Tutorial 1 (2019)

From ACE Lab

Follow the instructions below to complete this tutorial. The tutors are available to answer **valid, well thought out questions**. The tutors will not touch your keyboard or complete the tutorials for you. Answer questions provided in a text document and email to dmacleod@csir.co.za before Friday 5PM. Pay special attention to the instructions in red, you will need to demonstrate this to a tutor in order to complete that task.

The text editor 'vim' is referenced for editing files in these tutorials. You should use the text editor with which you are most comfortable. If you are new to Linux, nano is a simple editor that is quick to learn.

In these tutorials, an asterisks * or triangle brackets <> are placeholders. You need to fill in the correct relevant to you.

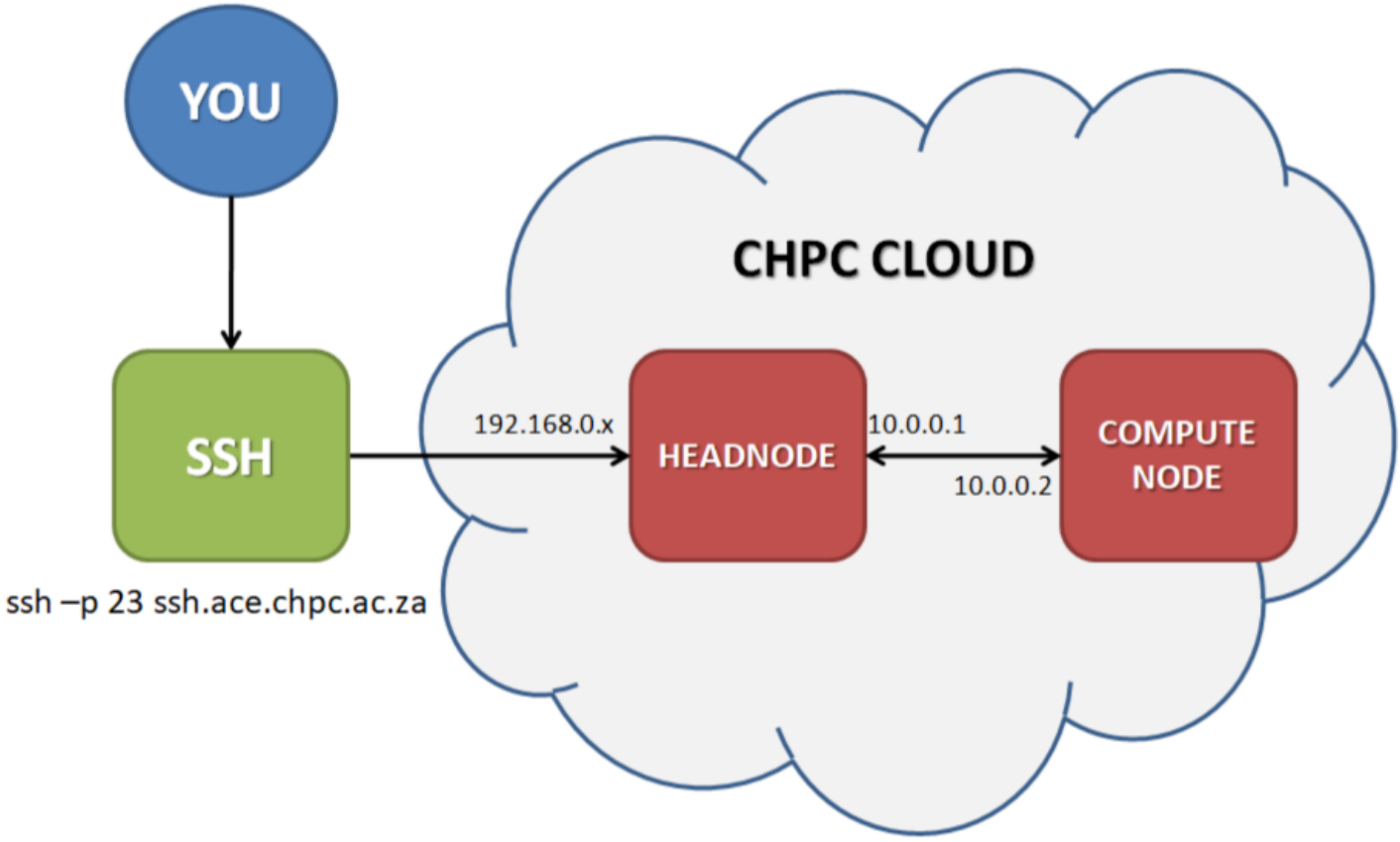
Part 1 - Network Setup

CHPC Private Cloud

In this tutorial, you will learn how to access the CHPC's private cloud and assess your virtual machine cluster. You will work remotely on these virtual machines to complete tutorials throughout the week. **DO NOT** try to follow the instructions below directly on your local lab workstation.

1. Open your web browser and visit **cloud.ace.chpc.ac.za**.
2. Send one member of your team to ask Israel for your login credentials, then login.
3. Under 'Instances', click 'VMs'. In the list of active virtual machines you will see 2 VMs representing your head node and compute node.
4. Note the IP addresses. The 192.168.0.* address is your **public address**. The 10.0.0.* address is your **private address** and can only be accessed through your head node (only your head node has assess to the 'outside world').

The diagram below describes how you connect to your VMs, as well as the internal and external network layout.



NOTE: You need to set up networking on the virtual machines using the **IP addresses from OpenNebula**. To do this, you must access the VM via the VNC utility integrated into OpenNebula (blue monitor icon).

Part 1A - IP adresses and Gateways

1. Use the command below to list your network adapter(s), from this you can see the names of the interfaces (eg: eth1, ens3):

```
ip a
```

2. Use a text editor to modify the networking scripts for the appropriate interfaces to the correct state (your head node should have 2 interfaces and your compute node should have 1).

```
vim (or nano) /etc/sysconfig/network-scripts/ifcfg-*  (an asterisks is a placeholder)
```

Create the network script file with the correct name in this directory if it doesn't exist.

The file should look like:

```
DEVICE=ens3
BOOTPROTO=none
ONBOOT=yes
NM_CONTROLLED=no
TYPE=Ethernet
NETMASK=255.255.255.0
IPADDR=*.*.*.*
GATEWAY=*.*.*.1
```

A gateway is a route out of your network, to an external network. Set an appropriate GATEWAY on each of your nodes (where they should look to, to access the Internet).

3. After editing the files, restart the networking on your VM:

```
systemctl restart network
```

4. Your VM should now have the correct IP address. To check the networking setup is correct execute the command:

```
ip route
```

Your **headnode** now has a functioning network interface and is now accessible within the ACE Lab Cloud. However, your public address (192.168.0.*) is not exposed to the Internet. In order for you to access your headnode, you will need to login via the CHPC **ssh server** (diagram above).
To do this, open Putty and SSH to **ssh.ace.chpc.ac.za** and login with your team's account. You are now logged into the 'ssh' server, located at the CHPC in Cape Town. From this machine you can SSH to your headnode via its external IP address, and from your headnode to your compute node, via its internal address (diagram above).
Remember that using the command ssh creates a new bash shell on the target machine, to end this session you must exit. Using the ssh command over and over will nest multiple bash shells and is not recommended.

Part 1B - IPTABLE, Firewall and NAT

NOTE: Only your ***headnode*** has an interface on the 192.168.0.* network and hence access to the Internet. To give your compute nodes access to the Internet you will have to setup a **NAT on headnode (allowing it to function as a router)**. This can be done using firewalld (the Linux firewall service).

```
systemctl start firewalld
iptables --flush
echo "net.ipv4.ip_forward = 1" >> /etc/sysctl.d/ip_forward.conf
echo 1 > /proc/sys/net/ipv4/ip_forward
firewall-cmd --permanent --direct --passthrough ipv4 -t nat -I POSTROUTING -o ens3 -j MASQUERADE -s 10.0.0.0/24
firewall-cmd --zone=public --add-port=80/tcp --permanent
firewall-cmd --reload
systemctl disable firewalld
```

To validate your NAT is working properly test that you can ping the Google **DNS** servers from your compute node.

```
ping 8.8.8.8
```

If you pass this test you should be able to now install packages using yum on your compute node.

```
yum install bind-utils
nslookup 8.8.8.8
```

Note: Without access to a working DNS server you won't be able to install packages.

Part 2 Hostnames

1. To make it easier to distinguish between your headnode and your compute node you should change their hostnames to something logical. Use the hostnamectl commands.

```
hostnamectl set-hostname --static <host_name>
```

2. You can test connectivity between your two nodes by pinging from one to the other, from your headnode:

```
ping 10.0.0.2
```

3. In order to access your nodes by hostname rather than IP address, you need to add the following line to /etc/hosts file for each of your nodes. Make sure this file exists on all of your machines

```
<10.0.0.*> <host_name> <host_name>.cluster.scc
```

4. Test that you can access your compute node by it's host name:

```
ping <host_name>
```

At this point your Virtual Machines and network should be correctly configured and you can continue with setting up some important Linux services below.

Part 3 - NTP

NTP enables you to synchronise the time across all the computers in your cluster. This is important for HPC clusters as some applications require that system time be accurate between different nodes (imagine receiving a message 'before' it was sent). You will now setup the NTP service on your head node and then allow your compute node to connect to it.

On your head node:
1. Install the NTP package using the CentOS package manager (yum)

```
yum install ntp
```

2. Edit /etc/ntp.conf, add:

```
vim (or nano ) /etc/ntp.conf
restrict 192.168.0.0 mask 255.255.255.0 nomodify notrap
```

3. NTP runs as a service (daemon) and needs to be started manually. Start the NTP daemon with:

```
systemctl start ntpd
```

then enable NTP to start automatically the next time CentOS boots:

```
systemctl enable ntpd
```

4. Use "ntpq -p" to monitor the sync state

On your compute node:
1. Edit /etc/ntp.conf, remove the other time servers and specify your head node as the only server

```
server <host_name> iburst
```

2. Start and enable the NTP service (step 3 from above).

3. Use "ntpq -p" to monitor the sync state

QUESTION 1:
What is a daemon?

_____

_____
_____

Part 4 - LDAP

LDAP allows you to centralise user account authentication on your cluster. With LDAP users to have a single account which is shared across all the nodes in the cluster. This is useful if you system has many users and many computers (eg, think about manually creating a user account on every Lab computer for each student at your University.) LDAP, like NTP, uses a client-server model. You will now setup the server on your head node and then setup both head node and compute node as clients.

Server Setup

1. Install dependencies

```
yum install openldap-servers openldap-clients
```

2. Setup LDAP server

```
cp /usr/share/openldap-servers/DB_CONFIG.example /var/lib/ldap/DB_CONFIG
chown ldap. /var/lib/ldap/DB_CONFIG
```

3. Start the LDAP service

```
systemctl start slapd
```

4. Import the provided configuration into LDAP:

```
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/chrootpw.ldif
```

5. Import basic Schemas:

```
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
```

6. Import LDAP database config

```
ldapmodify -Y EXTERNAL -H ldapi:/// -f /etc/openldap/chdomain.ldif
```

QUESTION 2:
What is a domain?

_____
_____
_____

7. Restart the LDAP service and enable auto start at boot time

9. Generate an SSL certificate

```
openssl req -new -x509 -nodes -out  /etc/pki/tls/certs/slapdcert.pem -keyout  /etc/pki/tls/certs/slapdkey.pem -days 365
```

QUESTION 3:
What is the purpose of an SSL certificate?

_____
_____
_____

9. Copy and change ownership of the certificates you generated

```
cp /etc/pki/tls/certs/slapdkey.pem /etc/pki/tls/certs/slapdcert.pem /etc/pki/tls/certs/ca-bundle.crt /etc/openldap/certs/
chown ldap. /etc/openldap/certs/slapdkey.pem /etc/openldap/certs/slapdcert.pem /etc/openldap/certs/ca-bundle.crt
```

10. Link SSL certificates

```
ldapmodify -Y EXTERNAL -H ldapi:/// -f /etc/openldap/mod_ssl.ldif
```

11. Edit /etc/sysconfig/slapd

```
# line 9: add
SLAPD_URLS="ldapi:/// ldap:/// ldaps:///"
```

12. Restart slapd service

LDAP Account Manager (LAM)

LDAP Account Manager (LAM) is a web frontend for managing entries (e.g. users, groups, DHCP settings) stored in an LDAP directory. LAM was designed to make LDAP management as easy as possible for the user.

1. Install LAM web frontend packages

```
yum install php php-ldap
```

2. Download LAM source code using the wget command from the link:

```
http://prdownloads.sourceforge.net/lam/ldap-account-manager-5.7-0.fedora.1.noarch.rpm
```

and install it:

```
rpm -i ldap-account-manager*
```

3. Restart and enable the httpd service

4. With your web-browser visit http://<your_headnode>/lam

5. Follow "LAM Configuration" link

6. Follow "Edit general settings" link, login with password "lam" and set master password

7. Follow "Edit server profiles" link and then follow the Manage server profiles. In this page set a new Profile password, then login

8. Enter relevant value in fields:

```
Sever address: ldap://localhost
Tree suffix: dc=cluster,dc=scc
List of valid users: cn=Manager,dc=cluster,dc=scc
```

Select the "Account types" tab and change the relevant sections of the LDAP suffix for Users and Groups to

```
dc=cluster,dc=scc
```

9. Login with LDAP root password

10. Click create to create base LDAP Configuration

11. Click "groups", "new group"

12. Create group "users"

13. Create group "admins"

14. Create an user account for yourself, set a password and add it to the admins group.

15. Create another 'guest' account, set a password and add it to the users group.

LDAP Client Setup

These steps need to executed on the head node and the compute nodes.

1. Install nss-pam-ldapd

2. Enable TLS support for the client

```
echo "TLS_REQCERT allow" >> /etc/openldap/ldap.conf
echo "tls_reqcert allow" >> /etc/nslcd.conf
```

3. Run through the LDAT client setup wizard

```
authconfig-tui
```

```
[*] Use LDAP
[*] Use Shadow Passwords
[*] Use LDAP Authetication
[*] Local authorization is sufficient
```

```
[*] Use TLS
Server: ldap://<headnode>.cluster.scc
Base DN: dc=cluster,dc=scc
```

4. By default user home directories are not created automatically, enable it

```
authconfig --enablemkhomedir --update
```

5. Test the LDAP Server

```
slaptest -u
ldapsearch -x -b "dc=<localdomain>,dc=<com>"
```

it should return “search: 2”

6. Test with TLS encryption:

```
ldapsearch -x -b "dc=<localdomain>,dc=<com> -ZZ"
```

it should return “search: 3”

7. To allow sudo access for your admin group, add it sudoers file on each node. Edit /etc/sudoers file, add:

```
%<admin_group>    ALL=(ALL)       ALL
```

Test the accounts:

```
su <admin_user>
sudo su
su <guest_user>
sudo su
```

Retrieved from ‘https://www.ace.chpc.ac.za/acewiki/index.php?title=Tutorial_1_(2019)&oldid=1644’
