Q11: MapReduce on Ubuntu

Hadoop Installation on Google Cloud Platform (GCP) Compute Engine

Note: This document is mainly about “Hadoop: Setting up a Single Node Cluster”.

Reference Link: [Apache Hadoop 3.3.4 – Hadoop: Setting up a Single Node Cluster.]

1. Prerequisites
   Cloud Platform: GCP
   OS Platform: Ubuntu 18.04 LTS
   Requisite Software: Java 8, ssh, sshd, pdsh (Recommended)
   Hadoop Version: Apache Hadoop 3.3.4 (which is available by this link [Apache Downloads])

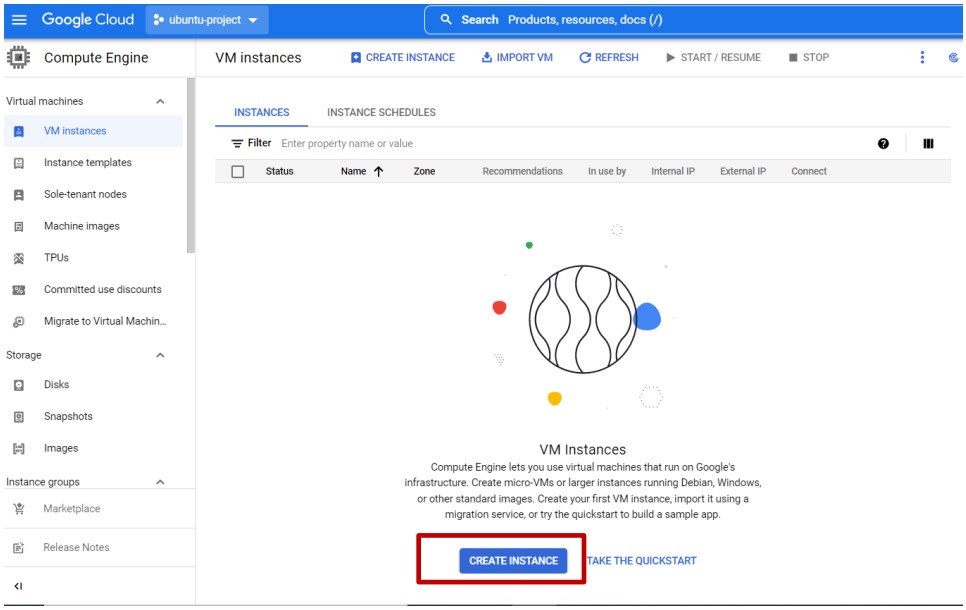
Hadoop Java Versions

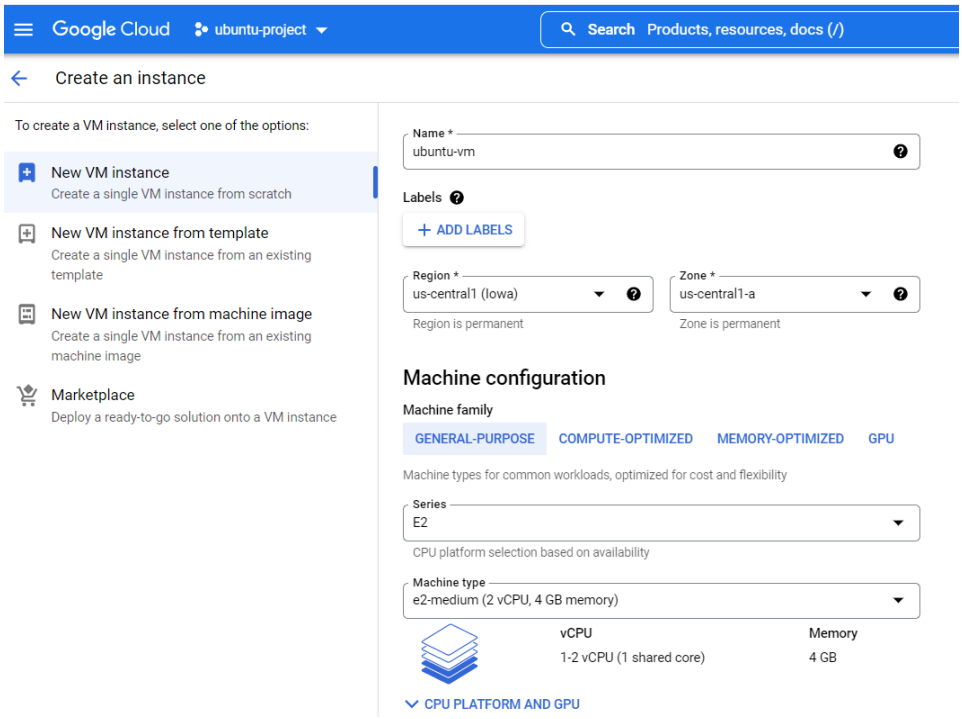
Created by Akira Ajisaka, last modified on Oct 19, 2020

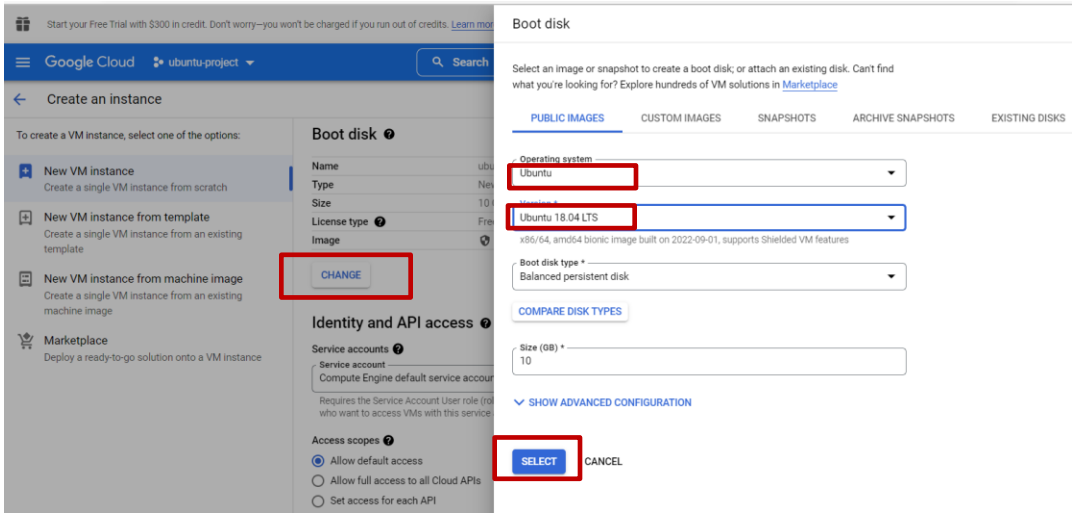
Supported Java Versions

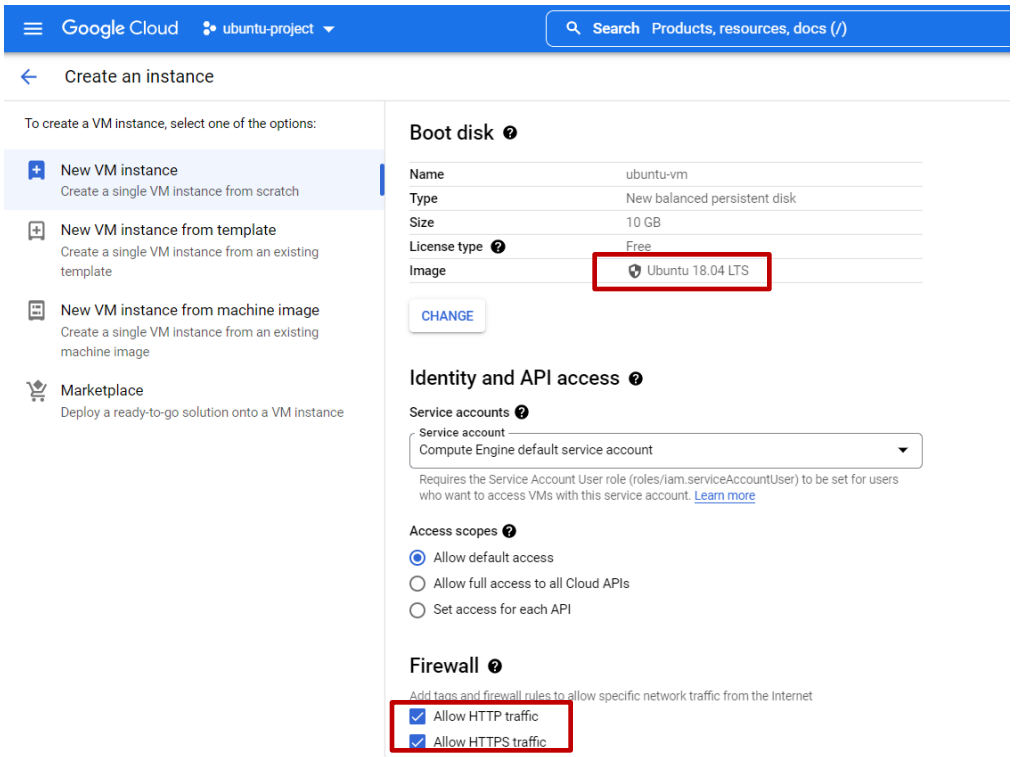
- Apache Hadoop 3.3 and upper supports Java 8 and Java 11 (runtime only)
  - Please compile Hadoop with Java 8. Compiling Hadoop with Java 11 is not supported:
    HADOOP-16795 - Java 11 compile support OPEN
- Apache Hadoop from 3.0.x to 3.2.x now supports only Java 8
- Apache Hadoop from 2.7.x to 2.10.x support both Java 7 and 8

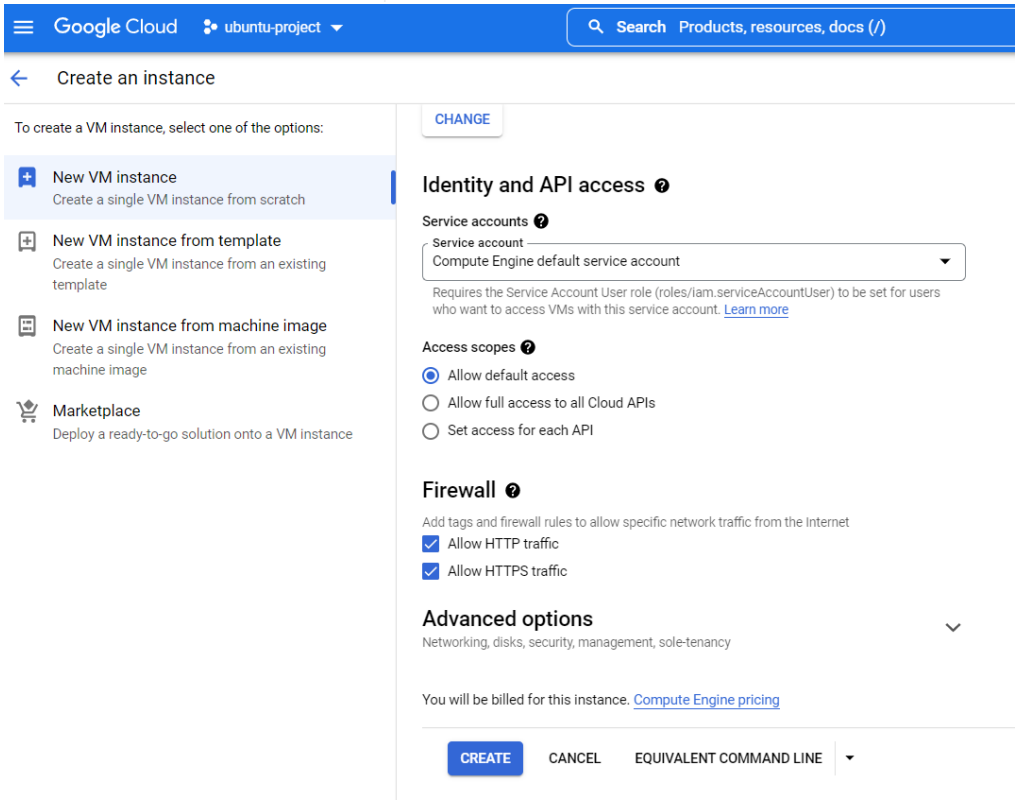
2. Hands-on Practice
   1> Provision a Linux server virtual machine with ubuntu OS
   (Reference link [014] Provisioning a Linux server using Google Cloud Platform - YouTube
   [017] Connecting to a Google VM with a user provided SSH key - YouTube)
   GCP → Dashboard page -> ubuntu-project
   Navigation Menu → Compute Engine → Virtual Instances → CREATE INSTANCE

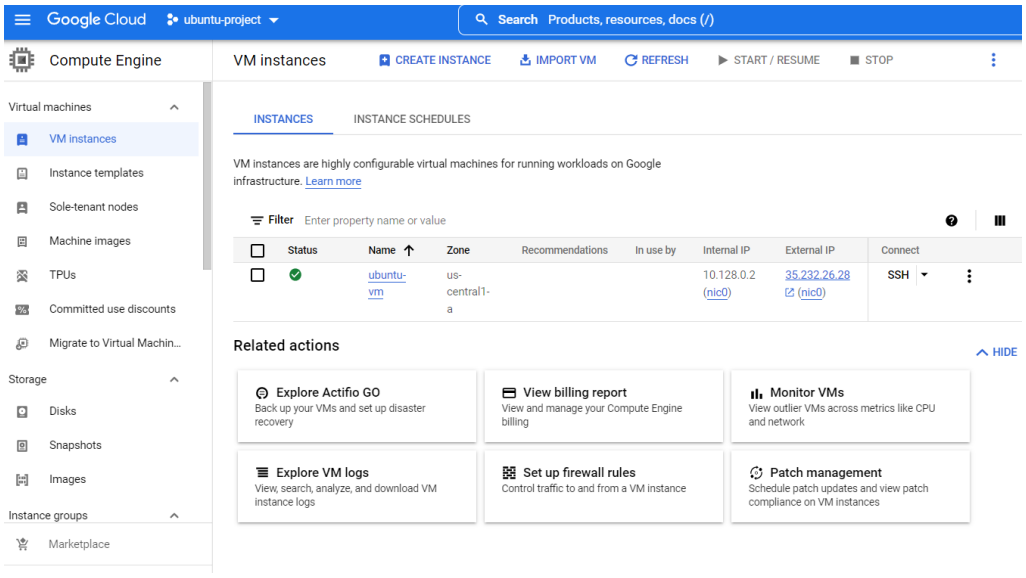



Google Cloud
ubuntu-project
Search Products, resources, docs (/)
Create an instance
To create a VM instance, select one of the options:
New VM instance
Create a single VM instance from scratch
New VM instance from template
Create a single VM instance from an existing template
New VM instance from machine image
Create a single VM instance from an existing machine image
Marketplace
Deploy a ready-to-go solution onto a VM instance
Name *
ubuntu-vm
Labels
ADD LABELS
Region *
us-central1 (Iowa)
Region is permanent
Zone *
us-central1-a
Zone is permanent
Machine configuration
Machine family
GENERAL-PURPOSE
COMPUTE-OPTIMIZED
MEMORY-OPTIMIZED
GPU
Machine types for common workloads, optimized for cost and flexibility
Series
E2
CPU platform selection based on availability
Machine type
e2-medium (2 vCPU, 4 GB memory)
vCPU
1-2 vCPU (1 shared core)
Memory
4 GB
CPU PLATFORM AND GPU

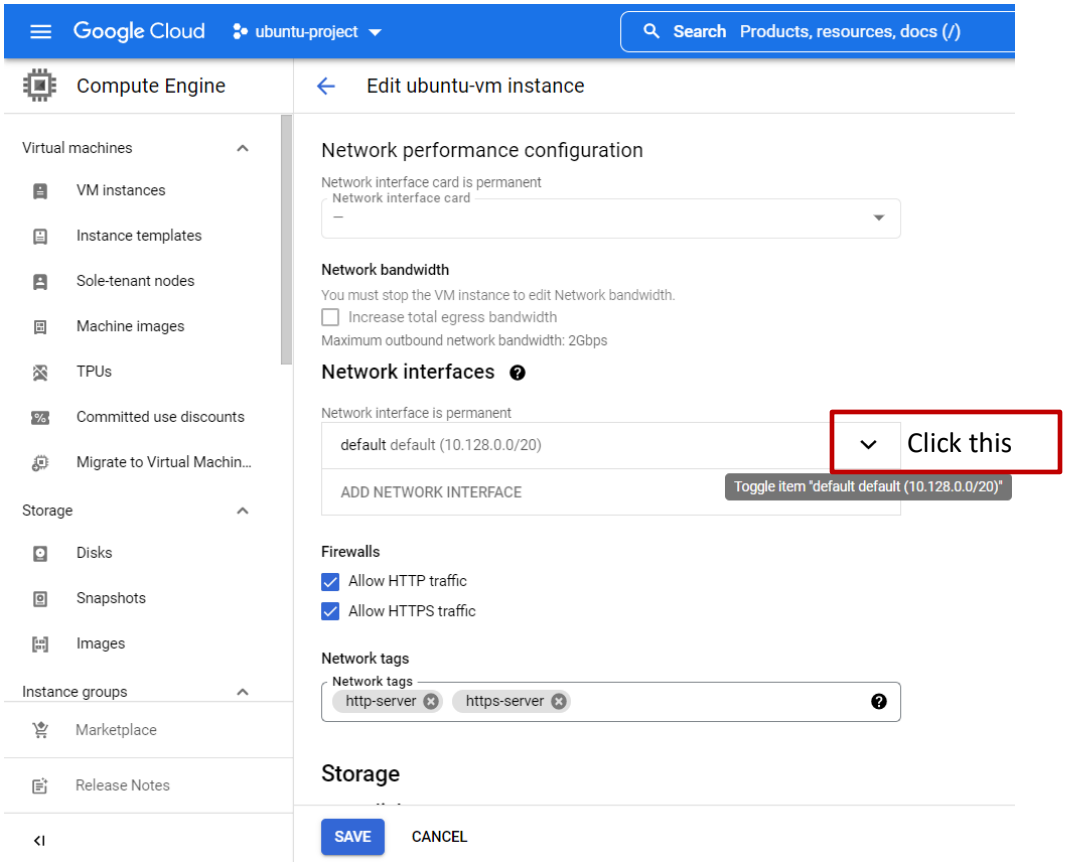

Start your Free Trial with $300 in credit. Don't worry—you won't be charged if you run out of credits.
Google Cloud
ubuntu-project
Search
Create an instance
To create a VM instance, select one of the options:
New VM instance
Create a single VM instance from scratch
New VM instance from template
Create a single VM instance from an existing template
New VM instance from machine image
Create a single VM instance from an existing machine image
Marketplace
Deploy a ready-to-go solution onto a VM instance
Boot disk
Name
Type
Size
License type
Image
CHANGE
Identity and API access
Service accounts
Service account
Compute Engine default service account
Access scopes
Allow default access
Allow full access to all Cloud APIs
Set access for each API
Boot disk
Select an image or snapshot to create a boot disk; or attach an existing disk. Can't find what you're looking for? Explore hundreds of VM solutions in Marketplace
PUBLIC IMAGES
CUSTOM IMAGES
SNAPSHOTS
ARCHIVE SNAPSHOTS
EXISTING DISKS
Operating system
Ubuntu
Ubuntu 18.04 LTS
x86/64, amd64 bionic image built on 2022-09-01, supports Shielded VM features
Boot disk type *
Balanced persistent disk
COMPARE DISK TYPES
Size (GB) *
10
SHOW ADVANCED CONFIGURATION
SELECT
CANCEL

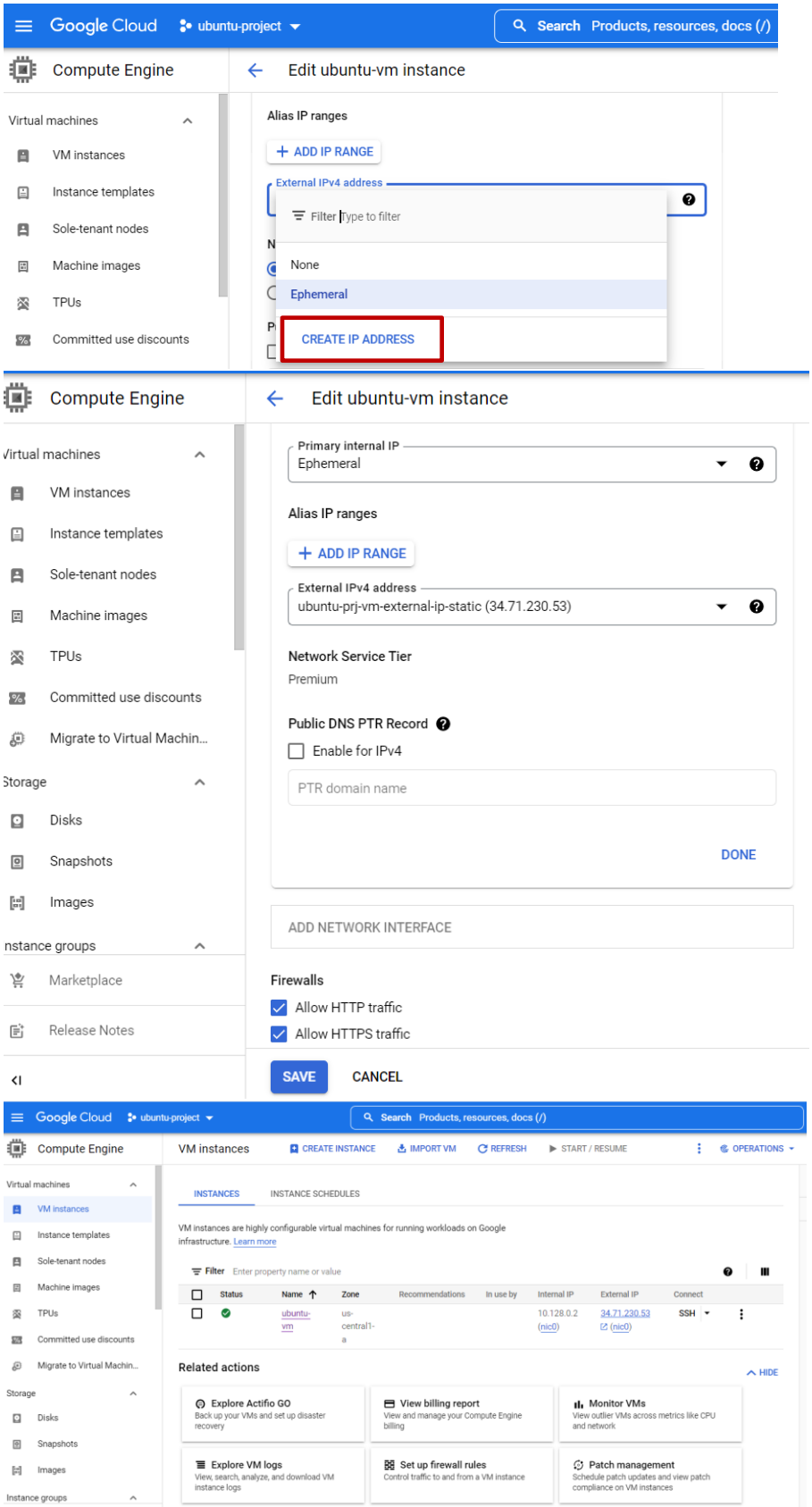

Google Cloud
ubuntu-project
Search Products, resources, docs (/)
Create an instance
To create a VM instance, select one of the options:
New VM instance
Create a single VM instance from scratch
New VM instance from template
Create a single VM instance from an existing template
New VM instance from machine image
Create a single VM instance from an existing machine image
Marketplace
Deploy a ready-to-go solution onto a VM instance
Boot disk
Name ubuntu-vm
Type New balanced persistent disk
Size 10 GB
License type Free
Image Ubuntu 18.04 LTS
CHANGE
Identity and API access
Service accounts
Service account
Compute Engine default service account
Requires the Service Account User role (roles/iam.serviceAccountUser) to be set for users who want to access VMs with this service account. Learn more
Access scopes
Allow default access
Allow full access to all Cloud APIs
Set access for each API
Firewall
Add tags and firewall rules to allow specific network traffic from the Internet
Allow HTTP traffic
Allow HTTPS traffic


Google Cloud
ubuntu-project
Search Products, resources, docs (/)
Create an instance
To create a VM instance, select one of the options:
New VM instance
Create a single VM instance from scratch
New VM instance from template
Create a single VM instance from an existing template
New VM instance from machine image
Create a single VM instance from an existing machine image
Marketplace
Deploy a ready-to-go solution onto a VM instance
CHANGE
Identity and API access
Service accounts
Service account
Compute Engine default service account
Requires the Service Account User role (roles/iam.serviceAccountUser) to be set for users who want to access VMs with this service account. Learn more
Access scopes
Allow default access
Allow full access to all Cloud APIs
Set access for each API
Firewall
Add tags and firewall rules to allow specific network traffic from the Internet
Allow HTTP traffic
Allow HTTPS traffic
Advanced options
Networking, disks, security, management, sole-tenancy
You will be billed for this instance. Compute Engine pricing
CREATE
CANCEL
EQUIVALENT COMMAND LINE

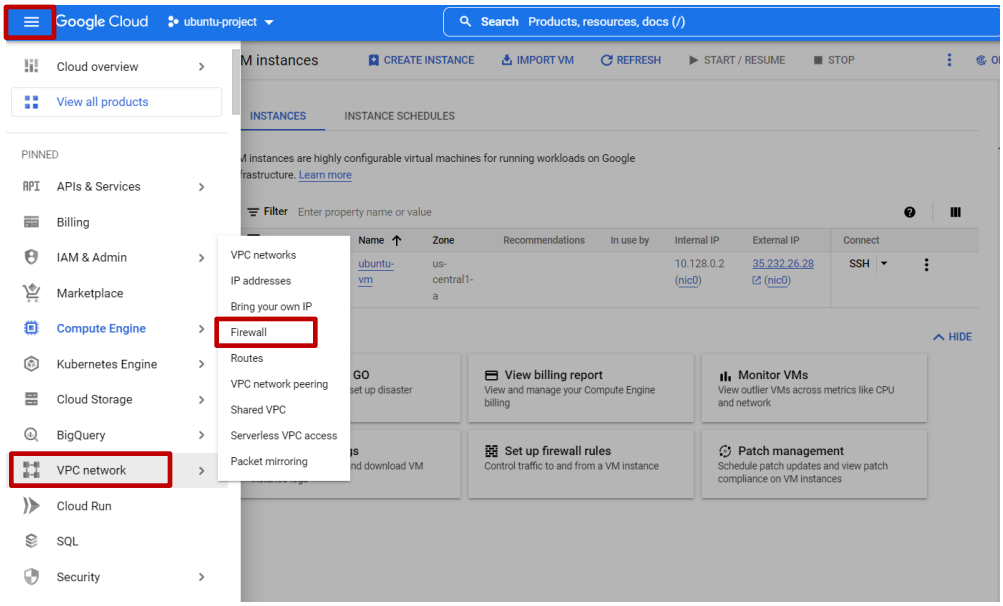


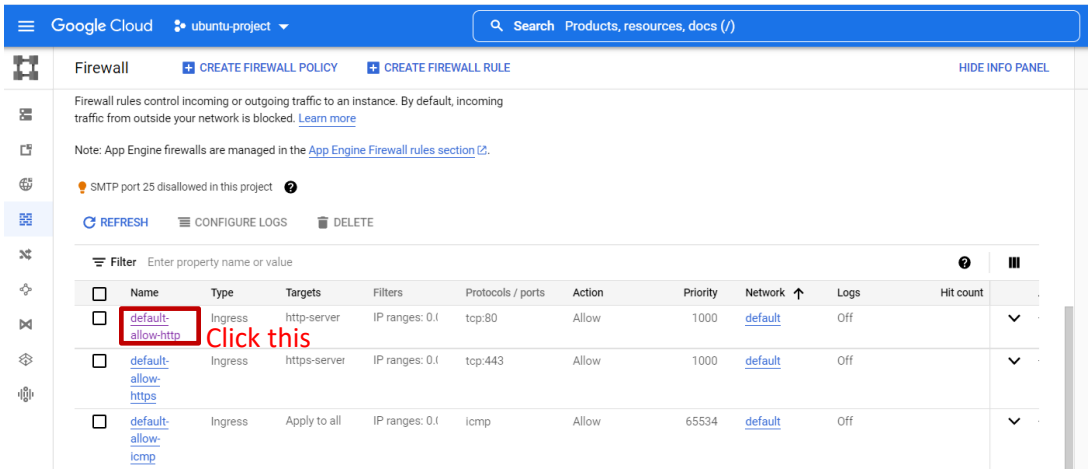
Set static external IP address

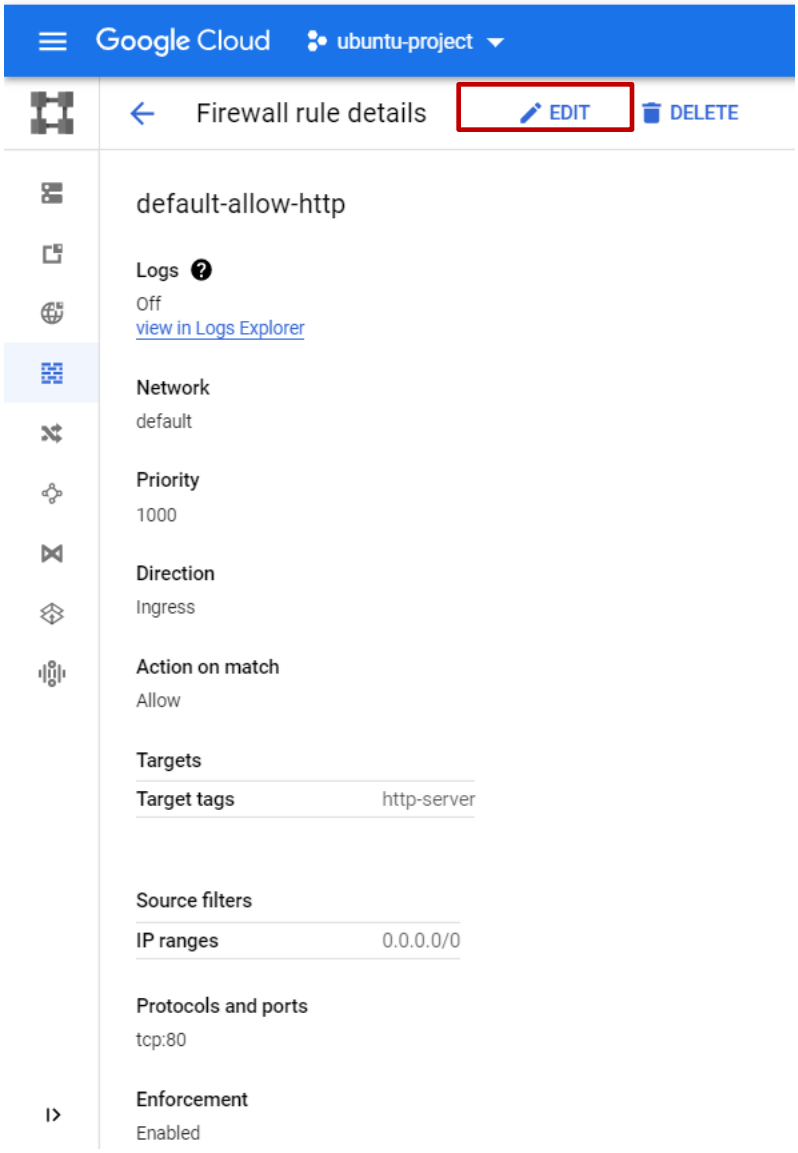


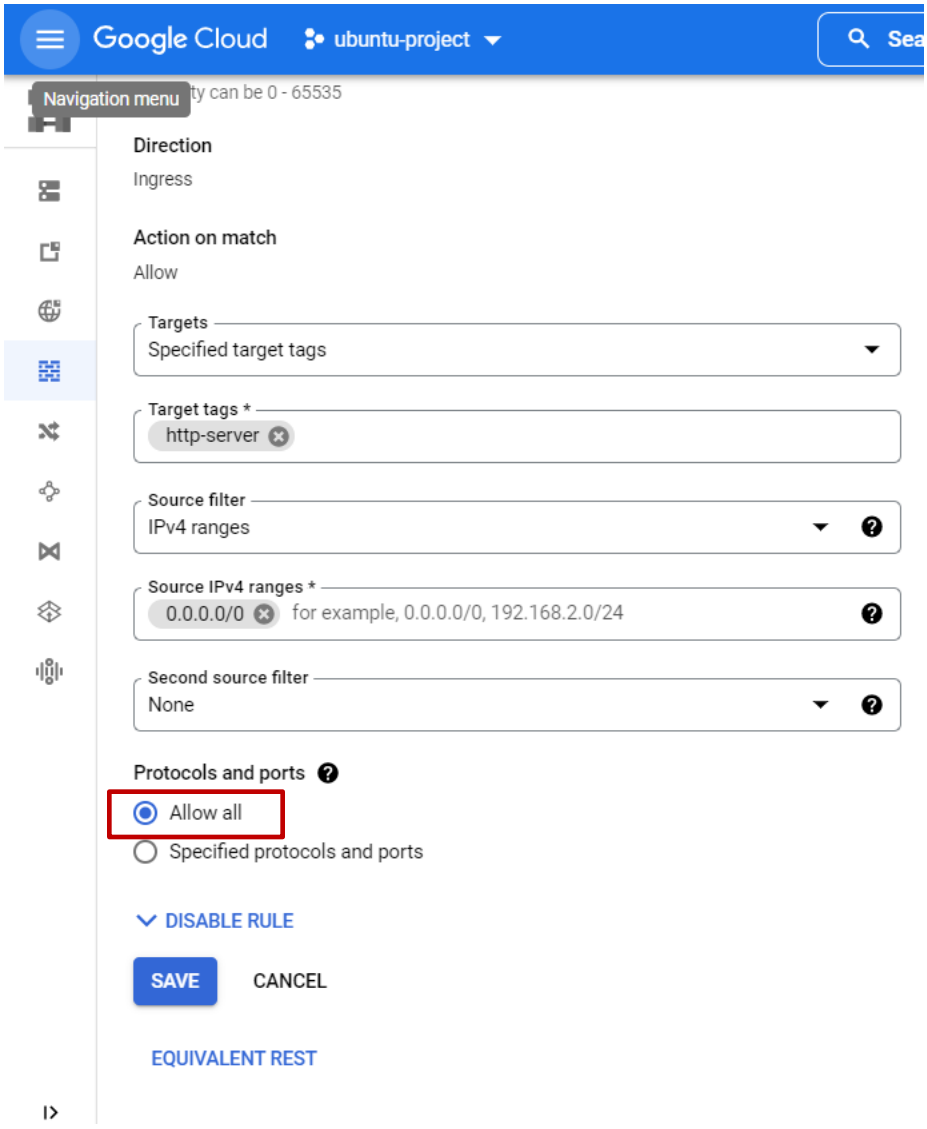

Google Cloud
ubuntu-project
Search Products, resources, docs (/)
Compute Engine
Edit ubuntu-vm instance
Virtual machines
VM instances
Instance templates
Sole-tenant nodes
Machine images
TPUs
Committed use discounts
Alias IP ranges
ADD IP RANGE
External IPv4 address
Filter Type to filter
None
Ephemeral
CREATE IP ADDRESS
Compute Engine
Edit ubuntu-vm instance
Virtual machines
VM instances
Instance templates
Sole-tenant nodes
Machine images
TPUs
Committed use discounts
Migrate to Virtual Machin...
Storage
Disks
Snapshots
Images
Instance groups
Marketplace
Release Notes
Primary internal IP
Ephemeral
Alias IP ranges
ADD IP RANGE
External IPv4 address
ubuntu-prj-vm-external-ip-static (34.71.230.53)
Network Service Tier
Premium
Public DNS PTR Record
Enable for IPv4
PTR domain name
DONE
ADD NETWORK INTERFACE
Firewalls
Allow HTTP traffic
Allow HTTPS traffic
SAVE
CANCEL
Google Cloud
ubuntu-project
Search Products, resources, docs (/)
Compute Engine
VM instances
CREATE INSTANCE
IMPORT VM
REFRESH
START / RESUME
OPERATIONS
INSTANCES
INSTANCE SCHEDULES
VM instances are highly configurable virtual machines for running workloads on Google infrastructure. Learn more
Filter Enter property name or value
Status
Name
Zone
Recommendations
In use by
Internal IP
External IP
Connect
ubuntu-vm
us-central1-a
10.128.0.2 (nic0)
34.71.230.53 (nic0)
SSH
Related actions
HIDE
Explore Actifio GO
Back up your VMs and set up disaster recovery
View billing report
View and manage your Compute Engine billing
Monitor VMs
View outlier VMs across metrics like CPU and network
Explore VM logs
View, search, analyze, and download VM instance logs
Set up firewall rules
Control traffic to and from a VM instance
Patch management
Schedule patch updates and view patch compliance on VM instances

Next step, if you want to check the Hadoop HDFS, NameNode and ResourceManger information through browser later, do this step to make sure the firewall allow all protocol & ports for http server, otherwise, skip this step.
Click Navigation Menu → VPC network → Firewall

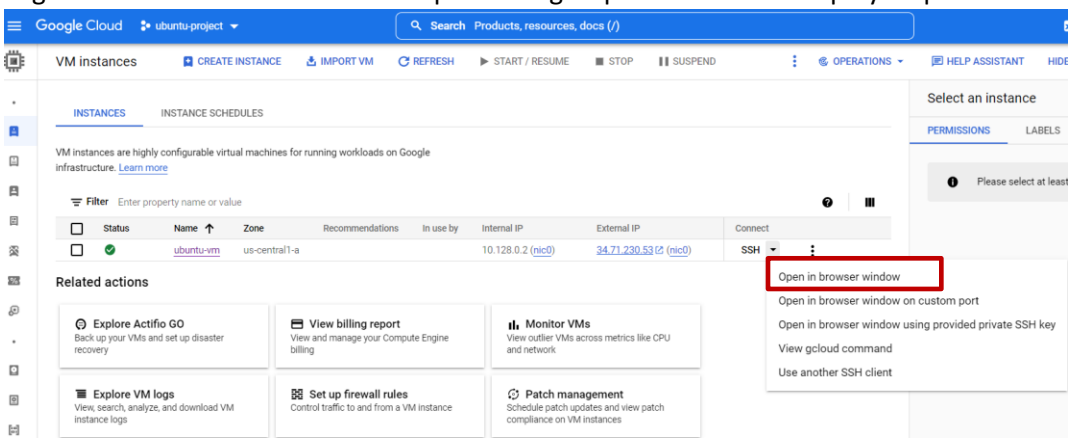





Google Cloud
ubuntu-project
Firewall rule details
EDIT
DELETE
default-allow-http
Logs
Off
view in Logs Explorer
Network
default
Priority
1000
Direction
Ingress
Action on match
Allow
Targets
Target tags
http-server
Source filters
IP ranges
0.0.0.0/0
Protocols and ports
tcp:80
Enforcement
Enabled



2> Login in ubuntu VM to install Hadoop including requisite software step by step



Install java 8:

$ sudo apt-get update

$ sudo apt-get install openjdk-8-jdk

$ java -version

This command will display the java version info if installation is successful

$ sudo update-alternatives –config java

To know the java path, write down this path for later path setting

```
xwu@ubuntu-vm:~$ java -version
openjdk version "1.8.0_342"
OpenJDK Runtime Environment (build 1.8.0_342-8u342-b07-0ubuntu1~18.04-b07)
OpenJDK 64-Bit Server VM (build 25.342-b07, mixed mode)
xwu@ubuntu-vm:~$ sudo update-alternative --config java
sudo: update-alternative: command not found
xwu@ubuntu-vm:~$ sudo update-alternatives --config java
There is only one alternative in link group java (providing /usr/bin/java): /usr/lib/jvm/java-8-openjdk-amd64/jr
e/bin/java
Nothing to configure.
```

Check if ssh/sshd/pdsh exists already, if not, install them:

$ which ssh

$ which sshd

$ which pdsh

Install ssh & pdsh:

$ sudo apt-get install ssh

$ sudo apt-get install pdsh

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ which ssh
/usr/bin/ssh
xwu@ubuntu-vm:~/hadoop-3.3.4$ which sshd
/usr/sbin/sshd
xwu@ubuntu-vm:~/hadoop-3.3.4$ sudo apt-get install pdsh
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  libnuma1
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  genders libgenders0 libltdl7
Suggested packages:
  rdist
The following NEW packages will be installed:
  genders libgenders0 libltdl7 pdsh
0 upgraded, 4 newly installed, 0 to remove and 29 not upgraded.
Need to get 209 kB of archives.
After this operation, 906 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic/universe amd64 lib
3 kB]
Get:2 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic/universe amd64 ger
]
```

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ which pdsh
/usr/bin/pdsh
```

Download Hadoop 3.3.4, and unpack them.

$ wget https://dlcdn.apache.org/hadoop/common/hadoop-3.3.4/hadoop-3.3.4.tar.gz

$ tar xvzf hadoop-3.3.4.tar.gz

```
xwu@ubuntu-vm:~$ wget https://dlcdn.apache.org/hadoop/common/hadoop-3.3.4/hadoop-3.3.4.tar.gz
--2022-09-30 04:15:21--  https://dlcdn.apache.org/hadoop/common/hadoop-3.3.4/hadoop-3.3.4.tar.gz
Resolving dlcdn.apache.org (dlcdn.apache.org)... 151.101.2.132, 2a04:4e42::644
Connecting to dlcdn.apache.org (dlcdn.apache.org)|151.101.2.132|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 695457782 (663M) [application/x-gzip]
Saving to: 'hadoop-3.3.4.tar.gz'

hadoop-3.3.4.tar.gz         100%[=========================================>] 663.24M   213MB/s    in 3

2022-09-30 04:15:24 (210 MB/s) - 'hadoop-3.3.4.tar.gz' saved [695457782/695457782]

xwu@ubuntu-vm:~$ tar xvzf hadoop-3.3.4.tar.gz
```

Set java path for Hadoop:
$ cd hadoop-3.3.4/
$ vi etc/Hadoop/hadoop-env.sh
Add " export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java"

Set other path/variables for Hadoop
$ vi ~/.bashrc
Add the following lines:
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java
export PDSH_RCMD_TYPE=ssh
export PATH=${JAVA_HOME}/bin:${PATH}
export HADOOP_CLASSPATH=${JAVA_HOME}/lib/tools.jar
$ source ~/.bashrc

Test the Hadoop command:
$ bin/Hadoop
This will display the usage documentation for the Hadoop script if Hadoop is installed successfully.

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hadoop
Usage: hadoop [OPTIONS] SUBCOMMAND [SUBCOMMAND OPTIONS]
 or    hadoop [OPTIONS] CLASSNAME [CLASSNAME OPTIONS]
  where CLASSNAME is a user-provided Java class

  OPTIONS is none or any of:

--config dir                     Hadoop config directory
--debug                          turn on shell script debug mode
--help                           usage information
buildpaths                       attempt to add class files from build tree
hostnames list[,of,host,names]   hosts to use in slave mode
hosts filename                   list of hosts to use in slave mode
loglevel level                   set the log4j level for this command
workers                          turn on worker mode

  SUBCOMMAND is one of:


    Admin Commands:

daemonlog     get/set the log level for each daemon

    Client Commands:

archive       create a Hadoop archive
checknative   check native Hadoop and compression libraries availability
classpath     prints the class path needed to get the Hadoop jar and the required libraries
conftest      validate configuration XML files
credential    interact with credential providers
distch        distributed metadata changer
distcp        copy file or directories recursively
```

Now, we could run some test examples in different operations according to this link Apache Hadoop 3.3.4 – Hadoop: Setting up a Single Node Cluster.

3> Standalone Operation

By default, Hadoop is configured to run in a non-distributed mode, as a single Java process. This is useful for debugging.

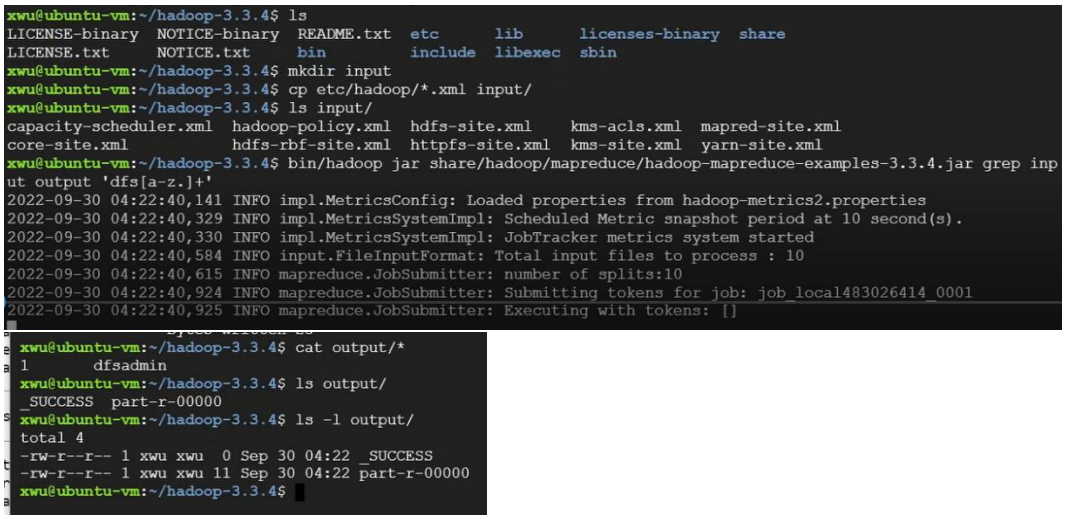
The following example copies the unpacked conf directory to use as input and then finds and displays every match of the given regular expression. Output is written to the given output directory.

$ mkdir input

$ cp etc/hadoop/*.xml input

$ bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.4.jar grep input output 'dfs[a-z.]+'

$ cat output/*



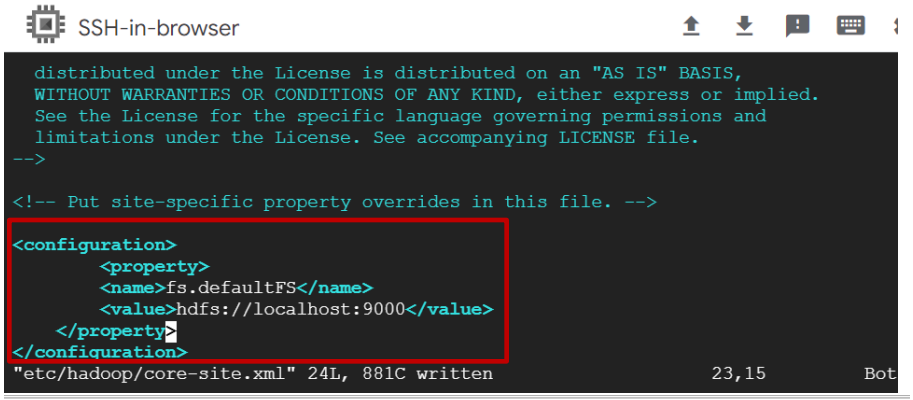
4> Pseudo-Distributed Operation

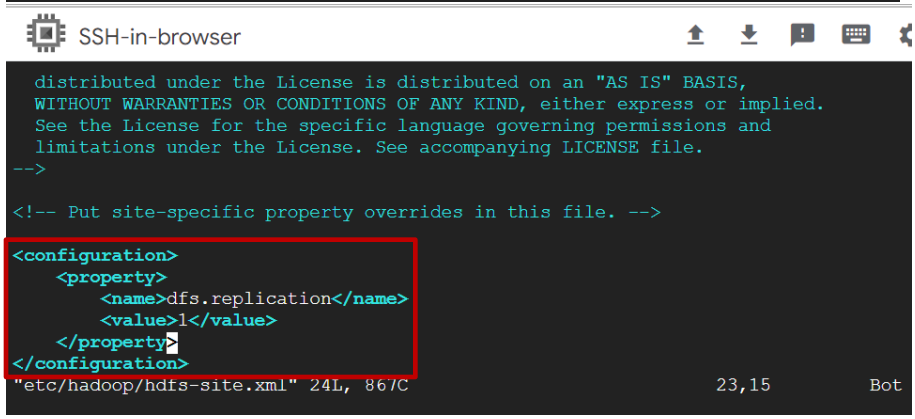
Hadoop can also be run on a single-node in a pseudo-distributed mode where each Hadoop daemon runs in a separate Java process.

Configure parameters:

$ vi etc/hadoop/core-site.xml

$ vi etc/hadoop/hdfs-site.xml





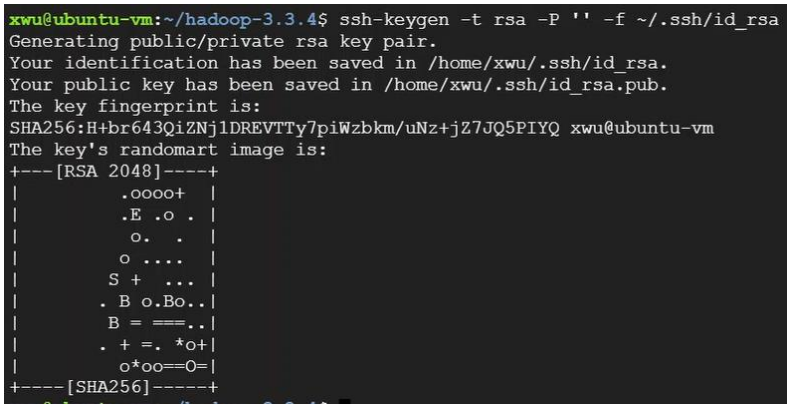
ssh passphraseless setting:

$ ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa

$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys

$ chmod 0600 ~/.ssh/authorized_keys $ ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa

$ ssh localhost



Note: In the whole process, make sure that “ssh localhost” has been executed. Otherwise, we could NOT run Hadoop scripts successfully.

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ ssh localhost
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-1087-gcp x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Fri Sep 30 04:28:46 UTC 2022

  System load:  0.0               Processes:           109
  Usage of /:   45.9% of 9.51GB   Users logged in:     1
  Memory usage: 9%                IP address for ens4: 10.128.0.2
  Swap usage:   0%


29 updates can be applied immediately.
29 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

New release '20.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


Last login: Fri Sep 30 03:37:45 2022 from 35.235.244.32
```

The following instructions are to run a MapReduce job locally.

a. Format file system

   $ bin/hdfs namenode -format

```
xwu@ubuntu-vm:~$ bin/hdfs namenode -format
-bash: bin/hdfs: No such file or directory
xwu@ubuntu-vm:~$ bin/hdfs namenode -format
-bash: bin/hdfs: No such file or directory
xwu@ubuntu-vm:~$ cd hadoop-3.3.4/
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs namenode -format
WARNING: /home/xwu/hadoop-3.3.4/logs does not exist. Creating.
2022-09-30 04:30:48,109 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = ubuntu-vm.us-central1-a.c.ubuntu-project-363919.internal/10.128.0.2
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 3.3.4
STARTUP_MSG:   classpath = /home/xwu/hadoop-3.3.4/etc/hadoop:/home/xwu/hadoop-3.3.4/share/hadoop/
```

b. Start NameNode and DataNode daemon:

   $ sbin/start-dfs.sh

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ sbin/start-dfs.sh
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [ubuntu-vm]
ubuntu-vm: Warning: Permanently added 'ubuntu-vm,10.128.0.2' (ECDSA) to the lis
t of known hosts.
```

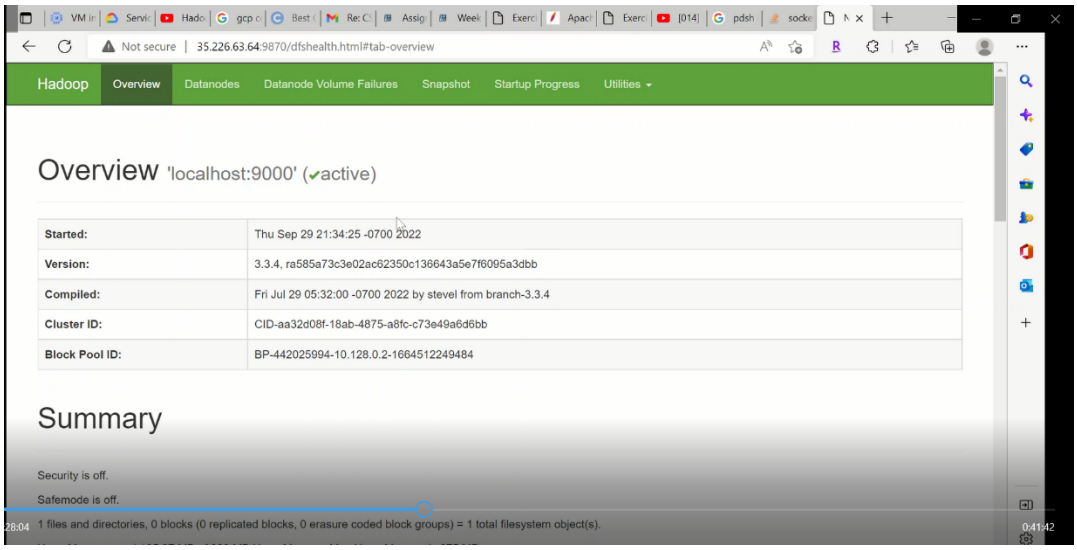
   Note: If you failed to start dfs with “pdsh permission denied” error message, please check variable PDSH_RCMD_TYPE setting:

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ sbin/start-dfs.sh
Starting namenodes on [localhost]
pdsh@ubuntu-vm: localhost: rcmd: socket: Permission denied
Starting datanodes
pdsh@ubuntu-vm: localhost: rcmd: socket: Permission denied
Starting secondary namenodes [ubuntu-vm]
pdsh@ubuntu-vm: ubuntu-vm: rcmd: socket: Permission denied
xwu@ubuntu-vm:~/hadoop-3.3.4$ echo $PDSH_RCMD_TYPE

xwu@ubuntu-vm:~/hadoop-3.3.4$ export PDSH_RCMD_TYPE=ssh
xwu@ubuntu-vm:~/hadoop-3.3.4$ echo $PDSH_RCMD_TYPE
ssh
xwu@ubuntu-vm:~/hadoop-3.3.4$ sbin/start-dfs.sh
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [ubuntu-vm]
ubuntu-vm: Warning: Permanently added 'ubuntu-vm,10.128.0.2' (ECDSA) to the list of known hosts.
xwu@ubuntu-vm:~/hadoop-3.3.4$
```

c. Browser the web interface for the NameNode; by default it's available at port 9870
Check the NameNode through browser:

Note: We can NOT use this URL: http://localhost:8088/, we must replace localhost with the external IP address created in GCP VM instance. e.g. http://35.226.63.64:9870/ in my case.



d. Make the HDFS directories required to execute MapReduce jobs:

Note: We cannot create username randomly, we must use the username with which we login the this Linux server, e.g. cindy is not accepted at my first try, then I had to use xwu in my case.

Create input files and run the mapreduce examples:

$ bin/hdfs dfs -mkdir /user
$ bin/hdfs dfs -mkdir /user/<username>
$ bin/hdfs dfs -mkdir input
$ bin/hdfs dfs -put etc/hadoop/*.xml input
$ bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.4.jar grep input output 'dfs[a-z.]+'

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -mkdir /user
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -mkdir /user/cindy
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -mkdir input
mkdir: `hdfs://localhost:9000/user/xwu': No such file or directory
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -rm -r /user
Deleted /user
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -mkdir /user
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -mkdir /user/xwu
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -mkdir input
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -put etc/hadoop/*.xml input
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.4.jar grep input output 'dfs[a-z.]+'
2022-09-30 04:42:48,509 INFO impl.MetricsConfig: Loaded properties from hadoop-metrics2.properties
2022-09-30 04:42:48,673 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
2022-09-30 04:42:48,674 INFO impl.MetricsSystemImpl: JobTracker metrics system started
2022-09-30 04:42:49,076 INFO input.FileInputFormat: Total input files to process : 10
2022-09-30 04:42:49,109 INFO mapreduce.JobSubmitter: number of splits:10
2022-09-30 04:42:49,283 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_local856803711_0001
2022-09-30 04:42:49,283 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-09-30 04:42:49,512 INFO mapreduce.Job: The url to track the job: http://localhost:8080/
2022-09-30 04:42:49,513 INFO mapreduce.Job: Running job: job_local856803711_0001
2022-09-30 04:42:49,521 INFO mapred.LocalJobRunner: OutputCommitter set in config null
2022-09-30 04:42:49,533 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2022-09-30 04:42:49,533 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false,
es: false
```

Copy and display the output files

$ bin/hdfs dfs -get output output

$ cat output/*

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ ls -rtl output/output/*
-rw-r--r-- 1 xwu xwu  0 Sep 30 04:44 output/output/_SUCCESS
-rw-r--r-- 1 xwu xwu 29 Sep 30 04:44 output/output/part-r-00000
```

If you want to stop here, before you leave, you'd better stop the daemons, otherwise, skip this step.

$ sbin/stop-dfs.sh

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ sbin/stop-dfs.sh
Stopping namenodes on [localhost]
Stopping datanodes
Stopping secondary namenodes [ubuntu-vm]
```

5> Yarn on a Single Node

You can run a MapReduce job on YARN in a pseudo-distributed mode by setting a few parameters and running ResourceManager daemon and NodeManager daemon in addition.

a. Configure parameters as shown

$ vi etc/hadoop/mapred-site.xml

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

$ vi etc/Hadoop/yarn-site.xml

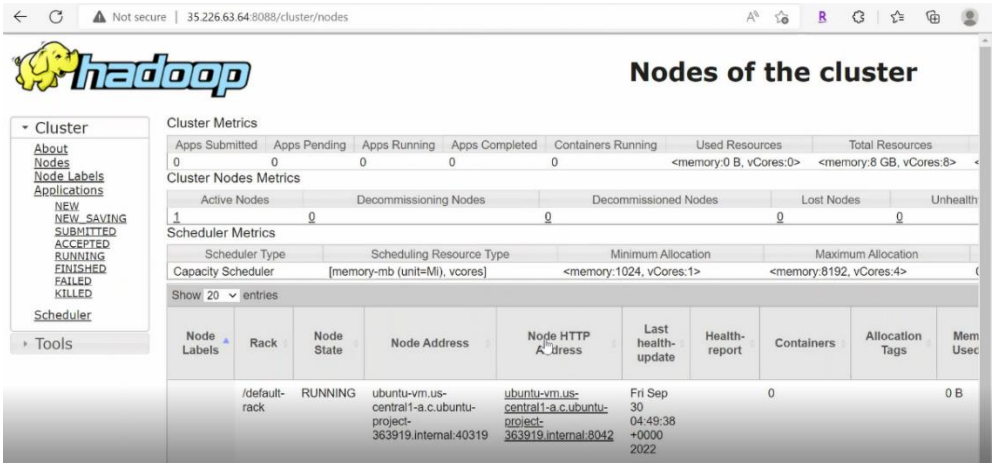
```
<configuration>
    <property>
        <name>yarn.nodemanager.aux-services</name>
        <value>mapreduce_shuffle</value>
    </property>
    <property>
        <name>yarn.nodemanager.env-whitelist</name>
        <value>JAVA_HOME,HADOOP_COMMON_HOME,HADOOP_HDFS_HOME,HADOOP_CONF_DIR,CL
ASSPATH_PREPEND_DISTCACHE,HADOOP_YARN_HOME,HADOOP_HOME,PATH,LANG,TZ,HADOOP_MAPR
ED_HOME</value>
    </property>
</configuration>
```

b. Start ResourceManager daemon and NodeManager daemon

$ sbin/start-yarn.sh

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ vi etc/hadoop/mapred-site.xml
xwu@ubuntu-vm:~/hadoop-3.3.4$ vi etc/hadoop/yarn-site.xml
xwu@ubuntu-vm:~/hadoop-3.3.4$ vi etc/hadoop/mapred-site.xml
xwu@ubuntu-vm:~/hadoop-3.3.4$ sbin/start-yarn.sh
Starting resourcemanager
Starting nodemanagers
xwu@ubuntu-vm:~/hadoop-3.3.4$
```

c. Browser the web interface – It's available at prot 8088, URL like this: http://localhost:8088/
Here, we use http://35.226.63.64:8088/ for GCP Hadoop environment in my case.



d. Run a MapReduce case.

WordCount Example (Reference link: Apache Hadoop 3.3.4 – MapReduce Tutorial)

1> Create Java source code for WordCount.
(Source codes are copied from the above link)
$ vi WordCount.java

2> Compile java code and create a jar
$ $ bin/hadoop com.sun.tools.javac.Main WordCount.java
$ jar cf wc.jar WordCount*.class

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ vi WordCount.java
xwu@ubuntu-vm:~/hadoop-3.3.4$ echo $JAVA_HOME

xwu@ubuntu-vm:~/hadoop-3.3.4$ export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
xwu@ubuntu-vm:~/hadoop-3.3.4$ export PATH=${JAVA_HOME}/bin:${PATH}
xwu@ubuntu-vm:~/hadoop-3.3.4$ export HADOOP_CLASSPATH=${JAVA_HOME}/lib/tools.jar
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hadoop com.sun.tools.javac.Main WordCount.java
xwu@ubuntu-vm:~/hadoop-3.3.4$ jar cf wc.jar WordCount*.class
xwu@ubuntu-vm:~/hadoop-3.3.4$
```

3> Create hdfs directories, file01 and file02
$ bin/hadoop fs -ls /user/xwu/wordcount/input/

$ bin/hadoop fs -cat /user/xwu/wordcount/input/file01
➔ Hello World Bye World

$ bin/hadoop fs -cat /user/xwu/wordcount/input/file02
➔ Hello Hadoop Goodbye Hadoop

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -mkdir wordcount/input
mkdir: `hdfs://localhost:9000/user/xwu/wordcount': No such file or directory
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -mkdir /user
mkdir: `/user': File exists
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -mkdir /user/xwu
mkdir: `/user/xwu': File exists
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -mkdir wordcount
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -mkdir wordcount/input
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -put wordcount/input/* wordcount/input/
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hadoop fs -ls /user/xwu/wordcount/input/
Found 2 items
-rw-r--r--   1 xwu supergroup         22 2022-09-30 05:06 /user/xwu/wordcount/inpu
t/file01
-rw-r--r--   1 xwu supergroup         28 2022-09-30 05:06 /user/xwu/wordcount/inpu
t/file02
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hdfs dfs -mkdir wordcount/output
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hadoop fs -cat /user/joe/wordcount/input/file01
cat: `/user/joe/wordcount/input/file01': No such file or directory
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hadoop fs -cat /user/xwu/wordcount/input/file01
Hello World Bye World
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hadoop fs -cat /user/joe/wordcount/input/file02
cat: `/user/joe/wordcount/input/file02': No such file or directory
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hadoop fs -cat /user/xwu/wordcount/input/file02
Hello Hadoop Goodbye Hadoop
```

4> Run the application

$ bin/hadoop jar wc.jar WordCount /user/xwu/wordcount/input /user/xwu/wordcount/output

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hadoop jar wc.jar WordCount /user/xwu/wordcount/input /user/xwu/wordcount/output
2022-09-30 05:11:36,779 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032
2022-09-30 05:11:37,400 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement t
ication with ToolRunner to remedy this.
2022-09-30 05:11:37,427 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/x
2022-09-30 05:11:37,855 INFO input.FileInputFormat: Total input files to process : 2
2022-09-30 05:11:38,353 INFO mapreduce.JobSubmitter: number of splits:2
2022-09-30 05:11:38,605 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1664513379865_0002
2022-09-30 05:11:38,605 INFO mapreduce.JobSubmitter: Executing with tokens: []
2022-09-30 05:11:38,983 INFO conf.Configuration: resource-types.xml not found
2022-09-30 05:11:38,983 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-09-30 05:11:39,131 INFO impl.YarnClientImpl: Submitted application application_1664513379865_0002
2022-09-30 05:11:39,204 INFO mapreduce.Job: The url to track the job: http://ubuntu-vm.us-central1-a.c.ubuntu-project-363
4513379865_0002/
2022-09-30 05:11:39,205 INFO mapreduce.Job: Running job: job_1664513379865_0002
2022-09-30 05:11:49,409 INFO mapreduce.Job: Job job_1664513379865_0002 running in uber mode : false
2022-09-30 05:11:49,410 INFO mapreduce.Job:  map 0% reduce 0%
```

……

```
Map-Reduce Framework
        Map input records=2
        Map output records=8
        Map output bytes=82
        Map output materialized bytes=85
        Input split bytes=236
        Combine input records=8
        Combine output records=6
        Reduce input groups=5
        Reduce shuffle bytes=85
        Reduce input records=6
        Reduce output records=5
        Spilled Records=12
        Shuffled Maps =2
        Failed Shuffles=0
        Merged Map outputs=2
        GC time elapsed (ms)=484
        CPU time spent (ms)=2560
        Physical memory (bytes) snapshot=759103488
        Virtual memory (bytes) snapshot=7788699648
        Total committed heap usage (bytes)=646447104
        Peak Map Physical memory (bytes)=286932992
        Peak Map Virtual memory (bytes)=2594250752
        Peak Reduce Physical memory (bytes)=186675200
        Peak Reduce Virtual memory (bytes)=2600357888
Shuffle Errors
        BAD_ID=0
        CONNECTION=0
        IO_ERROR=0
        WRONG_LENGTH=0
        WRONG_MAP=0
        WRONG_REDUCE=0
File Input Format Counters
        Bytes Read=50
```

……

```
                Failed Shuffles=0
                Merged Map outputs=2
                GC time elapsed (ms)=484
                CPU time spent (ms)=2560
                Physical memory (bytes) snapshot=759103488
                Virtual memory (bytes) snapshot=7788699648
                Total committed heap usage (bytes)=646447104
                Peak Map Physical memory (bytes)=286932992
                Peak Map Virtual memory (bytes)=2594250752
                Peak Reduce Physical memory (bytes)=186675200
                Peak Reduce Virtual memory (bytes)=2600357888
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=50
        File Output Format Counters
                Bytes Written=41
```

……

5> Display the output

$ bin/hadoop fs -cat /user/joe/wordcount/output/part-r-00000

```
xwu@ubuntu-vm:~/hadoop-3.3.4$ bin/hadoop fs -cat /user/xwu/wordcount/output/part-r-00000
Bye     1
Goodbye 1
Hadoop  2
Hello   2
World   2
```

e. Stop the daemons.

$ sbin/stop-dfs.sh

$ sbin/stop-yarn.sh

Or

$ sbin/stop-all.sh

```
WARNING: Stopping all Apache Hadoop daemons as hduser in 10 seconds.
WARNING: Use CTRL-C to abort.
Stopping namenodes on [localhost]
Stopping datanodes
Stopping secondary namenodes [ubuntu-hadoop-vm]
Stopping nodemanagers
Stopping resourcemanager
```

Done!!!