# MODUL PRASYARAT – SPARK – HADOOP

## Prasyarat

Sebelum kita masuk pada proses install, hal yang wajib dilakukan adalah menginstall semua *basic tools*. Dengan kita menginstal *tools* tersebut, kita akan siap untuk proses instalasi selanjutnya. Daftar *tools* yang akan kita install adalah sebagai berikut :

* Java 8
* Scala
* SBT
* Git
* Python PIP

### Instalasi Java 8

Pertama lakukan instalasi oracle java 8. Tambahkan repositori dan jalankan perintah berikut.

|  |
| --- |
| # add-apt-repository ppa:webupd8team/java  # apt update  # apt install oracle-java8-installer |

Selanjutnya set variabel JAVA\_HOME pada sistem, lakukan perubahan pada /etc/environment tambahkan script berikut ini pada baris selanjutnya

|  |
| --- |
| JAVA\_HOME="/usr/lib/jvm/java-8-oracle" |

Kemudian update source dengan menjalankan perintah berikut

|  |
| --- |
| # source /etc/environtment |

### Instalasi Scala

Scala atau Scalable Language adalah bahasa pemrograman yang untuk memperbaiki kekurangan yang ada pada Java. Scala dibuat untuk menunjukkan pola suatu pemrograman secara singkat, praktis, dan elegan. Scala merupakan bahasa yang berorientasi objek, fungsional, extesible, dan interoperable.

Lakukan instalasi scala dengan menjalankan perintah berikut

|  |
| --- |
| # apt install scala |

### Instalasi SBT

SBT adalah [open-source](https://translate.googleusercontent.com/translate_c?depth=1&hl=id&prev=search&rurl=translate.google.com&sl=en&sp=nmt4&u=https://en.wikipedia.org/wiki/Open-source_software&xid=17259,15700019,15700186,15700190,15700248&usg=ALkJrhhH4Qzu8A9iNWeCJp0TM_ICaeZGUA) build tools untuk proyek-proyek [Scala](https://translate.googleusercontent.com/translate_c?depth=1&hl=id&prev=search&rurl=translate.google.com&sl=en&sp=nmt4&u=https://en.wikipedia.org/wiki/Scala_(programming_language)&xid=17259,15700019,15700186,15700190,15700248&usg=ALkJrhj5B5P_AReXnn6VQPGq5MjHAFw21A" \o "Scala (bahasa pemrograman)) dan [Java](https://translate.googleusercontent.com/translate_c?depth=1&hl=id&prev=search&rurl=translate.google.com&sl=en&sp=nmt4&u=https://en.wikipedia.org/wiki/Java_(programming_language)&xid=17259,15700019,15700186,15700190,15700248&usg=ALkJrhgIj54KoKRBJYVas4ZfDDpESLreLA) , mirip dengan Java [Maven](https://translate.googleusercontent.com/translate_c?depth=1&hl=id&prev=search&rurl=translate.google.com&sl=en&sp=nmt4&u=https://en.wikipedia.org/wiki/Apache_Maven&xid=17259,15700019,15700186,15700190,15700248&usg=ALkJrhgoNw29ElqF1Jnv9cJfQLEzr3pvng" \o "Apache Maven) dan [Ant](https://translate.googleusercontent.com/translate_c?depth=1&hl=id&prev=search&rurl=translate.google.com&sl=en&sp=nmt4&u=https://en.wikipedia.org/wiki/Apache_Ant&xid=17259,15700019,15700186,15700190,15700248&usg=ALkJrhhimKkNvrdD6Nmcl_qVeegntSeKsw) .

Lakukan instalasi SBT dengan menjalankan perintah berikut

|  |
| --- |
| # echo "deb https://dl.bintray.com/sbt/debian /" | sudo tee -a /etc/apt/sources.list.d/sbt.list  # sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv 2EE0EA64E40A89B84B2DF73499E82A75642AC823  # apt update  # apt install sbt |

### Git

Git adalah version control system yang digunakan para developer untuk mengembangkan software secara bersama-bersama. Version control pada Git bertugas mencatat setiap perubahan pada file proyek yang dikerjakan oleh banyak orang maupun sendiri.

Lakukan instalasi git dengan menjalankan perintah berikut

|  |
| --- |
| # apt install git |

### Python PIP

PIP adalah PIP Installs Packages atau sebagai package manager yang berfungsi untuk mencari, menginstall, me-manage modules atau package pada installasi python kita.

Lakukan instalasi pip dengan menjalankan perintah berikut

|  |
| --- |
| # apt install python-pip |

### Hosts

File hosts adalah file pada sistem operasi yang berguna untuk melakukan resolve domain atau hostname ke alamat IP. Petakan nama host / domain pada alamat IP server.

Ubah file hosts seperti contoh berikut

|  |
| --- |
| # nano /etc/hosts  127.0.0.1 localhost  157.230.241.208 BigData |

## Apache Spark

*Apache* *Spark* adalah *engine cluster computing* *open* *source* yang khusus didesain untuk memproses big data, dibangun dengan menggunakan *scala*, sehingga membutuhkan beberapa *packages* di antaranya adalah *scala, sbt, java* yang sudah kita install pada materi sebelumnya. Langkah instalasinya adalah sebagai berikut,

### Install Spark

Pilih paket spark 2.3 pada repositori resmi yang telah disediakan oleh Apache dan lakukan instalasi.

|  |
| --- |
| # wget https://www-us.apache.org/dist/spark/spark-2.3.2/spark-2.3.2-bin-hadoop2.7.tgz  # tar xzf spark-2.3.2-bin-hadoop2.7.tgz  # mv spark-2.3.2-bin-hadoop2.7 /usr/local/spark |

Tambahkan environment variabel spark pada /etc/profile dengan menambahkan script berikut ini pada baris selanjutnya

|  |
| --- |
| PATH=$PATH:/usr/local/spark/bin:/usr/local/spark/sbin |

Kemudian update source dengan menjalankan perintah berikut

|  |
| --- |
| # source /etc/profile |

Lakukan uji coba dengan menjalankan perintah berikut, dan tekan ctrl + c untuk keluar. jika berhasil maka instalasi sukses.

|  |
| --- |
| # spark-shell |



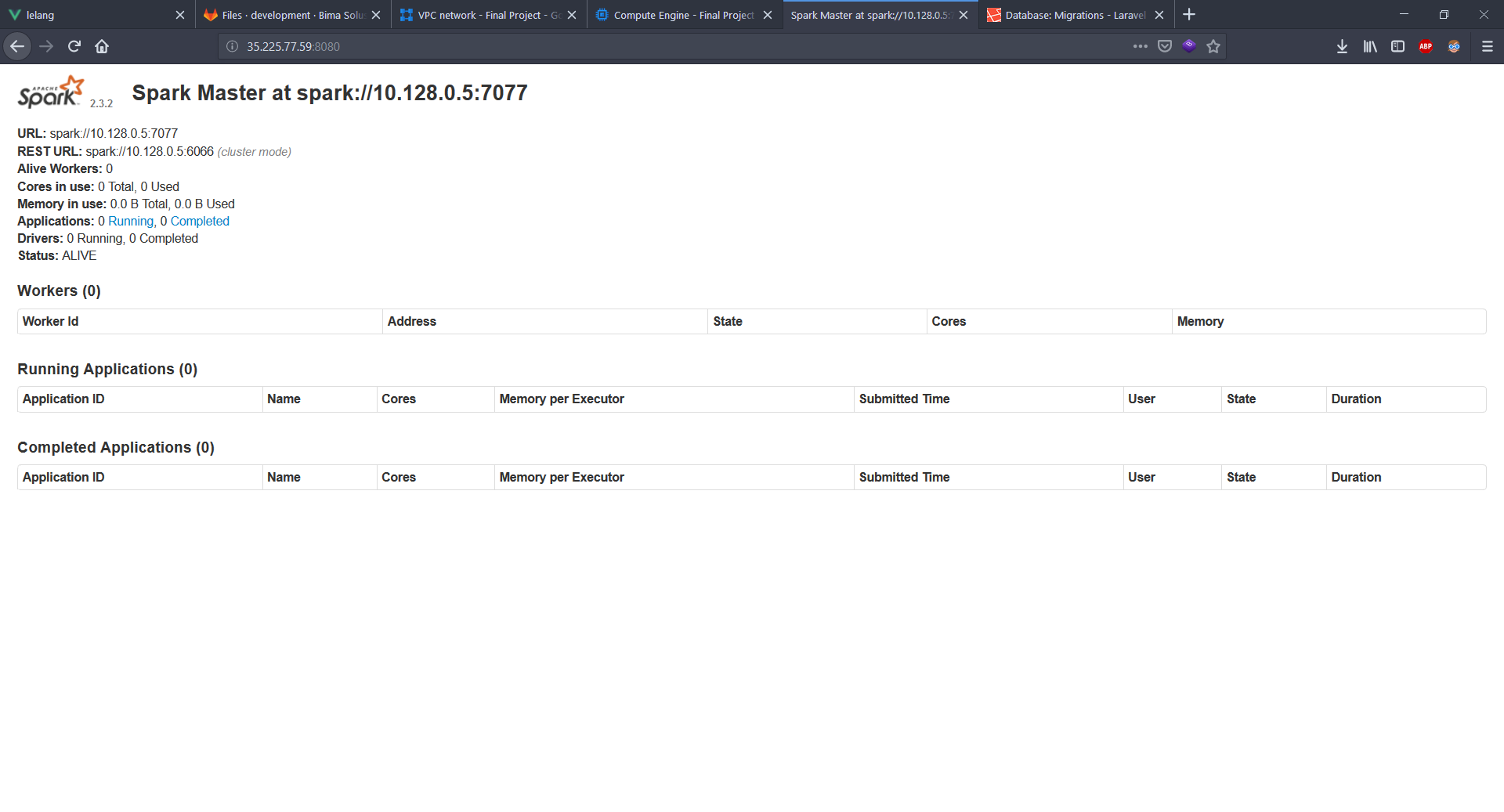
Gambar ‎3.1 Uji coba spark shell

### Menjalankan Spark Cluster

Pada master node yang telah terpasang spark, jalankan perintah berikut.

|  |
| --- |
| # start-master.sh -h <ip-address>  Contoh: start-master.sh -h 10.128.0.5 |

Jika berhasil akan muncul seperti berikut jika mengakses menggunakan port 8080 pada node master.



Gambar ‎3.2 Tampilan Web UI Spark

Jika memiliki sebuah server / komputer yang berguna sebagai slave maka jalankan perintah berikut untuk membuat node menjadi slave.

|  |
| --- |
| # spark-slave.sh <url\_spark\_master>  Contoh: spark-slave.sh spark://10.128.0.5:7077 |

## Apache Hadoop

Hadoop adalah framework atau platform open source berbasis Java di bawah lisensi Apache untuk support aplikasi yang berjalan pada Big Data. Langkah instalasinya adalah sebagai berikut,

### Prasyarat

Buat user terlebih dahulu untuk hadoop

|  |
| --- |
| # adduser hduser  # usermod -aG sudo hduser |

Kemudian pindah ke user hduser

|  |
| --- |
| # su hduser  $ cd ~ |

Tanda **$** melambangkan bahwa menggunakan user yang telah dibuat yaitu **hduser**, pada setiap perintah yang dijalankan pada modul ini.

### Instalasi Hadoop

Pertama unduh terlebih dahulu paket Hadoop dengan menjalankan perintah dibawah ini, dan lakukan instalasi.

|  |
| --- |
| $ wget https://www-eu.apache.org/dist/hadoop/common/hadoop-3.1.1/hadoop-3.1.1.tar.gz  $ tar xzf hadoop-3.1.1.tar.gz  $ mv hadoop-3.1.1 hadoop |

Tambahkan environment variabel hadoop di akhir baris file .bashrc

|  |
| --- |
| $ nano ~/.bashrc  PATH=$PATH:/home/hduser/hadoop/bin:/home/hduser/hadoop/sbin |

Update source dengan menjalankan perintah berikut

|  |
| --- |
| $ source ~/.bashrc |

Jalankan perintah berikut untuk melakukan cek versi hadoop yang telah berhasil di install

|  |
| --- |
| # hadoop version |

### Single Node Cluster

Pertama pastikan anda dapat melakukan koneksi ssh ke localhost tanpa password, untuk memastikannya jalankan perintah berikut

|  |
| --- |
| $ ssh-keygen -t rsa -P '' -f ~/.ssh/id\_rsa  $ cat ~/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys  $ chmod 0600 ~/.ssh/authorized\_keys |

Untuk menguji apakah sudah berhasil atau tidak, maka lakukan perintah berikut

|  |
| --- |
| $ ssh localhost  $ exit |

Ubah script file /home/hduser/hadoop/etc/hadoop/core-site.xml seperti berikut

|  |
| --- |
| $ nano /home/hduser/hadoop/etc/hadoop/core-site.xml  <?xml version="1.0" encoding="UTF-8"?>  <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  <configuration>  <property>  <name>fs.defaultFS</name>  <value>hdfs://localhost:9000</value>  </property>  </configuration> |
|  |

Ubah script file /home/hduser/hadoop/etc/hadoop/core-site.xml seperti berikut

|  |
| --- |
| $ nano /home/hduser/hadoop/etc/hadoop/hdfs-site.xml  <?xml version="1.0" encoding="UTF-8"?>  <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  <configuration>  <property>  <name>dfs.namenode.name.dir</name>  <value>/home/hduser/hadoop/dfs/name</value>  </property>  <property>  <name>dfs.datanode.data.dir</name>  <value>/home/hduser/hadoop/dfs/data</value>  </property>  <property>  <name>dfs.replication</name>  <value>1</value>  </property>  <property>  <name>dfs.namenode.rpc-bind-host</name>  <value>0.0.0.0</value>  </property>  </configuration> |

Kemudian lakukan konfigurasi terhadap YARN, dengan mengubah script file /home/hduser/hadoop/etc/hadoop/mapred-site.xml seperti berikut

|  |
| --- |
| $ nano /home/hduser/hadoop/etc/hadoop/mapred-site.xml  <?xml version="1.0" encoding="UTF-8"?>  <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  <configuration>  <property>  <name>mapreduce.framework.name</name>  <value>yarn</value>  </property>  <property>         <name>yarn.resourcemanager.address</name>         <value>localhost:8032</value>      </property>  <property>  <name>mapreduce.application.classpath</name>  <value>$HADOOP\_MAPRED\_HOME/share/hadoop/mapreduce/\*:$HADOOP\_MAPRED\_HOME/share/hadoop/mapreduce/lib/\*</value>  </property>  </configuration> |

Selanjutnya ubah script file /home/hduser/hadoop/etc/hadoop/yarn-site.xml seperti berikut

|  |
| --- |
| $ nano /home/hduser/hadoop/etc/hadoop/yarn-site.xml  <?xml version="1.0" encoding="UTF-8"?>  <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  <configuration>  <property>  <name>yarn.nodemanager.aux-services</name>  <value>mapreduce\_shuffle</value>  </property>  <property>  <name>yarn.nodemanager.env-whitelist</name>  <value>JAVA\_HOME,HADOOP\_COMMON\_HOME,HADOOP\_HDFS\_HOME,HADOOP\_CONF\_DIR,CLASSPATH\_PREPEND\_DISTCACHE,HADOOP\_YARN\_HOME,HADOOP\_MAPRED\_HOME</value>  </property>  </configuration> |

Setelah itu lakukan format pada DFS

|  |
| --- |
| $ hdfs namenode -format |

Langkah terakhir ialah menjalankan namenode dan yarn.

|  |
| --- |
| $ start-dfs.sh  $ start-yarn.sh |

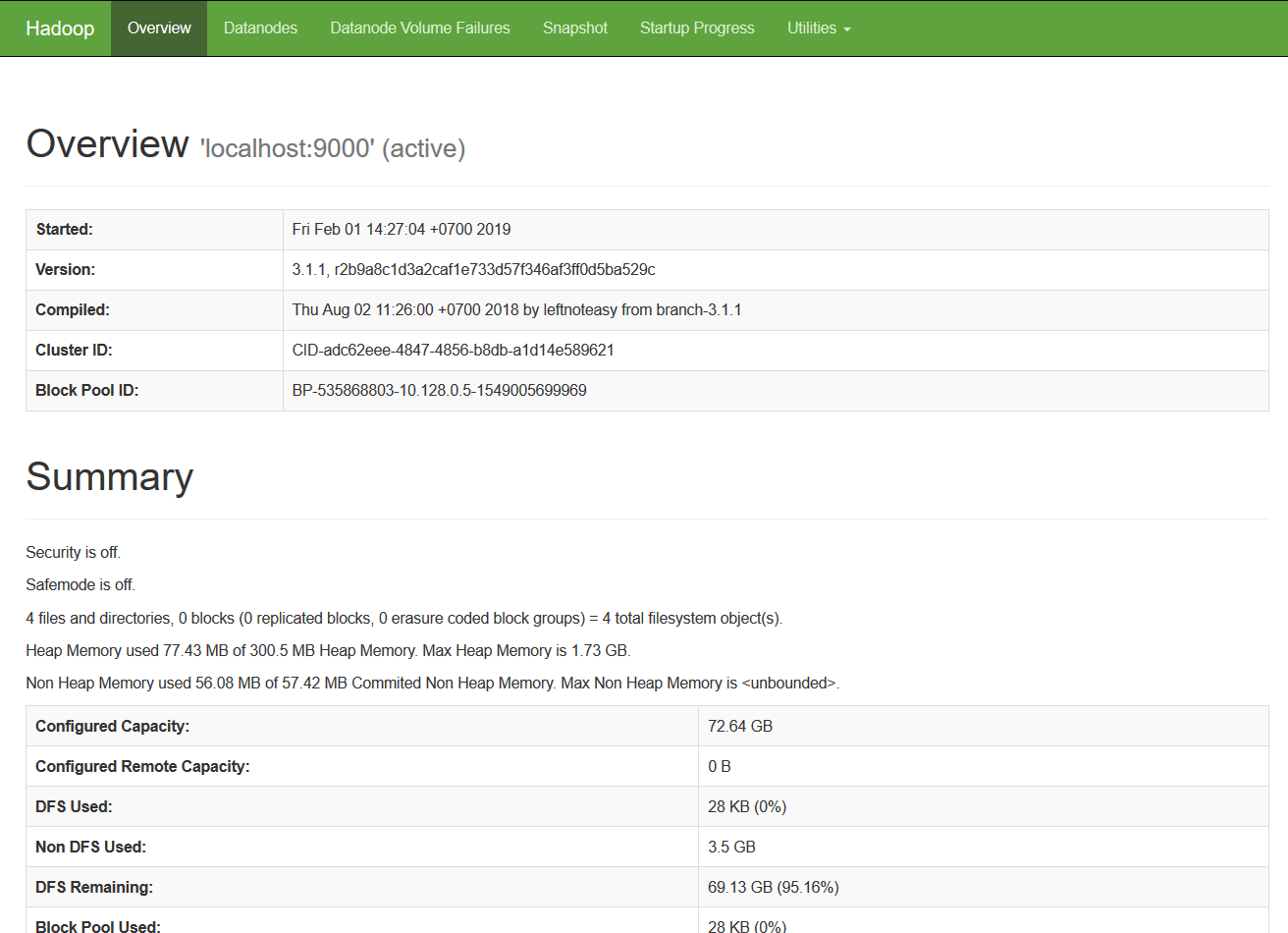
Untuk mematikan namenode dan yarn gunakan perintah berikut.

|  |
| --- |
| $ stop-dfs.sh  $ stop-yarn.sh |

Lakukan perintah berikut untuk melakukan uji coba terhadap hdfs, jika terdapat direktori test maka setup telah berhasil.

|  |
| --- |
| $ hdfs dfs -mkdir hdfs://localhost:9000/test  $ hdfs dfs -ls hdfs://localhost:9000/ |

Dan Anda bisa mengakses dengan port 9870 untuk melihat hadoop web ui.



Gambar ‎3.3 Tampilan Hadoop Web UI pada single node cluster

### Multi Node Cluster

Pertama pastikan setiap node baik master maupun worker telah di set hostname-nya, untuk cek jalankan perintah berikut, dan pastikan nama host yang muncul bukan localhost.

|  |
| --- |
| $ hostname |

Pertama pastikan anda dapat melakukan koneksi ssh ke localhost tanpa password, untuk memastikannya jalankan perintah berikut

|  |
| --- |
| $ ssh-keygen -t rsa -P '' -f ~/.ssh/id\_rsa  $ cat ~/.ssh/id\_rsa.pub >> ~/.ssh/authorized\_keys  $ chmod 0600 ~/.ssh/authorized\_keys |

Untuk menguji apakah sudah berhasil atau tidak, maka lakukan perintah berikut

|  |
| --- |
| $ ssh localhost  $ exit |

Setelah itu tambahkan host master dan worker ke dalam /etc/hosts, seperti contoh berikut.

|  |
| --- |
| $ sudo nano /etc/hosts  10.128.0.5 master-1  10.128.0.6 worker-1  10.128.0.7 worker-2  10.128.0.8 worker-3 |

Copy ssh public key ke master dan setiap worker.

|  |
| --- |
| $ ssh-copy-id -i ~/.ssh/id\_rsa.pub hduser@master-1  $ ssh-copy-id -i ~/.ssh/id\_rsa.pub hduser@worker-1  $ ssh-copy-id -i ~/.ssh/id\_rsa.pub hduser@worker-2  $ ssh-copy-id -i ~/.ssh/id\_rsa.pub hduser@worker-3 |

Ubah script file /home/hduser/hadoop/etc/hadoop/core-site.conf seperti berikut

|  |
| --- |
| $ nano /home/hduser/hadoop/etc/hadoop/core-site.conf  <?xml version="1.0" encoding="UTF-8"?>  <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  <configuration>  <property>  <name>fs.defaultFS</name>  <value>hdfs://master-1:9000</value>  </property>  </configuration> |

Ubah script file /home/hduser/hadoop/etc/hadoop/core-site.conf seperti berikut

|  |
| --- |
| $ nano /home/hduser/hadoop/etc/hadoop/hdfs-site.conf  <?xml version="1.0" encoding="UTF-8"?>  <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  <configuration>  <property>  <name>dfs.namenode.name.dir</name>  <value>/home/hduser/hadoop/dfs/name</value>  </property>  <property>  <name>dfs.datanode.data.dir</name>  <value>/home/hduser/hadoop/dfs/data</value>  </property>  <property>  <name>dfs.replication</name>  <value>1</value>  </property>  <property>  <name>dfs.namenode.rpc-bind-host</name>  <value>0.0.0.0</value>  </property>  </configuration> |

Lakukan konfigurasi terhadap YARN, dengan mengubah script file /home/hduser/hadoop/etc/hadoop/mapred-site.conf seperti berikut

|  |
| --- |
| $ nano /home/hduser/hadoop/etc/hadoop/mapred-site.conf  <?xml version="1.0" encoding="UTF-8"?>  <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  <configuration>  <property>  <name>mapreduce.framework.name</name>  <value>yarn</value>  </property>  <property>         <name>yarn.resourcemanager.address</name>         <value>localhost:8032</value>      </property>  <property>  <name>mapreduce.application.classpath</name>  <value>$HADOOP\_MAPRED\_HOME/share/hadoop/mapreduce/\*:$HADOOP\_MAPRED\_HOME/share/hadoop/mapreduce/lib/\*</value>  </property>  </configuration> |

Selanjutnya ubah script file /home/hduser/hadoop/etc/hadoop/yarn-site.conf seperti berikut

|  |
| --- |
| $ nano /home/hduser/hadoop/etc/hadoop/yarn-site.conf  <?xml version="1.0" encoding="UTF-8"?>  <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>  <configuration>  <property>  <name>yarn.nodemanager.aux-services</name>  <value>mapreduce\_shuffle</value>  </property>  <property>  <name>yarn.nodemanager.env-whitelist</name>  <value>JAVA\_HOME,HADOOP\_COMMON\_HOME,HADOOP\_HDFS\_HOME,HADOOP\_CONF\_DIR,CLASSPATH\_PREPEND\_DISTCACHE,HADOOP\_YARN\_HOME,HADOOP\_MAPRED\_HOME</value>  </property>  </configuration> |

Lalu set master dan worker dari hdfs cluster, ubah script file /home/hduser/hadoop/etc/hadoop/worker seperti berikut

|  |
| --- |
| $ nano /home/hduser/hadoop/etc/hadoop/worker  master-1  worker-1  worker-2  worker-3 |

Setelah itu lakukan format pada DFS

|  |
| --- |
| $ hdfs namenode -format |

Sinkronisasi semua file konfigurasi ke setiap worker

|  |
| --- |
| for node in **worker-1 worker-2 worker-3**; do  scp /home/hduser/hadoop/etc/hadoop/\* $node:/home/hduser/hadoop/etc/hadoop/;  done |

Langkah terakhir yaitu jalankan namenode dan yarn.

|  |
| --- |
| $ start-dfs.sh  $ start-yarn.sh |

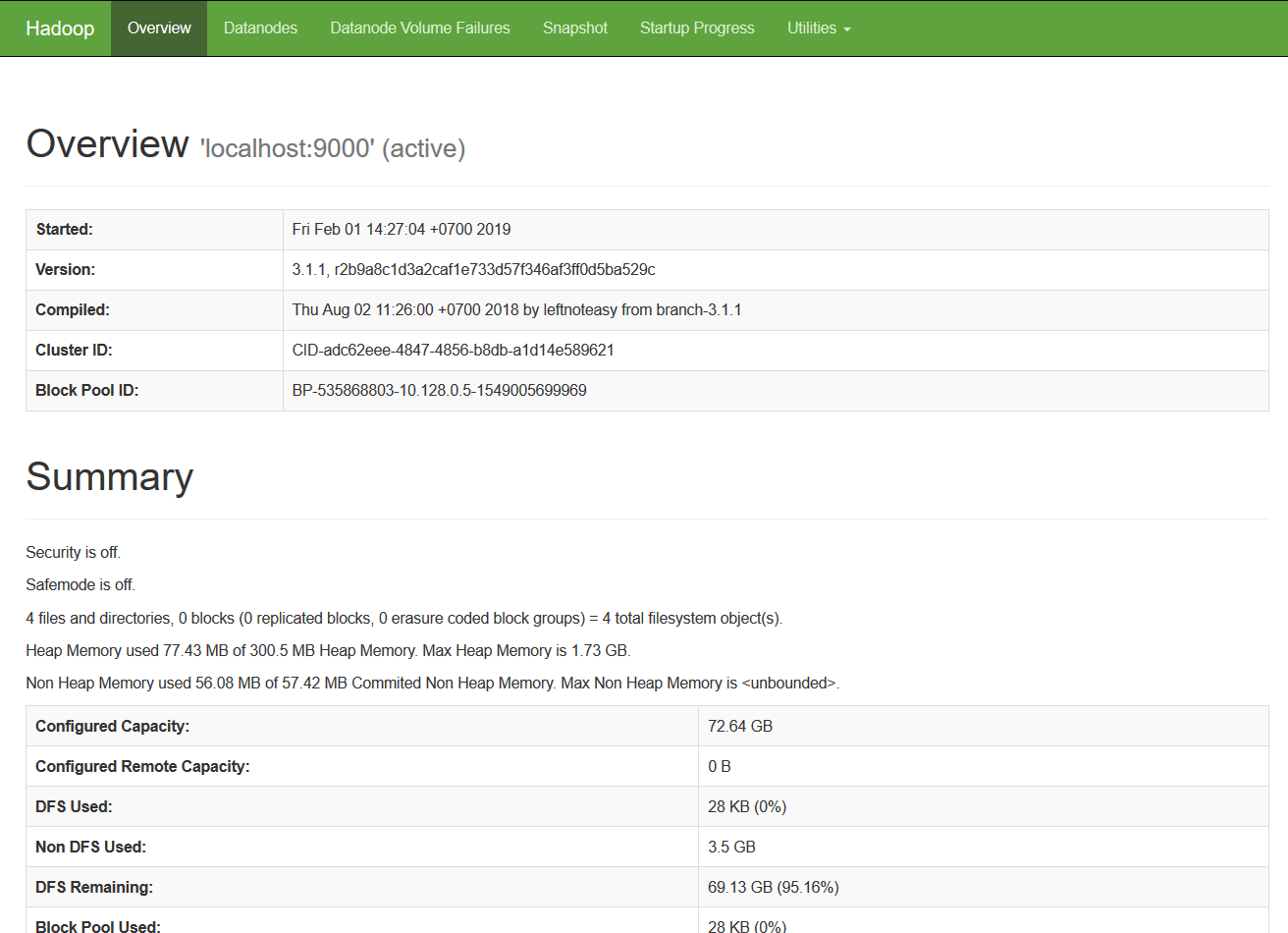
Untuk mematikan namenode dan yarn gunakan perintah berikut.

|  |
| --- |
| $ stop-dfs.sh  $ stop-yarn.sh |

Lakukan perintah berikut untuk melakukan uji coba terhadap hdfs, jika terdapat direktori test maka setup telah berhasil.

|  |
| --- |
| $ hdfs dfs -mkdir hdfs://localhost:9000/test  $ hdfs dfs -ls hdfs://localhost:9000/ |

Dan Anda bisa mengakses dengan port 9870 untuk melihat hadoop web ui.



Gambar ‎3.4 Tampilan Hadoop Web UI pada multi node cluster

### Membuat Direktori Tambahan

Di sini kita membuat direktori tambahan untuk keperluan aplikasi nantinya, jalankan perintah berikut.

|  |
| --- |
| $ hdfs dfs -mkdir -p hdfs://localhost:9000/user/hduser/job  $ hdfs dfs -mkdir -p hdfs://localhost:9000/user/hduser/kaspa  $ hdfs dfs -mkdir -p hdfs://localhost:9000/user/hduser/kafka-checkpoint  $ hdfs dfs -mkdir -p hdfs://localhost:9000/user/hduser/kaspa-checkpoint  $ hdfs dfs -mkdir -p hdfs://localhost:9000/user/hduser/schema/raw\_kaspa  $ hdfs dfs -mkdir -p hdfs://localhost:9000/user/hduser/file/maxmind |