INSTALLING HADOOP

DOWNLOAD CENTOS 7 64-BIT OPERATING SYSTEM

The minimal install media si sufficient for our purposes and will be what is assumed throughout. If you want to directly interact with the console, then you will want to use an install that includes either the Gnome or KDE GUI environment.

DURING INSTALL OF CENTOS

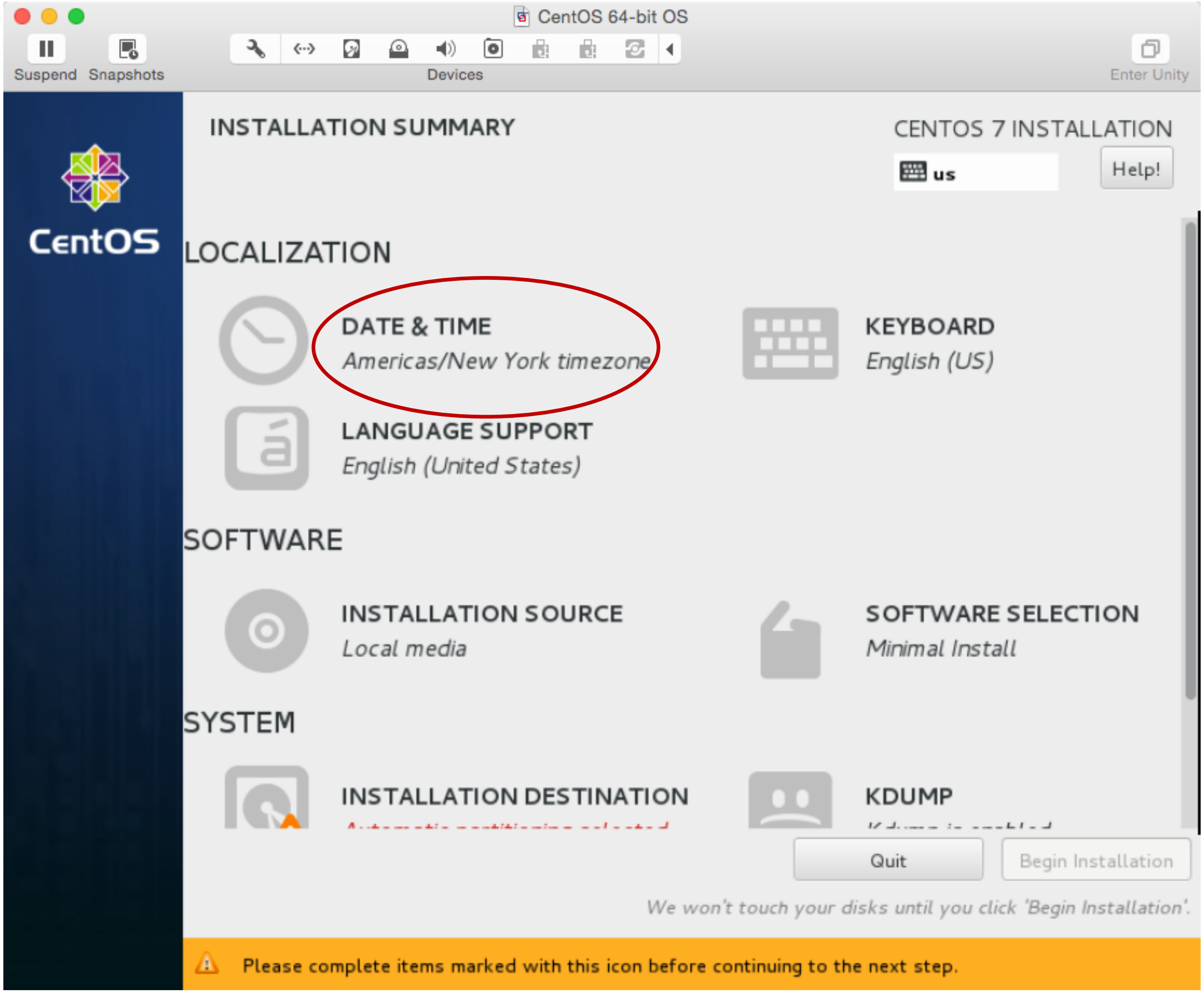
On the Installation Summary Page

1. Turn on networking under Network & Host Name
2. Turn on ntp under Date & Time
3. Select an Installation Destination

4. Note the Software Selection, which should be Minimal Install

5. Disable kdump (Optional)

The three figures above first show the top part and the bottom part of the installation summary page before making any changes. The third figure below shows the page after these changes have been made.



INSTALLATION SUMMARY

CENTOS 7 INSTALLATION

LOCALIZATION

DATE & TIME
Americas/New York timezone

KEYBOARD
English (US)

LANGUAGE SUPPORT
English (United States)

SOFTWARE

INSTALLATION SOURCE
Local media

SOFTWARE SELECTION
Minimal Install

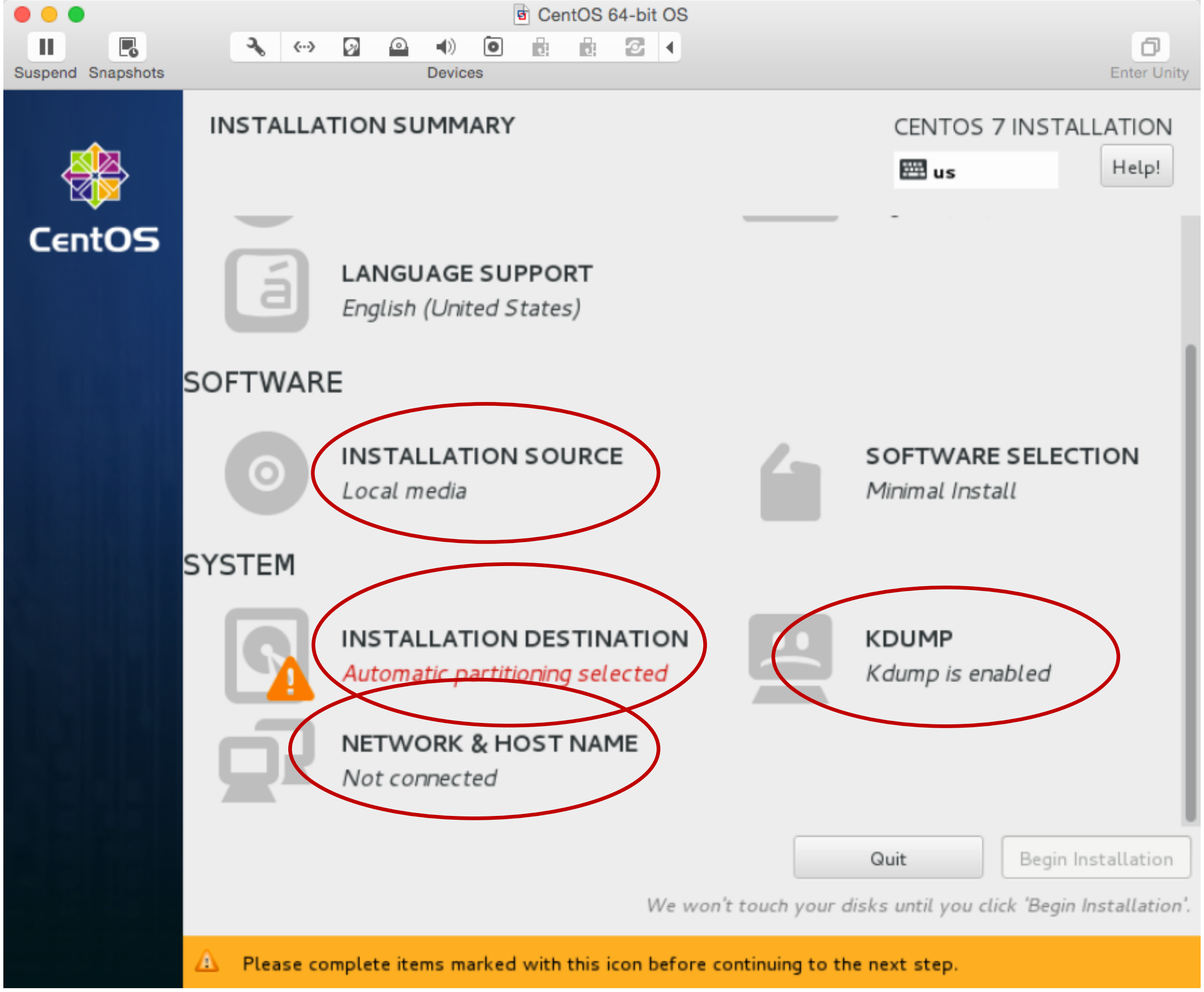
SYSTEM

INSTALLATION DESTINATION
Automatic partitioning selected

KDUMP
Kdump is enabled

We won't touch your disks until you click 'Begin Installation'.

Please complete items marked with this icon before continuing to the next step.



INSTALLATION SUMMARY

CENTOS 7 INSTALLATION

LANGUAGE SUPPORT
English (United States)

SOFTWARE

INSTALLATION SOURCE
Local media

SOFTWARE SELECTION
Minimal Install

SYSTEM

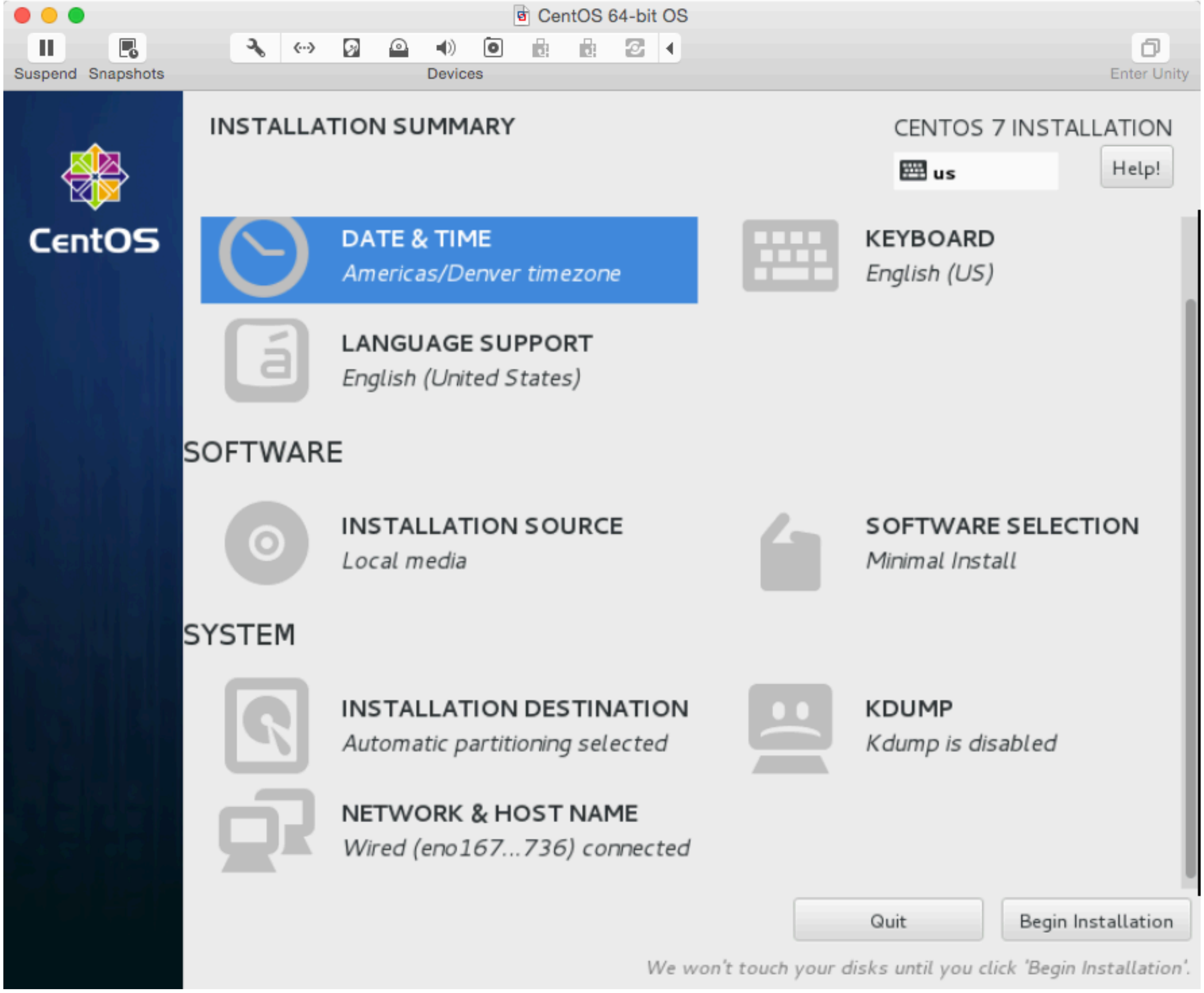
INSTALLATION DESTINATION
Automatic partitioning selected

KDUMP
Kdump is enabled

NETWORK & HOST NAME
Not connected

We won't touch your disks until you click 'Begin Installation'.

Please complete items marked with this icon before continuing to the next step.



INSTALLATION SUMMARY

CENTOS 7 INSTALLATION

DATE & TIME
Americas/Denver timezone

KEYBOARD
English (US)

LANGUAGE SUPPORT
English (United States)

SOFTWARE

INSTALLATION SOURCE
Local media

SOFTWARE SELECTION
Minimal Install

SYSTEM

INSTALLATION DESTINATION
Automatic partitioning selected

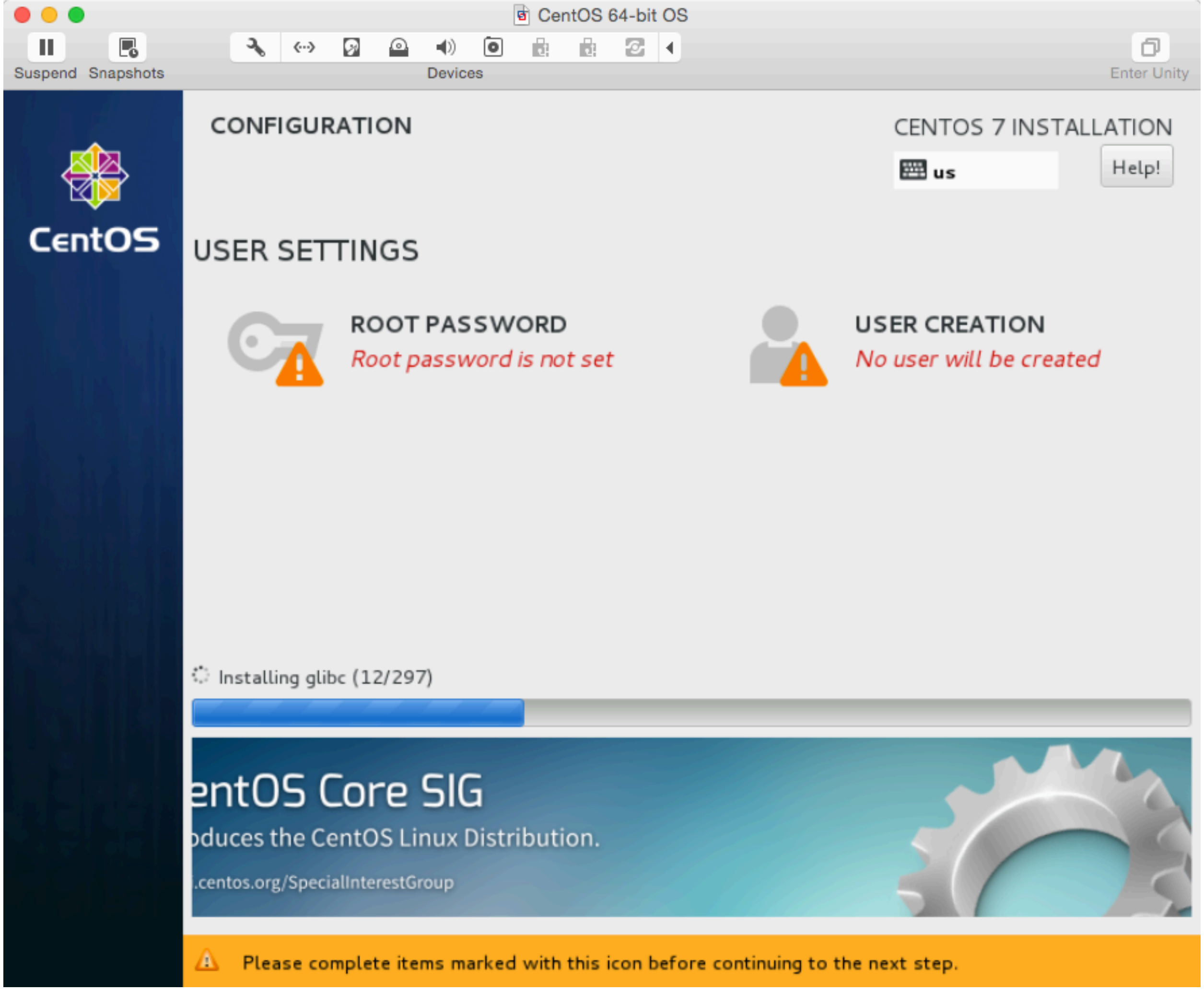
KDUMP
Kdump is disabled

NETWORK & HOST NAME
Wired (eno167...736) connected

We won't touch your disks until you click 'Begin Installation'.

On the Configuration page

1. Set a root password
2. Create a user for yourself



Root password is not set

No user will be created

Installing glibc (12/297)

Please complete items marked with this icon before continuing to the next step.

Login to Server

Login into VM console using the user credentials you created during the OS install. At this point you have access to the server through the VM console. You may continue to work from here. However, you may be better served by gaining access to the server secure shell, which is invoked using an ssh client.

If you wish to use a remote shell to login to your server

First, find the IP address of your virtual machine

1. Login to the VM console
2. Find the IP address of the virtual machine

```
# ip addr
```

i. You should first see the IP address of the so called loopback device 127.0.0.1
ii. Second you should see a private IP address like 192.168.x.y or 172.a.c.b, or 10.r.s.t, where a, b, c, r, s, t, x, and y are numbers from 0-255.
iii. See the example below. The IP address for the server is highlighted in yellow. It is 10.2.2.133.

```
[dhart@localhost ~]$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eno16777736: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
state UP qlen 1000
    link/ether 00:0c:29:1a:4b:2e brd ff:ff:ff:ff:ff:ff
    inet 10.2.2.133/24 brd 10.2.2.255 scope global dynamic eno16777736
       valid_lft 980sec preferred_lft 980sec
    inet6 fe80::20c:29ff:fe1a:4b2e/64 scope link
       valid_lft forever preferred_lft forever
[dhart@localhost ~]$
```

Remember the IP address

Next, open ssh client software, such as Putty on Window. Alternatives on Windows include the Bitvise SSH client as well as the ssh client that is installed as part of Cygwin. An ssh client is also available to you if you are using the GitHub software. Finally, you could use MobaXterm. MobaXterm provides an ssh client and provides the functionality to run X applications over ssh. Meaning you can run the Linux GUI applications on your Windows computer. If you installed the minimal install CentOS system, then the X windows system was NOT installed.

If you are using OS X, the the ssh client software is part of the operating system. You can use it from the terminal prompt. If you are interested in running X over ssh on OS X, see the XQuartz project.

The basic format of the ssh command is

```
# ssh <IP addr>,
```

So, for example, to access the server at 10.2.2.133 you woule enter

```
# ssh 10.2.2.133
```

This assumes that you are logging into the server with the same userid. If you need to login as another user then you need to add a parameter to your ssh command. The form of the command is

```
# ssh –l <userid> <IP addr>,
```

and

```
# ssh –l doug 10.2.2.133
```

will login to the server 10.2.2.133 with userid doug.

Install OS Updates

Become root and install updates using the yum utility

```
# su
# yum update
```

Prepare to install Java

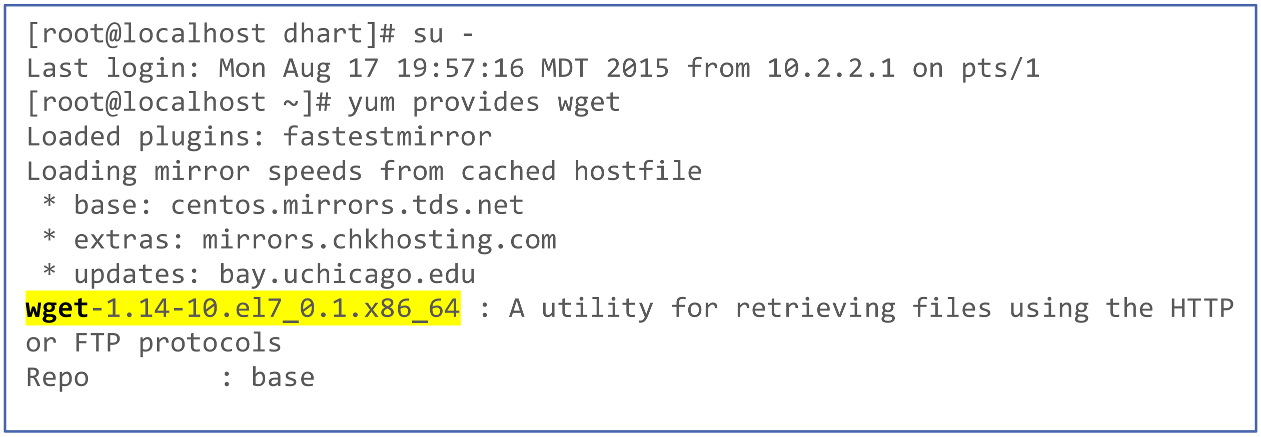
The Linux utility wget is used to download files from the Internet. It is not part of the minimal CentOS install. The following steps will will download and install the wget utility.

Become root (if needed)

```
# su
```

Determine what package contains wget

```
# yum provides wget
```



```
[root@localhost dhart]# su -
Last login: Mon Aug 17 19:57:16 MDT 2015 from 10.2.2.1 on pts/1
[root@localhost ~]# yum provides wget
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
 * base: centos.mirrors.tds.net
 * extras: mirrors.chkhosting.com
 * updates: bay.uchicago.edu
wget-1.14-10.el7_0.1.x86_64 : A utility for retrieving files using the HTTP
or FTP protocols
Repo        : base
```

To install wget use the yum command with the install option

```
# yum install wget
```

Download Java

Become root (if necessary)

Download and install Java in the /opt directory.

```
# cd /opt

# wget --no-cookies --no-check-certificate --header "Cookie:
gpw_e24=http%3A%2F%2Fwww.oracle.com%2F; oraclelicense=accept-
securebackup-cookie" http://download.oracle.com/otn-pub/java/jdk/8u60-
b27/jdk-8u60-linux-x64.tar.gz

# tar xzf jdk-8u60-linux-x64.tar.gz
```

After performing the steps above and then listing the contents of the directory you should see something like this –

```
[root@localhost opt]# ls -la
total 177000
drwxr-xr-x.  3 root root        56 Aug 18 11:54 .
dr-xr-xr-x. 17 root root      4096 Aug 17 18:58 ..
drwxr-xr-x.  8   10  143      4096 Aug  4 13:32 jdk1.8.0_60
-rw-r--r--.  1 root root 181238643 Aug  6 17:32 jdk-8u60-linux-x64.tar.gz
```

Install Java using Alternatives

Alternatives is a tool for managing version control of software applications, like java which changes update frequently.

```
# cd /opt/jdk1.8.0_60/
# alternatives --install /usr/bin/java java /opt/jdk1.8.0_60/bin/java 2
# alternatives --config java
```

```
[root@localhost jdk1.8.0_60]# alternatives --install /usr/bin/java java
/opt/jdk1.8.0_60/bin/java 2
[root@localhost jdk1.8.0_60]# alternatives --config java

There is 1 program that provides 'java'.

  Selection    Command
-----------------------------------------------
*+ 1           /opt/jdk1.8.0_51/bin/java

Enter to keep the current selection[+], or type selection number:
[root@localhost jdk1.8.0_60]#
```

```
# alternatives --install /usr/bin/jar jar /opt/jdk1.8.0_60/bin/jar 2
# alternatives --install /usr/bin/javac javac
/opt/jdk1.8.0_60/bin/javac 2
# alternatives --set jar /opt/jdk1.8.0_60/bin/jar
# alternatives --set javac /opt/jdk1.8.0_60/bin/javac
```

CREATE THE HADOOP USER

Linux is a multiuser operating system. Administering the system requires that you remain aware of what user you are. At present you should have both a personal userid and a root id. The root id is what gives you administrative privileges for your machine.

The switch user or su command is what allows you to switch your role from one user to another. The exit command will return you to userid you where before you executed the su command. You may have noticed that sometimes su is executed with a "-" and sometimes without the dash. When you use the dash with su you will move the the the home directory for that user. Additionally, any configurations that exist in the .bashrc file for that user will be executed.

Become root (if needed)

```
# su -

# useradd hadoop
# passwd hadoop
```

When prompted for a password for the hadoop user you will need to select one. The password for my hadoop user is hadoop. If this was a installation for deployment, then it would need to be more secure. But I do not want to take on password management at this time.

Become the hadoop user and create the ability for password-less login using ssh-keygen

```
# su - hadoop
$ ssh-keygen -t rsa
$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
$ chmod 0600 ~/.ssh/authorized_keys
```

```
[root@localhost jdk1.8.0_60]# useradd hadoop
[root@localhost jdk1.8.0_60]# passwd hadoop
Changing password for user hadoop.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@localhost jdk1.8.0_60]# su - hadoop
[hadoop@localhost ~]$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hadoop/.ssh/id_rsa):
Created directory '/home/hadoop/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/hadoop/.ssh/id_rsa.
Your public key has been saved in /home/hadoop/.ssh/id_rsa.pub.
The key fingerprint is:
26:e5:ef:5f:ec:d1:45:f2:4f:3a:3c:08:6e:5d:57:38
hadoop@localhost.localdomain
The key's randomart image is:
+--[ RSA 2048]----+
|                 |
|               . |
|        .     E o|
|       o       =.|
|      . S .   . =|
|       o o o = =o|
|          + o O o|
|         o   o + |
|          ... .  |
+-----------------+
[hadoop@localhost ~]$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
[hadoop@localhost ~]$ chmod 0600 ~/.ssh/authorized_keys
```

Make sure that he password-less login works correctly for the hadoop user.

```
# ssh localhost
# exit
```

EDITING TEXT FILES ON YOUR SERVER

Up until now it has not been necessary to edit text files on your server. In what follows it will be necessary to edit various configuration files. Since we are working with a server without a graphical user interface the text editors that we will use may seem archaic compared to what you have seen. However, they have the benefit of low network overhead making it possible to work on servers anywhere on the Internet.

The vi, or vim, editor is a text editor that has been part of Unix and Linux for a very long time. It is now part of the operating system install. It is invoked with the "vi" command followed by the name of the file. This editor is somewhat cryptic in its interface, in that no information about commands for interacting with the editor are displayed. However, it is a quite powerful editor.

A second choice is the nano editor. It provides basic text editing functionality which will be sufficient for our purposes. It, however, is not installed along with the operating system. It must be downloaded and installed. This is easily done with the yum update command. First we will determine what package contains the nano editor and then install it. Software installation must be done as root so first execute su to become root

```
# su
# yum provides nano
```

The output from yum provides nano indicates that nano exists in its own package. So to execute

```
# yum install nano
```

to install the editor.

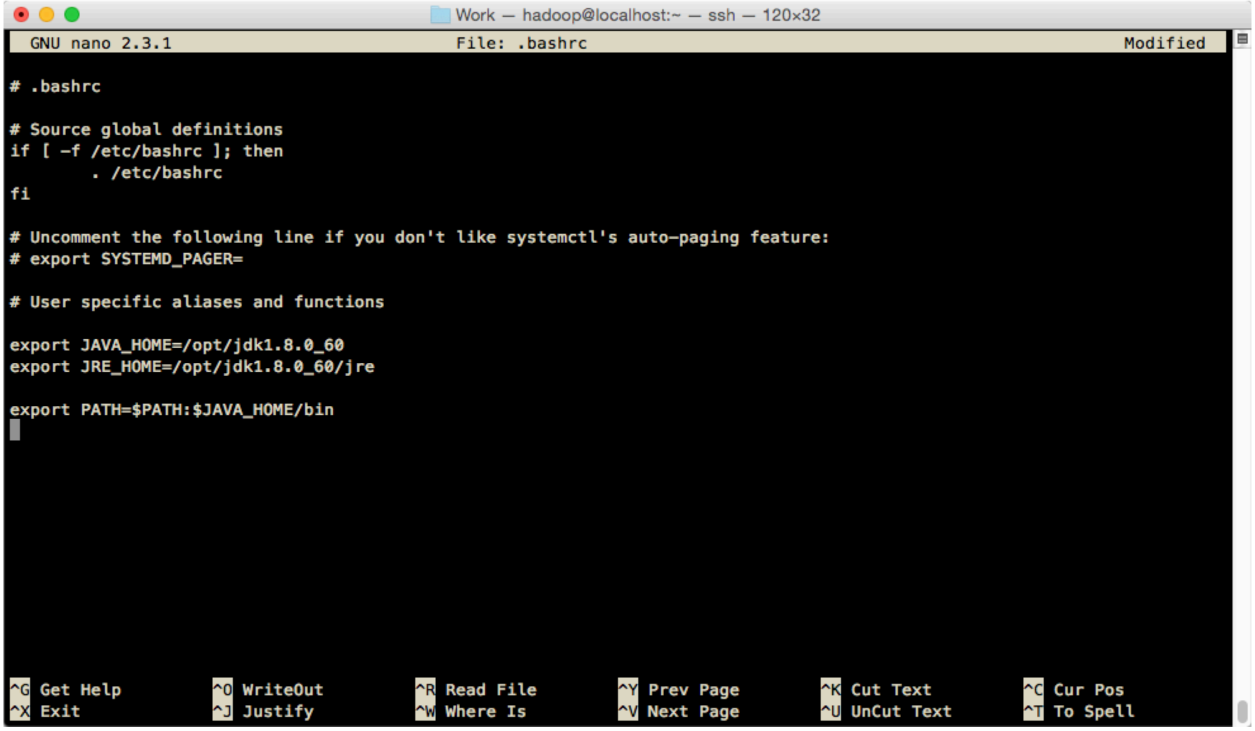
CONFIGURE ENVIRONMENT VARIABLES FOR THE HADOOP USER

Add environment variables to .bashrc for the hadoop user.

```
export JAVA_HOME=/opt/jdk1.8.0_60
export JRE_HOME=/opt/jdk1.8.0_60/jre

export PATH=$PATH:$JAVA_HOME/bin
```

The image below shows what my .bashrc file looks like after putting these three lines in it.



```
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
        . /etc/bashrc
fi

# Uncomment the following line if you don't like systemctl's auto-paging feature:
# export SYSTEMD_PAGER=

# User specific aliases and functions

export JAVA_HOME=/opt/jdk1.8.0_60
export JRE_HOME=/opt/jdk1.8.0_60/jre

export PATH=$PATH:$JAVA_HOME/bin
```

Finally cause the changes in the .bashrc to be executed. (You only need to do this when you make changes to your .bashrc file. The contents of the .bashrc file are execute whenever you login to it

```
# source .bashrc
```

You can check to see that your PATH is configured correctly by executing the printenv command

```
# printenv PATH
```

My output looks like

```
[hadoop@localhost ~]$ nano .bashrc
[hadoop@localhost ~]$ source .bashrc
[hadoop@localhost ~]$ printenv PATH
/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/home/hadoop/.local
/bin:/home/hadoop/bin:/opt/jdk1.8.0_60/bin:/opt/jdk1.8.0_60/bin
[hadoop@localhost ~]$
```