Hostname

node1

password

sohr666888314

superuser

sohr

Do the following node by node by hands

ssh node1

sudo visudo

Add the following line to the very end of this file

sohr ALL=(ALL) NOPASSWD: ALL

checkout /etc/hosts

Change mount point

Firstly get to know the partition reference. Run

sudo blkid

to get the UUID of the parition.

Next you need to modify your /etc/fstab file to point to the desired mount point. Run sudo xdg-open /etc/fstab and add a line or modify the line referencing the partition.

An example line:

UUID="xxxx-xxxx" /media/Radi ext4 defaults,user,auto 0 1

Then reboot your computer or run sudo mount -a

UUID=c491c72c-e5fc-48b8-8cc1-5a5187bdc0ea /ssd2t ext4 defaults,user,auto 0 0

Network File System (NFS) setup

Headnode

Sudo apt-get install nfs-server

Sudo blkid to see UUID of disk/folder

Sudo vim /etc/exports, add two lines

/ssd2t \*(rw,sync)

/hdd4t \*(rw,sync)

Sudo service nfs-kernel-server restart

Compute nodes

Sudo apt-get install nfs-client

Sudo vim /etc/fstab

Add lines

headnode:/ssd2t /ssd2t nfs

headnode:/hdd4t /hdd4t nfs

sudo mkdir /ssd2t

sudo mkdir /hdd4t

sudo mount –a

Boot into command line mode

Yes you can. As described here (ubuntuhandbook.org - Boot into text console ubuntu) you need to edit /etc/default/grub to have the next boot end up in text mode. In summary you will set these parameters:

GRUB\_CMDLINE\_LINUX="text"

GRUB\_TERMINAL=console

After saving changes you need to run:

sudo update-grub

sudo apt-get install mailutils

sudo apt-get update

sudo apt-get upgrade

sudo apt-get update

sudo apt-get autoremove

sudo apt-get install vim

sudo apt-get install build-essential

sudo apt-get install gfortran

sudo apt-get install openssh-server

sudo apt-get install mpich

sudo apt-get install openmpi-bin

sudo apt-get install libboost-all-dev

anaoncda

/opt/anaconda/anaconda3

headnode

ssd2t -> ssd 2T in size

mount as /ssd2t

hdd4t -> hard drive disk 4T in size

mount as /hdd4t

set up slurm, set up shared storage first ->nfs

lscpu

<https://github.com/mknoxnv/ubuntu-slurm>

<https://github.com/CU-Boulder-Phd-work/ubuntu-slurm>

sudo apt-get install autoconf

sudo apt-get install libffi-dev

sudo apt-get update

sudo apt-get install git gcc make ruby ruby-dev libpam0g-dev libmariadb-client-lgpl-dev libmysqlclient-dev

sudo gem install fpm

cd /ssd2t/storage

git clone <https://github.com/mknoxnv/ubuntu-slurm.git>

Customize slurm.conf with your slurm controller and compute node hostnames:

$ vi ubuntu-slurm/slurm.conf

ControlMachine=slurm-ctrl

NodeName=linux1 (you can specify a range of nodes here, for example: linux[1-10])

Install munge

MUNGE (MUNGE Uid 'N' Gid Emporium) is an authentication service for creating and validating credentials. https://dun.github.io/munge/

Ubuntu 16.04

$ sudo apt-get install libmunge-dev libmunge2 munge

$ sudo systemctl enable munge

$ sudo systemctl start munge

Test munge

$ munge -n | unmunge | grep STATUS

STATUS: Success (0)

Install MariaDB for Slurm accounting

MariaDB is an open source Mysql compatible database. https://mariadb.org/

In the following steps change the DB password "slurmdbpass" to something secure.

Ubuntu 16.04

$ sudo apt-get install mariadb-server

$ sudo systemctl enable mysql

$ sudo systemctl start mysql

$ sudo mysql -u root

create database slurm\_acct\_db;

create user 'slurm'@'localhost';

set password for 'slurm'@'localhost' = password('slurmdbpass');

grant usage on \*.\* to 'slurm'@'localhost';

grant all privileges on slurm\_acct\_db.\* to 'slurm'@'localhost';

flush privileges;

exit

Download, build, and install Slurm

Download tar.bz2 from https://www.schedmd.com/downloads.php to ~/Downloads

$ cd ~/Downloads

$ wget https://download.schedmd.com/slurm/slurm-17.02.9.tar.bz2

$ tar xvjf slurm-17.02.9.tar.bz2

$ cd slurm-17.02.9

$ ./configure --prefix=/tmp/slurm-build --sysconfdir=/etc/slurm --enable-pam --with-pam\_dir=/lib/x86\_64-linux-gnu/security/

$ make

$ make contrib

$ sudo make install

$ cd ..

$ fpm -s dir -t deb -v 1.0 -n slurm-17.02.9 --prefix=/usr -C /tmp/slurm-build .

$ sudo dpkg -i slurm-17.02.9\_1.0\_amd64.deb

$ sudo useradd slurm

$ sudo mkdir -p /etc/slurm /var/spool/slurm/ctld /var/spool/slurm/d /var/log/slurm

$ sudo chown slurm /var/spool/slurm/ctld /var/spool/slurm/d /var/log/slurm

Copy into place config files from this repo which you've already cloned into /storage

$ cd /ssd2t/storage

$ sudo cp ubuntu-slurm/slurmdbd.service /etc/systemd/system/

$ sudo cp ubuntu-slurm/slurmctld.service /etc/systemd/system/

Edit /storage/ubuntu-slurm/slurm.conf and replace AccountingStoragePass=slurmdbpass with AccountingStoragePass=/var/run/munge/munge.socket.2

$ sudo cp ubuntu-slurm/slurm.conf /etc/slurm/

Edit /storage/ubuntu-slurm/slurmdbd.conf and replace StoragePass=slrumdbpass with the DB password you used

in the above SQL section.

$ sudo cp ubuntu-slurm/slurmdbd.conf /etc/slurm/

$ sudo systemctl daemon-reload

$ sudo systemctl enable slurmdbd

$ sudo systemctl start slurmdbd

$ sudo systemctl enable slurmctld

$ sudo systemctl start slurmctld

Install slurm and associated components on a compute node.

Install munge

MUNGE (MUNGE Uid 'N' Gid Emporium) is an authentication service for creating and validating credentials. https://dun.github.io/munge/

$ sudo apt-get update

$ sudo apt-get install libmunge-dev libmunge2 munge

$ scp slurm-ctrl:/etc/munge/munge.key /etc/munge/

$ sudo chown munge:munge /etc/munge/munge.key

$ sudo chmod 400 /etc/munge/munge.key

Ubuntu 16.04

$ sudo systemctl enable munge

$ sudo systemctl restart munge