# CSCI 5800 – Big Data Systems

## Prelab: Kafka and Stream Processing

Contents

[A. Installing Oracle VirtualBox 2](#_Toc511057386)

[B. Setting Up a Virtual Machine 2](#_Toc511057387)

[C. Configure the VM 4](#_Toc511057388)

[a. Motherboard Tab 4](#_Toc511057389)

[b. Processor Tab 4](#_Toc511057390)

[c. Displays: 5](#_Toc511057391)

[D. Installing Ubuntu 5](#_Toc511057392)

[E. Sharing Clipboards and Files 8](#_Toc511057393)

[F. Installing Python, Spark, and Jupyter Notebook 10](#_Toc511057394)

[a. Installing pip3 11](#_Toc511057395)

[b. Installing Jupyter Notebook 11](#_Toc511057396)

[c. Installing Spark 2.1 13](#_Toc511057397)

[G. Creating a Twitter Application Account 19](#_Toc511057398)

[F. Installing Kafka 21](#_Toc511057399)

[Installing the IntelliJ IDE 21](#_Toc511057400)

This document will guide you on the setup of an Ubuntu virtual machine and to install all necessary tools for our Kafka Lab.

# A. Installing Oracle VirtualBox

Download and install Oracle VirtualBox

<http://www.oracle.com/technetwork/server-storage/virtualbox/downloads/index.html>

# B. Setting Up a Virtual Machine

1. Download Ubuntu

<https://releases.ubuntu.com/20.04.3/ubuntu-20.04.3-desktop-amd64.iso?_ga=2.70947347.2057137009.1636568623-540270321.1636568623>

1. Open VirtualBox and create a new virtual machine (VM)

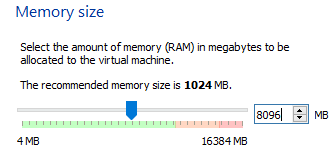
Graphical user interface, text, application

Description automatically generated

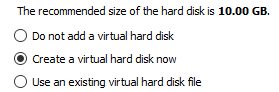
Graphical user interface, application

Description automatically generated

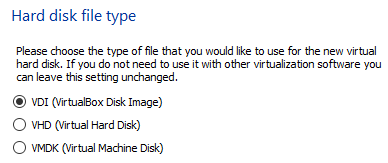
1. Allocate 4-8 GB of Ram (stay in the Green zone)

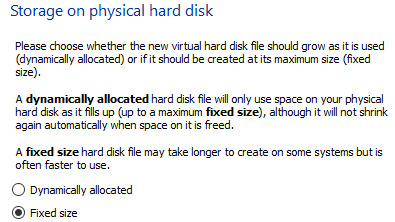


1. Create Virtual Hard Disk



1. Choose the VDI option



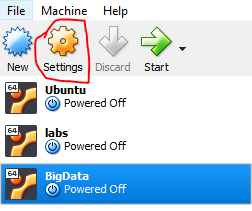
1. Choose the Fixed Size option (dynamically allocated option may slow down the VM)
2. Allocated 30-50GB of your disk

Graphical user interface, text, application

Description automatically generated

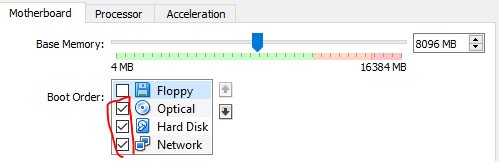
# C. Configure the VM

Choose the created VM and click on Settings



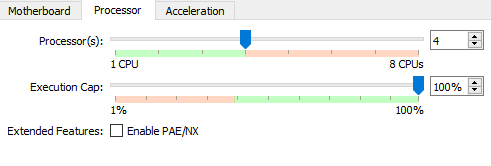
## Motherboard Tab

Make sure *Network* and *Optical* options are checked



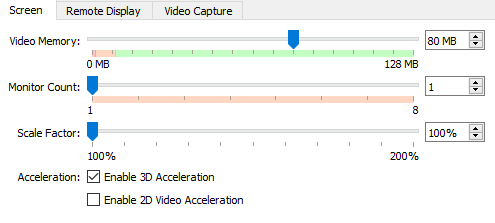
## Processor Tab

Set it to the max number of CPUs in the green zone:



## Displays:

1. More video memory will help things look a bit nicer



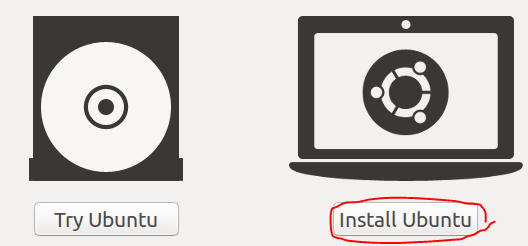
1. Enable 3D acceleration

# D. Installing Ubuntu

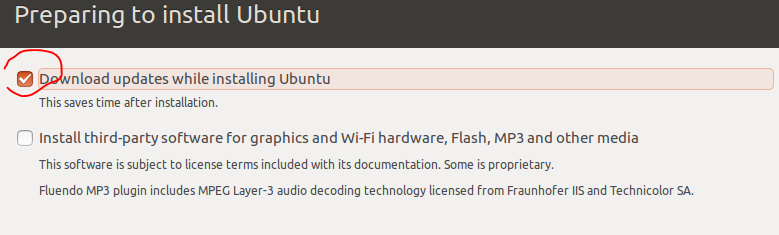
1. Double click on the created VM to start it and then select the Ubuntu iso file that you downloaded in Section B to start the installation process:

Graphical user interface, application, Word

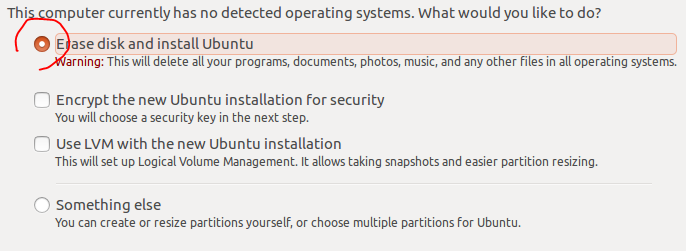
Description automatically generated



2. Download updates



1. Erase (only clears the memory that is allocated to the VM)



1. Do not forget your password

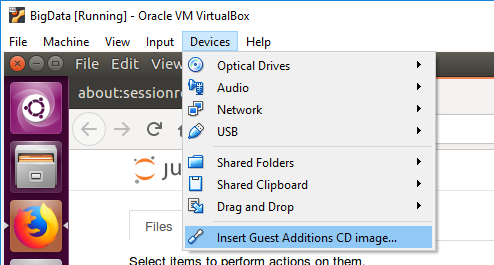
Set up username and password

1. Restart the VM
2. If it did not boot and stuck on a black terminal page for more than 2 minutes, restart the VM again.

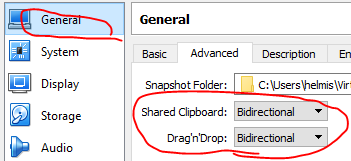
# E. Sharing Clipboards and Files

Performing the following steps will enable you to share files between your OS and the VM. You will be also to copy and paste clipboard contents between your OS and VM which will become handy during the installation process.

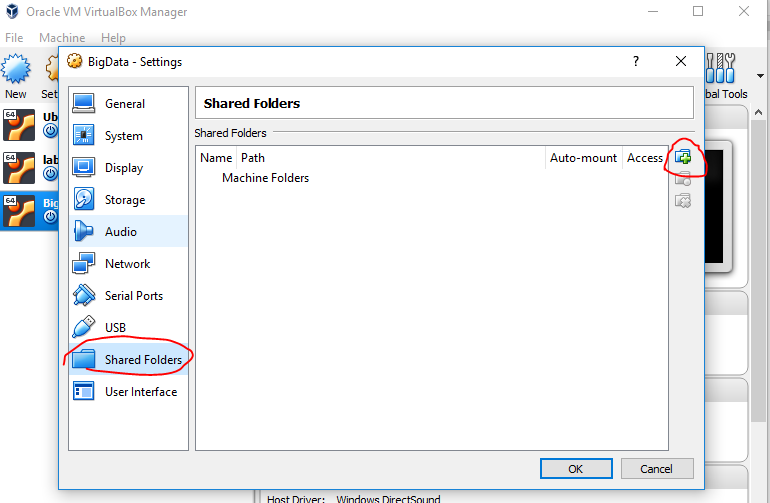
1. From the Device menu click on Insert Guest Additions CD Image. Click on Run button in the pop-up dialog. Wait until the installation is complete. Then power off the VM.



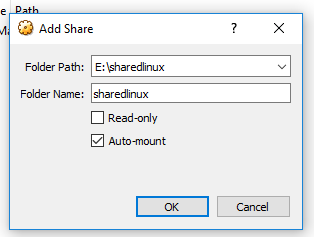
1. Open the VM setting on Oracle VM and go to the General tab first. Go to the Advanced tab and choose Bidirectional from both shared clipboard and drag’n’Drop dropdown menus.



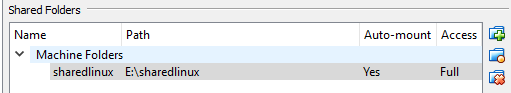
1. Then go to the Shared folder tab. Create a folder (directory) on your original OS which you would like to use as the shared folder. For example, I have created a folder named *sharedlinux* on my E drive.



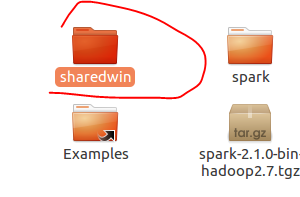
1. Then click on the add Shared Folder button on the right side (circled in the above image). Choose the folder you just created and check the Auto-mount box.



1. Click on OK.



1. Now boot the VM. Go to the Home directory and create a new folder. I named mine *sharedwin*.



Now open a terminal and execute the following command (if your folder names are different you need to modify them in the command; also, replace **khang** with your own username):

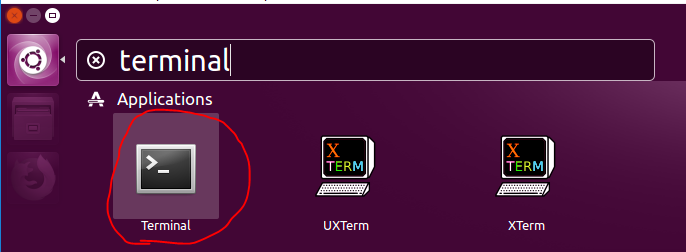
*sudo mount -t vboxsf sharedlinux /home/****khang****/sharedwin*

Text

Description automatically generated

Now, you should be able to copy and paste things off of your OS (VM) to the VM (OS). Also, you can share files by putting them into the shared folders. Note that you need to execute the above command every time that you log into your VM account. So, save it as a text file on your desktop!

# F. Installing Python, Spark, and Jupyter Notebook



Ubuntu comes with Python. To double check, open a terminal and use the following command:

*python3*

Text

Description automatically generated

Then use the *quit()* command to quit the Python shell.

## a. Installing pip3

*sudo apt install python3-pip*

Text

Description automatically generated

## b. Installing Jupyter Notebook

*pip3 install jupyter*

Graphical user interface

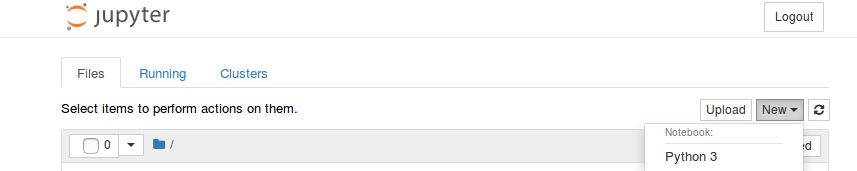
Description automatically generated with low confidence

Now open the Jupyter notebook using:

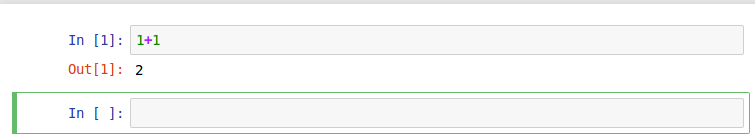
~/.local/bin/jupyter-notebook

When you run Jupyter Notebook for the first time, it generates a URL for you (as shown in the above picture). Copy and paste the generated URL in Firefox and press enter.

Create a new Python3 notebook.



Add something like 1+1 and press shift+enter . You should get an output



Go back to the terminal and shut down the Jupyter notebook by pressing ctrl+c

Close Firefox

## c. Installing Spark 2.1

Note that we have tested this lab with **Python 3.5** and **Spark 2.1** and it may not work with other versions. For example, the tweet reader application may not work with other versions of Spark. So, make sure that you use the same versions as described in this document.

**Installing Java:** go back to the terminal and type the following command to update the apt mechanism:

*sudo apt-get update*

Text

Description automatically generated

Then install Java by entering the following command:

*sudo apt-get install default-jre*

Text

Description automatically generated

make sure that Java is installed by using the following command:

*java -version*

Text

Description automatically generated

**Installing Scala:**

*sudo apt-get install scala*

Text

Description automatically generated

make sure that scala is installed using the following command:

*scala -version*

Text

Description automatically generated

**Installing PySpark:** This package connects Scala, Java, and Python

*pip3 install py4j*

Text

Description automatically generated

**Installing Spark:** Open Firefox on Ubuntu and go to the following link: <http://spark.apache.org/downloads.html>

To make sure everything will work, install Spark 2.1 (newer versions may not work with the instructions provided in the labs).

Graphical user interface, text, application, chat or text message

Description automatically generated

Once it is downloaded, go to the Downloads folder.

Graphical user interface, application

Description automatically generated

Cut the spark package and paste it to your home folder.

Graphical user interface

Description automatically generated

Then go back to terminal and type: *sudo tar -zxvf spark* and press tab and the command will be completed for you.

Text

Description automatically generated

Now, refresh the page and you will see an unzipped folder named spark-3.2.0-bin.hadoop3.2. Rename it to S*park*.

Graphical user interface

Description automatically generated with medium confidence

**Setting the system variables:**

Use the following command (replace ***khang*** with your own username)*:*

*export SPARK\_HOME=’home/* ***khang****/spark’*

*export PATH=$SPARK\_HOME:$PATH*

*export PYTHONPATH=$SPARK\_HOME/python:$PYTHONPATH*

Text

Description automatically generated

And link things together using:

*export PYSPARK\_DRIVER\_PYTHON=”jupyter”*

*export PYSPARK\_DRIVER\_PYTHON\_OPTS=”notebook”*

*export PYSPARK\_PYTHON=python3*

Text

Description automatically generated

**Fixing possible permission errors:**

*sudo chmod 777 spark*

Text

Description automatically generated

and the lock icon must be disappeared from the spark folder

Now we need to modify permissions to the python folder:



*cd ~*

*cd spark*

*sudo chmod 777 python*

Text

Description automatically generated

The lock icon must be disappeared from the Python folder.

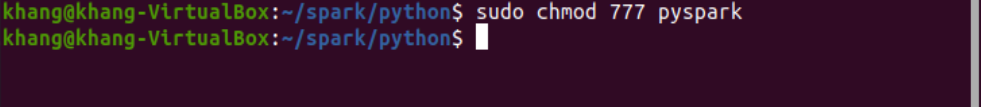


Do the same thing for the pyspark folder inside the python folder:

*cd python*

*sudo chmod 777 pyspark*





To make sure everything works:

*cd ~*

*cd spark*

*cd python*

*python3*

*import pyspark*

*quit()*

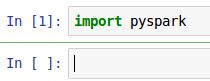
Text

Description automatically generated

now open Jupyter notebook again:

~/.local/bin/jupyter-notebook

create a new python3 notebook and import pyspark. You do not have to get any error messages:



Congratulations! You survived.

Now, every time that you want to work with Spark, you need to open the Jupyter notebook from the pyspark folder, i.e.:

*cd ~*

*cd spark*

*cd python*

~/.local/bin/jupyter-notebook

By installing the *findspark* package, you would be able to open the Jupyter notebook from anywhere and work with Spark (do not need to cd to the spark folder anymore):

*pip3 install findspark*

Text

Description automatically generated

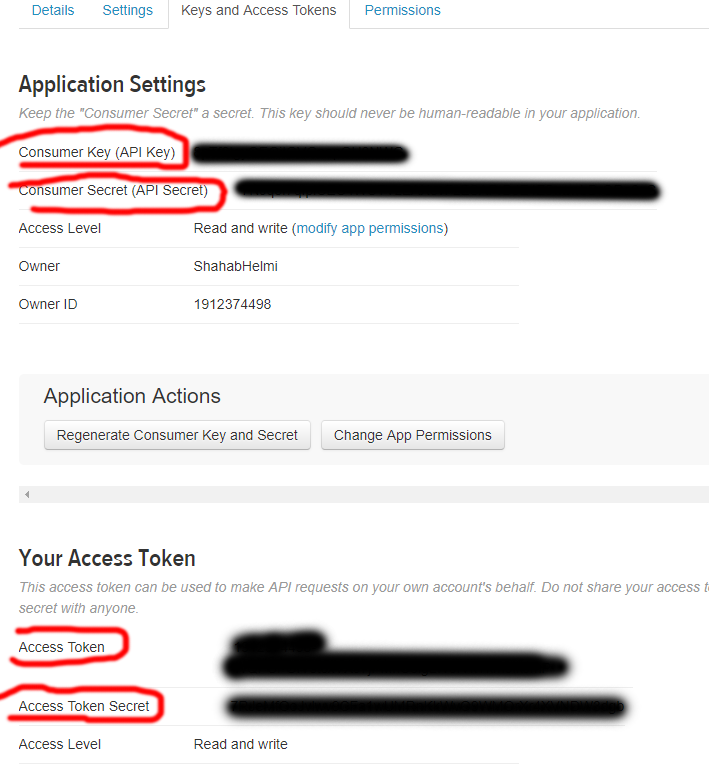
Now you can open the Jupyter notebook without going to the pyspark directory. You just need to add the following lines on top of your Jupyter notebook (do not forget to replace ***khang*** with your own username):

Graphical user interface, text, application, email

Description automatically generated

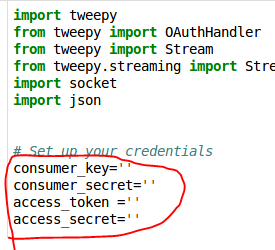
# G. Creating a Twitter Application Account

1. Go to <https://apps.twitter.com/> and create on Create New App. Fill out the requested information (you can enter [www.google.com](http://www.google.com) as your website if you do not have one).
2. When you are done, go to the Keys and Access Tokens tab. You will need the circled information for the streaming lab.

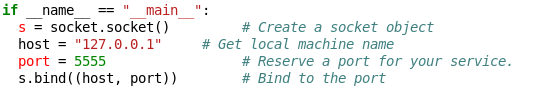


Create a folder in your home directory and name it *streaming*. Then download the streaming.zip package from Canvas and extract it in the streaming folder.

Open the tweetread.py file in the Jupyter notebook and enter your twitter information into the corresponding fields:



We are going to read tweets and send them to the port 5555. If this port is already busy for another application, use another port, like 5556:



Also, we tell the application to only receive tweets which contain the word **news** in them. You can replace it with your desired word, but make sure to choose a popular topic so you get enough tweets.



Use the following commands to install tweepy library (skip this if you are using the provided image). Note that before starting the lab you need to restart the VM.

*pip3 install tweepy*

# F. Installing Kafka

Go to this page and download Kafka (choose the binary version): <https://kafka.apache.org/downloads>

Then un-tar it:

*tar -xzf kafka\_2.13-3.0.0.tgz*

Rename the generated folder to *kafka* and move it to your home directory.

**Note**: whenever Kafka server does not start, you can simply fix it by removing the kafka folder and repeating the above step (your data will be lost).

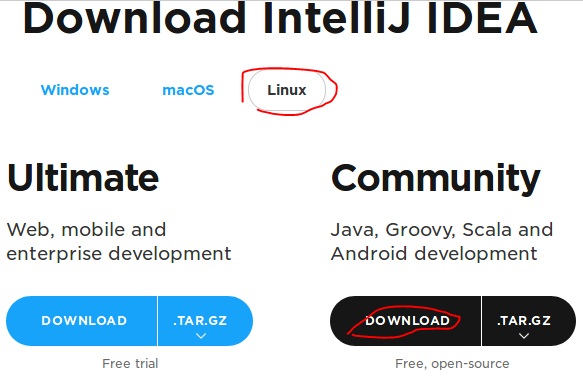
## Installing the IntelliJ IDE

First install Java Development Kit:

*sudo apt-get install openjdk-8-jdk*

Go to this page and download the Community version:

<https://www.jetbrains.com/idea/download/#section=linux>



Extract the download file using:

*tar -xzf ideaIC-2021.2.3.1.tar.gz*

Rename the downloaded folder to IntelliJ and move it to your home directory.

To open IntelliJ, open terminal and run the following commands;

*cd intellij*

*cd bin*

*./idea.sh*

Note that you can lock IntelliJ on taskbar, so you do not need to use the above commands to run IntelliJ every time.