# Spark Installation Instructions

Download at: https://spark.apache.org/downloads.html
Instructions + examples here: https://spark.apache.org/docs/latest/ and https://spark.apache.org/docs/latest/spark-standalone.html

## Apache Spark v3.1.3 Installation (Ubuntu 16 LTS terminal at Okeanos-knossos)

1. Setup your VM at https://cyclades.okeanos-knossos.grnet.gr/ui/#ips/ and attach the public IP to it
2. **Connect:** ssh user@snf-33040.ok-kno.grnetcloud.net (replace with your host name here and use your password)
3. **Install python3.8**
    a. sudo apt update
    b. sudo apt install build-essential zlib1g-dev libncurses5-dev libgdbm-dev libnss3-dev libssl-dev libreadline-dev libffi-dev libsqlite3-dev wget libbz2-dev
    c. wget https://www.python.org/ftp/python/3.8.0/Python-3.8.0.tgz
    d. tar -xf Python-3.8.0.tgz
    e. cd Python-3.8.0
    f. ./configure --enable-optimizations
    g. make -j 8
    h. sudo make altinstall
    i. python3.8 --version (you should expect: **Python 3.8.0**)
    j. Delete old links
    sudo rm -rf /usr/bin/python3.5
    sudo rm -rf /usr/bin/python3.5m
    sudo rm -rf /usr/lib/python3.5
    sudo rm -rf /etc/python3.5
    sudo rm -rf /usr/local/lib/python3.5
4. **Install pip**
    a. cd ../
    b. wget https://bootstrap.pypa.io/get-pip.py
    c. python3.8 get-pip.py
5. **Install PySpark**
    a. pip3.8 install pyspark==3.1.3
6. **Install Apache Spark**
    a. wget https://downloads.apache.org/spark/spark-3.1.3/spark-3.1.3-bin-hadoop2.7.tgz
    b. tar -xzf spark-3.1.3-bin-hadoop2.7.tgz
    c. nano ~/.bashrc
    export SPARK_HOME=/home/user/spark-3.1.3-bin-hadoop2.7

export PATH=$PATH:$SPARK_HOME/sbin
export PYSPARK_PYTHON=python3.8
export PYSPARK_DRIVER_PYTHON=python3.8

d. source ~/.bashrc

7. **Install Java**
    a. sudo apt-get install openjdk-8-jdk
    b. java -version (you should expect: **openjdk version "1.8.0_292"**)
8. **Setup a Cluster (1 master and 1 worker)**
    a. Create a network at Okeanos and assign IPs to each VM
    b. cd spark-3.1.3-bin-hadoop2.7/conf
    c. touch spark-env.sh
    d. nano spark-env.sh
    e. `SPARK_MASTER_HOST='192.168.0.2'`
    f. start-master.sh

**Deploy Workers - Custom resources (ports must be between 1024 - 65535)**

- spark-daemon.sh start org.apache.spark.deploy.worker.Worker 1 --webui-port 8080 --port 65509 --cores 4 --memory 8g spark://192.168.0.2:7077
- spark-daemon.sh start org.apache.spark.deploy.worker.Worker 2 --webui-port 8080 --port 65510 --cores 4 --memory 8g spark://192.168.0.2:7077

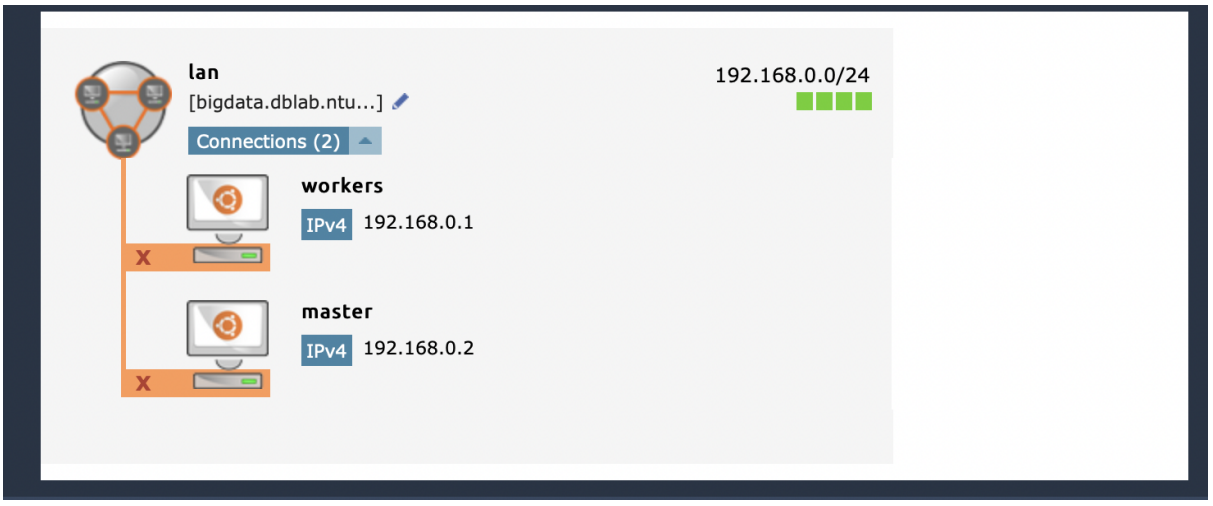
**Deploy 1 worker per VM**

- start-worker.sh spark://192.168.0.2:7077

Go to http://83.212.80.9:8080/ (use your Public IP here)

Create a second VM and repeat the steps **3**, **6** and **7**.

- spark-daemon.sh start org.apache.spark.deploy.worker.Worker 3 --webui-port 8080 --port 65511 --cores 2 --memory 4g spark://192.168.0.2:7077
- spark-daemon.sh start org.apache.spark.deploy.worker.Worker 4 --webui-port 8080 --port 65512 --cores 2 --memory 4g spark://192.168.0.2:7077



# Spark Master at spark://192.168.0.2:7077

**URL:** spark://192.168.0.2:7077
**Alive Workers:** 4
**Cores in use:** 12 Total, 0 Used
**Memory in use:** 24.0 GiB Total, 0.0 B Used
**Resources in use:**
**Applications:** 0 Running, 0 Completed
**Drivers:** 0 Running, 0 Completed
**Status:** ALIVE

## Workers (4)

| Worker Id | Address | State | Cores | Memory | Resources |
|---|---|---|---|---|---|
| worker-20221119235442-192.168.0.2-65509 | 192.168.0.2:65509 | ALIVE | 4 (0 Used) | 8.0 GiB (0.0 B Used) | |
| worker-20221119235511-192.168.0.2-65510 | 192.168.0.2:65510 | ALIVE | 4 (0 Used) | 8.0 GiB (0.0 B Used) | |
| worker-20221119235545-192.168.0.1-65511 | 192.168.0.1:65511 | ALIVE | 2 (0 Used) | 4.0 GiB (0.0 B Used) | |
| worker-20221119235557-192.168.0.1-65512 | 192.168.0.1:65512 | ALIVE | 2 (0 Used) | 4.0 GiB (0.0 B Used) | |