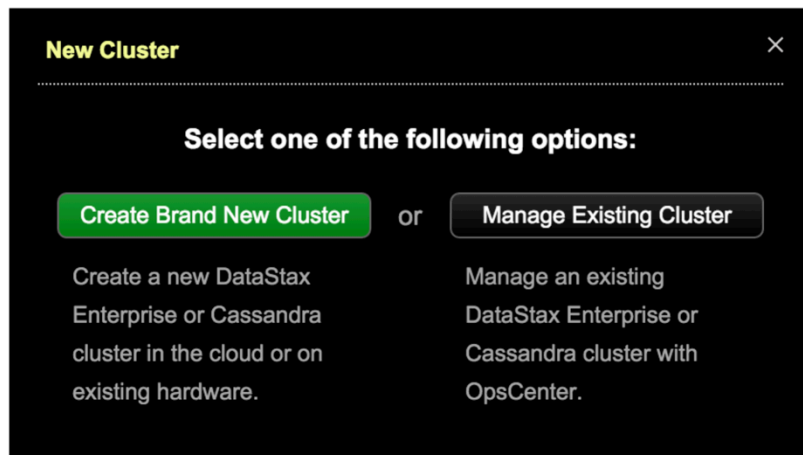
18. Click on Create Brand New Cluster

19. Now fill in the following:

Cluster name: oxcloXX-cassandra
Provisioning type: Cloud
Package: Datastax Community 2.1.11
Access Key/Secret Access Key: your keys from exercise 1

20. Now click Add Datacenter

21. On the next page fill in:
    Number of Nodes: 3
    Leave alone the next few
    Size: m3.large
    Untick **Use OpsCenter specific security group**
     since we already have a Cassandra security group
    Security Group: cassandra-security-group
    Untick **Use OpsCenter specific keypair**
   Keypair   your keypair

    Now copy and paste the contents of your keypair .PEM file into the box
    (it will look a bit like this but longer):

    ```
    -----BEGIN RSA PRIVATE KEY-----
    MIIEogIBAAKCAQEAjk5z0qNOjayxkP//c6nSxD17CfLPYPwgh0iJ0dy+up4pYkrW/3OYoqVDPlWE
    …
    t6TYiViupkIXqzwdkrkvWkgUqyVqfxzBRhsrcsGHOyWzyZFSS45+eYh8mx00eLB4dhw=
    -----END RSA PRIVATE KEY-----
    ```

    It should look like:

22. Click Add

23. On the review page click **Build Cluster** and you will see:

Build in Progress

Launching instances (0/3)
This may take a few minutes
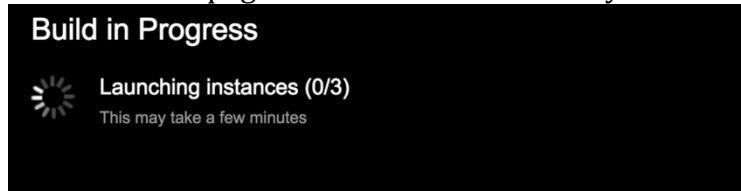
24. Go check on the EC2 Instances page and you should see three new instances being launched.

25. Now that we have completed the wizard, opscenter will let us change our password from admin. Best do that right away.

26. Meanwhile while this is working we can install the stuff we need on our opscenter instance (which I assume you are still SSHed into!).

    Type (each on one line) This snippet is available here:
    http://freo.me/oxclo-instcass

```
echo "deb http://debian.datastax.com/community stable main" | sudo tee -a
/etc/apt/sources.list.d/cassandra.sources.list

curl -L http://debian.datastax.com/debian/repo_key | sudo apt-key add

sudo apt-get update

sudo apt-get install openjdk-7-jre-headless -y

sudo apt-get install cassandra=2.1.11 cassandra-tools=2.1.11 -y
```

27. Now find the IP address of one of the newly created nodes. You can see the IP address from the EC2 dashboard or from the opscenter webpage. Check that you can connect to the cluster using cqlsh from your SSH session:

    cqlsh ipaddress

    You should see something like:

```
Connected to oxclo03-cass at 172.31.6.174:9042
[cqlsh 5.0.1 | Cassandra 2.1.11 | CQL spec 3.2.1 | Native
protocol v3]
Use HELP for help.
```

28. Now let's try cassandra-stress again:

```
cassandra-stress write n=10000 -node ip-address
```

29. You will see some log showing that the stress tool is now connected to three different servers. You can check on the cluster from the opscenter. Have a good browse around. You can see nice graphs of the status like this. You can also add new graphs to the dashboard.
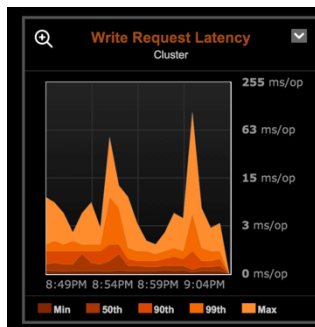

30. Once the write test is finished, do a mixed test:
    ```
    cassandra—stress mixed n=10000 —node ip—address
    ```

31. While the mixed test is running, you can go to the opscenter. Click on the nodes icon and then click on part of the ring. You will see a panel pop up specific to that node.
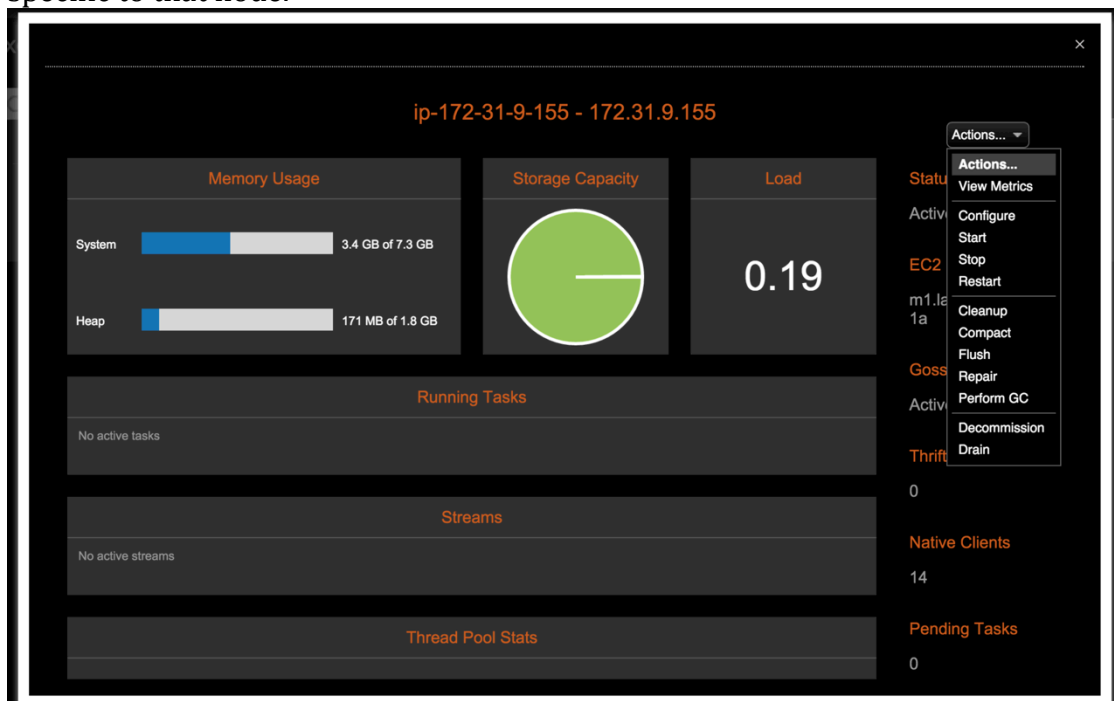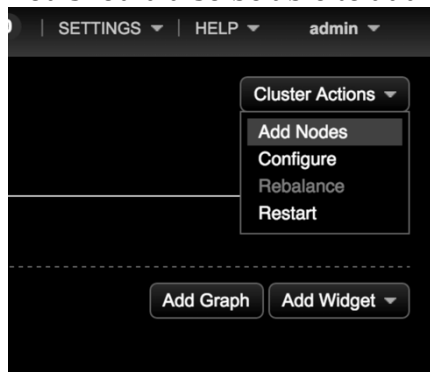

32. Try stopping the node.

33. Wait a bit and restart it.

34. Ideally the stress test will continue, but just a bit slower.

35. You can even restart the whole cluster while the test is running (as long as quorum is maintained). In the top right corner:
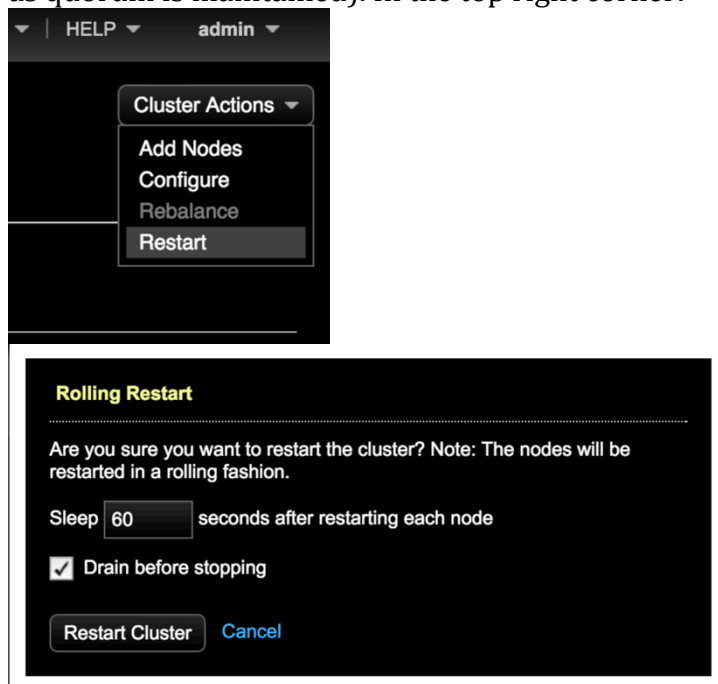
36. You should also be able to add nodes to a running cluster.

37. When you get to the second page of the Add Nodes wizard, you need to scroll down to change the security group and add your key:

38. It didn't actually work for me, but I'm hoping it will for you ☺

39. Recap: we have set up a cluster of Cassandra servers in the EC2 cloud and performance tested them, as well as doing various administration aspects to them.

40. Before you finish the lab, you must **terminate** all 5 running instances that this lab has created. These m3.large servers are quite expensive to leave running!

41. Congratulations.