Getting Web Server Data Into Hadoop

This presentation will include:

>install single node Hadoop on Ubuntu 14.04
>building (as an example only) a tiny web server
>getting data into Hadoop through network traffic

Installing Hadoop single node on Ubuntu

1. Install Java

```
darthhadoop@darthhadoop:~$
darthhadoop@darthhadoop:~$
darthhadoop@darthhadoop:~$ sudo apt-get install default-jdk
```

2_1. Create hadoop user and group

```
darthhadoop@darthhadoop:~$ sudo addgroup hadoop
Adding group `hadoop' (GID 1001) ...
Done.
darthhadoop@darthhadoop:~$ sudo adduser --ingroup hadoop hdpUser
adduser: Please enter a username matching the regular expression configured
via the NAME_REGEX[_SYSTEM] configuration variable.  Use the `--force-badname'
option to relax this check or reconfigure NAME_REGEX.
darthhadoop@darthhadoop:~$
```

2_2. Add hadoop user to hadoop group

```
darthhadoop@darthhadoop:~$
darthhadoop@darthhadoop:~$ sudo adduser --ingroup hadoop hdpuser
Adding user `hdpuser' ...
Adding new user `hdpuser' (1001) with group `hadoop' ...
Creating home directory `/home/hdpuser' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for hdpuser
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] Y
darthhadoop@darthhadoop:~$
darthhadoop@darthhadoop:~$ sudo adduser hdpuser sudo
Adding user `hdpuser' to group `sudo' ...
Adding user hdpuser to group sudo
Done.
darthhadoop@darthhadoop:~$
```

3. Login as hadoop user:

```
darthhadoop@darthhadoop:~$
darthhadoop@darthhadoop:~$ su - hdpuser
Password:
hdpuser@darthhadoop:~$
```

```
hdpuser@darthhadoop:~$ pwd
/home/hdpuser
hdpuser@darthhadoop:~$
hdpuser@darthhadoop:~$
```

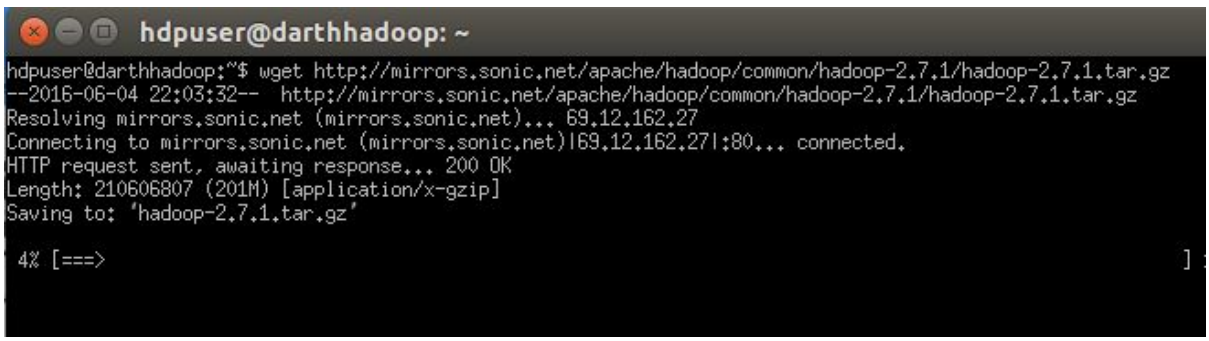
4. Generate ssh keys for hdpuser:

```
hdpuser@darthhadoop:~$
hdpuser@darthhadoop:~$ ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hdpuser/.ssh/id_rsa):
Created directory '/home/hdpuser/.ssh'.
Your identification has been saved in /home/hdpuser/.ssh/id_rsa.
Your public key has been saved in /home/hdpuser/.ssh/id_rsa.pub.
The key fingerprint is:
68:f5:7e:0d:dc:dc:e2:e9:5d:89:61:97:7c:25:5a:ce hdpuser@darthhadoop
The key's randomart image is:
+--[ RSA 2048]----+
|                 |
|                 |
|        .     o .|
|       o . . B.oo|
|      o S . +oE+o|
|     .   .  .++oo|
|          . ..+..|
|           . . ..|
|              . .|
+-----------------+
hdpuser@darthhadoop:~$
hdpuser@darthhadoop:~$
hdpuser@darthhadoop:~$ cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
hdpuser@darthhadoop:~$
hdpuser@darthhadoop:~$
hdpuser@darthhadoop:~$
```

5.Check if you can ssh into localhost

```
hdpuser@darthhadoop:~$
hdpuser@darthhadoop:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is 62:93:38:65:f5:d4:dc:ee:82:15:51:96:ac:e8:36:30.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.19.0-25-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

351 packages can be updated.
193 updates are security updates.


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

6.download Hadoop



7. After downloading is finished, unzip, check if everything installed, and move everything under /usr/local/hadoop path:

```
2016-06-04 22:06:22 (1.18 MB/s) - 'hadoop-2.7.1.tar.gz' saved [210606807/210606807]

hdpuser@darthhadoop:~$ ls hado*
hadoop-2.7.1.tar.gz
hdpuser@darthhadoop:~$ tar xvfz hadoop-2.7.1.tar.gz
```

```
hdpuser@darthhadoop:~$
hdpuser@darthhadoop:~$ sudo mv hadoop-2.7.1 /usr/local/hadoop/
[sudo] password for hdpuser:
hdpuser@darthhadoop:~$ ls /usr/local/hadoop/
bin  etc  include  lib  libexec  LICENSE.txt  NOTICE.txt  README.txt  sbin  share
hdpuser@darthhadoop:~$
```

```
hdpuser@darthhadoop:~$
hdpuser@darthhadoop:~$ sudo mv hadoop-2.7.1 /usr/local/hadoop/
[sudo] password for hdpuser:
hdpuser@darthhadoop:~$
```

8.Check the Java version:

```
darthhadoop@darthhadoop:~$ cd /usr/lib/jvm
darthhadoop@darthhadoop:/usr/lib/jvm$ ls
default-java  java-1.7.0-openjdk-amd64  java-7-openjdk-amd64
darthhadoop@darthhadoop:/usr/lib/jvm$
```

9. **Modify .bashrc** by going at the end of the file, and adding the Hadoop environment variables:

```
# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64
export HADOOP_HOME=/usr/local/hadoop
export PATH=$PATH:$HADOOP_HOME/bin
export PATH=$PATH:$HADOOP_HOME/sbin
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"
```

10. Edit hadoop-env.sh

This file contains **Hadoop's environment settings**

```
hdpuser@darthhadoop:~$
hdpuser@darthhadoop:~$ cd /usr/local/hadoop/etc/hadoop
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ ls hadoop-env.
hadoop-env.cmd  hadoop-env.sh
```

And modify the Java implementation from this:

```
# The java implementation to use.
export JAVA_HOME=${JAVA_HOME}
```

To this:

```
# set JAVA_HOME in this file, so that it is correctly defined on
# remote nodes.

# The java implementation to use.
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64

# The jsvc implementation to use. Jsvc is required to run secure datanodes
# that bind to privileged ports to provide authentication of data transfer
# protocol.  Jsvc is not required if SASL is configured for authentication of
```

11. Modify XML configuration files

```
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ ls -ltr *.xml
-rw-r--r-- 1 hdpuser hadoop 9683 iun 29  2015 hadoop-policy.xml
-rw-r--r-- 1 hdpuser hadoop  774 iun 29  2015 core-site.xml
-rw-r--r-- 1 hdpuser hadoop  690 iun 29  2015 yarn-site.xml
-rw-r--r-- 1 hdpuser hadoop 5511 iun 29  2015 kms-site.xml
-rw-r--r-- 1 hdpuser hadoop 3518 iun 29  2015 kms-acls.xml
-rw-r--r-- 1 hdpuser hadoop  620 iun 29  2015 httpfs-site.xml
-rw-r--r-- 1 hdpuser hadoop  775 iun 29  2015 hdfs-site.xml
-rw-r--r-- 1 hdpuser hadoop 4436 iun 29  2015 capacity-scheduler.xml
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$
```

11_1. Modify **core-site.xml**

This file contains configuration information that overrides the default values for the core Hadoop properties

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>

</configuration>
```

11_2. Add following lines:

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

12_1. Modify hdfs-site.xml

This file enables you to overwrite a number of default values that control
the HDFS

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>

</configuration>
```

12_2. Add following lines:

```
<configuration>
<property>
        <name>dfs.replication</name>
        <value>1</value>
</property>
<property>
        <name>dfs.namenode.name.dir</name>
        <value>file:/usr/local/hadoop_tmp/hdfs/namenode</value>
</property>
<property>
        <name>dfs.datanode.data.dir</name>
        <value>file:/usr/local/hadoop_tmp/hdfs/datanode</value>
</property>
</configuration>
```

13_1. Modify yarn-site.xml

This file enables you to overwrite a number of default values controlling
the YARN components

```
<?xml version="1.0"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->
<configuration>

<!-- Site specific YARN configuration properties -->

</configuration>
```

13_2. Add necessary lines, as below:

```
<configuration>

<property>
        <name>yarn.nodemanager.aux-services</name>
        <value>mapreduce_shuffle</value>
</propery>
<property>
        <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
        <value>org.apache.hadoop.mapred.ShuffleHandler</value>
</property>

</configuration>
```

yarn.nodemanager.aux-services : the name of an auxiliary service being added to the Node Manager

Yarn.nodemanager.aux-services.mapreduce_shuffle.class : implements the mapreduce_shuffle service

14_1: Create mapred-site.xml by copying the contain of mapred-site.xml.template:

```
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ ls mapred-site.*
mapred-site.xml.template
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ cp mapred-site.xml.template mapred-site.xml
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ ls mapred-site.*
mapred-site.xml  mapred-site.xml.template
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$
```

14_2 Modify the mapred-site.xml

This file enables you to overwrite a number of default values controlling the MapReduce job execution components.

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>

</configuration>
```

14_3. Add the following lines:

```
<!-- Put site-specific property overrides in this file. -->

<configuration>

<property>
        <name>mapreduce.framework.name</name>
        <value>yarn</value>
</property>

</configuration>
```

15. Create directories for hadoop-tmp, namenode and datanode

```
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ sudo mkdir -p /usr/local/hadoop_tmp
[sudo] password for hdpuser:
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ sudo mkdir -p /usr/local/hadoop_tmp/hdfs/namenode
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ sudo mkdir -p /usr/local/hadoop_tmp/hdfs/datanode
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ sudo chown -R hdpuser /usr/local/hadoop_tmp
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ ls /usr/local
bin  etc  games  hadoop  hadoop_tmp  include  lib  man  sbin  share  src
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ ls /usr/local/hadoop_tmp/
hdfs
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ ls /usr/local/hadoop_tmp/hdfs
datanode  namenode
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$
```

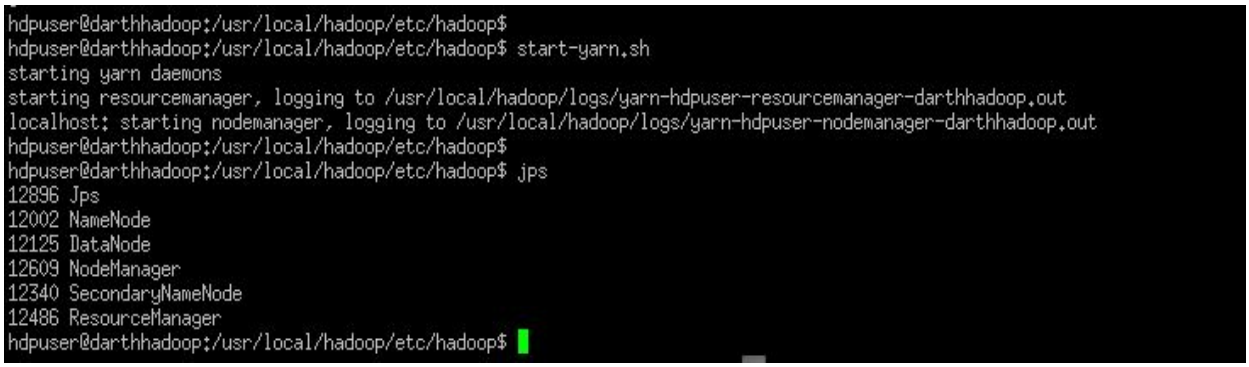
16. Format a new distributed filesystem - prepare to start the Hadoop cluster by formatting the NameNode (this must be done only once)

```
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ hdfs namenode -format
16/06/04 22:58:27 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = darthhadoop/127.0.1.1
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 2.7.1
```

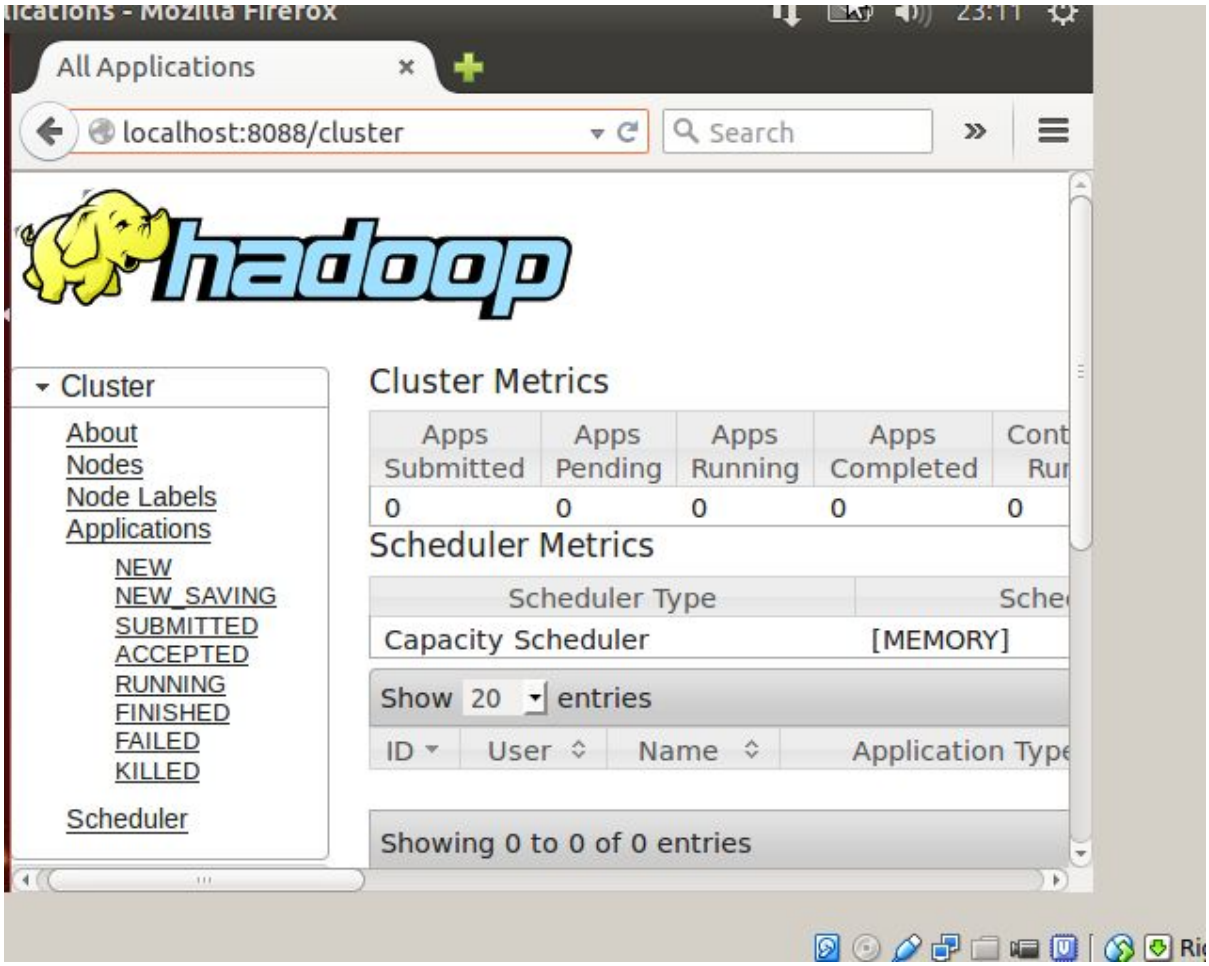
17. If successfully formatted, start the HDFS

```
16/06/04 23:01:38 INFO util.GSet: 0.029999999329447746% max memory 966.7 MB = 297.0 KB
16/06/04 23:01:38 INFO util.GSet: capacity      = 2^15 = 32768 entries
16/06/04 23:01:39 INFO namenode.FSImage: Allocated new BlockPoolId: BP-548142179-127.0.1.1-1465070498775
16/06/04 23:01:39 INFO common.Storage: Storage directory /usr/local/hadoop_tmp/hdfs/namenode has been successfully formatted.
16/06/04 23:01:39 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
16/06/04 23:01:39 INFO util.ExitUtil: Exiting with status 0
16/06/04 23:01:39 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at darthhadoop/127.0.1.1
************************************************************/
hdpuser@darthhadoop:/usr/local/hadoop/etc/hadoop$ start-dfs.sh
16/06/04 23:01:47 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java clas
e
Starting namenodes on [localhost]
localhost: starting namenode, logging to /usr/local/hadoop/logs/hadoop-hdpuser-namenode-darthhadoop.out
localhost: starting datanode, logging to /usr/local/hadoop/logs/hadoop-hdpuser-datanode-darthhadoop.out
```

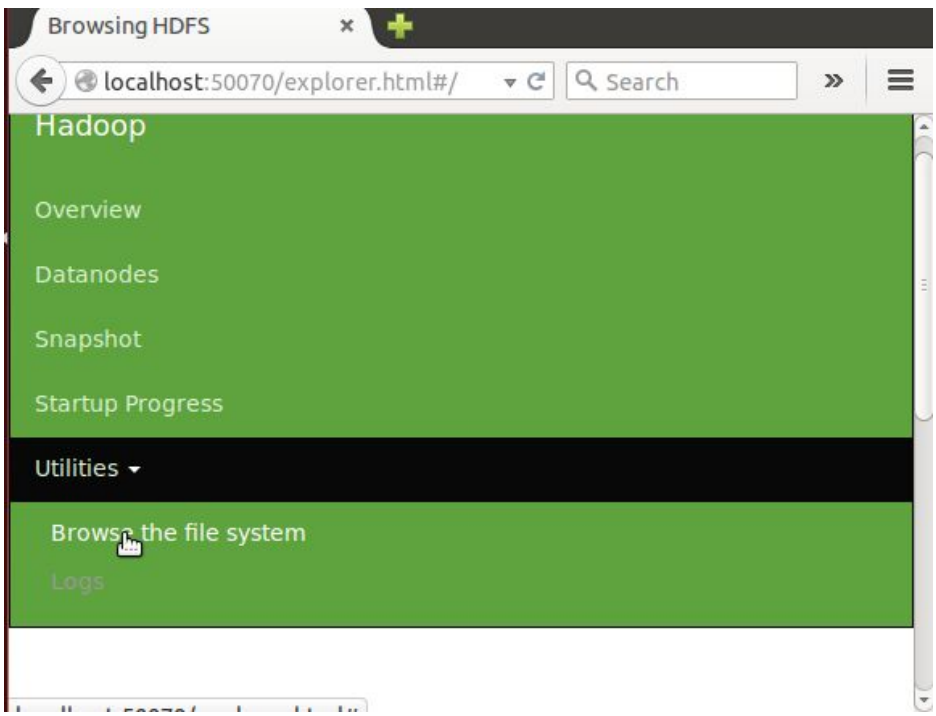
18. Start MapReduce daemons, and check if Hadoop's six services have started correctly

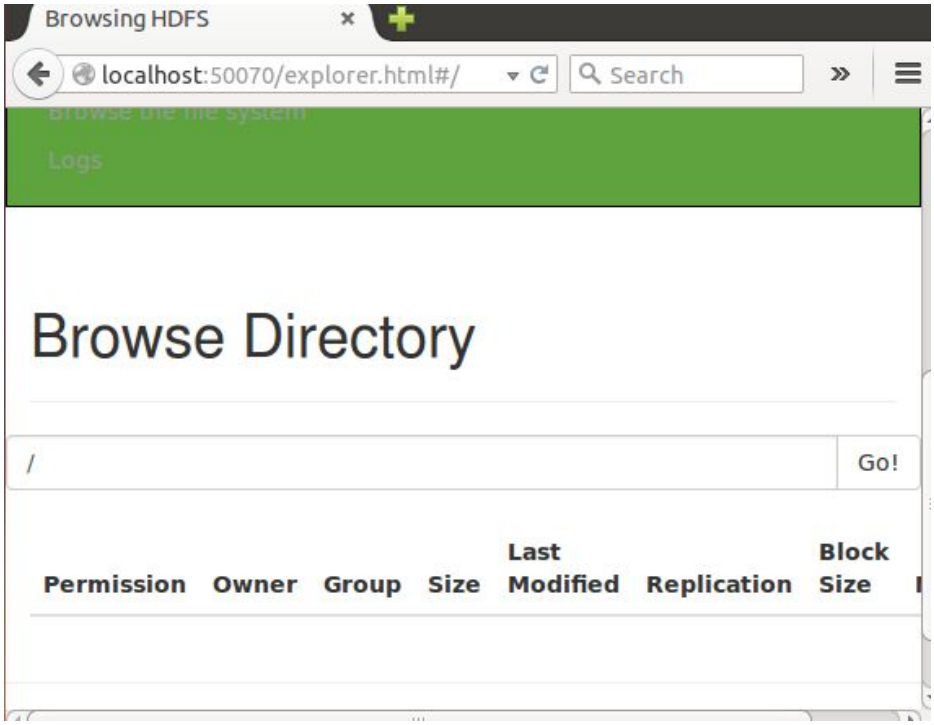


19. If all services are up and running, go to a browser and check Hadoop monitoring at **localhost:8088**



For browsing HDFS, go to **localhost:50070.**
And choose **“Browse the filesystem”**





Success! We have installed Hadoop! :)

Web servers and Hadoop

When it comes to Hadoop, data is categorized into two types:

1. Network traffic - data is generated by a system and sent across a network
2. File Data - data is generated by a system, but written to files on a filesystem

Obviously, we are at nr 1 - for now- you soon will see that nr. 2 applies as well.

Getting web server data into Hadoop

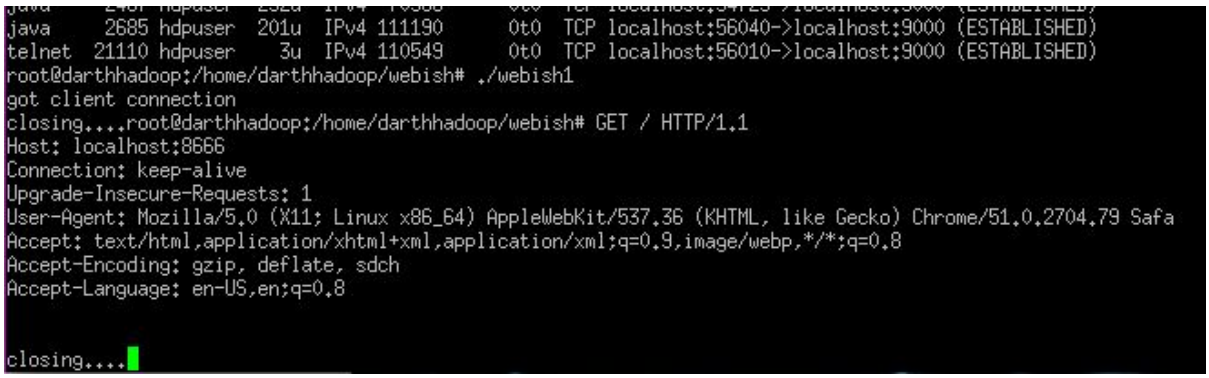
Basically, for this example, we will simply retrieve data via HTTP connection.

A tiny web server has been built for this example

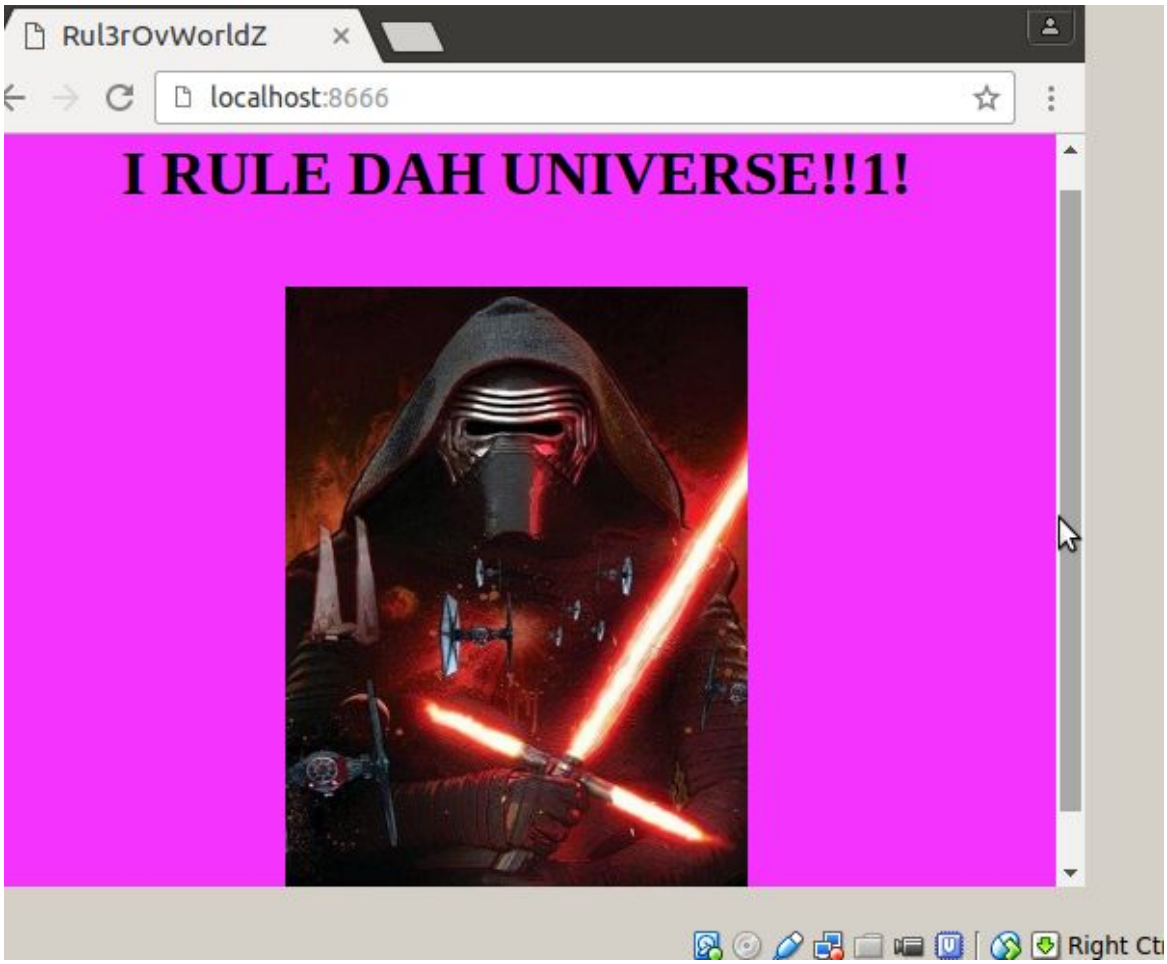
(source code included at the end)

Let’s run the program from the terminal (I named it *webish1*), as root user.

Open a browser and type **localhost:8666**



You should be getting here:



Good, the web server is working!

And now, let's see how to send this web server's data into Hadoop!

Run the program in a separate terminal;
Then, as a hadoop user, use **curl** tool for sending network data.

```
root@darthhadoop:/home/darthhadoop/webish# ./webish1
got client connection
closing....root@darthhadoop:/home/darthhadoop/webish# GET / HTTP/1.1
User-Agent: curl/7.35.0
Host: localhost:8666
Accept: */*


closing....
```

(using curl)

```
hdpuser@darthhadoop:/usr/local/hadoop$
hdpuser@darthhadoop:/usr/local/hadoop$ hadoop fs -ls /
16/06/05 20:56:56 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classe
 applicable
hdpuser@darthhadoop:/usr/local/hadoop$ curl localhost:8666 > LeRoi.txt
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   304    0   304    0     0  21757      0 --:--:-- --:--:-- --:--:-- 23384
hdpuser@darthhadoop:/usr/local/hadoop$
```

Check the filesize and copy the file to HDFS

```
hdpuser@darthhadoop:/usr/local/hadoop$
hdpuser@darthhadoop:/usr/local/hadoop$ ls -ldh LeRoi.txt
-rw-r--r-- 1 hdpuser hadoop 304 jun  5 20:57 LeRoi.txt
hdpuser@darthhadoop:/usr/local/hadoop$
hdpuser@darthhadoop:/usr/local/hadoop$ hadoop fs -put /usr/local/hadoop/LeRoi.txt /
16/06/05 20:58:03 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your plat
 applicable
```

Check if the transfer has been done successfully:

```
 applicable
hdpuser@darthhadoop:/usr/local/hadoop$ hadoop fs -ls /
16/06/05 20:58:28 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform...
 applicable
Found 1 items
-rw-r--r--   1 hdpuser supergroup        304 2016-06-05 20:58 /LeRoi.txt
hdpuser@darthhadoop:/usr/local/hadoop$
```

Let’s see what LeRoi.txt contains:

```
Found 1 items
-rw-r--r--   1 hdpuser supergroup        304 2016-06-05 20:58 /LeRoi.txt
hdpuser@darthhadoop:/usr/local/hadoop$
hdpuser@darthhadoop:/usr/local/hadoop$ hadoop fs -tail /LeRoi.txt
16/06/05 20:59:59 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes wh
 applicable
<!DOCTYPE html>
<html><head><title>Rul3rOvWorldZ</title>
<style>body {background-color: #F433FF} </style></head>
<body><center><h1> I RULE DAH UNIVERSE!!1! </h1><br>
<img src="https://upload.wikimedia.org/wikipedia/en/0/08/KyloRen-2015.jpg" style="width:240px; height:328px"></center></body></html>
hdpuser@darthhadoop:/usr/local/hadoop$
```

Nice! Html text (which kinda brings us to nr 2 data case scenario)

Let’s try an online HTML Formatter, to see if we obtained the correct information:
http://www.w3schools.com/html/tryit.asp?filename=tryhtml_default

If we type the same HTML text that we have in **LeRoi.txt**:

```
<!DOCTYPE html>
<html><head><title>Rul3rOvWorldZ</title>
<style>body{background-color: #F433FF} </style></head>
<body><center><h1>I RULE DAH UNIVERSE!!1!</h1><br>
<img
src="https://upload.wikimedia.org/wikipedia/en/0/08/KyloRen-2015.jpg"
style="width:240px; height:328px">
</center></body></html>
```

The output will be the same

Data has been successfully transferred! :)

Thank you for your attention!

```
//Web Server source code ***************************************



#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<netdb.h>
#include<arpa/inet.h>
#include<string.h>
#include<sys/stat.h>
#include<fcntl.h>
#include<sys/types.h>
#include<netinet/in.h>


char webish[]=
"<!DOCTYPE html>\r\n"
"<html><head><title>Rul3rOvWorldZ</title>\r\n"
"<style>body{background-color: #F433FF} </style></head>\r\n"
"<body><center><h1> I RULE DAH UNIVERSE!!1! </h1><br>\r\n"
"<img
src=\"https://upload.wikimedia.org/wikipedia/en/0/08/KyloRen-2015.jpg\"
style width=\"240px; height:328px\"></center></body></html>\r\n";

int main(int argc, char **argv)
{
      struct sockaddr_in server_addr, client_addr;
      socklen_t sin_len=sizeof(client_addr);
      int server, client;
      char buf[70000];
      int lilkylo;
      int on=1;


      server=socket(AF_INET, SOCK_STREAM, 0);
      if(server<0)
      {
      perror("socket");
      exit(1);

      }
```

```
      setsockopt(server, SOL_SOCKET, SO_REUSEADDR, &on, sizeof(int));

      server_addr.sin_family=AF_INET;
      server_addr.sin_addr.s_addr=INADDR_ANY;
      server_addr.sin_port=htons(8666);

      if(bind(server, (struct sockaddr *)&server_addr,
sizeof(server_addr))==-1)
      {
                  perror("bind");
                  close(server);
                  exit(1);
      }

      if(listen(server, 10) == -1)
                  {
                        perror("listen");
                        close(server);
                        exit(1);
                  }


      while(1)
      {
                  client=accept(server, (struct sockaddr *)&client_addr,
&sin_len);
                  if(client == -1)
                  {
                        perror("can't connect\n");
                        continue;
                  }

      printf("got client connection \n");
      if(!fork())
      {
                  close(server);
                  memset(buf, 0, sizeof(buf));
                  read(client, buf, sizeof(buf)-1);

                  printf("%s\n",buf);



            if(!strncmp(buf, "GET /doctest.jpg", 16))
                        {
```

```
                              lilkylo=open("doctest.jpg", O_RDONLY);
                              sendfile(client, lilkylo, NULL,68000);
                              close(lilkylo);
                        }
            else

            write(client, webish, sizeof(webish) - 1);

      }
      close(client);
      printf("closing....");
      exit(0);
      }

      return 0;
}

//*****************************************************************
```