**Ubuntu 16.04 LTS Ambari 2.5.1 HDP 2.6.1 Setup guide**

1. After installing Ubuntu 16.04 update system with “apt-get update” command

2. Be sure the home directory is not encrypted. By running ‘ecryptfs-verify’ command it can be determined. If the tool is not installed, install it with “apt install ecryptfs-utils”.

Run the below command:

snag@ubuntugnome1604:~$ ecryptfs-verify -h

ERROR: [/home/snag/.ecryptfs] does not exist

ERROR: Configuration invalid

snag@ubuntugnome1604:~$

3. Be sure the system has apt, scp, curl, unzip, tar and wget

4. Python 2.7 should be the default python

5. JDK 8, 64-bit (minimum JDK 1.8.0\_77) should be installed and set default

6. Minimum 1 GB available memory is required

7. Setting up Password-less SSH

This is required to install Ambari agent automatically in the host. If manual Ambari agent installation is being performed no need of this step. Since installing agent is easier through the Ambari server webui installer better to set up password-less SSH. This is most useful when multinode cluster is set up. Using Amabari server host agent can be installed easily in all other hosts in the cluster.

For more information about ssh visit the link: <https://www.ssh.com/ssh/>

Before this setup if trying to ssh any host the connection will be refused:

snag@ubuntugnome1604:~$ ssh snag@ubuntugnome1604

ssh: connect to host ubuntugnome1604 port 22: Connection refused

snag@ubuntugnome1604:~$

The first step is to generate Public/Private key pair:

snag@ubuntugnome1604:~$ ssh-keygen

Follow the steps (do not provide any passphrase, just hit ENTER)

In user’s home directory .ssh folder will be created and two keys will be generated.

snag@ubuntugnome1604:~/.ssh$ ls

id\_rsa id\_rsa.pub

snag@ubuntugnome1604:~/.ssh$

The second step is to copy the public key (id\_rsa.pub) in to root account directory of target hosts. In case of single node cluster do the same in the system’s root account directory (~/.ssh/id\_rsa.pub).

Third step is to add the public key to the authorised keys file. Run the below command to do the same.

cat id\_rsa.pub >> authorized\_keys

So, now the “~/ .ssh” directory contains two files as below:

root@ubuntugnome1604:~/.ssh# ls -a

. .. authorized\_keys id\_rsa.pub

root@ubuntugnome1604:~/.ssh#

Final step is to provide required access level to .ssh directory and authorized\_keys file:

chmod 700 ~/.ssh

chmod 600 ~/.ssh/authorized\_keys

Now, we can test the password less ssh:

snag@ubuntugnome1604:~$ ssh root@ubuntugnome1604

Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.10.0-28-generic x86\_64)

Remember if the home directory is encrypted ssh will not work without root password.

8. At this point we can check some networking bits e.g. DNS (Domain Name System) and NTP (Network Time Protocol) but the default settings for these are normally compliant with Hadoop system.

9. FQDN (Fully Qualified Domain Name):

By considering the ‘/etc/hosts’ file the FQDN can be determined. In my system, the information is as below:

127.0.0.1 localhost

127.0.1.1 ubuntugnome1604

Now this is confusing!

We are using Ubuntu, which is a Debian based system, maintains DNS in a way which is different than ‘rpm’ systems e.g. Fedora or CentOS.

‘localhost’ is a special name which points to the ip address 127.0.0.1. We can think of this ‘localhost’ as a pseudo device which is only valid in the local system environment. When one system interacts with internet, it is known/recognized by its hostname which is ‘ubuntugnome1604’ in this case.

So, the hostname or FQDN here is ‘ubuntugnome1604’.

You can just run hostname or hostname -f to get the FQDN:

root@ubuntugnome1604:~# hostname -f

ubuntugnome1604

root@ubuntugnome1604:~# hostname

ubuntugnome1604

root@ubuntugnome1604:~#

10. Disabling iptables

iptables is a built-in firewall for Linux systems.

To disable iptables run the below commands:

snag@ubuntugnome1604:~$ sudo ufw disable

[sudo] password for snag:

Firewall stopped and disabled on system startup

snag@ubuntugnome1604:~$ sudo ufw status

Status: inactive

snag@ubuntugnome1604:~$

11. Disable SELinux

In Ubuntu by default SELinux is off. In CentOS or Fedora you need to disable SELinux.

12. GPG check

In Ubuntu GPG check is disable by default. For CentOS you need to edit the file ‘/etc/yum/pluginconf.d/priorities.conf’

INSTALLATION OF AMBARI SERVER

First step is to download ambari public repo information file (ambari.list). Public repository means all installation files will be downloaded at the time of installation. For this to happen you need an active internet connection throughout the installation procedure.

Run the below command with root privilege:

root@ubuntugnome1604:~# wget -O /etc/apt/sources.list.d/ambari.list <http://public-repo-1.hortonworks.com/ambari/ubuntu16/2.x/updates/2.5.1.0/ambari.list>

‘ambari.list’ file will be downloaded in /etc/apt/sources.list.d/ directory.

--2017-08-01 00:59:42-- http://public-repo-1.hortonworks.com/ambari/ubuntu16/2.x/updates/2.5.1.0/ambari.list

Resolving public-repo-1.hortonworks.com (public-repo-1.hortonworks.com)... 52.222.135.100, 52.222.135.88, 52.222.135.21, ...

Connecting to public-repo-1.hortonworks.com (public-repo-1.hortonworks.com)|52.222.135.100|:80... connected.

HTTP request sent, awaiting response... 200 OK

Length: 117 [binary/octet-stream]

Saving to: ‘/etc/apt/sources.list.d/ambari.list’

This file is a simple test file with just one line but very important and should not be modified. At the time of agent registration this information will be used.

Now, we can start preparing installation of Ambari server.

First update the apt cache with the below two commands:

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

apt-get update

root@ubuntugnome1604:~# apt-key adv --recv-keys --keyserver keyserver.ubuntu.com B9733A7A07513CAD

Executing: /tmp/tmp.0HOvHxjsoP/gpg.1.sh --recv-keys

--keyserver

keyserver.ubuntu.com

B9733A7A07513CAD

gpg: requesting key 07513CAD from hkp server keyserver.ubuntu.com

gpg: key 07513CAD: public key "Jenkins (HDP Builds) <jenkin@hortonworks.com>" imported

gpg: Total number processed: 1

gpg: imported: 1 (RSA: 1)

root@ubuntugnome1604:~#

root@ubuntugnome1604:~# apt-get update

This will update the apt cache for Ambari server, Ambari agent and Ambari metrics assembly.

Better to check the package information has been updated properly in cache.

root@ubuntugnome1604:~# apt-cache showpkg ambari-server

Package: ambari-server

Versions:

2.5.1.0-159 (/var/lib/apt/lists/public-repo-1.hortonworks.com\_ambari\_ubuntu16\_2.x\_updates\_2.5.1.0\_dists\_Ambari\_main\_binary-amd64\_Packages)

Description Language:

File: /var/lib/apt/lists/public-repo-1.hortonworks.com\_ambari\_ubuntu16\_2.x\_updates\_2.5.1.0\_dists\_Ambari\_main\_binary-amd64\_Packages

MD5: c6d904389bc0d41429b0c7c52796924c

Reverse Depends:

Dependencies:

2.5.1.0-159 - openssl (0 (null)) postgresql (2 8.1) python (2 2.6) curl (0 (null))

Provides:

2.5.1.0-159 -

Reverse Provides:

root@ubuntugnome1604:~# apt-cache showpkg ambari-agent

Package: ambari-agent

Versions:

2.5.1.0-159 (/var/lib/apt/lists/public-repo-1.hortonworks.com\_ambari\_ubuntu16\_2.x\_updates\_2.5.1.0\_dists\_Ambari\_main\_binary-amd64\_Packages)

Description Language:

File: /var/lib/apt/lists/public-repo-1.hortonworks.com\_ambari\_ubuntu16\_2.x\_updates\_2.5.1.0\_dists\_Ambari\_main\_binary-amd64\_Packages

MD5: 65ddc175277c4d4f172223d875012a91

Reverse Depends:

Dependencies:

2.5.1.0-159 - openssl (0 (null)) zlibc (0 (null)) python (2 2.6)

Provides:

2.5.1.0-159 -

Reverse Provides:

root@ubuntugnome1604:~# apt-cache showpkg ambari-metrics-assembly

Package: ambari-metrics-assembly

Versions:

2.5.1.0-159 (/var/lib/apt/lists/public-repo-1.hortonworks.com\_ambari\_ubuntu16\_2.x\_updates\_2.5.1.0\_dists\_Ambari\_main\_binary-amd64\_Packages)

Description Language:

File: /var/lib/apt/lists/public-repo-1.hortonworks.com\_ambari\_ubuntu16\_2.x\_updates\_2.5.1.0\_dists\_Ambari\_main\_binary-amd64\_Packages

MD5: c604a9ec482bcdd63ec5a13eb7e00f6f

Reverse Depends:

Dependencies:

2.5.1.0-159 - python (2 2.6) python-dev (0 (null)) gcc (0 (null))

Provides:

2.5.1.0-159 -

Reverse Provides:

Now, we can run the command for installing Ambari server:

root@ubuntugnome1604:~# apt-get install ambari-server

This command will install ‘PostgreSQL’ database also:

The following additional packages will be installed:

libpq5 postgresql postgresql-9.5 postgresql-client-9.5

postgresql-client-common postgresql-common postgresql-contrib-9.5 sysstat

Suggested packages:

postgresql-doc locales-all postgresql-doc-9.5 libdbd-pg-perl isag

The following NEW packages will be installed:

ambari-server libpq5 postgresql postgresql-9.5 postgresql-client-9.5

postgresql-client-common postgresql-common postgresql-contrib-9.5 sysstat

0 upgraded, 9 newly installed, 0 to remove and 13 not upgraded.

Need to get 702 MB of archives.

After this operation, 772 MB of additional disk space will be used.

Do you want to continue? [Y/n] y

From the above text we can see the installation of ambari server and PostgreSQL will take 772 MB diskspace.

Almost at the end of result script you see the below line which will confirm the successful installation of Ambari server:

Setting up ambari-server (2.5.1.0-159) ...

Next command we need to run is:

root@ubuntugnome1604:~# ambari-server setup

When the set-up script will be running few warnings/confirmations will be there. You need to choose those depending on your scenario.

**Warning 1:** First warning you will get about customizing account for the server:

Customize user account for ambari-server daemon [y/n] (n)? n

Adjusting ambari-server permissions and ownership...

Notice that I have chosen NO(n) in this case. It means that I am not going to add any exclusive user account for running ambari server. I will be using any my system’s user account or the root account.

**Warning 2:** This is about Java or specifically JDK

Checking JDK...

[1] Oracle JDK 1.8 + Java Cryptography Extension (JCE) Policy Files 8

[2] Oracle JDK 1.7 + Java Cryptography Extension (JCE) Policy Files 7

[3] Custom JDK

==============================================================================

Enter choice (1): 3

Here I have chosen 3 because I would like to use my existing JDK. Then the below lines will appear:

WARNING: JDK must be installed on all hosts and JAVA\_HOME must be valid on all hosts.

WARNING: JCE Policy files are required for configuring Kerberos security. If you plan to use Kerberos,please make sure JCE Unlimited Strength Jurisdiction Policy Files are valid on all hosts.

Path to JAVA\_HOME: /usr/local/jdk1.8.0\_144

Validating JDK on Ambari Server...done.

Completing setup...

Notice that I have just provided my existing JDK location (JAVA\_HOME directory)

**Warning 3:** Third warning will ask you whether to set up advanced database (PostgreSQL) configuration. If you want to use some other database service other than the default one (PostgreSQL) then choose YES(y). I have chosen NO(n).

Configuring database...

Enter advanced database configuration [y/n] (n)? n

Configuring database...

Default properties detected. Using built-in database.

The setup process will complete with the below lines:

Adjusting ambari-server permissions and ownership...

Ambari Server 'setup' completed successfully.

root@ubuntugnome1604:~#

Last step is to run Ambari server. This is the last step before we go with Ambari GUI.

root@ubuntugnome1604:~# ambari-server start

.

.

.

DB configs consistency check: no errors and warnings were found.

Ambari Server 'start' completed successfully.

root@ubuntugnome1604:~#

As the server is running we can just go to browser and type:

[http://ubuntugnome1604:8080](http://ubuntugnome1604:8080/)

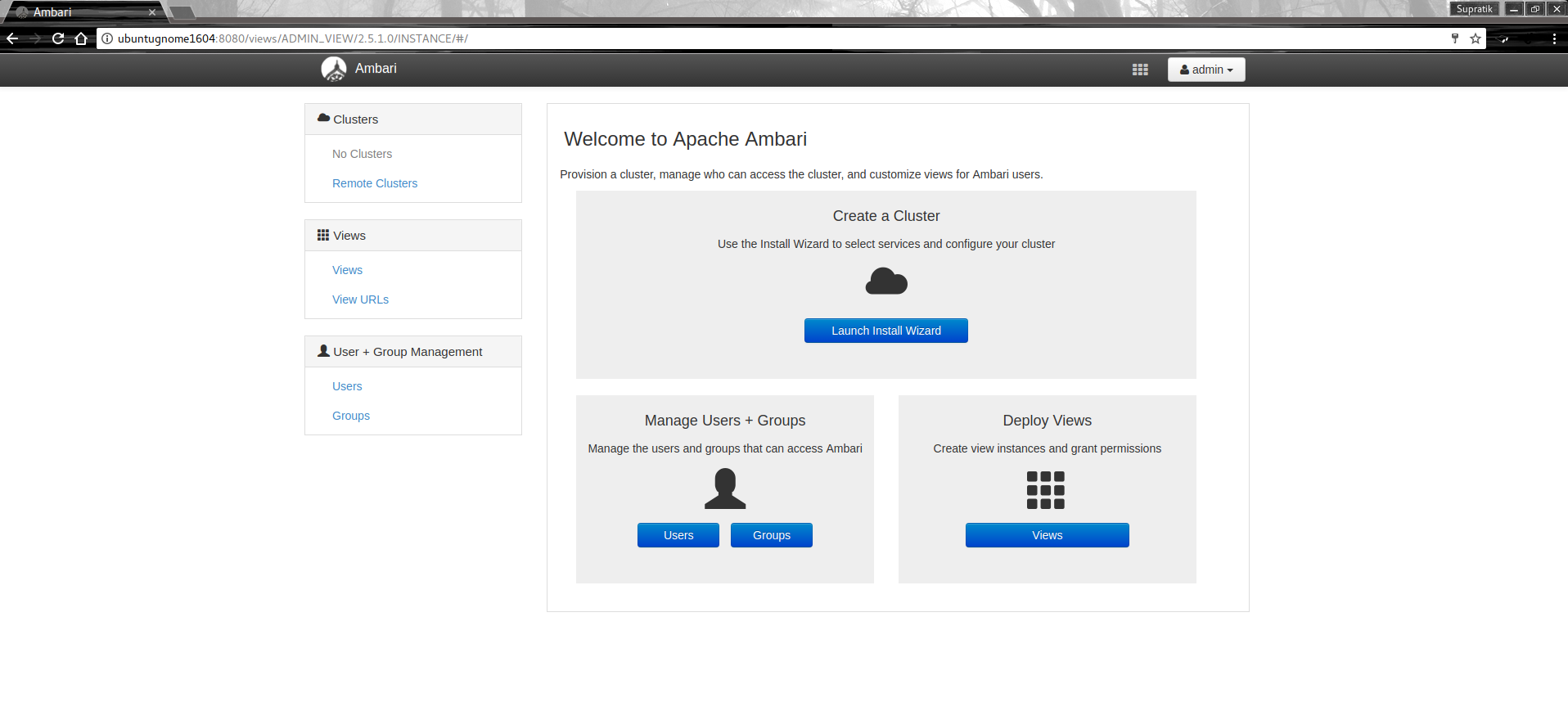
In general **http://<FQDN>:8080**

Username: admin

Password: admin

Now, you are logged in to Ambari GUI:

Below is the screen shot of the first look of Ambari GUI

Now, we will prepare a cluster. For that we click on ‘Launch Install Wizard’