Apache Hadoop – Hands-On

Objetivo: O objetivo deste hands-on é criar uma máquina virtual Linux e configurar o Apache Hadoop no modo pseudo-distribuído.

Passos

1. Crie uma máquina virtual Ubuntu 16.04 – 64 bits (Xenial) usando VirtualBox e Vagrant (https://www.vagrantup.com/)

```
$ mkdir ~/HadoopVM
$ cd ~/HadoopVM
$ vagrant init ubuntu/xenial64
```

2. Editar o arquivo Vagrantfile. Incluir as linhas abaixo antes do "end" final.

```
  # Forward ports
  # HDFS Namenode Web UI
  config.vm.network "forwarded_port", guest: 50070, host: 50070, host_ip: "127.0.0.1"
  # YARN Resource Manager Web UI
  config.vm.network "forwarded_port", guest: 8088, host: 8088, host_ip: "127.0.0.1"

  config.vm.provider "virtualbox" do |vb|
    # 2 cores
    vb.cpus = 2
    # 4GB of RAM
    vb.memory = 4096
  end
```

3. Iniciar a máquina virtual, esperar completar o provisionamento e fazer login na VM.

```
$ vagrant up
$ vagrant ssh
```

4. Substituir o conteúdo do arquivo sources.list para utilizar mirrors. (Obs. Utilizar sudo)

```
deb mirror://mirrors.ubuntu.com/mirrors.txt xenial main restricted universe multiverse
deb mirror://mirrors.ubuntu.com/mirrors.txt xenial-updates main restricted universe multiverse
deb mirror://mirrors.ubuntu.com/mirrors.txt xenial-backports main restricted universe multiverse
deb mirror://mirrors.ubuntu.com/mirrors.txt xenial-security main restricted universe multiverse
```

5. Atualizar a lista de pacotes

```
$ sudo apt update
```

6. Instalar pacotes

```
$ sudo apt install ssh rsync wget openjdk-8-jdk
```

7. Configurar a variável de ambiente JAVA_HOME

```
$ export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
```

8. Fazer download do Hadoop (versão 2.9.1) no site (Tarball, binary).

http://hadoop.apache.org/

9. Descompactar e copiar para diretório /usr/share

```
$ tar -zxvf hadoop-2.9.1.tar.gz
$ sudo cp -r hadoop-2.9.1 /usr/share/hadoop
$ export HADOOP_HOME=/usr/share/hadoop
```

10. Desabilitar IPv6

```
$ sudo sed -i "\$anet.ipv6.conf.all.disable_ipv6 = 1" /etc/sysctl.conf
$ sudo sed -i "\$anet.ipv6.conf.default.disable_ipv6 = 1" /etc/sysctl.conf
$ sudo sed -i "\$anet.ipv6.conf.lo.disable_ipv6 = 1" /etc/sysctl.conf
$ sudo sysctl -p
```

11. Organizar diretório com arquivos de configuração do hadoop

```
$ sudo mkdir -p /etc/hadoop/conf
$ sudo cp $HADOOP_HOME/etc/hadoop/mapred-site.xml.template
$HADOOP_HOME/etc/hadoop/mapred-site.xml
$ sudo sed -i "\$aexport JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64"
$HADOOP_HOME/etc/hadoop/hadoop-env.sh
$ sudo ln -s $HADOOP_HOME/etc/hadoop/* /etc/hadoop/conf/
```

12. Criar diretório de logs

```
$ sudo mkdir $HADOOP_HOME/logs
```

13. Criar usuários e grupos para HDFS e YARN

```
$ sudo groupadd hadoop
$ sudo useradd -g hadoop hdfs
$ sudo useradd -g hadoop yarn
```

14. Setar permissões

```
$ sudo chgrp -R hadoop /usr/share/hadoop
$ sudo chmod -R 777 /usr/share/hadoop
```

15. Editar arquivo /etc/hadoop/conf/core-site.xml. Adicionar as linhas abaixo entre as tags <configuration> e </configuration>. (Obs. Utilizar sudo)

```
<property>
<name>fs.defaultFS</name>
<value>hdfs://0.0.0.0:9000</value>
</property>
```

16. Editar arquivo /etc/hadoop/conf/hdfs-site.xml. Adicionar as linhas abaixo entre as tags <configuration> e </configuration>. (Obs. Utilizar sudo)

```
<property>
<name>dfs.replication</name>
<value>1</value>
```

```
</property>
```

17. Editar arquivo /etc/hadoop/conf/yarn-site.xml. Adicionar as linhas abaixo entre as tags <configuration> e </configuration>. (Obs. Utilizar sudo)

```
<property>
<name>yarn.resourcemanager.hostname</name>
<value>0.0.0.0</value>
</property>
<property>
<name>yarn.nodemanager.aux-services</name>
<value>mapreduce_shuffle</value>
</property>
```

18. Editar arquivo /etc/hadoop/conf/mapred-site.xml. Adicionar as linhas abaixo entre as tags <configuration> e </configuration>. (Obs. Utilizar sudo)

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

19. Formatar HDFS NameNode

```
$ sudo -u hdfs $HADOOP_HOME/bin/hdfs namenode -format
```

20. Iniciar os daemons NameNode e DataNode (HDFS)

```
$ sudo -u hdfs $HADOOP_HOME/sbin/hadoop-daemon.sh start namenode
$ sudo -u hdfs $HADOOP_HOME/sbin/hadoop-daemon.sh start datanode
```

21. Iniciar os daemons ResourceManager e NodeManager (YARN)

```
$ sudo -u yarn $HADOOP_HOME/sbin/yarn-daemon.sh start resourcemanager
$ sudo -u yarn $HADOOP_HOME/sbin/yarn-daemon.sh start nodemanager
```

22. Verificar se todos os daemons estão executando (ResourceManager, NodeManager, DataNode, Jps, NameNode)

```
$ sudo jps
```

23. Configurar os diretórios HDFS

```
$ sudo -u hdfs $HADOOP_HOME/bin/hadoop fs -mkdir -p /user/$USER
$ sudo -u hdfs $HADOOP_HOME/bin/hadoop fs -chown $USER:$USER /user/$USER
$ sudo -u hdfs $HADOOP_HOME/bin/hadoop fs -mkdir /tmp
$ sudo -u hdfs $HADOOP_HOME/bin/hadoop fs -chmod 777 /tmp
```

24. Acessar as interfaces do NameNode e ResourceManager usando navegador da máquina local

```
NameNode – http://localhost:50070
ResourceManager – http://localhost:8088
```

25. Executar a aplicação Pi usando MapReduce

```
$HADOOP_HOME/bin/hadoop jar $HADOOP_HOME/share/hadoop/mapreduce/hadoop-mapreduce-
examples-2.9.1.jar pi 16 1000
```