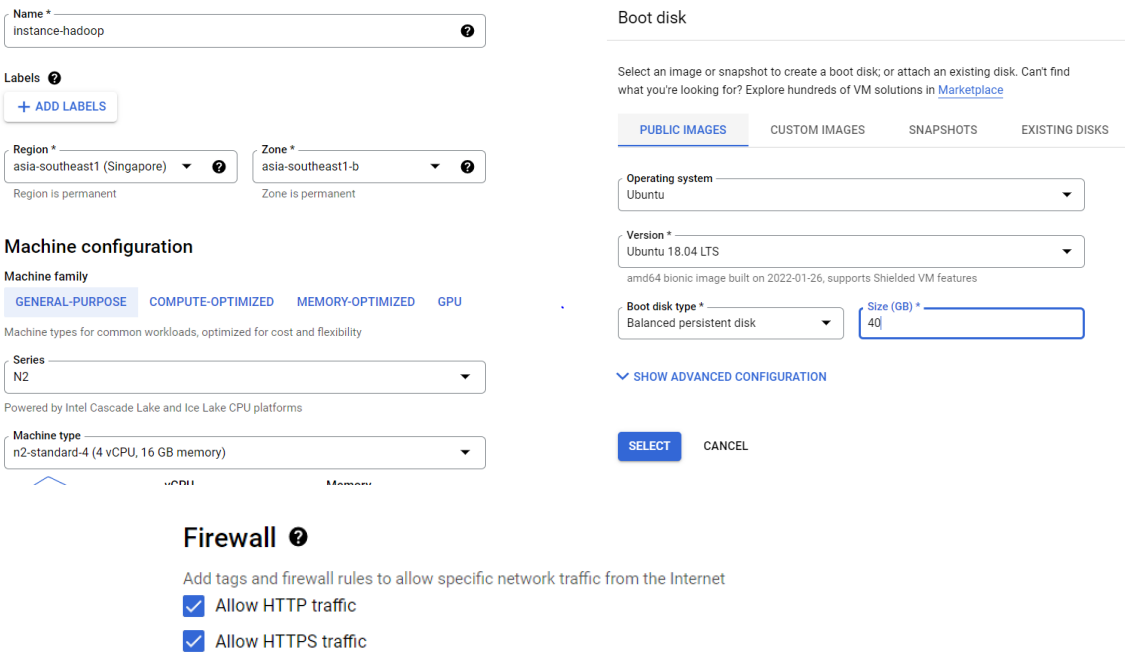
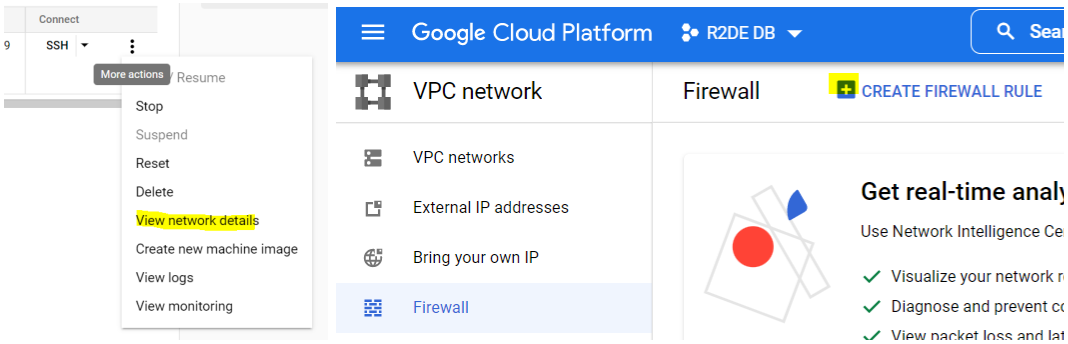
Special Live by Shane : Hadoop Docker Installation Guide

การ Set-up Google Compute Engine เพื่อทำโปรเจ็ค Hadoop

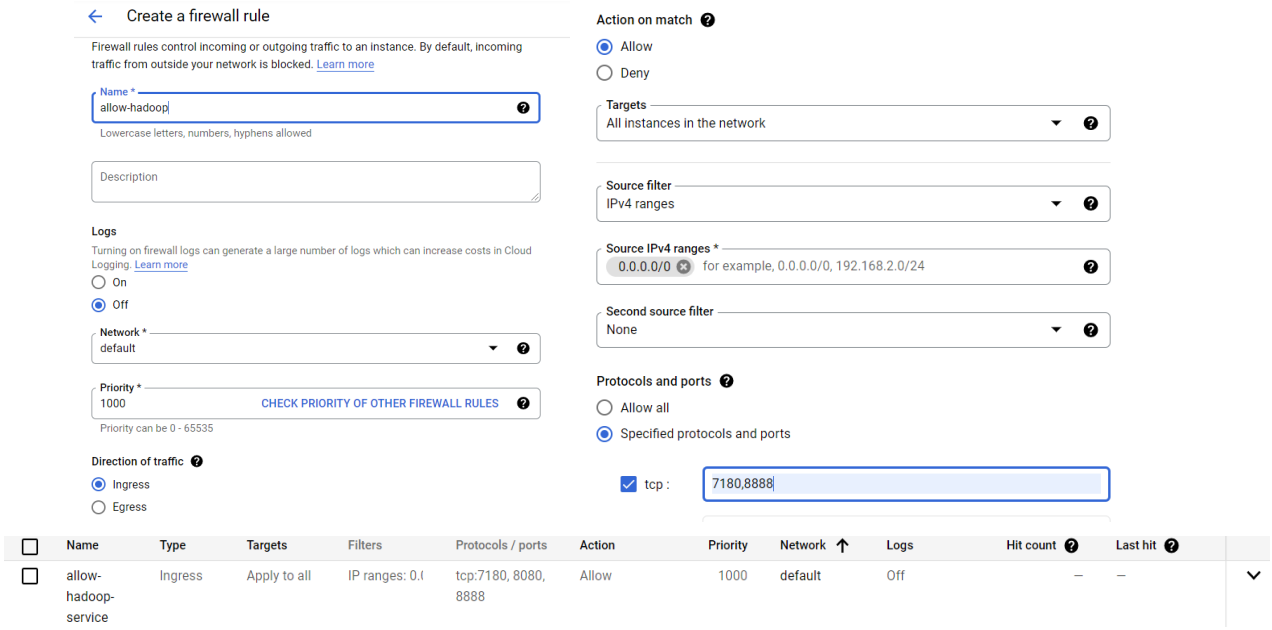
1) Provision GCP compute engine for Docker (configuration ที่นอกเหนือจากนี้ ให้ปล่อยเป็นค่า default)



2) หลังจาก provision VM เรียบร้อยแล้ว ให้ทำการ set-up Network Firewall ของ VM ตัวนั้น



3) Set-up Firewall Network ตาม configuration ข้างต้น
(**หมายเหตุ** การ set-up Firewall ข้างต้นเป็นการเปิด Firewall for all inbound เพื่อลดความซับซ้อน ของ configuration สำหรับผู้เริ่มต้นใช้งาน Networking ควรใช้ในกรณีศึกษาเท่านั้น ไม่ควรใช้ในงาน Production หรืองานที่มีความสำคัญทางข้อมูล อาจทำให้ผิด policy ด้าน security ได้)
(จุดประสงค์หลักเพื่อเป็นการเปิด tcp port : 7180 สำหรับ Cloudera Manager และ tcp port : 8888 สำหรับ Cloudera HUE ให้เข้าถึงผ่านหน้า browser ได้)



4) หลังจาก set-up เสร็จเรียบร้อย กด ปุ่ม SSH ที่หน้า Compute Engine ของตัวที่ create ขึ้นมา เพื่อเข้าสู่หน้า SSH terminal

การเริ่มต้นใช้งาน Cloudera Hadoop via Docker container

1) ติดตั้ง docker บน Ubuntu 18 ตาม **Step 1-4 จาก guide ของ Digital Ocean** https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-on-ubuntu-18-04
2) หลังจากติดตั้งเสร็จเรียบร้อย ให้ ใช้คำสั่ง docker pull mikelemikelo/cloudera-spark:latest เพื่อ pull docker image มาใช้ที่ VM ของคุณ
3) หลังจาก pull image มาสำเร็จเรียบร้อยแล้ว สามารถเช็คได้ด้วยการใช้คำสั่ง docker images
4) ใช้คำสั่ง docker run --hostname=quickstart.cloudera --privileged=true -it -p 8888:8888 -p 8080:8080 -p 7180:7180 -p 88:88/udp -p 88:88 mikelemikelo/cloudera-spark:latest /usr/bin/docker-quickstart-light

```
nuttatun_c@hadoop-sandbox:~$ docker run --hostname=quickstart.cloudera --privileged=true -i
t -p 8888:8888 -p 8080:8080 -p 7180:7180 -p 88:88/udp -p 88:88 mikelemikelo/cloudera-spark:
latest /usr/bin/docker-quickstart-light
```

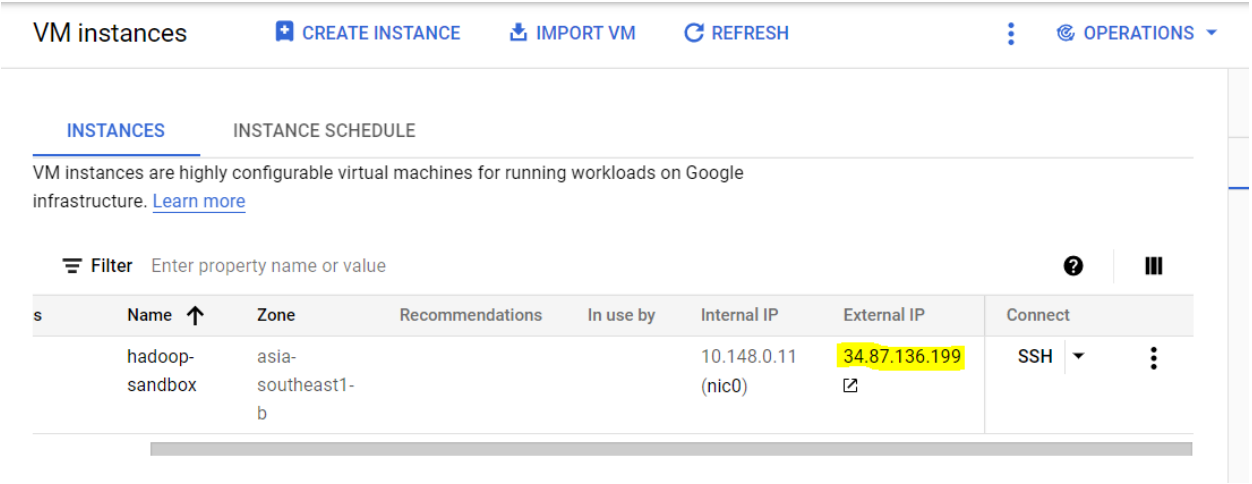
5) รอจนกว่า container จะรัน Hadoop fundamental commands จนเสร็จสิ้น

```
Using   CATALINA_OPTS:        -Doozie.https.port=11443 -Doozie.https.keystore.pass=password
 -Xmx1024m -Doozie.https.port=11443 -Doozie.https.keystore.pass=password -Xmx1024m -Dderby.
stream.error.file=/var/log/oozie/derby.log
Adding to CATALINA_OPTS:     -Doozie.home.dir=/usr/lib/oozie -Doozie.config.dir=/etc/oozie/
conf -Doozie.log.dir=/var/log/oozie -Doozie.data.dir=/var/lib/oozie -Doozie.instance.id=qui
ckstart.cloudera -Doozie.config.file=oozie-site.xml -Doozie.log4j.file=oozie-log4j.properti
es -Doozie.log4j.reload=10 -Doozie.http.hostname=quickstart.cloudera -Doozie.admin.port=110
01 -Doozie.http.port=11000 -Doozie.https.port=11443 -Doozie.base.url=http://quickstart.clou
dera:11000/oozie -Doozie.https.keystore.file=/var/lib/oozie/.keystore -Doozie.https.keystor
e.pass=password -Djava.library.path=:/usr/lib/hadoop/lib/native:/usr/lib/hadoop/lib/native

Using CATALINA_BASE:   /var/lib/oozie/tomcat-deployment
Using CATALINA_HOME:   /usr/lib/bigtop-tomcat
Using CATALINA_TMPDIR: /var/lib/oozie
Using JRE_HOME:        /usr/lib/jvm/jre-1.8.0-openjdk.x86_64
Using CLASSPATH:       /usr/lib/bigtop-tomcat/bin/bootstrap.jar
Using CATALINA_PID:    /var/run/oozie/oozie.pid
[root@quickstart cloudera]#
[root@quickstart cloudera]# sudo /home/cloudera/cloudera-manager --express && service ntpd
start
[QuickStart] Shutting down CDH services via init scripts...
kafka-server: unrecognized service
```

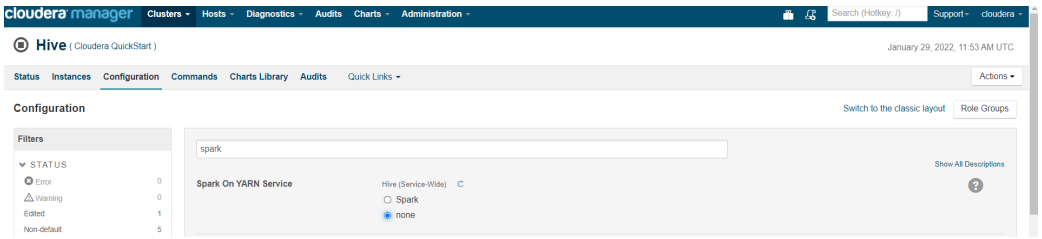
6) ใช้คำสั่ง sudo /home/cloudera/cloudera-manager --express && service ntpd start เพื่อเริ่มต้นการใช้งาน Cloudera Manager

```
[QuickStart] Configuring deployment...
Submitted jobs: 14
[QuickStart] Deploying client configuration...
Submitted jobs: 16
[QuickStart] Starting Cloudera Management Service...
Submitted jobs: 24
 [QuickStart] Enabling Cloudera Manager daemons on boot...
________________________________________________________________________________

Success! You can now log into Cloudera Manager from the QuickStart VM's browser:

    http://quickstart.cloudera:7180

    Username: cloudera
    Password: cloudera

Starting ntpd:                                             [  OK  ]
[root@quickstart cloudera]#
```

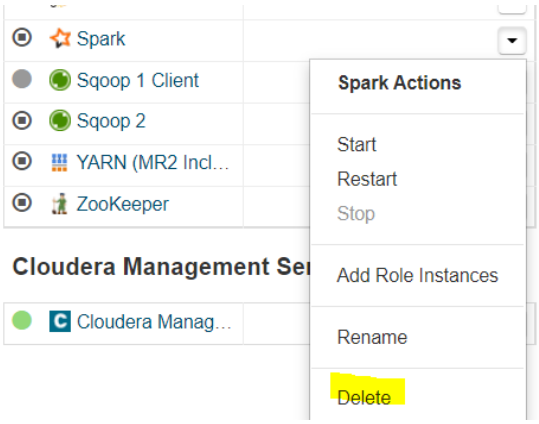
7) ใช้ Public / External IP ของ VM ในการเข้า Cloudera Manager ด้วย port 7180 ยกตัวอย่างเช่น http://34.87.136.199:7180/ -> user:cloudera, pw:cloudera

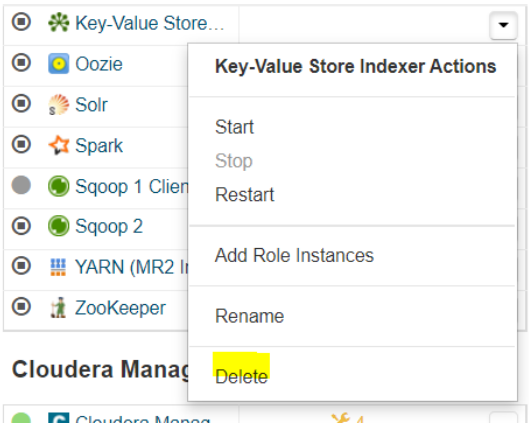


8) ก่อนจะเริ่มต้น start cluster ให้ไปที่เซอร์วิส Hive ทั้งด้าน ซ้ายมือ set ค่า ของ configuration ข้างต้น เป็น none

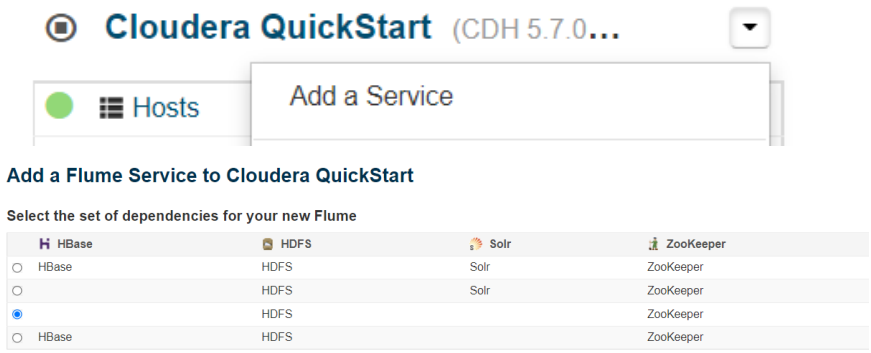


9) ลบ Service ที่ไม่ได้ใช้งานและไม่จำเป็นออก เช่น Spark และ KV Indexer
(Spark ของ CDH นี้เป็น version 1.x.x ที่ไม่สามารถใช้งาน spark structured streaming ได้ ให้ลบ service ตรงนี้ แล้วไปใช้ Spark Local ที่ติดตั้งไว้แล้วใน docker container แทน)

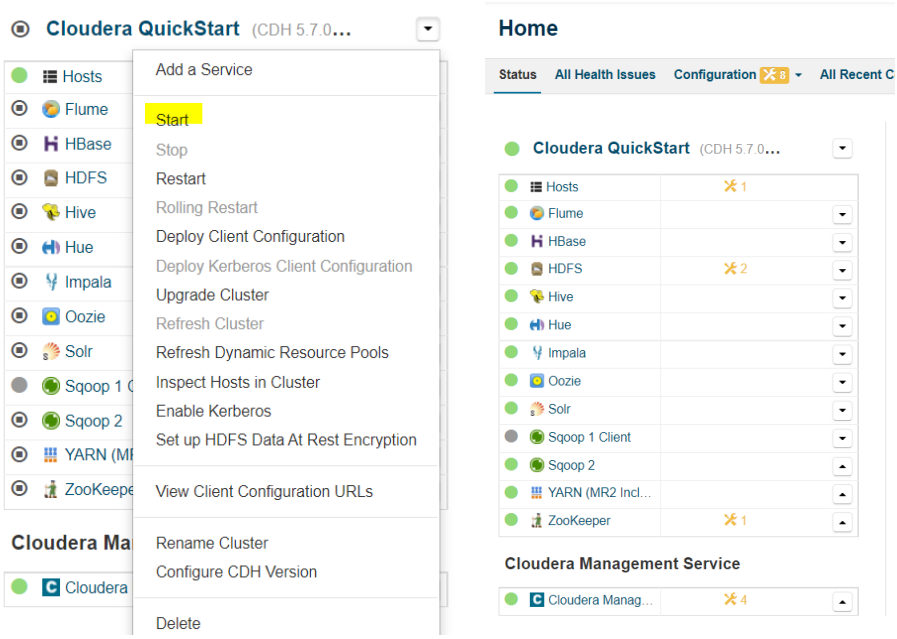




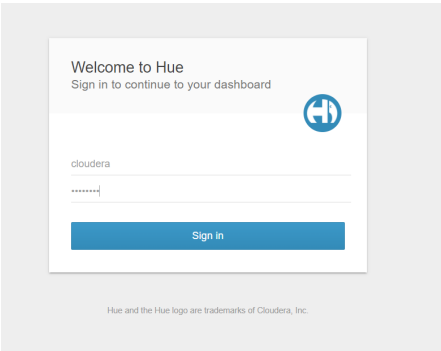
10) ทำการ Add service “Flume”



11) Start cluster และ รอจนกว่า service จะเข้า status all-up-and-running



12) ทำการเข้าสู่ระบบด้วย Cloudera HUE โดยใช้ Public / External IP ของ VM ด้วย port 8888 ยกตัวอย่างเช่น http://34.87.136.199:8888/ -> user:cloudera, pw:cloudera



การ Clone Git Repository เพื่อใช้ Materials จากตัวอย่างใน Special Live

1) กด Ctrl + P + Q เพื่อออกจาก docker container มาที่ VM terminal หลัก
2) ใช้คำสั่ง git clone https://github.com/nuttatunc/hadoop-r2de-data.git เพื่อ clone repository จาก special live
3) ใช้คำสั่ง docker container ls เพื่อ ดู container_id ของ container ที่รัน Hadoop Service อยู่
4) ใช้คำสั่ง docker cp hadoop-r2de-data/data <container_id>:/ เพื่อนำ ไฟล์ที่ได้จาก git repo ก็อปปี้เข้าไปใน container
5) ใช้คำสั่ง docker exec -it <container_id> /bin/bash เพื่อกลับไปเข้าใช้งาน container ตามปกติ โดยจะมี folder ที่ชื่อว่า data ที่มี code และ ส่วนประกอบอื่นๆเหมือนภายใน Special Live