HadoopClustersInstall

☆ Dockerfile

```
FROM Ubuntu:18.04

MAINTAINER amber

# apt-get clean, update
RUN apt-get clean all && apt-get update

# Install
RUN apt-get install -y python3.6 ipython3 python3-pip iputils-
  ping
RUN apt-get install -y openssh-server wget git vim curl
RUN pip3 install numpy pandas jieba

# Install Java ( jdk-8u144 )
RUN cd /tmp && wget https://mail-tp.fareoffice.com/java/jdk-
  8u144-linux-x64.tar.gz && tar -zxvf /tmp/jdk-8u144-linux-
    x64.tar.gz
RUN cd /tmp && mkdir /usr/java && mv /tmp/jdk1.8.0_144 /usr/java
  && ln -s /usr/java/jdk1.8.0_144/ /usr/java/java

# Create Public Key and SSH without Key
RUN ssh-keygen -t rsa -f ~/.ssh/id_rsa -P '' && \
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys

ENTRYPOINT ["/bin/bash", "-c", "service ssh start; /bin/bash"]

ENV JAVA_HOME /usr/java/java
ENV JRE_HOME $JAVA_HOME/jre
ENV
  CLASSPATH .:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar:
    $JRE_HOME/lib/rt.jar
ENV PATH $PATH:$JAVA_HOME/bin

CMD service ssh status && /bin/bash
```

☆ Build Docker Image

```
sudo docker build -t chintz/hadoop:19.05.25 . --no-cache
sudo docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
chintz/hadoop	19.05.25	33eb2aea1132	4 hours ago	1.74GB

☆ Create 3 Container

```
sudo docker run --name master --hostname master -p 50070:50070 –p
  8088:8088 -it 33eb2aea1132
sudo docker run --name slaver1 --hostname slaver1 –it
  33eb2aea1132
sudo docker run --name slaver2 --hostname slaver2 –it
  33eb2aea1132
sudo docker ps –a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
22d9ba077432	33eb2aea1132	"/bin/bash -c 'servi…"	4 hours ago	Up 4 hours		slaver2
72839ce6dd6f	33eb2aea1132	"/bin/bash -c 'servi…"	4 hours ago	Up 4 hours		slaver1
e382608548b3	33eb2aea1132	"/bin/bash -c 'servi…"	4 hours ago	Up 4 hours	0.0.0.0:8088->8088/tcp, 0.0.0.0:50070->50070/tcp	master

---

☆ Other settings (Every Sever)

```
apt-get clean all
apt-get update
apt-get -y upgrade
apt-get install ntp

vi /etc/profile
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
```

查看 Java 版本，確認是否安裝成功

```
java -version
```

```
java version "1.8.0_144"
Java(TM) SE Runtime Environment (build 1.8.0_144-b01)
Java HotSpot(TM) 64-Bit Server VM (build 25.144-b01, mixed mode)
```

☆ SSH Connection Setting (Master)

設定主機資訊

```
vi /etc/hosts
已有 172.17.0.2      master
加入 172.17.0.3      slaver1
加入 172.17.0.4      slaver2
```

複製 Key 到其他兩台

```
ssh-copy-id -i ~/.ssh/id_rsa.pub root@slaver1
ssh-copy-id -i ~/.ssh/id_rsa.pub root@slaver2
```

下載 Hadoop 2.7.3

```
cd /tmp; wget https://archive.apache.org/dist/hadoop/core/hadoop-
  2.7.3/hadoop-2.7.3.tar.gz
```

☆ Other Settings (Slaver1)

複製 Key 到 Master、Slaver2

```
ssh-copy-id -i ~/.ssh/id_rsa.pub root@172.17.0.2
ssh-copy-id -i ~/.ssh/id_rsa.pub root@172.17.0.4
```

☆ Install Hadoop (Master)

安裝 Hadoop

```
tar -zxvf /tmp/hadoop-2.7.3.tar.gz
mv hadoop-2.7.3 /opt
```

新增環境變數

```
vi /etc/profile
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_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib"

vi /opt/hadoop-2.7.3/libexec/hadoop-config.sh
export JAVA_HOME=/usr/java/java

vi /opt/hadoop-2.7.3/etc/hadoop/hadoop-env.sh
export JAVA_HOME=/usr/java/java
export HADOOP_HOME=/opt/hadoop
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
export HDFS_NAMENODE_USER=root export HDFS_DATANODE_USER=root
export HDFS_JOURNALNODE_USER=root
export YARN_RESOURCEMANAGER_USER=root
export YARN_NODEMANAGER_USER=root export HDFS_ZKFC_USER=root
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"
```

Git Clone 設定檔

```
cd /tmp
git clone https://github.com/orozcohsu/hadoop-2.7.3-ha.git
cd /tmp/hadoop-2.7.3-ha
cp * /opt/hadoop-2.7.3/etc/hadoop/
```

Install Zookeeper

先在本機將 Zookerper 放進 Master

```
sudo docker cp ~/Templates/zookeeper-3.4.9.tar.gz
  e382608548b3:/tmp
```

回到 Msater

```
cd /tmp
tar -zxvf zookeeper-3.4.9.tar.gz
mv zookeeper-3.4.9 /opt
ln -s /opt/zookeeper-3.4.9 /opt/zookeeper
cp /opt/zookeeper/conf/zoo_sample.cfg /opt/zookeeper/conf/zoo.cfg
```

```
vi /opt/zookeeper/conf/zoo.cfg
dataDir=/tmp/zookeeper    # 將 /tmp 改成 /opt
# 找到這一行 clientPort=2181 後加入
server.1=master:2888:3888
server.2=slaver1:2888:3888
server.3=slaver2:2888:3888

vi /opt/zookeeper/myid
輸入 1

# 複製至 Slaver1, Slaver2 ( 完成後，個別編輯 vi /opt/zookeeper/myid 檔案編號為 2、3 )
scp -rp /opt/zookeeper root@slaver1:/opt/zookeeper
scp -rp /opt/zookeeper root@slaver2:/opt/zookeeper

# 建立 Slaver 資訊
vi /opt/hadoop-2.7.3/etc/hadoop/slaves
master
slaver1
slaver2

# 複製檔案至 Slaver1, Slaver2
scp -rp /etc/hosts root@slaver1:/etc/hosts
scp -rp /etc/hosts root@slaver2:/etc/hosts
scp -rp /opt/hadoop-2.7.3/ root@slaver1:/opt/hadoop-2.7.3
scp -rp /opt/hadoop-2.7.3/ root@slaver2:/opt/hadoop-2.7.3
scp -rp /etc/profile root@slaver1:/etc/profile
scp -rp /etc/profile root@slaver2:/etc/profile

# 建立 Hadoop 軟連結 ( !!!在每台電腦做!!! )
ln -s /opt/hadoop-2.7.3 /opt/hadoop

service ntp start

# 啟動 Zookeeper ( !!!在每台電腦做!!! )
/opt/zookeeper/bin/zkServer.sh start

# 待三台 ZK 皆開啟後，查看每台 ZK 狀態
/opt/zookeeper/bin/zkServer.sh status

# 啟動 Journalndoe ( !!!在每台電腦做!!! )
hadoop-daemon.sh start journalnode
```

建立 tmp 目錄
```
mkdir -p $HADOOP_HOME/tmp
mkdir -p $HADOOP_HOME/tmp/dfs/name
mkdir -p $HADOOP_HOME/tmp/dfs/data
mkdir -p $HADOOP_HOME/tmp/journal
```

修改權限
```
chmod 777 $HADOOP_HOME/tmp
```

複製檔案至 Slaver1, Slaver2
```
scp -rp $HADOOP_HOME/tmp slaver1:/opt/hadoop
scp -rp $HADOOP_HOME/tmp slaver2:/opt/Hadoop
```
HDFS 格式化
```
hdfs namenode -format
```

ZK 格式化
```
hdfs zkfc -formatZK
```

確定 ZK 格式化成功
```
/opt/zookeeper/bin/zkCli.sh -server 127.0.0.1:2181
```

進入交談視窗後
```
ls /
```
會看到[zookeeper, hadoop-ha]

如果 Format 失敗 (!!!三台都要做!!!)

Step 1. 進入交談視窗，刪除 hadoop-ha
```
rmr /hadoop-ha
```
Step 2. 砍掉目錄
```
rm -r -f /opt/hadoop/tmp
```
Step 3. 關閉所有服務
```
stop-all.sh
jps  # 查看服務
kill -p 服務代號  # 移除未關閉服務
```
Step 4. 啟動 Zookeeper 與 Journalnode
```
/opt/zookeeper/bin/zkServer.sh start
hadoop-daemon.sh start journalnode
```
Step 5. 再從建立 /tmp 目錄那段開始從做

啟動 Hadoop 服務 (Master ，確定三台的 ZK、JN 服務都啟動)
```
start-all.sh
```

☆ Other Settings (Slaver1)

在 Slaver1 上，將此主機進行 NameNode 目錄服務格式化，並把 Master 上的 NameNode 目錄資 料複製過來

```
hdfs namenode -bootstrapStandby
```

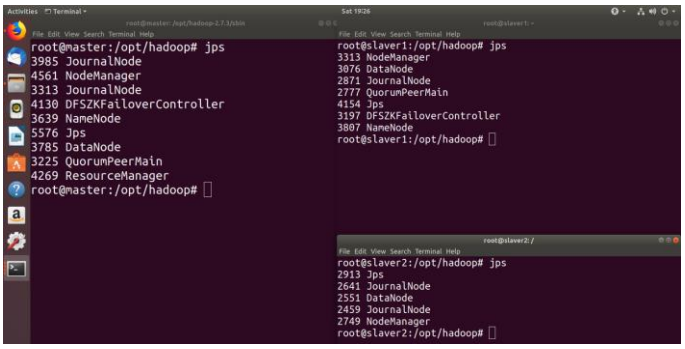
做完上述後，啟動 Slaver1 上的 NN 服務

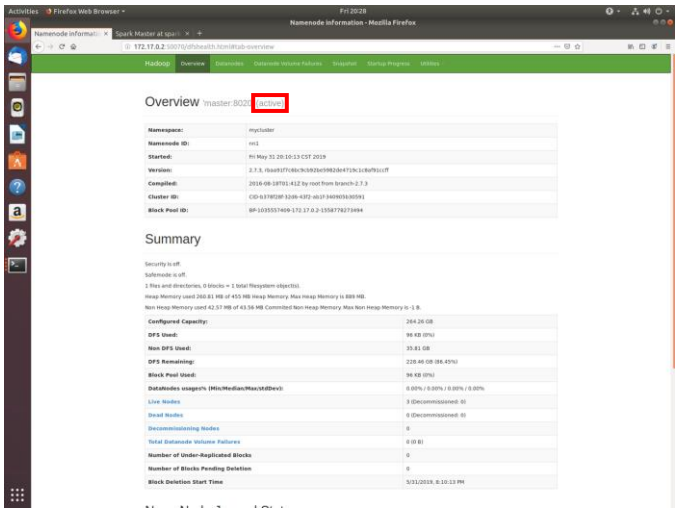
```
hadoop-daemon.sh start namenode
```

☆ Success

瀏覽二台 (Master 與 Slaver1) 的 NameNode
http://ip:50070　會有 3 個 Datanode ，二台狀態分別為 Active 與 Standby
http://ip:8088 (Slaver1 看不到 8088)

查看服務狀態
jps





☆ If Reboot

設定主機資訊 (!!!每次開機都要重新加入!!!)
vi /etc/hosts
已有 **172.17.0.2 master**
加入 **172.17.0.3 slaver1**
加入 **172.17.0.4 slaver2**

複製檔案至 Slaver1, Slaver2
scp -rp /etc/hosts root@slaver1:/etc/hosts
scp -rp /etc/hosts root@slaver2:/etc/hosts

確認資訊是否存在
vi /opt/zookeeper/myid
vi /opt/hadoop-2.7.3/etc/hadoop/slaves

啟動 Zookeeper 與 Journalnode (!!!三台都要做!!!)
/opt/zookeeper/bin/zkServer.sh start
hadoop-daemon.sh start journalnode
有可能會顯示 bash: hadoop-daemon.sh: command not found
先 **source /etc/profile** 後應該就可啟動

啟動 Hadoop 服務 (Master ，確定三台的 ZK、JN 服務都啟動)
start-all.sh

☆ Helper

查詢 Container IP
sudo docker inspect --format '{{ .NetworkSettings.IPAddress }}'
containername
sudo docker inspect --format '{{ .NetworkSettings.IPAddress }}'
master slaver1 slaver2

將 Zookeeper 壓縮檔，搬到 Container 的 /tmp 目錄下
sudo docker cp ~/Templates/zookeeper-3.4.9.tar.gz
e382608548b3:/tmp