**High Performance Computing**

**System Administrator**

****

**CENTRE FOR DEVELOPMENT OF ADVANCED COMPUTING PUNE**

**CASE STUDY**

**Submitted By:**

Batch September 2022-23

**AIM**

Build a two node Disk-less HPC-Cluster using OpenHPC with warewulf, slurm, Nagios and do a HPL benchmark and document the result.

**Group Members:**

| **S No.** | **NAME** | **PRN** |
| --- | --- | --- |
| **01** | **Lalit Painkra** | **220940127042** |
| **02** | **Langde Dhammdip Govindrao** | **220940127043** |
| **03** | **Mahendra Kumar Pankaj** | **220940127044** |
| **04** | **Megha Chayrulal Kalyankar** | **220940127046** |
| **05** | **Numesh Kumar Sahare** | **220940127047** |

**TABLE OF CONTENT**

**Requirements ……………………………………………………….04**

**Hardware requirements……………………………………….**

**Software requirements………………………………………...**

**Installation …………………………………………………………..05**

**Pre-Configuration ..…………………………………………………06**

**OpenHPC with Warewulf…………………………………………..08**

**Slurm ..………………………………………………………………12**

**Nagios ………………………………………………………………..15**

**Ganglia ……………………………………………………………...21**

**HPL Benchmarking ………………………………………………...23**

**Commands History.............................................................................29**

**REQUIREMENTS**

**Hardware requirements:**

* RAM : 32 GB
* PROCESSOR : i7 10 gen
* HDD : 200GB

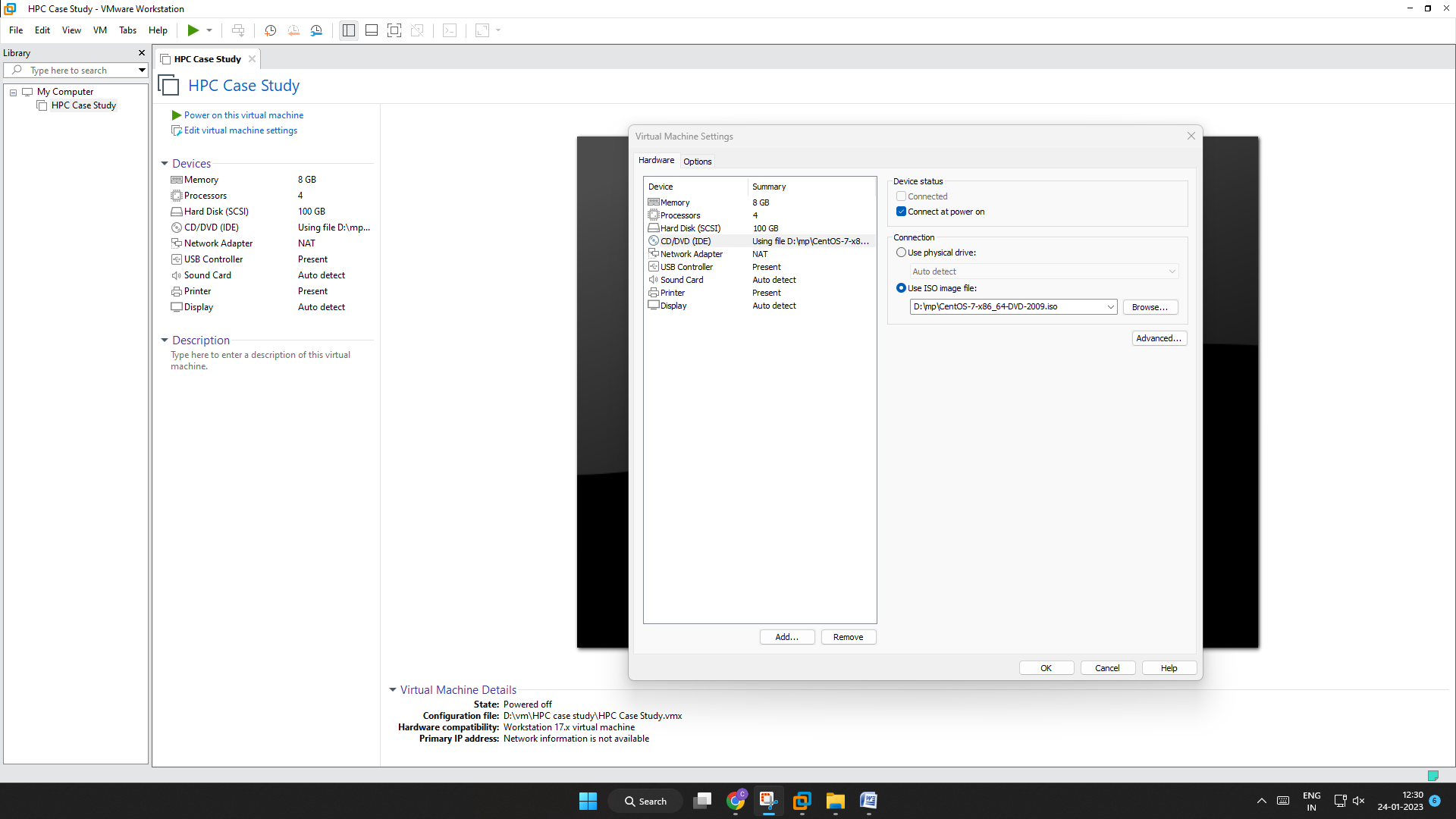
**Software requirements:**

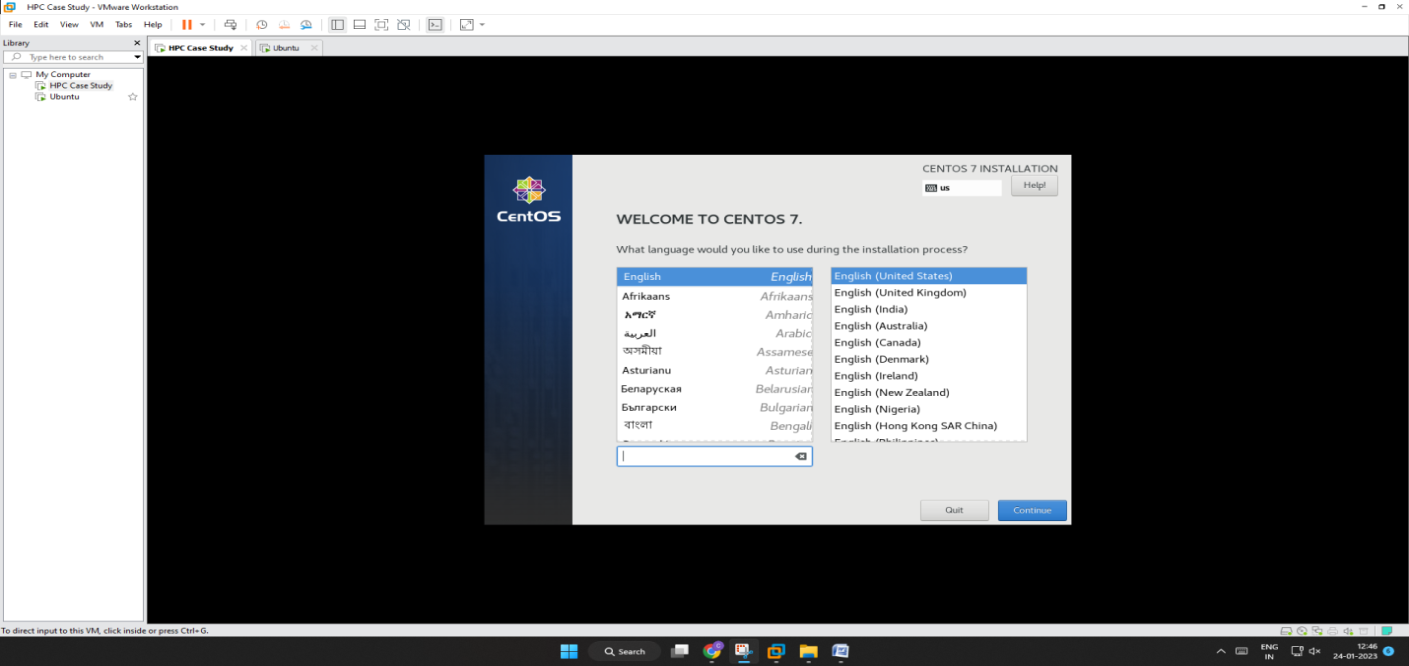
* Vmware workstation
* Centos 7 iso

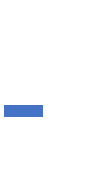
\*Internet connectivity **INSTALLATION**

The head node is configured as the primary node in the cluster and is setup to manage and install all compute nodes.

Create new virtual machine and Boot from CentOS\*installmedia(DVD).



****



After done to create virtual machine of Centos 7 with master configuration few must configuration are required

1. Setting hostname : master
2. Firewalld must be disabled
3. Selinux disable
4. Network configuration must be done

# hostnamectl set-hostname master

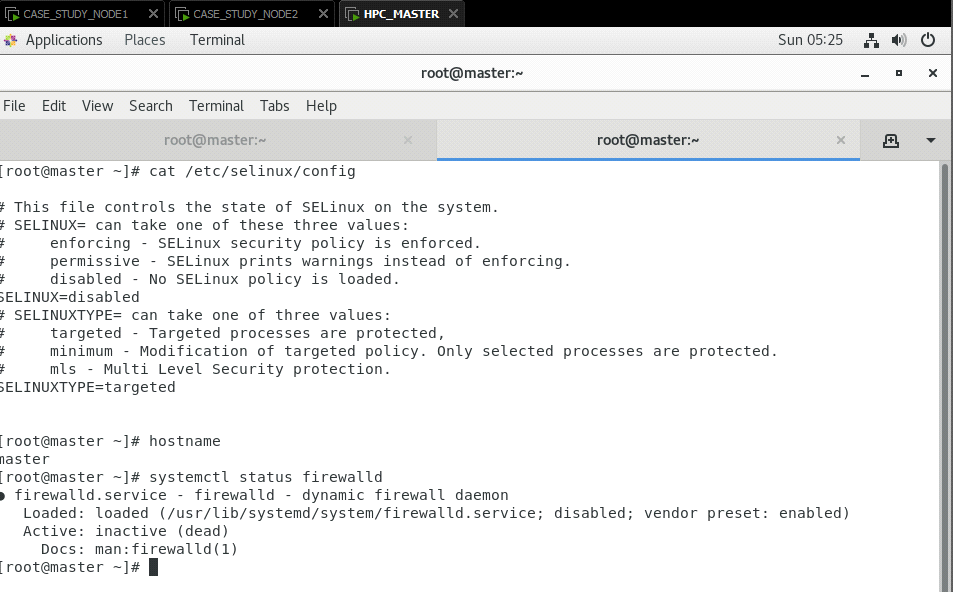
[root@master ~]#

# systemctl stop firewalld

#systemctl disable firewalld

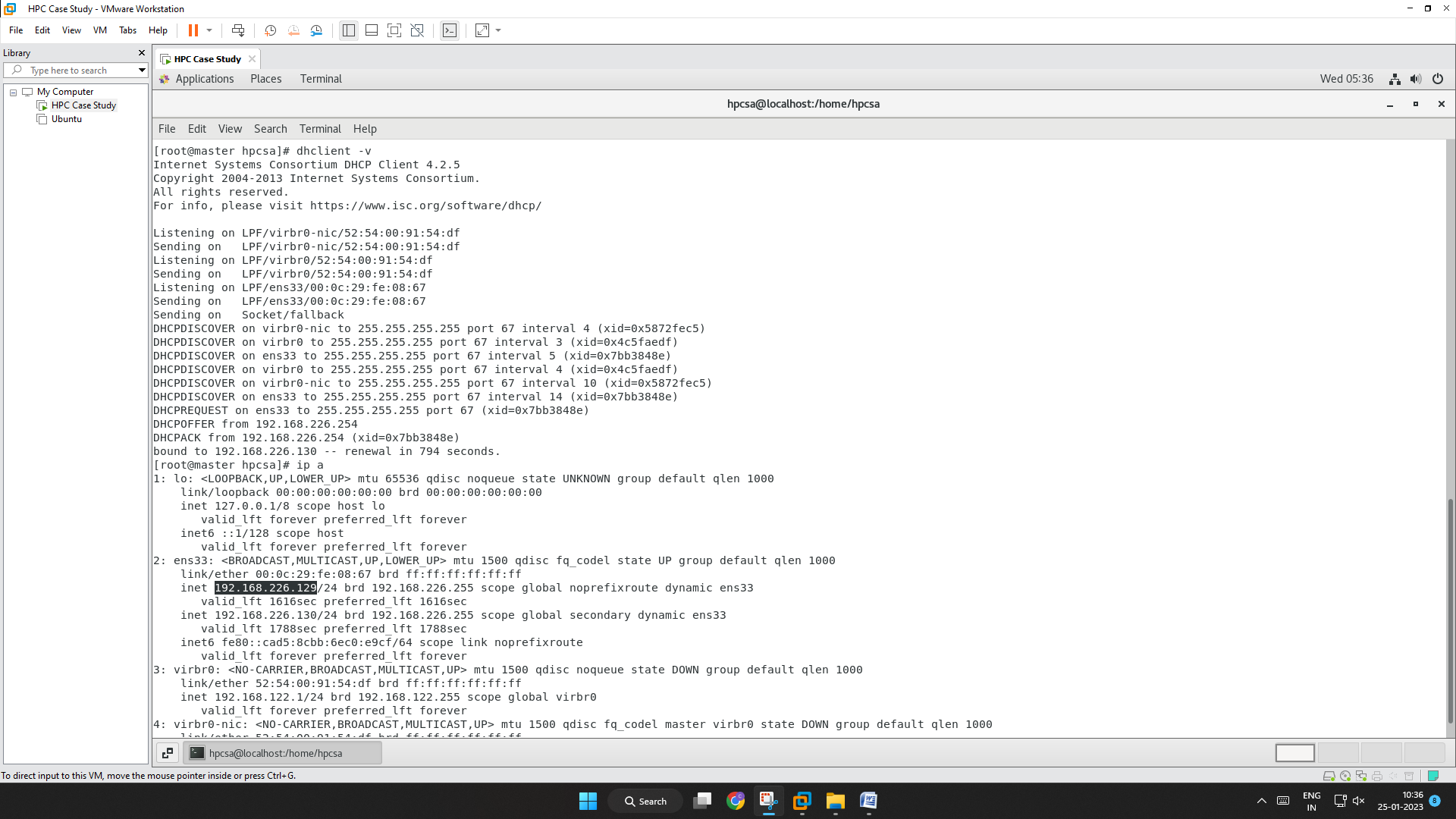
# vi /etc/selinux/conf

🡪 Change enforcing to disabled

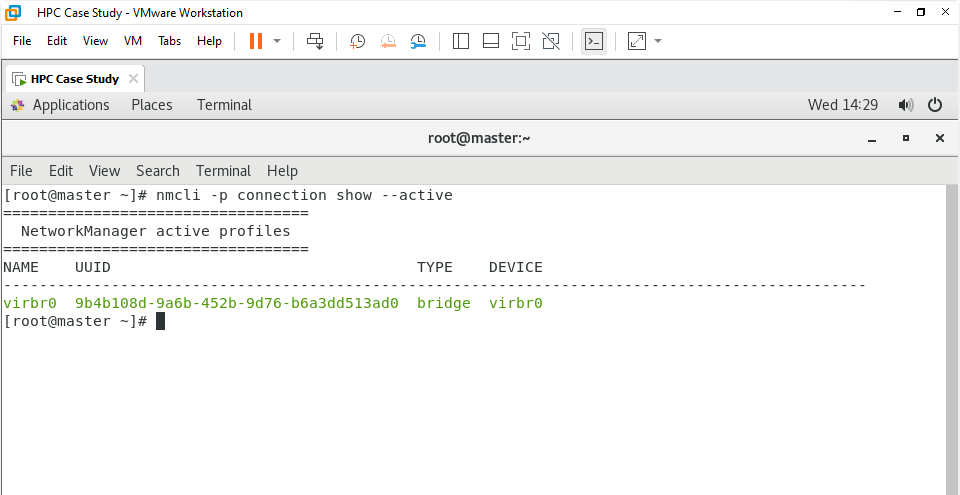


# dhclient -v

# ip a



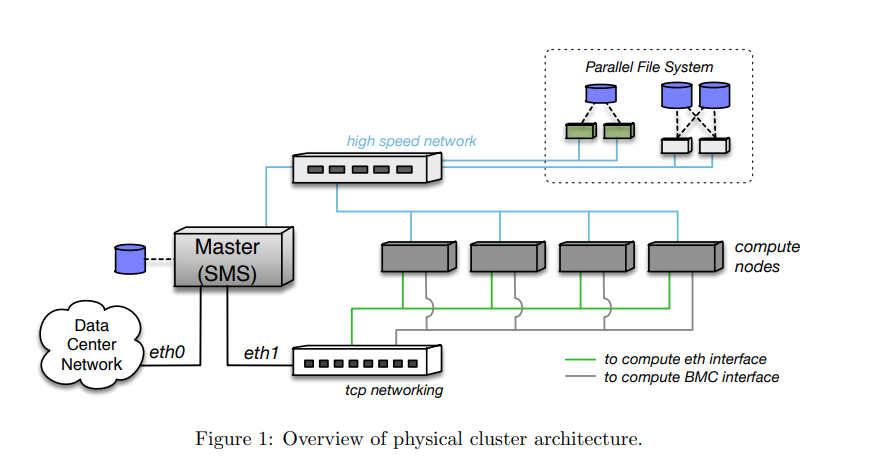
# nmcli -p connection show --active



**openHPC with Warewulf**

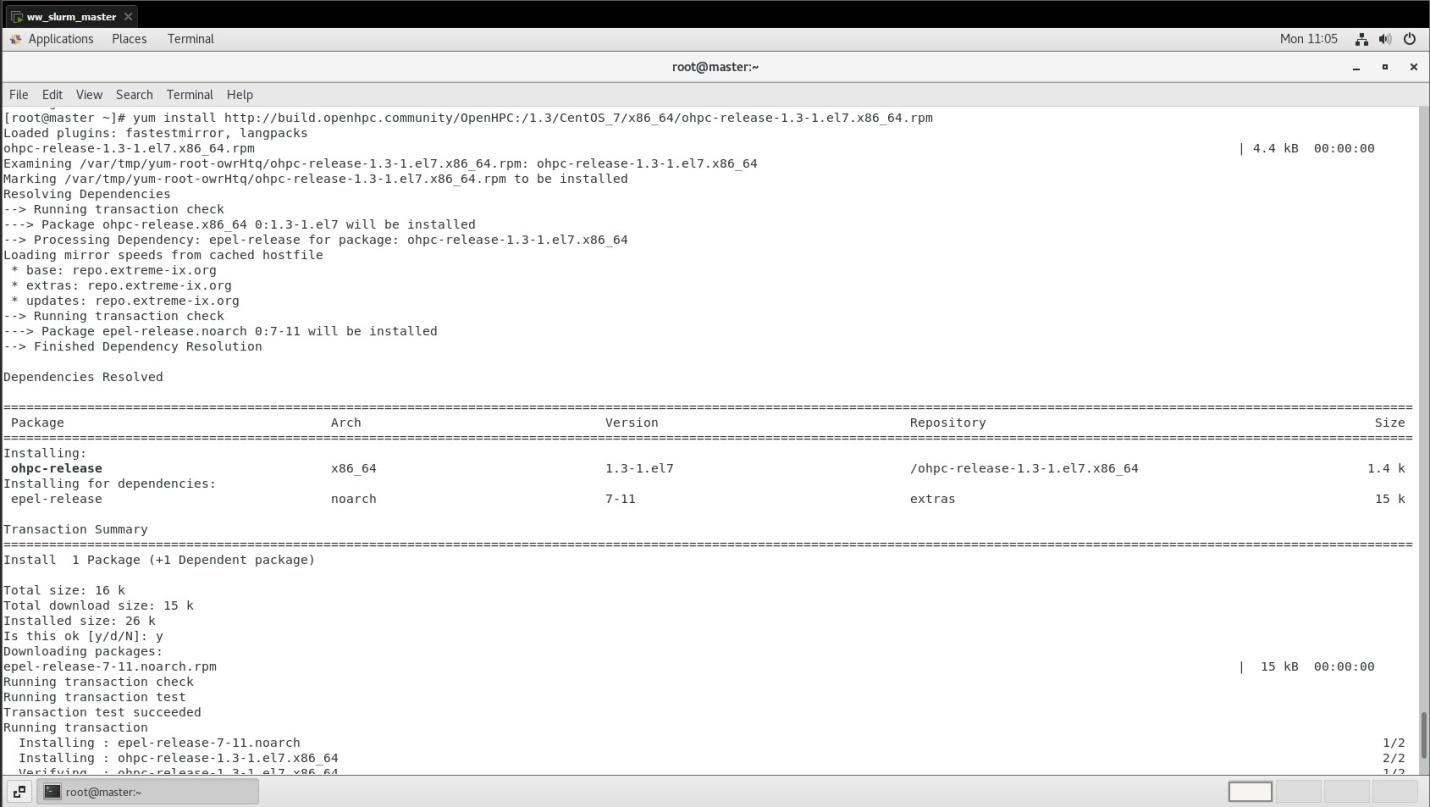
**OpenHPC** is a set of community-driven FOSS tools for Linux based HPC. OpenHPC does not have specific hardware requirements.

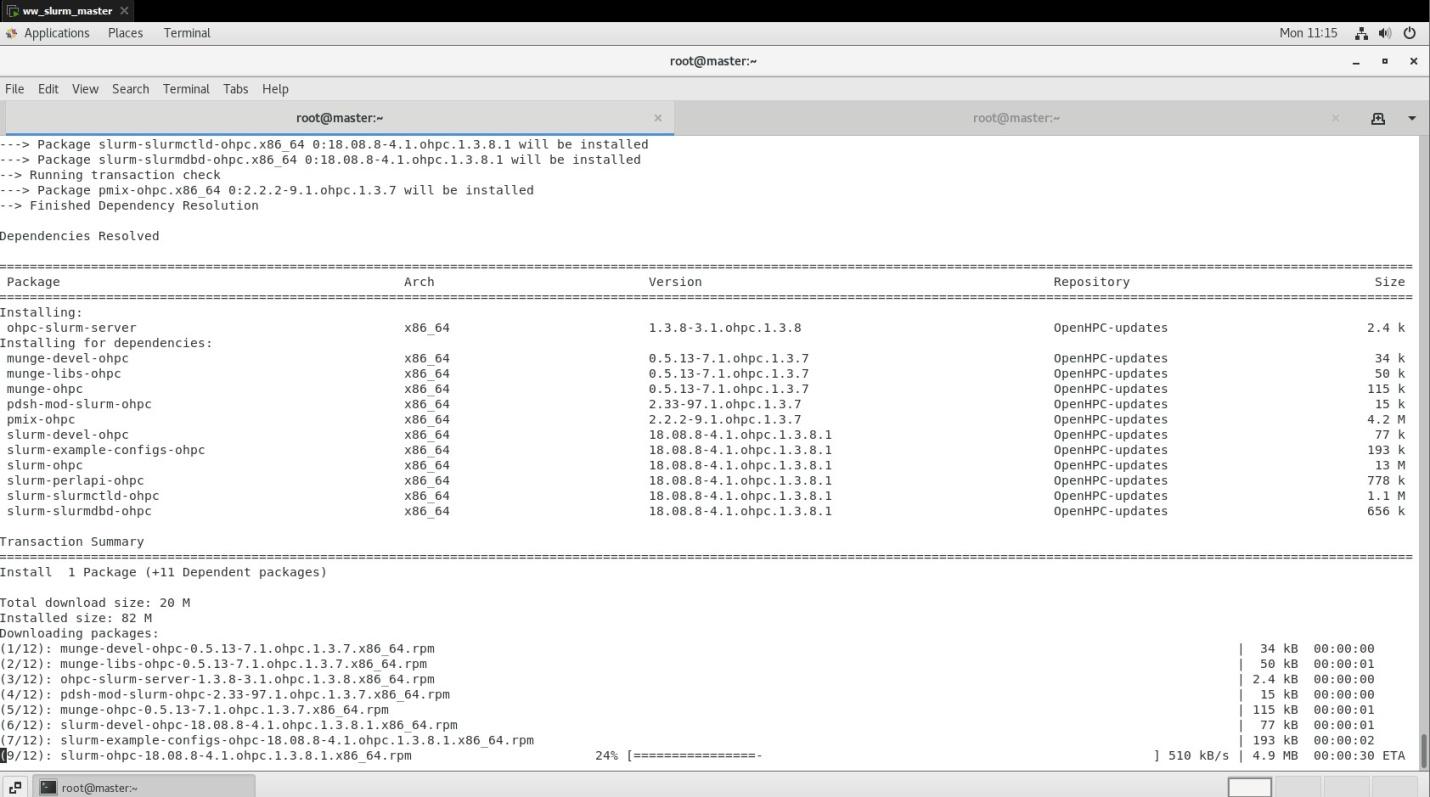
Warewulf is **a bare metal, stateless, cluster provisioning solution to facilitate the operating system deployment and management of large quantities of clustered hardware resources**. Extensible. Easy to change the default functionality, node images, and customize for any clustering use-case**.**

****

#Yum install http://build.openhpc.community/OpenHPC:/1.3/CentOS\_7/aarch64/ohpc-release-1.3-1.el7.aarch64.rpm

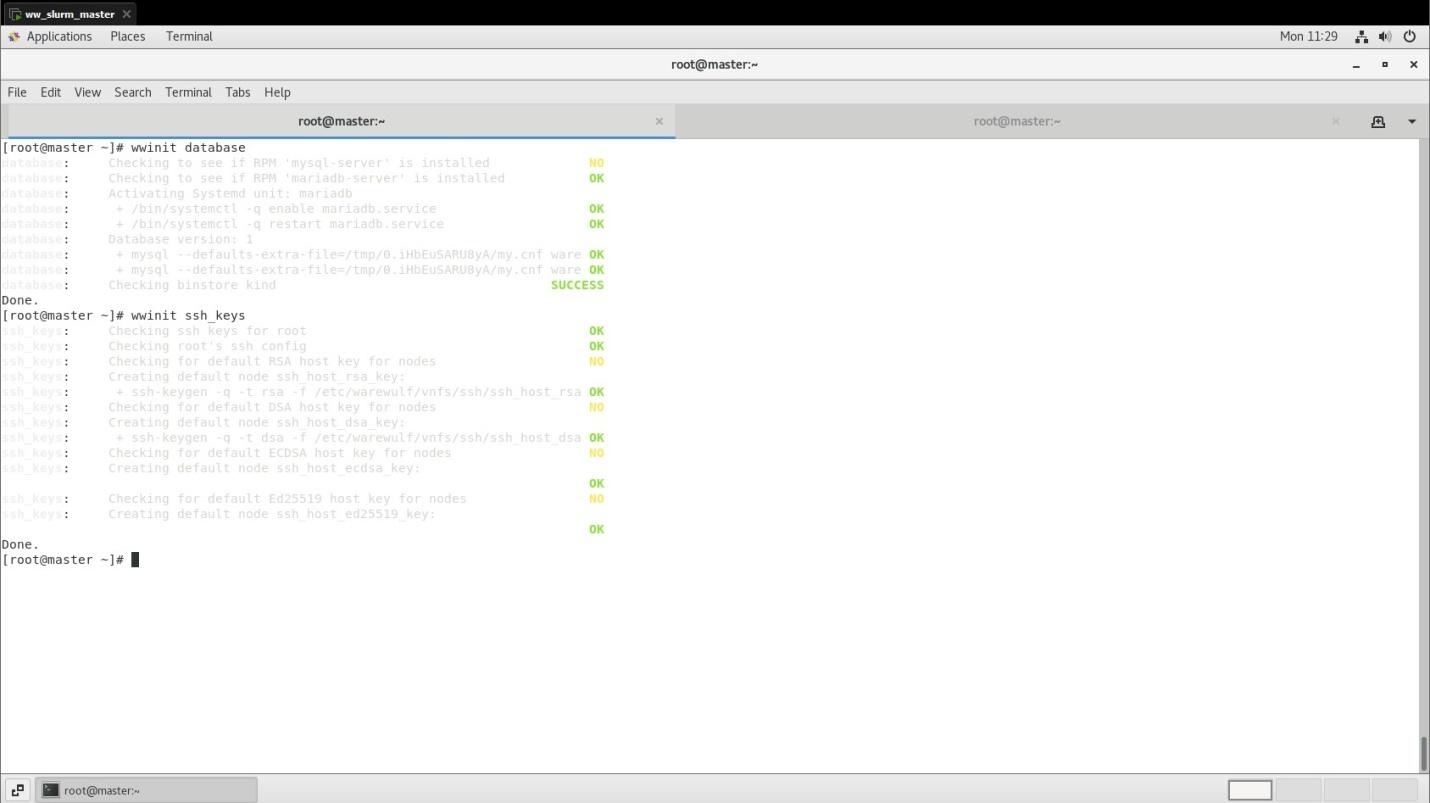
# yum -y install ohpc-base





# wwinit database

# wwinit ssh\_keys



# df -hT | grep -v tmpfs

# echo "master:/home /home nfs nfsvers=3,nodev,nosuid 0 0" >> $CHROOT/etc/fstab

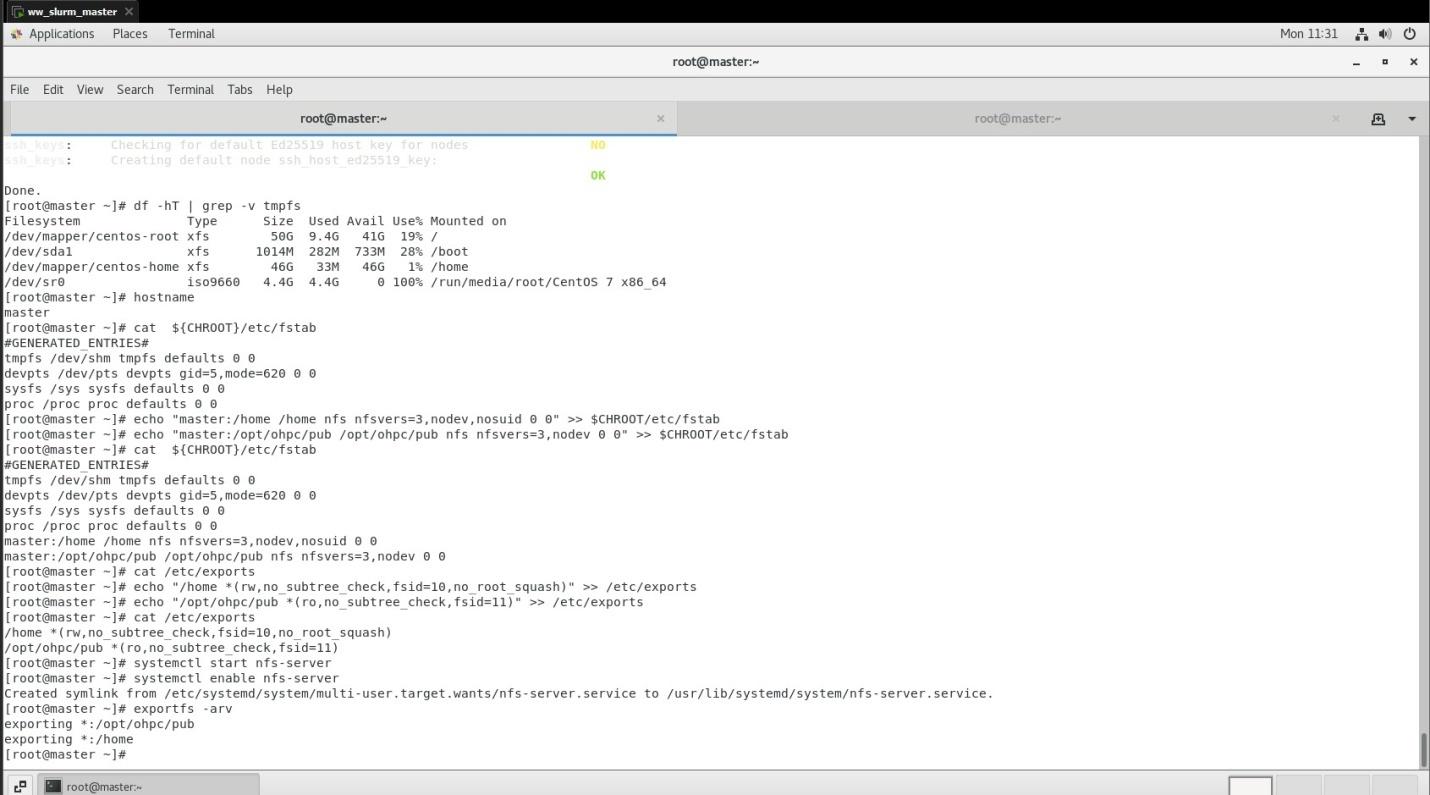
# echo "master:/opt/ohpc/pub /opt/ohpc/pub nfs nfsvers=3,nodev 0 0" >> $CHROOT/etc/fstab

# cat ${CHROOT}/etc/fstab

# cat /etc/exports

# echo "/home \*(rw,no\_subtree\_check,fsid=10,no\_root\_squash)" >> /etc/exports

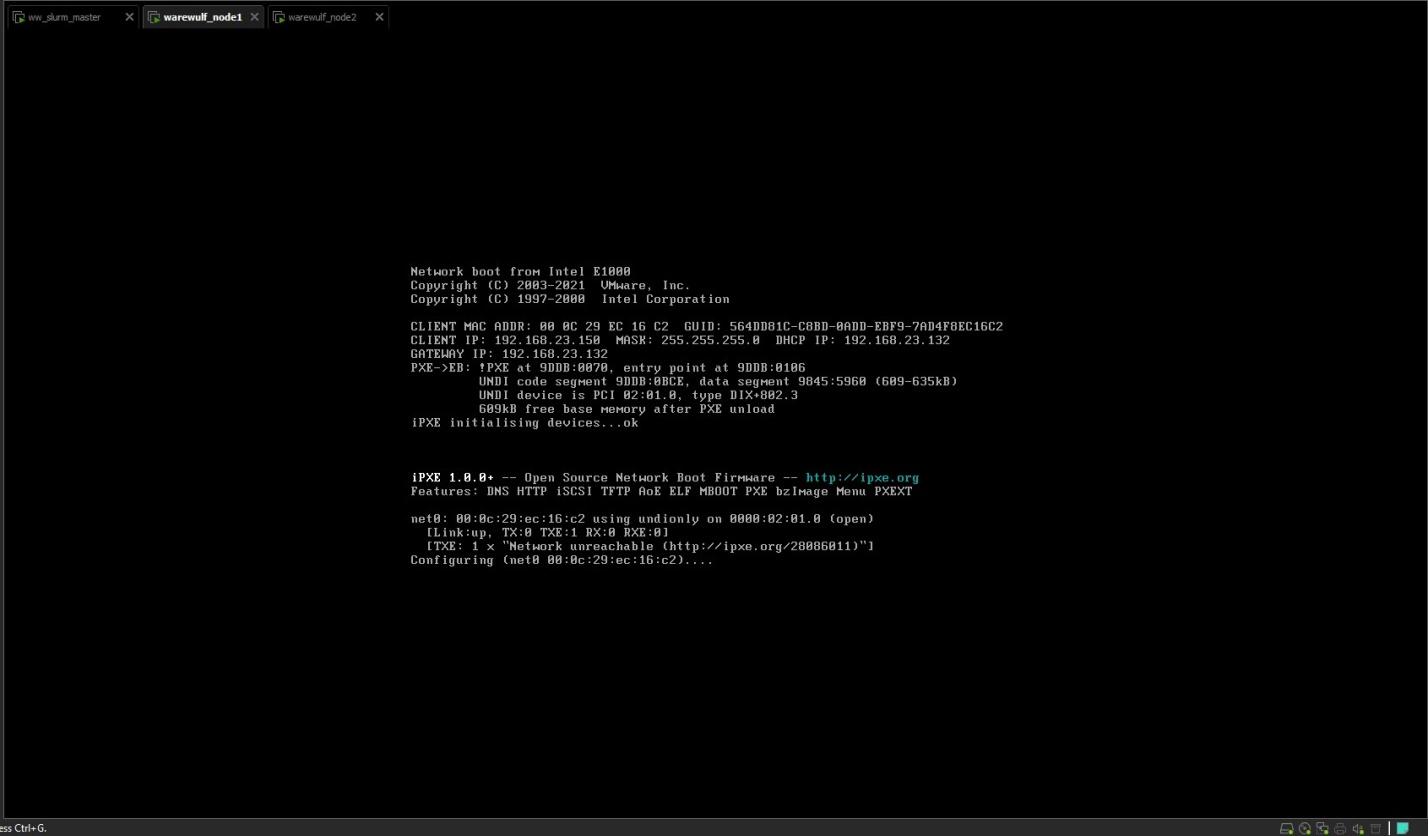
# echo "/opt/ohpc/pub \*(ro,no\_subtree\_check,fsid=11)" >> /etc/exports

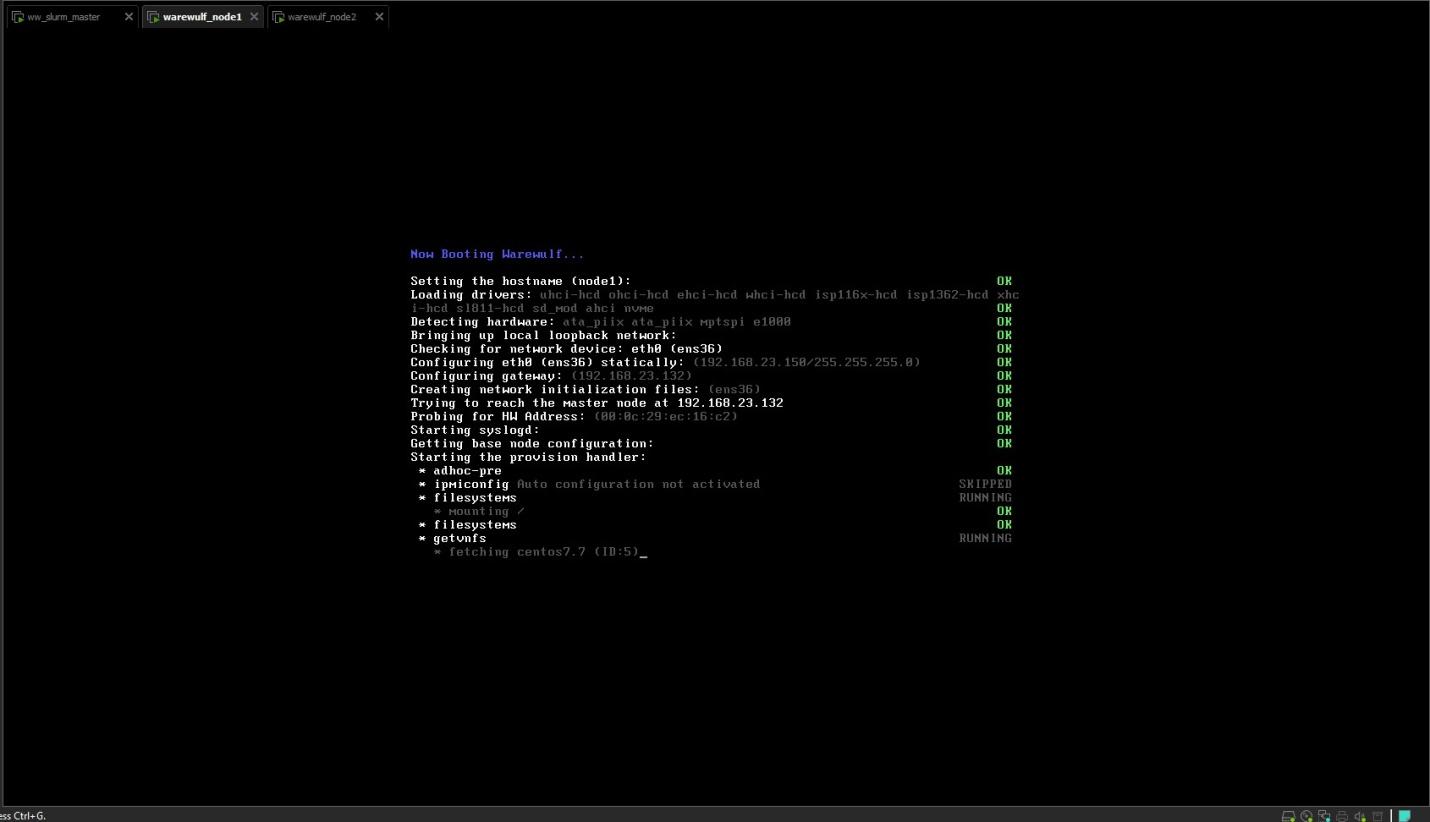


# systemctl start nfs-server

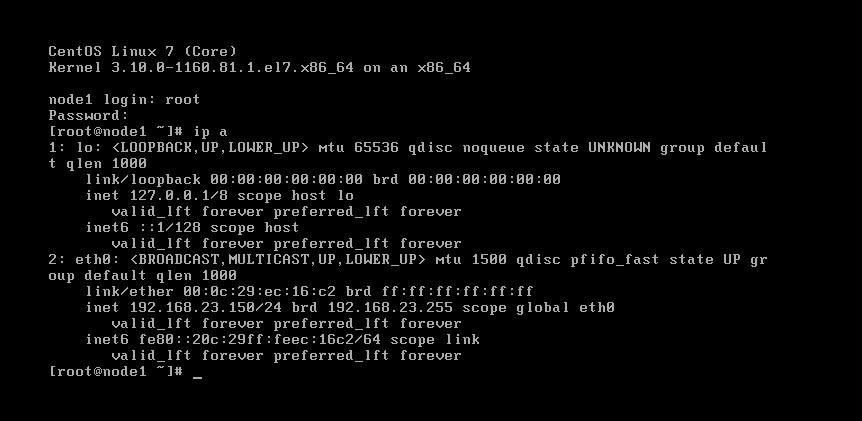
# systemctl enable nfs-server

Booting Disk-less Node



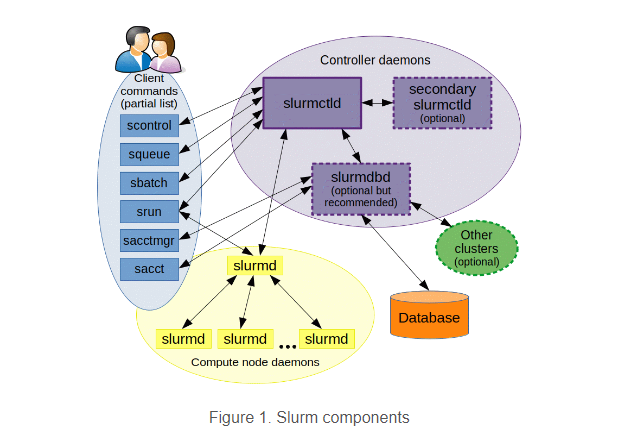


# ip a (Booted node)



**SLURM**

The Slurm Workload Manager, formerly known as Simple Linux Utility for Resource Management (SLURM), or simply Slurm, is a free and open-source job scheduler for Linux and Unix-like kernels, used by many of the world's supercomputers and computer clusters.



# yum -y install ohpc-slurm-server

# yum -y install slurm-sview-ohpc slurm-torque-ohpc

# vi /etc/slurm/slurm.conf

edit -> ClusterName=pearl

-> ControlMachine=master

-> NodeName=node[1-2]

# export CHROOT=/opt/ohpc/admin/images/centos7.7

# wwmkchroot centos-7 $CHROOT

# chroot ${CHROOT} uname -r

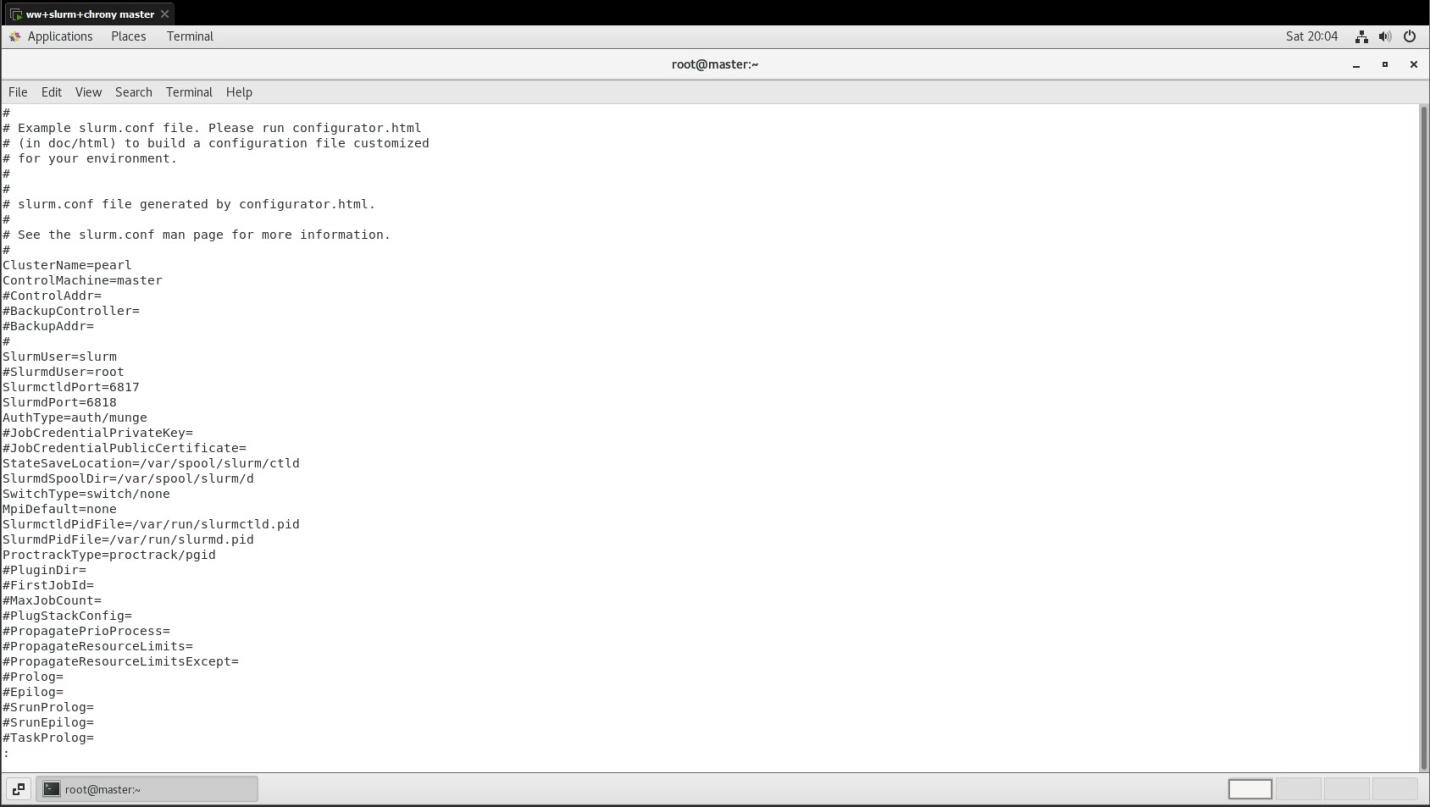
# yum -y --installroot=${CHROOT} install \

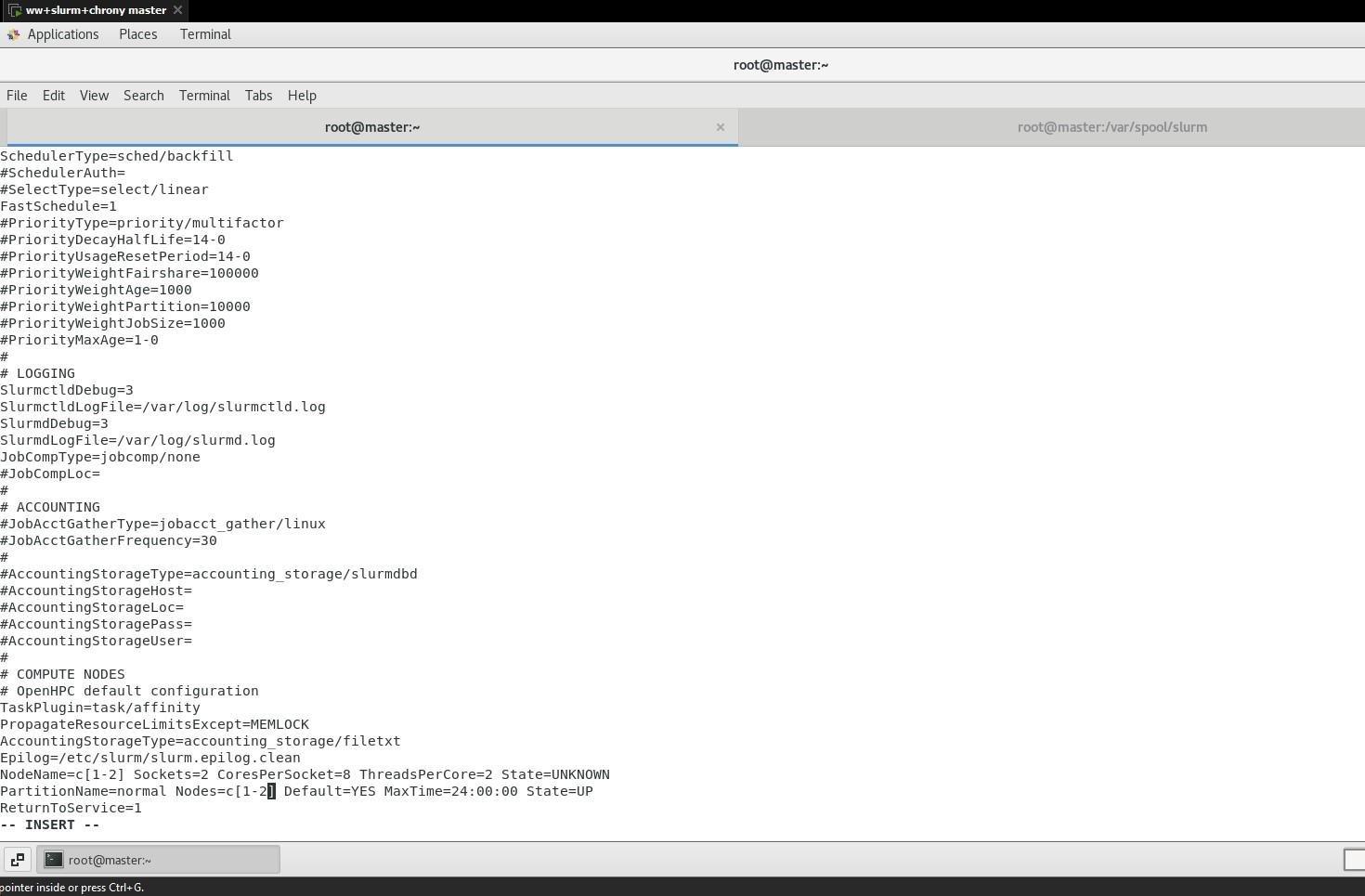
ohpc-base-compute kernel kernel-headers kernel-devel kernel-tools parted \

xfsprogs python-devel yum htop ipmitool glibc\* perl perl-CPAN perl-CPAN \

sysstat gcc make xauth firefox squashfs-tools

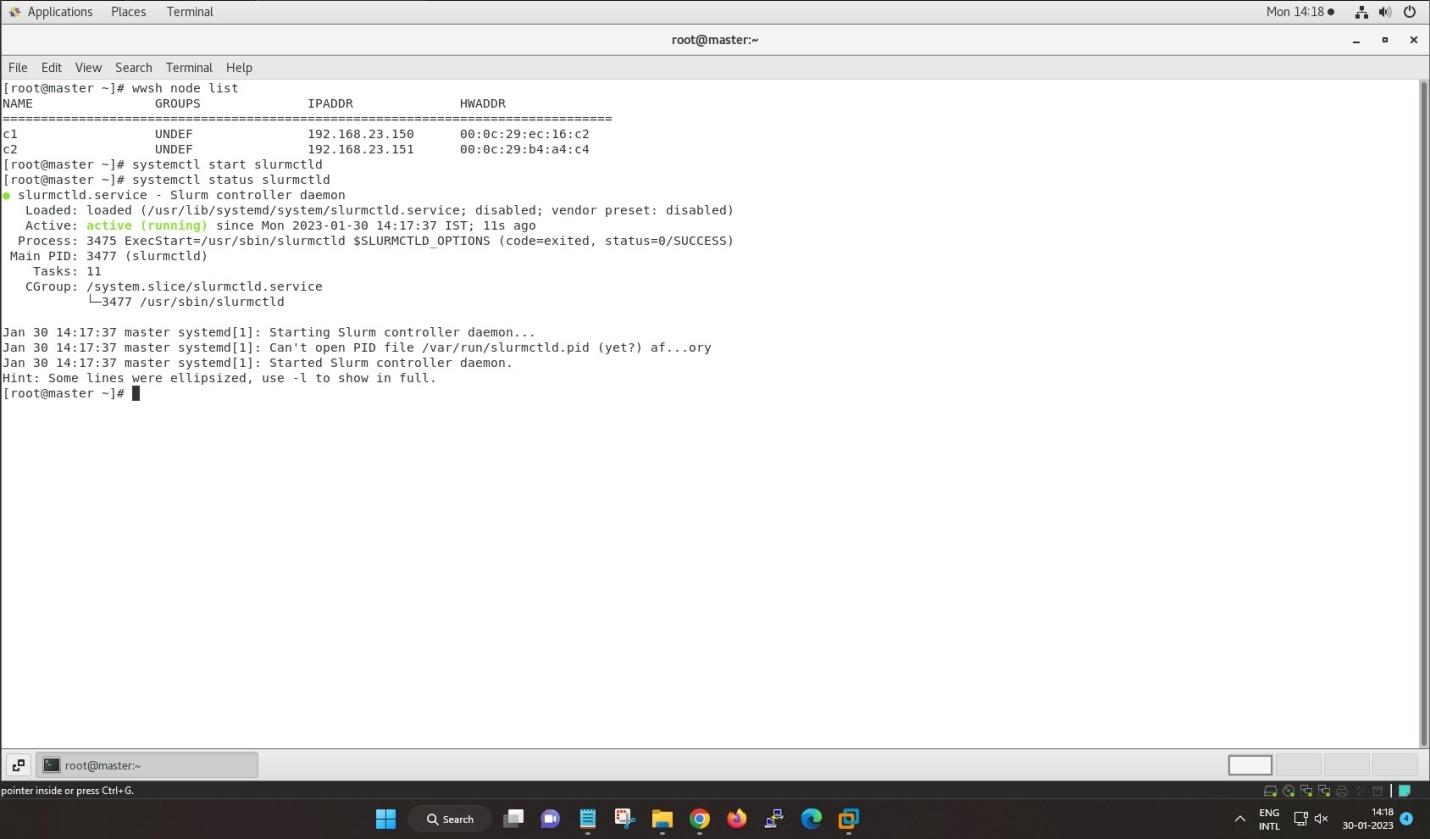
**Slurm Configuration**





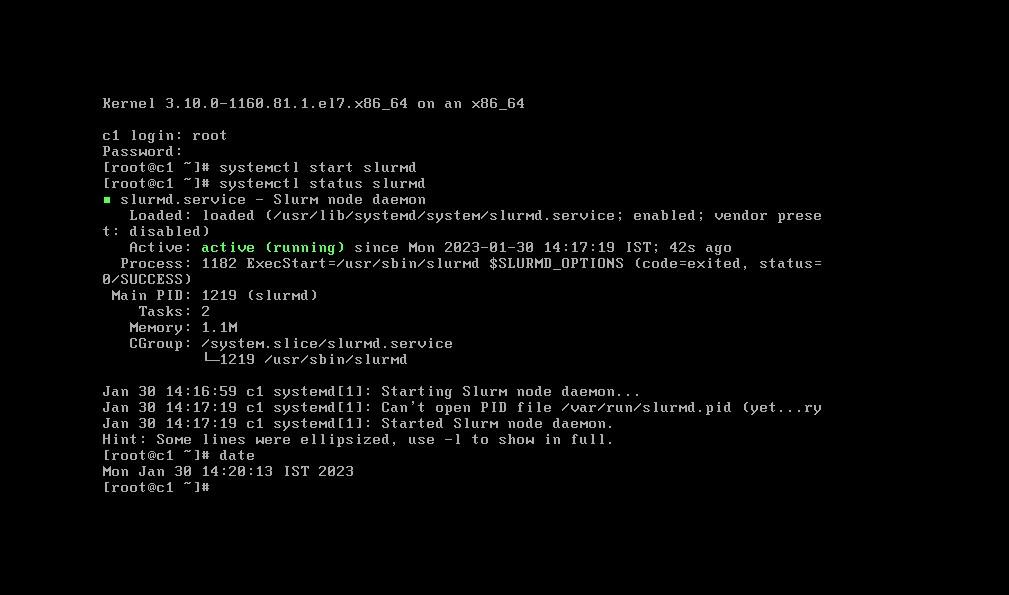
# systemctl status slurmctld

Slurm status on master



# systemctl status slurmd

Slurm status on node



**NAGIOS**

Nagios is **an open source monitoring system for computer systems**. It was designed to run on the Linux operating system and can monitor devices running Linux, Windows and Unix operating systems (OSes). Nagios software runs periodic checks on critical parameters of application, network and server resources.

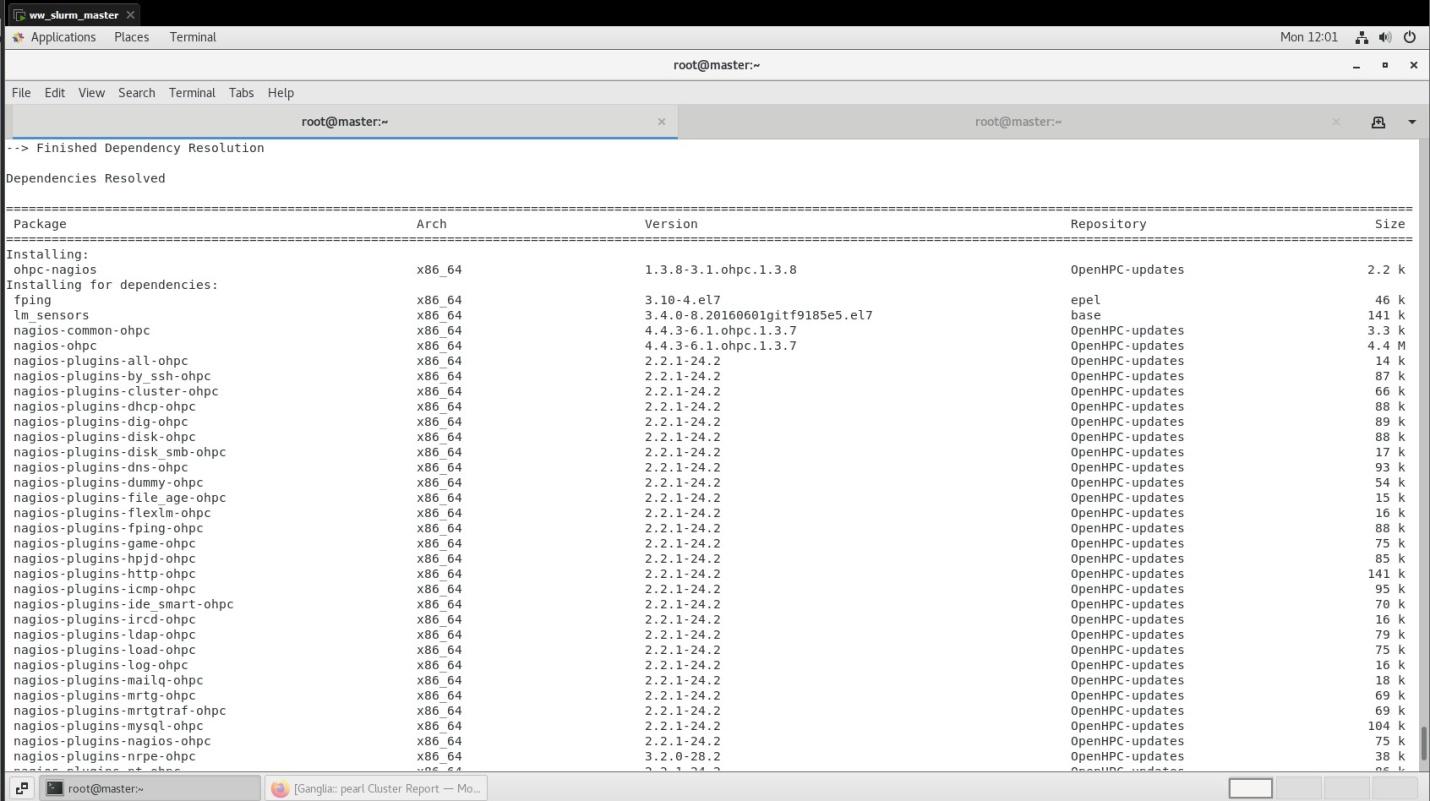
# yum -y install ohpc-nagios -> Install Nagios meta-package on master host

# yum -y --installroot=$CHROOT install nagios-plugins-all-ohpc nrpe-ohpc

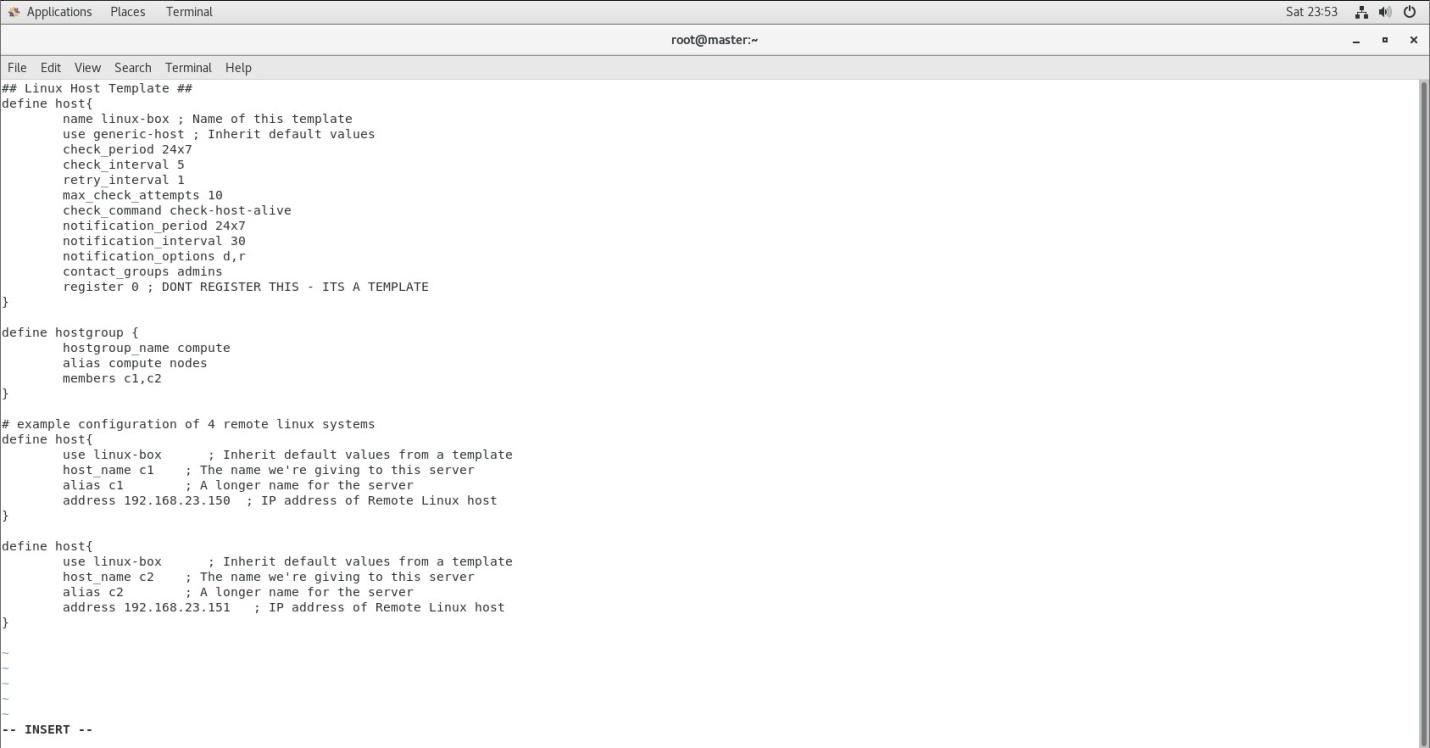
# vi $CHROOT/etc/nagios/nrpe.cfg

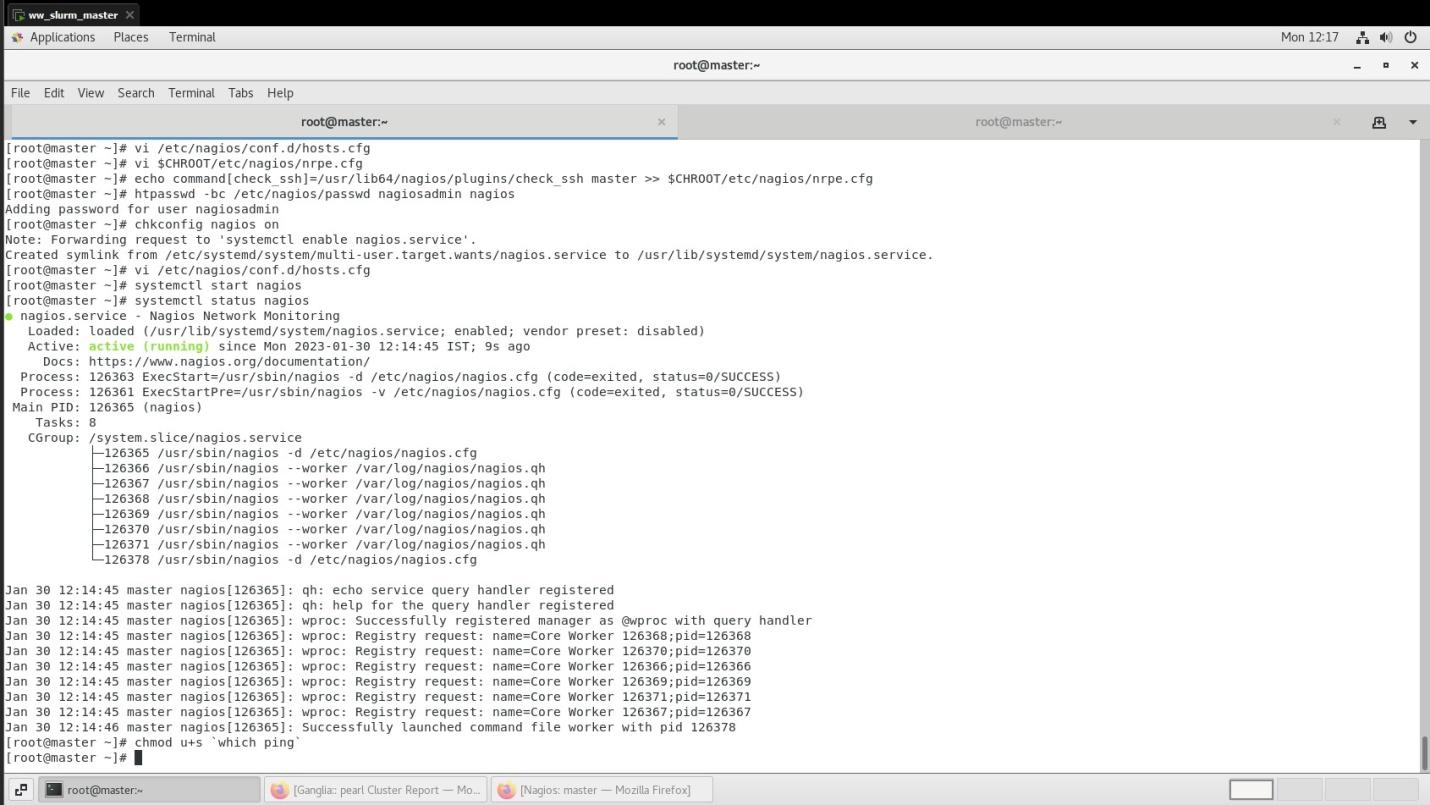
# vi $CHROOT/etc/hosts.allow

Nagios Packages:

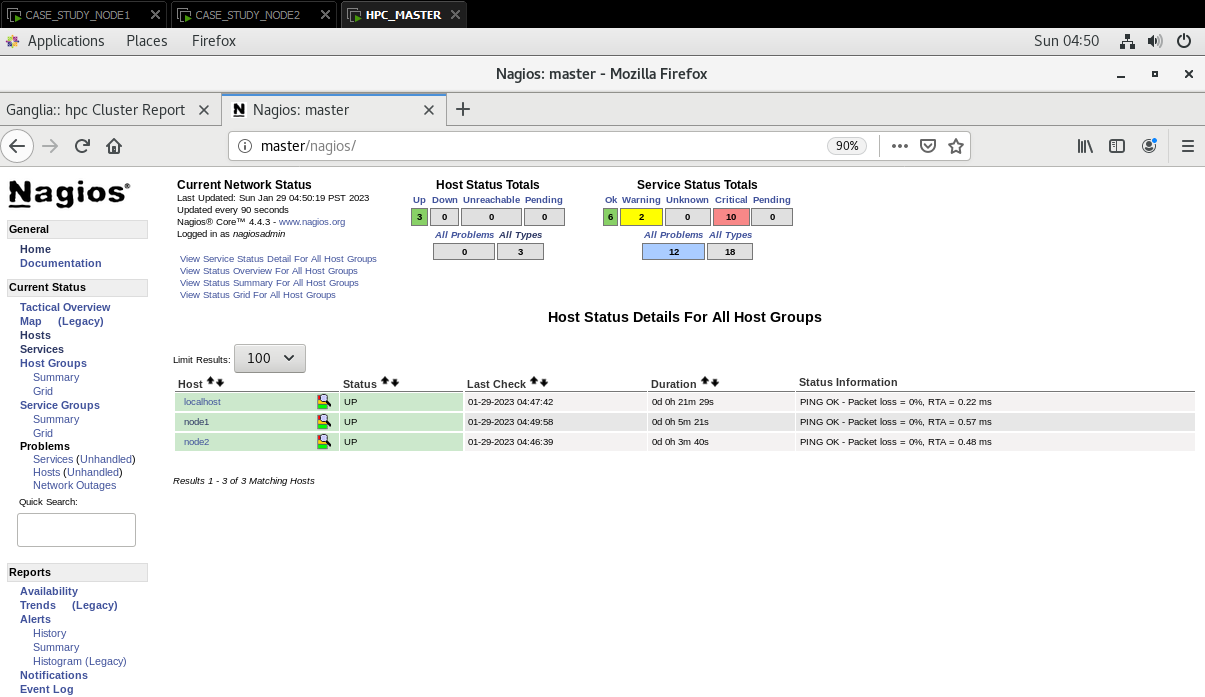


**Nagios Configuration file:**

****



Nagios result on browser:

****

Node1 & Node2 showed

**GANGLIA**

Ganglia is **an open-source scalable distributed monitoring system for high-performance computing systems such as clusters and Grids**. It is carefully engineered to achieve very low per-node overheads and high concurrency.

# yum -y install ohpc-ganglia

# yum -y --installroot=${CHROOT} install ganglia-gmond-ohpc

# cp /opt/ohpc/pub/examples/ganglia/gmond.conf /etc/ganglia/gmond.conf

# sed -i "s/<sms>/master/" /etc/ganglia/gmond.conf

# sed -i "s/OpenHPC/pearl/" /etc/ganglia/gmond.conf

# cp /etc/ganglia/gmond.conf $CHROOT/etc/ganglia/gmond.conf

# echo "gridname pearl" >> /etc/ganglia/gmetad.conf

# echo "

systemctl enable gmond

systemctl enable gmetad

systemctl start gmond

systemctl start gmetad

chroot ${CHROOT} systemctl enable gmond

" > /tmp/start\_ganglia\_service.sh

# bash /tmp/start\_ganglia\_service.sh

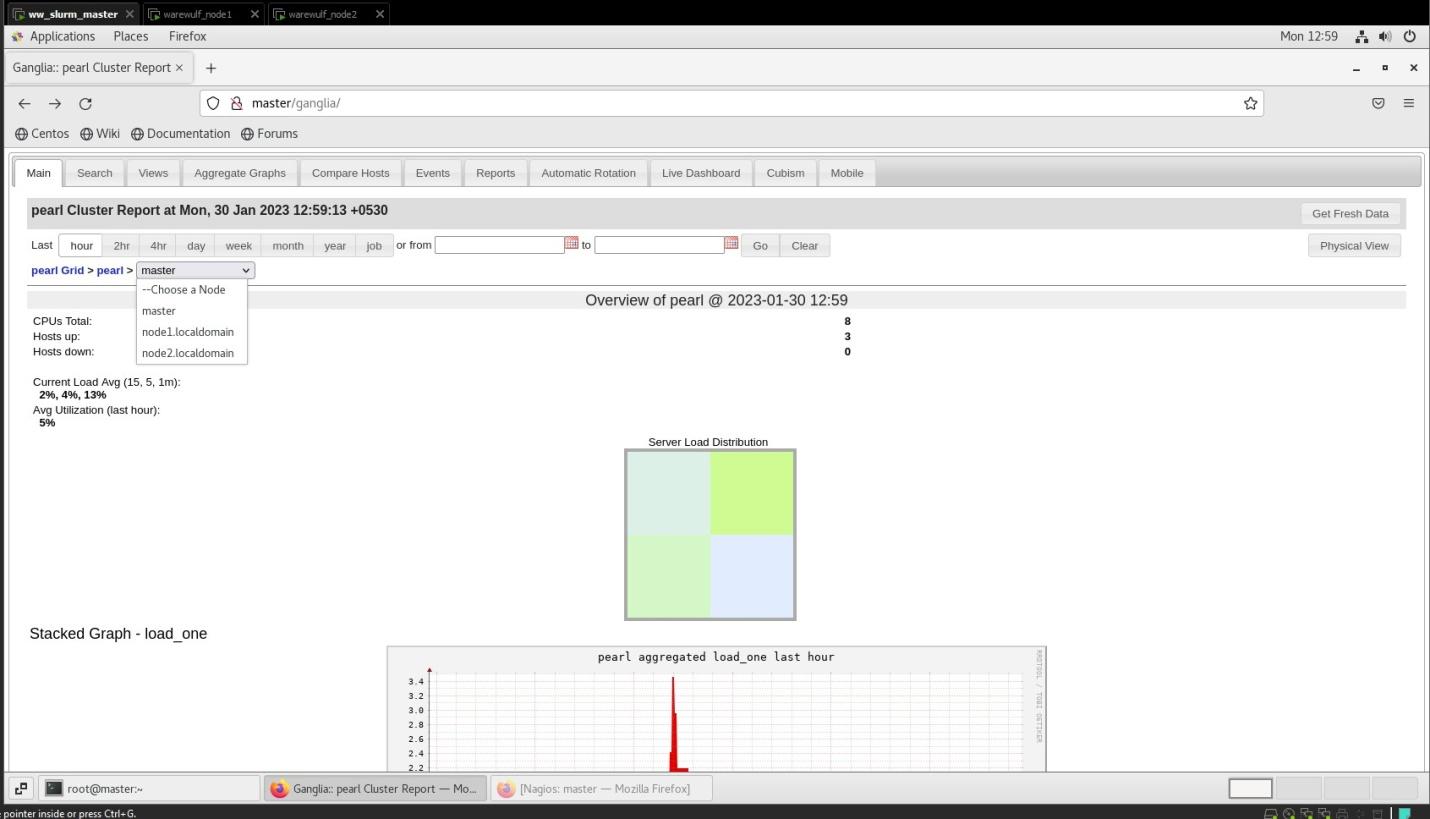
# grep "^date.timezone =" /etc/php.ini

# echo "date.timezone = Asia/Kolkata" >> /etc/php.ini

# grep "^date.timezone =" /etc/php.ini

# systemctl try-restart httpd

**Go to browser :** [**http://master/ganglia**](http://master/ganglia)

****

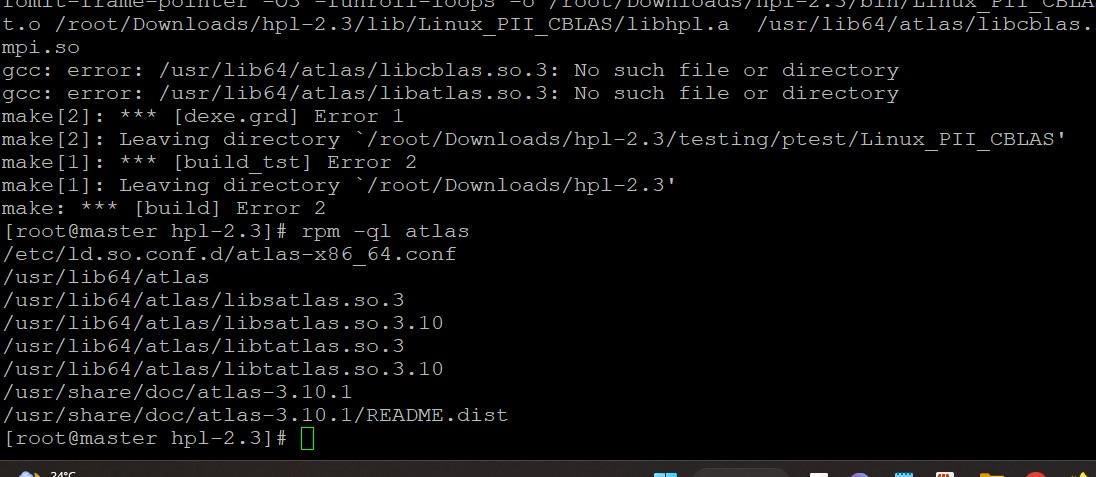
**HPL Benchmarking**

HPL is **a High-Performance Linpack benchmark implementation**. The code solves a uniformly random system of linear equations and reports time and floating-point execution rate using a standard formula for operation count.

# yum install atlas -y

# wget https://netlib.org/benchmark/hpl/hpl-2.3.tar.gz

# wget <https://download.open-mpi.org/release/open-mpi/v4.1/openmpi-4.1.4.tar.gz>

****

# vim Make.Linux\_PII\_CBLAS

>> edit # ----------------------------------------------------------------------

# - HPL Directory Structure / HPL library -------------------------

# ---------------------------------------------------------------------

TOPdir = /root/Downloads/hpl-2.

# ----------------------------------------------------------------------

# - Message Passing library (MPI) ----------------------------------

# ---------------------------------------------------------------------

MPdir = /opt/openmpi-4.1.

MPlib = $(MPdir)/lib/libmpi.so

# ----------------------------------------------------------------------

# - Compilers / linkers - Optimization flags -----------

# ----------------------------------------------------------------------

# CC = /usr/bin/gcc

LINKER = /usr/bin/gcc

# ----------------------------------------------------------------------

# - Linear Algebra library (BLAS or VSIPL) ---------------------

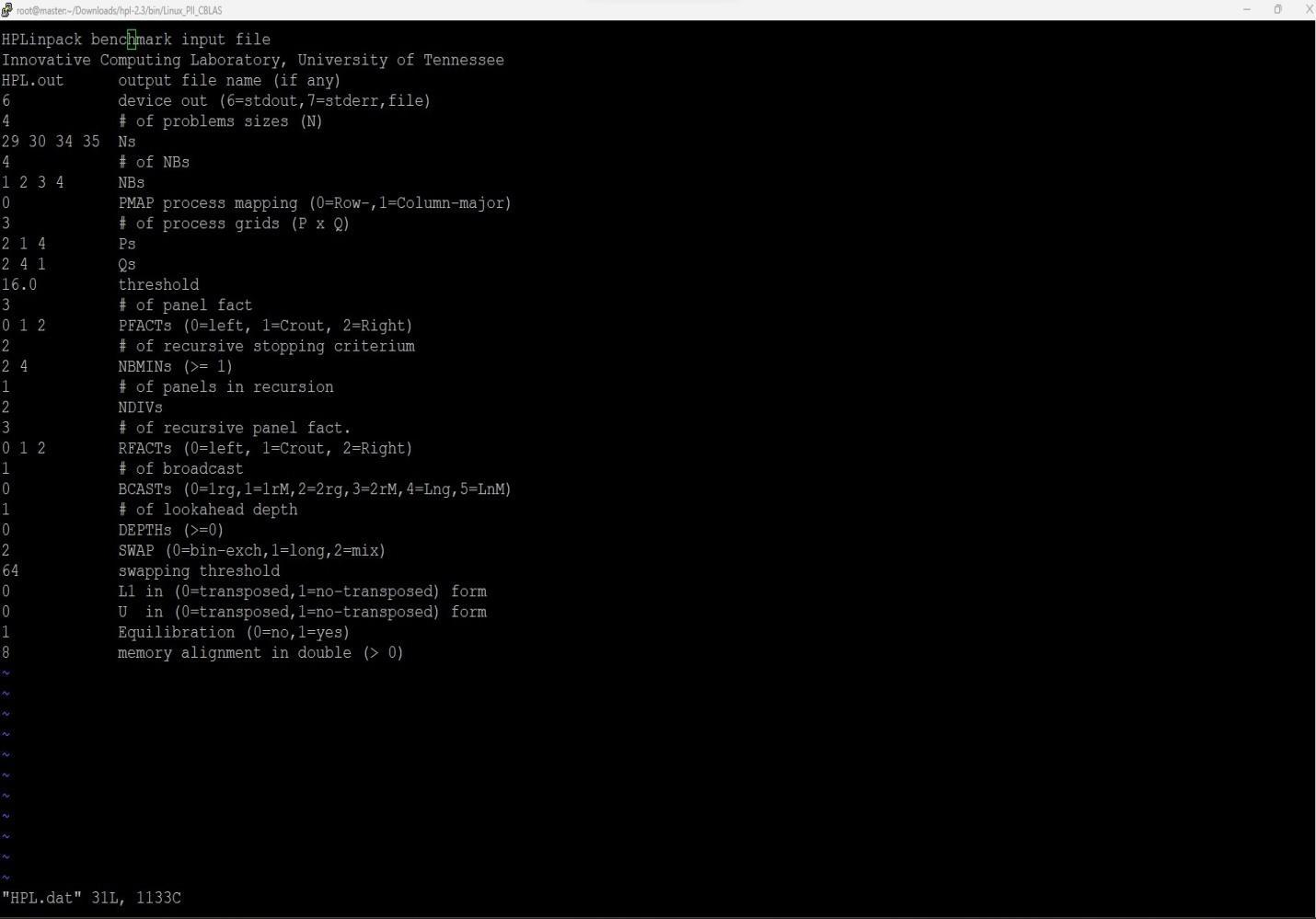
# ----------------------------------------------------------------------

LAlib = $(LAdir)/libsatlas.so.3 $(LAdir)/libtatlas.so.3

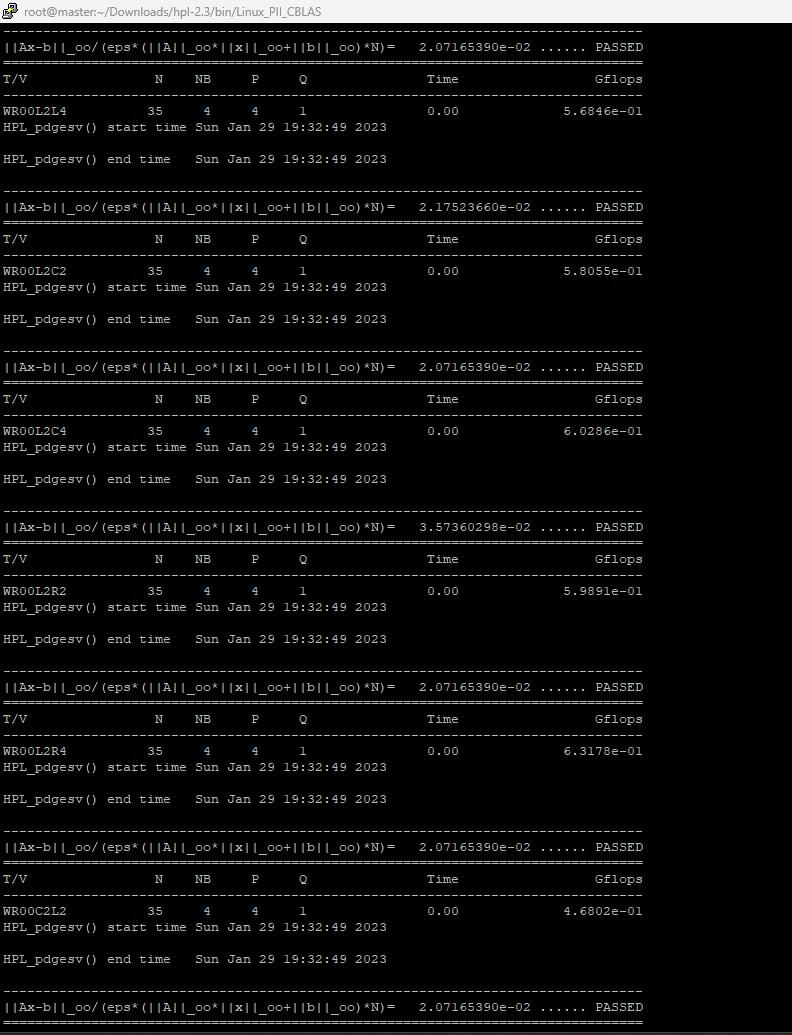
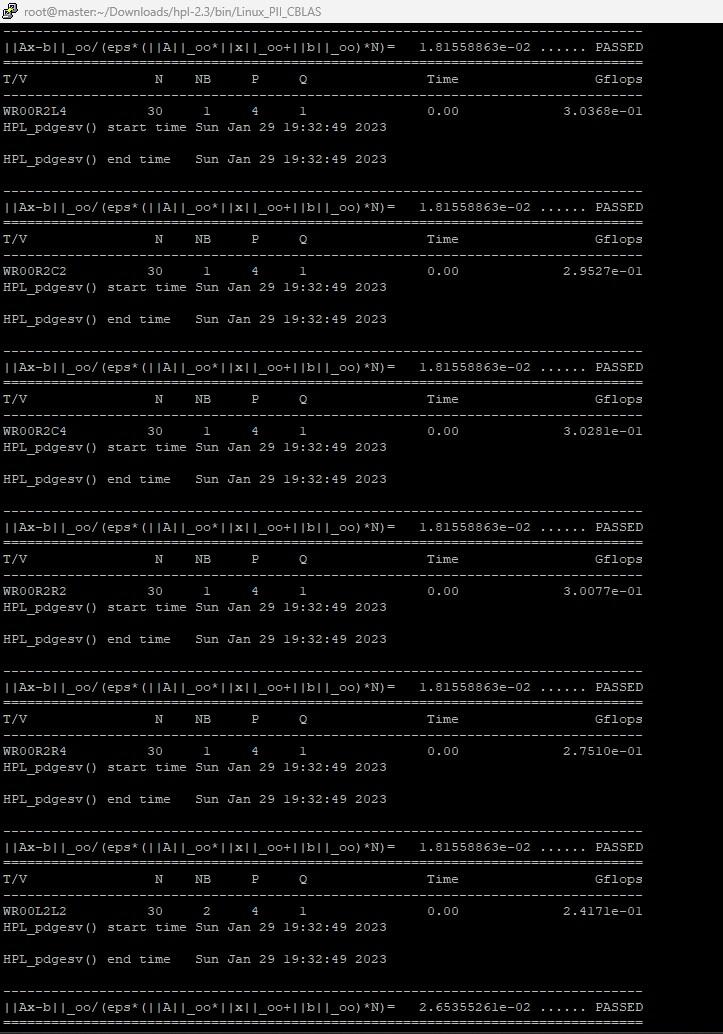
>> <Escape Key> : wq

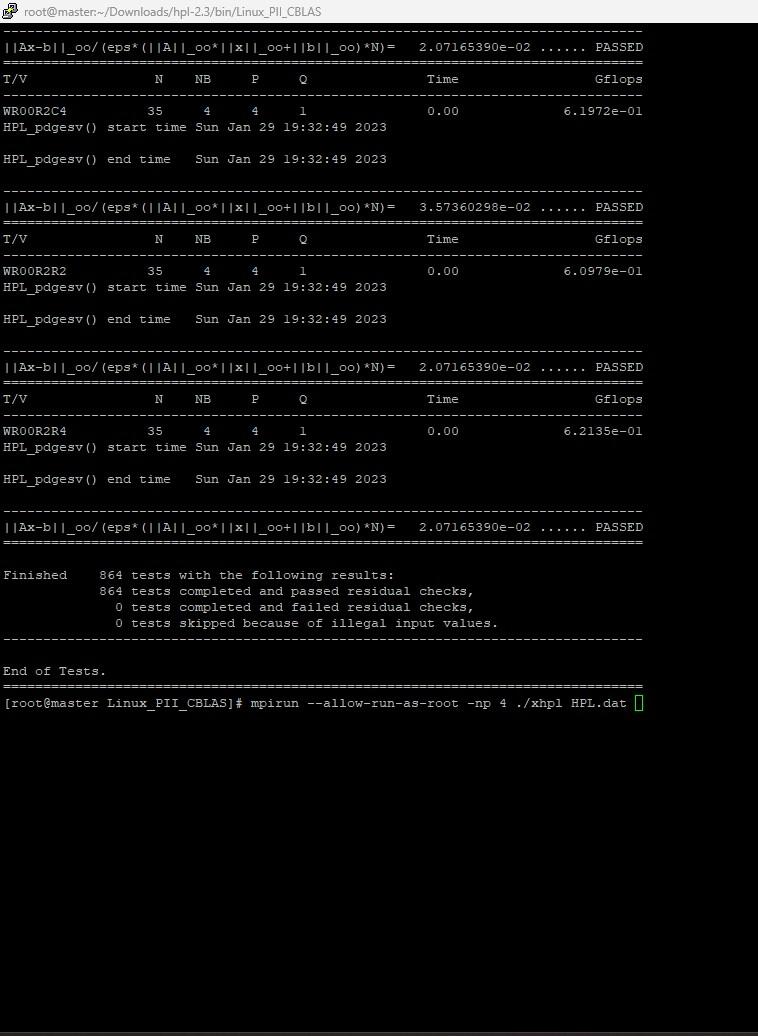
# cd /root/Downloads/hpl-2.3/bin/Linux\_PII\_CBLAS/

# vi HPL.dat



# mpirun --allow-run-as-root -np 4 ./xhpl HPL.dat





**Commands Used**

**Warewulf installation (Network Boot in HPC Cluster) Centos-7**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Pre-requisite:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_**

**We have to stop and disable firewall and disable selinux**

**sethostname of machine as master**

**# hostnamectl set-hostname master**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Check for the file of ens36 (if not there use #nmtui command and edit Wired Connection 1 to ens36)**

**# cat /etc/sysconfig/network-scripts/ifcfg-ens36**

**# ifconfig ens36**

**# vi /etc/hosts**

**-> edit -> <ip.address> master**

**# yum -y install yum-utils**

**# yum install http://build.openhpc.community/OpenHPC:/1.3/CentOS\_7/x86\_64/ohpc-release-1.3-1.el7.x86\_64.rpm**

**# yum repolist**

**# yum -y install ohpc-base**

**# yum -y install ohpc-warewulf**

**# yum -y install chrony**

**# vi /etc/chrony.conf**

**-> Edit this Conf. file -> server 192.168.23. iburst**

**-> allow 192.168.23.0/24 (uncomment and edit network address)**

**-> local stratum 10 (uncomment)**

**-> SAVE and Exit**

**# systemctl start chronyd**

**# systemctl enable chronyd**

**# yum install ntpdate**

**# ntpdate -q 192.168.23.130**

**# vi /etc/warewulf/provision.conf**

**edit -> change network device = ens36**

**# grep device /etc/warewulf/provision.conf**

**# vi /etc/xinetd.d/tftp**

**edit -> disable = no**

**# grep disable /etc/xinetd.d/tftp**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Resource Manager Installation**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**# yum -y install ohpc-slurm-server**

**# yum -y install slurm-sview-ohpc slurm-torque-ohpc**

**# vi /etc/slurm/slurm.conf**

**edit -> ClusterName=pearl**

**-> ControlMachine=master**

**-> NodeName=c[1-2]**

**-> Nodes=c[1-2] --> This is my nodename**

**# grep NodeName= /etc/slurm/slurm.conf**

**# echo ens36**

**# ifconfig ens36**

**# systemctl restart xinetd**

**# systemctl enable mariadb.service**

**# systemctl restart mariadb**

**# systemctl enable httpd.service**

**# systemctl restart httpd**

**# systemctl enable dhcpd.service**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**# echo ${CHROOT}**

**# export CHROOT=/opt/ohpc/admin/images/centos7.7**

**# echo ${CHROOT}**

**# wwmkchroot centos-7 $CHROOT -> Building initial BIOS image**

**# uname -r**

**# chroot ${CHROOT} uname -r**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**# yum -y --installroot=${CHROOT} update**

**# yum -y --installroot=${CHROOT} install \**

**ohpc-base-compute kernel kernel-headers kernel-devel kernel-tools parted \**

**xfsprogs python-devel yum htop ipmitool glibc\* perl perl-CPAN perl-CPAN \**

**sysstat gcc make xauth firefox squashfs-tools**

**# cat /etc/resolv.conf**

**# vi /etc/resolv.conf**

**add -> master 192.168.23.130**

**# cp -p /etc/resolv.conf $CHROOT/etc/resolv.conf**

**# yum -y --installroot=${CHROOT} install ohpc-slurm-client**

**# chroot ${CHROOT} systemctl enable slurmd**

**# yum -y --installroot=$CHROOT install chrony**

**# yum -y --installroot=$CHROOT install kernel lmod-ohpc**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**# # Initialize warewulf database and ssh\_keys**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**# wwinit database**

**# wwinit ssh\_keys**

**# df -hT | grep -v tmpfs**

**# hostname**

**# cat ${CHROOT}/etc/fstab**

**# echo "master:/home /home nfs nfsvers=3,nodev,nosuid 0 0" >> $CHROOT/etc/fstab**

**# echo "master:/opt/ohpc/pub /opt/ohpc/pub nfs nfsvers=3,nodev 0 0" >> $CHROOT/etc/fstab**

**# cat ${CHROOT}/etc/fstab**

**# cat /etc/exports**

**# echo "/home \*(rw,no\_subtree\_check,fsid=10,no\_root\_squash)" >> /etc/exports**

**# echo "/opt/ohpc/pub \*(ro,no\_subtree\_check,fsid=11)" >> /etc/exports**

**# cat /etc/exports**

**# systemctl start nfs-server**

**# systemctl status nfs-server**

**# systemctl enable nfs-server**

**# exportfs -arv**

**# chroot $CHROOT systemctl enable chronyd**

**# echo "server 192.168.23.130 iburst" >> $CHROOT/etc/chrony.conf**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Add Ganglia monitoring**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**# yum -y install ohpc-ganglia -> # Install Ganglia meta-package on master**

**# yum -y --installroot=${CHROOT} install ganglia-gmond-ohpc -> Install Ganglia compute node daemon**

**# Use example configuration script to enable unicast receiver on master host**

**-----------------------------------------------------------------------------**

**# cp /opt/ohpc/pub/examples/ganglia/gmond.conf /etc/ganglia/gmond.conf -> yes**

**# grep 'host =' /etc/ganglia/gmond.conf**

**# sed -i "s/<sms>/master/" /etc/ganglia/gmond.conf**

**# grep 'host =' /etc/ganglia/gmond.conf**

**# grep OpenHPC /etc/ganglia/gmond.conf**

**# sed -i "s/OpenHPC/pearl/" /etc/ganglia/gmond.conf**

**# grep pearl /etc/ganglia/gmond.conf**

**# cp /etc/ganglia/gmond.conf $CHROOT/etc/ganglia/gmond.conf -> yes**

**# echo "gridname pearl" >> /etc/ganglia/gmetad.conf**

**# grep gridname /etc/ganglia/gmetad.conf**

**# echo "**

**systemctl enable gmond**

**systemctl enable gmetad**

**systemctl start gmond**

**systemctl start gmetad**

**chroot ${CHROOT} systemctl enable gmond**

**" > /tmp/start\_ganglia\_service.sh**

**# bash /tmp/start\_ganglia\_service.sh**

**# grep "^date.timezone =" /etc/php.ini**

**# echo "date.timezone = Asia/Kolkata" >> /etc/php.ini**

**# grep "^date.timezone =" /etc/php.ini**

**# systemctl try-restart httpd**

**Go to browser : http://master/ganglia**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Add Nagios monitoring**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**# yum -y install ohpc-nagios -> Install Nagios meta-package on master host**

**# yum -y --installroot=$CHROOT install nagios-plugins-all-ohpc nrpe-ohpc -> Install plugins into compute node image**

**# chroot $CHROOT systemctl enable nrpe**

**# touch /var/log/nagios/nrpe.pid**

**# chown -R nrpe:nrpe /var/log/nagios/nrpe.pid**

**# perl -pi -e "s/^allowed\_hosts=/# allowed\_hosts=/" $CHROOT/etc/nagios/nrpe.cfg**

**# echo "nrpe 5666/tcp # NRPE" >> $CHROOT/etc/services**

**# echo "nrpe : 192.168.23.130 : ALLOW" >> $CHROOT/etc/hosts.allow**

**# echo "nrpe : ALL : DENY" >> $CHROOT/etc/hosts.allow**

**# chroot $CHROOT /usr/sbin/useradd -c "NRPE user for the NRPE service" -d /var/run/nrpe \**

**-r -g nrpe -s /sbin/nologin nrpe**

**# chroot $CHROOT /usr/sbin/groupadd -r nrpe**

**\*\*\*\***

**# # Configure remote services to test on compute nodes**

**------------------------------------------------------**

**# mv /etc/nagios/conf.d/services.cfg.example /etc/nagios/conf.d/services.cfg**

**# mv /etc/nagios/conf.d/hosts.cfg.example /etc/nagios/conf.d/hosts.cfg**

**# for ((i=0; i<2; i++)) ; do perl -pi -e "s/HOSTNAME$(($i+1))/${c[$i]}/ || s/HOST$(($i+1))\_IP/${c\_ip[$i]}/" /etc/nagios/conf.d/hosts.cfg; done**

**# perl -pi -e "s/ \/bin\/mail/ \/usr\/bin\/mailx/g" /etc/nagios/objects/commands.cfg**

**# perl -pi -e "s/nagios\@localhost/root\@master/" /etc/nagios/objects/contacts.cfg**

**# echo command[check\_ssh]=/usr/lib64/nagios/plugins/check\_ssh localhost >> $CHROOT/etc/nagios/nrpe.cfg**

**# htpasswd -bc /etc/nagios/passwd nagiosadmin nagios -> username : nagiosadmin | password: nagios**

**# chkconfig nagios on**

**# vi /etc/nagios/conf.d/hosts.cfg -> Add clients and hostname**

**# systemctl start nagios**

**# chmod u+s `which ping`**

**Go to browser : http://master/nagios**

**username : nagiosadmin**

**password : nagios**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**# wwsh file list**

**# wwsh file import /etc/passwd**

**# wwsh file import /etc/group**

**# wwsh file import /etc/shadow**

**# wwsh file list**

**# export WW\_CONF=/etc/warewulf/bootstrap.conf**

**# echo "drivers += updates/kernel/" >> $WW\_CONF**

**# echo "modprobe += ahci, nvme" >> $WW\_CONF**

**# echo "drivers += overlay" >> $WW\_CONF**

**# wwbootstrap `uname -r`**

**# echo ${CHROOT}**

**# wwvnfs --chroot $CHROOT**

**or**

**# wwvnfs --chroot /opt/ohpc/admin/images/centos7.7**

**# wwsh vnfs list**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**# echo "GATEWAYDEV=ens36" > /tmp/network.wwsh**

**# wwsh -y file import /tmp/network.wwsh --name network**

**# wwsh -y file set network --path /etc/sysconfig/network --mode=0644 --uid=0**

**# wwsh node new c1**

**# wwsh node set c1 --netdev ens36 --ipaddr=192.168.23.150 --hwaddr=00:0C:29:EC:16:C2 --netmask=255.255.255.0 --gateway 192.168.23.130**

**# wwsh node new c2**

**# wwsh node set c2 --netdev ens36 --ipaddr=192.168.23.151 --hwaddr=00:0C:29:B4:A4:C4 --netmask=255.255.255.0 --gateway 192.168.23.130**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**# wwsh node list**

**# wwsh -y provision set c1 --vnfs=centos7.7 --bootstrap=`uname -r` --files=dynamic\_hosts,passwd,group,shadow,network**

**# wwsh -y provision set c2 --vnfs=centos7.7 --bootstrap=`uname -r` --files=dynamic\_hosts,passwd,group,shadow,network**

**# systemctl restart dhcpd && wwsh pxe update**

**Script : For Restart/enable relevant services to support provisioning**

**---------------------------------------------------------------------**

**echo "**

**systemctl enable dhcpd**

**systemctl restart xinetd**

**systemctl enable mariadb**

**systemctl restart mariadb**

**systemctl enable httpd**

**systemctl restart httpd**

**" > /tmp/provisioning\_service\_run.sh**

**bash /tmp/provisioning\_service\_run.sh**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Installation of HPL Benchmarking (HPC-Cluster)**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**# yum install epel-release -y**

**# yum install atlas -y**

**# rpm -ql atlas**

**# wget https://netlib.org/benchmark/hpl/hpl-2.3.tar.gz**

**# mv hpl-2.3.tar.gz /root/Downloads/**

**# cd /root/Downloads**

**# tar -zxvf hpl-2.3.tar.gz**

**# ls**

**# cd hpl-2.3/**

**# ls**

**# cd setup**

**# vim Make.Linux\_Intel64**

**# wget https://download.open-mpi.org/release/open-mpi/v4.1/openmpi-4.1.4.tar.gz**

**# mv openmpi-4.1.4.tar.gz /root/Downloads/**

**# tar -xvf openmpi-4.1.4.tar.gz**

**# ls**

**# cd openmpi-4.1.4/**

**# ./configure --prefix=/opt/openmpi-4.1.4 --enable-orterun-prefix-by-default**

**# make -j 8**

**# make install**

**# echo $PATH**

**# export PATH=/opt/openmpi-4.1.4/bin/:$PATH**

**# mp <Press TAB KEY>**

**# export LD\_LIBRARY\_PATH=/opt/openmpi-4.1.4/bin:$LD\_LIBRARY\_PATH**

**# echo $LD\_LIBRARY\_PATH**

**# cd ~/Downloads/hpl-2.3/setup**

**# cp Make.Linux\_PII\_CBLAS /root/Downloads/hpl-2.3**

**# cd /root/Downloads/hpl-2.3/**

**# ls**

**# rpm -ql atlas**

**# vim Make.Linux\_PII\_CBLAS**

**>> edit # ----------------------------------------------------------------------**

**# - HPL Directory Structure / HPL library**

**# ----------------------------------------------------------------------**

**TOPdir = /root/Downloads/hpl-2.3**

**# ----------------------------------------------------------------------**

**# - Message Passing library (MPI)**

**# ----------------------------------------------------------------------**

**MPdir = /opt/openmpi-4.1.4**

**MPlib = $(MPdir)/lib/libmpi.so**

**# ----------------------------------------------------------------------**

**# - Compilers / linkers - Optimization flags**

**# ----------------------------------------------------------------------**

**#**

**CC = /usr/bin/gcc**

**LINKER = /usr/bin/gcc**

**#**

**# ----------------------------------------------------------------------**

**# - Linear Algebra library (BLAS or VSIPL)**

**# ----------------------------------------------------------------------**

**LAlib = $(LAdir)/libsatlas.so.3 $(LAdir)/libtatlas.so.3**

**>> <Escape Key> : wq**

**# make arch=Linux\_PII\_CBLAS**

**# cd /root/Downloads/hpl-2.3/bin/Linux\_PII\_CBLAS/**

**# ls**

**# vi HPL.dat**

**# mpirun --allow-run-as-root -np 4 ./xhpl HPL.dat**