|  |
| --- |
| Hadoop cluster |
| 分散式架構設定手冊 |
|  |
| 測試於VMware player4.0.6, CentOS6.6, Hadoop-2.2.0, jdk-7u79-linux-x64.rpm |
| 記錄人:陳建宇  日期:2016/03/23 |

Hadoop cluster

分散式架構設定手冊

# 註解

• • •

為了使運作順利, 建議以具有root權限/群組的帳號來進行整個過程

請避免使用sudo或su的方式來建置hadoop系統, 因為hadoop name node自動啟動data node的行為本身需要root權限, 以及SSH連線權限

如何使普通帳號具有與root相同權限的設定不在此手冊範圍內

SSH連線設定會有說明

CentOS7.X版一部分指令與6.X版不同, 此手冊以6.X版為主

指令行會用不同的字體呈現:

*Ex.This is command*

This is note.

複製文中句子的時候注意word檔的”引號”

# 綱要

1. 在VM下新增CentOS虛擬機器
2. 檢查CentOS虛擬機器的網路狀態
3. 安裝Java設定
4. 關閉CentOS各種網路安全設定
5. 修改CentOS虛擬機器網路映射設定
6. CentOS虛擬機器互ping測試
7. SSH免密碼連線設定與測試
8. 在name node(或稱master主機)建置hadoop的環境變數
9. 設定hadoop核心必要的5份文件
10. 複製master主機上的hadoop設定到data node(或稱slave主機)上
11. 驗收測試
12. 錯誤修正
13. 參考資料

# 在VM下新增CentOS虛擬機器

注意Fig.1的Customize Hardware…的設定

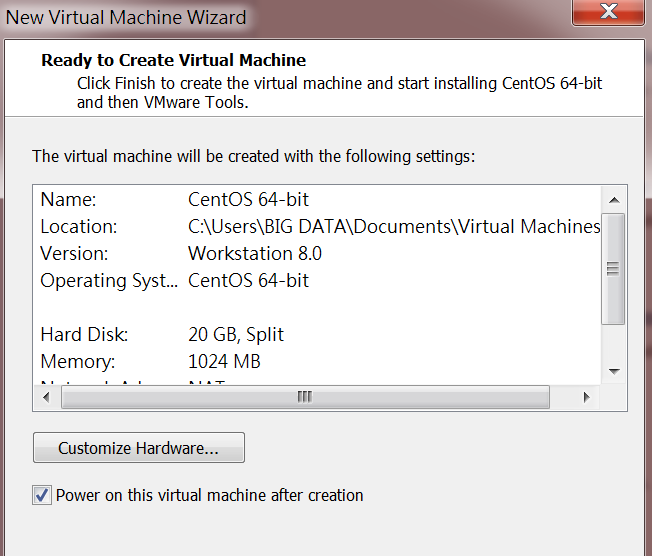


Fig.1

若是在一個host下建置多台虛擬機器互連可以維持NAT連線設定, 若是要與實體的其他電腦連線請更改成bridged模式, 見Fig.2

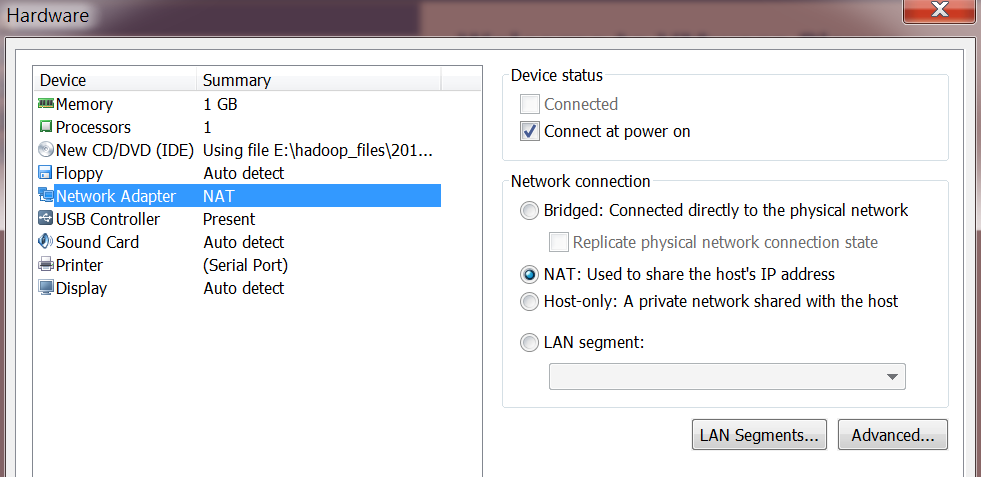


Fig.2

記憶體設定請記得改大, 預設只有1G太少了, 設定完畢請安裝CentOS並以root登入

# 檢查CentOS虛擬機器的網路狀態

登入CentOS後, 右上角滑鼠右鍵會有選單如Fig.3, 選Edit Connections進去, 再按Edit, 點選IPv4 Settings, Method這項選擇Manual, 輸入網路IP等資料, 完成後如Fig.4, 可以打開瀏覽器看能否上網, 或是去ping其他網路上的主機測試

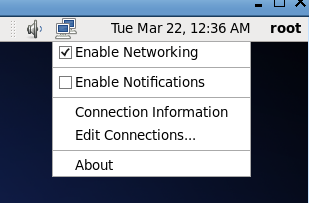


Fig.3

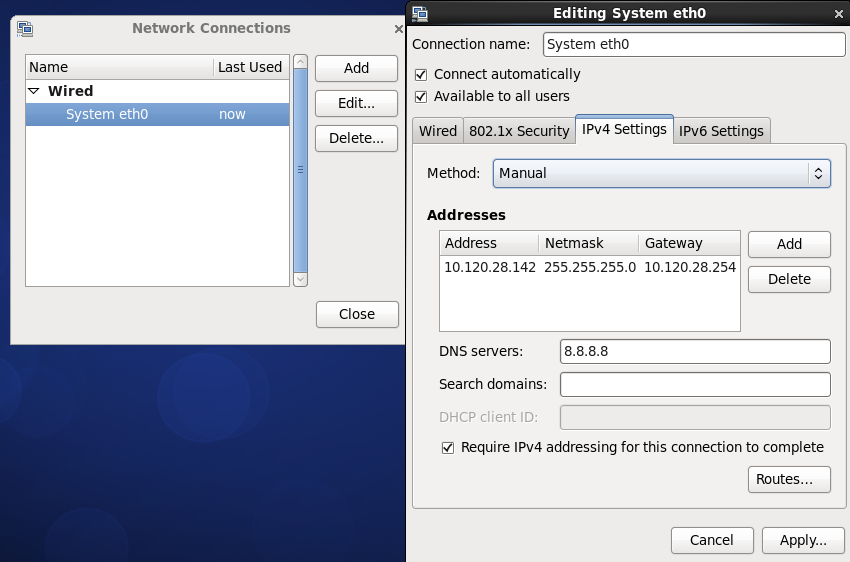


Fig.4

# 安裝Java設定

Hadoop是在Java執行環境(JRE)下運作的, 所以安裝Hadoop之前務必確認Java是否有裝好.

建議在CentOS內直接到Oracle官網上下載JDK套件, 只要是1.7X以上的rpm版本應該都可以.

在下載回來的目錄執行:

r*pm -ivh jdk-7u79-linux-x64.rpm*

*ln -s /usr/java/jdk1.7.0\_79/ /usr/java/java*

然後以vi去修改profile設定:

*vi /etc/profile*

在profile文件內加入下面幾行如Fig.5:

*export JAVA\_HOME=/usr/java/java*

*export JRE\_HOME=$JAVA\_HOME/jre*

*export CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar:$JRE\_HOME/lib/rt.jar*

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

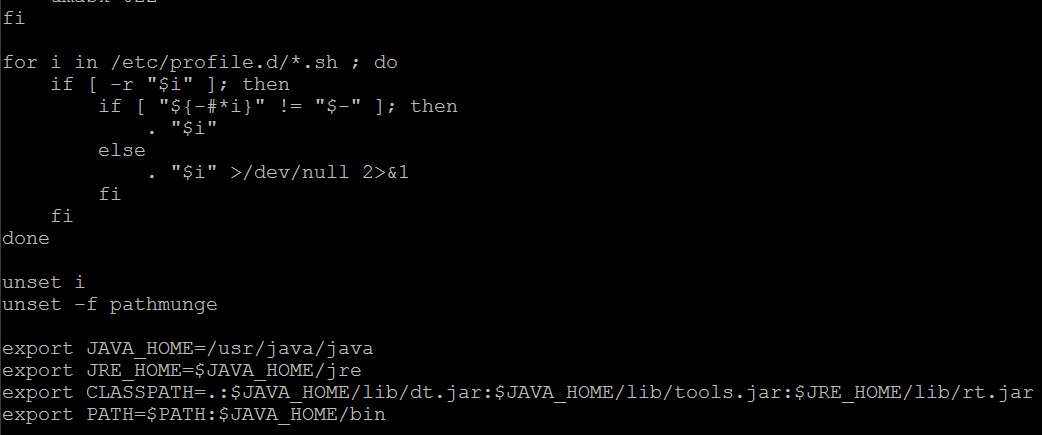


Fig.5

存檔離開profile確認Java:

*java -version*

如果出現類似下面的訊息:

java version "1.7.0\_79"

Java(TM) SE Runtime Environment (build 1.7.0\_79-b15)

Java HotSpot(TM) 64-Bit Server VM (build 24.79-b02, mixed mode)

就是Java裝好了.

注意本節步驟必須所有主機(master與slave)都要做過一遍

# 關閉CentOS各種網路安全設定

這是master主機與slave主機直接連線的設定, 網路上有許多案例表示防火牆會阻擋Hadoop分散式架構的連線, 由於Hadoop使用的port非常多, 所以建議直接關閉防火牆,而不是針對每一個port去設定, 請輸入:

*setenforce 0*

然後用vi去修改config:

*vi /etc/selinux/config*

在config內變更SELINUX如Fig.6:

*SELINUX=disabled*

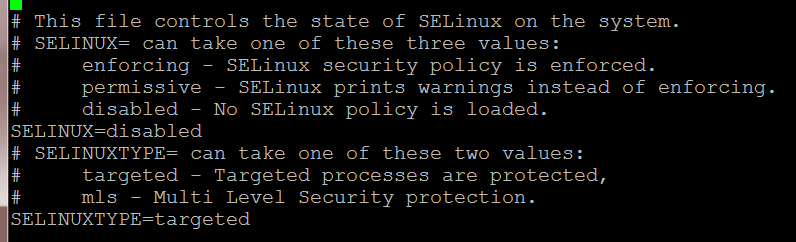


Fig.6

config存檔離開後輸入:

*service iptables stop*

*chkconfig iptables off*

完成後如Fig.7

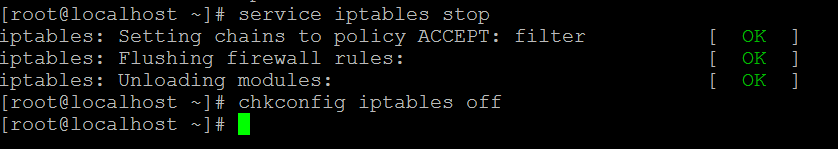


Fig.7

接著關閉SSH密碼, 修改ssh\_config文件:

*vi /etc/ssh/ssh\_config*

在ssh\_config文件內找到StrictHostKeyChecking, 去掉註解的#並把ask修改成no:

*StrictHostKeyChecking n*o

完成如Fig.8後重啟設定:

*service sshd restart*

注意本節步驟必須所有主機(master與slave)都要做過一遍

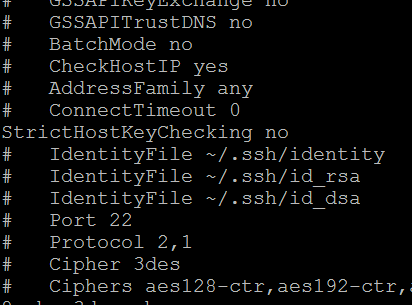


Fig.8

# 修改CentOS虛擬機器網路映射設定

為了方便, 建議修改主機名稱(注意CentOS7.X版不同,改*/etc/hostname*),

注意master與slave要個別設定,

注意主機名稱大小寫有別.

*vi /etc/sysconfig/network*

*HOSTNAME=master*

若是slave主機則是:

*HOSTNAME=slave*

若有多台slave則自己加上編號,如Fig.9

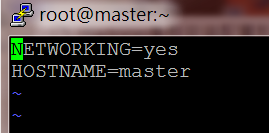
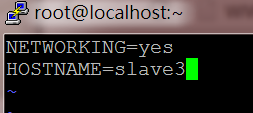
 

Fig.9

再來是修改映射名單:

*vi /etc/hosts*

在/etc/hosts中輸入所有Hadoop要連線的主機IP與主機名稱

且所有主機(master與slave)都要建立此名單, 如Fig.10

注意輸入空格用Tab輸入.

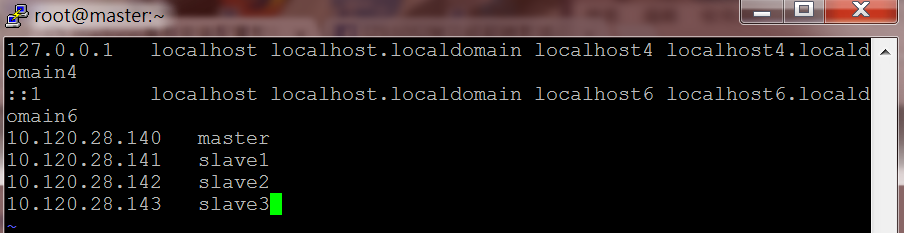


Fig.10

建立名單完成後, 將每台主機都重新開機, 修改才會生效

# CentOS虛擬機器互ping測試

請輸入指令去ping Hadoop連線的主機作測試, 注意請用主機名稱而不是輸入IP,

-c 3是ping 3次後結束

*ping master –c 3*

設定成功的話, 測試結果會如Fig.11, 圖中是slave1去ping slave2與master

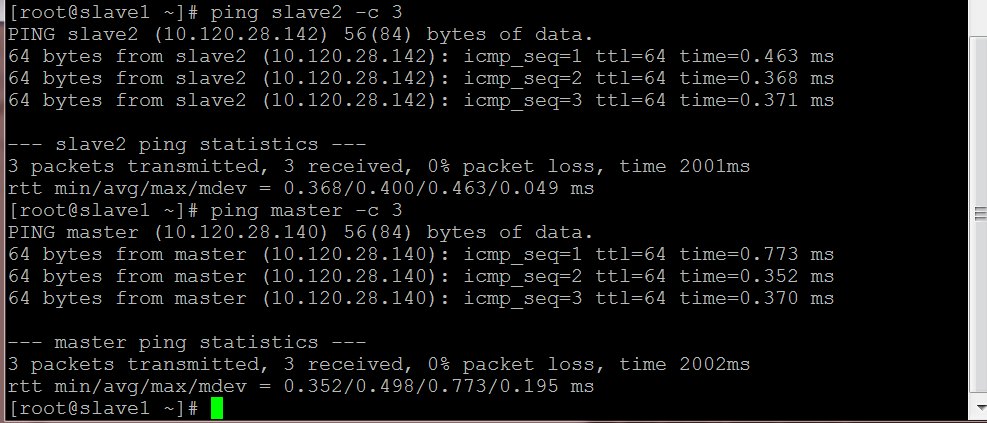


Fig.11

請確定到此為止的所有相關設定都有成功才繼續往下作,

請確保每台主機之間連線都是互通的

# SSH免密碼連線設定與測試

這個步驟是要讓master可以免密碼登入每一台slave,

但slave之間彼此不需要這樣的權限

實現這個狀態的原理是, 以mater主機去產生一個認證檔案(id\_rsa.pub),

然後將此檔案傳到每一台slave上作記錄

在master主機上進行下列操作:

先切換到.ssh資料夾內:

*cd ~/.ssh*

接著清除舊的認證檔案:

*rm ./id\_rsa.\**

產生新的認證檔案,所有的詢問都直接按enter就可以:

*ssh-keygen -t rsa*

讓master本機可以認得自己的認證(加入授權紀錄):

*cat ./id\_rsa.pub >> ./authorized\_keys*

成功的話如Fig.12

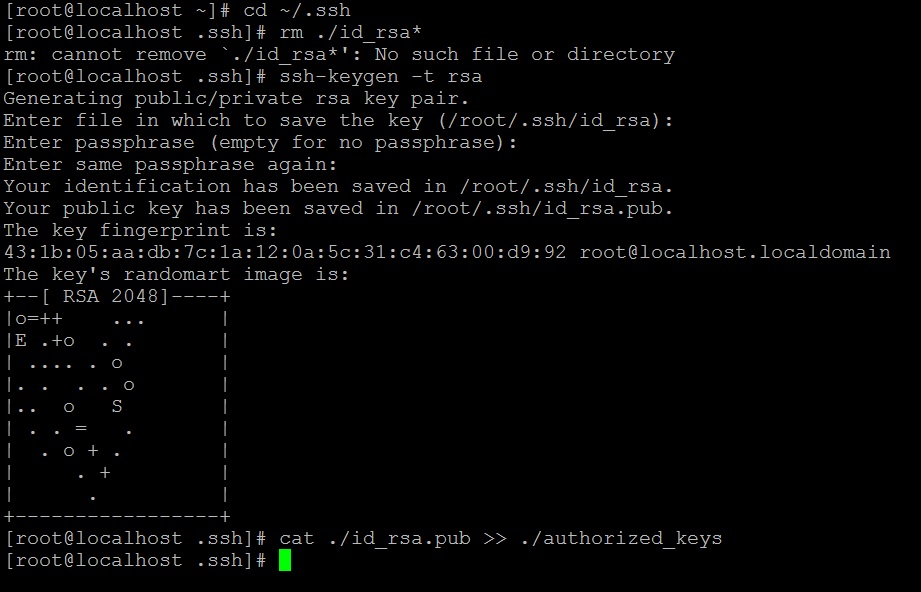


Fig.12

完成後可以輸入:

*ssh master*

第一次執行會有詢問yes/no, 如Fig.13, 請輸入*yes*

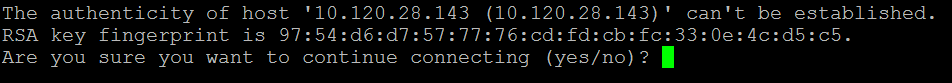


Fig.13

成功的話, 不用密碼就可以執行完畢, 如果仍然要求密碼就是前幾步驟有出錯,

測試完成後請記得輸入*exit*回到*ssh master*之前的狀態

接著要把這個認證檔案丟(傳送)到每一台slave上:

*scp ~/.ssh/id\_rsa.pub root@slave2:/root*

每一台slave都要個別丟一次, 傳送完成的話如Fig.14

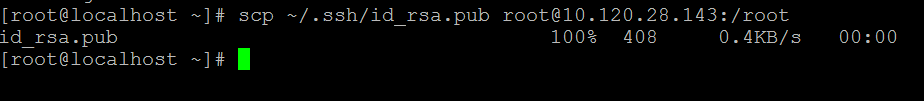


Fig.14

接著在slave主機上作以下的操作:

先切換到.ssh資料夾內:

*cd ~/.ssh*

讓slave主機可以認得master的認證(加入授權紀錄):

*cat ~/id\_rsa.pub >> ./authorized\_keys*

基於安全性的考量, 用過的認證檔案需要移除(這步不做對Hadoop沒有影響):

*rm ~/id\_rsa.pub*

請確定每一台slave主機都有進行上述的步驟, 把master的認證檔案加入授權記錄內

現在來測試是否master不需要密碼就能登入slave主機

在master主機上輸入:

*ssh slave2*

然後登出:

*exit*

成功的話如Fig.15, 注意前面的開頭變化,

登入:[root@master ~]變成[root@slave2 ~]且不用輸入密碼

登出:從[root@slave2 ~]變回[root@master ~]

請登出變回master之後才去試登入下一台slave

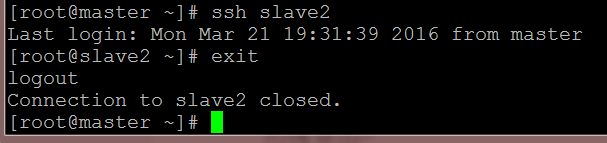


Fig.15

請確認master主機可以無密碼登入到所有slave主機上

請確定到此為止的所有相關設定都有成功才繼續往下作

# 在name node(或稱master主機)建置hadoop的環境變數

下載Hadoop:

*wget* ”[*https://archive.apache.org/dist/hadoop/core/hadoop-2.2.0/hadoop-2.2.0.tar.gz*](https://archive.apache.org/dist/hadoop/core/hadoop-2.2.0/hadoop-2.2.0.tar.gz)”

解壓縮:

*tar -zxvf /tmp/hadoop-2.2.0.tar.gz*

把Hadoop移到opt路徑下:

mv hadoop-2.2.0 /opt

新增環境變數(以下步驟所有主機(master與slave)都要做一次):

*ln -s /opt/hadoop-2.2.0 /opt/hadoop*

修改profile:

*vi /etc/profile*

在profile內增加下列幾行如Fig.16:

*export HADOOP\_HOME=/opt/hadoop/*

*export HADOOP\_MAPRED\_HOME=$HADOOP\_HOME*

*export HADOOP\_COMMON\_HOME=$HADOOP\_HOME*

*export HADOOP\_HDFS\_HOME=$HADOOP\_HOME*

*export YARN\_HOME=$HADOOP\_HOME*

*export HADOOP\_CONF\_DIR=$HADOOP\_HOME/etc/hadoop*

*export YARN\_CONF\_DIR=$HADOOP\_HOME/etc/hadoop*

*export PATH=$PATH:$HADOOP\_HOME/bin:$HADOOP\_HOME/sbin*

*export HADOOP\_COMMON\_LIB\_NATIVE\_DIR=$HADOOP\_INSTALL/lib/native*

*export HADOOP\_OPTS="-Djava.library.path=$HADOOP\_INSTALL/lib"*

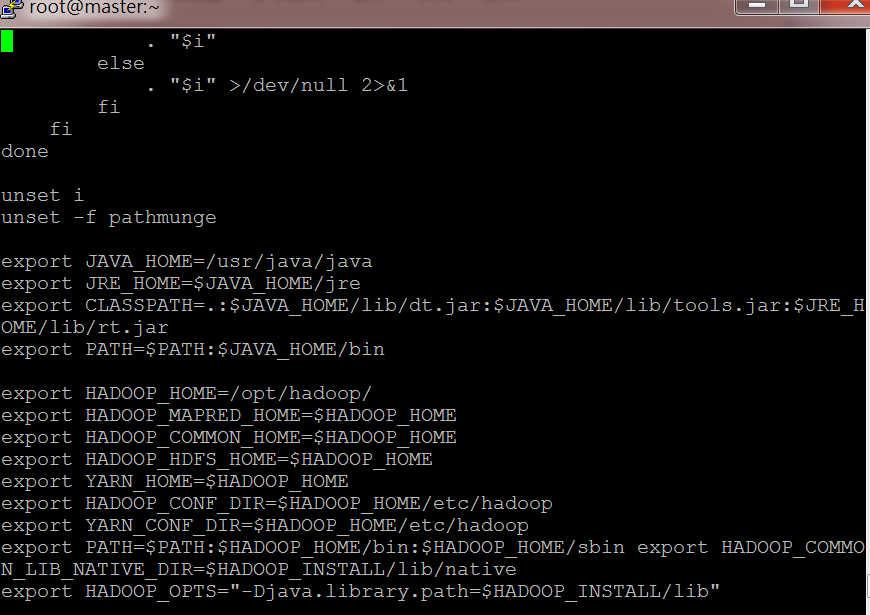


Fig.16

存檔退出後輸入:

*source /etc/profile*

以上步驟所有主機(master與slave)都要做一次

接著以vi修改hadoop-config.sh:

*vi /opt/hadoop/libexec/hadoop-config.sh*

增加下列一行如Fig.17:

*export JAVA\_HOME=/usr/java/java*

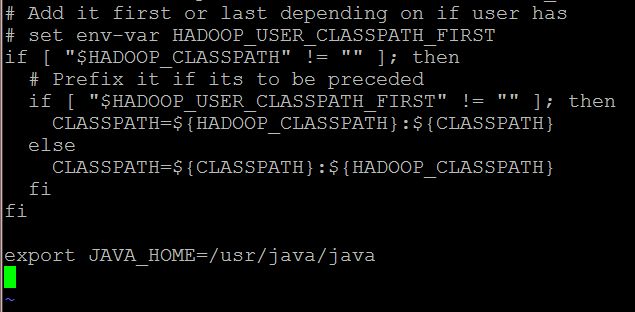


Fig.17

以vi修改hadoop-env.sh:

*vi /opt/hadoop/etc/hadoop/hadoop-env.sh*

增加下列幾行, 如Fig.18:

*export JAVA\_HOME=/usr/java/java*

*export HADOOP\_HOME=/opt/hadoop*

*export PATH=$PATH:$HADOOP\_HOME/bin*

*export PATH=$PATH:$HADOOP\_HOME/sbin*

*export HADOOP\_MAPRED\_HOME=$HADOOP\_HOME*

*export HADOOP\_COMMON\_HOME=$HADOOP\_HOME*

*export HADOOP\_HDFS\_HOME=$HADOOP\_HOME*

*export YARN\_HOME=$HADOOP\_HOME*

*export HADOOP\_COMMON\_LIB\_NATIVE\_DIR=$HADOOP\_HOME/lib/native*

*export HADOOP\_OPTS="-Djava.library.path=$HADOOP\_HOME/lib"*

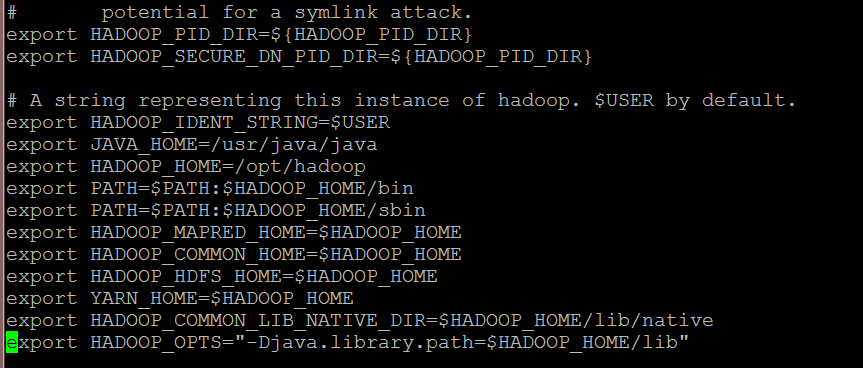


Fig.18

到此為止, master必要的環境變數設置完成, slave主機的前置作業也到此完成

# 設定hadoop核心必要的5份文件

5份文件指的是:

1. slaves
2. core-site.xml
3. hdfs-site.xml
4. mapred-site.xml
5. yarn-site.xml

設定時請仔細核對, 不要有遺漏, 裡面只要有錯誤就無法正常運行hadoop

### 設定slaves文件:

*vi /opt/hadoop/etc/hadoop/slaves*

文件內輸入data node名單, 若不希望master成為data node的話, 可以不加master,

注意格式是固定的, 一行一個主機名字

*slave1*

*slave2*

*master*

完成如Fig.19

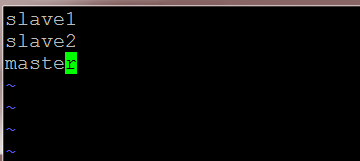


Fig.19

以下的文件請把輸入的內容放在*<configuration></configuration>*之間

### 設定core-site.xml:

*vi /opt/hadoop/etc/hadoop/core-site.xml*

內容輸入:

*<configuration>*

*<property>*

*<name>fs.defaultFS</name>*

*<value>hdfs://master:9000</value>*

*</property>*

*<property>*

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

*<value>/opt/hadoop/tmp</value>*

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

*</property>*

*</configuration>*

完成如Fig.20

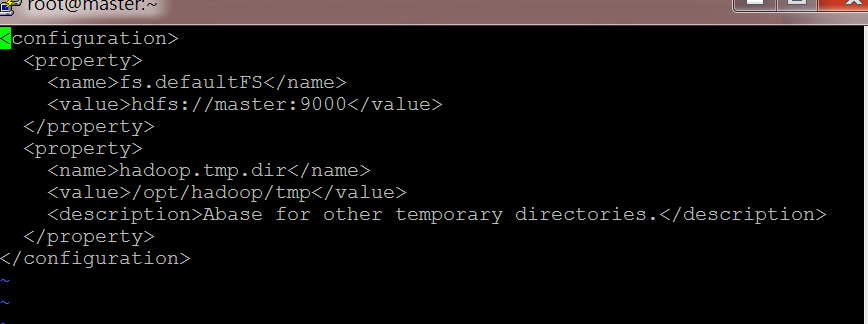


Fig.20

### 設定hdfs-side.xml:

*vi /opt/hadoop/etc/hadoop/hdfs-site.xml*

內容輸入:

*<configuration>*

*<property>*

*<name>dfs.namenode.secondary.http-address</name>*

*<value>master:50090</value>*

*</property>*

*<property>*

*<name>dfs.replication</name>*

*<value>2</value>*

*</property>*

*<property>*

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

*<value>/opt/hadoop/tmp/dfs/name</value>*

*</property>*

*<property>*

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

*<value>/opt/hadoop/tmp/dfs/data</value>*

*</property>*

*</configuration>*

完成如Fig.21

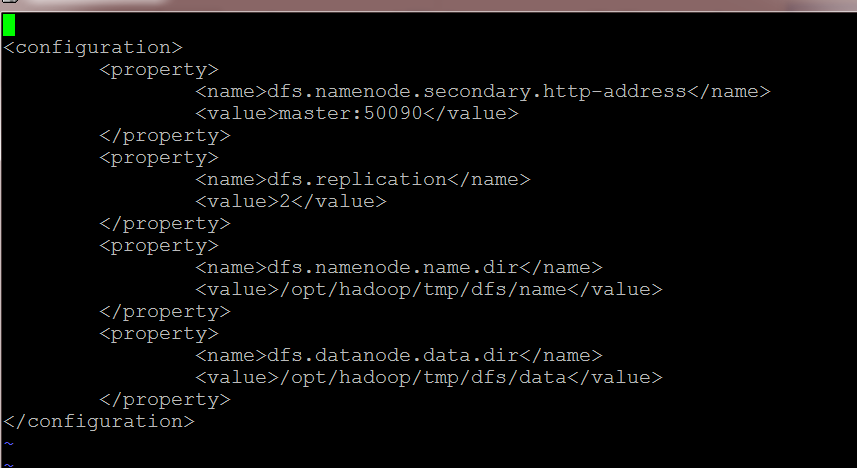


Fig.21

### 設定mapred-side.xml:

*vi /opt/hadoop/etc/hadoop/mapred-site.xml*

內容輸入, 如Fig.22:

*<configuration>*

*<property>*

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

*<value>yarn</value>*

*</property>*

*<property>*

*<name>mapreduce.jobhistory.address</name>*

*<value>master:10020</value>*

*</property>*

*<property>*

*<name>mapreduce.jobhistory.webapp.address</name>*

*<value>master:19888</value>*

*</property>*

*</configuration>*

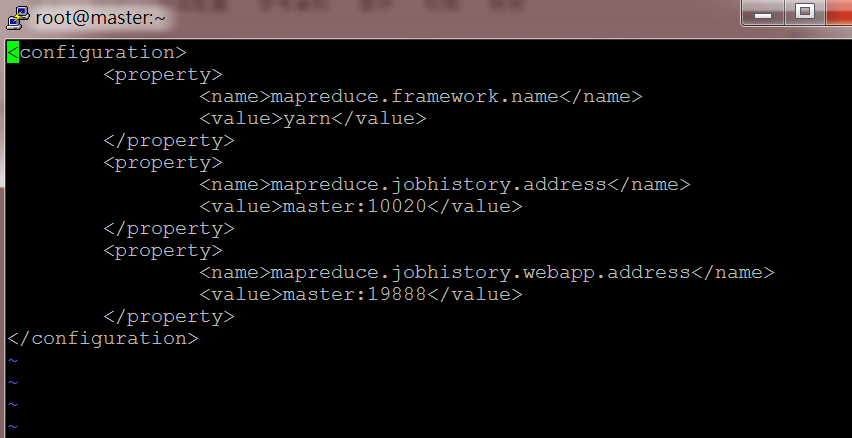


Fig.22

### 設定yarn-side.xml:

*vi /opt/hadoop/etc/hadoop/yarn-site.xml*

內容輸入, 如Fig.23:

*<configuration>*

*<property>*

*<name>yarn.resourcemanager.hostname</name>*

*<value>master</value>*

*</property>*

*<property>*

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

*<value>mapreduce\_shuffle</value>*

*</property>*

*</configuration>*

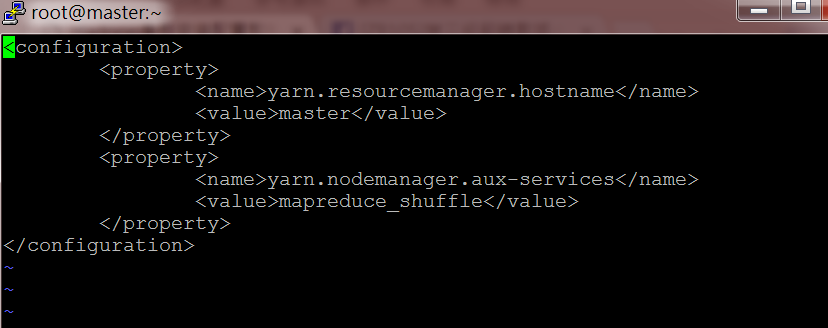


Fig.23

到此為止master主機的hadoop設定全部結束

### 補充說明:

此手冊的hadoop安裝路徑是/opt/hadoop-2.2.0/

另有產生一條捷徑/opt/hadoop/指向上述路徑,

若希望把hadoop安裝到別的路徑上, 則所有提到hadoop路徑的文件都需要自行修改

若希望主機名稱不是master或slave,

則需要在前面映射名稱那邊修改,

且後續提到master或slave的地方都需要自行修改,

若不想設定映射名稱, 則提到主機名稱的地方可以直接輸入IP替代

# 複製master主機上的hadoop設定到data node(或稱slave主機)上

切換到/opt目錄:

*cd /opt*

清除舊的紀錄(若安裝錯誤失敗,修正錯誤後; 或安裝不同模式):

*rm –rf ./hadoop/tmp*

*rm –rf ./hadoop/logs/\**

打包master主機上的hadoop檔案:

*tar -zcf ./hadoop.master.tar.gz ./hadoop-2.2.0*

注意打包的時候不要打包捷徑hadoop,

這會變成把捷徑檔打包,

必須指定是hadoop-2.2.0才是正確的,

輸入ls查看結果如Fig.24

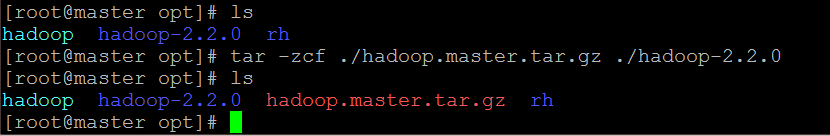


Fig.24

將打包的檔案丟到每一台slave主機的/opt目錄上:

*scp ./hadoop.master.tar.gz slave1:/opt*

丟成功會出現Fig.25的訊息.

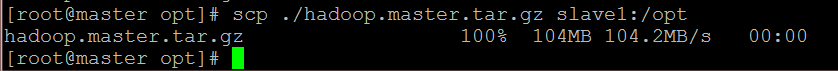


Fig.25

你可以在master主機上輸入*ssh slave*的方式遙控slave主機完成以下的步驟,

也可以真的換去slave主機進行以下的步驟

在slave主機上進到/opt目錄:

*cd /opt*

若/opt下有舊的hadoop-2.2.0資料夾目錄, 則先刪除掉:

*rm –rf hadoop-2.2.0*

在slave主機上將hadoop.master.tar.gz解開,會得到hadoop-2.2.0的資料夾:

*tar –zxf ./hadoop.master.tar.gz*

接著改變檔案的擁有者,讓外來檔案就地合法化:

*chown –R root:root /opt/hadoop-2.2.0*

上述步驟必須在每一台slave上都做過一遍

# 驗收測試

在master上建立一個目錄:

*mkdir -p $HADOOP\_HOME/tmp*

接著對master主機的hadoop namenode執行格式化:

*hadoop namenode -format*

做完之後, 啟動hadoop master主機

(master會自動啟動slave名單上的所有slave主機):

*start-all.sh*

接著可以啟動歷程記錄器(可以不啟動):

*mr-jobhistory-daemon.sh start historyserver*

啟動完成後, 可以用jps指令查看該主機啟動了哪些服務項目:

*jps*

正常運作應該會有6個服務項目如Fig26, 若master不負責datanode的話可能會是5個

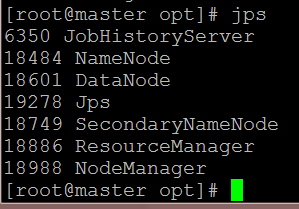


Fig.26

而slave則會有3個服務項目,如Fig27

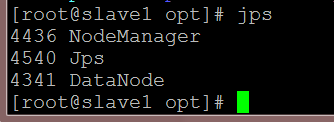


Fig.27

在master主機輸入以下指令可以確認所有data node狀態, 如Fig.28:

*hdfs dfsadmin –report*

也可以在網頁上確認狀況, 如Fig29(網址為master主機的IP與port8088,如10.120.28.140:8088)

若架設數台data node但顯示”Datanodes available: 1 (1 total, 0 dead)”

這表示slave的設定有錯誤, 沒有被正確的啟動

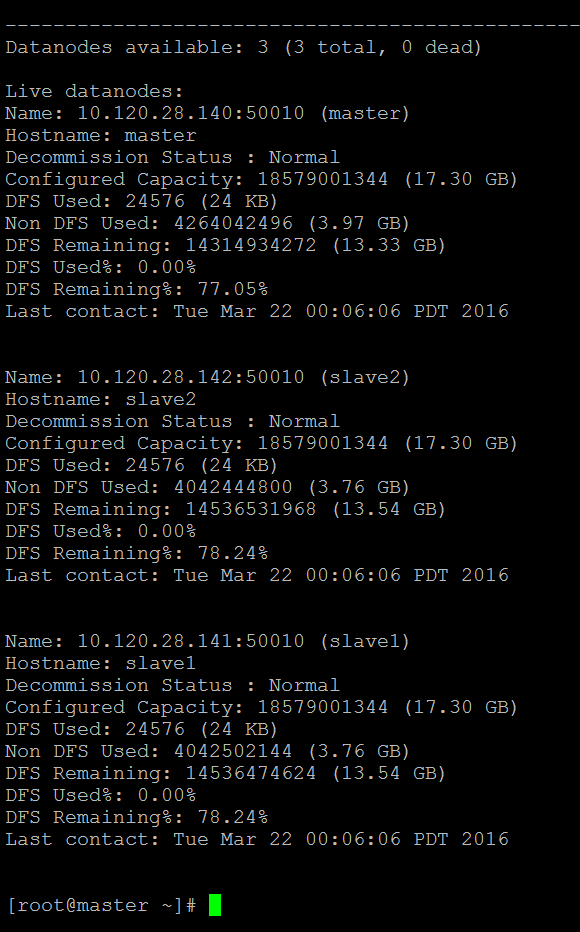


Fig.28

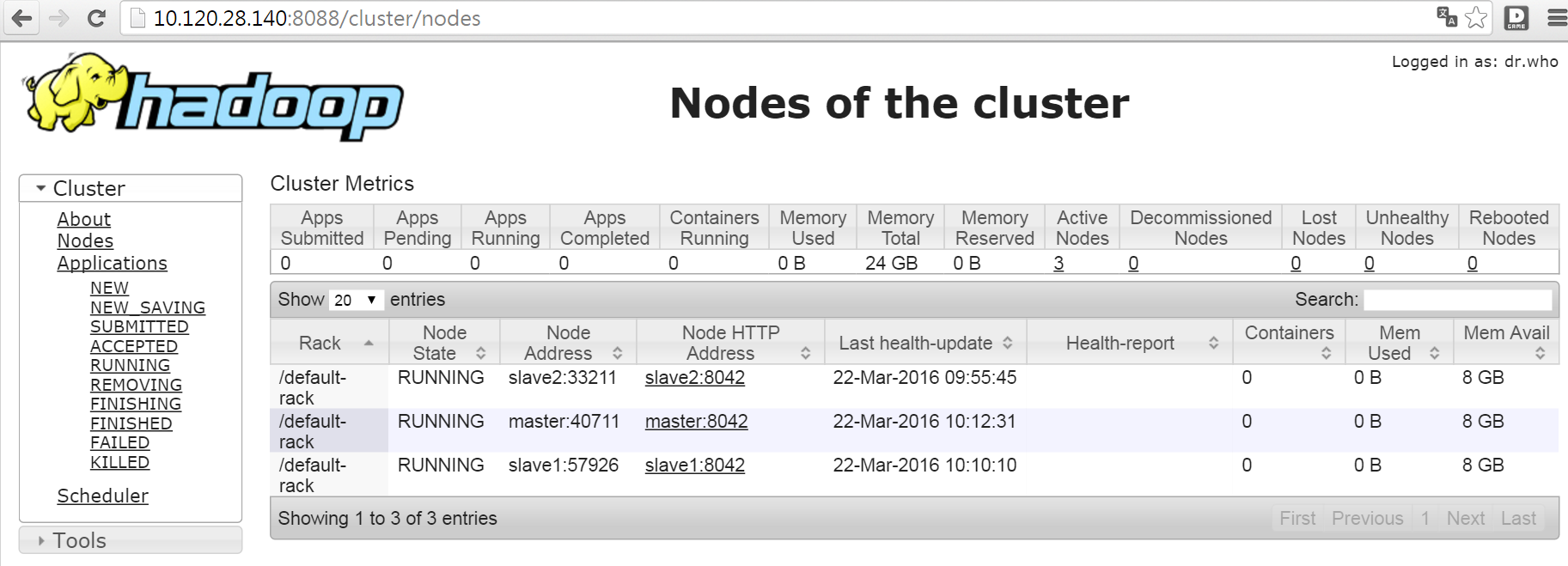


Fig.29

# 錯誤修正

要除錯的話, 可以從網頁的NameNode logs去檢查, 如Fig30

(網址為master主機的IP與port50070,如10.120.28.140:50070)

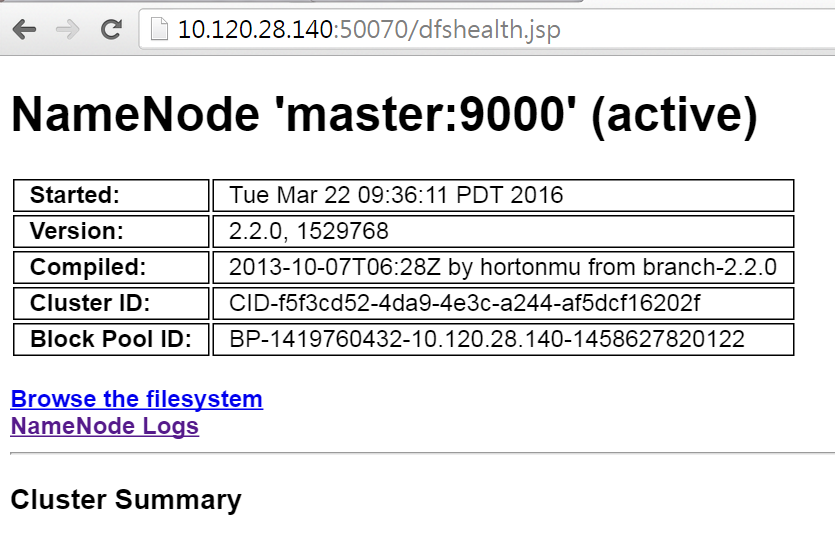


Fig.30

點入Namenode logs後, 出現如Fig.30的清單,

通常只需要在hadoop-root-datanode-master.log裡面找尋warn或error或exception的字眼,

再回頭核對是哪份文件設定錯誤來除錯

(這些文件存放於/opt/hadoop/logs路徑下, 若網頁打不開也可以直接在master主機上查詢)

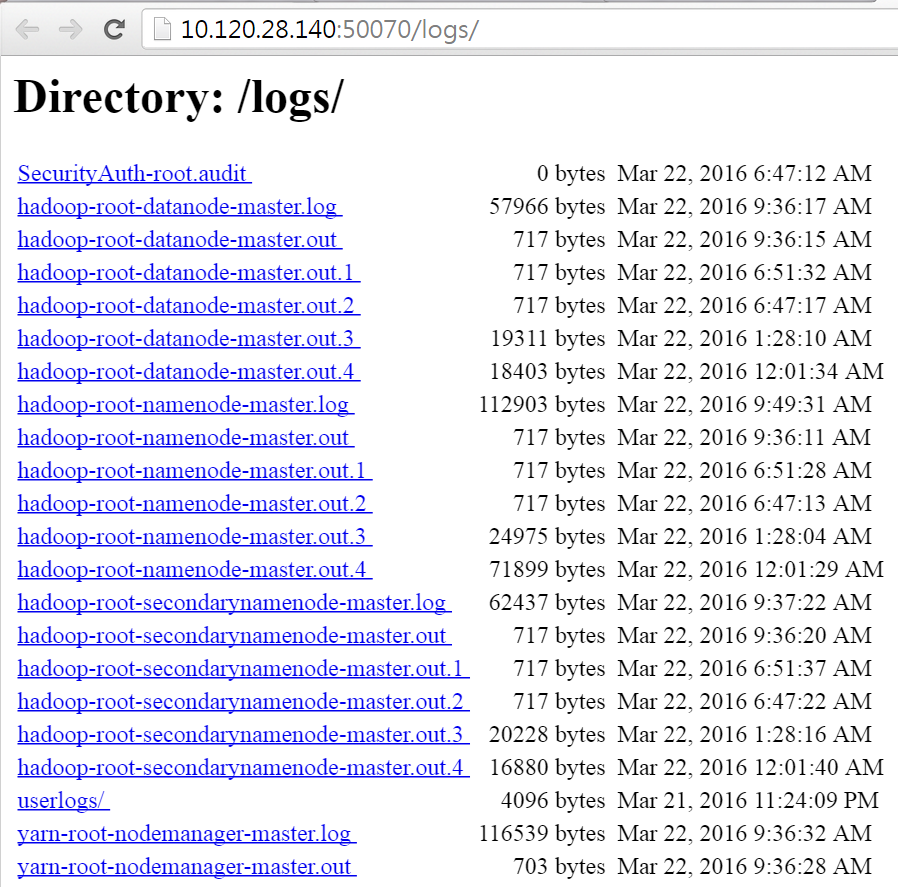


Fig.30

若要修正錯誤, 則需要先停止hadoop運作, 在master主機上輸入:

*stop-all.sh*

在master主機上修正錯誤的文件之後,

重跑一遍*複製master主機上的hadoop設定到data node(或稱slave主機)上,*

到*驗收測試*為止的流程

若要關閉linux主機, 則必須先關閉Hadoop架構(*stop-all.sh*)後才可以執行

# 參考資料

* Big Data之處理與分析實務班-徐凡耘老師的pdf檔
* Hadoop集群安裝配置教程 http://www.powerxing.com/install-hadoop-cluster/