**CDAC**

**PGDHPCSA**

**HPCSA CASE STUDY**

****

Submitted by-

**Abhishek Kumar 220940127001**

**Aishwarya 220940127002**

**Anita Apparao Khamitkar 220940127003**

**Ashutosh Kuril 220940127004**

**Divya Chaudhary 220940127005**

**Problem Statement-**

Build a Two node Disk-less HPC-Cluster using OpenHPC with xCAT,OpenLDAP, Slurm, Ganglia, HPL benchmark and Document the result.

**Introduction-**

The master node serves as the overall system management server (SMS) and is provisioned with CentOS7.7 and is subsequently configured to provision the compute node with xCAT in a stateless configuration.

**XCAT-**

In VMWare Create 1 Machine namely-masterwith HDD 100GB, RAM 15GB, two Network 1.NAT 2. HostOnly.

Commands on master:

# systemctl status firewalld

# systemctl stop firewalld

# systemctl disable firewalld

* Disabling Selinux

# vi /etc/selinux/config

# getenforce

# setenforce 0

# getenforce

# vi /etc/selinux/config

# syslinux=disabled

# reboot

# getenforce

* Enable use of the public xCAT repository by adding it to the local list of available package repositories. This also requires network access from your master server to the internet, or alternatively, that the repository be mirrored locally

# yum install yum-utils

# cat /etc/yum.conf

# wget -P /etc/yum.repos.d https://xcat.org/files/xcat/repos/yum/latest/xcat-core/xcat-core.repo

# yum install xCAT

# yum update

# ll /etc/profile.d/xcat.sh



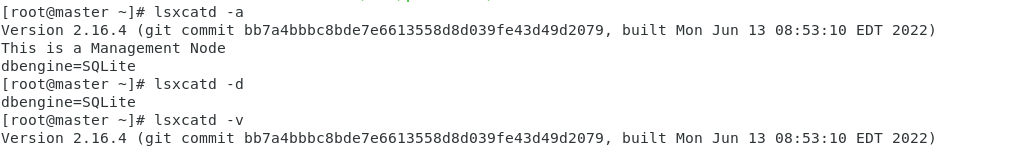
# . /etc/profile.d/xcat.sh

# echo $PATH

# lsxcatd -a

#  lsxcatd -d

#  lsxcatd –v



#  tabdump site

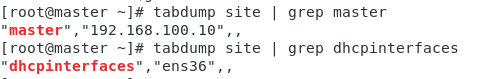
#  chdef -t site dhcpinterfaces="ens36"

* Assigning host only IP to master

# chdef -t site master="192.168.100.10”

#  tabdump site | grep master

#  tabdump site | grep dhcpinterfaces



#  lsblk

#  lsb\_release

#  cat /etc/os-release

#  dd if=/dev/sr0 of=/root/Centos7.iso

#  ll –h

* Building a default image for use with xCAT. To begin, we will first need to have a local copy of the ISO image available for the underlying OS. The relevant ISO image is CentOS-7-x86 64-minimal.iso (available from the CentOS mirrors). We initialize the image creation process using the copycds command assuming that the necessary ISO image is available locally.

#  copycds /root/Centos7.iso

* Once completed, several OS images should be available for use within xCAT. These can be queried via:

#  lsdef -t osimage

* we leverage the stateless (netboot) image for compute nodes and proceed by using genimage to initialize a chroot-based install. Note that the previous query highlights the existence of other provisioning images as well.

#  genimage centos7.9-x86\_64-netboot-compute

#  mkdir -p /install/custom/netboot/

#  lsdef -t osimage centos7.9-x86\_64-netboot-compute

# chdef -t osimage centos7.9-x86\_64-netboot-compute synclists="/install/custom/netboot/compute.synclist"

* Syncing users, groups and passwords

#  echo "/etc/passwd -> /etc/passwd" >> /install/custom/netboot/compute.synclist

#  echo "/etc/group -> /etc/group" >> /install/custom/netboot/compute.synclist

#  echo "/etc/hosts -> /etc/hosts" >> /install/custom/netboot/compute.synclist

#  echo "/etc/shadow -> /etc/shadow" >> /install/custom/netboot/compute.synclist

* To finalize the xCAT provisioning configuration, this section first highlights packing of the stateless image from the chroot environment followed by the registration of desired compute nodes. To assemble the final compute image use packimage as follows:

#  packimage centos7.9-x86\_64-netboot-compute

* Assigning IP in range of host only IP and mac address of newly created VM machine (compute node)

#  mkdef -t node cn00 groups=compute,all ip=192.168.100.115 mac=00:0C:29:0F:1F:93 netboot=xnba

* xnba-neworkboot loader

#  lsdef cn00

#  chdef -t node node1 provmethod=centos7.9-x86\_64-netboot-compute

#  chdef -t group compute provmethod=centos7.9-x86\_64-netboot-compute

#  lsdef node1

#  chdef -t site domain=xcat.in

#  cat /etc/hosts



#  makehosts

#  makenetworks

#  makedhcp -n

#  systemctl start dhcpd

#  makedns -n

#  yum install dhcp\*

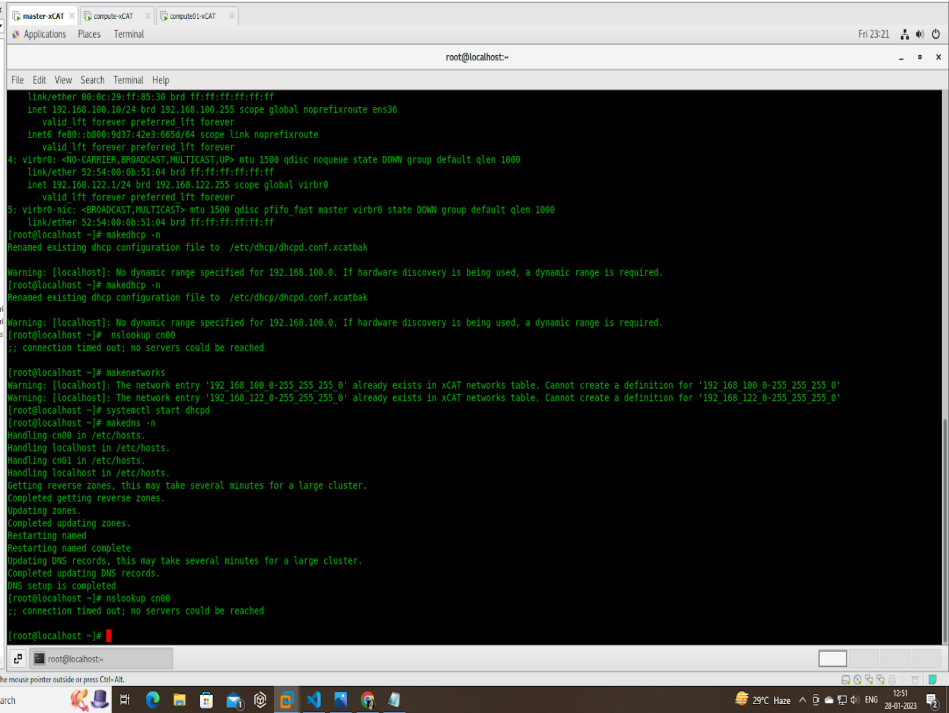
#  makedhcp -n

#  makedns

#  nslookup cn00

#  vi /etc/resolv.conf

#  nslookup cn00



#  vi /etc/resolv.conf

Generated by NetworkManager

search localdomain xcat.in

nameserver 192.168.100.10

nameserver 192.168.207.2

#  lsdef -t osimagewe

#  nodeset compute osimage=centos7.9-x86\_64-netboot-compute

#  systemctl restart dhcpd

#  systemctl start dhcpd

#  makedhcp -n

#  nodeset compute osimage=centos7.9-x86\_64-netboot-compute

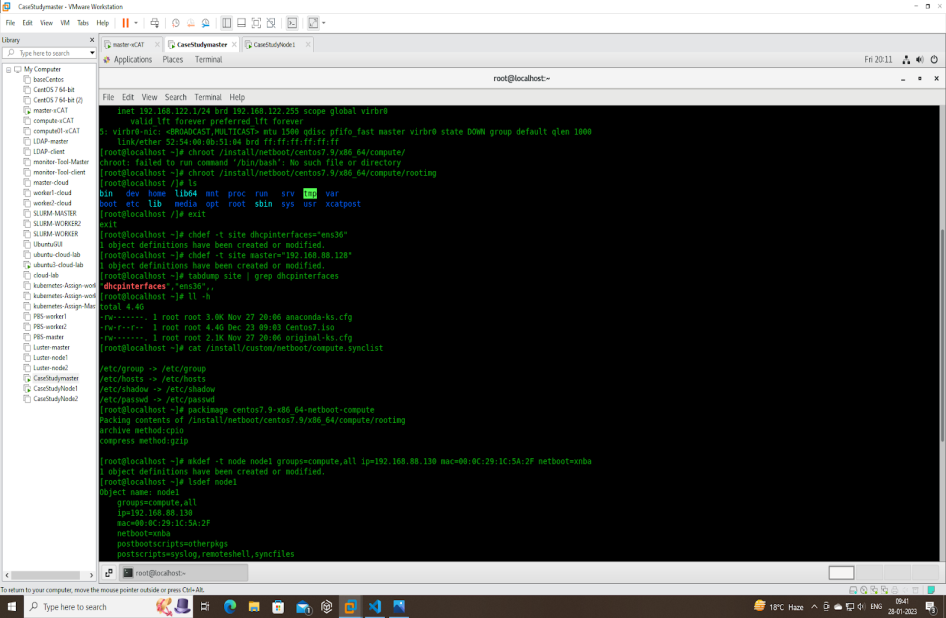
#  makedhcp cn00

#  systemctl status dhcpd

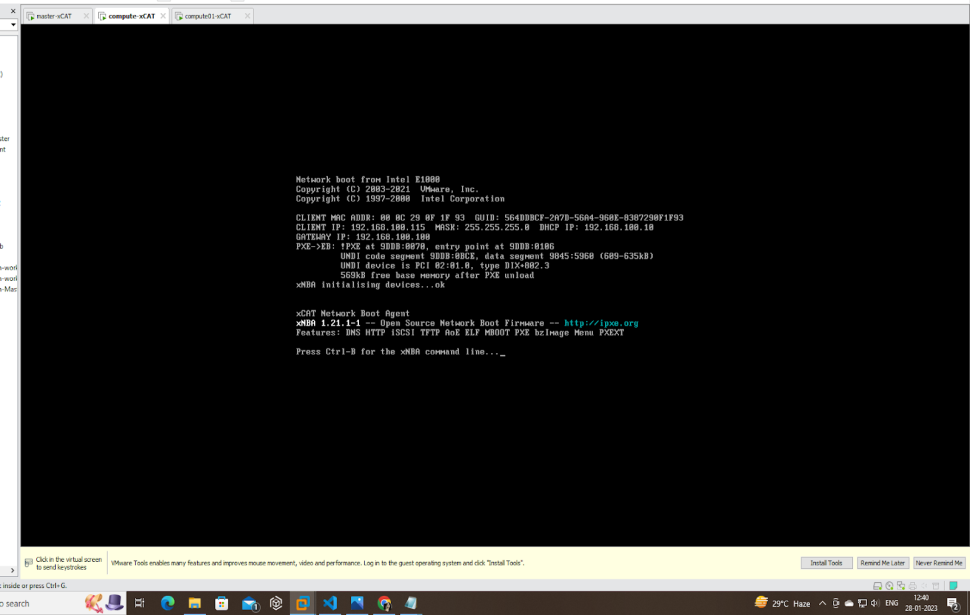
#  systemctl restart dhcpd

#  systemctl status dhcpd

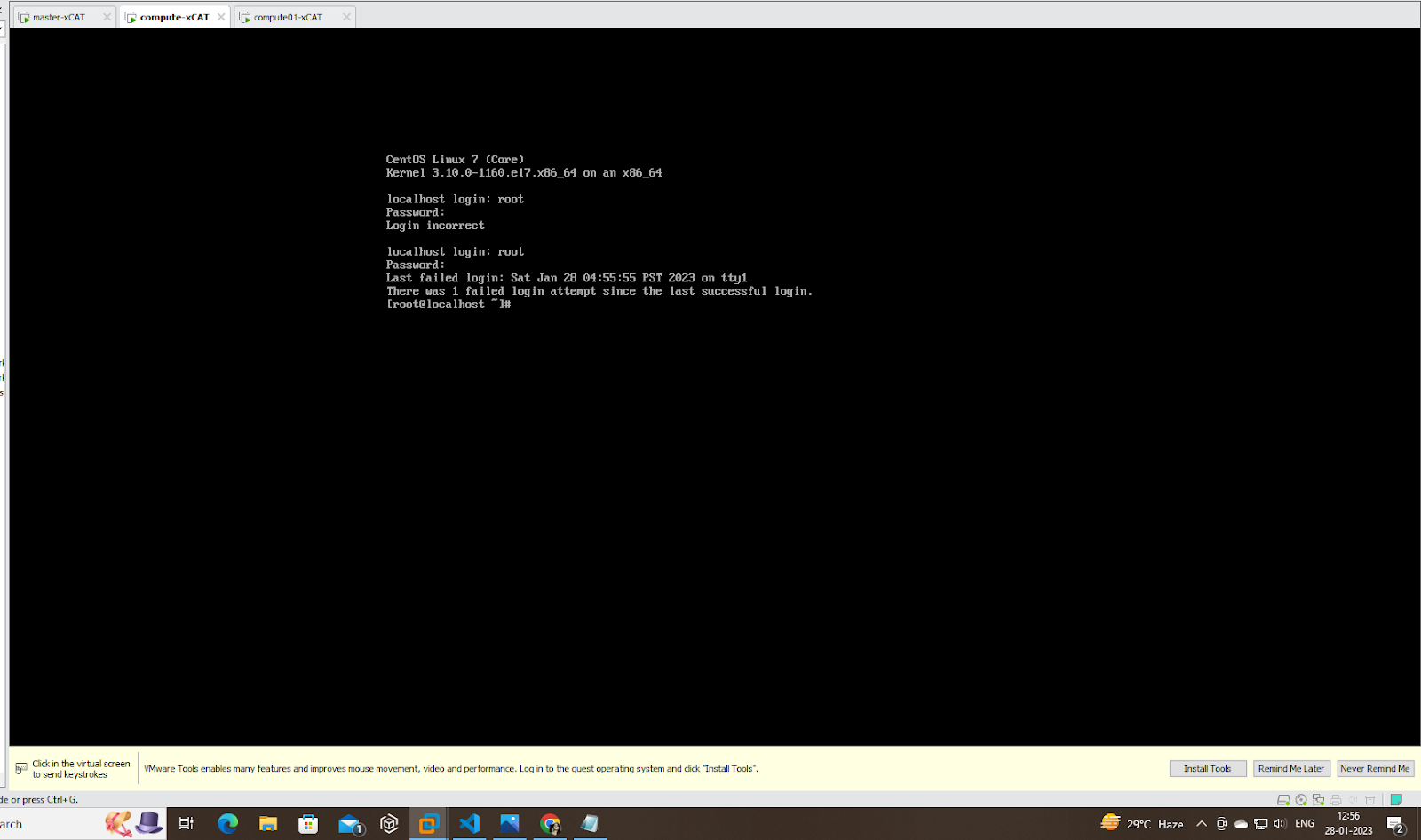
#  chdef -t osimage -o centos7.9-x86\_64-netboot-compute synclists="/install/custom/netboot/compute.synclist"



* Booting Compute node through XCAT-



* Successfully booted via XCAT on node1



**SLURM-**

Installing Slurm via XCAT is a process of configuring the Slurm workload manager to run on a cluster managed by XCAT. It involves setting up XCAT as the frontend node and the compute nodes, configuring the Slurm daemons (such as slurmd and slurmctld), and defining the Slurm partition and nodes.

Commands for master

# export CHROOT=/install/netboot/centos7.9/x86\_64/compute/rootimg/

# wget<https://download.schedmd.com/slurm/slurm-22.05.8.tar.bz2>



#yum install mariadb-server mariadb-devel –y

#yum install epel-release

#yum --installroot=$CHROOT install epel-release

# yum install munge munge-libs munge-devel –y

# yum --installroot=$CHROOT install munge munge-libs munge-devel –y

# yum install rpm-build

# yum install python3 readline-devel pam-devel

# rpmbuild -ta slurm-22.05.8.tar.bz2

# yum install gcc

* Creating munge key

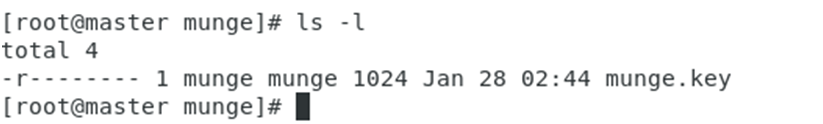
# /usr/sbin/create-munge-key –r



* Copying munge.key to node

# scp /etc/munge/munge.key $CHROOT/etc/munge

#ls -l



* Changing permission and ownership of the munge.key on both nodes

# chown -R munge:munge /etc/munge/

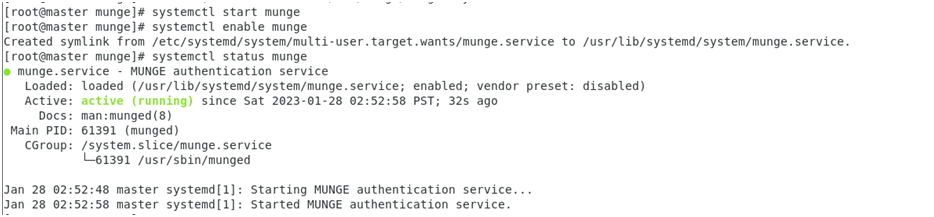
# chroot $CHROOT chown -R munge:munge /etc/munge/

# chroot $CHROOT chmod 400 /etc/munge/munge.key

# systemctl start munge

# systemctl enable munge

# systemctl status munge



* On master

# scp -r /root/rpmbuild/RPMS/x86\_64/ $CHROOT/home

# chroot $CHROOT

# cd /home/x86\_64

* Removing Slurmctld service from compute node

# rm –rf slurm-slurmctld-22.05.8-1.el7.x86\_64.rpm

# yum --installroot=$CHROOT install slurm\*

* On  both nodes-

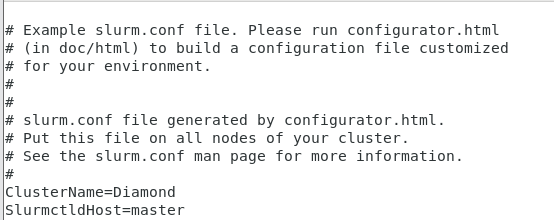
# export SLURMUSER=1500

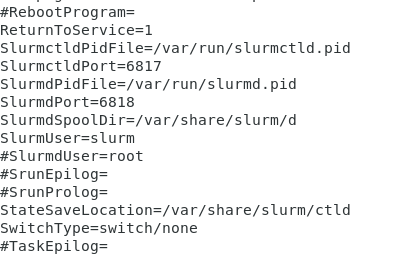
# groupadd -g $SLURMUSER slurm

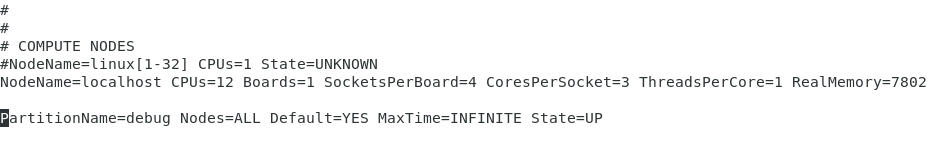
# useradd  -m -c "SLURM workload manager" -d /var/lib/slurm -u $SLURMUSER -g slurm  -s /bin/bash slurm

# cp /etc/slurm/slurm.conf.example /etc/slurm.conf

# vi /etc/slurm/slurm.conf







# mkdir -p /var/share/slurm/ctld

# chown -R slurm:slurm /var/share/slurm

# touch /var/log/slurmctld.log

# systemctl start slurmd

# systemctl enable slurmd

# systemctl start slurmctld

# systemctl enable slurmctld

# chroot $CHROOT mkdir –p /var/share/slurm/d

# chroot $CHROOT chown –R slurm:slurm /var/share/slurm

# chroot $CHROOT touch /var/log/slurmd.log

# cp /etc/slurm/cgroup.conf.example /etc/slurm/cgroup.conf

# scp /etc/slurm/cgroup.conf $CHROOT/etc/slurm

# systemctl start slurmctld

# systemctl enable slurmctld

# systemctl start munge

# systemctl enable munge

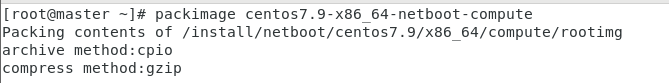
# chroot $CHROOT systemctl enable slurmctld

# chroot $CHROOT systemctl enable munge

Slurm configuration done successfully-

Packaging the image –

# packimage centos7.9-x86\_64-netboot-compute



**LDAP**

* Install OpenLDAP on master

# yum -y install openldap-servers openldap-clients

# cp /usr/share/openldap-servers/DB\_CONFIG.example /var/lib/ldap/DB\_CONFIG

# chown ldap. /var/lib/ldap/DB\_CONFIG

# systemctl start slapd

# systemctl enable slapd

# generate encrypted password

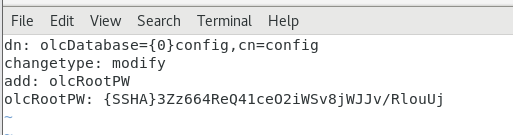
* Saving the password from below command for further authentication

# slappasswd

{SSHA}3Zz664ReQ41ceO2iWSv8jWJJv/RlouUj

# vi chrootpw.ldif

* Adding the above generated password in below file



# ldapadd -Y EXTERNAL -H ldapi:/// -f chrootpw.ldif

# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/cosine.ldif

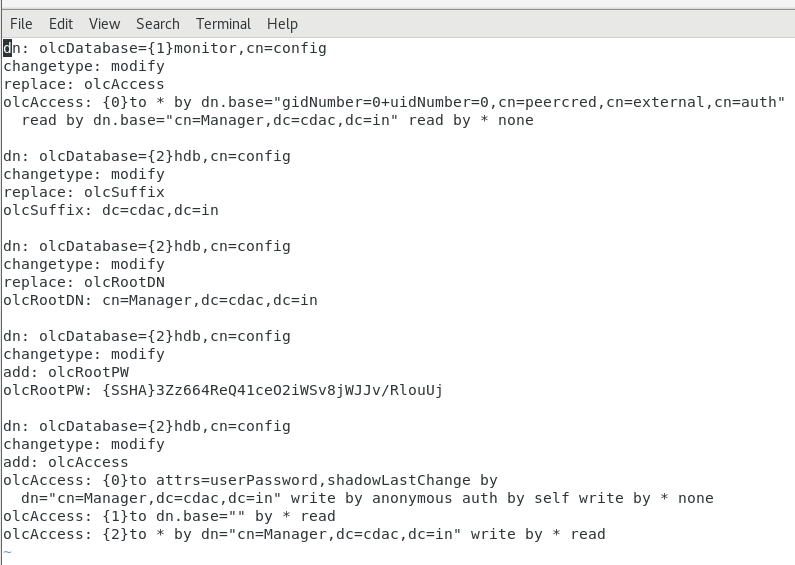
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/nis.ldif

# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/inetorgperson.ldif

* Generate directory manager's password

# vi chdomain.ldif

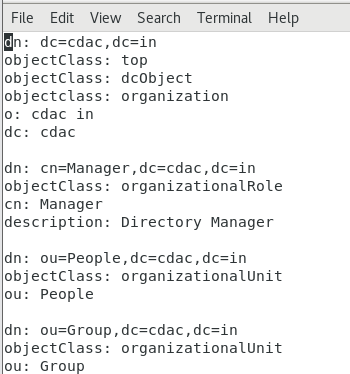
* replace to your own domain name for "dc=\*\*\*,dc=\*\*\*" section
* specify the password generated above for "olcRootPW" section



# ldapmodify -Y EXTERNAL -H ldapi:/// -f chdomain.ldif

# vi basedomain.ldif

* replace to your own domain name for "dc=\*\*\*,dc=\*\*\*" section

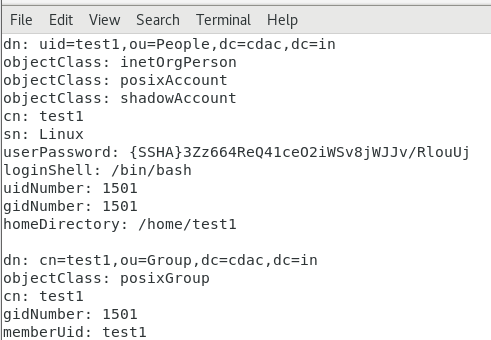


# ldapadd -x -D cn=Manager,dc=cdac,dc=in -W -f basedomain.ldif

* Adding user account

# vi ldapuser.ldif

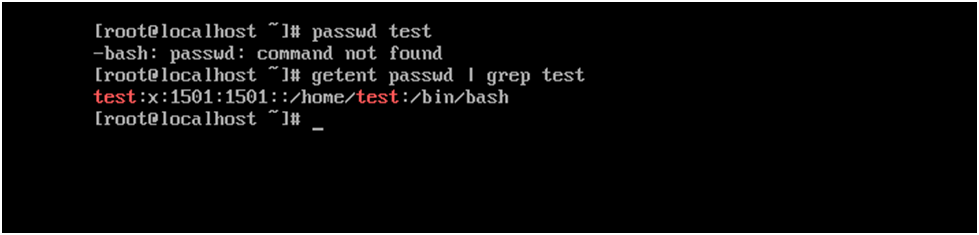
* create new
* replace to your own domain name for "dc=\*\*\*,dc=\*\*\*" section



# ldapadd -x -D cn=Manager,dc=cdac,dc=in -W -f ldapuser.ldif

* If you'd like to delete LDAP User or Group, Do as below.

# ldapdelete -x -W -D 'cn=Manager,dc=cdac,dc=in' "uid=test,ou=People,dc=cdac,dc=in"



* Install OpenLDAP Client.

# yum --installroot=/install/netboot/centos7.9/x86\_64/compute/rootimg install openldap-clients

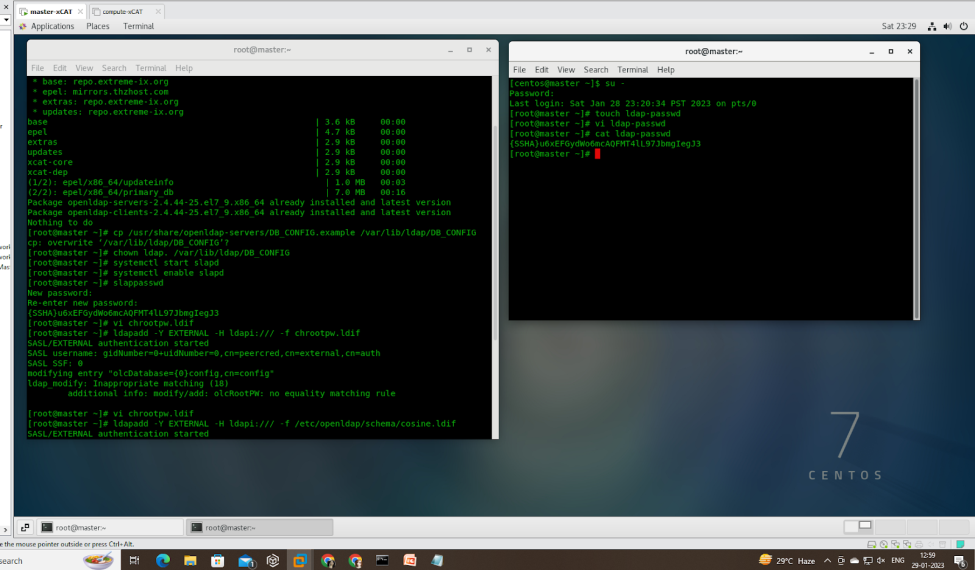
# exports CHROOT=/install/netboot/centos7.9/x86\_64/compute/rootimg

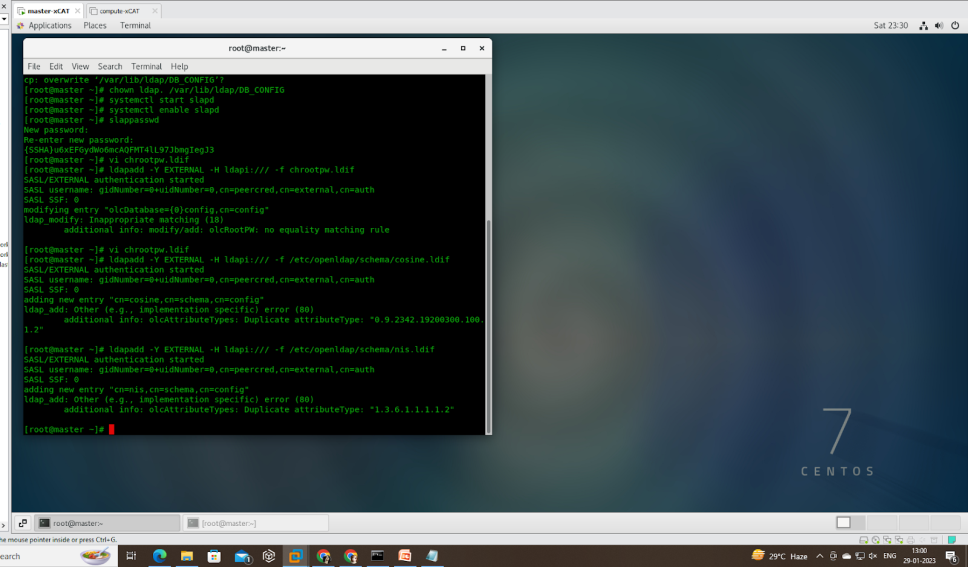
# chroot $CHROOT

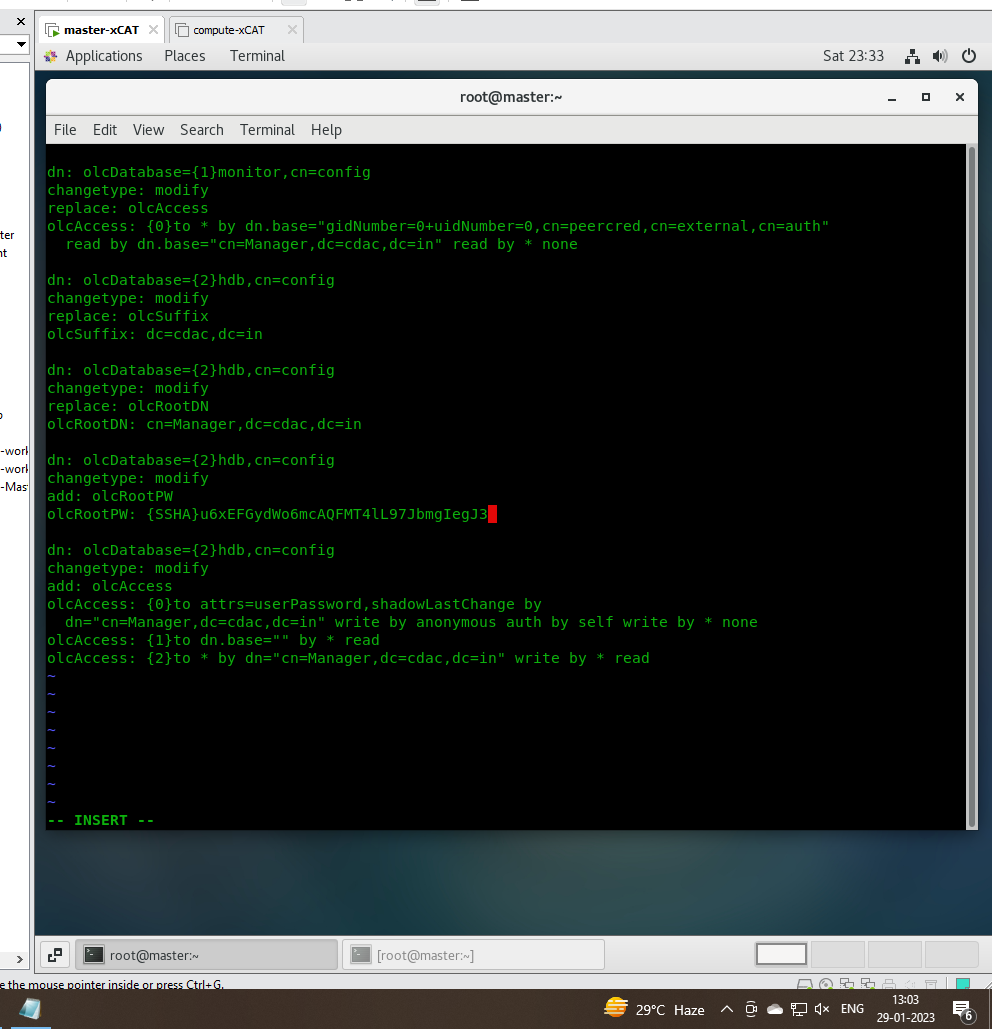
# authconfig --enableldap --enableldapauth --ldapserver=master --ldapbasedn="dc=cdac,dc=in" --enablemkhomedir --update

# systemctl restart nslcd

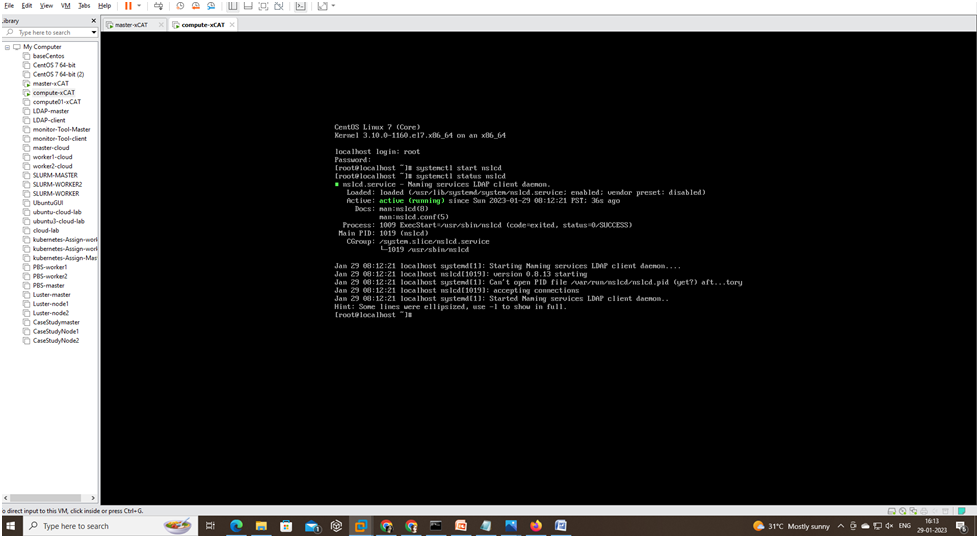
# ldapsearch -x







Service started successfully on node —

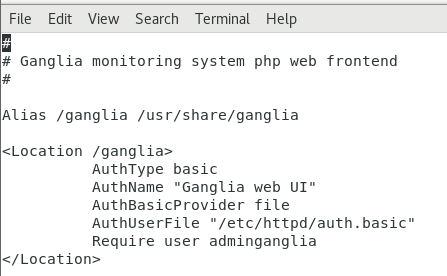


**GANGLIA**

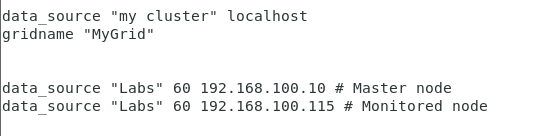
# yum install ganglia rrdtool ganglia-gmetad ganglia-gmond ganglia-web

# htpasswd -c /etc/httpd/auth.basic adminganglia

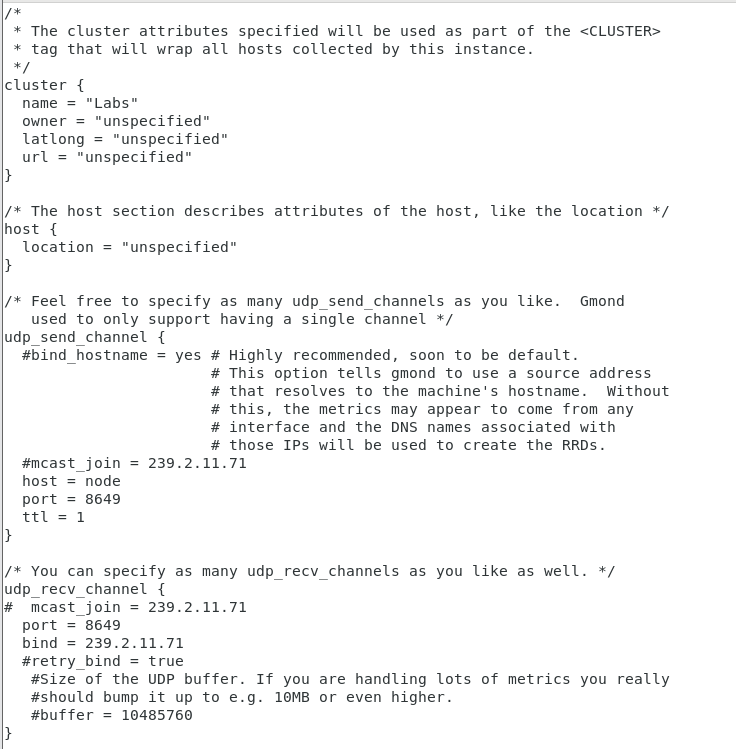
# vi /etc/httpd/conf.d/ganglia.conf



# vi /etc/ganglia/gmetad.conf



# vi /etc/ganglia/gmond.conf



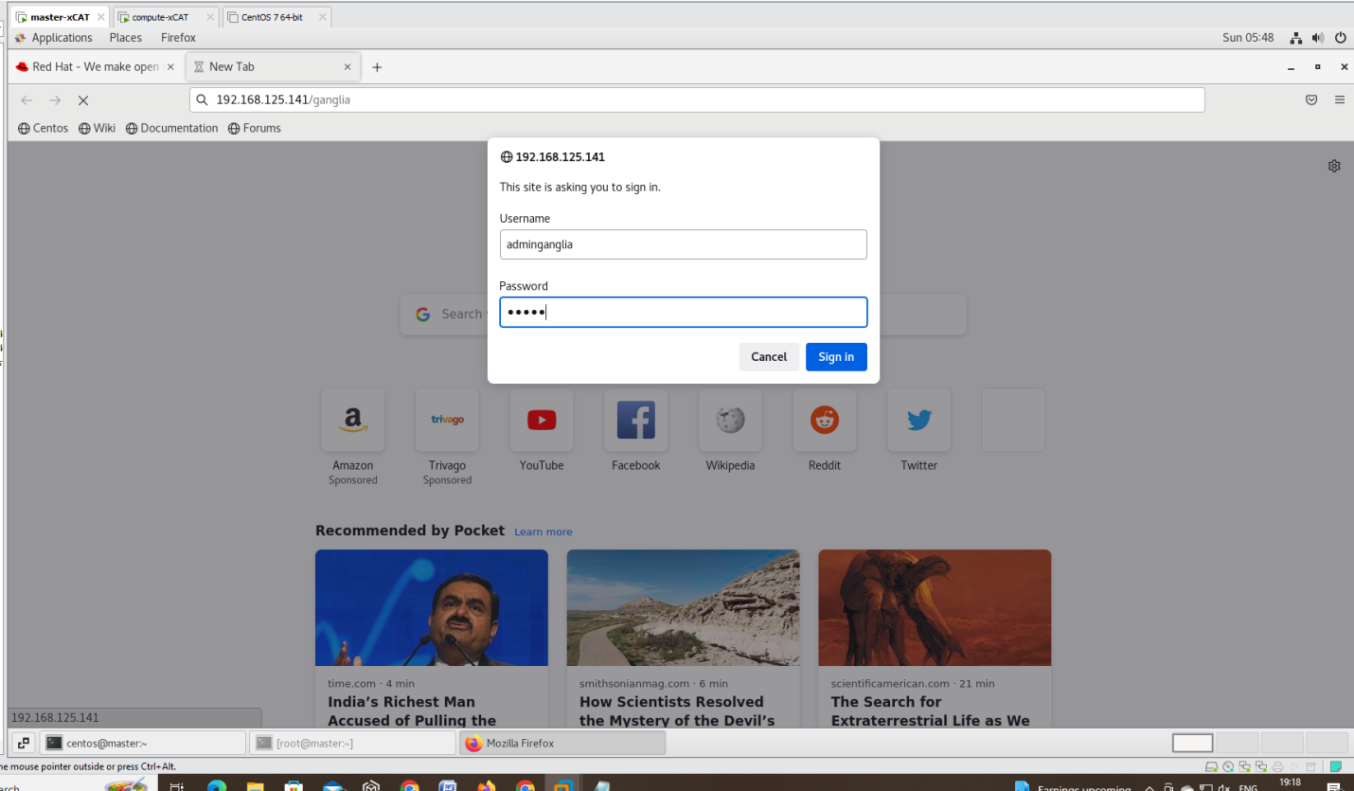
# setsebool -P httpd\_can\_network\_connect 1

# systemctl restart httpd gmetad gmond

# systemctl enable httpd gmetad httpd

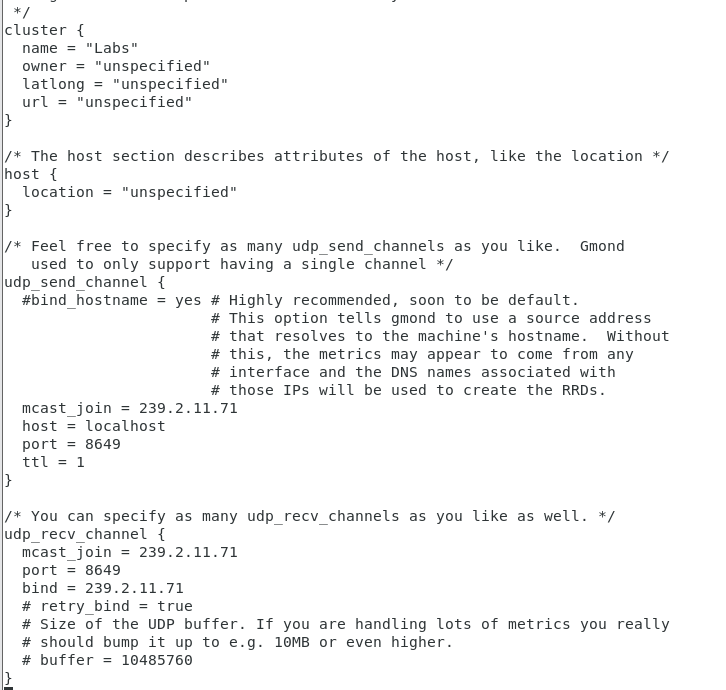
# ip a

* Browser-----> http://<ganglia\_master\_ip>/ganglia



# yum --installroot=$CHROOT install ganglia rrdtool ganglia-gmetad ganglia-gmond ganglia-web

# vi /etc/ganglia/gmond.conf



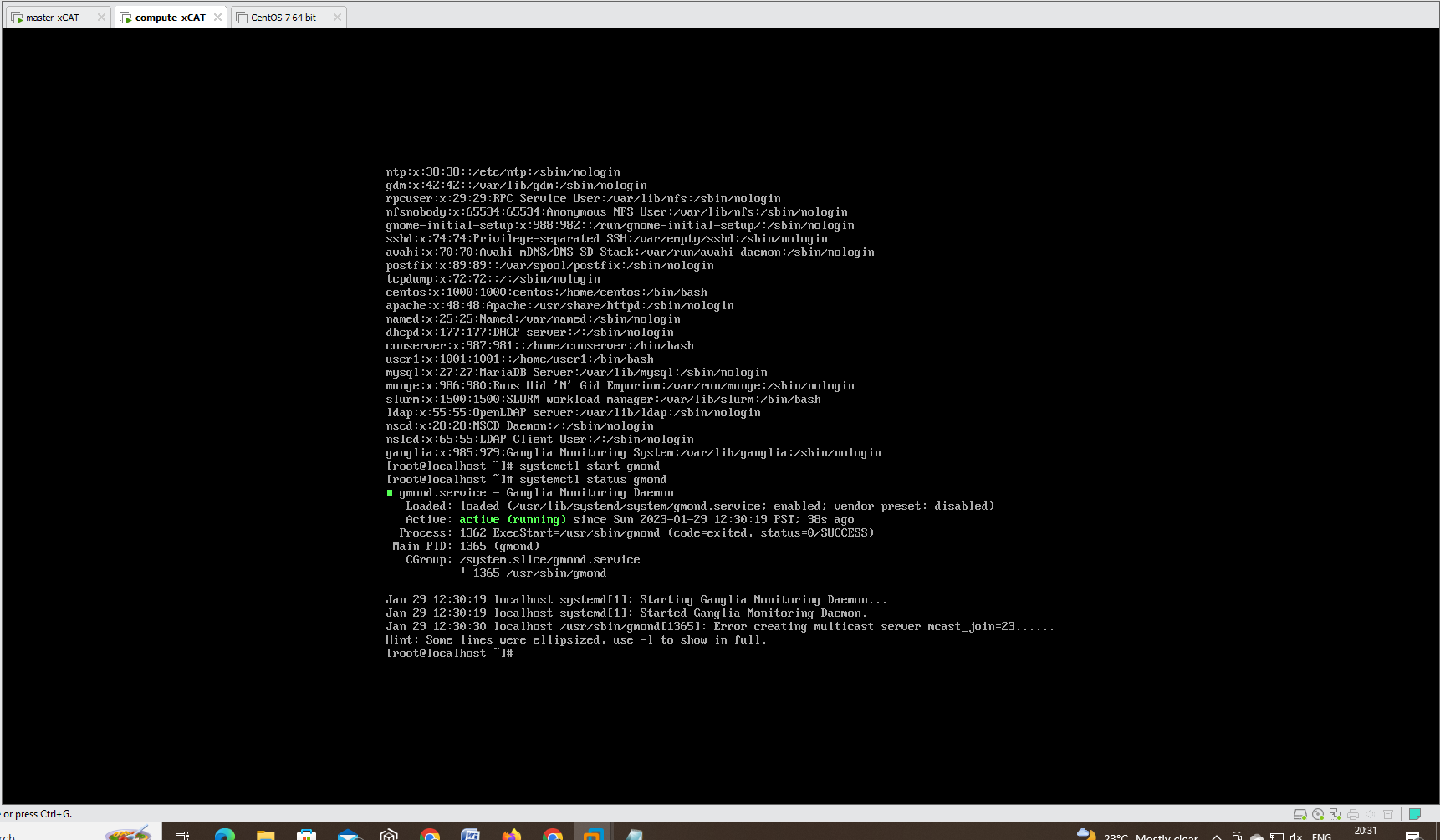
# chroot $CHROOT

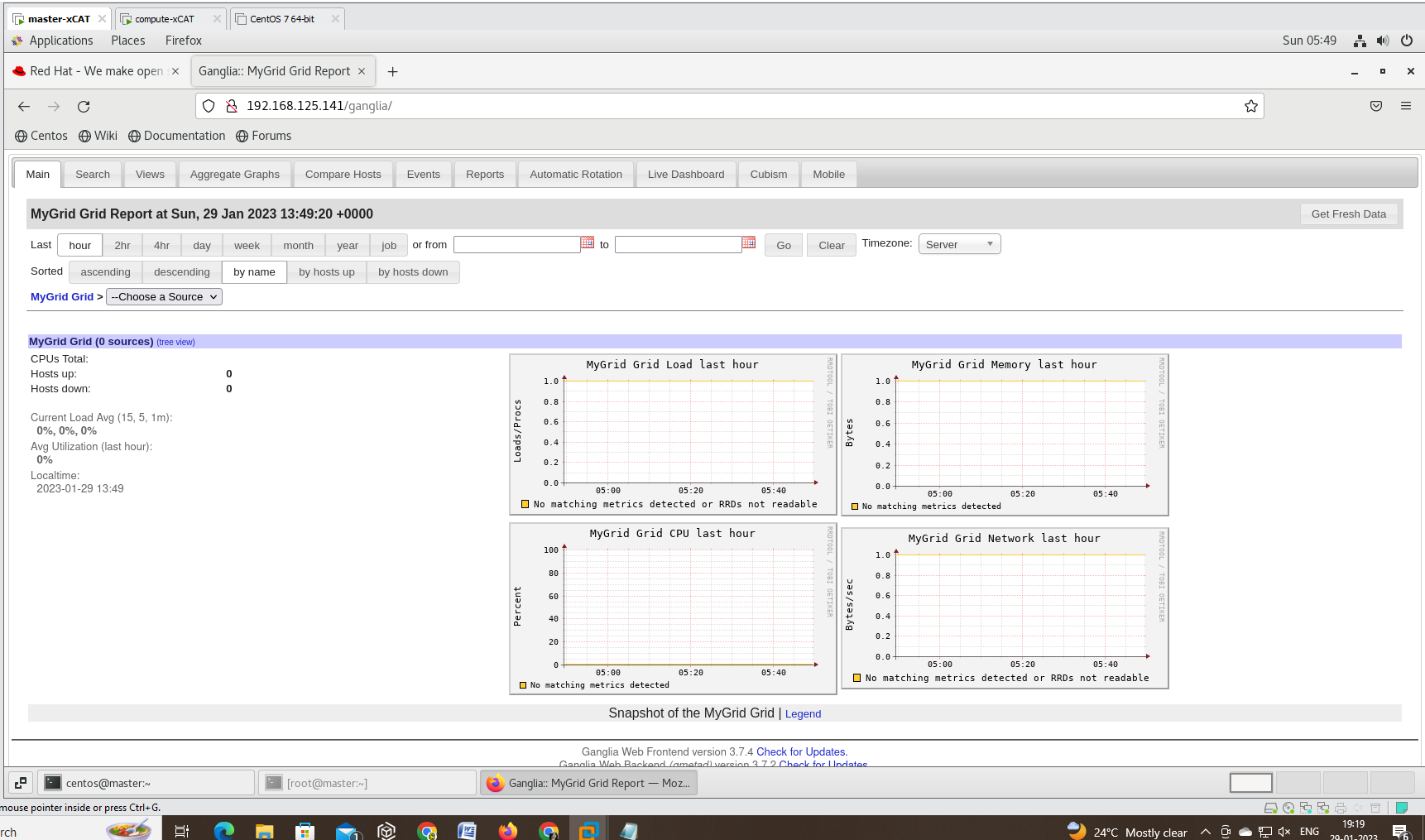
# systemctl enable gmond

# systemctl restart httpd gmetad gmond

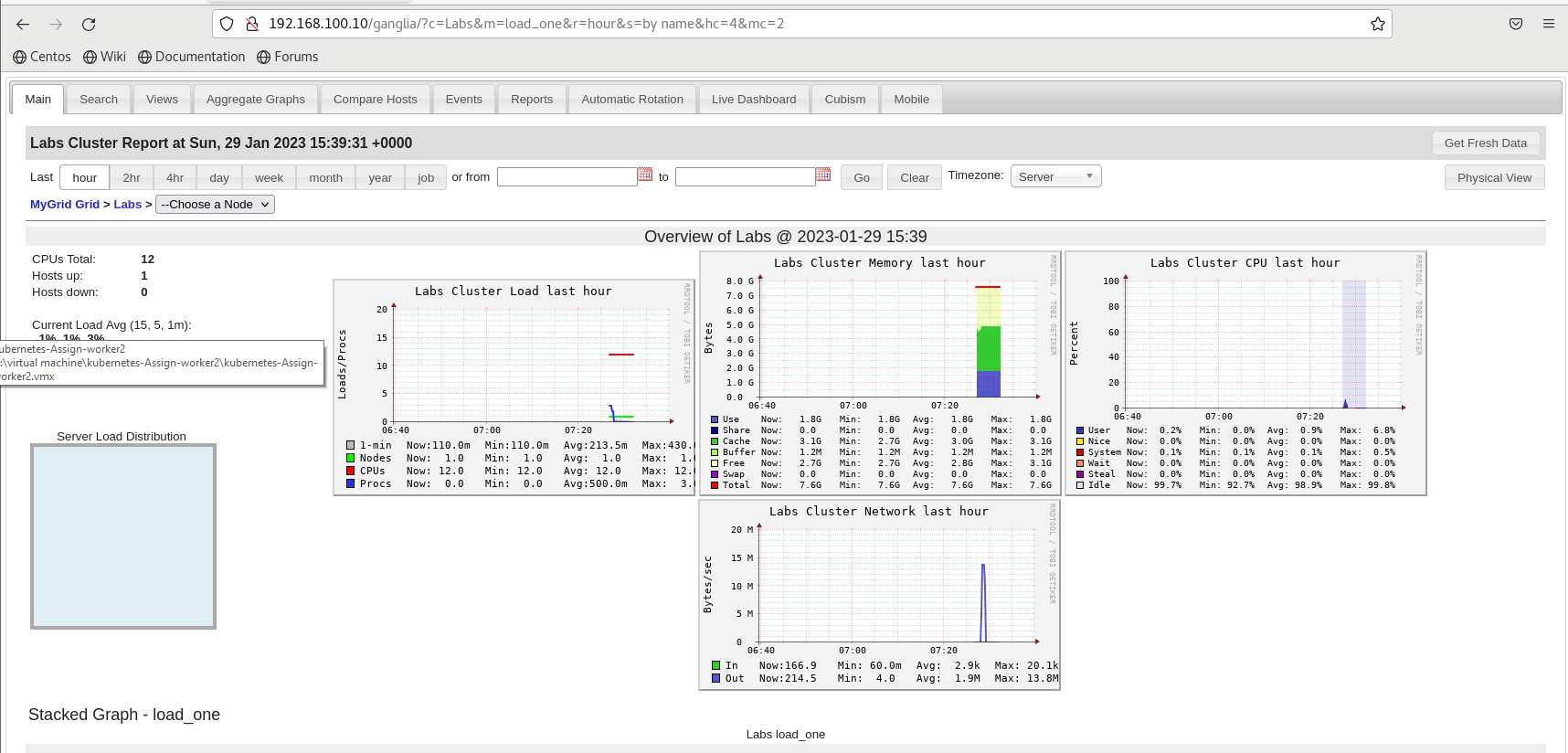
# systemctl enable httpd gmetad httpd

# packimage centos7.9-x86\_64-netboot-compute

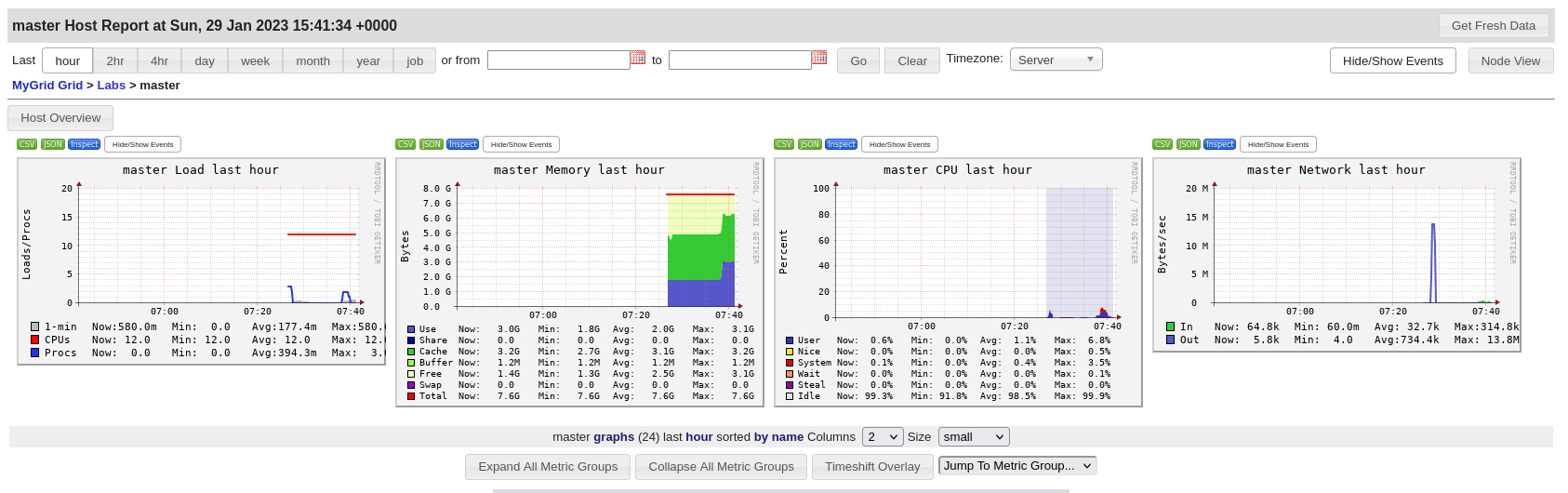




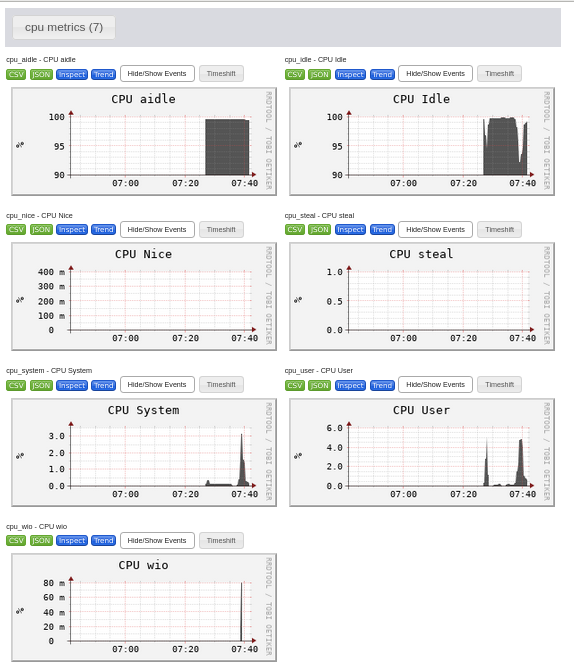
* Cluster labs-



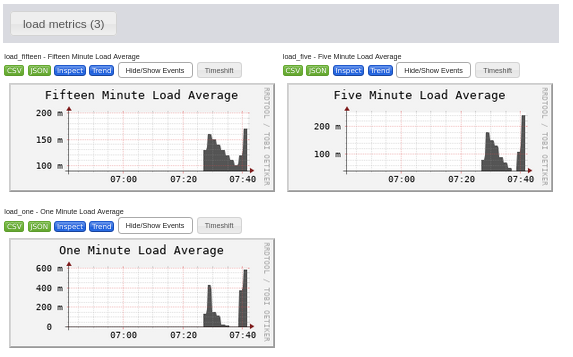
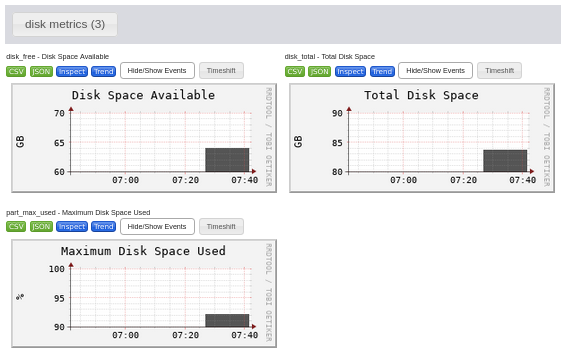
* Master-



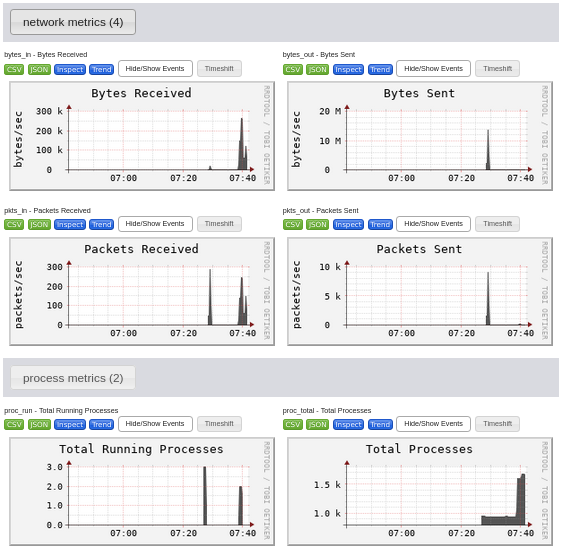
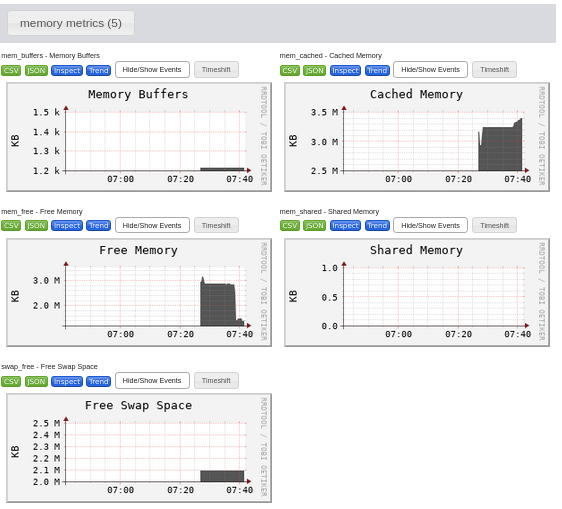
* CPU metrics of master-



* Disk and Load metrics of master



* Memory and network metrics of the master-



**HPL BENCHMARKING**

# yum install epel-release

# yum install atlas

* Download below file from ->google->hpl netlib->hpl-2.3.tar.gz save in /etc/yum.repos.d

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

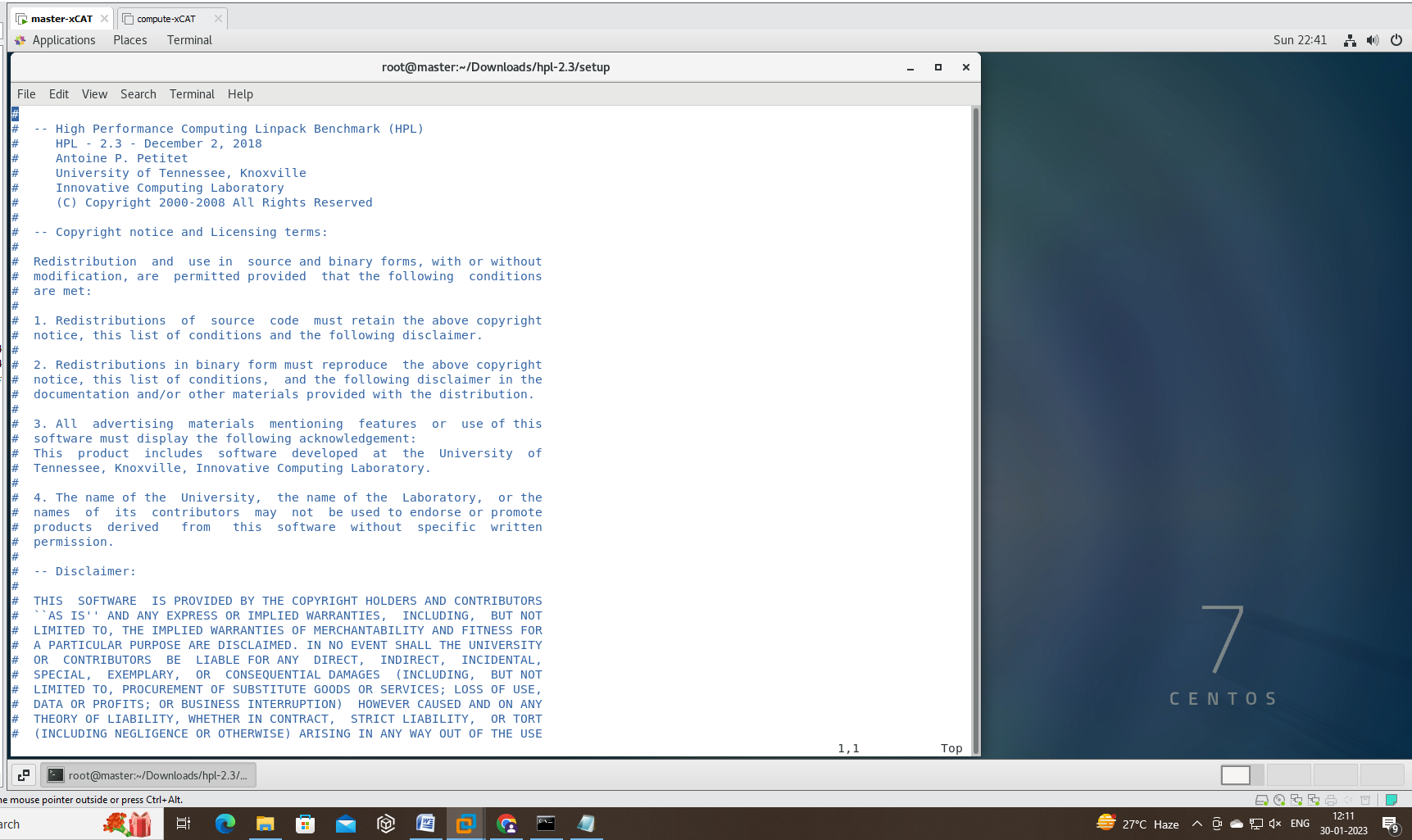
# tar -xf /hpl-2.3.tar.gz

# ls /hpl-2.3

# cd /hpl-2.3/setup/

# ls

# cat Make.Linux\_Intel64



# cp /root/Downloads/hpl-2.3/setup/Make.Linux\_Intel64 /root/Downloads/hpl-2.3

# rpm -ql

# rpm -ql atlas

* Download below file from ->google->Open MPI: Version 4.1 -> openmpi-4.1.4.tar.gz save in /etc/yum.repos.d

# tar -xf openmpi-4.1.4.tar.gz

# ls

* if not execute install gcc yum install gcc

# yum install gcc-c++

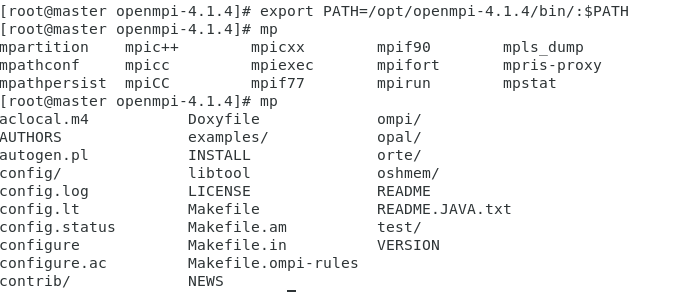
# ./configure --prefix=/opt/openmpi4.1.4 --enable-orterun-prefix-by-default

# make –43j 8

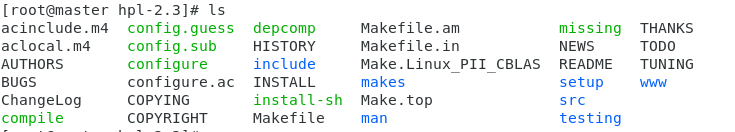
# make install

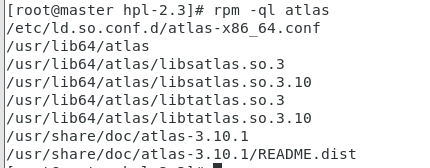
# echo $PATH

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



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





#vi Make.Linux\_PII\_CBLAS

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

# - shell --------------------------------------------------------------

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

#

SHELL = /bin/bash

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

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

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

#

TOPdir = /root/Downloads/hpl-2.3

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

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

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

*#*

MPdir = /opt/openmpi4.1.4

MPlib = $(MPdir)/lib/libmpi.so.40

#

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

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

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

#

CC = /usr/bin/gcc

#

LINKER = /usr/bin/gcc

#

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

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

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

#

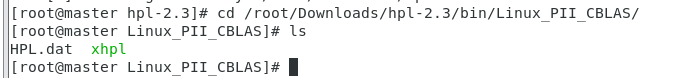
LAdir = /usr/lib64/atlas

LAlib = $(LAdir)/libsatlas.so.3 $(LAdir)/libtatlas.so.3.10

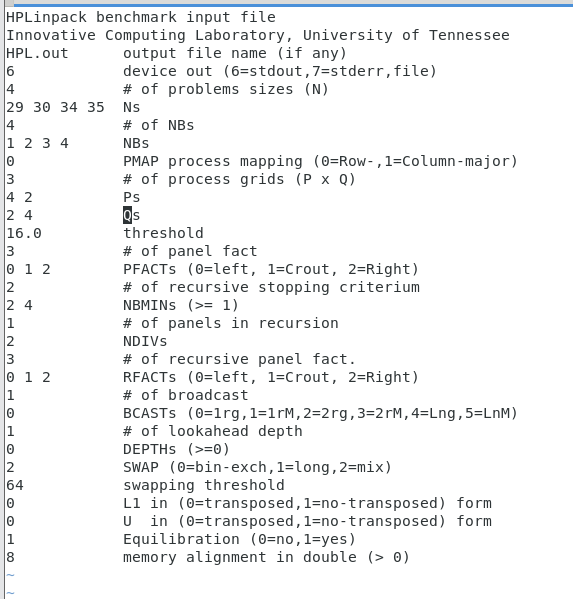
#

# make arch=Linux\_PII\_CBLAS

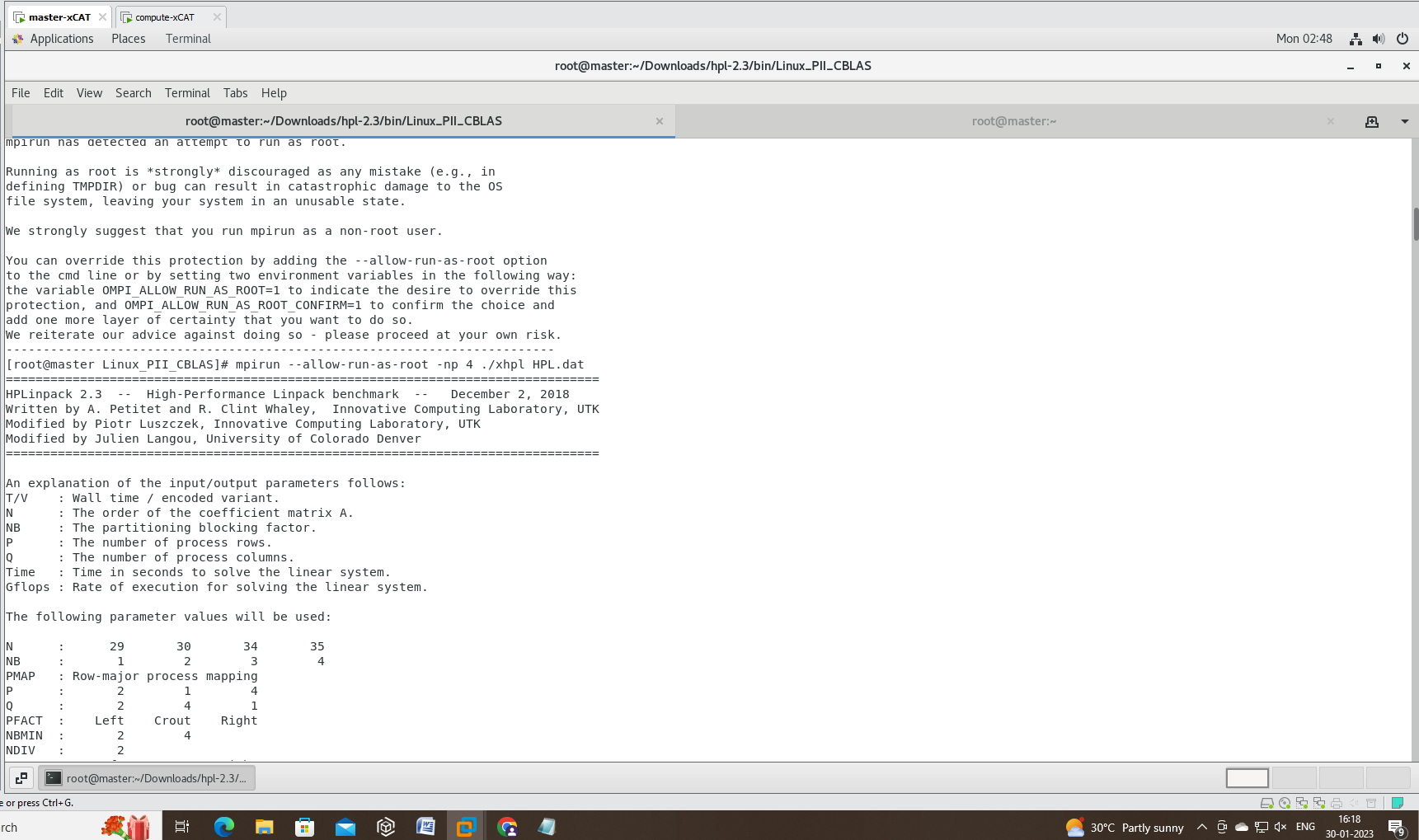
#cd /root/Downloads/hpl-2.3/bin/Linux\_PII\_CBLAS/

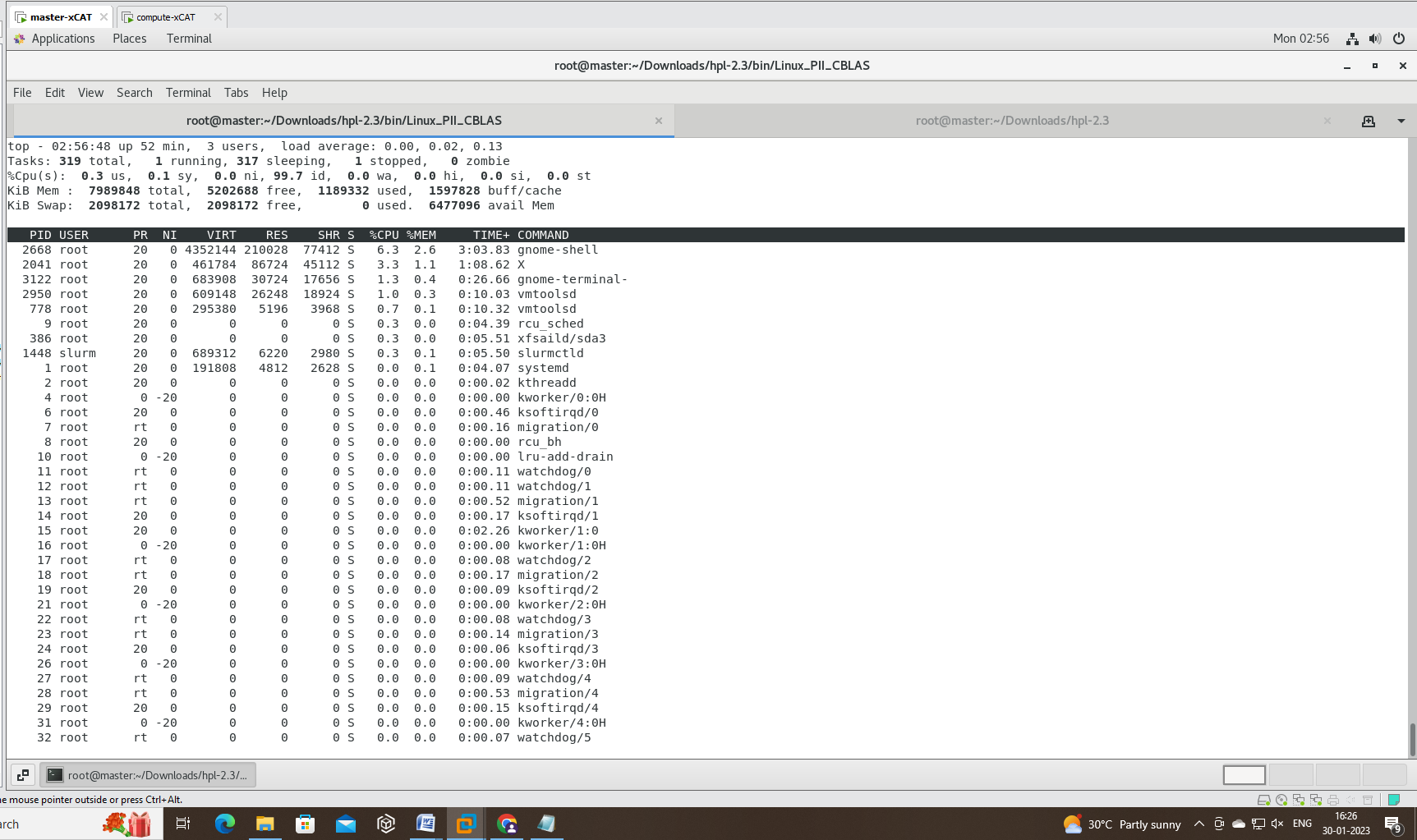


#vi HPL.dat

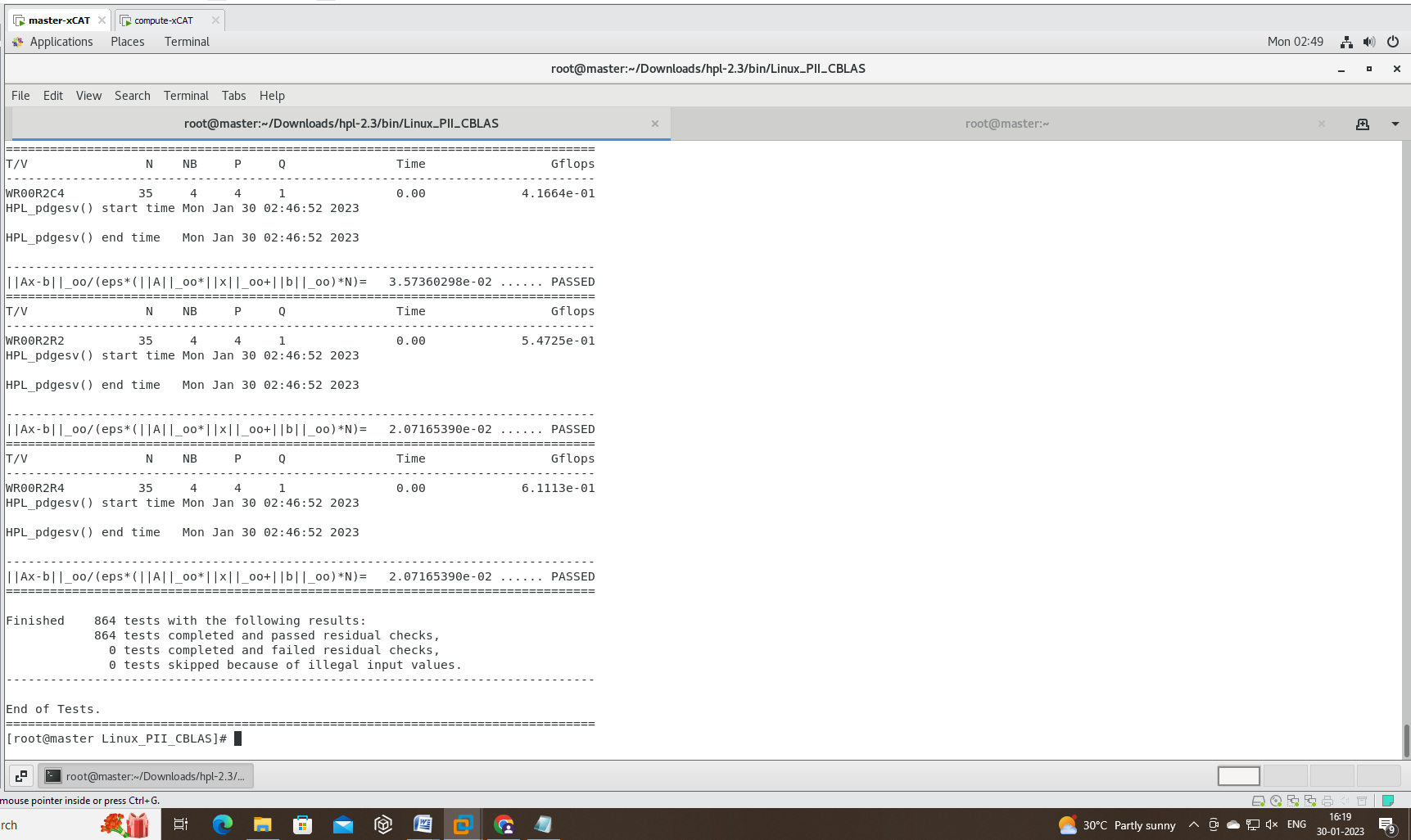


# mpirun --allow-run-as-root -np 4 ./xhpl HPL.dat





* Final Result of Benchmarking



**Conclusion-**

We have successfully booted our Compute node via network using XCAT. We further, configured LDAP for user authentication supported by creation of a new user. Furthermore, added Slurm in our cluster for performing job scheduling. Added, Ganglia for monitoring of both the nodes. And finally, checked the efficiency of our cluster through HPL Benchmarking.