CS300 Couchbase NoSQL Server Administration

Lab 1 Exercise Manual

Release: 6.5.1

Revised: June 22nd, 2020

Lab #1: Installation and overview of Couchbase Server

Objective: This 1-hour lab will introduce you to Couchbase server, specifically its installation on one node and the web UI. Future labs will go much more in depth into the administrator and configuration components of Couchbase, so consider the coverage in this lab only your initial glance at Couchbase.

Please keep in mind that the major objective of these labs is not to build out a production-level cluster, but rather a prototype lab environment where various concepts and features can be explained and demonstrated.

For example, in these labs we simply turn off the Linux firewall and don't follow the best practice to store the Couchbase data files and index files on separate devices/volumes. Also, note that the VMs in Amazon used for these labs are t2.medium with only 2 cores and 4GB memory.

These VMs do not give a realistic representation of the performance you can get out of a Couchbase cluster. In other words, although these small VMs may show performance of 15,000 iops per second, in a larger Amazon VM, you will see 100,000 iops per second or even 200,000 iops per second in a physical data center.

Warning: Do not copy + paste commands from this lab into your PuTTY/Terminal session. Some commands, especially commands that span multiple lines. A multi-line command will break into 2 lines when you copy it as the PDF will insert a /n character after the first line. Instead, please type each command individually into the SSH session!

If you disregard this and insist on cutting and pasting please paste to a notepad or text file editor and then cut and paste again from there (this should strip out 99% of extraneous characters.

Please send any comments or corrections in this lab or future labs to Couchbase Learning Services at cls@couchbase.com

Overview: The following high-level steps are involved in this lab:

- Installation of Couchbase 6.5.1 EE on 1-node in Amazon Web Services (AWS)
- Explore the UI: Cluster overview, cluster summary, viewing buckets, viewing server nodes, viewing data buckets, logs
- Explore cmd line options
- How to start and stop Couchbase server
- Examine the Beer sample database
- Examine the Couchbase DB storage files in the Linux file system
- Initial glance at log files for Couchbase

Couchbase Server Installation Introduction:

Reference Documentation

Below are some links to explore on your own time to learn more about Couchbase Server's installation and administration. The main, critical parts of these guides have been distilled into the abbreviated labs we will do in this class. However, for a deep dive into managing Couchbase, you must spend some time with these documents.

Here is a link to all of the available official Couchbase documentation in HTML format:
http://www.couchbase.com/documentation

The official Couchbase Server 6.X admin guide (note that you can choose topics from the blue dropdown in the top left that says "Couchbase Server"):

https://docs.couchbase.com/server/6.0/introduction/whats-new.html

Couchbase's YouTube channel has many excellent videos from recent conferences and webinars:
https://www.youtube.com/channel/UCGUDXCRwJi-fuQp7sJyIZmg

About a dozen technical white papers on Couchbase can be found here (We recommend starting with "Couchbase Server Under the Hood: An Architectural Overview"):
http://www.couchbase.com/nosql-resources/nosql-whitepapers

Couchbase 101 – 105 webinar training series:
http://www.couchbase.com/nosql-resources/webinar

Couchbase presentations and slides from the Couchbase community:
http://www.couchbase.com/nosql-resources/presentations

You can follow the latest technical developments in Couchbase the official blog:
http://blog.couchbase.com

Where to get help

The Couchbase communities' website allows you to post technical questions:
http://www.couchbase.com/open-source

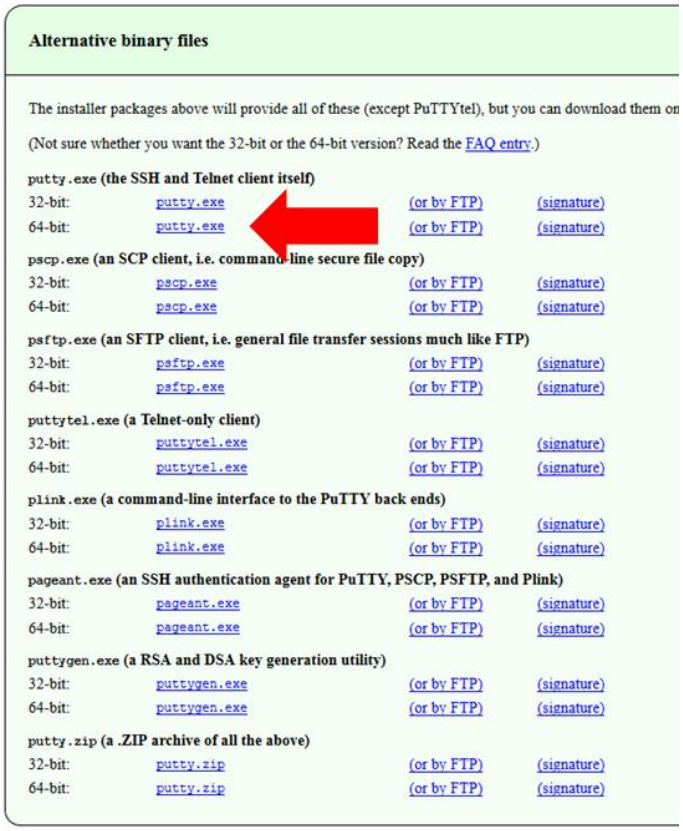
Installing PuTTY & Connecting to the 1st VM (Windows Only):

If you are on a Mac, skip this step and go to the next bold, blue heading. You can use the built in Terminal or iTerm2 (a more feature-rich replacement for Terminal) to connect/SSH to the Amazon VM, so you don't need PuTTY (which is a Windows-only app).

If you're on Windows, it is highly recommended to install PuTTY, a free telnet/SSH client. With PuTTY, you can connect to the Amazon VM from a lightweight client in Windows and open multiple cmd-line sessions to the same VM.

Download PuTTY from:
https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html

Look for the file named putty.exe under "Windows on Intel x86":



Alternative binary files

The installer packages above will provide all of these (except PuTTYtel), but you can download them on

(Not sure whether you want the 32-bit or the 64-bit version? Read the FAQ entry.)

putty.exe (the SSH and Telnet client itself)

32-bit:	putty.exe	(or by FTP)	(signature)
64-bit:	putty.exe	(or by FTP)	(signature)

pscp.exe (an SCP client, i.e. command-line secure file copy)

32-bit:	pscp.exe	(or by FTP)	(signature)
64-bit:	pscp.exe	(or by FTP)	(signature)

psftp.exe (an SFTP client, i.e. general file transfer sessions much like FTP)

32-bit:	psftp.exe	(or by FTP)	(signature)
64-bit:	psftp.exe	(or by FTP)	(signature)

puttytel.exe (a Telnet-only client)

32-bit:	puttytel.exe	(or by FTP)	(signature)
64-bit:	puttytel.exe	(or by FTP)	(signature)

plink.exe (a command-line interface to the PuTTY back ends)

32-bit:	plink.exe	(or by FTP)	(signature)
64-bit:	plink.exe	(or by FTP)	(signature)

pageant.exe (an SSH authentication agent for PuTTY, PSCP, PSFTP, and Plink)

32-bit:	pageant.exe	(or by FTP)	(signature)
64-bit:	pageant.exe	(or by FTP)	(signature)

puttygen.exe (a RSA and DSA key generation utility)

32-bit:	puttygen.exe	(or by FTP)	(signature)
64-bit:	puttygen.exe	(or by FTP)	(signature)

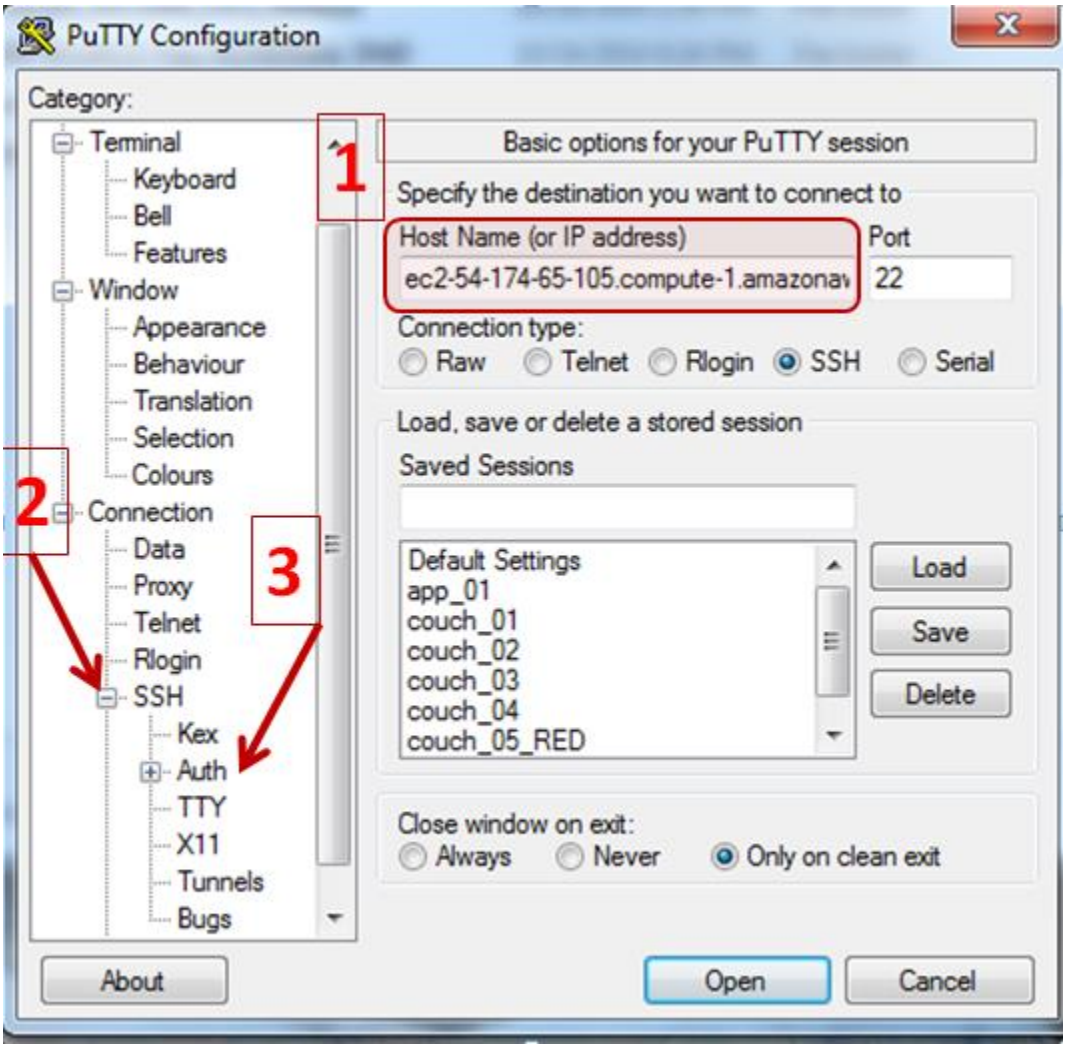
putty.zip (a .ZIP archive of all the above)

32-bit:	putty.zip	(or by FTP)	(signature)
64-bit:	putty.zip	(or by FTP)	(signature)

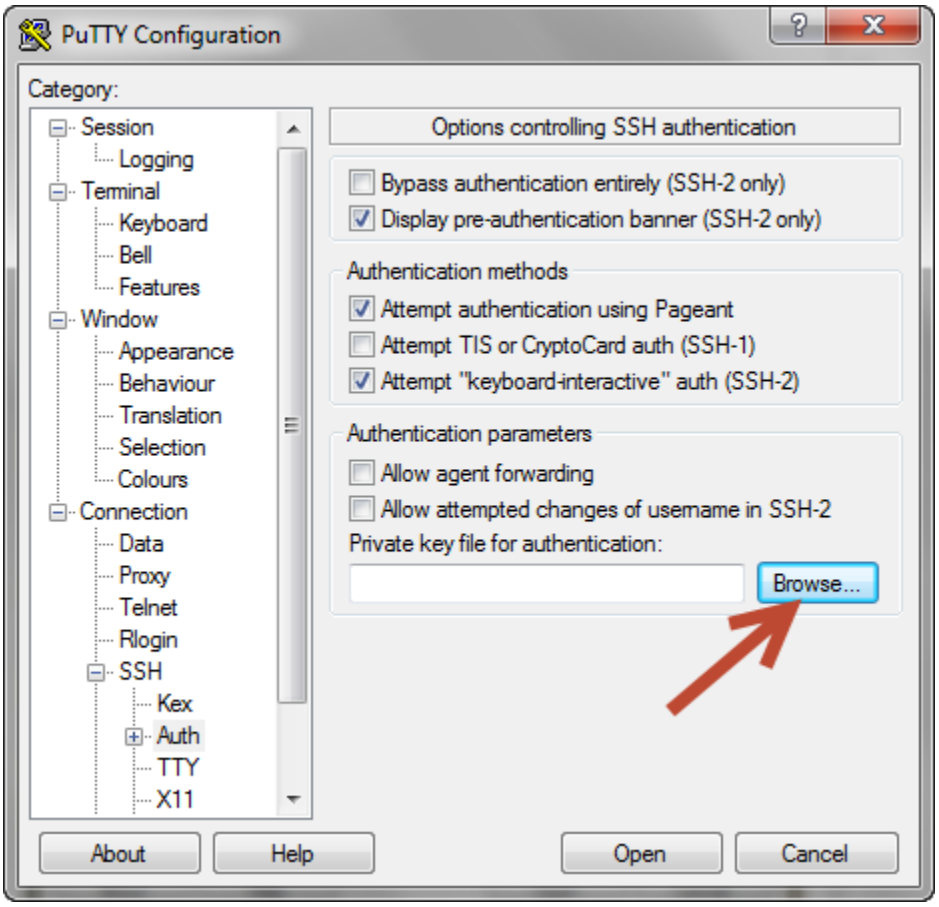
There is no installation for PuTTY. You can just run it from the downloaded `.exe` file.

After starting PuTTY, enter the public IP address of the first Amazon VM into PuTTY. You can get this IP from the `Cluster-IPs` spreadsheet that the instructor gave you along with this lab. The connection type will be SSH and the port will be 22.

Type the public hostname that the instructor gave you for your first Amazon VM into PuTTY and then **click on the + next to SSH** to expand its options and finally **select Auth**:

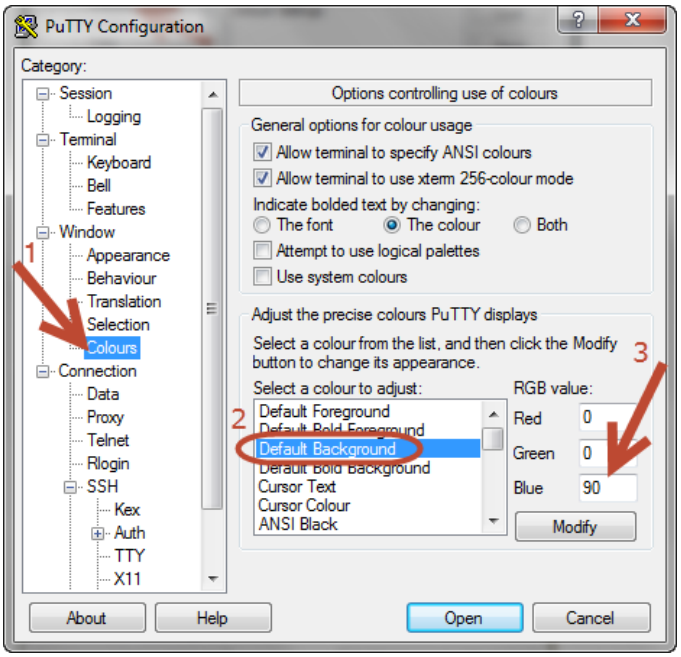


Click Browse to select the Private key file for authentication:

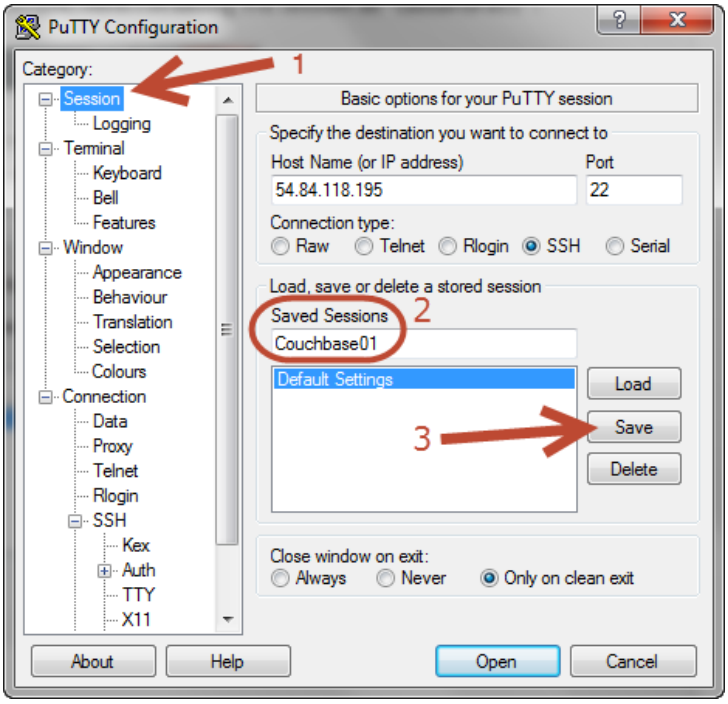


Choose the **"Amazon-Private-Key2.ppk"** file that the instructor provided you with.

In the left pane, **click on Colors**, then under "Select a Colours to adjust" **choose Default Background** and alter the **Blue RGB value to 90**.

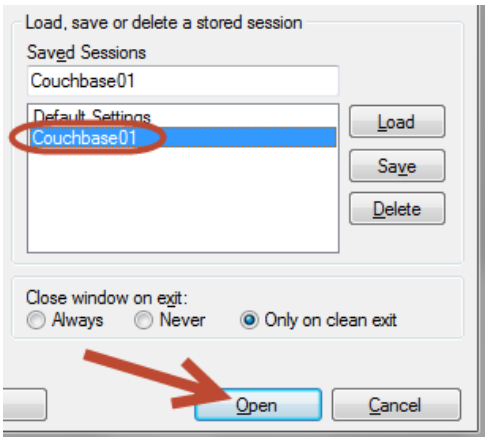


Next, **click on Session** and type to **save the session as "CouchbaseXX"**, where XX is the # of your node from the hostname. Then **click on Save**. For example, here the session is being saved as "Couchbase01":

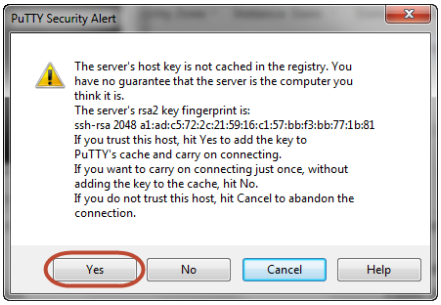


Now highlight **Couchbase01** and click **Open** to connect to this VM:

Couchbase

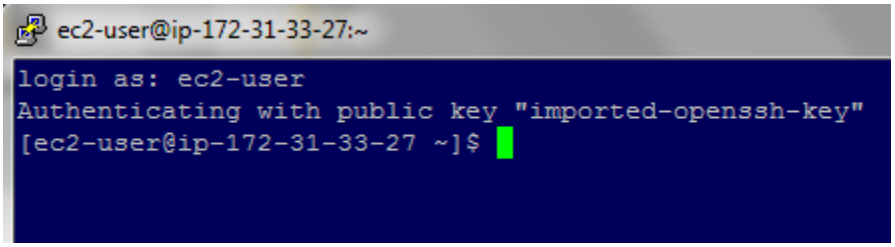


You will have to click **"Yes"** to a message about the server's rsa2 key before a successful connection.



The username for your login is:

Login as: **`ec2-user`**



Connecting to the 1st VM via Terminal/iTerm2 (Mac only):

The general instructions to log in via Mac Terminal are:

Open up your terminal app of choice and type in the following…

Change the permissions of the .pem key file like this:

```
chmod 400 Amazon-Private-Key2.pem
```

SSH into the VM using this command:

```
ssh -i Amazon-Private-Key2.pem ec2-user@<public hostname of 1st VM>
```

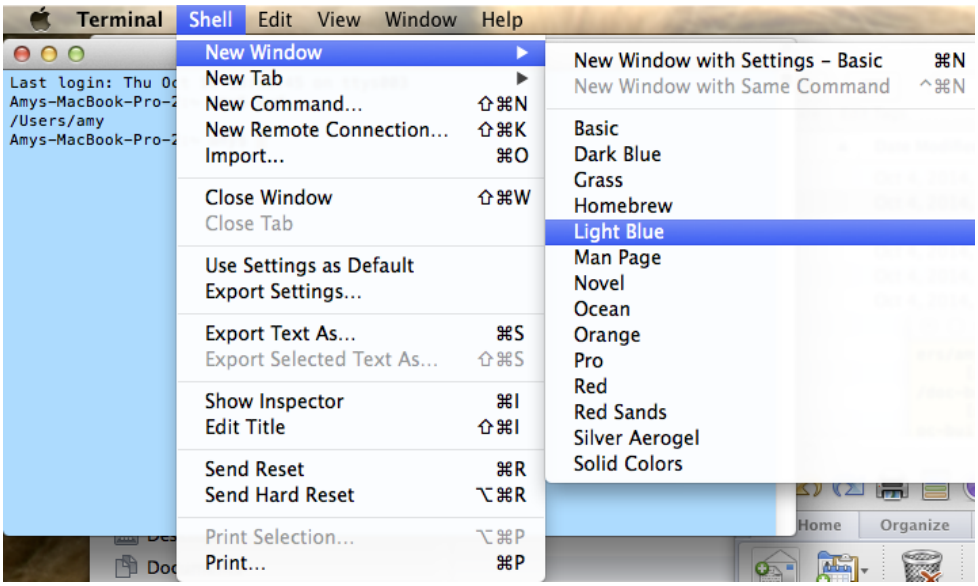
Say Yes to this prompt:

```
The authenticity of host 'ec2-198-51-100-x.compute-1.amazonaws.com
(10.254.142.33)'
can't be established.
RSA key fingerprint is
1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
Are you sure you want to continue connecting (yes/no)? yes
```

Here are the official details on how to log in via Mac:

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/

Setting screen colors can also be accomplished on a Mac.
Here is the screen shot of affecting color change on a Macbook.



Please ask the instructor for any further help needed with logging in from an Apple laptop!

Exploring the 1st Amazon server:

Here are the specs for the Couchbase server VM that you just launched:

Amazon AMI:

Red Hat Enterprise Linux 7.6 (HVM) - ami-18726478 (64-bit)
Root device type: ebs
Virtualization type: paravirtual
Amazon Instance Type: **t2.medium**
ECUs: **3** vCPU: **2**
Memory: **4.0 GiB**
Storage: **20GB magnetic** *(Note, SSDs are available, but the labs will use magnetic storage)*
Network performance: **moderate**
CloudWatch Monitoring: **disabled**
Tenancy: **Shared tenancy** (multi-tenant hardware)
Cost: **$0.11 per hour**

Note that the above specs are not enough for a production-worthy Couchbase install! For production, you should have 4-6 CPU cores and at least 16 GB of RAM, but your VM's specs should be enough for a prototype lab environment.
We choose to run Red Hat Linux for these labs because RHEL is aimed at enterprise-level servers, which means that it is stable and handles heavy loads well. RHEL is also one of the supported OS's for Couchbase 6.X Enterprise Edition.

Here is a link to the supported OS platforms for Couchbase Server:
https://docs.couchbase.com/server/6.0/install/install-platforms.html

Go to the PuTTY or Terminal window and…

Check the hostname of your machine:

```
[ec2-user@ip-172-31-20-35 ~]$ hostname
ip-172-31-20-35.us-west-1.compute.internal
```

Note: this hostname is for inside Amazon resolution(with inside name server from amazon) all access for this class will use outside ec2-w-x-y-z-.amazon.com names.

Sudo to root and change the hostname to Couchbase01

```
[ec2-user@ip-172-31-20-35 ~]$ sudo -i
[root@ip-172-31-20-35 ~]# hostnamectl set-hostname Couchbase01
[root@ip-172-31-20-35 ~]# hostnamectl status
  Static hostname: Couchbase01
        Icon name: computer-vm
          Chassis: vm
       Machine ID: 80efbea85b654c408ee6bdf762386b7c
          Boot ID: 8732f73604214f2dab2bc0d4be8738fb
   Virtualization: xen
 Operating System: Red Hat Enterprise Linux 8.0 (Ootpa)
      CPE OS Name: cpe:/o:redhat:enterprise_linux:8.0:GA
           Kernel: Linux 4.18.0-80.4.2.el8_0.x86_64
     Architecture: x86-64
[root@ip-172-31-20-35 ~]# exit
logout
```

[ec2-user@ip-172-31-20-35 ~]$

Now close the putty window and reopen a new one to verify that the hostname has been changed

First verify that this server has ~3.5GB of RAM and only 90-120 MB or so is currently being used. (*this could vary in your environment depending on how long the VM has been running for*):

```
[ec2-user@Couchbase01 ~]$ free -mh
              total        used        free      shared  buff/cache   available
Mem:           3.7Gi       168Mi       3.2Gi        48Mi       365Mi       3.3Gi
Swap:            0B          0B          0B
```

```
[ec2-user@Couchbase01~]$ sudo fdisk -l

Disk /dev/xvda: 20 GiB, 21474836480 bytes, 41943040 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xe6e324f2

Device     Boot Start      End  Sectors Size Id Type
/dev/xvda1       2048     4095     2048   1M 83 Linux
/dev/xvda2 *     4096 41943006 41938911  20G 83 Linux
```

Verify that a ~20.0 GB data disk is showing up on the server:

Check what type of file system is carved on the VM:

```
[ec2-user@Couchbase01 ~]$ df -Th

Filesystem     Type      Size  Used Avail Use% Mounted on
devtmpfs       devtmpfs  1.9G     0  1.9G   0% /dev
tmpfs          tmpfs     1.9G     0  1.9G   0% /dev/shm
tmpfs          tmpfs     1.9G   17M  1.9G   1% /run
tmpfs          tmpfs     1.9G     0  1.9G   0% /sys/fs/cgroup
/dev/xvda2     xfs        20G  1.2G   19G   6% /
tmpfs          tmpfs     378M     0  378M   0% /run/user/1000
```

Notice that the main file system is /dev/xvda2, which is of type xfs and size 20 GB with 1.2 GB used .

We will place both the Couchbase data files and the index files on this single disk. *However, in a production setup, it is recommended to configure 3 separate volumes on multiple disks, one for the Linux OS, one diskgroup per bucket for the data files and one diskgroup per index for the index files. For cost/time constraints, we will leave all 3 items on one volume in this lab.*

Implement Best Practices for Couchbase:

1) Disable Swappiness

Swappiness levels tell the Linux virtual memory subsystem how much it should try and swap to disk. The problem is that the system will try to swap out items in memory even when there is plenty of RAM available to the system.

Check what value your VM is set to by running:

```
[ec2-user@Couchbase01 ~]$ cat /proc/sys/vm/swappiness
30
```

The default setting of '30' is a bit aggressive. The value of 30 is a percentage; the higher the percentage, the higher the I/O cache and the faster that pages are swapped. You can gain performance by setting the swappiness value to 0. This tells the virtual memory subsystem of the OS to not swap items from RAM to disk unless it absolutely has to. A setting of 100 would have meant that programs will be swapped to disk almost immediately. If you have sized your nodes correctly, swapping should not be needed.

Turn off swapping for the running system, but first switch to root user:

```
[ec2-user@Couchbase01~]$ sudo -s
[root@Couchbase01 ec2-user ~]# echo 0 > /proc/sys/vm/swappiness
```

Then permanently make this change in the sysctl.conf file, so the change persists after a reboot(DO NOT REBOOT!!) and exit root

(*Note: all of these echo commands should be entered on ONE line, do not spread them across two lines in the CMD prompt!*):

```
[root@couchbase01 ec2-user]# echo '' >> /etc/sysctl.conf

[root@couchbase01 ec2-user]# echo '#Set swappiness to 0 to avoid
swapping' >> /etc/sysctl.conf

[root@couchbase01 ec2-user]# echo 'vm.swappiness = 0' >>
/etc/sysctl.conf
```

2) Disable Transparent Huge Pages

In a production Couchbase cluster, it is very important to disable Transparent Huge pages on each node. (remember to enter command on one line)

```
# Disable THP on a running system
[root@Couchbase01 ec2-user]#  echo never >
/sys/kernel/mm/transparent_hugepage/enabled
[root@Couchbase01 ec2-user]#  echo never >
/sys/kernel/mm/transparent_hugepage/defrag
```

Installing Couchbase:

```
[root@ Couchbase01 ec2-user]#  yum install wget
Last metadata expiration check: 0:06:55 ago on Tue 21 Jan 2020 09:47:25 PM UTC.
Dependencies resolved.
================================================================================
 Package              Arch                   Version                        Repository
Size
================================================================================
Installing:
 wget                 x86_64                 1.19.5-8.el8_1.1               rhui-rhel-8-
appstream-rhui-rpms               735 k

Transaction Summary
================================================================================
Install  1 Package

Total download size: 735 k
Installed size: 2.9 M
Is this ok [y/N]: y
Downloading Packages:
wget-1.19.5-8.el8_1.1.x86_64.rpm
7.3 MB/s | 735 kB     00:00
--------------------------------------------------------------------------------
Total
660 kB/s | 735 kB     00:01
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :
1/1
  Installing       : wget-1.19.5-8.el8_1.1.x86_64
1/1
  Running scriptlet: wget-1.19.5-8.el8_1.1.x86_64
1/1
  Verifying        : wget-1.19.5-8.el8_1.1.x86_64
1/1

Installed:
  wget-1.19.5-8.el8_1.1.x86_64

Complete!
```

```
[root@ Couchbase01 ec2-user]#  yum install bzip2

Last metadata expiration check: 0:00:16 ago on Tue 03 Sep 2019 05:50:37 PM UTC.
Dependencies resolved.

================================================================================
 Package     Arch          Version         Repository                      Size
================================================================================
Installing:
 bzip2       x86_64    1.0.6-26.el8     rhui-rhel-8-baseos-rhui-rpms       60 k

Transaction Summary
================================================================================
Install  1 Package

Total download size: 60 k
Installed size: 91 k
Is this ok [y/N]: y
Downloading Packages:
bzip2-1.0.6-26.el8.x86_64.rpm
556 kB/s |  60 kB     00:00
--------------------------------------------------------------------------------
Total
500 kB/s |  60 kB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing                                                                1/1
  Installing       : bzip2-1.0.6-26.el8.x86_64                             1/1
  Running scriptlet: bzip2-1.0.6-26.el8.x86_64                             1/1
  Verifying        : bzip2-1.0.6-26.el8.x86_64                             1/1

Installed:
  bzip2-1.0.6-26.el8.x86_64

Complete!
```

```
# yum install python3
```

```
# exit
```

Download Couchbase 6.5.1 EE (*do not copy + paste this command!*):

```
[ec2-user@ Couchbase01~]$ wget
http://packages.couchbase.com/releases/6.5.1/couchbase-server-
enterprise-6.5.1-centos8.x86_64.rpm
```

```
--2020-01-21 21:57:23--  http://packages.couchbase.com/releases/6.5.1/couchbase-server-
enterprise-6.5.1-centos8.x86_64.rpm
Resolving packages.couchbase.com (packages.couchbase.com)... 13.224.29.29, 13.224.29.70,
13.224.29.116, ...
Connecting to packages.couchbase.com (packages.couchbase.com)|13.224.29.29|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 394755748 (376M) [application/x-rpm]
Saving to: âcouchbase-server-enterprise-6.5.1-centos8.x86_64.rpmâ

couchbase-server-enterprise-6.5.
100%[=====================================================>] 376.47M  71.3MB/s    in 5.3s
```

```
2020-01-21 21:57:29 (71.4 MB/s) - âcouchbase-server-enterprise-6.5.1-centos8.x86_64.rpmâ saved
[394755748/394755748]
```

Install Couchbase (note, this command might take 1-2 minutes to complete):

```
[ec2-user@Couchbase01 ~]$ sudo rpm --install couchbase-server-enterprise-6.5.1-centos8.x86_64.rpm
Minimum RAM required  : 4 GB
System RAM configured : 3.69 GB

Minimum number of processors required : 4 cores
Number of processors on the system    : 2 cores

Created symlink /etc/systemd/system/multi-user.target.wants/couchbase-server.service â
/usr/lib/systemd/system/couchbase-server.service.

You have successfully installed Couchbase Server.
Please browse to http://Couchbase01:8091/ to configure your server.
Refer to http://docs.couchbase.com for additional resources.

Please note that you have to update your firewall configuration to
allow external connections to a number of network ports for full
operation. Refer to the documentation for the current list:
https://docs.couchbase.com/server/6.5/install/install-ports.html

By using this software you agree to the End User License Agreement.
See /opt/couchbase/LICENSE.txt.
```

Congratulations! If you see the above message, you've successfully installed Couchbase. In the next section, we'll configure and start the cluster.

To check if Couchbase is running run the following **as root**:

```
[ec2-user@Couchbase01 ~]$ sudo -i

# systemctl   status   couchbase-server

â couchbase-server.service - Couchbase Server
   Loaded: loaded (/usr/lib/systemd/system/couchbase-server.service; enabled; vendor preset:
disabled)
   Active: active (running) since Tue 2019-09-03 17:52:35 UTC; 10min ago
     Docs: http://docs.couchbase.com
 Main PID: 16228 (beam.smp)
    Tasks: 159 (limit: 23940)
   Memory: 166.0M
   CGroup: /system.slice/couchbase-server.service
           ââ16228 /opt/couchbase/lib/erlang/erts-9.3.3.9/bin/beam.smp -A 16 -- -root
/opt/couchbase/lib/erlang -progname erl -- -home /opt/cou>
           ââ16245 /opt/couchbase/lib/erlang/erts-9.3.3.9/bin/epmd -daemon
           ââ16323 erl_child_setup 70000
           ââ16341 /opt/couchbase/bin/gosecrets
           ââ16345 /opt/couchbase/lib/erlang/erts-9.3.3.9/bin/beam.smp -A 16 -sbt u -P 327680 -K
true -swt low -MMmcs 30 -e102400 -- -root /opt>
           ââ16367 erl_child_setup 70000
           ââ16385 sh -s disksup
           ââ16387 /opt/couchbase/lib/erlang/lib/os_mon-2.4.4/priv/bin/cpu_sup
           ââ16388 /opt/couchbase/lib/erlang/lib/os_mon-2.4.4/priv/bin/memsup
           ââ16410 /opt/couchbase/lib/erlang/erts-9.3.3.9/bin/beam.smp -P 327680 -K true -- -root
/opt/couchbase/lib/erlang -progname erl -- -h>
           ââ16426 erl_child_setup 70000
           ââ16444 sh -s disksup
```

```
           ââ16446 /opt/couchbase/lib/erlang/lib/os_mon-2.4.4/priv/bin/memsup
           ââ16447 /opt/couchbase/lib/erlang/lib/os_mon-2.4.4/priv/bin/cpu_sup
           ââ16450 /opt/couchbase/bin/priv/godu
           ââ16454 sh -s ns_disksup
           ââ16457 /opt/couchbase/bin/priv/godu
           ââ16464 inet_gethost 4
           ââ16465 inet_gethost 4
           ââ16466 /opt/couchbase/bin/saslauthd-port
           ââ16470 portsigar for ns_1@cb.local 16228
           ââ16471 /opt/couchbase/bin/goport -window-size=524288
           ââ16475 /opt/couchbase/bin/goxdcr -sourceKVAdminPort=8091 -xdcrRestPort=9998 -
isEnterprise=true -ipv6=false
           ââ16483 /opt/couchbase/bin/memcached -C
/opt/couchbase/var/lib/couchbase/config/memcached.json

Sep 03 17:52:37 Couchbase01 couchbase[16228]:         []
Sep 03 17:52:37 Couchbase01 couchbase[16228]: 2019-09-03 17:52:37 cb_dist: Initial protos: ~p,
required protos: ~p
Sep 03 17:52:37 Couchbase01 couchbase[16228]:         [inet_tcp_dist,inet6_tcp_dist]
Sep 03 17:52:37 Couchbase01 couchbase[16228]:         [inet_tcp_dist]
Sep 03 17:52:37 Couchbase01 couchbase[16228]: 2019-09-03 17:52:37 cb_dist: Starting ~p listener
on ~p...
Sep 03 17:52:37 Couchbase01 couchbase[16228]:         inet_tcp_dist
Sep 03 17:52:37 Couchbase01 couchbase[16228]:         21200
Sep 03 17:52:37 Couchbase01 couchbase[16228]: 2019-09-03 17:52:37 cb_dist: Starting ~p listener
on ~p...
Sep 03 17:52:37 Couchbase01 couchbase[16228]:         inet6_tcp_dist
Sep 03 17:52:37 Couchbase01 couchbase[16228]:         21200
[root@Couchbase01 ~]# systemctl status couchbase-server
â couchbase-server.service - Couchbase Server
   Loaded: loaded (/usr/lib/systemd/system/couchbase-server.service; enabled; vendor preset:
disabled)
   Active: active (running) since Tue 2019-09-03 17:52:35 UTC; 10min ago
     Docs: http://docs.couchbase.com
 Main PID: 16228 (beam.smp)
    Tasks: 159 (limit: 23940)
   Memory: 166.5M
   CGroup: /system.slice/couchbase-server.service
           ââ16228 /opt/couchbase/lib/erlang/erts-9.3.3.9/bin/beam.smp -A 16 -- -root
/opt/couchbase/lib/erlang -progname erl -- -home /opt/cou>
           ââ16245 /opt/couchbase/lib/erlang/erts-9.3.3.9/bin/epmd -daemon
           ââ16323 erl_child_setup 70000
           ââ16341 /opt/couchbase/bin/gosecrets
           ââ16345 /opt/couchbase/lib/erlang/erts-9.3.3.9/bin/beam.smp -A 16 -sbt u -P 327680 -K
true -swt low -MMmcs 30 -e102400 -- -root /opt>
           ââ16367 erl_child_setup 70000
           ââ16385 sh -s disksup
           ââ16387 /opt/couchbase/lib/erlang/lib/os_mon-2.4.4/priv/bin/cpu_sup
           ââ16388 /opt/couchbase/lib/erlang/lib/os_mon-2.4.4/priv/bin/memsup
           ââ16410 /opt/couchbase/lib/erlang/erts-9.3.3.9/bin/beam.smp -P 327680 -K true -- -root
/opt/couchbase/lib/erlang -progname erl -- -h>
           ââ16426 erl_child_setup 70000
           ââ16444 sh -s disksup
           ââ16446 /opt/couchbase/lib/erlang/lib/os_mon-2.4.4/priv/bin/memsup
           ââ16447 /opt/couchbase/lib/erlang/lib/os_mon-2.4.4/priv/bin/cpu_sup
           ââ16450 /opt/couchbase/bin/priv/godu
           ââ16454 sh -s ns_disksup
           ââ16457 /opt/couchbase/bin/priv/godu
           ââ16464 inet_gethost 4
           ââ16465 inet_gethost 4
           ââ16466 /opt/couchbase/bin/saslauthd-port
           ââ16470 portsigar for ns_1@cb.local 16228
           ââ16471 /opt/couchbase/bin/goport -window-size=524288
           ââ16475 /opt/couchbase/bin/goxdcr -sourceKVAdminPort=8091 -xdcrRestPort=9998 -
isEnterprise=true -ipv6=false
           ââ16483 /opt/couchbase/bin/memcached -C
/opt/couchbase/var/lib/couchbase/config/memcached.json

Sep 03 17:52:37 Couchbase01 couchbase[16228]:         []
```

Couchbase


```
Sep 03 17:52:37 Couchbase01 couchbase[16228]: 2019-09-03 17:52:37 cb_dist: Initial protos: ~p,
required protos: ~p
Sep 03 17:52:37 Couchbase01 couchbase[16228]:         [inet_tcp_dist,inet6_tcp_dist]
Sep 03 17:52:37 Couchbase01 couchbase[16228]:         [inet_tcp_dist]
Sep 03 17:52:37 Couchbase01 couchbase[16228]: 2019-09-03 17:52:37 cb_dist: Starting ~p listener
on ~p...
Sep 03 17:52:37 Couchbase01 couchbase[16228]:         inet_tcp_dist
Sep 03 17:52:37 Couchbase01 couchbase[16228]:         21200
Sep 03 17:52:37 Couchbase01 couchbase[16228]: 2019-09-03 17:52:37 cb_dist: Starting ~p listener
on ~p...
Sep 03 17:52:37 Couchbase01 couchbase[16228]:         inet6_tcp_dist
Sep 03 17:52:37 Couchbase01 couchbase[16228]:         21200
lines 6-43/43 (END)
```

Type in **q** (for quit)

Then exit

exit

Note: stopping ,starting Couchbase is done via the following commands, this does not need to be done now it is just for future reference.

 # systemctl start couchbase-server
Starting couchbase-server

 # systemctl stop couchbase-server
Stopping couchbase-server

 # systemctl status couchbase-server
Obtaining system status

Configure a 1-node Couchbase cluster:

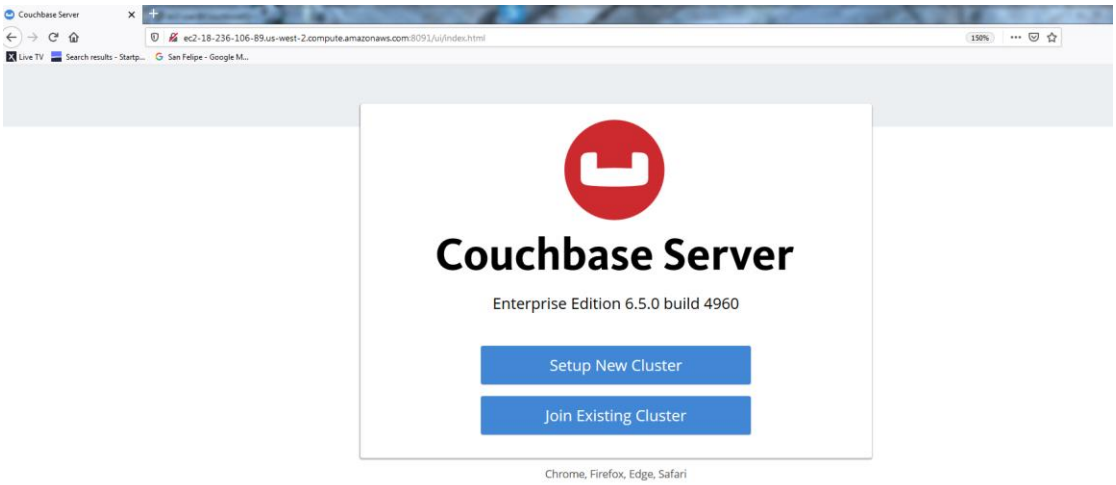
Note that the URL provided in the output from the install will not work directly. You have to replace the AWS hostname/Couchbase01 from the installation output above (for example: ip-172-31-33-xx) with the public hostname of the VM from the excel spreadsheet

(like: ec2-54-88-123-x.compute-1.amazonaws.com).

Open a Chrome or Firefox browser and go to the following URL:

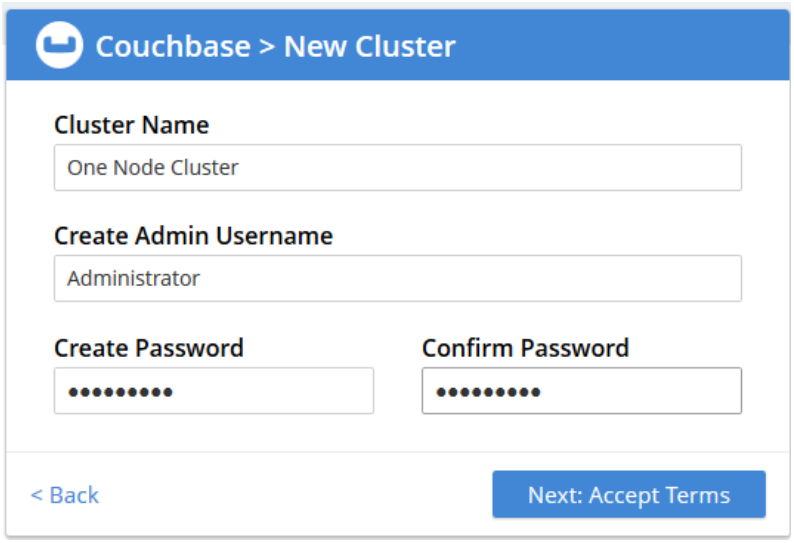
http://<public hostname of your VM>:8091

Couchbase



Click on the Setup New Cluster button in the bottom area to continue.
Edit in Cluster Name One Node Cluster
Password for Administrator is couchbase

Please do not deviate from this login and password. It makes troubleshooting student challenges much longer and cannot be done after class if the instructor does not know YOUR password!!

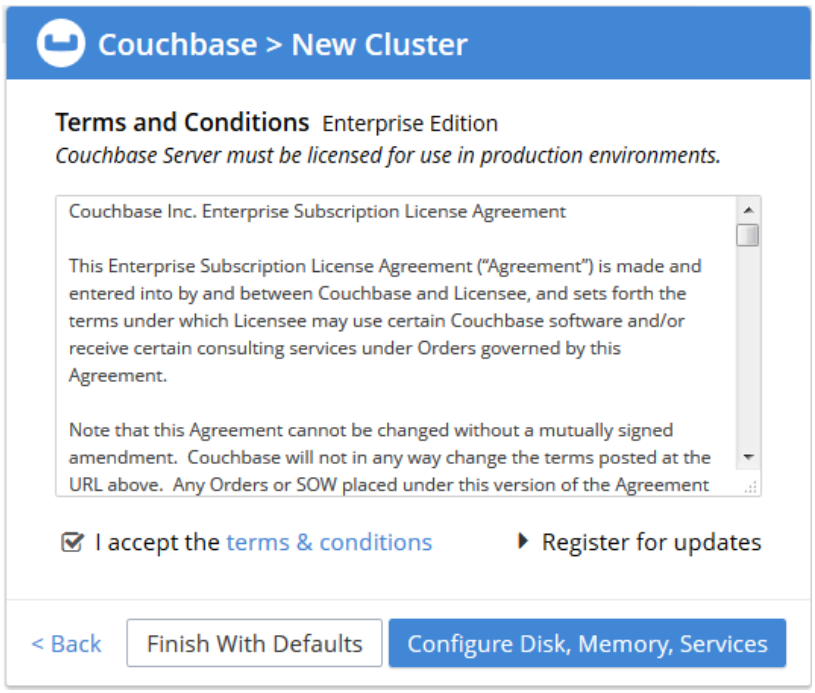


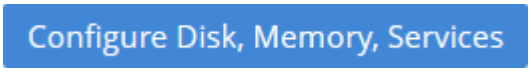
Do Not Change the Administrator name or password from these instructions!

Click Next: Accept Terms

Check the accept terms & conditions box

Couchbase



Click 

Some of the settings on the "Couchbase>New Cluster>Configure " page will need to be altered. Specifically, the items in red need to be changed

Hostname: <Public hostname of VM, retrieve this from the Cluster-IPs spreadsheet>
Or from the browser URL entry line. i.e. <**ec2-54-174-65-105.compute-1.amazonaws.com**>

Check the box to **enable cluster encryption**

Data Disk Path: **/opt/couchbase/var/lib/couchbase/data**
Indexes Disk Path: **/opt/couchbase/var/lib/couchbase/indexes**
Eventing Disk Path: **/opt/couchbase/var/lib/couchbase/eventing**
Analytics Disk Path: **/opt/couchbase/var/lib/couchbase/analytics**

Leave Java Runtime path blank

Services select boxes: Data, Index, and Query

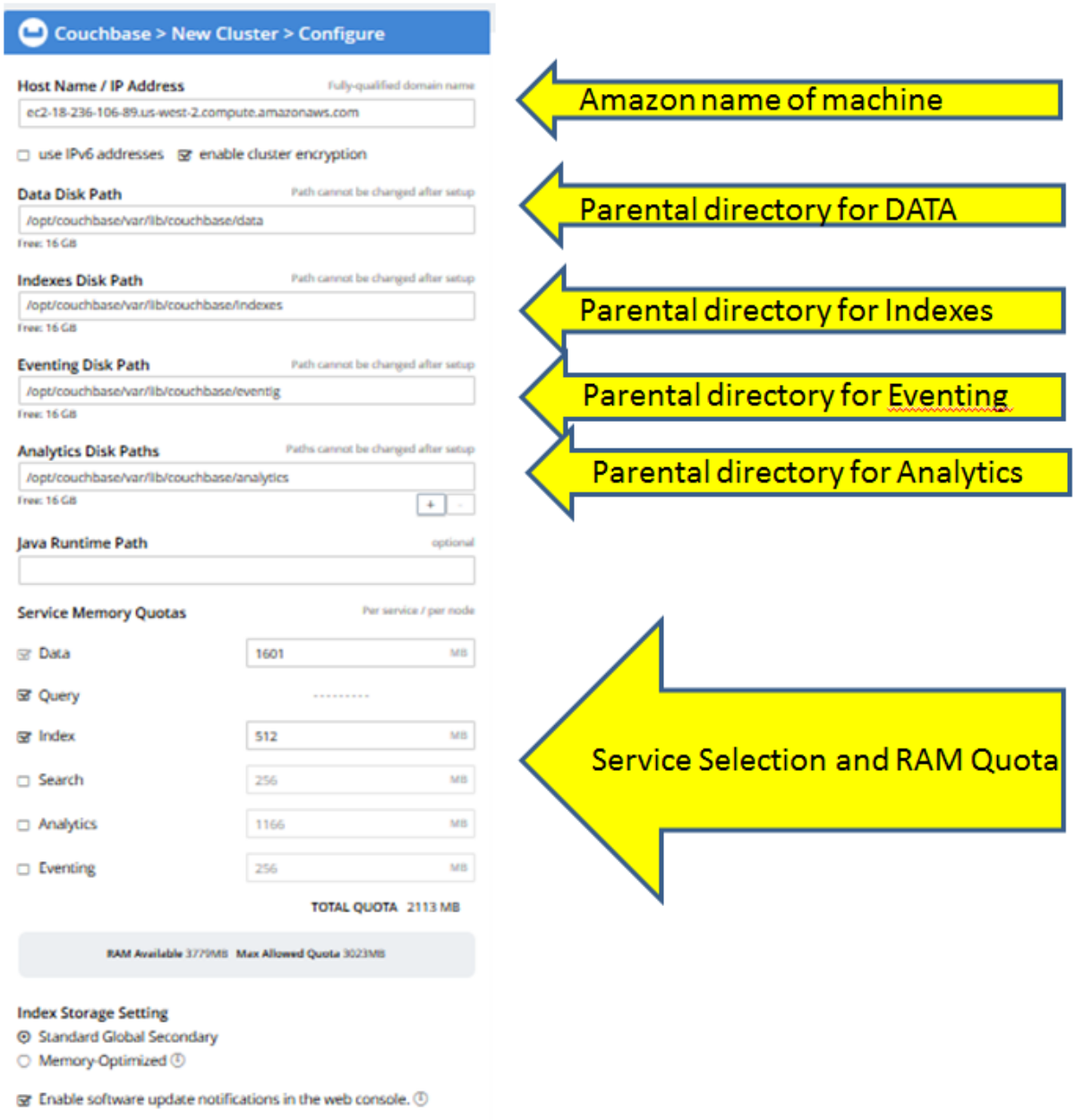
Deselect ALL other service boxes

Per Server DATA RAM Quota: 1601 MB
Per Server Index RAM Quota: 512 MB *(Min ram value shown to right)*

Accept the default calculation if your memory value is different than shown or you are configuring a dedicated INDEX service with more RAM available.

Per Server Search, Analytics & Eventing RAM Quotas:
DO NOT SELECT SERVICE *(Min ram value shown to right)*

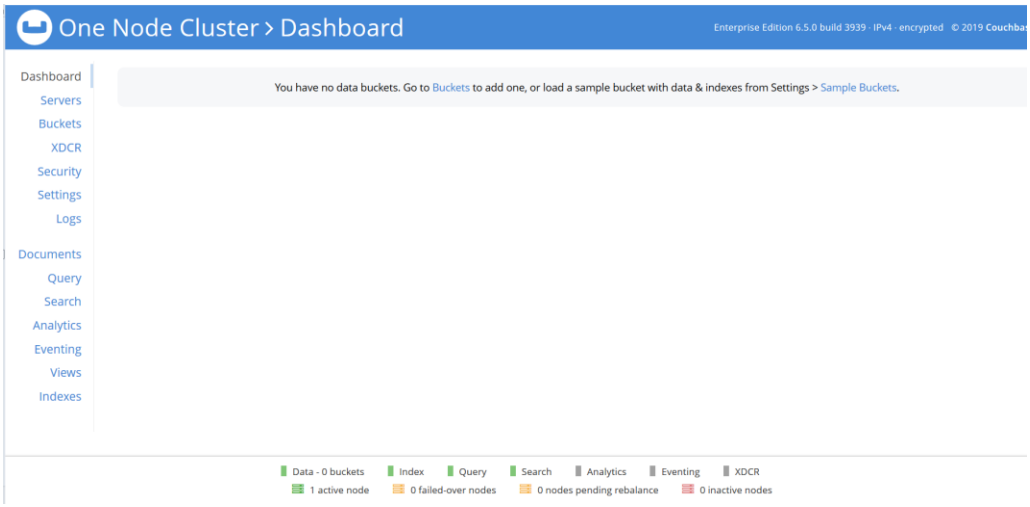
Leave the Standard Global Secondary **Indexes radio button selected**



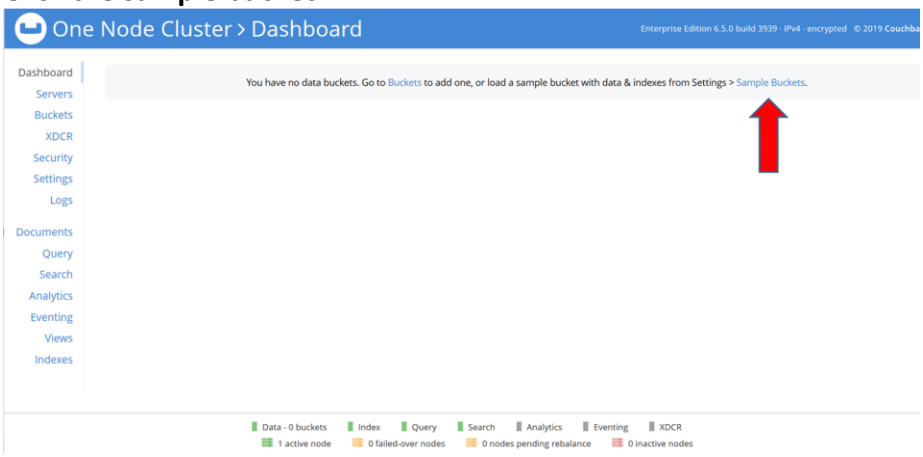
Leave box selected for software update notification

Click Save & Finish

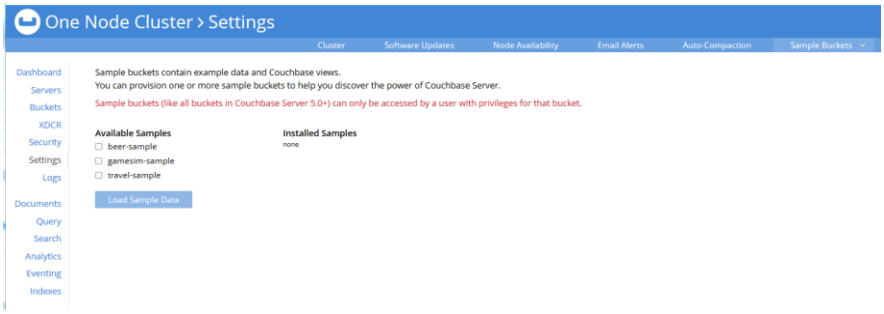
This will bring you to the Cluster Name> Dashboard :



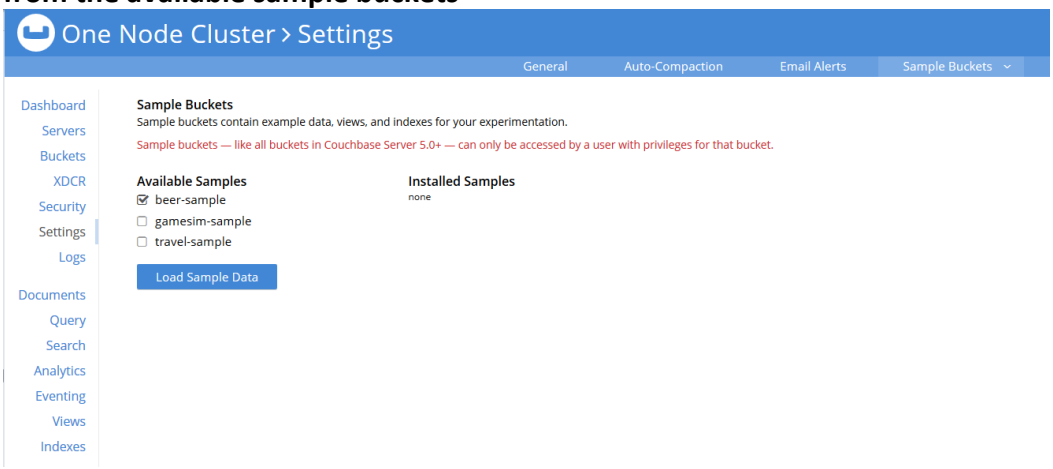
Click the sample bucket link



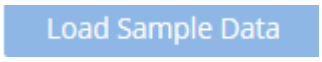
This will take you to the Cluster Name > Settings page



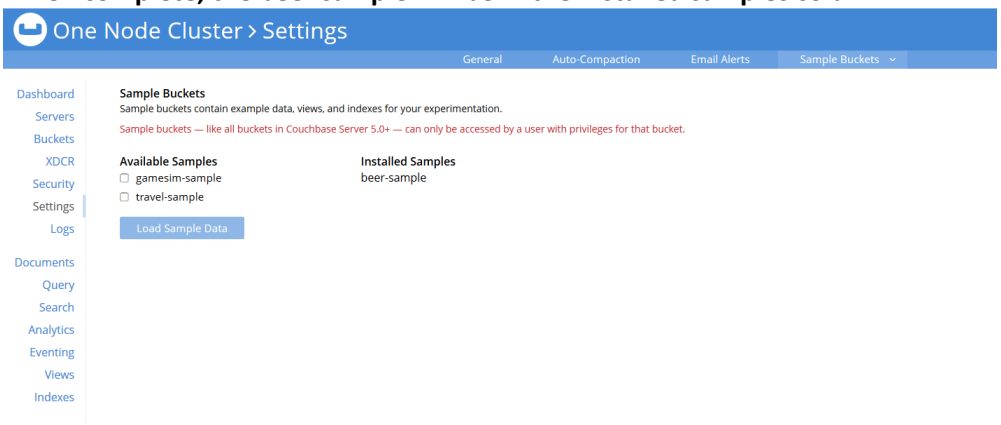
Select the beer-sample checkbox from the available sample buckets



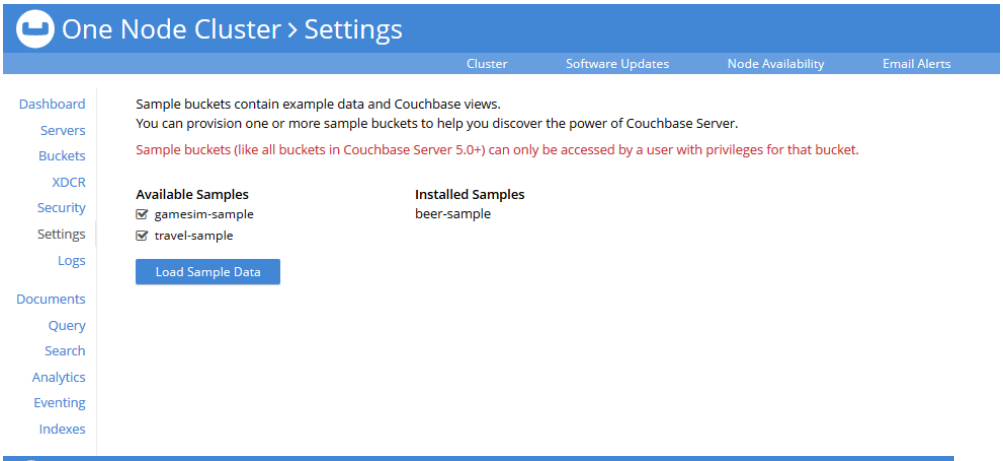
and click

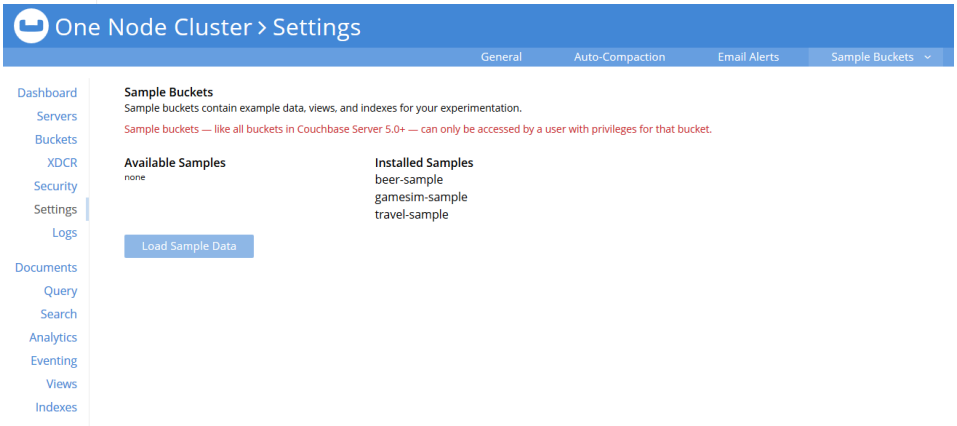


When complete, the beer sample will be in the Installed samples column

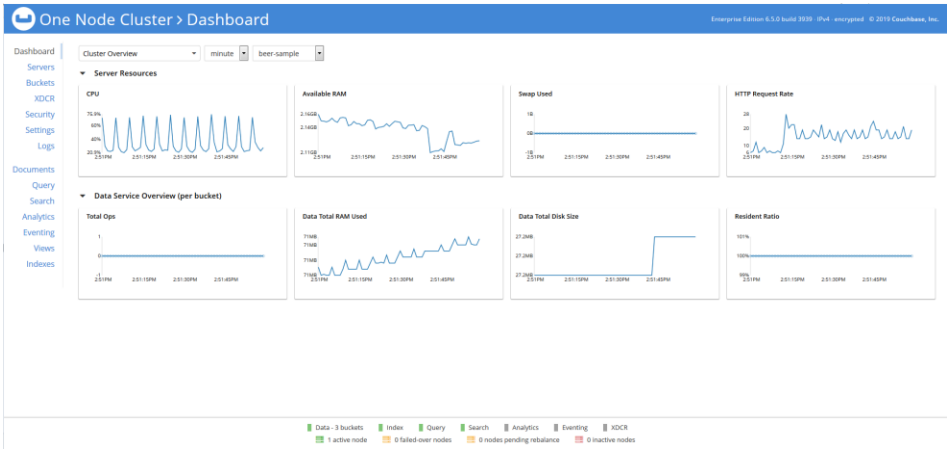


Now repeat the steps necessary to load gamesim-sample and travel-sample



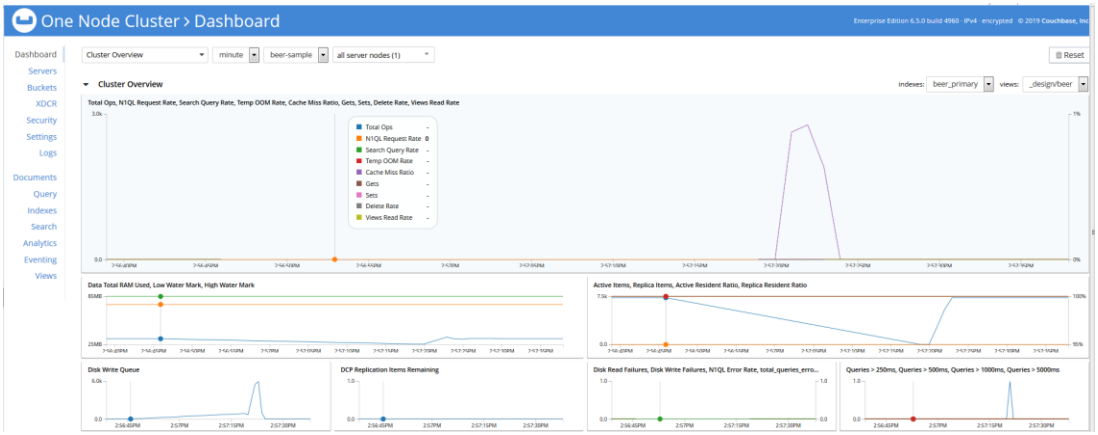


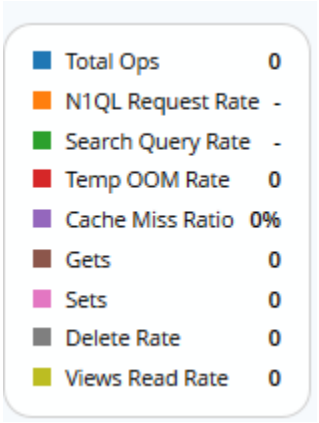
Now, go to the Cluster Name > Dashboard: Select Cluster overview from the pulldown
You will start to see some available RAM and Disk usage movement(use) occurring as the sample databases are loaded:



Exploring the Couchbase Web UI:

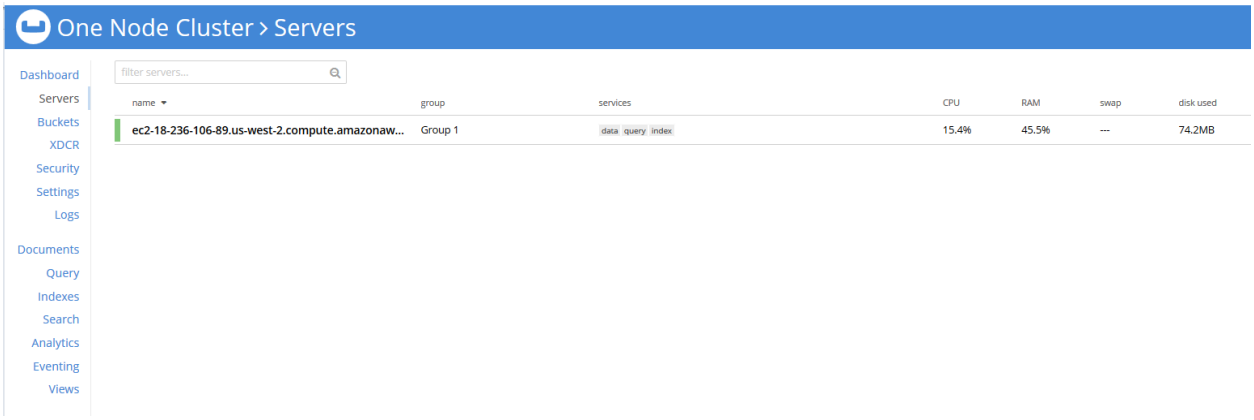
Now that Couchbase is installed, let's do a quick tour of the Couchbase Web User Interface. On the main dashboard, focus your mouse cursor in the cluster overview area.

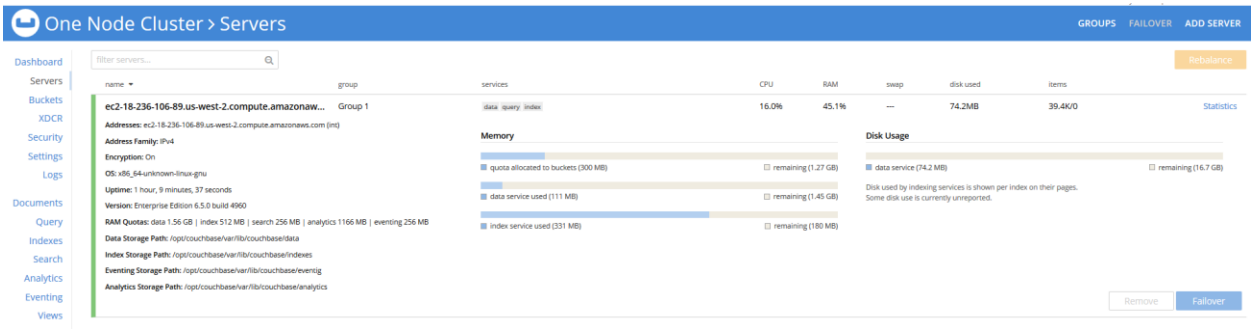




Colored popup gives active statistics for a given bucket.

Click on "Servers" link at the left hand side of the UI and then click on the Green Bar next to the Server Node Name.





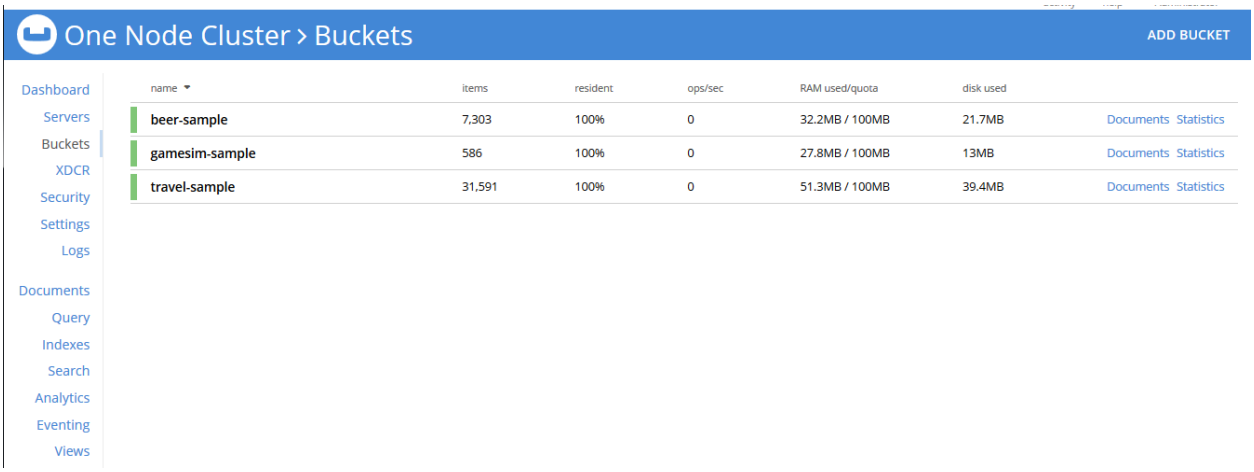
Notice a few things there. The RAM usage on this single node is about 30-50% and the CPU usage is 10-16%. There is an "Add Server" button at the top right that we'll explore in a future lab to grow the cluster.

On the far right, you can also see that there are a 39.4k active items on the server and 0 replica items.

Click on "Buckets" link on the side menu:

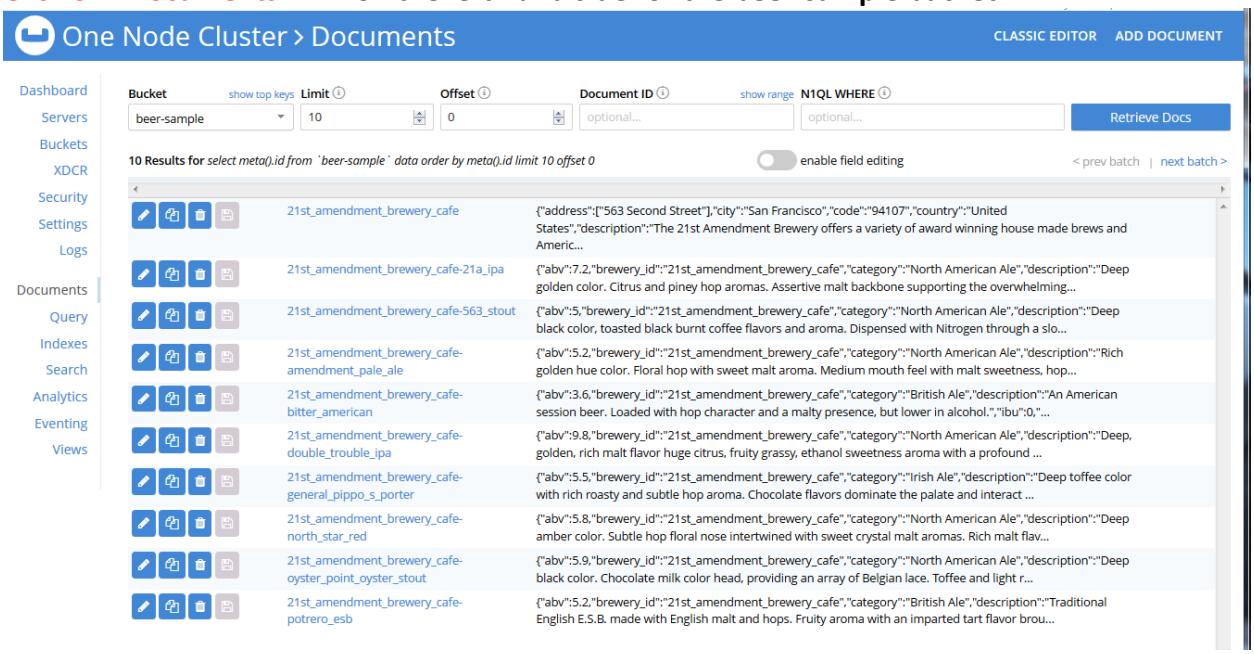
Couchbase

Lab-1: Installation of Couchbase Server, page 26



Here you will find 3 buckets. The Three are sample buckets

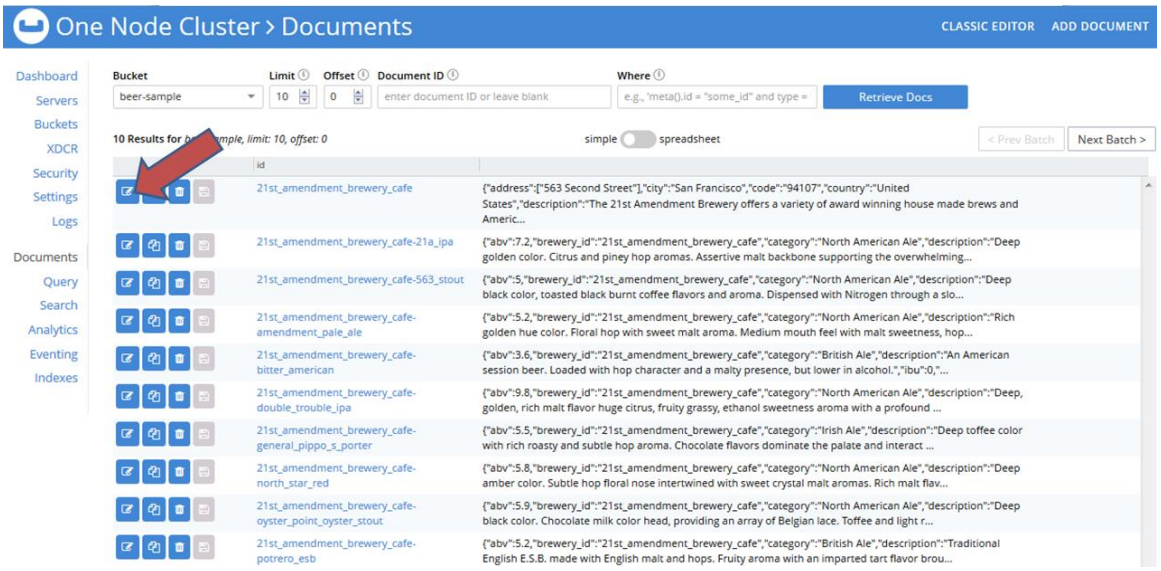
Click on "Documents" link on the left hand side for the beer-sample bucket.



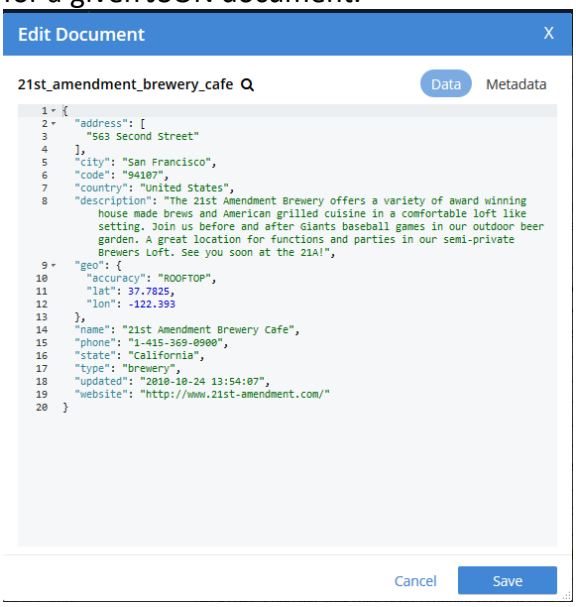
You will see the first 10 documents in this sample database displayed.

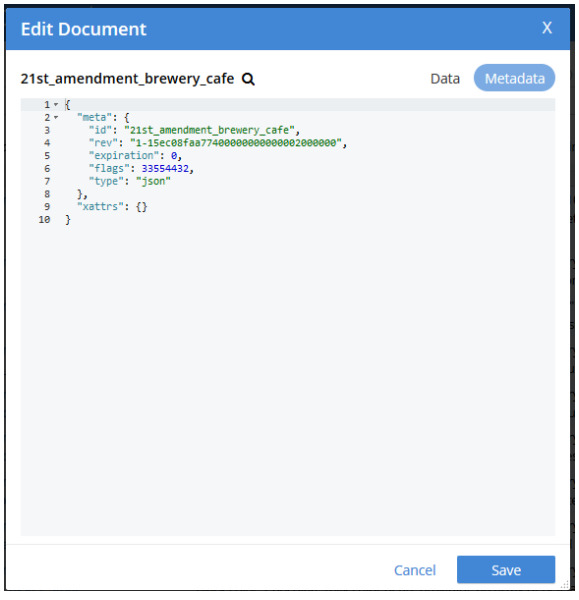
Couchbase

Click on the first "Edit" button on the left hand side to see the full first document.



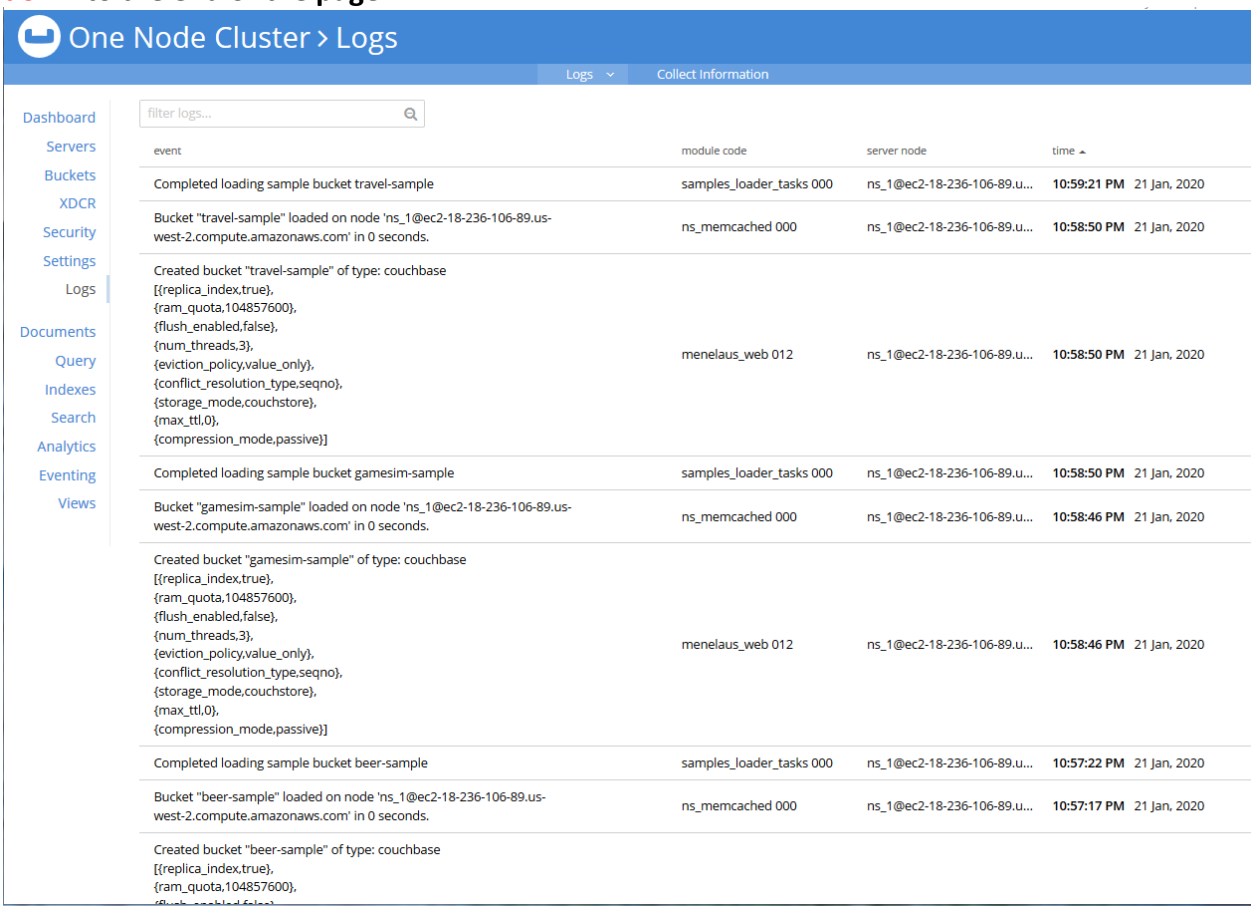
The raw JSON document editor displays DATA about the first brewery (like address, phone #, website, etc.) listed in the database along with a METADATA button to show and edit metadata for a given JSON document.





Couchbase

Next, we'll explore the Event Log for Couchbase. **At the side links bar, click on "Logs" and scroll down to the end of the page:**



After scrolling to the end of the page, you can see some of the first log messages generated by this Couchbase server as it started up. Notice the 2nd message: "I'm the only node, so I'm the master." This essentially means that Node #1 is the Couchbase Orchestrator. The orchestrator node in Couchbase manages the rebalance process.

This examines the current vBucket map and then combines that information with the node additions and removals in order to create a new vBucket map.

The orchestrator starts the process of moving the individual vBuckets from the current vBucket map to the new vBucket structure. The process is only started by the orchestrator - the nodes themselves are responsible for actually performing the movement of data between the nodes. The aim is to make the newly calculated vBucket map match the current situation.

Completed loading sample bucket beer-sample	samples_loader_tasks 000	ns_1@ec2-54-183-85-83.us-west-1...	**4:27:22 PM**	Tue Sep 26, 2017
Bucket "beer-sample" loaded on node 'ns_1@ec2-54-183-85-83.us-west-1.compute.amazonaws.com' in 0 seconds.	ns_memcached 000	ns_1@ec2-54-183-85-83.us-west-1...	**4:27:19 PM**	Tue Sep 26, 2017
Created bucket "beer-sample" of type: couchbase [{num_replicas,1}, {replica_index,false}, {ram_quota,104857600}, {flush_enabled,false}, {num_threads,3}, {eviction_policy,value_only}, {conflict_resolution_type,seqno}, {storage_mode,couchstore}]	menelaus_web 012	ns_1@ec2-54-183-85-83.us-west-1...	**4:27:19 PM**	Tue Sep 26, 2017
I'm the only node, so I'm the master. (repeated 1 times)	mb_master 000	ns_1@ec2-54-183-85-83.us-west-1...	**4:10:05 PM**	Tue Sep 26, 2017
Renamed node. New name is 'ns_1@ec2-54-183-85-83.us-west-1.compute.amazonaws.com'.	ns_cluster 000	ns_1@ec2-54-183-85-83.us-west-1...	**4:09:06 PM**	Tue Sep 26, 2017
Change of address to "ec2-54-183-85-83.us-west-1.compute.amazonaws.com" is requested.	ns_cluster 000	ns_1@127.0.0.1	**4:09:06 PM**	Tue Sep 26, 2017
Changing address to "ec2-54-183-85-83.us-west-1.compute.amazonaws.com" due to client request	ns_cluster 000	ns_1@127.0.0.1	**4:09:06 PM**	Tue Sep 26, 2017
I'm the only node, so I'm the master.	mb_master 000	ns_1@127.0.0.1	**4:09:05 PM**	Tue Sep 26, 2017
Couchbase Server has started on web port 8091 on node 'ns_1@127.0.0.1'. Version: "5.0.0-3519-enterprise".	menelaus_sup 001	ns_1@127.0.0.1	**4:09:05 PM**	Tue Sep 26, 2017
Setting database directory path to /opt/couchbase/var/lib/couchbase/data and index directory path to /opt/couchbase/var/lib/couchbase/index	ns_storage_conf 000	ns_1@127.0.0.1	**4:09:03 PM**	Tue Sep 26, 2017
Hot-reloaded memcached.json for config change of the following keys: [<<"xattr_enabled">>]	memcached_config_mgr 000	ns_1@127.0.0.1	**6:09:50 PM**	Mon Sep 25, 2017
Changed cluster compat mode from undefined to [5,0]	ns_orchestrator 000	ns_1@127.0.0.1	**6:09:50 PM**	Mon Sep 25, 2017
I'm the only node, so I'm the master.	mb_master 000	ns_1@127.0.0.1	**6:09:50 PM**	Mon Sep 25, 2017
Couchbase Server has started on web port 8091 on node 'ns_1@127.0.0.1'. Version: "5.0.0-3519-enterprise".	menelaus_sup 001	ns_1@127.0.0.1	**6:09:49 PM**	Mon Sep 25, 2017
Initial otp cookie generated: {sanitized, <<"D9uTX8dbV3Ehq1HBdbTRWwBlhZ77o12+B+T7lXF295Q=">>}	ns_cookie_manager 003	ns_1@127.0.0.1	**6:09:49 PM**	Mon Sep 25, 2017

Exploring Couchbase cmd-line interface:

The couchbase-cli tool provides various management operations for Couchbase clusters, nodes and buckets.

The tool is located in the following directory location on linux:
/opt/couchbase/bin/couchbase-cli

Add the /opt/couchbase/bin directory to your Linux PATH so that you can run the couchbase-cli tool (and other tools) by simply typing 'couchbase-cli' without providing the full directory path into the command.

Note, you will need to use a linux text editor for the following section. You can use either nano, vi, vim or emacs to open the XML file and all future files. If you are unfamiliar with the arcane vi/vim or emacs syntax, I recommend using nano, one of the simplest text editors to use on Linux. My preference is vi & vim, so you will see me opening all files with these 2 editors for the rest of the labs, but feel free to replace the word 'vi' or 'vim' with 'nano' or 'emacs' on your end.
If you want a 3 min crash course in vi/vim, go to this link and graduate levels 1 and 2 and then come back: http://yannesposito.com/Scratch/en/blog/Learn-Vim-Progressively/

Edit the .bashrc file:

```
[ec2-user@Couchbase01 ~]$ cd ~
[ec2-user@Couchbase01 ~]$ vi .bashrc
```

Line 9 should currently show the following:

```
PATH=$PATH:$HOME/.local/bin:$HOME/bin
```

Edit line 9 by appending the couchbase tools path to the end of the line, like so:

```
PATH="$HOME/.local/bin:$HOME/bin:$PATH:/opt/couchbase/bin"
export PATH
```

Add the following 2 lines after line 9 edit

```
MANPATH=$MANPATH:/opt/Couchbase/share/man
export MANPATH
```

USING AMAZON worldwide DNS names provided at the start of this course by your instructor on the .XLS spreadsheet

Make the following edits(add lines)…:

```
APPSERVER=ec2-48-54-199-69.us-west-1.compute.amazonaws.com
export APPSERVER

NODE1=ec2-113-156-188-191.us-west-1.compute.amazonaws.com
NODE2=ec2-113-156-209-124.us-west-1.compute.amazonaws.com
NODE3=ec2-113-157-200-184.us-west-1.compute.amazonaws.com
NODE4=ec2-118-144-149-199.us-west-1.compute.amazonaws.com
NODE5=ec2-154-183-155-127.us-west-1.compute.amazonaws.com
NODE6=ec2-154-183-183-230.us-west-1.compute.amazonaws.com
NODE7=ec2-154-219-170-126.us-west-1.compute.amazonaws.com
NODE8=ec2-154-219-174-110.us-west-1.compute.amazonaws.com

export NODE1 NODE2 NODE3 NODE4 NODE5 NODE6 NODE7 NODE8

CB_REST_USERNAME=Administrator
CB_REST_PASSWORD=couchbase
export CB_REST_USERNAME CB_REST_PASSWORD
```

USING AMAZON worldwide DNS names provided at the start of this course

Save and quit the vi or nano session.

Source the .bashrc file so that the changes you made take effect in the current bash session:

```
[ec2-user@Couchbase01 ~]$ source ~/.bashrc
```

```
[ec2-user@Couchbase01 ~]$ couchbase-cli
usage: couchbase-cli [-h]  ...

Commands:

    bucket-compact             Compact database and view data
    bucket-create              Add a new bucket to the cluster
    bucket-delete              Delete an existing bucket
    bucket-edit                Modify settings for an existing bucket
    bucket-flush               Flush all data from disk for a given bucket
    bucket-list                List all buckets in a cluster
    cluster-init               Initialize a Couchbase cluster
    collect-logs-start         Start cluster log collection
    collect-logs-status        View the status of cluster log collection
    collect-logs-stop          Stop cluster log collection
    collection-manage          Manage collections in a bucket
    enable-developer-preview   Enable developer preview mode in target cluster
    eventing-function-setup    Manage Eventing Service Functions
    failover                   Failover one or more servers
    group-manage               Manage server groups
    host-list                  List all hosts in a cluster
    ip-family                  Change or get the address family
    master-password            Unlocking the master password
    node-init                  Set node specific settings
    node-to-node-encryption    Change or get the cluster encryption
                               configuration
    rebalance                  Start a cluster rebalancing
    rebalance-status           Show rebalance status
    rebalance-stop             Stop a rebalance
    recovery                   Recover one or more servers
    reset-admin-password       Resets the administrator password
    reset-cipher-suites        Rests cipher suites to the default
    server-add                 Add servers to the cluster
    server-info                Show details of a node in the cluster
    server-list                List all nodes in a cluster
    setting-alert              Modify email alert settings
    setting-alternate-address
                               Configure alternate addresses
    setting-audit              Modify audit settings
    setting-autofailover       Modify auto failover settings
    setting-autoreprovision    Modify auto-reprovision settings
    setting-cluster            Modify cluster settings
    setting-compaction         Modify auto-compaction settings
    setting-index              Modify index settings
    setting-ldap               Modify LDAP settings
    setting-master-password    Changing the settings of the master password
    setting-notification       Modify email notification settings
    setting-password-policy    Modify the password policy
    setting-query              Manage query settings
    setting-rebalance          Configure automatic rebalance settings
    setting-saslauthd          Modify saslauthd settings
    setting-security           Modify security settings
    setting-xdcr               Modify XDCR related settings
    ssl-manage                 Manage cluster certificates
    user-change-password       Change user password
    user-manage                Manage RBAC users
```

```
    xdcr-replicate            Manage XDCR cluster references
    xdcr-setup                Manage XDCR replications

Options:
  -h,--help                   Prints the short or long help message
  --version version           Get couchbase-cli version
```

Test out a few common couchbase-cli commands to get familiar with the interface.

First, get your Couchbase hostname from the Cluster-IPs spreadsheet. You will need this for the next few commands. This will look something like: ec2-54-85-43-128.compute-1.amazonaws.com

The basic usage format for CLI commands is:
`couchbase-cli COMMAND [BUCKET_NAME] CLUSTER [OPTIONS]`

We will provide the CLUSTER specification for the command using the long form syntax:
--cluster=HOST[:PORT]

The OPTIONS part includes the username and password to invoke the command as
(you set this as a VARIABLE in your .bashrc file).

Run the 'server-list' command. In this command, you will have to mention your public hostname and port.

Warning: if you did not set the hostname VARIABLES you must change the hostname below to match your specific server's public hostname!

If you declared NODE1-8 VARIABLES AND LOGIN AND PASSWORD VARIABLES then your command should look like this:

```
[ec2-user@Couchbase01 ~]$ couchbase-cli server-list --cluster=$NODE1:8091
--username=Administrator --password=couchbase
```

If you did not do this then you MUST type in the following:

```
[ec2-user@Couchbase01 ~]$ couchbase-cli server-list --cluster=ec2-54-174-65-
105.compute-1.amazonaws.com:8091 --username=Administrator
--password=couchbase
```

```
ns_1@ec2-54-174-65-105.compute-1.amazonaws.com ec2-54-174-65-
105.compute-1.amazonaws.com:8091 healthy active
```

Run the 'server-info' command, which requires the username and password also:

```
[ec2-user@Couchbase01 ~]$ couchbase-cli server-info --cluster=$NODE1:8091
```


```
{
  "availableStorage": {
    "hdd": [
      {
        "path": "/dev",
        "sizeKBytes": 1919920,
        "usagePercent": 0
      },
      {
        "path": "/dev/shm",
        "sizeKBytes": 1940204,
        "usagePercent": 0
      },
      {
        "path": "/run",
        "sizeKBytes": 1940204,
        "usagePercent": 1
      },
      {
        "path": "/sys/fs/cgroup",
        "sizeKBytes": 1940204,
        "usagePercent": 0
      },
      {
        "path": "/",
        "sizeKBytes": 10473452,
        "usagePercent": 22
      },
      {
        "path": "/run/user/1000",
        "sizeKBytes": 388044,
        "usagePercent": 0
      }
    ]
  },
  "cbasMemoryQuota": 1167,
  "clusterCompatibility": 327685,
  "clusterMembership": "active",
  "couchApiBase": "http://ec2-13-56-178-208.us-west-1.compute.amazonaws.com:8092/",
  "couchApiBaseHTTPS": "https://ec2-13-56-178-208.us-west-1.compute.amazonaws.com:18092/",
  "cpuCount": 2,
  "eventingMemoryQuota": 256,
  "ftsMemoryQuota": 256,
  "hostname": "ec2-13-56-178-208.us-west-1.compute.amazonaws.com:8091",
  "indexMemoryQuota": 512,
  "interestingStats": {
    "cmd_get": 0,
```

```
    "couch_docs_actual_disk_size": 119996726,
    "couch_docs_data_size": 92579995,
    "couch_spatial_data_size": 0,
    "couch_spatial_disk_size": 0,
    "couch_views_actual_disk_size": 787706,
    "couch_views_data_size": 787706,
    "curr_items": 39480,
    "curr_items_tot": 39480,
    "ep_bg_fetched": 0,
    "get_hits": 0,
    "mem_used": 118595568,
    "ops": 0,
    "vb_active_num_non_resident": 0,
    "vb_replica_curr_items": 0
  },
  "mcdMemoryAllocated": 3030,
  "mcdMemoryReserved": 3030,
  "memoryFree": 2419314688,
  "memoryQuota": 2120,
  "memoryTotal": 3972685824,
  "os": "x86_64-unknown-linux-gnu",
  "otpCookie": "8fa2f5590e174e256a0c4c23adfe3407d9c5dc0b632e9856ffdf269888001083",
  "otpNode": "ns_1@ec2-13-56-178-208.us-west-1.compute.amazonaws.com",
  "ports": {
    "direct": 11210,
    "httpsCAPI": 18092,
    "httpsMgmt": 18091,
    "proxy": 11211
  },
  "recoveryType": "none",
  "services": [
    "fts",
    "index",
    "kv",
    "n1ql"
  ],
  "status": "healthy",
  "storage": {
    "hdd": [
      {
        "cbas_dirs": [
          "/opt/couchbase/var/lib/couchbase/analytics"
        ],
        "index_path": "/opt/couchbase/var/lib/couchbase/index",
        "path": "/opt/couchbase/var/lib/couchbase/data",
        "quotaMb": "none",
        "state": "ok"
      }
    ],
    "ssd": []
  },
  "storageTotals": {
    "hdd": {
      "free": 8043611136,
      "quotaTotal": 10724814848,
      "total": 10724814848,
      "used": 2681203712,
      "usedByData": 120784432
    },
    "ram": {
      "quotaTotal": 2222981120,
      "quotaTotalPerNode": 2222981120,
      "quotaUsed": 314572800,
      "quotaUsedPerNode": 314572800,
      "total": 3972685824,
      "used": 2943258624,
      "usedByData": 118595568
    }
  },
```

```
  "systemStats": {
    "cpu_utilization_rate": 8.629441624365482,
    "mem_free": 2419314688,
    "mem_total": 3972685824,
    "swap_total": 0,
    "swap_used": 0
  },
  "thisNode": true,
  "uptime": "2744",
  "version": "6.5.1-4960-enterprise"}
```

Exploring Couchbase index and data directories:

Let's take a look at what's in the index and data directories on disk:

```
[ec2-user@Couchbase01 ~]$ sudo ls -alh
/opt/couchbase/var/lib/couchbase/indexes
total 8.0K
drwxrwx---.  6 couchbase couchbase   56 May  5 11:33 .
drwxr-xr-x.  8 couchbase couchbase 4.0K May  5 14:14 ..
drwxr-x---. 10 couchbase couchbase 4.0K May  5 12:18 @2i
drwxrwx---.  4 couchbase couchbase   84 May  5 12:17 .delete
drwxrwx---.  2 couchbase couchbase   22 May  5 12:16 @fts
drwxrwx---.  5 couchbase couchbase   65 May  5 12:17 @indexes

[ec2-user@Couchbase01 ~]$ sudo ls -alh
/opt/couchbase/var/lib/couchbase/indexes/@indexes
total 8.0K
drwxrwx---. 5 couchbase couchbase   65 May  5 12:17 .
drwxrwx---. 6 couchbase couchbase   56 May  5 11:33 ..
drwxrwx---. 3 couchbase couchbase 4.0K May  5 11:36 beer-sample
drwxrwx---. 2 couchbase couchbase   57 May  5 11:42 gamesim-sample

[ec2-user@Couchbase01 ~]$ sudo ls -alh
/opt/couchbase/var/lib/couchbase/indexes/@indexes/beer-sample
total 756K
drwxrwx---. 3 couchbase couchbase 4.0K May  5 11:36 .
drwxrwx---. 5 couchbase couchbase   65 May  5 12:17 ..
-rw-rw----. 1 couchbase couchbase 752K May  5 11:36 main_5a222b8c920aa5e3a28b51ee7eb609a0.view.1
drwxrwx---. 2 couchbase couchbase    6 May  5 11:36 tmp_5a222b8c920aa5e3a28b51ee7eb609a0_main
```

Views within Couchbase process the information stored in the Couchbase database, allowing you to index and query your data. A view creates an index on the stored information according to the format and structure defined within the view. Views in Couchbase will be covered in depth later in the course.

Next, explore the data directory, specifically the beer-sample database files:

```
[ec2-user@Couchbase01 ~]$ sudo ls -alh
/opt/couchbase/var/lib/couchbase/data

total 160K
drwxrwx---.  6 couchbase couchbase   83 Jul 31 16:48 .
drwxr-xr-x. 10 couchbase couchbase 4.0K Jul 31 17:26 ..
drwxrwx---.  2 couchbase couchbase  28K Jul 31 17:26 beer-sample
drwxrwx---.  2 couchbase couchbase    6 Jul 31 16:37 .delete
drwxrwx---.  2 couchbase couchbase  28K Jul 31 17:25 gamesim-sample
drwxrwx---.  2 couchbase couchbase  28K Jul 31 17:26 travel-sample
```

```
[ec2-user@Couchbase01 ~]$ sudo ls -alh
/opt/couchbase/var/lib/couchbase/data/beer-sample

total 16M
drwxrwx---. 2 couchbase couchbase  28K May  5 14:18 .
drwxrwx---. 7 couchbase couchbase 4.0K May  5 12:16 ..
-rw-rw----. 1 couchbase couchbase  17K May  5 11:36 0.couch.1
-rw-rw----. 1 couchbase couchbase 8.1K May  5 11:36 1000.couch.1
-rw-rw----. 1 couchbase couchbase 8.1K May  5 11:36 1001.couch.1
-rw-rw----. 1 couchbase couchbase 8.1K May  5 11:36 1002.couch.1
-rw-rw----. 1 couchbase couchbase  13K May  5 11:36 1003.couch.1
-rw-rw----. 1 couchbase couchbase 8.1K May  5 11:36 1004.couch.1
-rw-rw----. 1 couchbase couchbase 8.1K May  5 11:36 1005.couch.1
-rw-rw----. 1 couchbase couchbase 8.1K May  5 11:36 1006.couch.1
-rw-rw----. 1 couchbase couchbase  13K May  5 11:36 1007.couch.1
-rw-rw----. 1 couchbase couchbase 8.1K May  5 11:36 1008.couch.1
-rw-rw----. 1 couchbase couchbase 8.1K May  5 11:36 1009.couch.1
-rw-rw----. 1 couchbase couchbase  13K May  5 11:36 100.couch.1
-rw-rw----. 1 couchbase couchbase  13K May  5 11:36 1010.couch.1
-rw-rw----. 1 couchbase couchbase 8.1K May  5 11:36 1011.couch.1
-rw-rw----. 1 couchbase couchbase 8.1K May  5 11:36 1012.couch.1
-rw-rw----. 1 couchbase couchbase 8.1K May  5 11:36 1013.couch.1
-rw-rw----. 1 couchbase couchbase 8.1K May  5 11:36 1014.couch.1
-rw-rw----. 1 couchbase couchbase  13K May  5 11:36 1015.couch.1
 <output truncated>
```

Notice that there is 30 MB of data in this directory. (*Note: on your machine this can range between 25 – 35 MB*)

Try counting the number of files in this directory:

```
[ec2-user@Couchbase01 ~]$ sudo ls -al
/opt/couchbase/var/lib/couchbase/data/beer-sample | wc -l
1030
```

You should see about 1030 items. There will be 1024 couchbase partitions (vBucket files) along with a back index, metadata files, etc.

Next, run the couch_dbdump command to inspect one of the vbucket files in the beer-sample database:

```
[ec2-user@Couchbase01 ~]$ sudo /opt/couchbase/bin/couch_dbdump
/opt/couchbase/var/lib/couchbase/data/beer-sample/0.couch.1 | head -20

Dumping "/opt/couchbase/var/lib/couchbase/data/beer-sample/0.couch.1":
Doc seq: 1
     id: lafayette_brewing-black_angus_oatmeal_stout
     rev: 1
```

```
     content_meta: 128
     size (on disk): 230
     cas: 1438282944544964608, expiry: 0, flags: 0, datatype: 1, conflict_resolution_mode: 0
     size: 233
     data: (snappy) {"name":"Black Angus Oatmeal
Stout","abv":0.0,"ibu":0.0,"srm":0.0,"upc":0,"type":"beer","brewery_id":"lafayette_brewing","upda
ted":"2010-07-22 20:00:20","description":"","style":"American-Style Stout","category":"North
American Ale"}
Doc seq: 2
     id: el_toro_brewing_company
     rev: 1
     content_meta: 128
     size (on disk): 1496
     cas: 1438282944545030144, expiry: 0, flags: 0, datatype: 1, conflict_resolution_mode: 0
     size: 1826
     data: (snappy) {"name":"El Toro Brewing Company","city":"Morgan
Hill","state":"California","code":"95037","country":"United States","phone":"408-782-
2739","website":"http://www.eltorobrewing.com/","type":"brewery","updated":"2010-07-22
20:00:20","description":"Geno and Cindy Acevedo founded El Toro Brewing Company in the summer of
1992. After much planning, research and construction business was begun in March of 1994. The
brewery is a small 17-barrel (527 gallon) batch system and operates on the Acevedo's rural
residential property.  Within seven months of opening El Toro Brewing Company received its first
national recognition at the 1994 Great American Beer Festival (GABF). One of its flagships El
Toro Oro Golden Ale won a coveted Gold Medal for English Pale Ale. At the 1996 GABF Poppy Jasper
Amber Ale, the brewery's other flagship beer, won a Silver Medal for English Brown Ale. It is a
mild yet robust brown ale. The 1997 GABF saw El Toro garner another Gold Medal for the American
styled wheat beer named after the brewer's father-in-law, William Jones Wheat Beer.  After 16
years of planning, Geno and Cindy Acevedo of Morgan Hill finally opened a brewpub on November 25
2006. El Toro Brewpub is a two floored building with patio. Inside is the world's only Poppy
Jasper Bar. Featuring over 45 feet of gorgeously inlaid and polished Poppy Jasper rock into its
surface. To the back of the large mirrored bar you will find over 25 beers and handcrafted sodas
on tap. We also have a gleaming copper Pub Brewing system 3 BBL (100 gallon) brewery. Live music
and dancing will also be a regular nighttime happening at the brewpub consisting of mostly local
bands playing cover type and original Rock, Blues, Jazz and Reggae.","address":["17605 Monterey
Road"],"geo":{"accuracy":"RANGE_INTERPOLATED","lat":37.1553,"lon":-121.676}}
Doc seq: 3
     id: st_austell_brewery-hsd_hicks_special_draught
     rev: 1
```

Notice that there are 3(partial) JSON documents in this specific data file.

You can also print information about the data file with the couch_dbinfo command:

```
[ec2-user@ Couchbase01 ~]$ sudo /opt/couchbase/bin/couch_dbinfo
/opt/couchbase/var/lib/couchbase/data/beer-sample/0.couch.1

DB Info (/opt/couchbase/var/lib/couchbase/data/beer-sample/0.couch.1)
- header at 16384
   file format version: 12
   update_seq: 9
   purge_seq: 0
   crc: CRC-32C
   doc count: 9
   deleted doc count: 0
   data size: 6.6 kB
   B-tree size: 1.2 kB
   total disk size: 16.1 kB
```

Notice that the above output shows the doc count in this data file along with the actual data size and the total disk file size (which includes metadata). (*The total disk file size in your specific environment can vary between 24 kB and 40 kB*) The # of deleted doc count is also shown, but in our sample database there are currently no deleted docs in the data file.

Startup and shutdown on Linux:

On Linux, Couchbase Server is installed as a standalone application with support for running as a background (daemon) process during startup through the use of a standard control script, /etc/init.d/couchbase-server. The startup script is automatically installed during installation.

By default, Couchbase Server is configured to be started automatically at run levels 2, 3, 4, and 5, and explicitly shutdown at run levels 0, 1 and 6.

To manually stop Couchbase Server using the systemctl:

```
[ec2-user@ Couchbase01 ~]$ sudo systemctl stop couchbase-server
]
```

You may notice that the web UI will now start reporting that it has lost connection to the server:

Before starting Couchbase back up, take a look at the info log file for Couchbase. Promote yourself to root and continue:

```
[ec2-user@ Couchbase01 ~]$ sudo -s

[root@Couchbase01 ec2-user]# cd /opt/couchbase/var/lib/couchbase/logs

[root@Couchbase01 logs]# ls -alh
total 24M
drwxr-xr-x. 2 couchbase couchbase  4.0K Apr 19 22:44 .
drwxr-xr-x. 8 couchbase couchbase  4.0K Apr 19 22:44 ..
-rw-rw----. 1 couchbase couchbase  824K Apr 19 22:44 babysitter.log
-rw-rw----. 1 couchbase couchbase  726K Apr 19 22:44 couchdb.log
-rw-rw----. 1 couchbase couchbase    13 Apr 19 22:44 crash_log.bin
-rw-rw----. 1 couchbase couchbase     0 Apr 19 22:44 crash_log.bin.tmp
-rw-rw----. 1 couchbase couchbase  5.2M Apr 19 22:44 debug.log
-rw-rw----. 1 couchbase couchbase  8.9K Apr 19 22:44 error.log
-rw-rw----. 1 couchbase couchbase   19K Apr 19 22:44 goxdcr.log
-rw-rw----. 1 couchbase couchbase  529K Apr 19 22:44 http_access.log
-rw-rw----. 1 couchbase couchbase 1001K Apr 19 22:44 indexer.log
-rw-rw----. 1 couchbase couchbase  3.1M Apr 19 22:44 info.log
-rw-rw--    1 couchbase couchbase 0 Apr 19 17:35 mapreduce_errors.log
-rw-rw---   1 couchbase couchbase 2.4M Apr 19 22:44 memcached.log.0.txt
-rw-rw----. 1 couchbase couchbase  1.5M Apr 19 22:44 ns_couchdb.log
```

```
-rw-rw----. 1 couchbase couchbase  615K Apr 19 22:44 projector.log
-rw-rw----. 1 couchbase couchbase   637 Apr 19 22:44 query.log
-rw-rw----. 1 couchbase couchbase  276K Apr 19 22:44 reports.log
-rw-rw----. 1 couchbase couchbase  8.5K Apr 19 18:31 ssl_proxy.log
-rw-r--r--. 1 root      root          9 Apr 19 17:35 start.log
-rw-rw----. 1 couchbase couchbase  7.3M Apr 19 22:44 stats.log
-rw-rw----. 1 couchbase couchbase  107K Apr 19 21:45 views.log
-rw-rw----. 1 couchbase couchbase     0 Apr 19 17:35 xdcr_errors.log
-rw-rw----. 1 couchbase couchbase  1.2K Apr 19 18:31 xdcr.log
-rw-rw----. 1 couchbase couchbase     0 Apr 19 17:35 xdcr_trace.log
```

Print the last 21 lines of the info log file using the tail command:

```
[root@Couchbase01 logs]# tail -21 info.log
                                                        "travel-sample"]
[ns_server:warn,2017-09-26T17:57:12.714Z,ns_1@ec2-54-183-85-83.us-west-
1.compute.amazonaws.com:<0.2873.66>:ns_memcached:connect:1187]Unable to connect:
{error,{badmatch,{error,closed}}}, retrying.
[ns_server:warn,2017-09-26T17:57:12.714Z,ns_1@ec2-54-183-85-83.us-west-
1.compute.amazonaws.com:<0.2893.66>:ns_memcached:connect:1187]Unable to connect:
{error,{badmatch,{error,closed}}}, retrying.
[ns_server:warn,2017-09-26T17:57:12.714Z,ns_1@ec2-54-183-85-83.us-west-
1.compute.amazonaws.com:<0.2876.66>:ns_memcached:connect:1187]Unable to connect:
{error,{badmatch,{error,closed}}}, retrying.
[ns_server:warn,2017-09-26T17:57:12.714Z,ns_1@ec2-54-183-85-83.us-west-
1.compute.amazonaws.com:<0.2866.66>:ns_memcached:connect:1187]Unable to connect:
{error,{badmatch,{error,closed}}}, retrying.
[ns_server:info,2017-09-26T17:57:12.735Z,ns_1@ec2-54-183-85-83.us-west-
1.compute.amazonaws.com:<0.2.0>:ns_bootstrap:stop:42]Initiated server shutdown
[ns_server:info,2017-09-26T17:57:12.738Z,ns_1@ec2-54-183-85-83.us-west-
1.compute.amazonaws.com:mb_master<0.6413.50>:mb_master:terminate:298]Synchronously shutting down
child mb_master_sup
[ns_server:warn,2017-09-26T17:57:13.717Z,ns_1@ec2-54-183-85-83.us-west-
1.compute.amazonaws.com:<0.2873.66>:ns_memcached:connect:1187]Unable to connect:
{error,{badmatch,{error,econnrefused}}}, retrying.
[ns_server:warn,2017-09-26T17:57:13.717Z,ns_1@ec2-54-183-85-83.us-west-
1.compute.amazonaws.com:<0.2893.66>:ns_memcached:connect:1187]Unable to connect:
{error,{badmatch,{error,econnrefused}}}, retrying.
[ns_server:warn,2017-09-26T17:57:13.717Z,ns_1@ec2-54-183-85-83.us-west-
1.compute.amazonaws.com:<0.2876.66>:ns_memcached:connect:1187]Unable to connect:
{error,{badmatch,{error,econnrefused}}}, retrying.
[ns_server:info,2017-09-26T17:57:14.320Z,ns_1@ec2-54-183-85-83.us-west-
1.compute.amazonaws.com:ns_couchdb_port<0.6096.50>:ns_port_server:log:223]ns_couchdb<0.6096.50>:
22183: got shutdown request. Exiting
ns_couchdb<0.6096.50>: [os_mon] memory supervisor port (memsup): Erlang has closed
ns_couchdb<0.6096.50>: [os_mon] cpu supervisor port (cpu_sup): Erlang has closed

[ns_server:info,2017-09-26T17:57:14.324Z,ns_1@ec2-54-183-85-83.us-west-
1.compute.amazonaws.com:<0.2.0>:ns_bootstrap:stop:46]Successfully stopped ns_server


[root@Couchbase01 logs]# exit
exit
```

We will explore the rest of the logs in this directory in future labs, but if you're interested in what the rest of the logs contain, go here:
http://docs.couchbase.com/admin/admin/Misc/Trbl-logs.html

Manually start Couchbase Server back up:

```
[ec2-user@Couchbase01 ~]$ sudo systemctl start couchbase-server
```

There may be cases where you want to explicitly shutdown a server and then restart it. Typically the server had been running for a while and has data stored on disk when you restart it. In this case, the server needs to undergo a warmup process before it can again serve data requests.

Warmup is a process a restarted server must undergo before it can serve data. During this process the server loads items persisted on disk into RAM. One approach to load data is to do sequential loading of items from disk into RAM; however it is not necessarily an effective process because the server does not take into account whether the items are frequently used. In Couchbase Server, additional optimizations are provided during the warmup process to make data more rapidly available, and to prioritize frequently-used items in an access log. The server pre-fetches a list of most-frequently accessed keys and fetches these documents before it fetches any other items from disk.

Warmup will be discussed in more depth in a future lab. For now, you should know that the server can also switch into a ready mode before it has actually retrieved all documents for keys into RAM, and therefore can begin serving data before it has loaded all stored items. This is also a setting you can configure so that server warmup is faster.

Wait a full 40 seconds to allow Couchbase to start up, before running the status command:

```
[ec2-user@Couchbase01 ~]$ sudo systemctl status couchbase-server
couchbase-server.service - Couchbase Server
   Loaded: loaded (/usr/lib/systemd/system/couchbase-server.service;
enabled; vendor preset: disabled)
   Active: active (running) since Mon 2017-01-09 14:25:47 EST; 1 day
2h ago
     Docs: http://docs.couchbase.com
  Process: 9662 ExecStart=/opt/couchbase/bin/couchbase-server -- -
noinput -detached (code=exited, status=0/SUCCESS)
 Main PID: 9734 (beam.smp)
   CGroup: /system.slice/couchbase-server.service
           ├─ 9674 /opt/couchbase/lib/erlang/erts-
5.10.4.0.0.1/bin/epmd -daemon
           ├─ 9705 /opt/couchbase/lib/erlang/erts-
5.10.4.0.0.1/bin/beam.smp -A 16 -- -root /opt/couchbase/lib/erlang -
progname erl --...
           ├─ 9730 /opt/couchbase/bin/gosecrets
           ├─ 9734 /opt/couchbase/lib/erlang/erts-
5.10.4.0.0.1/bin/beam.smp -A 16 -sbt u -P 327680 -K true -swt low -
MMmcs 30 -e10240...
           ├─ 9762 sh -s disksup
```

```
           ├─ 9764 /opt/couchbase/lib/erlang/lib/os_mon-
2.2.14/priv/bin/memsup
           ├─ 9765 /opt/couchbase/lib/erlang/lib/os_mon-
2.2.14/priv/bin/cpu_sup
           ├─ 9766 inet_gethost 4
           ├─ 9767 inet_gethost 4
           ├─ 9817 /opt/couchbase/bin/saslauthd-port
           ├─ 9838 /opt/couchbase/bin/memcached -C
/opt/couchbase/var/lib/couchbase/config/memcached.json
{output truncated…….}
Jan 10 16:40:47 Couchbase01 systemd[1]: Starting Couchbase Server...
Jan 10 16:40:49 Couchbase01 systemd[1]: PID file
/opt/couchbase/var/lib/couchbase/couchbase-server.pid not readable
(yet?) after start.
Jan 10 16:40:52 Couchbase01 systemd[1]: couchbase-server.service:
Supervising process 16958 which is not our child. We'll most... exits.
Jan 10 16:40:52 Couchbase01 systemd[1]: Started Couchbase Server.
Hint: Some lines were ellipsized, use -l to show in full.
```

Type in **q** (for quit)

Then exit

exit

This concludes the first lab.