Configuring Passwordless SSH for root user

Execute the below commands from the hadoop02 Master Node: Just press "Enter" when prompted for a passphrase. Two files will be created in the folder **/root/.ssh**

```
[root@hadoop02 ~]# ssh-keygen
```

Just press "Enter" when prompted for a passphrase. Two files will be created in the folder **/root/.ssh**

```
[root@hadoop02 .ssh] #  ls -al
id_rsa  id_rsa.pub
```

Copy the contents of id_rsa.pub to authorized_keys and copy both the files to other nodes to all the remote-host's .ssh/authorized_key.
Copy the SSH Public Key (id_rsa.pub) to the root account on your target hosts.

```
[root@hadoop02 .ssh] # cat .ssh/id_rsa.pub | ssh root@192.168.192.18 'cat >>
.ssh/authorized_keys'
```

Disable firewall / iptables on all the cluster hosts

Execute the below command on all nodes to disable the firewall, type the following command as the root user to disable firewall for IPv6:

```
[root@hadoop02 ~] #  service ip6tables stop
ip6tables: Setting chains to policy ACCEPT: filter                [OK]
ip6tables: Flushing firewall rules:                [OK]
ip6tables: Unloading modules:                [OK]
```

Disable on boot start-up of iptables

```
[root@hadoop02 ~] # chkconfig ip6tables off
[root@hadoop02 ~] # service iptables status
iptables: Firewall is not running.
```

IPv4

```
[root@hadoop02 etc] # service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables: [OK]
```

Stop the iptables services

```
[root@hadoop02 etc] # service iptables stop
iptables: Setting chains to policy ACCEPT: filter                [OK]
iptables: Flushing firewall rules:                [OK]
iptables: Unloading modules:                [OK]
```

```
[root@hadoop02 etc] # chkconfig iptables off
```

Turn the iptables off

```
[root@hadoop02 etc] # chkconfig iptables off
```

Configure the sshd daemon to start on boot

```
[root@hadoop02 ~] # chkconfig --level 345 sshd on
[root@hadoop02 ~] # service sshd restart
Stopping sshd:                                [  OK  ]
Starting sshd:                                [  OK  ]
```

Disable SELinux

SELinux must be disabled for Ambari to function. To temporarily disable SELinux, run the following command on each host in your cluster:

```
[root@hadoop02 ~] # setenforce 0
```

Permanently disabling SELinux so that on system reboot it does not restart edit the SELinux config and set SELINUX to disabled. on each host:

```
root@hadoop02 ]# vi /etc/selinux/config
```

Add the below lines

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#    enforcing - SELinux security policy is enforced.
#    permissive - SELinux prints warnings instead of enforcing.
#    disabled - SELinux is fully disabled.
SELINUX=disabled
# SELINUXTYPE= type of policy in use. Possible values are:
#     targeted - Only targeted network daemons are protected.
#     strict - Full SELinux protection.
SELINUXTYPE=targeted
```

Disable Transparent Huge Pages (THP)

This must be disabled on all the hosts otherwise the Ambari install will fail

```
[root@hadoop02 ~] # echo never > /sys/kernel/mm/redhat_transparent_hugepage/enabled
[root@hadoop02 ~] # echo never > /sys/kernel/mm/redhat_transparent_hugepage/defrag
```

To disable or make these changes persistent across reboots I add this to the bottom of my vi /etc/rc.local

```
#disable THP at boot time
if test -f /sys/kernel/mm/redhat_transparent_hugepage/enabled; then
echo never > /sys/kernel/mm/redhat_transparent_hugepage/enabled
fi
if test -f /sys/kernel/mm/redhat_transparent_hugepage/defrag; then
echo never > /sys/kernel/mm/redhat_transparent_hugepage/defrag
fi
```

To validate THP is disabled, I run the below three commands, or any variant you choose from here .

```
[root@hadoop02 ~] # cat /sys/kernel/mm/redhat_transparent_hugepage/defrag
      always madvise [never]
[root@hadoop02 ~] # cat /sys/kernel/mm/redhat_transparent_hugepage/enabled
       always madvise [never]
```

Configure the NTPD services

You must setup the NTPD server on CentOS to successfully implement Ambari follow the below steps to install and start the NTPD server. As the root user.

```
[root@hadoop02 ~] # yum install ntp ntpdate
.....
......
Complete!
```

Turn on the service:

```
[root@hadoop02 ~] # chkconfig ntpd on
```

Synchronize the system clock with 0.pool.ntp.org server

```
[root@hadoop02 bin] # ntpdate pool.ntp.org
 2 Jan 21:42:49 ntpdate[5101]: 31.3.135.236 rate limit response from server.
 2 Jan 21:43:56 ntpdate[5101]: step time server 212.51.144.44 offset 67.520940 sec
```

Check the NTPD services configure it start at system boot

```
[root@hadoop02 bin] # chkconfig --list ntpd
ntpd            0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

Start the NTP server. The following will continuously adjusts system time from upstream NTP server. No need to run ntpdate:

```
[root@hadoop02 ~]# /etc/init.d/ntpd start or
[root@hadoop02 ~]#  service ntpd start
Starting ntpd:                                             [  OK  ]
```