



node0:

1) Setup DHCP on enp8s0f0

```
$ vi /etc/sysconfig/network-scripts/ifcfg-enp8s0f0
BOOTPROTO=dhcp
```

```
ONBOOT=yes
```

2) Set internal IP address on enp8s0f1

```
$ vi /etc/sysconfig/network-scripts/ifcfg-enp8s0f1
```

```
BOOTPROTO=static
```

```
ONBOOT=yes
```

```
IPADDR=10.0.0.1
```

Note: do NOT set GATEWAY in either of the ifcfg files on node0, that will override the default gateway set by DHCP.

3) Add node1 as the gateway for internal 10.0.0.* addresses

Create a route-enp8s0f0 file.

```
$ vi /etc/sysconfig/network-scripts/route-enp8s0f0
```

```
GATEWAY0=10.0.0.2
```

ADDRESS0=10.0.0.0

NETMASK0=255.255.255.0

node1:

1) Setup interface enp8s0f0 and configure the default gateway to be the headnode:

```
$ vi /etc/sysconfig/network-scripts/ifcfg-enp8s0f0
```

```
BOOTPROTO=static
```

```
ONBOOT=yes
```

```
IPADDR=10.0.0.2
```

```
GATEWAY=10.0.0.1
```

2) Setup interface enp8s0f1:

```
$ vi /etc/sysconfig/network-scripts/ifcfg-enp8s0f1
```

```
BOOTPROTO=static
```

```
ONBOOT=yes
```

```
IPADDR=10.0.0.3
```

Note: do NOT define a GATEWAY in this file as it will override the default gateway set in ifcfg-enp8s0f0.

3) Define a static route to node2:

Since we have a conflicting rule in the routing tables to search for 10.0.0.* addresses directly attached to enp8s0f0 (while node2 is attached to enp8s0f1), we need to add a static route to the last node.

Create a route-enp8s0f0 file.

```
$ vi /etc/sysconfig/network-scripts/route-enp8s0f1
```

```
10.0.0.4 dev enp8s0f1
```

node2:

1) Setup interface enp8s0f0 and configure the default gateway to be the middle node:

```
$ vi /etc/sysconfig/network-scripts/ifcfg-enp8s0f0
```

```
BOOTPROTO=static
```

```
ONBOOT=yes
```

```
IPADDR=10.0.0.4
```

```
GATEWAY=10.0.0.3
```

2) Add node1 as the gateway to node0:

We have a default routing rule to look for 10.0.0.* addresses directly attached to enp8s0f0, but since node0 is not directly attached, we need to add node1 as a gateway to get to the headnode. We cannot add a general rule for 10.0.0.* addresses, however, because a default one exists through enp8s0f0, so ours would be ignored as a "duplicate". We can, however, add a rule specifically to use node1 as the gateway for 10.0.0.1.

Create a route-enp8s0f0 file.

```
$ vi /etc/sysconfig/network-scripts/route-enp8s0f0
```

```
GATEWAY0=10.0.0.3
```

```
ADDRESS0=10.0.0.1
```

```
NETMASK0=255.255.255.255
```

3) (Optional) Disable the unused interface:

```
$ ifdown enp8s0f1
```

```
$ vi /etc/sysconfig/network-scripts/route-enp8s0f1
```

```
ONBOOT=no
```

Hostnames:

You should assign each of your nodes a unique hostname so we don't have to type out its ip address every time we want to access it. This also helps identify which node you're logged into.

1) Set the hostname

On node0:

```
$ hostname node0
```

```
$ vi /etc/hostname
```

```
node0
```

```
$ vi /etc/sysconfig/network
```

```
HOSTNAME=node0
```

ssh into node1 and node2 and repeat.

2) Setup the hosts file (on each node)

```
$ vi /etc/hosts
```

```
10.0.0.1 node0
```

```
10.0.0.2 node1
```

10.0.0.3 node1

10.0.0.4 node2

Passwordless ssh:

With a larger cluster, you would make an NFS mount with the home directories and place your ssh key there. However, with only 3 nodes it's easier just to do it manually for each of them.

Note that you need to do this from every node to every other node in order for things like MPI to work correctly.

1) Generate a key pair on each node (do not give it a password)

```
$ ssh-keygen -t rsa
```

2) node0

```
$ ssh-copy-id -i ~/.ssh/id_rsa.pub root@node1
```

```
$ ssh-copy-id -i ~/.ssh/id_rsa.pub root@node2
```

3) node1

```
$ ssh-copy-id -i ~/.ssh/id_rsa.pub root@node0
```

```
$ ssh-copy-id -i ~/.ssh/id_rsa.pub root@node2
```

4) node2

```
$ ssh-copy-id -i ~/.ssh/id_rsa.pub root@node0
```

```
$ ssh-copy-id -i ~/.ssh/id_rsa.pub root@node1
```

NAT:

You need to setup NAT and port forwarding on your headnode (node0) and your middle node (node1).

1) Disable the firewall and flush iptables on all nodes:

```
# Disable firewall
```

```
$ systemctl disable firewalld
```

```
$ systemctl stop firewalld
```

```
$ systemctl status firewalld
```

```
# Flush IP tables
```

```
$ iptables --flush
```

```
$ iptables --flush -t nat
```

2) Setup NAT (on node0 and node1):

```
$ iptables -t nat -A POSTROUTING -o enp8s0f0 -j MASQUERADE
$ iptables-save
```

3) Enable port forwarding (on node0 and node1):

```
$ sysctl -w net.ipv4.ip_forward=1
$ vi /etc/sysctl.conf
net.ipv4.ip_forward=1
net.ipv6.conf.default.disable_ipv6=1
net.ipv6.conf.all.disable_ipv6=1
```

```
# enable the change
$ sysctl -p /etc/sysctl.conf
```

Internet Access:

1) Copy DNS server list

The DNS server list was generated by DHCP on your headnode, you need to copy it to your other nodes in order to be able to access servers by their name.

On headnode:

```
$ scp /etc/resolv.conf root@node1:/etc/
```

```
$ scp /etc/resolv.conf root@node2:/etc/
```

2) Set the proxy for yum etc. (do on all nodes for internet access at Wits)

```
$ vi ~/.bashrc
```

```
export http_proxy='http://students\student-number:password@proxyss.wits.ac.za:80'
```

```
export https_proxy='https://students\student-number:password@proxyss.wits.ac.za:80'
```

```
export HTTP_PROXY=$http_proxy
```

```
export HTTPS_PROXY=$https_proxy
```

```
$ source ~/.bashrc
```

NTP:

Network time protocol (NTP) is important for any cluster because it prevents certain security vulnerabilities and makes log file times consistent across nodes when tracing events or errors.

1) Setup NTP server on node0

Ensure the datetime is correct on your headnode

```
date -R
```

```
# Set the right timezone if it's wrong
```

```
timedatectl set-timezone Africa/Johannesburg
```

```
# Install NTP
```

```
$ yum install ntp -y
```

```
$ ntpdate pool.ntp.org
```

```
# UDP port 123 is blocked on the Wits network, so we cannot synchronise with external servers.
```

```
# This marks our NTP server as 'unreliable' so need to make a change to appear reliable.
```

```
# Add the following above the normal servers (fudge our stratum so our local network syncs to us)
```

```
$ vi /etc/ntp.conf
```

```
server 127.127.1.0 # local clock
```

```
fudge 127.127.1.0 stratum 10
```

```
# Set ntpd to start on boot
```

```
$ systemctl enable ntpd
```

```
# Start the NTP server
```

```
$ service ntpd start
```

```
$ ntpq -p
```

```
2) Setup NTP clients on node1, node2
```

```
$ yum install ntp -y
```

```
# Set the right timezone
```

```
timedatectl set-timezone Africa/Johannesburg
```

```
# Comment out the other servers in ntp.conf and add the headnode
```

```
$ vi /etc/ntp.conf
```

```
server 10.0.0.1
```

```
# Set ntpd to start on boot
```

```
$ systemctl enable ntpd
```

```
# Start ntpd
```

```
service ntpd start
```

```
# Ensure the jitter, etc. is non-zero (so we know it's working)
```

```
ntpq -p
```

```
# Ensure the date is correct (after waiting a few minutes to synchronise)
```

date -R