Name: Xinrun Zhang

Date: 11/21/2018

1. Software installation

   A. MySQL

      Command: sudo apt-get install mysql-server mysql-common mysql-client

```
zhangxinrun@ubuntu:~$ sudo apt-get install mysql-server mysql-common mysql-clie
nt
[sudo] password for zhangxinrun:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libaio1 libevent-core-2.1-6 libhtml-template-perl mysql-client-5.7
  mysql-client-core-5.7 mysql-server-5.7 mysql-server-core-5.7
```

   B. Flume

      a. Download Apache Flume

```
zhangxinrun@ubuntu:~$ cd  Downloads
zhangxinrun@ubuntu:~/Downloads$ ls
apache-flume-1.8.0-bin.tar.gz  hadoop-2.9.1.tar.gz
```

      b. Extract the file

```
zhangxinrun@ubuntu:~/Downloads$ tar xvfz apache-flume-1.8.0-bin.tar.gz
apache-flume-1.8.0-bin/lib/flume-ng-configuration-1.8.0.jar
apache-flume-1.8.0-bin/lib/slf4j-api-1.6.1.jar
apache-flume-1.8.0-bin/lib/slf4j-log4j12-1.6.1.jar
apache-flume-1.8.0-bin/lib/log4j-1.2.17.jar
```

      c. Move the folder to /usr/local/flume

```
zhangxinrun@ubuntu:~/Downloads$ sudo mv apache-flume-1.8.0-bin /usr/local/flume
```

      d. Change the ownership

```
zhangxinrun@ubuntu:~$ sudo chown -R zhangxinrun:zhangxinrun /usr/local/flume
```

      e. Update bashrc file

```
export FLUME_HOME=/usr/local/flume
export PATH=$PATH:$FLUME_HOME/bin/
```

```
zhangxinrun@ubuntu:~$ gedit ~/.bashrc
zhangxinrun@ubuntu:~$ source .bashrc
```

      f. Start Hadoop service

```
zhangxinrun@ubuntu:~$ flume-ng help
Usage: /usr/local/flume/bin/flume-ng <command> [options]...

commands:
  help                      display this help text
  agent                     run a Flume agent
  avro-client               run an avro Flume client
  version                   show Flume version info

global options:
  --conf,-c <conf>          use configs in <conf> directory
  --classpath,-C <cp>       append to the classpath
  --dryrun,-d               do not actually start Flume, just print the command
  --plugins-path <dirs>     colon-separated list of plugins.d directories. See
the
```

C. Sqoop

a. Download Sqoop

```
zhangxinrun@ubuntu:~/Downloads$ ls
apache-flume-1.8.0-bin.tar.gz  sqoop-1.4.7.bin__hadoop-2.6.0.tar.gz
hadoop-2.9.1.tar.gz
```

b. Extract the file

```
zhangxinrun@ubuntu:~/Downloads$ tar xvfz sqoop-1.4.7.bin__hadoop-2.6.0.tar.gz
sqoop-1.4.7.bin__hadoop-2.6.0/
sqoop-1.4.7.bin__hadoop-2.6.0/CHANGELOG.txt
sqoop-1.4.7.bin__hadoop-2.6.0/COMPILING.txt
sqoop-1.4.7.bin__hadoop-2.6.0/LICENSE.txt
sqoop-1.4.7.bin__hadoop-2.6.0/NOTICE.txt
sqoop-1.4.7.bin__hadoop-2.6.0/README.txt
```

c. Move the folder to /usr/local/sqoop

```
zhangxinrun@ubuntu:~/Downloads$ sudo mv sqoop-1.4.7.bin__hadoop-2.6.0 /usr/loca
l/sqoop
[sudo] password for zhangxinrun:
```

d. Change the ownership

```
zhangxinrun@ubuntu:~$ sudo chown -R zhangxinrun:zhangxinrun /usr/local/sqoop
```

e. Update bashrc file

```
export SQOOP_HOME=/usr/local/sqoop
export PATH=$PATH:$SQOOP_HOME/bin/
```

```
zhangxinrun@ubuntu:~$ gedit ~/.bashrc
zhangxinrun@ubuntu:~$ source .bashrc
```

f. Download mysql connector

```
zhangxinrun@ubuntu:~/Downloads$ ls
mysql-connector-java-5.1.47.tar.gz
```

g. Extract it

```
zhangxinrun@ubuntu:~/Downloads$ tar xvfz mysql-connector-java-5.1.47.tar.gz
mysql-connector-java-5.1.47/
mysql-connector-java-5.1.47/src/
mysql-connector-java-5.1.47/src/com/
mysql-connector-java-5.1.47/src/com/mysql/
mysql-connector-java-5.1.47/src/com/mysql/fabric/
mysql-connector-java-5.1.47/src/com/mysql/fabric/hibernate/
mysql-connector-java-5.1.47/src/com/mysql/fabric/jdbc/
```

h. Copy mysql-connector-java-5.1.47-bin.jar file from the extracted folder and paste it into /usr/local/sqoop/lib folder.

```
zhangxinrun@ubuntu:~/Downloads$ cd mysql-connector-java-5.1.47
zhangxinrun@ubuntu:~/Downloads/mysql-connector-java-5.1.47$ ls
build.xml  mysql-connector-java-5.1.47-bin.jar  README.txt
CHANGES    mysql-connector-java-5.1.47.jar      src
COPYING    README
zhangxinrun@ubuntu:~/Downloads/mysql-connector-java-5.1.47$ sudo cp mysql-connect
or-java-5.1.47-bin.jar /usr/local/sqoop/lib
zhangxinrun@ubuntu:~/Downloads/mysql-connector-java-5.1.47$ cd usr/local/sqoop/li
b
bash: cd: usr/local/sqoop/lib: No such file or directory
zhangxinrun@ubuntu:~/Downloads/mysql-connector-java-5.1.47$ cd /usr/local/sqoop/l
ib
zhangxinrun@ubuntu:/usr/local/sqoop/lib$ ls
ant-contrib-1.0b3.jar           kite-data-hive-1.1.0.jar
ant-eclipse-1.0-jvm1.2.jar      kite-data-mapreduce-1.1.0.jar
avro-1.8.1.jar                  kite-hadoop-compatibility-1.1.0.jar
avro-mapred-1.8.1-hadoop2.jar   mysql-connector-java-5.1.47-bin.jar
commons-codec-1.4.jar           opencsv-2.3.jar
commons-compress-1.8.1.jar      paranamer-2.7.jar
commons-io-1.4.jar              parquet-avro-1.6.0.jar
commons-jexl-2.1.1.jar          parquet-column-1.6.0.jar
commons-lang3-3.4.jar           parquet-common-1.6.0.jar
commons-logging-1.1.1.jar       parquet-encoding-1.6.0.jar
hsqldb-1.8.0.10.jar             parquet-format-2.2.0-rc1.jar
jackson-annotations-2.3.1.jar   parquet-generator-1.6.0.jar
jackson-core-2.3.1.jar          parquet-hadoop-1.6.0.jar
jackson-core-asl-1.9.13.jar     parquet-jackson-1.6.0.jar
jackson-databind-2.3.1.jar      slf4j-api-1.6.1.jar
jackson-mapper-asl-1.9.13.jar   snappy-java-1.1.1.6.jar
kite-data-core-1.1.0.jar        xz-1.5.jar
```

i. Execute sqoop help

```
zhangxinrun@ubuntu:~$ sqoop help
Warning: /usr/local/sqoop/../hbase does not exist! HBase imports will fail.
Please set $HBASE_HOME to the root of your HBase installation.
Warning: /usr/local/sqoop/../hcatalog does not exist! HCatalog jobs will fail.
Please set $HCAT_HOME to the root of your HCatalog installation.
Warning: /usr/local/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
Warning: /usr/local/sqoop/../zookeeper does not exist! Accumulo imports will fail
.
Please set $ZOOKEEPER_HOME to the root of your Zookeeper installation.
18/11/18 14:10:00 INFO sqoop.Sqoop: Running Sqoop version: 1.4.7
usage: sqoop COMMAND [ARGS]

Available commands:
  codegen            Generate code to interact with database records
  create-hive-table  Import a table definition into Hive
  eval               Evaluate a SQL statement and display the results
  export             Export an HDFS directory to a database table
  help               List available commands
  import             Import a table from a database to HDFS
  import-all-tables  Import tables from a database to HDFS
  import-mainframe   Import datasets from a mainframe server to HDFS
  job                Work with saved jobs
  list-databases     List available databases on a server
  list-tables        List available tables in a database
  merge              Merge results of incremental imports
  metastore          Run a standalone Sqoop metastore
  version            Display version information

See 'sqoop help COMMAND' for information on a specific command.
```

D. Hive

a. Download Hive

```
zhangxinrun@ubuntu:~/Downloads$ ls
apache-hive-2.3.4-bin.tar.gz
```

b. Extract the setup file

```
zhangxinrun@ubuntu:~/Downloads$ tar xvfz sqoop-1.4.7.bin__hadoop-2.6.0.tar.gz
sqoop-1.4.7.bin__hadoop-2.6.0/
sqoop-1.4.7.bin__hadoop-2.6.0/CHANGELOG.txt
sqoop-1.4.7.bin__hadoop-2.6.0/COMPILING.txt
sqoop-1.4.7.bin__hadoop-2.6.0/LICENSE.txt
sqoop-1.4.7.bin__hadoop-2.6.0/NOTICE.txt
sqoop-1.4.7.bin__hadoop-2.6.0/README.txt
```

c. Move the folder

```
zhangxinrun@ubuntu:~/Downloads$ sudo mv apache-hive-2.3.4-bin /usr/local/hive
[sudo] password for zhangxinrun:
```

d. Change the ownership

```
zhangxinrun@ubuntu:~$ sudo chown -R zhangxinrun:zhangxinrun /usr/local/hive
```

e. Update bashrc file

```
export HIVE_HOME=/usr/local/hive
export PATH=$PATH:$HIVE_HOME/bin/
export HIVE_CONF_DIR=$HIVE_HOME/conf
```

```
zhangxinrun@ubuntu:~$ gedit ~/.bashrc
zhangxinrun@ubuntu:~$ source .bashrc
```

f. Open hive-env.sh file and add HADOOP_HOME=/usr/local/hadoop in the file

```
zhangxinrun@ubuntu:~$ cd /usr/local/hive/conf
zhangxinrun@ubuntu:/usr/local/hive/conf$ ls
beeline-log4j2.properties.template    ivysettings.xml
hive-default.xml.template             llap-cli-log4j2.properties.template
hive-env.sh.template                  llap-daemon-log4j2.properties.template
hive-exec-log4j2.properties.template  parquet-logging.properties
hive-log4j2.properties.template
zhangxinrun@ubuntu:/usr/local/hive/conf$ gedit /usr/local/hive/conf/hive-env.sh.t
emplate
```

```
# Set HADOOP_HOME to point to a specific hadoop install directory
HADOOP_HOME=/usr/local/hadoop
```

```
zhangxinrun@ubuntu:/usr/local/hive/conf$ mv hive-env.sh.template hive-env.sh
zhangxinrun@ubuntu:/usr/local/hive/conf$ ls
beeline-log4j2.properties.template    ivysettings.xml
hive-default.xml.template             llap-cli-log4j2.properties.template
hive-env.sh                           llap-daemon-log4j2.properties.template
hive-exec-log4j2.properties.template  parquet-logging.properties
hive-log4j2.properties.template
```

g. Create hive-site.xml inside /usr/local/hive/conf folder and edit it

Firstly, see what we have in the folder

```
zhangxinrun@ubuntu:~$ cd /usr/local/hive/conf
zhangxinrun@ubuntu:/usr/local/hive/conf$ ls
beeline-log4j2.properties.template    ivysettings.xml
hive-default.xml.template             llap-cli-log4j2.properties.template
hive-env.sh                           llap-daemon-log4j2.properties.template
hive-exec-log4j2.properties.template  parquet-logging.properties
hive-log4j2.properties.template
```

create hive-site.sml

```
zhangxinrun@ubuntu:/usr/local/hive/conf$ touch hive-site.xml
zhangxinrun@ubuntu:/usr/local/hive/conf$ ls
beeline-log4j2.properties.template    hive-site.xml
hive-default.xml.template             ivysettings.xml
hive-env.sh                           llap-cli-log4j2.properties.template
hive-exec-log4j2.properties.template  llap-daemon-log4j2.properties.template
hive-log4j2.properties.template       parquet-logging.properties
zhangxinrun@ubuntu:/usr/local/hive/conf$ gedit hive-site.xml
```

edit hive-site.xml

```
Open ▾   *hive-site.xml   Save
/usr/local/hive/conf
<configuration>
<property>
        <name>system:java.io.tmpdir</name>
        <value>/tmp/hive/java</value>
</property>
<property>
        <name>system:user.name</name>
        <value>${user.name}</value>
</property>
<property>
        <name>javax.jdo.option.ConnectionURL</name>
        <value>jdbc:derby:;databaseName=metastore_db;create=true</value>
</property>
</configuration>
```

h. Copy hive-default.xml.template to hive-default.xml inside hive/conf folder

```
zhangxinrun@ubuntu:/usr/local/hive/conf$ mv hive-default.xml.template hive-defaul
t.xml
zhangxinrun@ubuntu:/usr/local/hive/conf$ ls
beeline-log4j2.properties.template    hive-site.xml
hive-default.xml                      ivysettings.xml
hive-env.sh                           llap-cli-log4j2.properties.template
hive-exec-log4j2.properties.template  llap-daemon-log4j2.properties.template
hive-log4j2.properties.template       parquet-logging.properties
```

i. Execute schematool –initSchema –dbType derby

```
zhangxinrun@ubuntu:~$ schematool -initSchema -dbType derby
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/local/hive/lib/log4j-slf4j-impl-2.6.2.jar!
/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hadoop/share/hadoop/common/lib/slf4j
-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.u
til.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.9.
1.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.s
ecurity.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflecti
ve access operations
WARNING: All illegal access operations will be denied in a future release
Metastore connection URL:        jdbc:derby:;databaseName=metastore_db;create=tru
e
Metastore Connection Driver :    org.apache.derby.jdbc.EmbeddedDriver
Metastore connection User:       APP
Starting metastore schema initialization to 2.3.0
Initialization script hive-schema-2.3.0.derby.sql
Initialization script completed
schemaTool completed
```

j. Start Hadoop service and execute hive command

```
zhangxinrun@ubuntu:~$ hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/local/hive/lib/log4j-slf4j-impl-2.6.2.jar!/org/slf4j/impl/StaticL
oggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.25.jar!/or
g/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]

Logging initialized using configuration in jar:file:/usr/local/hive/lib/hive-common-2.3.4.jar!/hive-log4
j2.properties Async: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a dif
ferent execution engine (i.e. spark, tez) or using Hive 1.X releases.
hive>
    > quit
    > ?
    > quit;
```

2. Practice

A. Flume

1. Create /tmp/flume directory on local file system

```
zhangxinrun@ubuntu:~$ mkdir /tmp/flume
```

```
zhangxinrun@ubuntu:~$ cp /home/zhangxinrun/handson-3/flume.txt /tmp/flume
zhangxinrun@ubuntu:~$ cp /home/zhangxinrun/handson-3/sqoop.txt /tmp/flume
```

2. Create /user/flume directory on HDFS cluster

```
zhangxinrun@ubuntu:~$ hdfs dfs -mkdir /user/flume
```

3. Create agent file and save it in /usr/local/flume/conf

```
zhangxinrun@ubuntu:/usr/local/flume/conf$ touch flume-conf.properties
zhangxinrun@ubuntu:/usr/local/flume/conf$ ls
flume-conf.properties           flume-env.ps1.template  log4j.properties
flume-conf.properties.template  flume-env.sh.template
```

```
zhangxinrun@ubuntu:/usr/local/flume/conf$ gedit flume-conf.properties
```

Open ▾ *flume-conf.properties /usr/local/flume/conf Save

```
agent1.sources = source1
agent1.sinks = sink1
agent1.channels = channel1

agent1.sources.source1.channels = channel1
agent1.sources.source1.type = spooldir
agent1.sources.source1.spoolDir = /tmp/flume

agent1.sinks.sink1.channel = channel1
agent1.sinks.sink1.type = hdfs
agent1.sinks.sink1.hdfs.path = /user/flume
agent1.sinks.sink1.hdfs.filePrefix = events
agent1.sinks.sink1.hdfs.fileSuffix = .log
agent1.sinks.sink1.hdfs.inUsePrefix = _
agent1.sinks.sink1.hdfs.fileType = DataStream

agent1.channels.channel1.type = file
```

4. Run the flume agent

```
zhangxinrun@ubuntu:/usr/local/flume/conf$ flume-ng agent -n agent1 -c conf -f f
lume-conf.properties
Info: Including Hadoop libraries found via (/usr/local/hadoop/bin/hadoop) for H
DFS access
Info: Including Hive libraries found via (/usr/local/hive) for Hive access
+ exec /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java -Xmx20m -cp 'conf:/usr/lo
cal/flume/lib/*:/usr/local/hadoop/etc/hadoop:/usr/local/hadoop/share/hadoop/com
mon/lib/*:/usr/local/hadoop/share/hadoop/common/*:/usr/local/hadoop/share/hadoo
p/hdfs:/usr/local/hadoop/share/hadoop/hdfs/lib/*:/usr/local/hadoop/share/hadoop
/hdfs/*:/usr/local/hadoop/share/hadoop/yarn:/usr/local/hadoop/share/hadoop/yarn
/lib/*:/usr/local/hadoop/share/hadoop/yarn/*:/usr/local/hadoop/share/hadoop/map
reduce/lib/*:/usr/local/hadoop/share/hadoop/mapreduce/*:/usr/local/hadoop/contr
ib/capacity-scheduler/*.jar:/usr/local/hive/lib/*' -Djava.library.path=:/usr/lo
cal/hadoop/lib/native org.apache.flume.node.Application -n agent1 -f flume-conf
.properties
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/local/flume/lib/slf4j-log4j12-1.6.1.jar!
```

```
18/11/21 15:14:39 INFO file.EventQueueBackingStoreFile: Start checkpoint for /h
ome/zhangxinrun/.flume/file-channel/checkpoint/checkpoint, elements to sync = 7
18/11/21 15:14:39 INFO file.EventQueueBackingStoreFile: Updating checkpoint met
adata: logWriteOrderID: 1542842049079, queueSize: 0, queueHead: 5
18/11/21 15:14:39 INFO file.Log: Updated checkpoint for file: /home/zhangxinrun
/.flume/file-channel/data/log-2 position: 323 logWriteOrderID: 1542842049079
18/11/21 15:14:41 INFO hdfs.BucketWriter: Closing /user/flume/_events.154284204
9528.log.tmp
18/11/21 15:14:41 INFO hdfs.BucketWriter: Renaming /user/flume/_events.15428420
49528.log.tmp to /user/flume/events.1542842049528.log
18/11/21 15:14:42 INFO hdfs.HDFSEventSink: Writer callback called.
```

check the HDFS cluster and found a log file:

```
zhangxinrun@ubuntu:/usr/local/flume/conf$ cd
zhangxinrun@ubuntu:~$ hdfs dfs -ls /user/flume
Found 1 items
-rw-r--r--   1 zhangxinrun supergroup        920 2018-11-21 15:14 /user/flume/e
vents.1542842049528.log
```

cat the log file and it showed the content of the two txt files:

```
zhangxinrun@ubuntu:~$ hdfs dfs -cat /user/flume/events.1542842049528.log
Flume is a distributed, reliable, and available service for efficiently collect
ing, aggregating, and moving large amounts of log data. It has a simple and fle
xible architecture based on streaming data flows. It is robust and fault tolera
nt with tunable reliability mechanisms and many failover and recovery mechanism
s. It uses a simple extensible data model that allows for online analytic appli
cation.

Apache Sqoop(TM) is a tool designed for efficiently transferring bulk data betw
een Apache Hadoop and structured datastores such as relational databases.

Sqoop successfully graduated from the Incubator in March of 2012 and is now a T
op-Level Apache project: More information

Latest stable release is 1.4.7 (download, documentation). Latest cut of Sqoop2
is 1.99.7 (download, documentation). Note that 1.99.7 is not compatible with 1.
4.7 and not feature complete, it is not intended for production deployment.
```

I think the flume works well but it seems I didn’t copy two files into HDFS cluster, I just put them into a log file on HDFS. So, I modified some of the configuration files and tried it again.

```
zhangxinrun@ubuntu:~$ cd /tmp/flume
zhangxinrun@ubuntu:/tmp/flume$ ls
flume.txt.COMPLETED  sqoop.txt.COMPLETED
zhangxinrun@ubuntu:/tmp/flume$ rm flume.txt.COMPLETED
zhangxinrun@ubuntu:/tmp/flume$ rm sqoop.txt.COMPLETED
zhangxinrun@ubuntu:/tmp/flume$ ls
zhangxinrun@ubuntu:/tmp/flume$ cd
zhangxinrun@ubuntu:~$ cd /usr/local/flume/conf
zhangxinrun@ubuntu:/usr/local/flume/conf$ ls
flume-conf.properties           flume-env.ps1.template  log4j.properties
flume-conf.properties.template  flume-env.sh.template
zhangxinrun@ubuntu:/usr/local/flume/conf$ gedit flume-conf.properties
zhangxinrun@ubuntu:/usr/local/flume/conf$ cd
zhangxinrun@ubuntu:~$ hdfs dfs -rm /user/flume/events.1542842049528.log
Deleted /user/flume/events.1542842049528.log
```

Open ▾ flume-conf.properties /usr/local/flume/conf Save

```
agent1.sources = source1
agent1.sinks = sink1
agent1.channels = channel1

agent1.sources.source1.channels = channel1
agent1.sources.source1.type = spooldir
agent1.sources.source1.spoolDir = /tmp/flume

agent1.sinks.sink1.channel = channel1
agent1.sinks.sink1.type = hdfs
agent1.sinks.sink1.hdfs.path = /user/flume
agent1.sinks.sink1.hdfs.fileType = DataStream

agent1.channels.channel1.type = file
```

start the agent1 again

zhangxinrun@ubuntu: /usr/local/flume/conf × | zhangxinrun@ubuntu: ~ ×

```
data
18/11/21 15:35:44 INFO file.Log: Roll start /home/zhangxinrun/.flume/file-chann
el/data
18/11/21 15:35:44 INFO file.LogFile: Opened /home/zhangxinrun/.flume/file-chann
el/data/log-3
18/11/21 15:35:44 INFO file.Log: Roll end
18/11/21 15:35:44 INFO file.EventQueueBackingStoreFile: Start checkpoint for /h
ome/zhangxinrun/.flume/file-channel/checkpoint/checkpoint, elements to sync = 0
18/11/21 15:35:44 INFO file.EventQueueBackingStoreFile: Updating checkpoint met
adata: logWriteOrderID: 1542843343797, queueSize: 0, queueHead: 5
18/11/21 15:35:44 INFO file.Log: Updated checkpoint for file: /home/zhangxinrun
/.flume/file-channel/data/log-3 position: 0 logWriteOrderID: 1542843343797
18/11/21 15:35:44 INFO file.FileChannel: Queue Size after replay: 0 [channel=ch
annel1]
18/11/21 15:35:44 INFO node.Application: Starting Sink sink1
18/11/21 15:35:44 INFO node.Application: Starting Source source1
18/11/21 15:35:44 INFO source.SpoolDirectorySource: SpoolDirectorySource source
 starting with directory: /tmp/flume
18/11/21 15:35:44 INFO instrumentation.MonitoredCounterGroup: Monitored counter
 group for type: SINK, name: sink1: Successfully registered new MBean.
18/11/21 15:35:44 INFO instrumentation.MonitoredCounterGroup: Component type: S
INK, name: sink1 started
18/11/21 15:35:44 INFO instrumentation.MonitoredCounterGroup: Monitored counter
 group for type: SOURCE, name: source1: Successfully registered new MBean.
18/11/21 15:35:44 INFO instrumentation.MonitoredCounterGroup: Component type: S
OURCE, name: source1 started
```

and make a new bash window, copy the files into /tmp/flume

```
zhangxinrun@ubuntu: /usr/local/flume/conf    zhangxinrun@ubuntu: ~
zhangxinrun@ubuntu:~$ cp /home/zhangxinrun/handson-3/flume.txt /tmp/flume
zhangxinrun@ubuntu:~$
```

as I copied it into the directory, the agent1 started to work:

```
18/11/21 15:38:15 INFO avro.ReliableSpoolingFileEventReader: Last read took us
just up to a file boundary. Rolling to the next file, if there is one.
18/11/21 15:38:15 INFO avro.ReliableSpoolingFileEventReader: Preparing to move
file /tmp/flume/flume.txt to /tmp/flume/flume.txt.COMPLETED
18/11/21 15:38:19 INFO hdfs.HDFSDataStream: Serializer = TEXT, UseRawLocalFileS
ystem = false
18/11/21 15:38:19 INFO hdfs.BucketWriter: Creating /user/flume/FlumeData.154284
3499205.tmp
18/11/21 15:38:43 INFO file.EventQueueBackingStoreFile: Start checkpoint for /h
ome/zhangxinrun/.flume/file-channel/checkpoint/checkpoint, elements to sync = 2
18/11/21 15:38:43 INFO file.EventQueueBackingStoreFile: Updating checkpoint met
adata: logWriteOrderID: 1542843343804, queueSize: 0, queueHead: 5
18/11/21 15:38:43 INFO file.Log: Updated checkpoint for file: /home/zhangxinrun
/.flume/file-channel/data/log-3 position: 649 logWriteOrderID: 1542843343804
18/11/21 15:38:43 INFO file.LogFile: Closing RandomReader /home/zhangxinrun/.fl
ume/file-channel/data/log-1
18/11/21 15:38:51 INFO hdfs.BucketWriter: Closing /user/flume/FlumeData.1542843
499205.tmp
18/11/21 15:38:51 INFO hdfs.BucketWriter: Renaming /user/flume/FlumeData.154284
3499205.tmp to /user/flume/FlumeData.1542843499205
18/11/21 15:38:51 INFO hdfs.HDFSEventSink: Writer callback called.
```

then I copied sqoop.txt into the /tmp/flume:

```
zhangxinrun@ubuntu: /usr/local/flume/conf    zhangxinrun@ubuntu: ~
zhangxinrun@ubuntu:~$ cp /home/zhangxinrun/handson-3/flume.txt /tmp/flume
zhangxinrun@ubuntu:~$ cp /home/zhangxinrun/handson-3/sqoop.txt /tmp/flume
zhangxinrun@ubuntu:~$
```

and agent1 was working:

```
18/11/21 15:40:52 INFO avro.ReliableSpoolingFileEventReader: Last read took us
just up to a file boundary. Rolling to the next file, if there is one.
18/11/21 15:40:52 INFO avro.ReliableSpoolingFileEventReader: Preparing to move
file /tmp/flume/sqoop.txt to /tmp/flume/sqoop.txt.COMPLETED
18/11/21 15:40:56 INFO hdfs.HDFSDataStream: Serializer = TEXT, UseRawLocalFileS
ystem = false
18/11/21 15:40:56 INFO hdfs.BucketWriter: Creating /user/flume/FlumeData.154284
3656497.tmp
18/11/21 15:41:13 INFO file.EventQueueBackingStoreFile: Start checkpoint for /h
ome/zhangxinrun/.flume/file-channel/checkpoint/checkpoint, elements to sync = 5
18/11/21 15:41:13 INFO file.EventQueueBackingStoreFile: Updating checkpoint met
adata: logWriteOrderID: 1542843343817, queueSize: 0, queueHead: 8
18/11/21 15:41:13 INFO file.Log: Updated checkpoint for file: /home/zhangxinrun
/.flume/file-channel/data/log-3 position: 1662 logWriteOrderID: 1542843343817
18/11/21 15:41:13 INFO file.Log: Removing old file: /home/zhangxinrun/.flume/fi
le-channel/data/log-1
18/11/21 15:41:13 INFO file.Log: Removing old file: /home/zhangxinrun/.flume/fi
le-channel/data/log-1.meta
18/11/21 15:41:26 INFO hdfs.BucketWriter: Closing /user/flume/FlumeData.1542843
656497.tmp
18/11/21 15:41:26 INFO hdfs.BucketWriter: Renaming /user/flume/FlumeData.154284
3656497.tmp to /user/flume/FlumeData.1542843656497
18/11/21 15:41:26 INFO hdfs.HDFSEventSink: Writer callback called.
```

let us check the HDFS cluster, we can see there are two flume data files:

```
zhangxinrun@ubuntu: /usr/local/flume/conf   zhangxinrun@ubuntu: ~
zhangxinrun@ubuntu:~$ cp /home/zhangxinrun/handson-3/flume.txt /tmp/flume
zhangxinrun@ubuntu:~$ cp /home/zhangxinrun/handson-3/sqoop.txt /tmp/flume
zhangxinrun@ubuntu:~$ hdfs dfs -ls /user/flume
Found 2 items
-rw-r--r--   1 zhangxinrun supergroup        406 2018-11-21 15:38 /user/flume/F
lumeData.1542843499205
-rw-r--r--   1 zhangxinrun supergroup        514 2018-11-21 15:41 /user/flume/F
lumeData.1542843656497
```

let us cat them:

```
zhangxinrun@ubuntu:~$ hdfs dfs -cat /user/flume/FlumeData.1542843656497
Apache Sqoop(TM) is a tool designed for efficiently transferring bulk data betw
een Apache Hadoop and structured datastores such as relational databases.

Sqoop successfully graduated from the Incubator in March of 2012 and is now a T
op-Level Apache project: More information

Latest stable release is 1.4.7 (download, documentation). Latest cut of Sqoop2
is 1.99.7 (download, documentation). Note that 1.99.7 is not compatible with 1.
4.7 and not feature complete, it is not intended for production deployment.
zhangxinrun@ubuntu:~$ hdfs dfs -cat /user/flume//FlumeData.1542843499205
Flume is a distributed, reliable, and available service for efficiently collect
ing, aggregating, and moving large amounts of log data. It has a simple and fle
xible architecture based on streaming data flows. It is robust and fault tolera
nt with tunable reliability mechanisms and many failover and recovery mechanism
s. It uses a simple extensible data model that allows for online analytic appli
cation.

zhangxinrun@ubuntu:~$
```

I’ m really glad that it worked well.

B. MySQL

1. Log into MySQL use command: *mysql –u root password [password]*
2. However, I can’ t log in to MySQL with access denied

```
zhangxinrun@ubuntu:~$ /etc/init.d/mysql restart
[ ok ] Restarting mysql (via systemctl): mysql.service.
zhangxinrun@ubuntu:~$ mysql -u root mysq
ERROR 1698 (28000): Access denied for user 'root'@'localhost'
zhangxinrun@ubuntu:~$ mysql -u root -p
Enter password:
ERROR 1698 (28000): Access denied for user 'root'@'localhost'
zhangxinrun@ubuntu:~$ mysql -u root -p
Enter password:
ERROR 1698 (28000): Access denied for user 'root'@'localhost'
```

3. I can’ t even reset my password. Then, after few hours attempts, I finally found that I have to use sudo first according to this answer:

```
zhangxinrun@ubuntu:~$ sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.7.24-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

4. Create a new database and use it

```
mysql> CREATE DATABASE bigdata;
Query OK, 1 row affected (0.00 sec)

mysql>
```

```
mysql> USE bigdata;
Database changed
```

5. Create a new table

```
mysql> CREATE TABLE `student`
    -> (`name` VARCHAR(20) NOT NULL,
    -> `PUID` VARCHAR(5) PRIMARY KEY,
    -> `major` VARCHAR(4) NOT NULL);
Query OK, 0 rows affected (0.03 sec)
```

6. Insert some data into the table

```
mysql> INSERT INTO `student`
    -> (`name`,`PUID`,`major`)
    -> VALUES
    -> ('ZXR','12345','ECE'),
    -> ('Jobs','54321','CS'),
    -> ('Gates','34521','CS')
    -> ;
Query OK, 3 rows affected (0.07 sec)
Records: 3  Duplicates: 0  Warnings: 0
```

```
mysql> SELECT * FROM `student`;
+-------+-------+-------+
| name  | PUID  | major |
+-------+-------+-------+
| ZXR   | 12345 | ECE   |
| Gates | 34521 | CS    |
| Jobs  | 54321 | CS    |
+-------+-------+-------+
3 rows in set (0.00 sec)
```

7. Create a new directory in HDFS:

```
zhangxinrun@ubuntu:~$ hdfs dfs -mkdir /user/sqoop
```

8. Run sqoop with the command:

sqoop import --connect jdbc:mysql://localhost/bigdata --username root --P --table student --m 1 --target-dir /user/bigdata/sqoop

some problems occur:

```
18/11/21 19:05:25 ERROR orm.CompilationManager: It seems as though you are runn
ing sqoop with a JRE.
18/11/21 19:05:25 ERROR orm.CompilationManager: Sqoop requires a JDK that can c
ompile Java code.
18/11/21 19:05:25 ERROR orm.CompilationManager: Please install a JDK and set $J
AVA_HOME to use it.
18/11/21 19:05:25 ERROR tool.ImportTool: Import failed: java.io.IOException: Co
uld not start Java compiler.
```

I solved this problem with link

9. Completed and we can see the table in Hadoop cluster:

```
zhangxinrun@ubuntu:~$ hdfs dfs -ls /user/bigdata/sqoop
Found 2 items
-rw-r--r--   1 zhangxinrun supergroup          0 2018-11-21 19:22 /user/bigdata
/sqoop/_SUCCESS
-rw-r--r--   1 zhangxinrun supergroup         43 2018-11-21 19:22 /user/bigdata
/sqoop/part-m-00000
zhangxinrun@ubuntu:~$ hdfs dfs -cat /user/bigdata/sqoop/part-m-00000
ZXR,12345,ECE
Gates,34521,CS
Jobs,54321,CS
```

C. Hive

1. Create a new directory in Hadoop Cluster:

```
zhangxinrun@ubuntu:~$ hdfs dfs -mkdir /user/bigdata/hive
```

2. Upload the students.txt into the directory

```
zhangxinrun@ubuntu:~$ hdfs dfs -copyFromLocal /home/zhangxinrun/handson-3/stude
nts.txt /user/bigdata/hive
```

3. Create a new table:

```
hive> CREATE TABLE students (name STRING, course STRING, grade INT) ROW FORMAT
DELIMITED FIELDS TERMINATED BY ',';
OK
Time taken: 1.73 seconds
```

4. Load data into the table:

```
hive> LOAD DATA INPATH '/user/bigdata/hive/students.txt' OVERWRITE INTO TABLE s
tudents;
Loading data to table default.students
OK
Time taken: 2.229 seconds
```

5. Execute the query to get average score for each subject:

```
hive> SELECT students.course, AVG(students.grade) FROM students GROUP BY course;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the futu
re versions. Consider using a different execution engine (i.e. spark, tez) or us
ing Hive 1.X releases.
Query ID = zhangxinrun_20181121195725_9ee12ccc-5ef7-454f-9d71-e41fc719b5e3
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1542855912710_0002, Tracking URL = http://ubuntu:8088/proxy/a
pplication_1542855912710_0002/
Kill Command = /usr/local/hadoop/bin/hadoop job  -kill job_1542855912710_0002
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2018-11-21 19:57:44,762 Stage-1 map = 0%,  reduce = 0%
2018-11-21 19:57:56,884 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 1.86 se
c
2018-11-21 19:58:07,491 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 3.97
sec
```

```
Total MapReduce CPU Time Spent: 3 seconds 970 msec
OK
ece354  82.4375
ece571  78.6875
ece595  80.28571428571429
Time taken: 43.031 seconds, Fetched: 3 row(s)
```