VirtualBox+Ubuntu+Hadoop 多節點安裝步驟教學

1. 複製單節點虛擬機，命名為 DataNode1
   (1) 在 VirtualBox 管理員中選取欲複製的虛擬機，再點選「機器」→「再製」(或將游標移至欲複製的虛擬機按下滑鼠右鍵，再點選「再製」)
   (2) 輸入虛擬機名稱：DataNode1，**勾選**「重新初始化所有網路卡的 MAC 位址」，點擊「下一步」
   (3) **勾選**「完整再製」，點擊「再製」

2. 設定網路介面卡
   (1) 在 VirtualBox 管理員中選取 DataNode1，再點選「設定值」
   (2) 點選左側「網路」
   (3) 點選「介面卡1」頁籤，勾選「啟用網路卡」，附加到：選取「NAT」
   (4) 點選「介面卡2」頁籤，勾選「啟用網路卡」，附加到：選取「內部網路」
   (5) 點擊「確定」

3. 將網路卡命名變更為 eth0、eth1、…
   (1) 啟動 DataNode1
   (2) sudo gedit /etc/default/grub
   將「GRUB_CMDLINE_LINUX=""」這一行
   改成「GRUB_CMDLINE_LINUX="net.ifnames=0 biosdevname=0"」
   (3) sudo grub-mkconfig -o /boot/grub/grub.cfg
   (4) sudo update-grub

4. 確認/etc/NetworkManager/NetworkManager.conf 中之 managed=false
   cat /etc/NetworkManager/NetworkManager.conf

5. 修改"/etc/sysctl.conf"，將# net.ipv4.ip_forward=1此行的# (註解)拿掉
   sudo gedit /etc/sysctl.conf

6. 將"/proc/sys/net/ipv4/ip_forward"的值修改為1
   sudo gedit /proc/sys/net/ipv4/ip_forward
   儲存時發生錯誤，點選「Drop change and reload」

7. 編輯網路設定檔設定固定 IP
   sudo gedit /etc/network/interfaces
   輸入下列內容

```
# interfaces(5) file used by ifup(8) and ifdown(8)

auto lo
iface lo inet loopback

#以上是檔案中原有的內容

# NAT interface
auto eth0
iface eth0 inet dhcp

# intnet interface
auto eth1
```

```
iface eth1 inet static
address        192.168.56.101
netmask        255.255.255.0
network        192.168.56.0
broadcast      192.168.56.255
```

8. 設定 hostname

   sudo gedit /etc/hostname

   輸入下列內容:

   ```
   DataNode1
   ```

9. 設定 hosts 檔案

   sudo gedit /etc/hosts

   加入下列內容:

   ```
   192.168.56.100 MasterNode
   192.168.56.101 DataNode1
   192.168.56.102 DataNode2
   192.168.56.103 DataNode3
   ```

10. 修改 core-site.xml

    sudo gedit /usr/local/hadoop/etc/hadoop/core-site.xml

    在<configuration></configuration>之間，輸入下列內容:

    ```
    <property>
      <name>fs.default.name</name>
      <value>hdfs://MasterNode:9000</value>
    </property>
    ```

11. 修改 yarn-site.xml

    sudo gedit /usr/local/hadoop/etc/hadoop/yarn-site.xml

    在<configuration></configuration>之間，加入下列內容:

    ```
    <property>
       <name>yarn.resourcemanager.resource-tracker.address</name>
       <value>MasterNode:8025</value>
    </property>
    <property>
       <name>yarn.resourcemanager.scheduler.address</name>
       <value>MasterNode:8030</value>
    </property>
    <property>
       <name>yarn.resourcemanager.address</name>
       <value>MasterNode:8050</value>
    </property>
    ```

12. 修改 mapred-site.xml

    sudo gedit /usr/local/hadoop/etc/hadoop/mapred-site.xml

    在<configuration></configuration>之間，輸入下列內容:

    ```
    <property>
      <name>mapred.job.tracker</name>
      <value>MasterNode:54311</value>
    </property>
    ```

13. 修改 hdfs-site.xml
    sudo gedit /usr/local/hadoop/etc/hadoop/hdfs-site.xml
    在<configuration></configuration>之間，輸入下列內容:

```
<property>
  <name>dfs.replication</name>
  <value>3</value>
</property>
<property>
  <name>dfs.datanode.data.dir</name>
  <value> file:/usr/local/hadoop/hadoop_data/hdfs/datanode</value>
</property>
```

14. 重新啟動
    reboot

15. 確認網路設定
    ifconfig

```
eth0      Link encap:Ethernet  HWaddr 08:00:27:60:7f:ab
          inet addr:10.0.2.15  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe60:7fab/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:55 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1873 (1.8 KB)  TX bytes:6191 (6.1 KB)

eth1      Link encap:Ethernet  HWaddr 08:00:27:c8:32:6a
          inet addr:192.168.56.101  Bcast:192.168.56.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fec8:326a/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:48 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:5716 (5.7 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:6 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:338 (338.0 B)  TX bytes:338 (338.0 B)
```

16. Shutdown DataNode1
    sudo shutdown -h now

17. 複製 DataNode1伺服器至 DataNode2、DataNode3、MasterNode

18. 設定記憶體
    MasterNode:8GB、DataNode1:4GB、DataNode2:4GB、DataNode3:4GB

(1) 在 VirtualBox 管理員中選取 MasterNode，再點選「設定值」
(2) 點選左側「系統」
(3) 點選「主機板」頁籤，基本記憶體調整為8192MB
(4) 點擊「確定」

19. 設定 DataNode2伺服器
(1) 啟動 DataNode2
(2) 設定 DataNode2固定 IP
sudo gedit /etc/network/interfaces

```
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp

auto eth1
iface eth1 inet static
address        192.168.56.102
netmask        255.255.255.0
network        192.168.56.0
broadcast      192.168.56.255
```

20. 設定 DataNode2主機名稱
sudo gedit /etc/hostname
輸入下列內容:

```
DataNode2
```

21. Shutdown DataNode2
sudo shutdown -h now

22. 設定 DataNode3伺服器
(1) 啟動 DataNode3
(2) 設定 DataNode3固定 IP
sudo gedit /etc/network/interfaces

```
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp

auto eth1
iface eth1 inet static
address        192.168.56.103
netmask        255.255.255.0
network        192.168.56.0
broadcast      192.168.56.255
```

23. 設定 DataNode3主機名稱
sudo gedit /etc/hostname

輸入下列內容:

DataNode3

24. 設定 MasterNode 伺服器

(1) 啟動 MasterNode

(2) 設定 MasterNode 固定 IP

sudo gedit /etc/network/interfaces

```
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp

auto eth1
iface eth1 inet static
address 192.168.56.100
netmask 255.255.255.0
network 192.168.56.0
broadcast 192.168.56.255
```

25. Shutdown DataNode3

sudo shutdown -h now

26. 設定 MasterNode 主機名稱

sudo gedit /etc/hostname

輸入下列內容:

MasterNode

27. 設定 hdfs-site.xml

sudo gedit /usr/local/hadoop/etc/hadoop/hdfs-site.xml

```
<property>
  <name>dfs.replication</name>
  <value>3</value>
</property>
<property>
  <name>dfs.namenode.name.dir</name>
  <value> file:/usr/local/hadoop/hadoop_data/hdfs/namenode</value>
</property>
```

28. 設定 MasterNode 檔案

sudo gedit /usr/local/hadoop/etc/hadoop/master

MasterNode

29. 設定 slaves 檔案

sudo gedit /usr/local/hadoop/etc/hadoop/slaves

DataNode1
DataNode2
DataNode3

30. 重啟 MasterNode

reboot

31. MasterNode 連線至 DataNode1、DataNode2、DataNode3建立 HDFS 目錄
    (1) 啟動 MasterNode、DataNode1、DataNode2、DataNode3
    (2) MasterNode 以 SSH 連線至 DataNode1並建立 HDFS 目錄
        ssh DataNode1
        sudo rm -rf /usr/local/hadoop/hadoop_data/hdfs
        sudo mkdir -p /usr/local/hadoop/hadoop_data/hdfs/datanode
        sudo chown hduser:hduser -R /usr/local/hadoop
        exit
    (3) MasterNode 以 SSH 連線至 DataNode2並建立 HDFS 目錄
        ssh DataNode2
        sudo rm -rf /usr/local/hadoop/hadoop_data/hdfs
        sudo mkdir -p /usr/local/hadoop/hadoop_data/hdfs/datanode
        sudo chown hduser:hduser -R /usr/local/hadoop
        exit

    (4) MasterNode 以 SSH 連線至 DataNode3並建立 HDFS 目錄
        ssh DataNode3
        sudo rm -rf /usr/local/hadoop/hadoop_data/hdfs
        sudo mkdir -p /usr/local/hadoop/hadoop_data/hdfs/datanode
        sudo chown hduser:hduser -R /usr/local/hadoop
        exit

32. MasterNode 測試 DataNode1的遠端安全連線
    (1) 測試 DataNode1連線
        ping -c 4 DataNode1
    (2) 將公鑰授權檔拷貝到 DataNode1
        scp ~/.ssh/authorized_keys hduser@DataNode1:/home/hduser/.ssh
    (3) 測試 DataNode1是否可以免密碼登入
        ssh DataNode1
    (4) 回到 MasterNode
        exit

33. 建立與格式化 NameNode HDFS 目錄
    (1) 重新建立 NameNode HDFS 目錄
        sudo rm -rf /usr/local/hadoop/hadoop_data/hdfs
        mkdir -p /usr/local/hadoop/hadoop_data/hdfs/namenode
        sudo chown -R hduser:hduser /usr/local/hadoop
    (2) 格式化 NameNode HDFS 目錄
        hadoop namenode -format

34. 啟動 Hadoop
    (1) 啟動全部
        start-all.sh
    (2) 或啟動 start-dfs.sh，再啟動 start-yarn.sh
        start-dfs.sh
        start-yarn.sh

35. 查看目前所執行的行程
    jps

36. 開啟 Hadoop Resource-Manager Web 介面
    Hadoop ResourceManager Web 介面網址：http://MasterNode:8088/

37. 開啟 NameNode Web 介面
    開啟 HDFS Web UI 網址：http://MasterNode:50070/

壹、 執行 WordCount.java 範例程式

1. 在下列網址 Hadoop 說明文件中有 WordCount.java v1.0的程式碼：
   http://hadoop.apache.org/docs/current/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html

2. 編輯 WordCount.java
   (1) 建立 wordcount 目錄
   mkdir -p ~/wordcount/input
   cd ~/wordcount
   (2) 編輯 WordCount.java
   gedit WordCount.java

(3) 在 gedit 輸入 WordCount.java 完整程式碼

```
import java.io.IOException;
import java.util.StringTokenizer;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class WordCount {
  public static class TokenizerMapper extends Mapper<Object, Text, Text, IntWritable>{
    private final static IntWritable one = new IntWritable(1);
    private Text word = new Text();

    public void map(Object key, Text value, Context context ) throws IOException, InterruptedException {
      StringTokenizer itr = new StringTokenizer(value.toString());
      while (itr.hasMoreTokens()) {
        word.set(itr.nextToken());
        context.write(word, one);
      }
    }
  }

  public static class IntSumReducer extends Reducer<Text,IntWritable,Text,IntWritable> {
    private IntWritable result = new IntWritable();
    public void reduce(Text key, Iterable<IntWritable> values, Context context) throws IOException, InterruptedException {
      int sum = 0;
      for (IntWritable val : values) {
        sum += val.get();
      }
      result.set(sum);
      context.write(key, result);
    }
  }

  public static void main(String[] args) throws Exception {
    Configuration conf = new Configuration();
    Job job = Job.getInstance(conf, "word count");
    job.setJarByClass(WordCount.class);
    job.setMapperClass(TokenizerMapper.class);
    job.setCombinerClass(IntSumReducer.class);
    job.setReducerClass(IntSumReducer.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(IntWritable.class);
    FileInputFormat.addInputPath(job, new Path(args[0]));
    FileOutputFormat.setOutputPath(job, new Path(args[1]));
    System.exit(job.waitForCompletion(true) ? 0 : 1);
  }
}
```

3. 編譯 wordCount.java

(1) 編輯~/.bashrc

sudo gedit ~/.bashrc

(2) 輸入下列內容

export PATH=${JAVA_HOME}/bin:${PATH}

export HADOOP_CLASSPATH=${JAVA_HOME}/lib/tools.jar

(3) 讓 ~/.bashrc 修改的設定值生效

source ~/.bashrc

(4) 開始編譯

hadoop com.sun.tools.javac.Main WordCount.java

jar cf wc.jar WordCount*.class

ll

4. 建立測試文字檔
   cp /usr/local/hadoop/LICENSE.txt ~/wordcount/input
   ll ~/wordcount/input
   start-all.sh
   hadoop fs -mkdir -p /user/hduser/wordcount/input
   cd ~/wordcount/input
   hadoop fs -copyFromLocal LICENSE.txt /user/hduser/wordcount/input
   hadoop fs -ls /user/hduser/wordcount/input

5. 執行 wordCount.java
   cd ~/wordcount
   hadoop jar wc.jar WordCount /user/hduser/wordcount/input/LICENSE.txt /user/hduser/wordcount/output

6. 查看執行結果
   hadoop fs -ls /user/hduser/wordcount/output
   hadoop fs -cat /user/hduser/wordcount/output/part-r-00000

7. 刪除執行結果
   hadoop fs -rm -R /user/hduser/wordcount/output