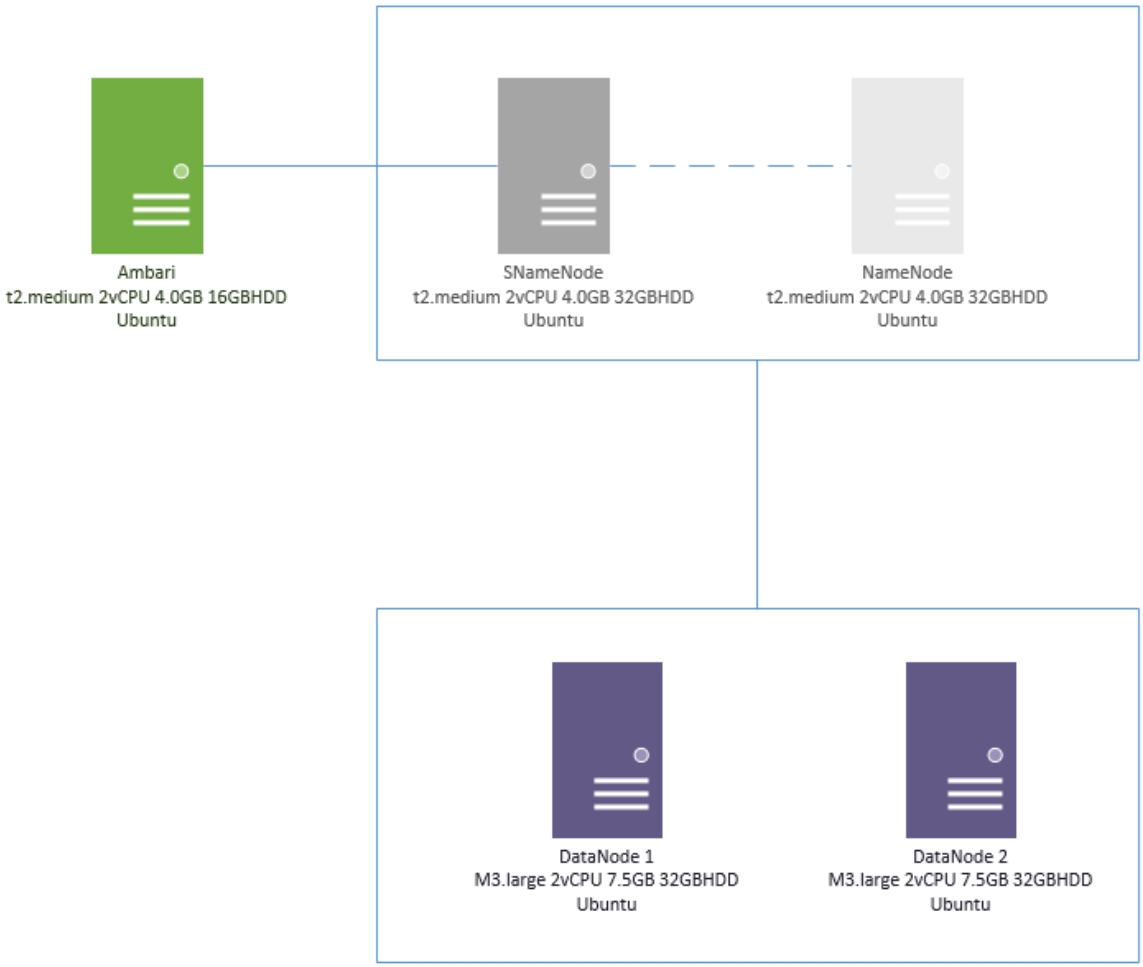
Cluster Topology



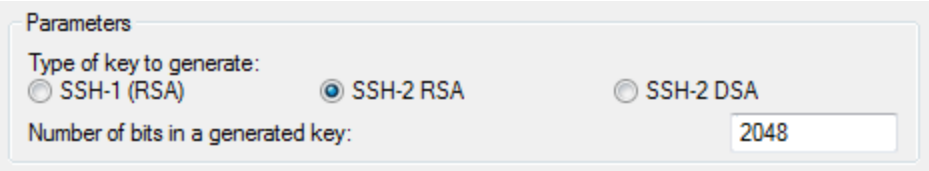
EC2 Instance Launch

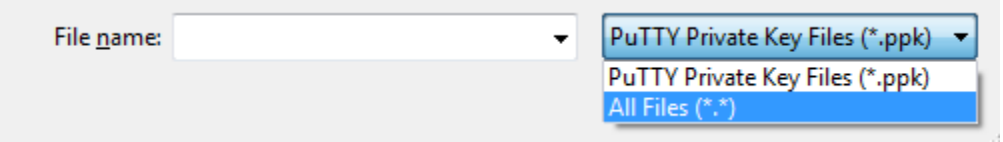
1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/ .
2. Choose Launch Instance.
3. Choose an Amazon Machine Image (AMI) - (Ubuntu)
4. Choose an Instance
5. Configure instance set the instance region etc.
6. Add more storage if needed
7. Add a tag to the instance
8. Set the security group during setup set - Ports needed for communication – 8080 for Ambari is required initially
9. Review and launch
10. Create a new key-pair or use existing.
11. Instance launch complete

Logging into EC2 from Putty

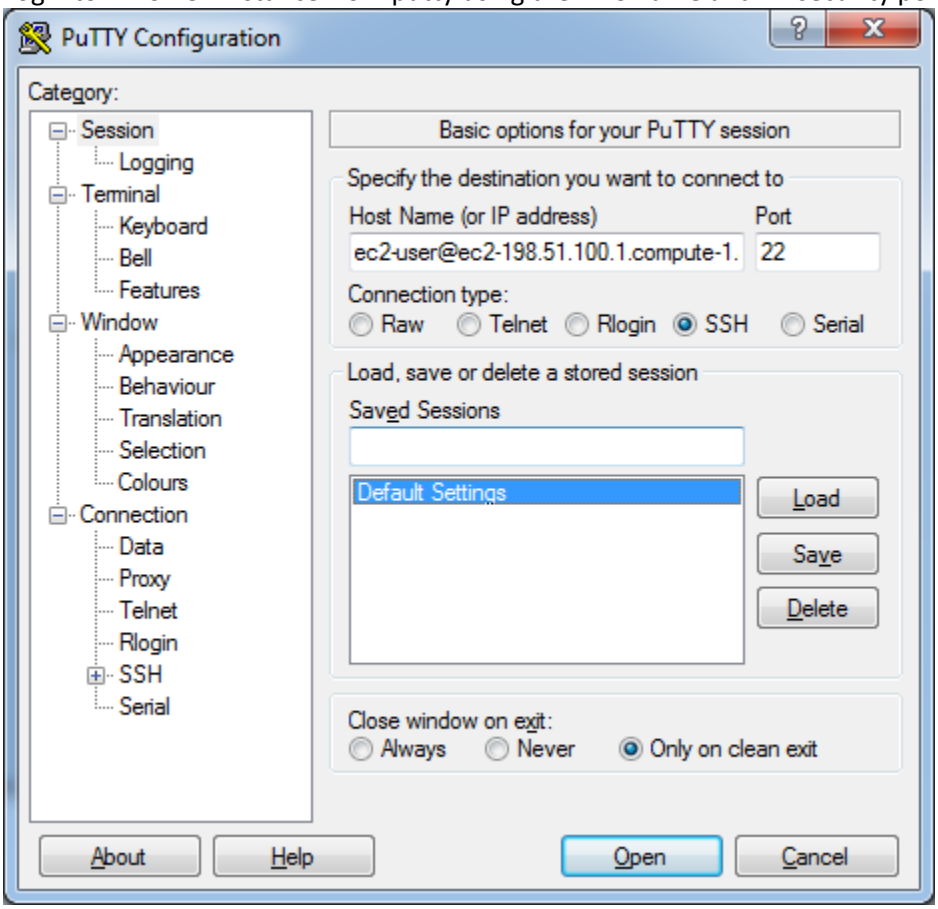
1. Download putty & puttygen from http://www.putty.org/
2. Note:
   .pem file is what we have download from AWS when we created our key-pair. This is only a one time download and we cannot download it again. our software is asking for .pem file. Now we are the one who needs to locate that file. we might have downloaded it on our laptop/desktop...etc.

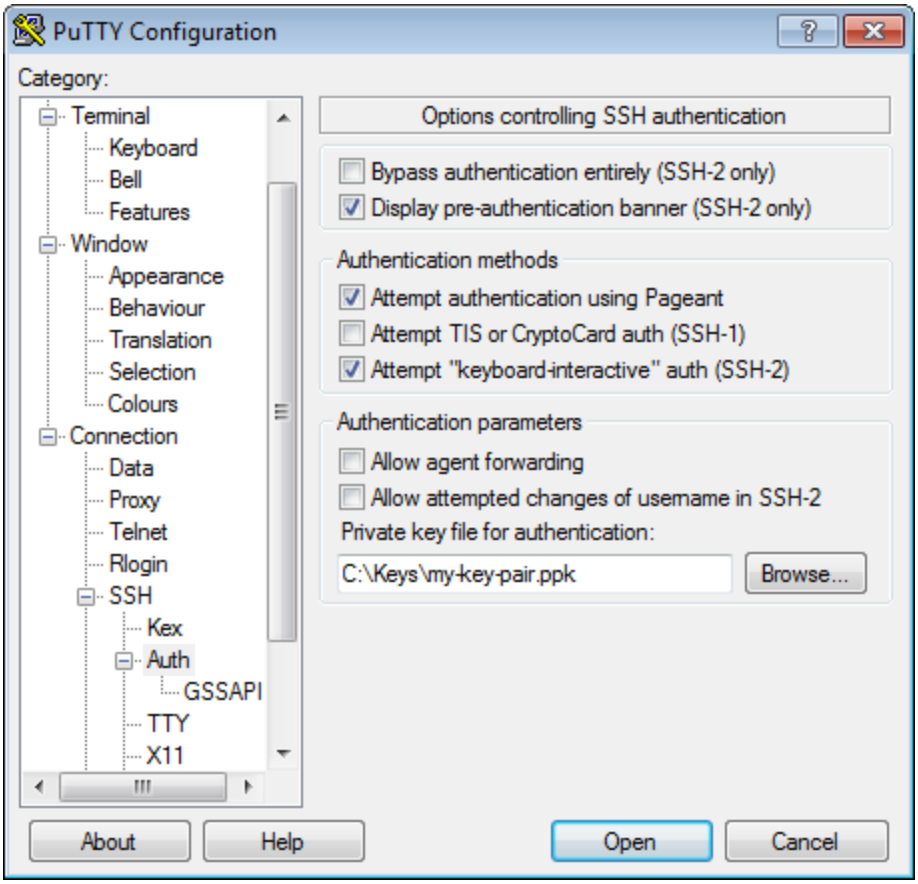
   For Putty (windows ssh client), it does not support .pem format. Hence we have to convert it to .ppk format using PuTTyGen. It's essentially .pem but in a different format so that Putty can work with it.
3. Generate the .ppk file for .pem for putty to use





4. Get the DNS name from amazon ec2 console
5. Login to AWS EC2 instance from putty using the DNS name and 22 security port set the auth to ppk file





6. Login successful

EC2 Instance Configuration

1. Add new user
   root>sudo adduser developer
2. switch to new user account
   root>sudo su – developer
3. create .ssh folder
   developer>mkdir .ssh
4. Create new file authorized_keys
   developer>touch .ssh/authorized_keys
5. Change setting to writ into file
   developer>chmod 777 .ssh/authorized_keys
   developer>chmod 777 .ssh
6. Copy privatekey to new user
   developer>logout
   root>cat .ssh/authorized_keys > /home/developer/.ssh/authorized_keys
7. Switch back to developer and change permissions
   root>sudo su – developer
   developer>chmod 700 .ssh/authorized_keys
   developer>chmod 700 .ssh
8. Assign root privileges to developer
   developer>logout
   root>sudo visudo

   developer ALL=(ALL:ALL) NOPASSWD: ALL
9. Install wget, curl, Oracle java
   developer>sudo su - developer

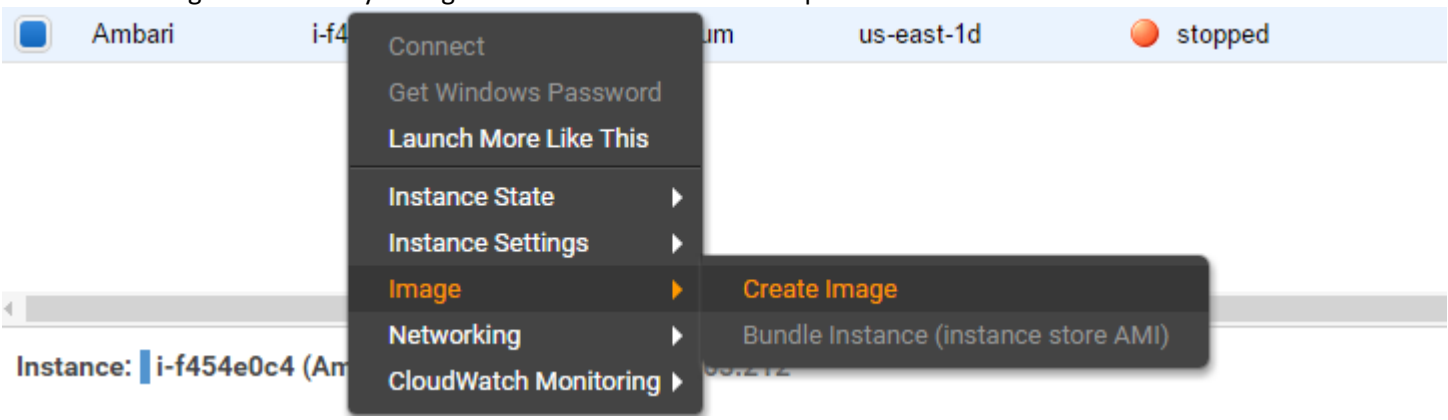
developer>sudo apt-get update
developer>sudo apt-get install wget
developer>sudo apt-get install curl
developer>sudo apt-get install default-jre
developer>sudo apt-get install default-jdk
developer>sudo apt-get install python-software-properties
developer>sudo add-apt-repository ppa:webupd8team/java
developer>sudo apt-get update
developer>sudo apt-get install oracle-java8-installer
developer>sudo update-alternatives --config java
developer>sudo update-alternatives --config javac
developer>sudo nano /etc/environment

JAVA_HOME="/usr/lib/jvm/java-7-oracle”

10. Install ntp
    developer> sudo apt-get install ntpd
    developer> sudo service ntp start

EC2 create Image

Create AMI image from already configured instance to launch multiple instances at once.



Ambari Installation

1. Login as developer in one of the machines
2. Download repositories and update apt-get
   wget -nv http://public-repo-1.hortonworks.com/ambari/ubuntu14/2.x/updates/2.1.2/ambari.list -O /etc/apt/sources.list.d/ambari.list

   apt-key adv --recv-keys --keyserver keyserver.ubuntu.com B9733A7A07513CAD

   apt-get update
3. Check successful installation
   apt-cache showpkg ambari-server

   apt-cache showpkg ambari-agent

   apt-cache showpkg ambari-metrics-assembly
   *We should see Ambari notes
4. Install the Ambari bits. This also installs the default PostgreSQL Ambari database.

apt-get install ambari-server

5. After installation is complete setup ambari using

   sudo ambari-server setup

6. By default, Ambari Server runs under root. Accept the default (n) at the Customize user account for ambari-server daemon prompt, to proceed as root. Otherwise select (y) and customize the user account
7. Select the default database and default user/password/schema or customize it.
8. Once the setup is complete start ambari using

   sudo ambary-server start

HDP Installation

1. Login to ambary console
   https://hostname:8080
2. Use default username and password as admin/admin
3. Launch a cluster
4. Set the name of cluster, select the stack and select the repo we want to install in this case its Ubuntu 14
5. In install options set the target hosts and provide the .pem key and confirm hosts
6. Once host registration is complete choose all the services we want to install (choose only the ones we want)
7. Set the namenode, secondarynamenode, resourcemanager, regionserver, nodemanager and all other servers (hive, spark, zookeeper, metrics etc.)
8. Review the option and install the services on those machines
9. Once all services all installed, services are started and your cluster is ready for use.