

Head Node

Make sure that you have completed the section on Precursor Steps and Storage. Key parts of that are necessary for you to continue on this. If you have issues, please let an instructor know to get you to the correct place.

1. Configure Puppet as a basic master server on the head node as well as configure Puppet on the compute nodes.

```
$ sudo yum install puppet-server
```

Make sure to hit yes.

```
$ sudo clush -w @compute yum install -y puppet
```

```
$ sudo systemctl enable puppetmaster.service
```

Add "puppet" entry to /etc/hosts

- a. "puppet" should point to "master" (i.e., append "puppet" to line with "master" in it. Use a text editor like Vim.

```
$ sudo vim /etc/hosts
```

- b. Copy to all compute nodes.

```
$ sudo clush -w @compute --copy /etc/hosts
```

- c. Edit /etc/puppet/puppet.conf with the appropriate DNS / hostfile names, dns_alt_names = puppet,master,node01 in [main] section.

- d. Generate the certificate for the puppet server and then hit CTRL + C

```
$ sudo puppet master --verbose --no-daemonize
```

CTRL+C after certificate generated

```
$ sudo systemctl start puppetmaster.service
```

```
$ sudo systemctl status puppetmaster.service
```

```
$ sudo systemctl is-active puppetmaster.service
```

- e. Start the puppet agents on the compute nodes:

```
$ sudo clush -w @compute systemctl enable puppet.service
```

```
$ sudo clush -w @compute systemctl start puppet.service
```

- f. Verify Puppet master is running and the compute node Puppet agents are running

```
$ sudo systemctl status puppetmaster.service
```

```
$ sudo clush -w @compute -b systemctl is-active puppet.service
```

```
$ sudo clush -w @compute systemctl is-active puppet.service
```

Note order of output

- g. Accept new Puppet Agent certs

```
$ sudo puppet cert list
```

```
$ sudo puppet cert sign --all
```

- h. Configure compute nodes as clients in `/etc/puppet/manifests/site.pp` (see `~/build-a-cluster/etc/puppet/manifests/site.pp`)

```
$ sudo cp \ ~/build-a-cluster/etc/puppet/manifests/site.pp \  
/etc/puppet/manifests/site.pp
```

```
$ cat /etc/puppet/manifests/site.pp
```

- i. Run basic Puppet test

```
$ sudo mkdir -p \ /etc/puppet/modules/test/{manifests,files}
```

```
$ sudo cp  
~/build-a-cluster/etc/puppet/modules/test/manifests/init.pp  
/etc/puppet/modules/test/manifests/init.pp
```

```
$ sudo clush -w node02 puppet agent -t
```

If this fails.. try and see what setting may have been wrong. Then run:

```
$ bash \ ~/build-a-cluster/scripts/hostname_setup.sh
```

```
$ bash ~/build-a-cluster/scripts/fix_puppet.sh
```

Retry parent step.

2. Configure a “Yum” repository to install software later which can be distributed in RPM format from our local master node.

- a. Install an Apache web server:

```
$ sudo yum install -y httpd
```

- b. Install the yum-utils and createrepo packages on master node:

```
$ sudo yum install -y yum-utils createrepo
```

- c. Navigate to the default web root location: /var/www/html and create a directory "hpc-repo"

```
$ cd /var/www/html
```

```
$ sudo mkdir -p hpc-repo/Packages
```

```
$ cd hpc-repo
```

```
$ sudo createrepo $(pwd)
```

- d. Start the Apache web server and make sure it will run on reboot

```
$ sudo systemctl start httpd.service
```

```
$ sudo systemctl enable httpd.service
```

Create a yum repo configuration file to house the packages in '/etc/yum.repos.d/hpc.repo' with the following content.

```
[hpc]
name=LCI HPC Build-A-Cluster
baseurl=http://master/hpc-repo/
enabled=1
gpgcheck=0
```

```
$ sudo yum repolist
```

```
$ sudo clush -b -w @compute --copy \ /etc/yum.repos.d/hpc.repo
```

```
$ sudo clush -b -w @compute yum makecache
```

3. Add a cluster user (who is not the ec2-user)

```
$ sudo -i
```

```
# useradd -b /dhome -m lcistudent
```

```
# passwd lcistudent
```

```
# clush -bw @compute ls /dhome
```

```
# exit
```

```
$ sudo su - lcistudent
```

```
$ ssh-keygen
```

Follow the on screen instructions

```
$ cat ~/.ssh/id_rsa.pub > ~/.ssh/authorized_keys
```

```
$ chmod 600 ~/.ssh/authorized_keys
```

```
$ exit
```

4. Installation of the Lmod modules system on master node (using clush)

```
$ mkdir -p ~/sw
```

```
$ cd ~/sw
```

```
$ sudo yum groupinstall -y Development\ tools
```

```
$ sudo yum install -y wget
```

```
$ wget -O Lmod-7.4.tar.bz2 \  
https://sourceforge.net/projects/lmod/files/Lmod-7.4.tar.bz2/download
```

```
$ tar xvf Lmod-7.4.tar.bz2
```

```
$ cd Lmod-7.4
```

```
$ sudo mkdir -p /apps/opt
```

```
$ sudo clush -w @all yum install -y \  
    tcl lua lua-posix lua-filesystem \  
    lua-lpeg lua-term lua-json
```

```
$ ./configure \  
    --prefix=/apps/opt \  
    --with-module-root-path=/apps/opt/lmod/mf \  
    --with-shortTime=86400 --with-siteName=LCI
```

```
$ make && sudo make install
```

```
$ sudo clush -w @all --copy ~/build-a-cluster/etc/profile.d/z0*sh \  
--dest /etc/profile.d/
```

```
$ sudo mkdir -p /apps/opt/lmod/mf/Core/lci
$ sudo touch /apps/opt/lmod/mf/Core/lci/1.0.lua
$ exec bash
$ module avail
```

5. Set up profile scripts (i.e., Puppet configuration management)

- a. See `~/build-a-cluster/etc/profile.d/z*_modules.*sh`
- b. Make sure "root" doesn't get modules!!!
- c. Copy files from the Git repository to puppet repo

```
$ sudo cp -R ~/build-a-cluster/etc/puppet/modules/lmod
/etc/puppet/modules/
```

```
$ sudo vim /etc/puppet/manifests/site.pp
```

Add "include lmod"

```
$ sudo clush -w @compute puppet agent -t
```

```
$ sudo clush -w @compute ls /etc/profile.d/z0*
```

```
$ sudo clush -w @compute rm -f /etc/profile.d/z0*sh
```

```
$ sudo clush -w @compute ls /etc/profile.d/z0*
```

```
$ sudo clush -w @compute puppet agent -t
```

6. Add modulefiles for lci and the system GCC

```
$ sudo mkdir /apps/opt/lmod/mf/Core/gcc
```

```
$ sudo cp ~/build-a-cluster/apps/opt/lmod/mf/Core/gcc/4.8.lua
/apps/opt/lmod/mf/Core/gcc/
```

```
$ module avail
```

```
$ module show gcc
```

```
$ cat /apps/opt/lmod/mf/Core/gcc/4.8.lua
```