EXERCISE-8 install hadoop single node cluster

All should be in C Drive

https://muhammadbilalyar.github.io/blogs/How-to-install-Hadoop-on-Window-10/

https://github.com/MuhammadBilalYar/HADOOP-INSTALLATION-ON-WINDOW-10/blob/master/Hadoop%20Configuration.zip

My Computer->properties->advanced sys settings->env.var->Path->
Hadoop
->create data folder->datanode,name node[fol]
->etc ->hadoop
-->core.site.xml(open in notepad)

```
        <property>
<name>fs.defaultFS</name>
<value>hdfs://localhost:9000</value>
 </property>
```

-->mapred-site.xml

```
  <property>
<name>mapreduce.framework.name</name>
<value>yarn</value>
 </property>
```

-->hdfs.site.xml

```
<configuration>
<property>
```

<name>dfa.replication</name> <value>1</value>

</property>

<property>

<name>dfs.namenode.name.dir</name>

<value>/C:\hadoop-2.8.0\hadoop-2.8.0\data\namenode</value>

</property>

<property>

<name>dfs.datanode.data.dir</name>

<value>/C:\hadoop-2.8.0\hadoop-2.8.0\data\datanode</value>

</property>
</configuration>
->yarn-site.xml
<property>

<name>yarn.nodemanager.aux-services</name> <value>mapreduce_shuffle</value>

</property>

<property>

<name>

yarn.nodemanager.auxservices.mapreduce.shuffle.class</name>
<value>org.apache.hadoop.mapred. ShuffleHandler</value>

</property>
</configuration>

→hadoop-env.cmd(ri8t clk->edit)
Set jdk path at
→The java implementation
Set JAVA_HOME→(PATH)
—----------
Hadoop COnfigurtaion

Bin:
Copy all files

Paste at -> Hadoop 2.8.0→ bin->(paste or replace)
—----------
Open cmd

1. hdfs namenode -format
2.cd\
3.cd C:\hadoop-2.8.0\sbin
4.start-all.cmd
->four windows will open
—-------
Open chrome

Localhost 8080 or 8088

Hapdoop interface will display

—----------

Open Netbeans 7.0.1

1.New file->java->java application
2.Change pro name and class name
2.1 write java code for wordcount
3. Right click project name→properties->libraries-> add jar file(c:hadoop\share or mapreduce).jar
4. run-> clean and built main project
5.o/p→ Building jar: (copy path)
→lib→documents→netbeansProjects→bulid->dist->pro name.jar
6.copy that jar file in C DRIVE (eg.mapreduce.jar)

—-------

Create notepad file
With words
In C drive

- —----
- Open cmd
- cd/
- cd/ hadoop path\bin
- hdfs namenode -format
- Cd ..
- Cd sbin
- Start-all.cmd
-
- jps

—----------

Open chrome

Localhost 8080 or 8088
Local host 50070

—----------

Open cmd
1.cd/
2.hadoop dfsadmin -safenode leave
3.hadoop fs -mkdir /input_dir
4.hadoop fs -put C:/wordcout.txt /input_dir
5.hadoop dfs -cat /input_dir
6.hadoop fs -ls /input_dir
7.hadoop dfs -cat /input_dir/wordcount.txt
8.hadoop jar C:/MapReduceCLient.jar wordcount /input_dir /output_dir
9.hadoop dfs -cat /Output_dir/*

EXERCISE-7

Installation of open stack

1.open vm ware
2.create virtual machine
3.start virtual machin
4.open terminal

Cmd:

```
systemctl disable firewalld
systemctl stop firewalld

systemctl disable NetworkManager
systemctl stop NetworkManager

systemctl enable network
systemctl start network

yum install -y centos-release-openstack-newton
yum update -y

Yum install -y openstack-packstack

packstack –allinone
```

If any error occr refer:

```
packstack - - answer - file
```

//open keyston admin

```
ls
cat keystonrc_admin
      →save your username & password
       →save the ip adress
       →open browser
       -> type–?eg 10.0.2.15/dashboard
      ->login
```

Open VM in openstack

Go to Networks
->create network
—---------->subnet:-->network address:192.168.37 0/24
—dns name→8.8.8.8
<create>

Go to compute->go to instances→launch instance
→name:
→ flavour: m1 small
->instancece boot: boot from imfage
->image name:cenos7

Go to access
->add key pair
Go to cmd →cat id_rsa.pub (copy and paste the key)

→paste in public key

Press Launch(instance created)

Go to network→router→create router
set GAteway:
External:external→set gateway

Click router name (you created)
 →add interface ---->subnet:(click downlink)

Go to access &security->sec.grps->
->MAnage rules->add rule
->rule:custom ipmp rule
->direction:ingress
->type:-1
->code:-1

>add rule
  >port:22

//open cmd

ping 8.21.28.113
ssh centosh@8.21.28.113
—----------

Ex5-Simulate a cloud scenario using CloudSim

1. Install java:https://www.oracle.com/java/technologies/javase/jdk15-archive-downloads.html
2. Install eclipse(2020-03):https://www.eclipse.org/downloads/packages/release/2020-03/r
3. Install cloud sim:https://github.com/Cloudslab/cloudsim/releases
4. https://github.com/Cloudslab/cloudsim/releases/tag/cloudsim-3.0.2

//Open Eclipse
->create new project then
Go to ->src->new->package
->click package->show in->system exp
->copy all java sdfs files in package folder
—-->constant
—-->datacenter
—-->Generatematrics
—-->sjf_scheuler
—-->sjfddatacenter

->click project naeme->bulid path->configure path->class path->add jar file

EXCERCISE-3 INSTALL GOOGLE APP ENGINE

1.Install python 3.9or 3.7
2.Google sdkhttps://google-app-engine.en.uptodown.com/windows
https://google-app-engine.en.softonic.com/

//open python
Create python “hello world”

Create App.yaml

runtime: python27
api_version: 1
threadsafe: false

handlers:

- url: / or - url: /.*

script: index.py

(or)
application:ae-01-trivival
version: 1
runtime: python
api_version: 1

handlers:

- url: / or - url: /.*

script: index.py

_
Reference: http://en.wikipedia.org/wiki/Stack_trace

When you make a mistake in the app.yaml file – you must the fix the mistake and attempt to start the application again.

Make a folder for your Google App Engine applications. I am going to make the Folder on my Desktop called “apps” – the path to this folder is: C:\Documents and Settings\csev\Desktop\apps And then make a sub--folder in within apps called “ae--01--trivial” – the path to this folder would be: C:\ Documents and Settings \csev\Desktop\apps\ae--01--trivial Using a text editor such as JEdit (www.jedit.org), c reate a file called app.yaml in the ae--01--trivial folder with the following contents:

```
application: ae-01-trivial
version: 1
 runtime: python
api_version: 1
handlers: - url: /.* script: index.py
```

Note: Please do not copy and paste these lines into your text editor – you might end up with strange characters – simply type them into your editor.
Then create a file in the ae--01--trivial folder called index.py with three lines in it:

```
print 'Content-Type: text/plain'
 print ' '
 print 'Hello there Chuck'
```

Then start the GoogleAppEngineLauncher program that can be found under Applications. Use the File --> Add Existing Application command and navigate into the apps directory and select the ae--01--trivial folder. Once you have added the application, select it so that you can control the application using the launcher.

//open local host or

//open cloud sdk shell
Cmd:
google-cloud-sdk\bin(use tab key)\dev-appserver.py “path of python file”

Note down the localhost and id

Exercise 6

Send file from one to another vm

How to attach windows folder with Ubuntu

Steps:

1. Click Devices - Shared Folders - Shared Folders Setting

2. Click on the Add New share folder button

3. Select your required windows folder. Check auto-mount and make permanent button.

4. Click Devices- Insert Guest Additions CD Images... and install guest addition.

5. Create new folder on Ubuntu where you want to attach the above windows folder.

6. Run the following command from terminal to attach the folder

sudo mount -t vboxsf <Windows folder name> <Ubuntu folder name>

Eg
Sudo mount -t vboxsf openstack mynewshare

—————
stack@stack-VirtualBox:-$
cd /media

stack@stack-VirtualBox: /nedta$ ls
sf_Linux_tnage sf share stack

stack@stack-VirtualBox: /
media$ mkdir mynewshare

mkdir: cannot create directory 'nynewshare': Permission denied

stack@stack-VirtualBox: /nedta$

sudo mkdir mynewshare
stack@stack-VirtualBox: /media$ ls

nynewshare sf_Linux_inage sf share stack

stack@stack-VirtualBox: /nedta$

sudo mount -t vboxsf Openstack mynewshare stack@stack-VirtualBox: /nedta$

Exercise 1 download Virtual box and ubuntu

http://www.virtualbox.org/wiki/downloads

http://www.ubuntu.com/download/ubuntu/download

Exercise 2 install C compiler
Open cmd in ubuntu

->sudo apt install gcc
->gcc - -version
//create c program
Open Terminal

->gcc helloworld.c
->./a.out

3.Using Apache Axis develop a Grid Service

OBJECTIVE:
To develop a Grid Service using Apache Axis.
PROCEDURE:
You will need to download and install the following software:
1. Java 2 SDK v1.4.1,

http://java.sun.com/j2se/1.4.1/download.html

2. Apache Tomcat v4.124

http://jakarta.apache.org/builds/jakarta-tomcat-4.0/release/v4.1.24/bin/jakarta tomcat4.1.24.exe.

3. XML Security v1.0.4,

http://www.apache.org/dist/xml/security/java-library/xmlsecurity bin1.0.4.zip

4. Axis v1.1,

http://ws.apache.org/axis/dist/1_1/axis-1_1.zip

1. Java 2 SDK
• Run the downloaded executable (j2sdk-1_4_1-windows-i586.exe) which will install the
• SDK in C:\j2sdk1.4.1.
Set the JAVA_HOME environment variable to point to this
directory as follows:
• Click on START->CONTROL PANEL->SYSTEM
• Click on the Advanced tab
• Click on the Environment Variables button
• Click on the New… button in the user variable section and enter the details
• Add the Java binaries to your PATH variable in the same way by setting a user variable called PATH with the value “%PATH%;C:\j2sdk1.4.1\bin”
2. Apache Tomcat

• Run the downloaded executable (jakarta-tomcat-4.1.24.exe), and assume the installation directory is C:\jakarta-tomcat-4.1.24.

• Edit C:\ jakarta-tomcat-4.1.24\conf\tomcat-users.xml and create an “admin” and “manager” role as well as a user with both roles. The contents of the file should be similar to:

```
<?xml version='1.0' encoding='utf8'?>
<tomcat-users>
<role rolename="manager"/>
<role rolename="admin"/>
<user username="myuser" password="mypass"
roles="admin,manager"/>
</tomcat-users>
```

• Start Tomcat by running C:\ jakarta-tomcat-4.1.24\bin\startup.bat and test it by browsing
http://localhost:8080/

• Stop Tomcat by running C:\ jakarta-tomcat-4.1.24\bin\shutdown.bat.

3. XML Security

• Download and unzip

http://www.apache.org/dist/xml/security/javalibrary/xmlsecurity-bin 1_0_4.zip

• Copy xml-sec.jar to C:\axis-1_1\lib\

• Set-up your CLASSPATH environment variable to including the following:
C:\axis1_1\lib\xml-sec.jar;

4. Apache Axis

• Unzip the downloaded Axis archive to C: (this will create a directory C:\axis-1_1).
• Extract the file xmlsec.jar from the downloaded security archive to C:\axis1_1\webapps\axis\WEB-INF\lib.
• Set-up your CLASSPATH environment variable to including the following:
o The current working directory
o All the AXIS jar files as found in C:\axis-1_1\lib
C:\jakarta-tomcat-4.1.24\common\lib\servlet.jar

• Your CLASSPATH should therefore look something like:

```
C:\axis-1_1\lib\axis.jar;
C:\axis 1_1\lib\axis-ant.jar;
C:\axis-1_1\lib\commons-discovery.jar;
C:\axis-1_1\lib\commons-logging.jar;
C:\axis-1_1\lib\jaxrpc.jar;
C:\axis-1_1\lib\log4j-1.2.8.jar;
C:\axis-1_1\lib\saaj.jar;
C:\axis-1_1\lib\wsdl4j.jar;
C:\axis-1_1\lib\xercesImpl.jar
C:\axis-1_1\lib\xmlParserAPIs.jar;
C:\jakarta-tomcat-4.1.24\common\lib\servlet.jar
C:\axis-1_1\lib\xml-sec.jar;
```

• Now tell Tomcat about your Axis web application by creating the file C:\jakarta-tomcat-4.1.24\webapps\axis.xml with the following content:

<Context path="/axis" docBase="C:\axis-1_1\webapps\axis" debug="0"
privileged="true">
<LoggerclassName="org.apache.catalina.logger.FileLogger"prefix="axis_log."
suffix=".txt" timestamp="false"/>

5. Deploy a Sample Web service packaged within Axis installations
Deploy one of the sample Web Services to test the system and to create the C:\axis-1_1\webapps\axis\WEB-INF\server-config.wsdd file. From C:\axis-1_1 issue the command (on one line):
java org.apache.axis.client.AdminClient
http://localhost:8080/axis/services/AdminService/samples/stock/deploy.wsdd
This should return the following:
.- Processing file samples/stock/deploy.wsdd.- <Admin>Done processing</Admin>