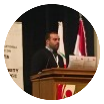
Installing Hadoop 3.2.1 Single node cluster on Windows 10



Hadi Fadlallah Follow
Apr 18 · 7 min read

While working on a project two years ago, I wrote a step-by-step guide to install Hadoop 3.1.0 on Ubuntu 16.04 operating system. Since we are currently working on a new project where we need to install a Hadoop cluster on Windows 10, I decided to write a guide for this process.

1. Prerequisites

First, we need to make sure that the following prerequisites are installed:

1. Java 8 runtime environment (JRE): Hadoop 3 requires a Java 8 installation. I prefer using the offline installer.

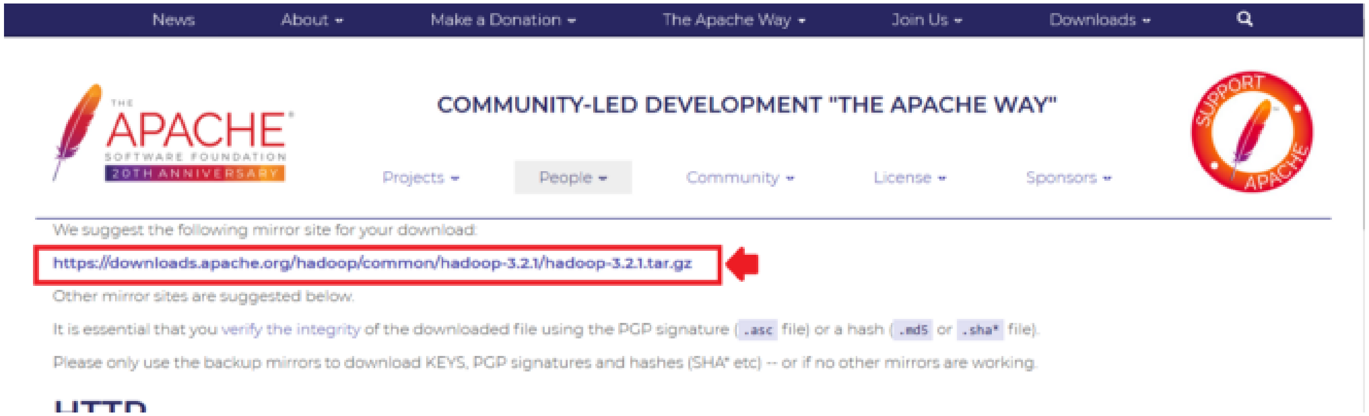
2. Java 8 development Kit (JDK)

3. To unzip downloaded Hadoop binaries, we should install 7zip.

4. I will create a folder “E:\hadoop-env” on my local machine to store downloaded files.

2. Download Hadoop binaries

The first step is to download Hadoop binaries from the official website. The binary package size is about 342 MB.



We suggest the following mirror site for your download:

https://downloads.apache.org/hadoop/common/hadoop-3.2.1/hadoop-3.2.1.tar.gz

Other mirror sites are suggested below.

It is essential that you verify the integrity of the downloaded file using the PGP signature (.asc file) or a hash (.md5 or .sha* file).

Please only use the backup mirrors to download KEYS, PGP signatures and hashes (SHA* etc) -- or if no other mirrors are working.

HTTP

Figure 1 — Hadoop binaries download link

After finishing the file download, we should unpack the package using 7zip int two steps. First, we should extract the hadoop-3.2.1.tar.gz library, and then, we should unpack the extracted tar file:

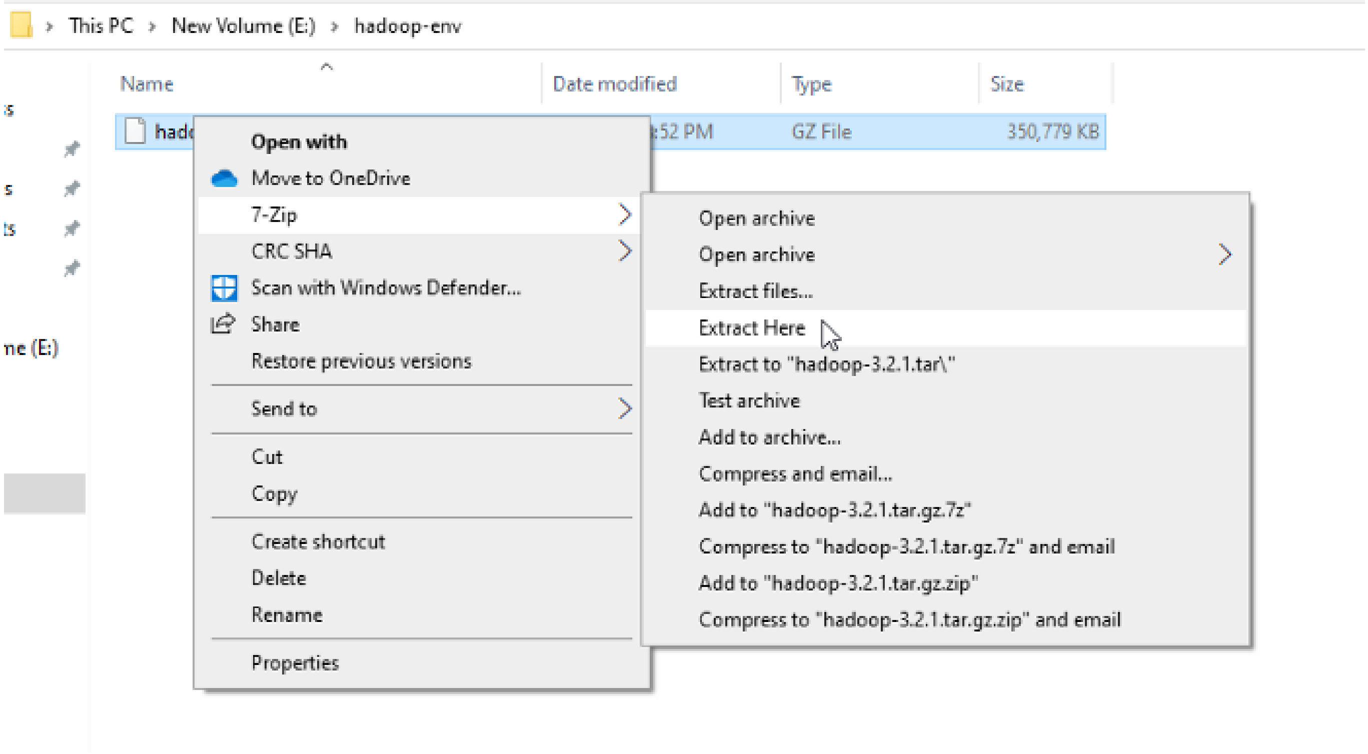
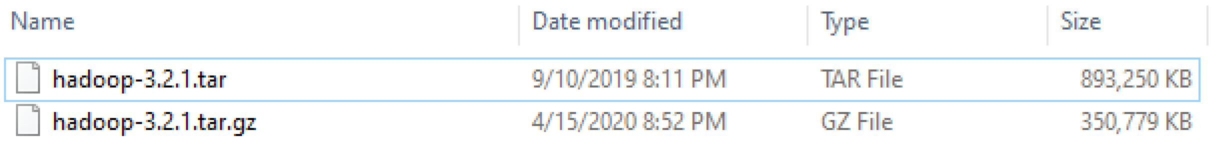

This PC › New Volume (E:) › hadoop-env
Name
Date modified
Type
Size
had
:52 PM
GZ File
350,779 KB
Open with
Move to OneDrive
7-Zip
CRC SHA
Scan with Windows Defender...
Share
Restore previous versions
Send to
Cut
Copy
Create shortcut
Delete
Rename
Properties
Open archive
Open archive
Extract files...
Extract Here
Extract to "hadoop-3.2.1.tar\"
Test archive
Add to archive...
Compress and email...
Add to "hadoop-3.2.1.tar.gz.7z"
Compress to "hadoop-3.2.1.tar.gz.7z" and email
Add to "hadoop-3.2.1.tar.gz.zip"
Compress to "hadoop-3.2.1.tar.gz.zip" and email
ne (E:)

Figure 2 — Extracting hadoop-3.2.1.tar.gz package using 7zip



Name	Date modified	Type	Size
hadoop-3.2.1.tar	9/10/2019 8:11 PM	TAR File	893,250 KB
hadoop-3.2.1.tar.gz	4/15/2020 8:52 PM	GZ File	350,779 KB

Figure 3 — Extracted hadoop-3.2.1.tar file

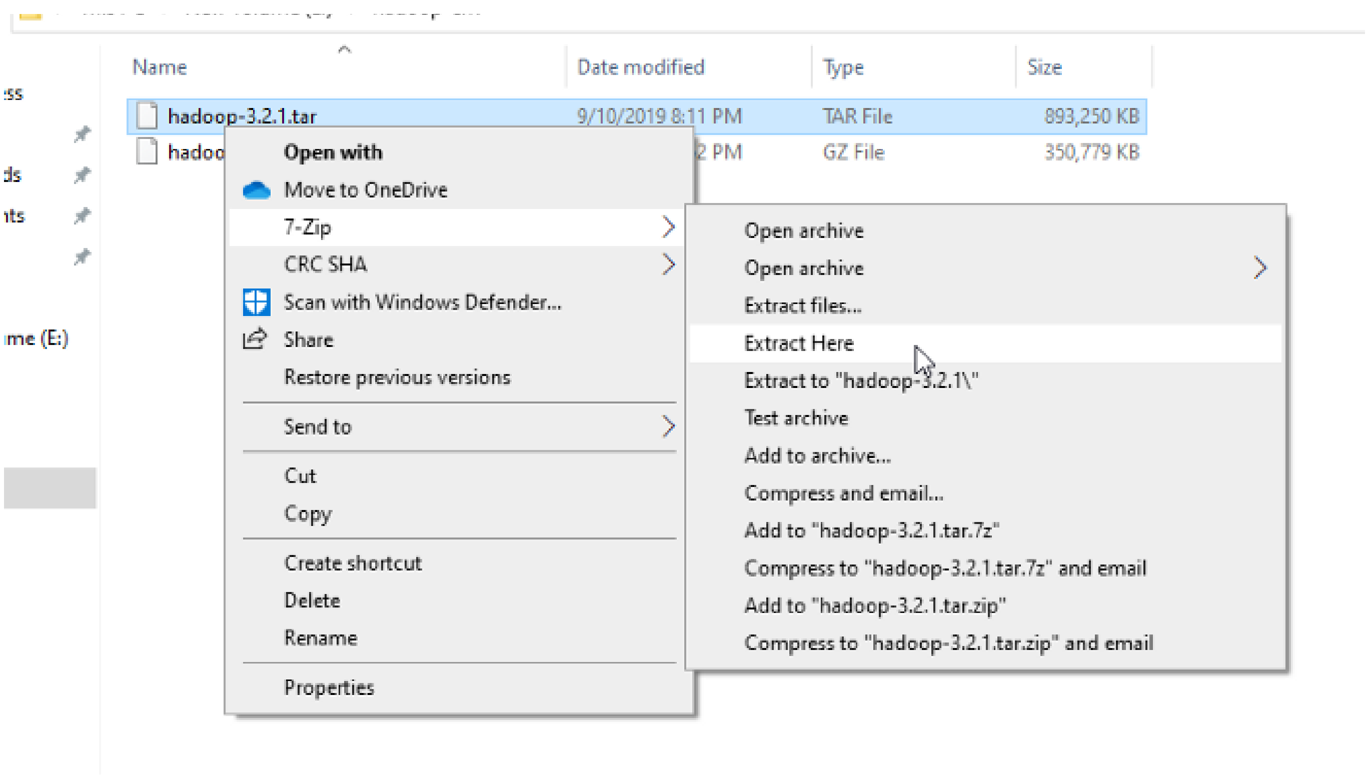
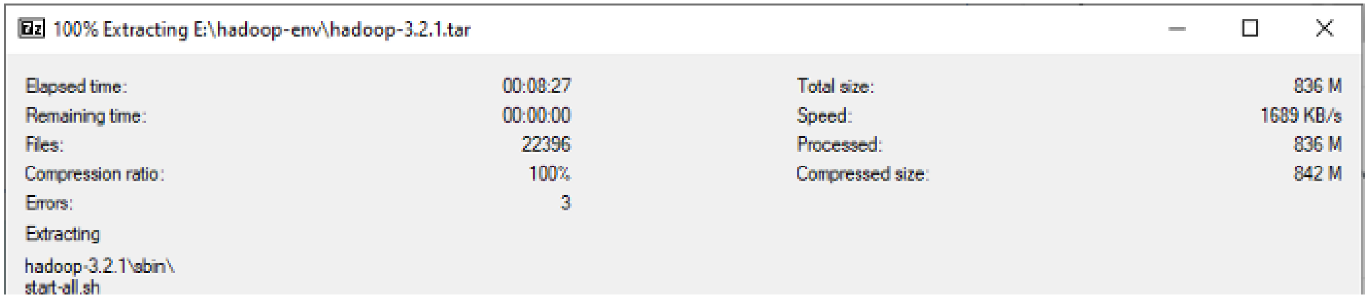


Figure 4 — Extracting the hadoop-3.2.1.tar file

The tar file extraction may take some minutes to finish. In the end, you may see some warnings about symbolic link creation. Just ignore these warnings since they are not related to windows.

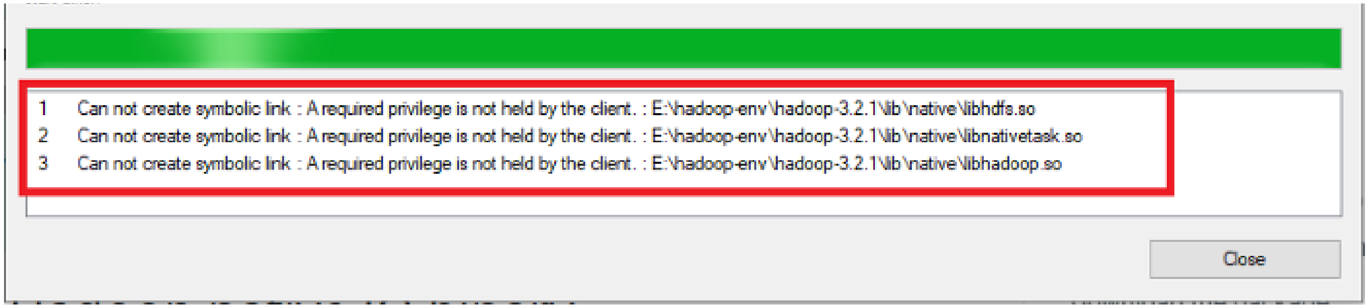


100% Extracting E:\hadoop-env\hadoop-3.2.1.tar

Elapsed time:	00:08:27	Total size:	836 M
Remaining time:	00:00:00	Speed:	1689 KB/s
Files:	22396	Processed:	836 M
Compression ratio:	100%	Compressed size:	842 M
Errors:	3		

Extracting

hadoop-3.2.1\sbin\
start-all.sh



```
1    Can not create symbolic link : A required privilege is not held by the client. : E:\hadoop-env\hadoop-3.2.1\lib\native\libhdfs.so
2    Can not create symbolic link : A required privilege is not held by the client. : E:\hadoop-env\hadoop-3.2.1\lib\native\libnativetask.so
3    Can not create symbolic link : A required privilege is not held by the client. : E:\hadoop-env\hadoop-3.2.1\lib\native\libhadoop.so
```

Figure 5 — Symbolic link warnings

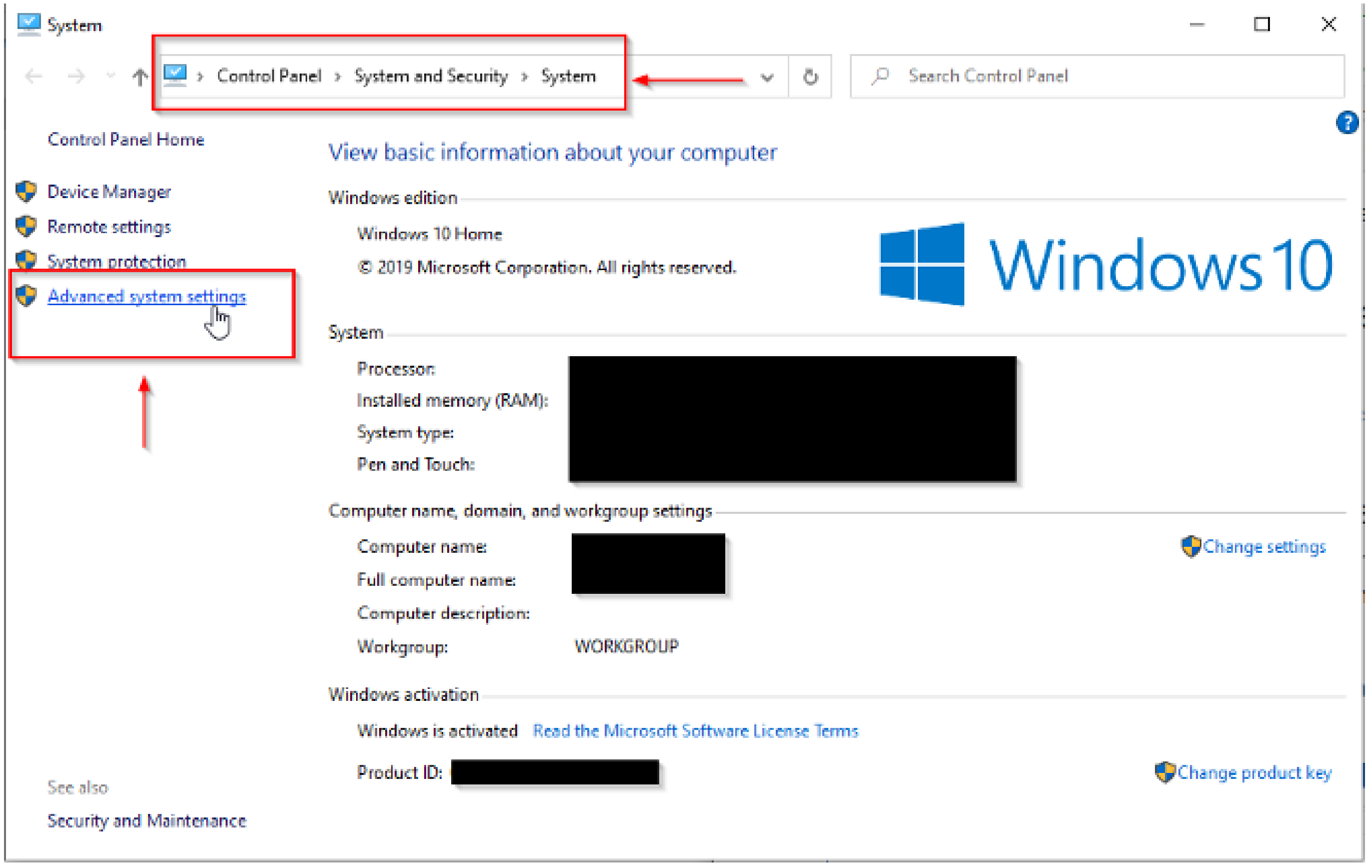
After unpacking the package, we should add the Hadoop native IO libraries, which can be found in the following GitHub repository: https://github.com/cdarlint/winutils.

Since we are installing Hadoop 3.2.1, we should download the files located in https://github.com/cdarlint/winutils/tree/master/hadoop-3.2.1/bin and copy them into the “hadoop-3.2.1\bin” directory.

3. Setting up environment variables

After installing Hadoop and its prerequisites, we should configure the environment variables to define Hadoop and Java default paths.

To edit environment variables, go to Control Panel > System and Security > System (or right-click > properties on My Computer icon) and click on the “Advanced system settings” link.



View basic information about your computer

Windows edition

Windows 10 Home

© 2019 Microsoft Corporation. All rights reserved.

System

Processor:
Installed memory (RAM):
System type:
Pen and Touch:
[redacted]

Computer name, domain, and workgroup settings

Computer name:
Full computer name:
[redacted]
Computer description:
Workgroup: WORKGROUP

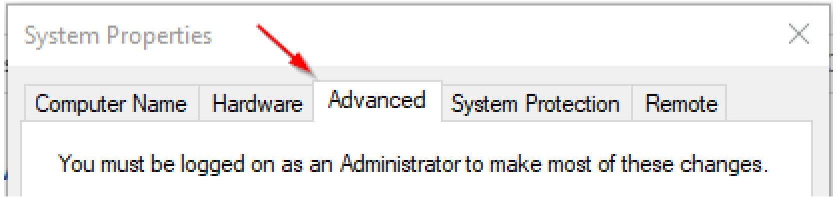
Windows activation

Windows is activated

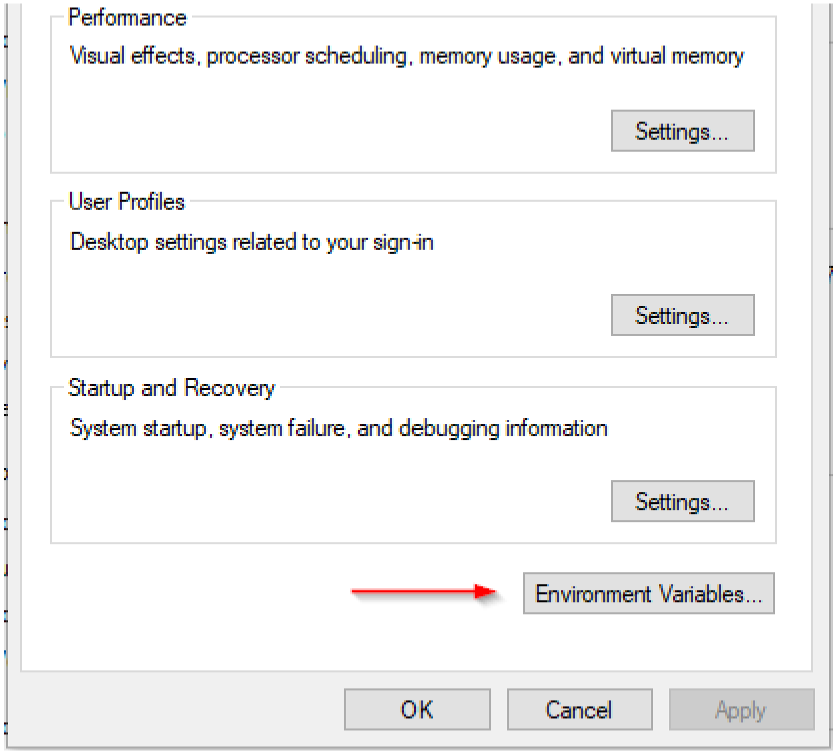
Product ID: [redacted]

Figure 6 — Opening advanced system settings

When the "Advanced system settings" dialog appears, go to the "Advanced" tab and click on the "Environment variables" button located on the bottom of the dialog.



You must be logged on as an Administrator to make most of these changes.



Performance

Visual effects, processor scheduling, memory usage, and virtual memory

User Profiles

Desktop settings related to your sign-in

Startup and Recovery

System startup, system failure, and debugging information

Figure 7 — Advanced system settings dialog

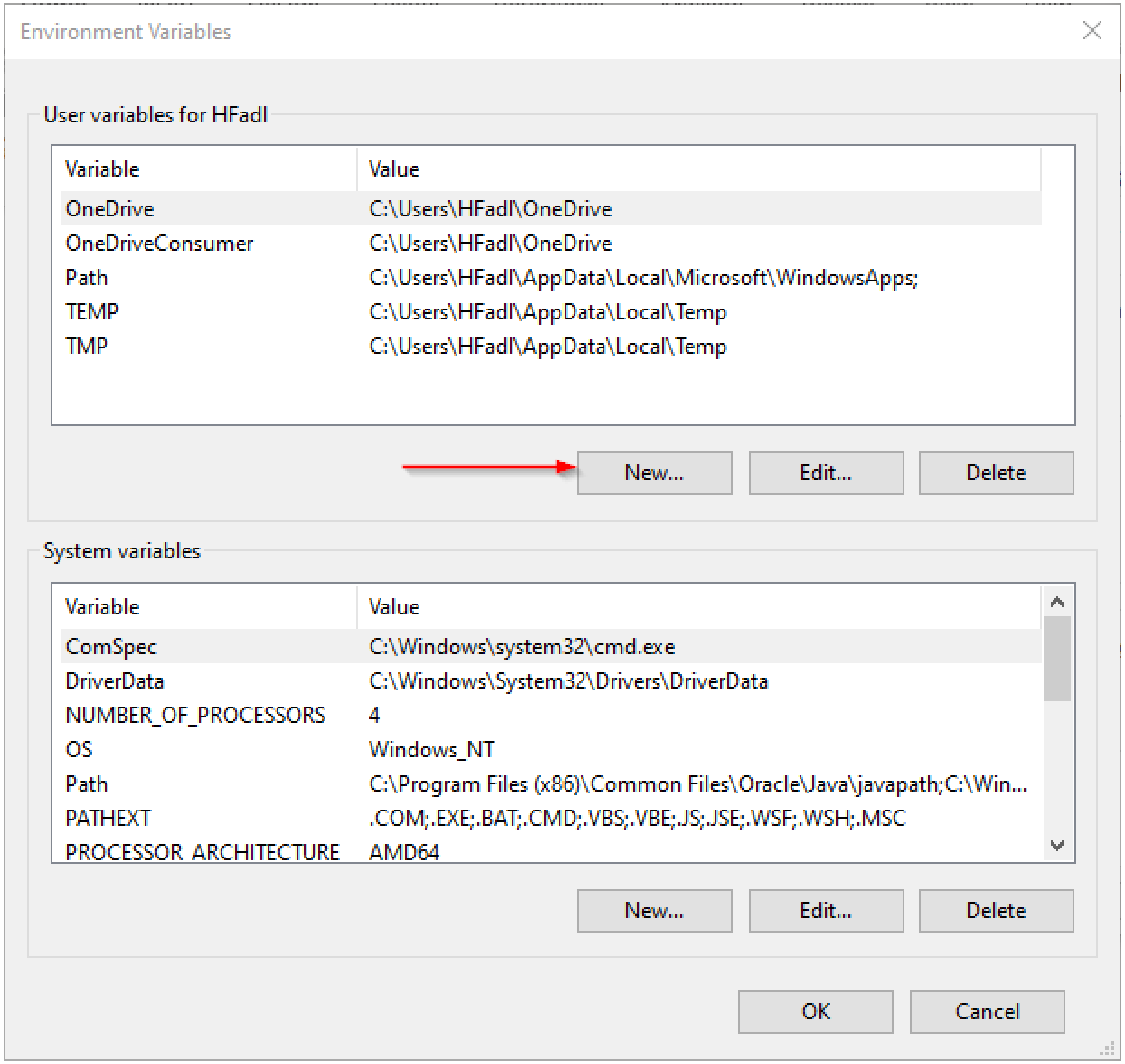
In the “Environment Variables” dialog, press the “New” button to add a new variable.

> *Note: In this guide, we will add user variables since we are configuring Hadoop for a single user. If you are looking to configure Hadoop for multiple users, you can define System variables instead.*

There are two variables to define:

1. JAVA_HOME: JDK installation folder path

2. HADOOP_HOME: Hadoop installation folder path



Environment Variables

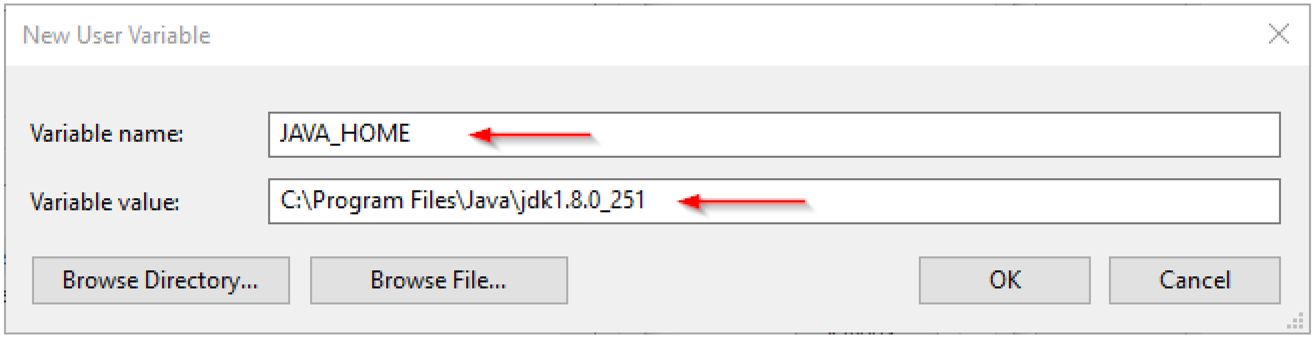
User variables for HFadl

Variable	Value
OneDrive	C:\Users\HFadl\OneDrive
OneDriveConsumer	C:\Users\HFadl\OneDrive
Path	C:\Users\HFadl\AppData\Local\Microsoft\WindowsApps;
TEMP	C:\Users\HFadl\AppData\Local\Temp
TMP	C:\Users\HFadl\AppData\Local\Temp

System variables

Variable	Value
ComSpec	C:\Windows\system32\cmd.exe
DriverData	C:\Windows\System32\Drivers\DriverData
NUMBER_OF_PROCESSORS	4
OS	Windows_NT
Path	C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:\Win...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE	AMD64

Figure 8 — Adding JAVA_HOME variable

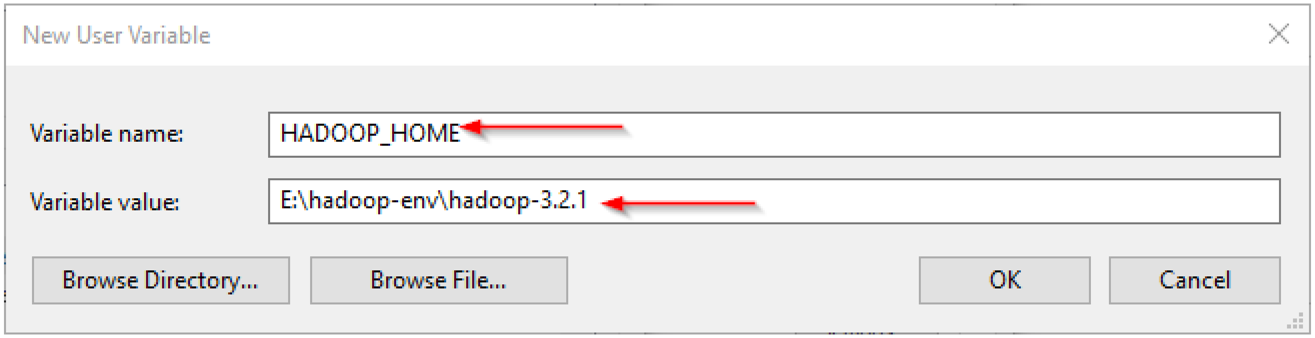


Variable name: JAVA_HOME

Variable value: C:\Program Files\Java\jdk1.8.0_251

Figure 9 — Adding HADOOP_HOME variable

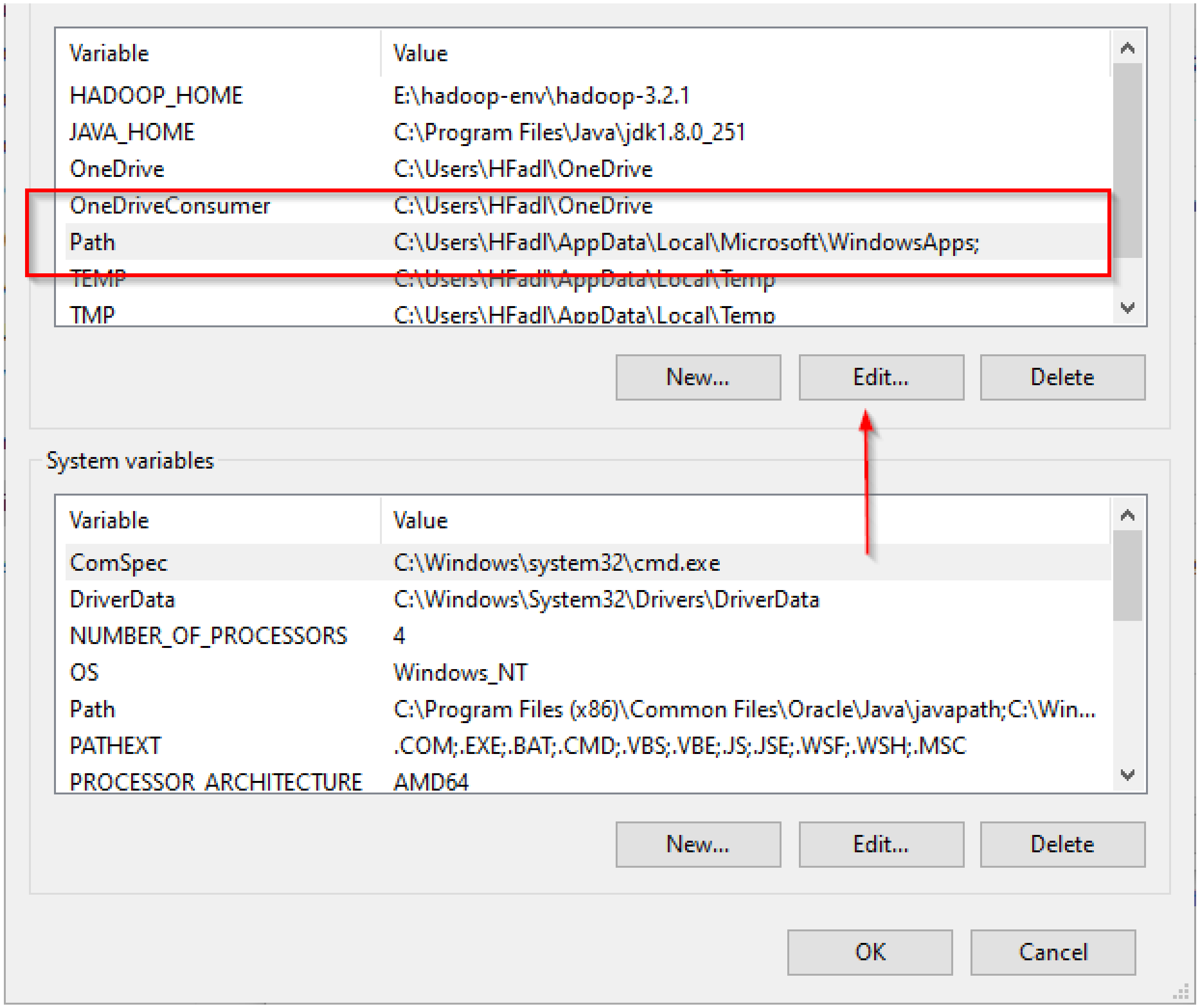
Now, we should edit the PATH variable to add the Java and Hadoop binaries paths as shown in the following screenshots.



Variable name: HADOOP_HOME

Variable value: E:\hadoop-env\hadoop-3.2.1

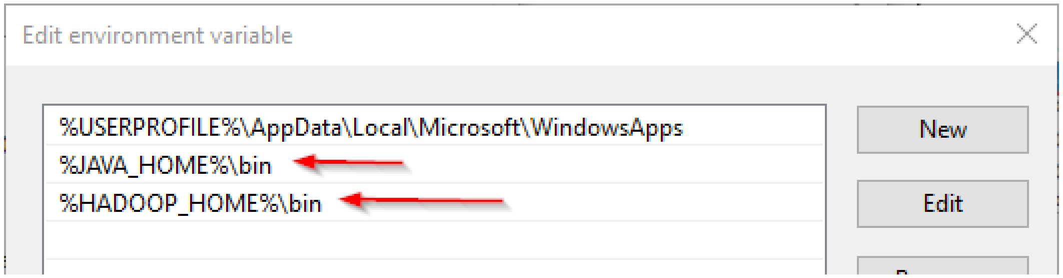
Figure 10 — Editing the PATH variable



Variable	Value
HADOOP_HOME	E:\hadoop-env\hadoop-3.2.1
JAVA_HOME	C:\Program Files\Java\jdk1.8.0_251
OneDrive	C:\Users\HFadl\OneDrive
OneDriveConsumer	C:\Users\HFadl\OneDrive
Path	C:\Users\HFadl\AppData\Local\Microsoft\WindowsApps;
TEMP	C:\Users\HFadl\AppData\Local\Temp
TMP	C:\Users\HFadl\AppData\Local\Temp

Variable	Value
ComSpec	C:\Windows\system32\cmd.exe
DriverData	C:\Windows\System32\Drivers\DriverData
NUMBER_OF_PROCESSORS	4
OS	Windows_NT
Path	C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:\Win...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE	AMD64

Figure 11 — Editing PATH variable



```
%USERPROFILE%\AppData\Local\Microsoft\WindowsApps
%JAVA_HOME%\bin
%HADOOP_HOME%\bin
```

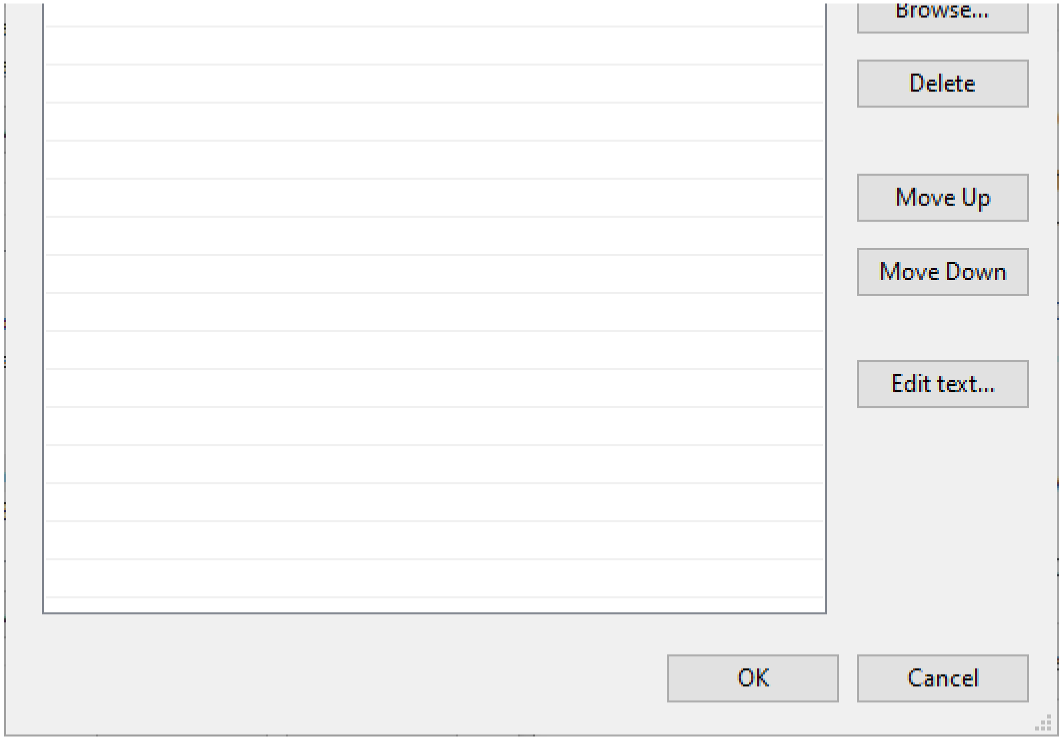


Figure 12— Adding new paths to the PATH variable

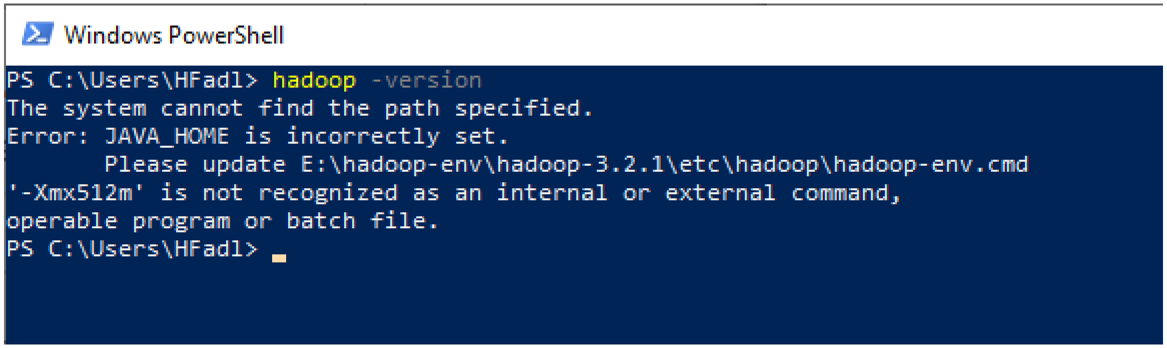
3.1. JAVA_HOME is incorrectly set error

Now, let’s open PowerShell and try to run the following command:

```
hadoop -version
```

In this example, since the JAVA_HOME path contains spaces, I received the following error:

```
JAVA_HOME is incorrectly set
```



```
PS C:\Users\HFadl> hadoop -version
The system cannot find the path specified.
Error: JAVA_HOME is incorrectly set.
       Please update E:\hadoop-env\hadoop-3.2.1\etc\hadoop\hadoop-env.cmd
'-Xmx512m' is not recognized as an internal or external command,
operable program or batch file.
PS C:\Users\HFadl>
```

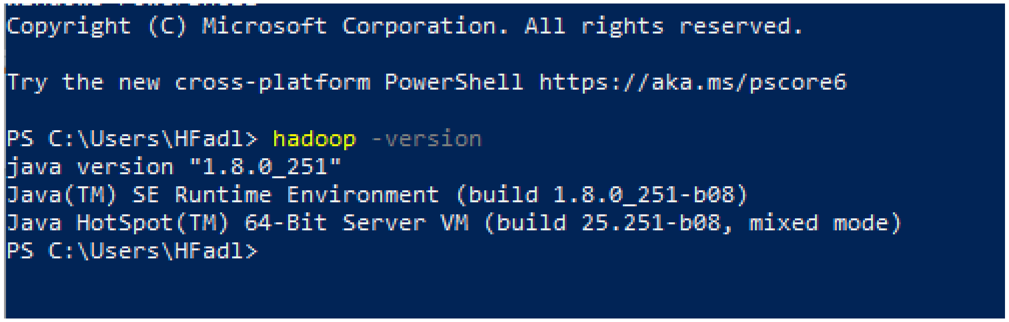
Figure 13 — JAVA_HOME error

To solve this issue, we should use the windows 8.3 path instead. As an example:

- Use “Progra~1” instead of “Program Files”
- Use “Progra~2” instead of “Program Files(x86)”

After replacing “Program Files” with “Progra~1”, we closed and reopened PowerShell and tried the same command. As shown in the screenshot below, it runs without errors.

```
Windows PowerShell
```



```
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\HFadl> hadoop -version
java version "1.8.0_251"
Java(TM) SE Runtime Environment (build 1.8.0_251-b08)
Java HotSpot(TM) 64-Bit Server VM (build 25.251-b08, mixed mode)
PS C:\Users\HFadl>
```

Figure 14 — hadoop -version command executed successfully

4. Configuring Hadoop cluster

There are four files we should alter to configure Hadoop cluster:

1. %HADOOP_HOME%\etc\hadoop\hdfs-site.xml
2. %HADOOP_HOME%\etc\hadoop\core-site.xml
3. %HADOOP_HOME%\etc\hadoop\mapred-site.xml
4. %HADOOP_HOME%\etc\hadoop\yarn-site.xml

4.1. HDFS site configuration

As we know, Hadoop is built using a master-slave paradigm. Before altering the HDFS configuration file, we should create a directory to store all master node (name node) data and another one to store data (data node). In this example, we created the following directories:

- E:\hadoop-env\hadoop-3.2.1\data\dfs\namenode
- E:\hadoop-env\hadoop-3.2.1\data\dfs\datanode

Now, let’s open “hdfs-site.xml” file located in “%HADOOP_HOME%\etc\hadoop” directory, and we should add the following properties within the <configuration></configuration> element:

```
<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

<name>dfs.namenode.name.dir</name>

<value>file:///E:/hadoop-env/hadoop-3.2.1/data/dfs/namenode</value>

</property>

<property>
```

```
<name>dfs.datanode.data.dir</name>

<value>file:///E:/hadoop-env/hadoop-3.2.1/data/dfs/datanode</value>

</property>
```

> *Note that we have set the replication factor to 1 since we are creating a single node cluster.*

4.2. Core site configuration

Now, we should configure the name node URL adding the following XML code into the <configuration></configuration> element within “core-site.xml”:

```
<property>

<name>fs.default.name</name>

<value>hdfs://localhost:9820</value>

</property>
```

4.3. Map Reduce site configuration

Now, we should add the following XML code into the <configuration></configuration> element within “mapred-site.xml”:

```
<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

<description>MapReduce framework name</description>

</property>
```

4.4. Yarn site configuration

Now, we should add the following XML code into the <configuration></configuration> element within “yarn-site.xml”:

```
<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce_shuffle</value>

<description>Yarn Node Manager Aux Service</description>
```

```
</property>
```

5. Formatting Name node

After finishing the configuration, let’s try to format the name node using the following command:

```
hdfs namenode -format
```

Due to a bug in the Hadoop 3.2.1 release, you will receive the following error:

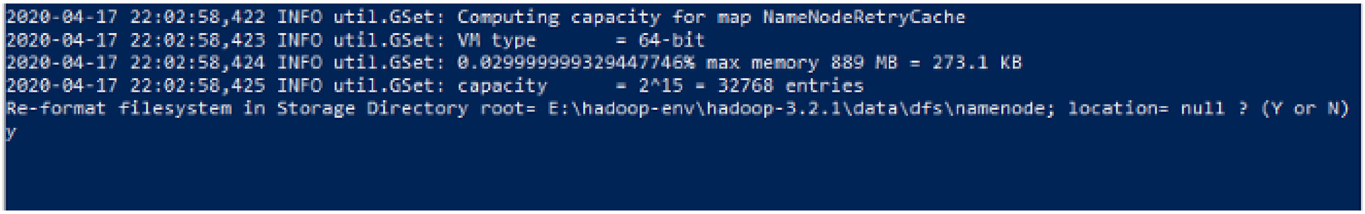
```
2020–04–17 22:04:01,503 ERROR namenode.NameNode: Failed to start
namenode.

java.lang.UnsupportedOperationException

at java.nio.file.Files.setPosixFilePermissions(Files.java:2044)

at
org.apache.hadoop.hdfs.server.common.Storage$StorageDirectory.clearDi
rectory(Storage.java:452)
```

```
at
org.apache.hadoop.hdfs.server.namenode.NNStorage.format(NNStorage.jav
a:591)

at
org.apache.hadoop.hdfs.server.namenode.NNStorage.format(NNStorage.jav
a:613)

at
org.apache.hadoop.hdfs.server.namenode.FSImage.format(FSImage.java:18
8)

at
org.apache.hadoop.hdfs.server.namenode.NameNode.format(NameNode.java:
1206)

at
org.apache.hadoop.hdfs.server.namenode.NameNode.createNameNode(NameNo
de.java:1649)

at
org.apache.hadoop.hdfs.server.namenode.NameNode.main(NameNode.java:17
59)

2020-04-17 22:04:01,511 INFO util.ExitUtil: Exiting with status 1:
java.lang.UnsupportedOperationException

2020-04-17 22:04:01,518 INFO namenode.NameNode: SHUTDOWN_MSG:
```

This issue will be solved within the next release. For now, you can fix it temporarily using the following steps (reference):

1. Download hadoop-hdfs-3.2.1.jar file from the following link.
2. Rename the file name hadoop-hdfs-3.2.1.jar to hadoop-hdfs-3.2.1.bak in folder %HADOOP_HOME%\share\hadoop\hdfs
3. Copy the downloaded hadoop-hdfs-3.2.1.jar to folder %HADOOP_HOME%\share\hadoop\hdfs

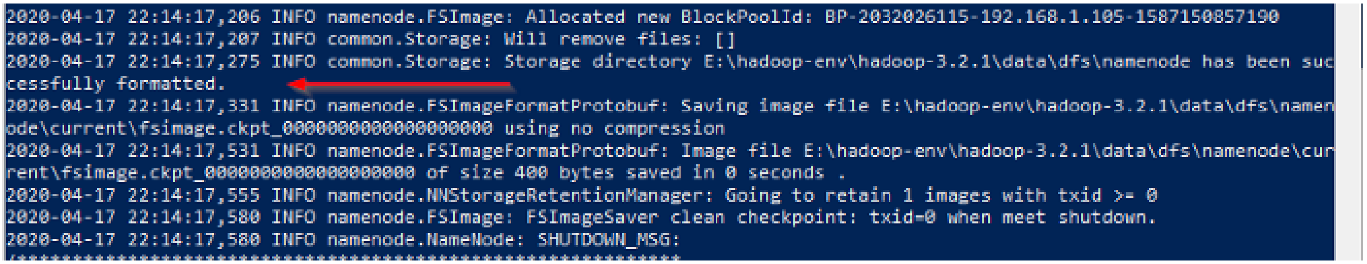
Now, if we try to re-execute the format command (Run the command prompt or PowerShell as administrator), you need to approve file system format.



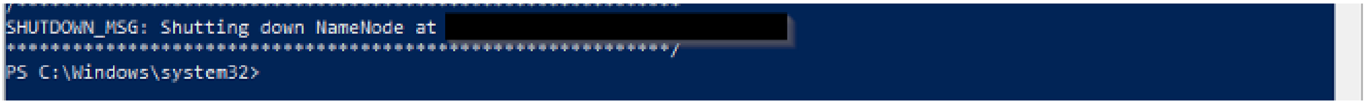
```
2020-04-17 22:02:58,422 INFO util.GSet: Computing capacity for map NameNodeRetryCache
2020-04-17 22:02:58,423 INFO util.GSet: VM type       = 64-bit
2020-04-17 22:02:58,424 INFO util.GSet: 0.029999999329447746% max memory 889 MB = 273.1 KB
2020-04-17 22:02:58,425 INFO util.GSet: capacity      = 2^15 = 32768 entries
Re-format filesystem in Storage Directory root= E:\hadoop-env\hadoop-3.2.1\data\dfs\namenode; location= null ? (Y or N)
y
```

Figure 15 — File system format approval

And the command is executed successfully:



```
2020-04-17 22:14:17,206 INFO namenode.FSImage: Allocated new BlockPoolId: BP-2032026115-192.168.1.105-1587150857190
2020-04-17 22:14:17,207 INFO common.Storage: Will remove files: []
2020-04-17 22:14:17,275 INFO common.Storage: Storage directory E:\hadoop-env\hadoop-3.2.1\data\dfs\namenode has been suc
cessfully formatted.
2020-04-17 22:14:17,331 INFO namenode.FSImageFormatProtobuf: Saving image file E:\hadoop-env\hadoop-3.2.1\data\dfs\namen
ode\current\fsimage.ckpt_0000000000000000000 using no compression
2020-04-17 22:14:17,531 INFO namenode.FSImageFormatProtobuf: Image file E:\hadoop-env\hadoop-3.2.1\data\dfs\namenode\cur
rent\fsimage.ckpt_0000000000000000000 of size 400 bytes saved in 0 seconds .
2020-04-17 22:14:17,555 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2020-04-17 22:14:17,580 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2020-04-17 22:14:17,580 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
```



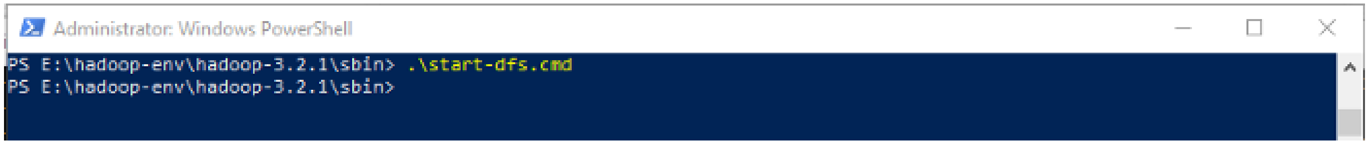
```
/**********************************************************
SHUTDOWN_MSG: Shutting down NameNode at [redacted]
**********************************************************/
PS C:\Windows\system32>
```

Figure 16 — Command executed successfully

6. Starting Hadoop services

Now, we will open PowerShell, and navigate to “%HADOOP_HOME%\sbin” directory. Then we will run the following command to start the Hadoop nodes:

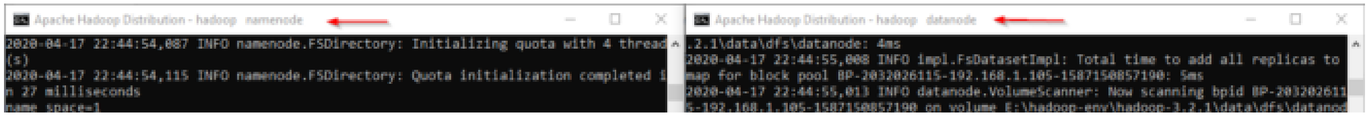
```
.\start-dfs.cmd
```



```
PS E:\hadoop-env\hadoop-3.2.1\sbin> .\start-dfs.cmd
PS E:\hadoop-env\hadoop-3.2.1\sbin>
```

Figure 17 — StartingHadoop nodes

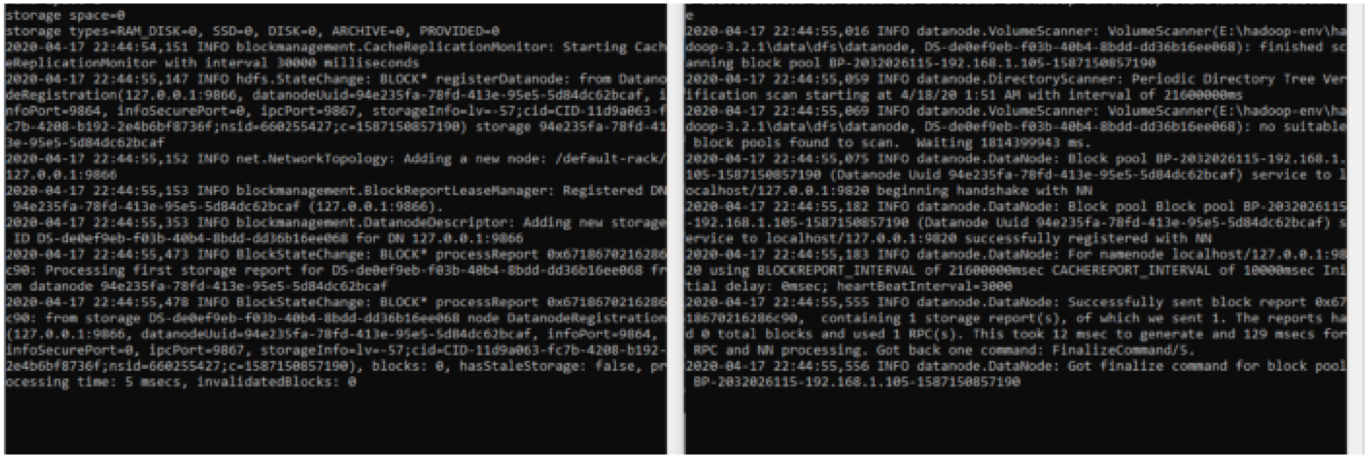
Two command prompt windows will open (one for the name node and one for the data node) as follows:



Apache Hadoop Distribution - hadoop namenode
Apache Hadoop Distribution - hadoop datanode

```
2020-04-17 22:44:54,087 INFO namenode.FSDirectory: Initializing quota with 4 thread
(s)
2020-04-17 22:44:54,115 INFO namenode.FSDirectory: Quota initialization completed i
n 27 milliseconds
name space=1
```

```
.2.1\data\dfs\datanode: 4ms
2020-04-17 22:44:55,008 INFO impl.FsDatasetImpl: Total time to add all replicas to
map for block pool BP-2032026115-192.168.1.105-1587150857190: 5ms
2020-04-17 22:44:55,013 INFO datanode.VolumeScanner: Now scanning bpid BP-203202611
5-192.168.1.105-1587150857190 on volume E:\hadoop-env\hadoop-3.2.1\data\dfs\datanod
```



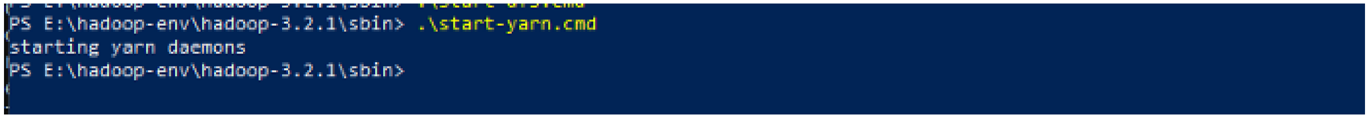
```
storage space=0
storage types=RAM_DISK=0, SSD=0, DISK=0, ARCHIVE=0, PROVIDED=0
2020-04-17 22:44:54,151 INFO blockmanagement.CacheReplicationMonitor: Starting Cach
eReplicationMonitor with interval 30000 milliseconds
2020-04-17 22:44:55,147 INFO hdfs.StateChange: BLOCK* registerDatanode: from Datano
deRegistration(127.0.0.1:9866, datanodeUuid=94e235fa-78fd-413e-95e5-5d84dc62bcaf, i
nfoPort=9864, infoSecurePort=0, ipcPort=9867, storageInfo=lv=-57;cid=CID-11d9a063-f
c7b-4208-b192-2e4b6bf8736f;nsid=660255427;c=1587150857190) storage 94e235fa-78fd-41
3e-95e5-5d84dc62bcaf
2020-04-17 22:44:55,152 INFO net.NetworkTopology: Adding a new node: /default-rack/
127.0.0.1:9866
2020-04-17 22:44:55,153 INFO blockmanagement.BlockReportLeaseManager: Registered DN
 94e235fa-78fd-413e-95e5-5d84dc62bcaf (127.0.0.1:9866).
2020-04-17 22:44:55,353 INFO blockmanagement.DatanodeDescriptor: Adding new storage
 ID DS-de0ef9eb-f03b-40b4-8bdd-dd36b16ee068 for DN 127.0.0.1:9866
2020-04-17 22:44:55,473 INFO BlockStateChange: BLOCK* processReport 0x671867021628
c90: Processing first storage report for DS-de0ef9eb-f03b-40b4-8bdd-dd36b16ee068 fr
om datanode 94e235fa-78fd-413e-95e5-5d84dc62bcaf
2020-04-17 22:44:55,478 INFO BlockStateChange: BLOCK* processReport 0x671867021628
c90: from storage DS-de0ef9eb-f03b-40b4-8bdd-dd36b16ee068 node DatanodeRegistration
(127.0.0.1:9866, datanodeUuid=94e235fa-78fd-413e-95e5-5d84dc62bcaf, infoPort=9864,
infoSecurePort=0, ipcPort=9867, storageInfo=lv=-57;cid=CID-11d9a063-fc7b-4208-b192-
2e4b6bf8736f;nsid=660255427;c=1587150857190), blocks: 0, hasStaleStorage: false, pr
ocessing time: 5 msecs, invalidatedBlocks: 0
```

```
e
2020-04-17 22:44:55,016 INFO datanode.VolumeScanner: VolumeScanner(E:\hadoop-env\ha
doop-3.2.1\data\dfs\datanode, DS-de0ef9eb-f03b-40b4-8bdd-dd36b16ee068): finished sc
anning block pool BP-2032026115-192.168.1.105-1587150857190
2020-04-17 22:44:55,059 INFO datanode.DirectoryScanner: Periodic Directory Tree Ver
ification scan starting at 4/18/20 1:51 AM with interval of 21600000ms
2020-04-17 22:44:55,069 INFO datanode.VolumeScanner: VolumeScanner(E:\hadoop-env\ha
doop-3.2.1\data\dfs\datanode, DS-de0ef9eb-f03b-40b4-8bdd-dd36b16ee068): no suitable
 block pools found to scan.  Waiting 1814399943 ms.
2020-04-17 22:44:55,075 INFO datanode.DataNode: Block pool BP-2032026115-192.168.1.
105-1587150857190 (Datanode Uuid 94e235fa-78fd-413e-95e5-5d84dc62bcaf) service to l
ocalhost/127.0.0.1:9820 beginning handshake with NN
2020-04-17 22:44:55,182 INFO datanode.DataNode: Block pool Block pool BP-2032026115
-192.168.1.105-1587150857190 (Datanode Uuid 94e235fa-78fd-413e-95e5-5d84dc62bcaf) s
ervice to localhost/127.0.0.1:9820 successfully registered with NN
2020-04-17 22:44:55,183 INFO datanode.DataNode: For namenode localhost/127.0.0.1:98
20 using BLOCKREPORT_INTERVAL of 21600000msec CACHEREPORT_INTERVAL of 10000msec Ini
tial delay: 0msec; heartBeatInterval=3000
2020-04-17 22:44:55,555 INFO datanode.DataNode: Successfully sent block report 0x67
18670216286c90,  containing 1 storage report(s), of which we sent 1. The reports ha
d 0 total blocks and used 1 RPC(s). This took 12 msec to generate and 129 msecs for
 RPC and NN processing. Got back one command: FinalizeCommand/5.
2020-04-17 22:44:55,556 INFO datanode.DataNode: Got finalize command for block pool
 BP-2032026115-192.168.1.105-1587150857190
```

Figure 18 — Hadoop nodes command prompt windows

Next, we must start the Hadoop Yarn service using the following command:

```
./start-yarn.cmd
```

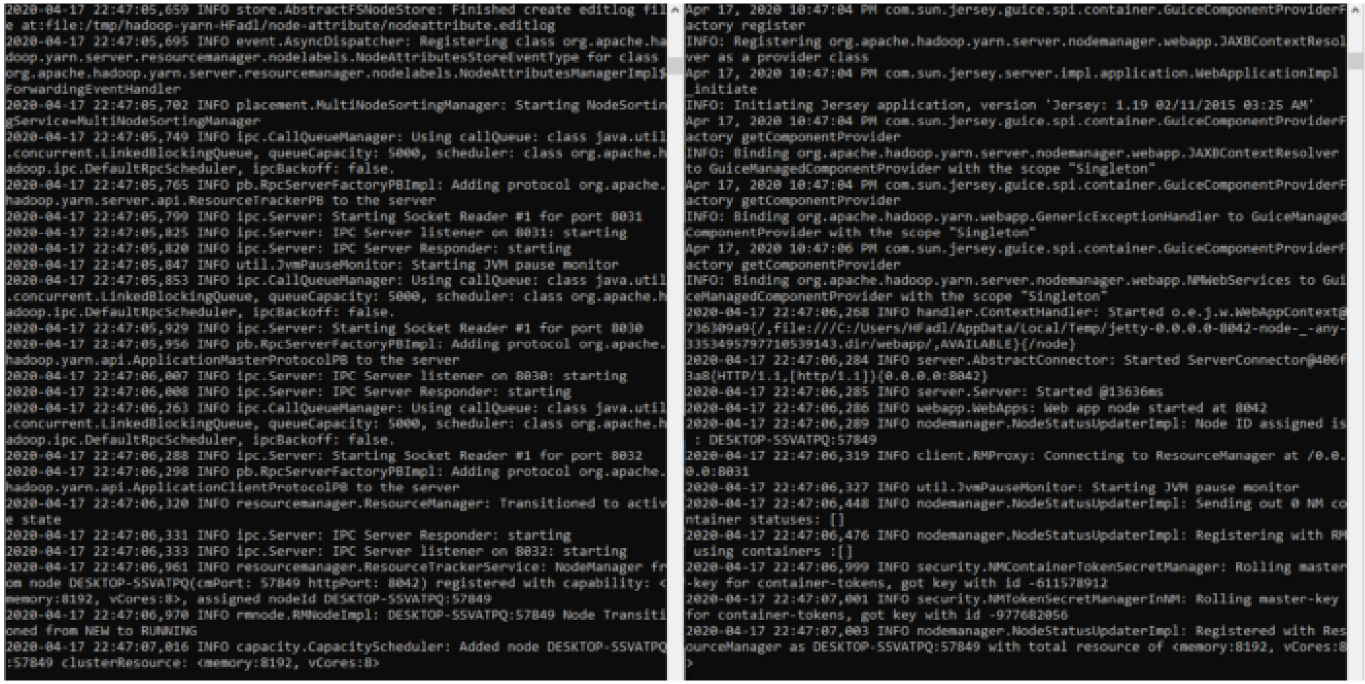


```
PS E:\hadoop-env\hadoop-3.2.1\sbin> .\start-yarn.cmd
starting yarn daemons
PS E:\hadoop-env\hadoop-3.2.1\sbin>
```

Figure 19 — Starting Hadoop Yarn services

Two command prompt windows will open (one for the resource manager and one for the node manager) as follows:

Apache Hadoop Distribution - yarn resourcemanager
Apache Hadoop Distribution - yarn nodemanager



```
2020-04-17 22:47:05,659 INFO store.AbstractFSNodeStore: Finished create editlog fil
e at:file:/tmp/hadoop-yarn-HFadl/node-attribute/nodeattribute.editlog
2020-04-17 22:47:05,695 INFO event.AsyncDispatcher: Registering class org.apache.ha
doop.yarn.server.resourcemanager.nodelabels.NodeAttributesStoreEventType for class
org.apache.hadoop.yarn.server.resourcemanager.nodelabels.NodeAttributesManagerImpl$
ForwardingEventHandler
2020-04-17 22:47:05,702 INFO placement.MultiNodeSortingManager: Starting NodeSortin
gService=MultiNodeSortingManager
2020-04-17 22:47:05,749 INFO ipc.CallQueueManager: Using callQueue: class java.util
.concurrent.LinkedBlockingQueue, queueCapacity: 5000, scheduler: class org.apache.h
adoop.ipc.DefaultRpcScheduler, ipcBackoff: false.
2020-04-17 22:47:05,765 INFO pb.RpcServerFactoryPBImpl: Adding protocol org.apache.
hadoop.yarn.server.api.ResourceTrackerPB to the server
2020-04-17 22:47:05,799 INFO ipc.Server: Starting Socket Reader #1 for port 8031
2020-04-17 22:47:05,825 INFO ipc.Server: IPC Server listener on 8031: starting
2020-04-17 22:47:05,820 INFO ipc.Server: IPC Server Responder: starting
2020-04-17 22:47:05,847 INFO util.JvmPauseMonitor: Starting JVM pause monitor
2020-04-17 22:47:05,853 INFO ipc.CallQueueManager: Using callQueue: class java.util
.concurrent.LinkedBlockingQueue, queueCapacity: 5000, scheduler: class org.apache.h
adoop.ipc.DefaultRpcScheduler, ipcBackoff: false.
2020-04-17 22:47:05,929 INFO ipc.Server: Starting Socket Reader #1 for port 8030
2020-04-17 22:47:05,956 INFO pb.RpcServerFactoryPBImpl: Adding protocol org.apache.
hadoop.yarn.api.ApplicationMasterProtocolPB to the server
2020-04-17 22:47:06,007 INFO ipc.Server: IPC Server listener on 8030: starting
2020-04-17 22:47:06,008 INFO ipc.Server: IPC Server Responder: starting
2020-04-17 22:47:06,263 INFO ipc.CallQueueManager: Using callQueue: class java.util
.concurrent.LinkedBlockingQueue, queueCapacity: 5000, scheduler: class org.apache.h
adoop.ipc.DefaultRpcScheduler, ipcBackoff: false.
2020-04-17 22:47:06,288 INFO ipc.Server: Starting Socket Reader #1 for port 8032
2020-04-17 22:47:06,298 INFO pb.RpcServerFactoryPBImpl: Adding protocol org.apache.
hadoop.yarn.api.ApplicationClientProtocolPB to the server
2020-04-17 22:47:06,320 INFO resourcemanager.ResourceManager: Transitioned to activ
e state
2020-04-17 22:47:06,331 INFO ipc.Server: IPC Server Responder: starting
2020-04-17 22:47:06,333 INFO ipc.Server: IPC Server listener on 8032: starting
2020-04-17 22:47:06,961 INFO resourcemanager.ResourceTrackerService: NodeManager fr
om node DESKTOP-SSVATPQ(cmPort: 57849 httpPort: 8042) registered with capability: <
memory:8192, vCores:8>, assigned nodeId DESKTOP-SSVATPQ:57849
2020-04-17 22:47:06,970 INFO rmnode.RMNodeImpl: DESKTOP-SSVATPQ:57849 Node Transiti
oned from NEW to RUNNING
2020-04-17 22:47:07,016 INFO capacity.CapacityScheduler: Added node DESKTOP-SSVATPQ
:57849 clusterResource: <memory:8192, vCores:8>
```

```
Apr 17, 2020 10:47:04 PM com.sun.jersey.guice.spi.container.GuiceComponentProviderF
actory register
INFO: Registering org.apache.hadoop.yarn.server.nodemanager.webapp.JAXBContextResol
ver as a provider class
Apr 17, 2020 10:47:04 PM com.sun.jersey.server.impl.application.WebApplicationImpl
_initiate
INFO: Initiating Jersey application, version 'Jersey: 1.19 02/11/2015 03:25 AM'
Apr 17, 2020 10:47:04 PM com.sun.jersey.guice.spi.container.GuiceComponentProviderF
actory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.server.nodemanager.webapp.JAXBContextResolver
to GuiceManagedComponentProvider with the scope "Singleton"
Apr 17, 2020 10:47:04 PM com.sun.jersey.guice.spi.container.GuiceComponentProviderF
actory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.webapp.GenericExceptionHandler to GuiceManaged
ComponentProvider with the scope "Singleton"
Apr 17, 2020 10:47:06 PM com.sun.jersey.guice.spi.container.GuiceComponentProviderF
actory getComponentProvider
INFO: Binding org.apache.hadoop.yarn.server.nodemanager.webapp.NMWebServices to Gui
ceManagedComponentProvider with the scope "Singleton"
2020-04-17 22:47:06,268 INFO handler.ContextHandler: Started o.e.j.w.WebAppContext@
736309a9{/,file:///C:/Users/HFadl/AppData/Local/Temp/jetty-0.0.0.0-8042-node-_-any-
3353495797710539143.dir/webapp/,AVAILABLE}{/node}
2020-04-17 22:47:06,284 INFO server.AbstractConnector: Started ServerConnector@406f
3a8{HTTP/1.1,[http/1.1]}{0.0.0.0:8042}
2020-04-17 22:47:06,285 INFO server.Server: Started @13636ms
2020-04-17 22:47:06,286 INFO webapp.WebApps: Web app node started at 8042
2020-04-17 22:47:06,289 INFO nodemanager.NodeStatusUpdaterImpl: Node ID assigned is
 : DESKTOP-SSVATPQ:57849
2020-04-17 22:47:06,319 INFO client.RMProxy: Connecting to ResourceManager at /0.0.
0.0:8031
2020-04-17 22:47:06,327 INFO util.JvmPauseMonitor: Starting JVM pause monitor
2020-04-17 22:47:06,448 INFO nodemanager.NodeStatusUpdaterImpl: Sending out 0 NM co
ntainer statuses: []
2020-04-17 22:47:06,476 INFO nodemanager.NodeStatusUpdaterImpl: Registering with RM
 using containers :[]
2020-04-17 22:47:06,999 INFO security.NMContainerTokenSecretManager: Rolling master
-key for container-tokens, got key with id -611578912
2020-04-17 22:47:07,001 INFO security.NMTokenSecretManagerInNM: Rolling master-key
for container-tokens, got key with id -977682056
2020-04-17 22:47:07,003 INFO nodemanager.NodeStatusUpdaterImpl: Registered with Res
ourceManager as DESKTOP-SSVATPQ:57849 with total resource of <memory:8192, vCores:8
>
```

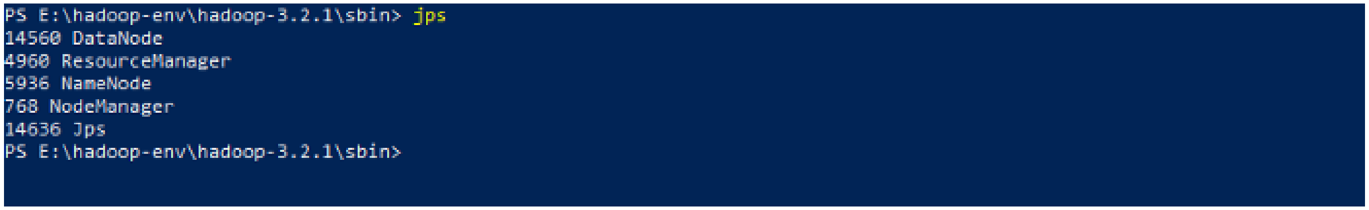
Figure 20— Node manager and Resource manager command prompt windows

To make sure that all services started successfully, we can run the following command:

```
jps
```

It should display the following services:

```
14560 DataNode
4960 ResourceManager
5936 NameNode
768 NodeManager
14636 Jps
```



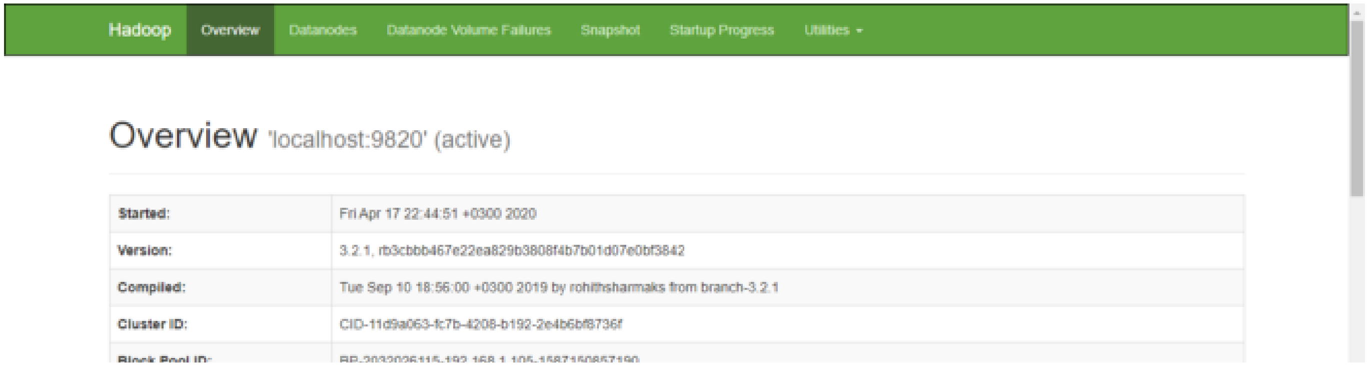
```
PS E:\hadoop-env\hadoop-3.2.1\sbin> jps
14560 DataNode
4960 ResourceManager
5936 NameNode
768 NodeManager
14636 Jps
PS E:\hadoop-env\hadoop-3.2.1\sbin>
```

Figure 21 — Executing jps command

7. Hadoop Web UI

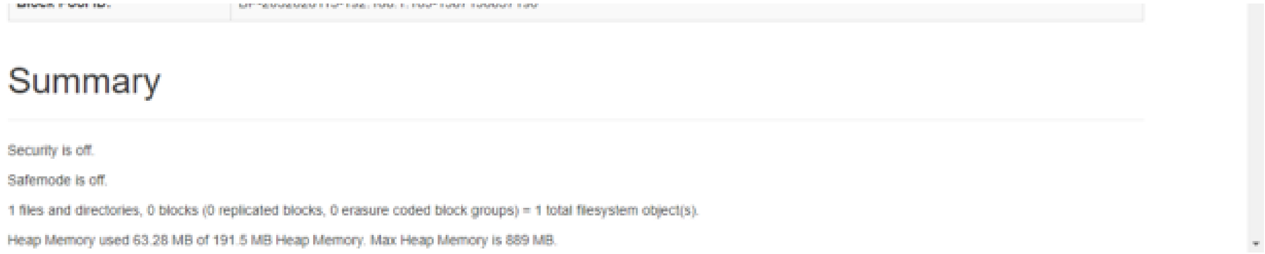
There are three web user interfaces to be used:

- Name node web page: http://localhost:9870/dfshealth.html



Overview 'localhost:9820' (active)

Started:	Fri Apr 17 22:44:51 +0300 2020
Version:	3.2.1, rb3cbbb467e22ea829b3808f4b7b01d07e0bf3842
Compiled:	Tue Sep 10 18:56:00 +0300 2019 by rohithsharmaks from branch-3.2.1
Cluster ID:	CID-11d9a063-fc7b-4208-b192-2e4b6bf8736f
Block Pool ID:	BP-2032026115-192.168.1.105-1587150857190



Summary

Security is off.

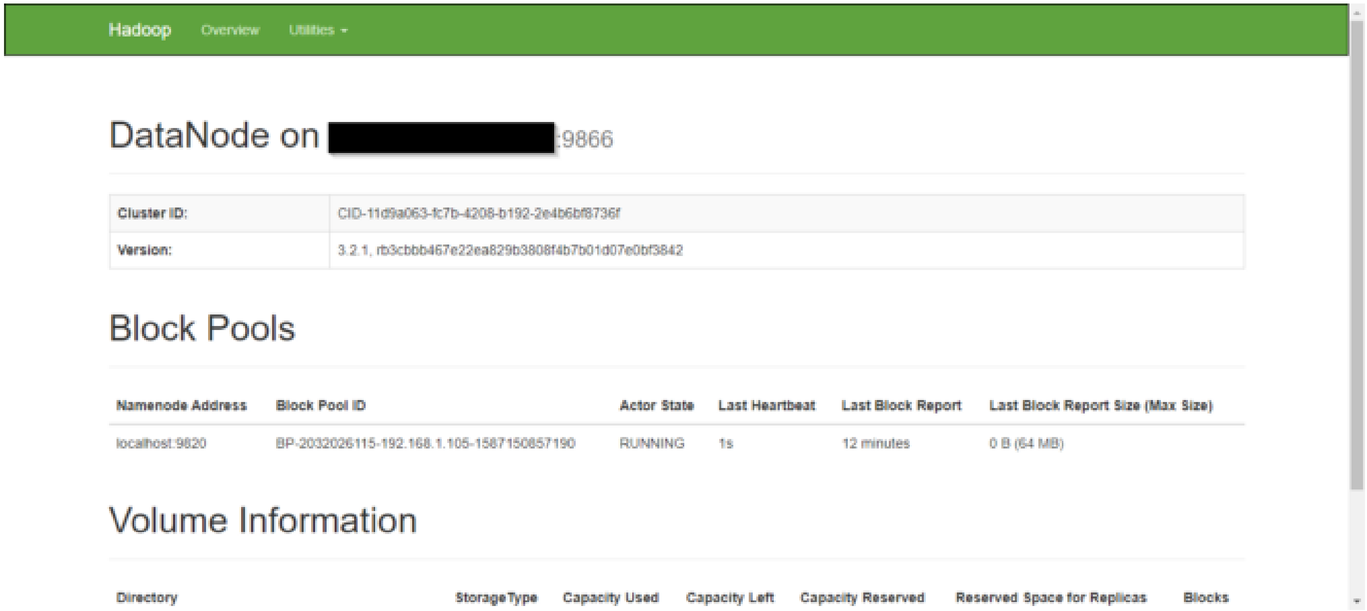
Safemode is off.

1 files and directories, 0 blocks (0 replicated blocks, 0 erasure coded block groups) = 1 total filesystem object(s).

Heap Memory used 63.28 MB of 191.5 MB Heap Memory. Max Heap Memory is 889 MB.

Figure 22 — Name node web page

- Data node web page: http://localhost:9864/datanode.html



DataNode on [redacted]:9866

Cluster ID:	CID-11d9a063-fc7b-4208-b192-2e4b6bf8736f
Version:	3.2.1, rb3cbbb467e22ea829b3808f4b7b01d07e0bf3842

Block Pools

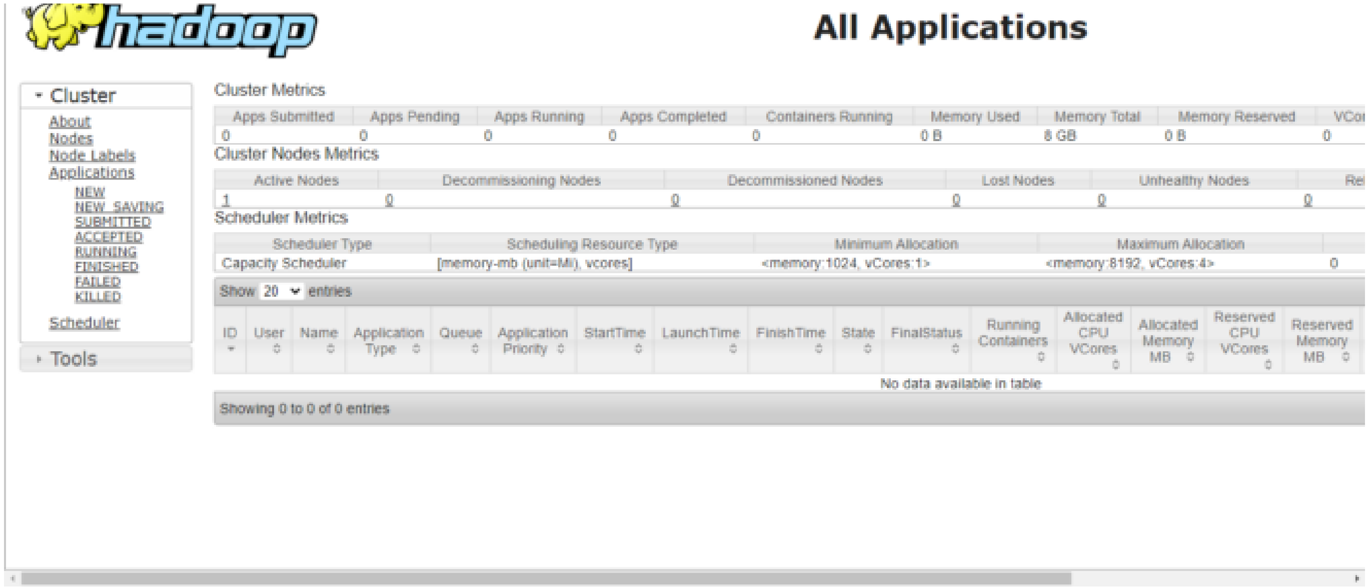
Namenode Address	Block Pool ID	Actor State	Last Heartbeat	Last Block Report	Last Block Report Size (Max Size)
localhost:9820	BP-2032026115-192.168.1.105-1587150857190	RUNNING	1s	12 minutes	0 B (64 MB)

Volume Information

Directory	StorageType	Capacity Used	Capacity Left	Capacity Reserved	Reserved Space for Replicas	Blocks

Figure 23 — Data node web page

- Yarn web page: http://localhost:8088/cluster



All Applications

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCor
0	0	0	0	0	0 B	8 GB	0 B	0

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Re
1	0	0	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation	
Capacity Scheduler	[memory-mb (unit=Mi), vcores]	<memory:1024, vCores:1>	<memory:8192, vCores:4>	0

Show 20 entries

ID	User	Name	Application Type	Queue	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU VCores	Allocated Memory MB	Reserved CPU VCores	Reserved Memory MB
No data available in table															

Showing 0 to 0 of 0 entries

Figure 24 — Yarn web page

8. References

1. Hadi Fadlallah, Installing Hadoop 3.1.0 multi-node cluster on Ubuntu 16.04 Step by Step, TowardsDataScience.com

2. Jozef Jarosciak, How to install a Hadoop single node cluster on Windows 10

3. Raymond Tang, Install Hadoop 3.2.1 on Windows 10 Step by Step Guide, kontext.tech

4. Stack overflow Q/A website

Hadoop Big Data Windows 10 Hadoop Cluster Hadoop 3