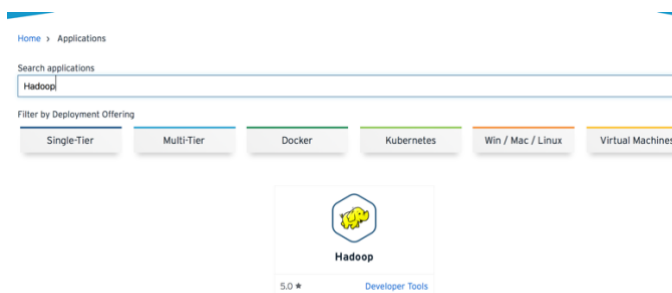


WEEK 5 ASSIGNMENT

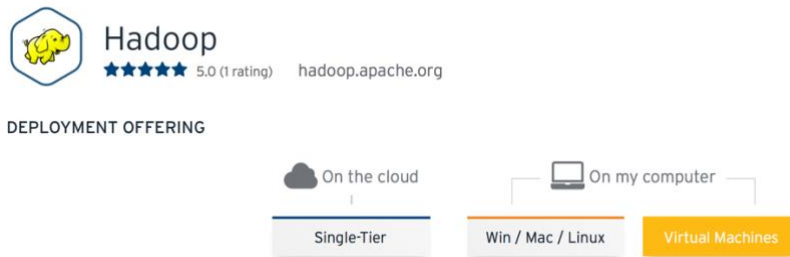
Large-Scale Data Storage Systems – DATA-5400 | Spring 2020

Christina Morgenstern

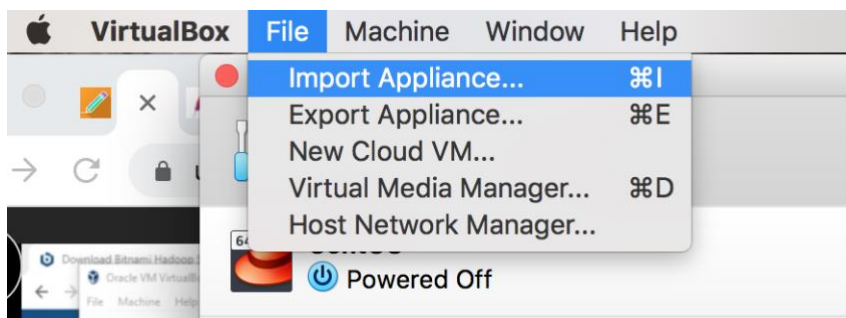
For this week's assignment, I have chosen the hands-on option because I want to get experience with Hadoop. As for the choice of option, I started with Option 1 of using Hadoop on my VirtualBox. To install Hadoop on the VM, I downloaded the Hadoop VirtualBox image from Bitnami. Bitnami offers easy to configure software apps for download, mostly open source and free (<https://bitnami.com>). On the bitnami website, go to Community and search for Hadoop.



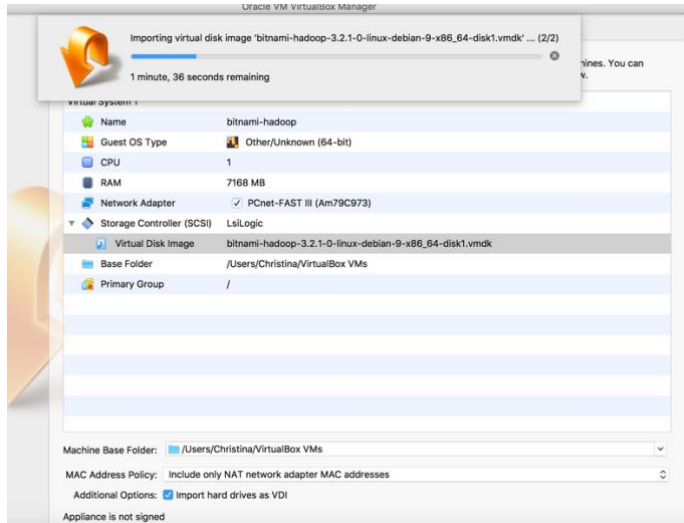
Choose to download the version for Virtual Machines.



On my Mac, I opened VirtualBox and imported the Hadoop VM image.



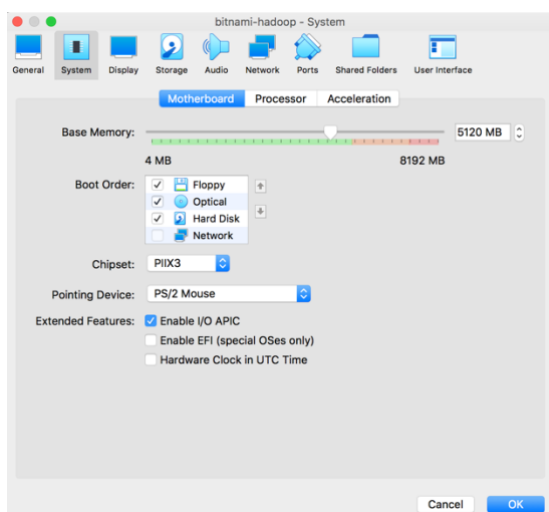
Choose the file previously downloaded for import.



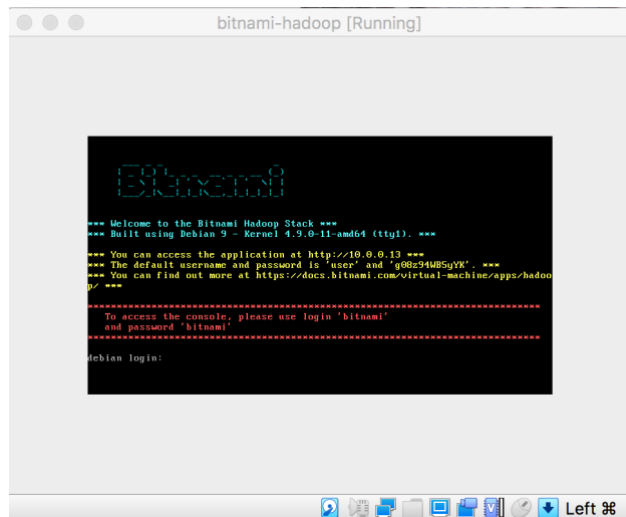
Bitnami Hadoop was successfully installed.



In the settings options, amend the memory. I couldn't choose 8GB of memory because of already too little memory. So, I chose 5120 MB and 1 CPU as suggested in the lecture.



Start bitnami-hadoop and log on using bitnami as login and password.



Disable the firewall.

```

bitnami@debian:~$ sudo -i -u root
root@debian:~# systemctl disable ufw
Synchronizing state of ufw.service with SysV service script with /lib/systemd/sy
stemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable ufw
root@debian:~# exit
logout
bitnami@debian:~$ _

```

SSH via Putty, MobaXterm or Mac is disabled by default in Hadoop. The following changes were made in order to allow SSH.

```

bitnami@debian:~$ sudo -i -u root
root@debian:~# cd/etc/ssh
-bash: cd/etc/ssh: No such file or directory
root@debian:~# cd/etc/ssh
-bash: cd/etc/ssh: No such file or directory
root@debian:~# cd /etc
root@debian:/etc# cd ssh
root@debian:/etc/ssh# ls
moduli          ssh_host_ecdsa_key      ssh_host_rsa_key
ssh_config      ssh_host_ecdsa_key.pub  ssh_host_rsa_key.pub
sshd_config     ssh_host_ed25519_key
sshd_not_to_be_run ssh_host_ed25519_key.pub
root@debian:/etc/ssh#

```

Remove the file “sshd_not_to_be_run”

```

root@debian:/etc/ssh# rm sshd_not_to_be_run
-bash: rm sshd_not_to_be_run: command not found
root@debian:/etc/ssh# rm sshd_not_to_be_run
root@debian:/etc/ssh# _

```

Restart the SSH service and check if it is working.

```
root@debian:/etc/ssh# rm_sshd_not_to_be_run
-bash: rm_sshd_not_to_be_run: command not found
root@debian:/etc/ssh# rm_sshd_not_to_be_run
root@debian:/etc/ssh# service ssh restart
root@debian:/etc/ssh# ps -fe | grep ssh
root      3587      1  0 21:33 ?        00:00:00 /usr/sbin/sshd -D
root      3623    3541  0 21:38 tty1      00:00:00 grep ssh
root@debian:/etc/ssh# netstat -tulnp | grep 22
tcp        0      0 0.0.0.0:22        0.0.0.0:*        LISTEN
1722/java
tcp        0      0 0.0.0.0:22        0.0.0.0:*        LISTEN
3587/sshd
tcp6       0      0 :::22            :::*             LISTEN
3587/sshd
tcp6       0      0 127.0.0.1:1527    :::*             LISTEN
2221/java
tcp6       0      0 :::9083          :::*             LISTEN
2278/java
udp        0      0 10.0.0.13:68     0.0.0.0:*
225/systemd-network
udp6       0      0 fe80::a00:27ff:fe70:546 :::*
225/systemd-network
root@debian:/etc/ssh# exit
logout
bitnami@debian:~$
```

[illegible][illegible]

Use the `cat` command to display the contents of the `testfile.txt`

```
Thu Feb 13 22:01:26 UTC 2020
bitnami@debian:~$ hdfs dfs -cat /tmp/testfile.txt
2020-02-13 22:02:08,765 INFO sasl.SaslDataTransferClient: SASL encryption trust
check: localhostTrusted = false, remoteHostTrusted = false
This is sentence 1.
This is sentence 2.
This is sentence 3.

bitnami@debian:~$ _
```

Create a folder named HDFS using the `hdfs dfs -mkdir` command and verify using the `hdfs dfs -ls` command. (I previously created a folder named HDFS by using only the `mkdir` command without the `hdfs dfs` in front).

```
bitnami@debian:~$ hdfs dfs -mkdir HDFS
bitnami@debian:~$ ls
apps  bitnami_credentials  HDFS  htdocs  stack  testfile.txt
bitnami@debian:~$ hdfs dfs -ls
Found 1 items
drwxr-xr-x  - hadoop supergroup          0 2020-02-16 21:11 HDFS
```

While I could create a folder and a text file, I was not able to copy the `testfile.txt` file to HDFS using the `cp` or `put` commands. It tells me that the file exists, but the folder remains empty.

```
bitnami@debian:~$ hdfs dfs -cp testfile.txt /HDFS
cp: '/HDFS': File exists
bitnami@debian:~$ ls
apps  bitnami_credentials  HDFS  htdocs  stack  testfile.txt
bitnami@debian:~$ cd HDFS
bitnami@debian:~/HDFS$ ls
bitnami@debian:~/HDFS$
```

Since, I am using less than 8 GB, I need to do the following changes in the `yarn-site.xml` file, otherwise the wordcount sample will be stuck forever.

Locate the `yarn-site.xml` file through going to the Hadoop folder using the `cd` command.

```

link/ether 08:00:27:70:31:c8 brd ff:ff:ff:ff:ff:ff
inet 10.0.0.13/24 brd 10.0.0.255 scope global dynamic enp0s3
    valid_lft 83087sec preferred_lft 83087sec
inet6 fe80::a00:27ff:fe70:31c8/64 scope link
    valid_lft forever preferred_lft forever
bitnami@debian:~$ cd /opt/bitnami/hadoop/
bitnami@debian:/opt/bitnami/hadoop$ cd etc/hadoop/
bitnami@debian:/opt/bitnami/hadoop/etc/hadoop$ ls
capacity-scheduler.xml      kms-log4j.properties
configuration.xml           kms-site.xml
container-executor.cfg      log4j.properties
core-site.xml               mapred-env.cmd
hadoop-env.cmd              mapred-env.sh
hadoop-env.sh               mapred-queues.xml.template
hadoop-metrics2.properties mapred-site.xml
hadoop-policy.xml           shellprofile.d
hadoop-user-functions.sh.example  ssl-client.xml.example
hdfs-site.xml               ssl-server.xml.example
https-env.sh                user-ec_policies.xml.template
https-log4j.properties      workers
https-signature.secret      yarn-env.cmd
https-site.xml              yarn-env.sh
kms-acls.xml                 yarnservice-log4j.properties
kms-env.sh                   yarn-site.xml
bitnami@debian:/opt/bitnami/hadoop/etc/hadoop$ vi yarn

```

Edit the yarn-site.xml file: Replace the two values of 2048 with 8192. Save the file and restart the VM.

```

<!-- versions "1.0" -->
<configuration>
  <!-- BITNAMI DEFAULT CONFIGURATION -->
  <!-- Note: This section will be overwritten on server size changes -->

  <!-- The minimum allocation for every container request at the RM (default: 1024) -->
  <property>
    <name>yarn.scheduler.minimum-allocation-mb</name>
    <value>1024</value>
  </property>

  <!-- The maximum allocation for every container request at the RM (default: 8192) -->
  <property>
    <name>yarn.scheduler.maximum-allocation-mb</name>
    <value>2048</value>
  </property>

  <!-- Amount of physical memory, in MB, that can be allocated for containers (default: 8192) -->
  <property>
    <name>yarn.nodemanager.resource.memory-mb</name>
    <value>2048</value>
  </property>

  <!-- Amount of physical memory, in MB, that can be allocated for containers (default: 8192) -->
  <property>
    <name>yarn.nodemanager.resource.memory-mb</name>
    <value>2048</value>
  </property>

  <!-- END BITNAMI DEFAULT CONFIGURATION -->
</configuration>
"yarn-site.xml" 95L, 3366C                               1,1          Top

```

```

<!-- The minimum allocation for every container request at the RM (default: 1024) -->
<property>
  <name>yarn.scheduler.minimum-allocation-mb</name>
  <value>1024</value>
</property>

<!-- The maximum allocation for every container request at the RM (default: 8192) -->
<property>
  <name>yarn.scheduler.maximum-allocation-mb</name>
  <value>8192</value>
</property>

<!-- Amount of physical memory, in MB, that can be allocated for containers (default: 8192) -->
<property>
  <name>yarn.nodemanager.resource.memory-mb</name>
  <value>8192</value>
</property>
<!-- END BITNAMI DEFAULT CONFIGURATION -->

<!--
Licensed under the Apache License, Version 2.0 (the "License");
-- INSERT --                                21,16          6x

```

The VM was restarted and the yarn file checked to confirm the changes.

```

<!-- The minimum allocation for every container request at the RM (default: 1024) -->
<property>
  <name>yarn.scheduler.minimum-allocation-mb</name>
  <value>1024</value>
</property>

<!-- The maximum allocation for every container request at the RM (default: 8192) -->
<property>
  <name>yarn.scheduler.maximum-allocation-mb</name>
  <value>8192</value>
</property>

<!-- Amount of physical memory, in MB, that can be allocated for containers (default: 8192) -->
<property>
  <name>yarn.nodemanager.resource.memory-mb</name>
  <value>8192</value>
</property>
<!-- END BITNAMI DEFAULT CONFIGURATION -->

<!--
Licensed under the Apache License, Version 2.0 (the "License");
"yarn-site.xml" 95L, 3366C                               21,23          6x

```

Unfortunately, the mapreduce is not working.

```
bitnami@debian:~$ hadoop jar/opt/bitnami/hadoop/share/hadoop/mapreduce/hadoop/mapreduce-examples-3.2.1.jar wordcount /tmp/testfile.txt /tmp/out
/opt/bitnami/hadoop/libexec/hadoop-functions.sh: line 2366: HADOOP_JAR/OPT/BITNAMI/HADOOP/SHARE/HADOOP/MAPREDUCE/HADOOP/MAPREDUCE-EXAMPLES-3.2.1.JAR_USER: bad substitution
/opt/bitnami/hadoop/libexec/hadoop-functions.sh: line 2461: HADOOP_JAR/OPT/BITNAMI/HADOOP/SHARE/HADOOP/MAPREDUCE/HADOOP/MAPREDUCE-EXAMPLES-3.2.1.JAR_OPTS: bad substitution
Error: Could not find or load main class jar.opt.bitnami.hadoop.share.hadoop.mapreduce.hadoop.mapreduce-examples-3.2.1.jar
bitnami@debian:~$ _
```

Option 2: Azure HDInsight

I signed up for a 30-day free trial within Azure.

To create an HDInsight Azure Cluster, I searched for the resource HDInsight cluster and filled in the forms as follows. I specified a cluster name, a region (Germany West Central), selected Hadoop for cluster type and specified login username and password.

Microsoft Azure Search resources, services, and docs (G+/J)

Home > HDInsight clusters > Create HDInsight cluster

Create HDInsight cluster

[Go to classic create experience](#)

Cluster name * Christina ✓

Region * Germany West Central ✓

Cluster type * Hadoop
[Change](#)

Version * Hadoop 3.1.0 (HDI 4.0) ✓

Cluster credentials

Enter new credentials that will be used to administer or access the cluster.

Cluster login username * ⓘ admin ✓

Cluster login password * ✓

Confirm cluster login password * ✓

Secure Shell (SSH) username * ⓘ sshuser

[Review + create](#) [« Previous](#) [Next: Storage »](#)

Unfortunately, I failed to create the cluster because of subscription limits. I tried to change regions, the number of nodes as well as choice of cores available. But in any case, the cluster creation was aborted due to subscription limits.

Node type	Node size	Number of ...	Estimated cost
Head node	D12 v2 (4 Cores, 28 GB RAM), 0.39 EUR/hour	2	0.79 EUR
Worker node	D4 v2 (8 Cores, 28 GB RAM), 0.62 EUR/hour	1	0.62 EUR

You have reached your subscription's limit of -8 cores in Germany West Central. Please choose a data source in a different region or billing support to increase your limit for Germany West Central. The value must be between 1 and 0.

☐ Enable autoscale
(Preview) [Learn More](#)