Command Terminal 指令快捷: ctrl+Alt+T

1. 安裝ubuntu (ubuntu 14~16 或其他linux系統) 、 CentOS
2. 建立hadoop user:

* 新建hadoop用户:

sudo useradd -m hadoop -s /bin/bash

* 帳號 hadoop 密碼建置:

sudo passwd hadoop (pw設為hadoop方便操作)

* 為hadoop帳號增加權限:

sudo adduser hadoop sudo (在此設為管理者權限)

* + 另一方法：打開sudoers文件，給hadoop用戶賦予和root一樣的權限：

執行sudo gedit /etc/sudoers，在裡面參照root添加hadoop ALL(ALL:ALL) ALL

1. 安装ssh

* 安裝SSH server:

sudo apt-get install openssh-server

* + Ubuntu 默認安裝SSH client，需另外安裝SSH server
* 啟動ssh service:

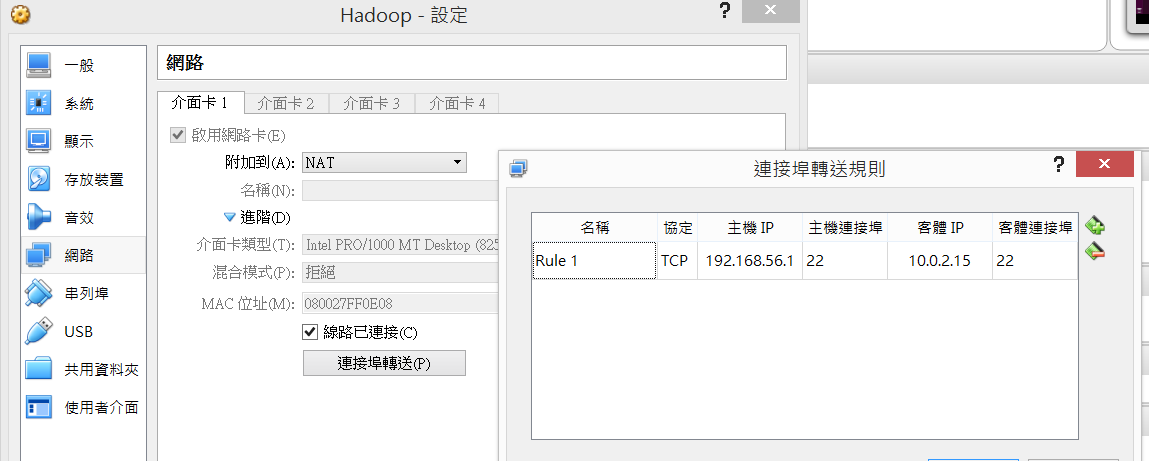
/etc/init.d/ssh start (安装好ssh服务后就打开它（尚未自動啟動）

* ssh service 起動檢查:
* 設置免密碼登錄:

ssh-keygen -t rsa

* + 在/home/hadoop/.ssh產生id\_rsa、id\_rsa.pub(私鑰, 公鑰)
  + 公鑰放置: cat ~/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys (將公鑰添加在authorized\_keys)
* authorized\_keys 保存所有以當前用戶身份登錄到ssh客戶端用戶的公鑰內容，"~" 代表用戶的主文件夾
* chmod 600 authorized\_keys (CentOS 會有權限的問題)
* ssh hadoop@localhost (應該就不會要求輸入密碼)
* 登錄ssh:

ssh hadoop@localhost

* + 其他Terminal(ssh hadoop@（IP|HOSTNAME)
  + 如要用其他Terminal測試(Xshell、putty)，做下設定。
* 退出ssh:

指令:exit

1. 安裝JAVA:

* CentOS 注意先移除 OpenJDK
* rpm -qa | grep java (List OpenJDK出來並卸載)
* rpm -e --nodeps java-1.7.0-openjdk-1.7.0.99-2.6.5.1.el6.x86\_64
* 下載JDK 二進制編譯包
  + 複製JDK壓縮案:

複製檔案至/usr/local/: sudo cp jdk-7u79-linux-x64.tar.gz /usr/local/

切換目錄: cd /usr/local/

* + 解壓縮JDK:

sudo tar xzvf jdk-7u79-linux-x64.tar.gz

* + 安裝vim:

sudo apt-get install vim

* + 配置環境:

sudo vim /etc/profile

* + 添加以下內容:

export JAVA\_HOME=/usr/local/jdk1.7.0\_79

export JRE\_HOME=$JAVA\_HOME/jre

export CLASSPATH=.:$JAVA\_HOME/lib

export PATH=$PATH:$JAVA\_HOME/bin

備註: source /etc/profile 使設置生效或請登出重開，設定才會生效

1. 安裝hadoop 2.7.3:

* rsync套件: apt-get install rsync
* Apache Hadoop下載網址: <http://hadoop.apache.org/releases.html>
* 官方文件: <http://hadoop.apache.org/docs/r2.7.3/hadoop-project-dist/hadoop-common/SingleCluster.html#YARN_on_Single_Node>
  + 解壓縮至 tar -zxvf hadoop-2.7.3.tar.gz /usr/local
  + cd /usr/local/
  + sudo mv ./hadoop-2.7.3/ ./hadoop #將資料夾改名為hadoop
  + sudo chown -R hadoop ./hadoop #修改文件權限
  + 修改配置core-site.xml，如下:

vim /usr/local/hadoop/etc/hadoop/core-site.xml:

<configuration>

<property>

<name>hadoop.tmp.dir</name>

<value>file:/usr/local/hadoop/tmp</value>

<description>Abase for other temporary directories.</description>

</property>

<property>

<name>fs.defaultFS</name>

<value>hdfs://localhost:9000</value>

</property>

</configuration>

* + 修改配置文件 hdfs-site.xml，如下:

vim /usr/local/hadoop/etc/hadoop/core-site.xml:

<configuration>

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

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

<value>file:/usr/local/hadoop/tmp/dfs/name</value>

</property>

<property>

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

<value>file:/usr/local/hadoop/tmp/dfs/data</value>

</property>

</configuration>

* + /usr/local/hadoop/bin/hdfs namenode -format # namenode 格式化
  + /usr/local/hadoop/sbin/start-dfs.sh # 開啟
  + jps # 判斷是否啟動成功

若成功會出現NameNode、DataNode和SecondaryNameNode

* 設環境變數:export PATH=$PATH:/usr/local/hadoop/sbin:/usr/local/hadoop/bin
* 運行WordCount 範例(建立hadoop文件系統中):
* /usr/local/hadoop/bin/hdfs dfs -mkdir -p /user/hadoop # 創建HDFS目錄
* /usr/local/hadoop/bin/hdfs dfs -mkdir input
* /usr/local/hadoop/bin/hdfs dfs -put etc/hadoop/\*.xml input # 將配置文件作為輸入
* /usr/local/hadoop/bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-\*.jar grep input output 'dfs[a-z.]+'
* /usr/local/hadoop/bin/hdfs dfs -cat output/\* # 查看輸出
  + Web 介面 :<http://localhost:50070>

1. 啟動Yarn(hadoop環境中):

mv ./etc/hadoop/mapred-site.xml.template ./etc/hadoop/mapred-site.xml # 改檔名

* + vim mapred-site.xml:

<configuration>

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

</property>

</configuration>

* + vim yarn-site.xml：

<configuration>

<property>

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

<value>mapreduce\_shuffle</value>

</property>

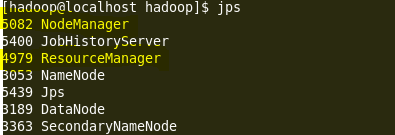
</configuration>

* + 啟動Yarn(./sbin/start-dfs.sh # 這要先啟動):

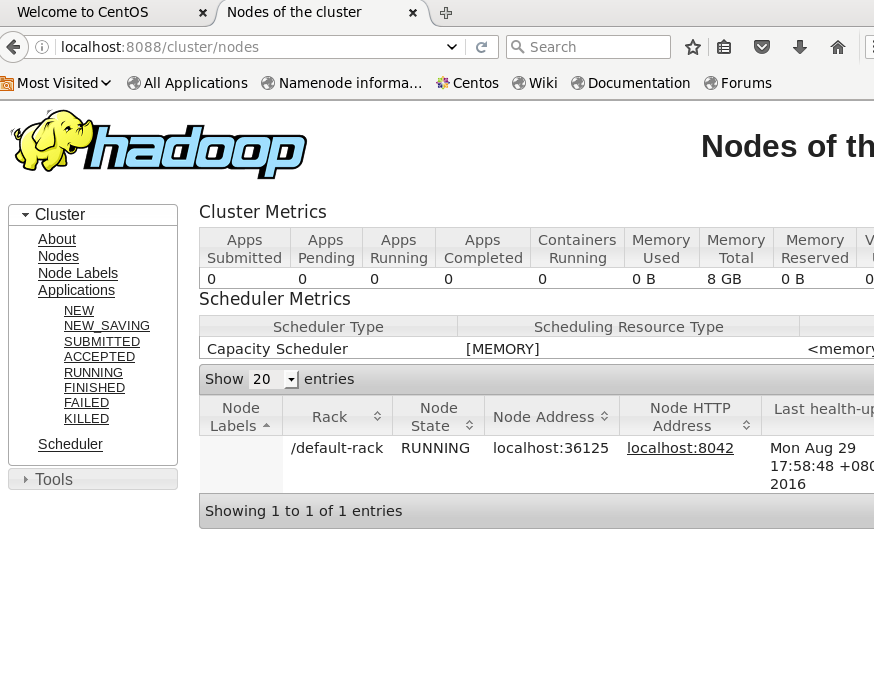
./sbin/start-yarn.sh # 啟動YARN

./sbin/mr-jobhistory-daemon.sh start historyserver # 開啟歷史服務器，就可以從web看執行情況

jps 查看，可以看到多了 NodeManager 和 ResourceManager



啟動 YARN 之後，運行實例的方法還是一樣的，僅僅是資源管理方式、任務調度不同。觀察日誌信息可以發現，不啟用 YARN 時，是 “mapred.LocalJobRunner” 在跑任務，啟用 YARN 之後，是 “mapred.YARNRunner” 在跑任務。啟動 YARN 有個好處是可以通過 Web 界面查看任務的運行情況：，如下圖所示(localhost:8088)。



* + 不啟動 YARN 需重命名 mapred-site.xml

如果不想啟動 YARN，務必把配置文件 mapred-site.xml 重命名，改成 mapred-site.xml.template，需要用時改回來就行。否則在該配置文件存在，而未開啟YARN 的情況下，運行程序會提示“Retrying connect to server: 0.0.0.0/0.0.0.0:8032” 的錯誤，這也是為何該配置文件初始文件名為mapred- site.xml.template。

* + 關閉Yarn

./sbin/stop-yarn.sh

./sbin/mr-jobhistory-daemon.sh stop historyserver

1. HDFS 操作

命令是以”./bin/hadoop dfs”開頭的Shell命令方式，實際上有三種shell命令方式。

* hadoop fs
* hadoop dfs
* hdfs dfs

1. hadoop fs適用於任何不同的文件系統，比如本地文件系統和HDFS文件系統
2. hadoop dfs只能適用於HDFS文件系統
3. hdfs dfs跟hadoop dfs的命令作用一樣，也只能適用於HDFS文件系統
4. X
5. X