1. Virtual Machines Setup

April 20, 2021

0.1 # Virtual Machines Setup

This notebook contains the preliminary steps necessary to set up and configure correctly the network and the software of our virtual machines. In particular:

0. We will retrieve the IP addresses of our virtual machines and assign them new hostnames we will use later.
1. We will create a new user called `hadoop`; we will install and use Hadoop as this user.
2. We will configure a password-less SSH access among your virtual machines.
3. We will configure your laptop to access the virtual machines with password-less SSH.
4. We will install the required software on each virtual machines.
5. We will configure the network of your virtual machines.

Each step assumes that you start with a open Bash shell on your local machine.

0.2 ## 0. Preliminaries

Retrieve and write down a list of the IP address of your virtual machines before moving on. For example, if you have 5 virtual machines, populate the following table with the actual IP addresses (leave the Hostname column empty, for now).

VM	IP address	Hostname
1	172.16.0.17	
2	172.16.0.3	
3	172.16.0.167	
4	172.16.0.49	
5	172.16.0.221	

Next, assign a unique hostname to each virtual machine. In our class, we will use the following convention. 1. A single VM, for exampe the VM 5, will get the hostname `namenode`. 2. All remaining VMs, i.e. VM 1, VM 2, VM 3, and VM 4, will get the hostnames `datanode1`, `datanode2`, `datanode3`, `datanode4`. For example, our 5 virtual machines will have the following hostnames:

VM	IP address	Hostname
1	172.16.0.17	`datanode1`
2	172.16.0.3	`datanode2`
3	172.16.0.167	`datanode3`
4	172.16.0.49	`datanode4`

VM	IP address	Hostname
5	172.16.0.221	namenode

Please double check that the table is correctly setup, as an error at this stage will compromise all future configuration activities.

0.3 ## 1. Hadoop user creation

On each virtual machine in your cluster, execute the following steps, one by one, line by line. *Do not copy-paste more than one line at a time.*

Note: When you replace the `<ip address>` field, **always use IP addresses, never use hostnames**.

1. Login as `root` providing the `root` user password when requested.

```
ssh root@<ip address>
```

2. Create the `hadoop` user account. Provide a new password for the new `hadoop` user when requested.

```
sudo adduser --gecos '' hadoop
sudo adduser hadoop sudo
```

3. Logout.

```
exit
```

Repeat the previous step for every virtual machines in your cluster.

0.4 ## 2. Configure SSH password-less access

On every virtual machine in your cluster, execute the following steps, one by one, line by line. *Do not copy-paste more than one line at a time.*

Note: When you replace the `<ip address>` field, **always use IP addresses, never use hostnames.**

1. Login as `hadoop` providing the `hadoop` user password when requested.

```
ssh hadoop@<ip address>
```

2. Create the `.ssh` folder.

```
mkdir ~/.ssh
chmod 700 ~/.ssh
```

3. Generate a SSH key-value pair.

```
ssh-keygen -t rsa -N '' -f ~/.ssh/id_rsa
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
chmod 700 ~/.ssh/authorized_keys
```

4. Configure the SSH access.

```
echo "StrictHostKeyChecking no" > ~/.ssh/config
chmod 644 ~/.ssh/config
```

5. Check that you can ssh **locally, i.e., from the virtual machine to the virtual machine,** without a passphrase.

```
ssh localhost
exit
```

6. Logout.

```
exit
```

Repeat the previous step for every virtual machines in your cluster.

0.5 ## 3. Configure password-less access from your laptop

On your local machine, execute the following steps, one by one, line by line. *Do not copy-paste more than one line at a time.*

Note: When you replace the `<ip address>` field, **always use IP addresses, never use hostnames.**

1. Generate a key-value pair using SSH. *You can skip this phase if you already have a SSH key-value pair on your local machine.*

   ```
   ssh-keygen -t rsa -N '' -f ~/.ssh/id_rsa
   ```

2. Run the following command **for each virtual machine** providing the `hadoop` user password when requested.

   ```
   ssh-copy-id -i ~/.ssh/id_rsa.pub hadoop@<ip address>
   ```

3. Check that you can ssh **remotely** without a passphrase **on each virtual machine**.

   ```
   ssh hadoop@<ip address>
   exit
   ```

   If asked for a password, you did not complete steps 2 and/or 3 successfully. Go back and repeat.

0.6 ## 4. Install required software

On each virtual machine in your cluster, execute the following steps, one by one, line by line. *Do not copy-paste more than one line at a time.*

Note: When you replace the `<ip address>` field, **always use IP addresses, never use hostnames.**

1. Login as `hadoop`.

   ```
   ssh hadoop@<ip address>
   ```

   If asked for a password, you did not complete steps 2 and/or 3 successfully. Go back and repeat.

2. Run the following commands, providing the `root` user password if requested. Some commands can take a lot of time.

   ```
   sudo apt update
   sudo apt upgrade -y
   sudo apt install -y nano python3 python3-pip ipython3 openjdk-8-jdk
   sudo apt autoremove -y --purge
   ```

3. Run the following commands, providing the `root` user password if requested, to create the folder we will use:

   ```
   sudo mkdir -p /opt/{hadoop/logs,hdfs/{datanode,namenode},yarn/logs,spark/logs}
   sudo chown -R hadoop /opt
   ```

   We will install all our software under the `/opt` directory and store HDFS underlying data there as well. The layout of the `/opt` directory will look like:

   ```
   /opt
       hadoop
           logs
       hdfs
           datanode
           namenode
       yarn
           logs
       spark
            logs
   ```

4. Logout.

   ```
   exit
   ```

0.7 ## 5. Configure the network

On each virtual machine in your cluster, execute the following steps, one by one, line by line. *Do not copy-paste more than one line at a time.*

Note: When you replace the `<ip address>` field, **always use IP addresses, never use hostnames.**

1. Login as `hadoop`.

   ```
   ssh hadoop@<ip address>
   ```

   If asked for a password, you did not complete steps 2 and/or 3 successfully. Go back and repeat.

2. Edit the `/etc/hosts` file with the following command, providing the `root` user password if requested:

   ```
   sudo nano /etc/hosts
   ```

   Delete all its contents and replace them with the following, modified according to your virtual machines and hostnames selected at step 0:

```
127.0.0.1    localhost
172.16.0.17  datanode1
172.16.0.3   datanode2
172.16.0.167 datanode3
172.16.0.49  datanode4
172.16.0.221 namenode
```

3. Edit the `/etc/hostname` file with the following command, providing the `root` user password if requested:

   ```
   sudo nano /etc/hostname
   ```

   Replace its content with the hostname corresponding to the IP address of the virtual machine selected at step 0. For example, on the machine with IP address 172.16.0.17, the `/etc/hostname` file should contain the following **single line**: `datanode1`

4. **IMPORTANT**: you must make sure that the **name node has a password-less access to the data nodes**. Hence, on the **namenode** machine only, run the following commands for every **datanode** machine:

   ```
   ssh-copy-id -i /home/hadoop/.ssh/id_rsa.pub hadoop@datanode1
   ssh-copy-id -i /home/hadoop/.ssh/id_rsa.pub hadoop@datanode2
   ssh-copy-id -i /home/hadoop/.ssh/id_rsa.pub hadoop@datanode3
   ssh-copy-id -i /home/hadoop/.ssh/id_rsa.pub hadoop@datanode4
   ```

5. Reboot the machine with the following command, providing the `root` user password if requested:

   ```
   sudo reboot
   ```

From now on, we will never use again the root privileges, i.e. `sudo` commands.