

Setup HBase-Spark Integration

Note: Please terminate the instance once your work is complete.

Note:

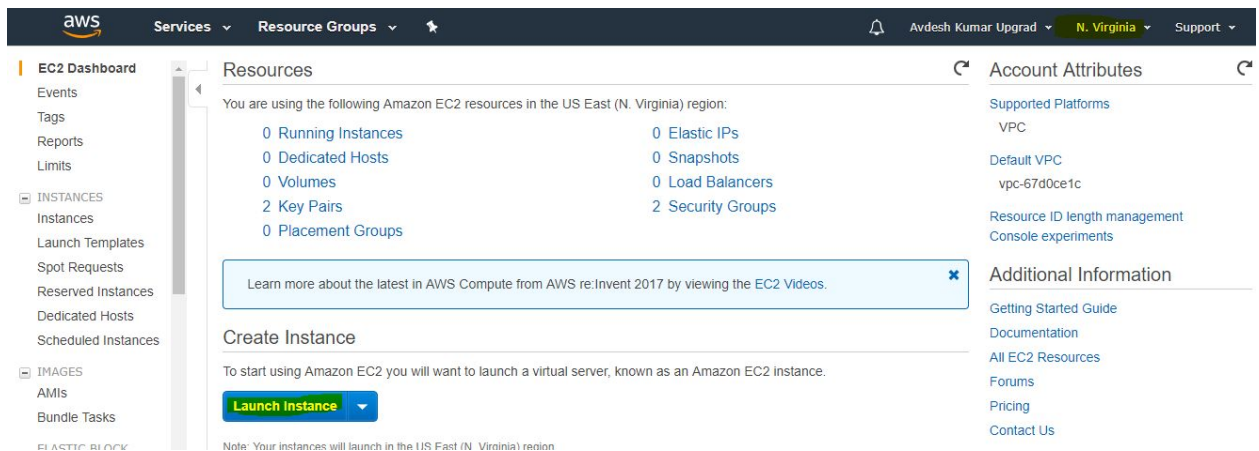
Region: N.verginia

Instance type: t2.medium

Vol- 10 GB(Megnatic)

AMI-ami-04681a1dbd79675a5

1. Go to the EC2 dashboard and click on Launch Instances.



The screenshot shows the AWS Management Console for the US East (N. Virginia) region. The left sidebar contains the navigation menu with categories like EC2 Dashboard, INSTANCES, and IMAGES. The main content area is titled 'Resources' and lists various EC2 resources: 0 Running Instances, 0 Elastic IPs, 0 Dedicated Hosts, 0 Snapshots, 0 Volumes, 0 Load Balancers, 2 Key Pairs, and 2 Security Groups. Below this is a 'Create Instance' section with a 'Launch Instance' button. A note at the bottom states: 'Note: Your instances will launch in the US East (N. Virginia) region'. On the right, there are sections for 'Account Attributes' and 'Additional Information' with links to documentation and support.



2. Click on Community AMI

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 1: Choose an Amazon Machine Image (AMI) [Cancel and Exit](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Search for an AMI by entering a search term e.g. "Windows"

Quick Start 1 to 50 of 117,107 AMIs

My AMIs

AWS Marketplace

Community AMIs

Operating system

- ☐ Amazon Linux
- ☐ Cent OS
- ☐ Debian
- ☐ Fedora
- ☐ CentOS

	amzn2-ami-hvm-2.0.20181008-x86_64-gp2 - ami-0922553b7b0369273 Amazon Linux 2 AMI 2.0.20181008 x86_64 HVM gp2 Root device type: ebs Virtualization type: hvm ENA Enabled: Yes 64-bit	Select
	amzn-ami-hvm-2018.03.0.20180811-x86_64-gp2 - ami-0ff8a91507f77f867 Amazon Linux AMI 2018.03.0.20180811 x86_64 HVM GP2 Root device type: ebs Virtualization type: hvm ENA Enabled: Yes 64-bit	Select
	suse-sles-15-v20180816-hvm-ssd-x86_64 - ami-06ea7729e394412c8 SUSE Linux Enterprise Server 15 (HVM, 64-bit, SSD-Backed) 64-bit	Select

3. in the search bar, Ssearch the AMI- **ami-04681a1dbd79675a5** and click on select.

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 1: Choose an Amazon Machine Image (AMI) [Cancel and Exit](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Search for an AMI by entering a search term e.g. "Windows"

Quick Start (0) 1 to 1 of 1 AMIs

My AMIs (0)

AWS Marketplace (3033)

Community AMIs (1)

Operating system

- ☐ Amazon Linux
- ☐ Cent OS
- ☐ Debian
- ☐ Fedora
- ☐ CentOS

	amzn2-ami-hvm-2.0.20180810-x86_64-gp2 - ami-04681a1dbd79675a5 Amazon Linux 2 AMI 2.0.20180810 x86_64 HVM gp2 Root device type: ebs Virtualization type: hvm ENA Enabled: Yes 64-bit	Select
--	--	------------------------

The following results for "ami-04681a1dbd79675a5" were found in other catalogs:

[3033 results in AWS Marketplace](#)
AWS Marketplace provides partnered Software that is pre-configured to run on AWS

4. Choose instance type should be **"t2.medium"** and click on **Next Configuration** details



1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All instance types Current generation Show/Hide Columns

Currently selected: t2.medium (Variable ECUs, 2 vCPUs, 2.3 GHz, Intel Broadwell E5-2686v4, 4 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes

5. Nothing to change click on **Add storage**.

Network [Create new VPC](#)

Subnet [Create new subnet](#)

Auto-assign Public IP

Placement group ☐ Add instance to placement group.

IAM role [Create new IAM role](#)

Shutdown behavior

Enable termination protection ☐ Protect against accidental termination

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

6. Give storage size **10 GB** and type **Magnetic**, click on **ADD Tags**

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/xvda	snap-0e848b692c3c7de9e	10	Magnetic (standard)	N/A	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Tags](#)



7. Click on **click to add a new tag**, and in value give your new instance a name, click on **next**

aws Services ▾ Resource Groups ▾ ⭐

Avdesh Kumar Upgrad ▾ N. Virginia ▾ Support ▾

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to volumes, instances or both.
Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key (127 characters maximum)	Value (255 characters maximum)	Instances ⓘ	Volumes ⓘ
Name	hbase spark	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Add another tag (Up to 50 tags maximum)

[Cancel](#) [Previous](#) [Review and Launch](#) Next: Configure Security Group

8. If you have already security group just select with my IP address only. and click on **Review an launch**.

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☐ Create a new security group
☒ Select an existing security group

Security Group ID	Name	Description	Actions
<input type="checkbox"/> sg-1a7abd51	default	default VPC security group	Copy to new
<input checked="" type="checkbox"/> sg-05f597abe00563ca4	launch-wizard-1	launch-wizard-1 created 2018-10-15T13:42:25.241+05:30	Copy to new

Inbound rules for sg-05f597abe00563ca4 (Selected security groups: sg-05f597abe00563ca4)

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ
All TCP	TCP	0 - 65535	125.99.155.86/32	

[Cancel](#) [Previous](#) [Review and Launch](#)

9. Choose **Magnetic** and click on next



Boot from General Purpose (SSD)



General Purpose (SSD) volumes provide the ability to burst to 3000 IOPS per volume, independent of volume size, to meet the performance needs of most applications and also deliver a consistent baseline of 3 IOPS/GiB.

- ☐ Make General Purpose (SSD) the default boot volume for all instance launches from the console going forward (recommended).
- ☐ Make General Purpose (SSD) the boot volume for this instance.
- ☒ Continue with Magnetic as the boot volume for this instance.

Free tier eligible customers can get up to 30GB of General Purpose (SSD) storage.

☐ Don't show again

Next

10. Click on **launch**

1. Choose AMI2. Choose Instance Type3. Configure Instance4. Add Storage5. Add Tags6. Configure Security Group7. Review

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

Your instance configuration is not eligible for the free usage tier
To launch an instance that's eligible for the free usage tier, check your AMI selection, instance type, configuration options, or storage devices. Learn more about [free usage tier](#) eligibility and usage restrictions.

Don't show me this again

▼ AMI Details

amzn2-ami-hvm-2.0.20180810-x86_64-gp2 - ami-04681a1dbd79675a5

Amazon Linux 2 AMI 2.0.20180810 x86_64 HVM gp2

Root Device Type: ebs Virtualization type: hvm

Edit AMI

▼ Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.medium	Variable	2	4	EBS only	-	Low to Moderate

Edit instance type

▼ Security Groups

Edit security groups

Cancel

Previous

Launch

11. Choose existing key pair (if you have) and click on **launch instance**.



Select an existing key pair or create a new key pair



A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Choose an **existing key pair**

Select a key pair

avdheshupgrad

☒ I acknowledge that I have access to the selected private key file (avdheshupgrad.pem), and that without this file, I won't be able to log into my instance.

Cancel

Launch Instances

12. Click on **view instances**.

Launch Status



Get notified of estimated charges

Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click **View Instances** to monitor your instances' status. Once your instances are in the **running** state, you can **connect** to them from the Instances screen. [Find out](#) how to connect to your instances.

Here are some helpful resources to get you started

- [How to connect to your Linux instance](#)
- [Amazon EC2: User Guide](#)
- [Learn about AWS Free Usage Tier](#)
- [Amazon EC2: Discussion Forum](#)

While your instances are launching you can also

[Create status check alarms](#) to be notified when these instances fail status checks. (Additional charges may apply)

[Create and attach additional EBS volumes](#) (Additional charges may apply)

[Manage security groups](#)

View Instances

Your New instance is created now let us see how to install HBase-3.0.0 Spark-2.1.0 on it.

Installing HBase-Spark Integration Setup

1. Now login to the instance on **putty** using the **ppk** file.



aws Services Resource Groups

EC2 Dashboard
Events
Tags
Reports
Limits

INSTANCES

Instances

Launch Templates
Spot Requests
Reserved Instances
Dedicated Hosts
Scheduled Instances

IMAGES

AMIs
Bundle Tasks

ELASTIC BLOCK STORE

Volumes
Snapshots
Lifecycle Manager

Launch Instance Connect Actions

Filter by tags and attributes or search by keyword

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IP
hbase spark	i-02ff3821a1f351dd6	t2.medium	us-east-1c	running	Initializing	None	ec2-35-170-61-30.com...	35

Instance: **i-02ff3821a1f351dd6 (hbase spark)** Public DNS: **ec2-35-170-61-30.compute-1.amazonaws.com**

Description Status Checks Monitoring Tags

Property	Value
Instance ID	i-02ff3821a1f351dd6
Instance state	running
Instance type	t2.medium
Elastic IPs	
Availability zone	us-east-1c
Security groups	launch-wizard-1 . view inbound rules . view outbound rules
Scheduled events	No scheduled events
AMI ID	amzn2-ami-hvm-2.0.20180810-x86_64-aa2
Public DNS (IPv4)	ec2-35-170-61-30.compute-1.amazonaws.com
IPv4 Public IP	35.170.61.30
IPv6 IPs	-
Private DNS	ip-172-31-3-215.ec2.internal
Private IPs	172.31.3.215
Secondary private IPs	
VPC ID	vpc-67d0ce1c
Subnet ID	subnet-7678a511

2. Login with ec2-user then switch to root user.

```
root@ip-172-31-3-215:~  
login as: ec2-user  
Authenticating with public key "imported-openssh-key"  
  
_ _ | _ _ | _ )  
_ | ( _ _ | /  Amazon Linux 2 AMI  
_ | \ _ _ | _ |  
  
https://aws.amazon.com/amazon-linux-2/  
6 package(s) needed for security, out of 337 available  
Run "sudo yum update" to apply all updates.  
[ec2-user@ip-172-31-3-215 ~]$ sudo -i  
[root@ip-172-31-3-215 ~]#
```

3. Install git, wget using the below command.

```
yum -y install git wget
```



```
[root@ip-172-31-3-215 ~]# yum -y install git wget
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
--> Package git.x86_64 0:2.14.4-1.amzn2.0.1 will be installed
--> Processing Dependency: perl-Git = 2.14.4-1.amzn2.0.1 for package: git-2.14.4-1.amzn2.0.1.x86_64
--> Processing Dependency: git-core-doc = 2.14.4-1.amzn2.0.1 for package: git-2.14.4-1.amzn2.0.1.x86_64
--> Processing Dependency: git-core = 2.14.4-1.amzn2.0.1 for package: git-2.14.4-1.amzn2.0.1.x86_64
--> Processing Dependency: emacs-filesystem >= 25.3 for package: git-2.14.4-1.amzn2.0.1.x86_64
--> Processing Dependency: perl(Term::ReadKey) for package: git-2.14.4-1.amzn2.0.1.x86_64
--> Processing Dependency: perl(Git::I18N) for package: git-2.14.4-1.amzn2.0.1.x86_64
--> Processing Dependency: perl(Git) for package: git-2.14.4-1.amzn2.0.1.x86_64
--> Processing Dependency: perl(Error) for package: git-2.14.4-1.amzn2.0.1.x86_64
--> Processing Dependency: libsecret-1.so.0()(64bit) for package: git-2.14.4-1.amzn2.0.1.x86_64
--> Processing Dependency: libpcre2-8.so.0()(64bit) for package: git-2.14.4-1.amzn2.0.1.x86_64
--> Package wget.x86_64 0:1.14-15.amzn2.1 will be updated
--> Package wget.x86_64 0:1.14-15.amzn2.1.1 will be an update
--> Running transaction check
```

4. Now we'll install the java.

First create the directory `/usr/java`.

```
[root@ip-172-31-3-215 ~]# mkdir /usr/java/
```

and then download the java.

```
[root@ip-172-31-3-215 ~]# wget https://s3.amazonaws.com/java-1.8/jdk-8u161-linux-x64.tar.gz
```

```
root@ip-172-31-3-215:~
[root@ip-172-31-3-215 ~]# mkdir /usr/java/
[root@ip-172-31-3-215 ~]# wget https://s3.amazonaws.com/java-1.8/jdk-8u161-linux-x64.tar.gz
--2018-10-24 12:39:51-- https://s3.amazonaws.com/java-1.8/jdk-8u161-linux-x64.tar.gz
Resolving s3.amazonaws.com (s3.amazonaws.com)... 52.216.82.11
Connecting to s3.amazonaws.com (s3.amazonaws.com)|52.216.82.11|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 189756259 (181M) [application/x-tar]
Saving to: 'jdk-8u161-linux-x64.tar.gz'

100%[=====]
2018-10-24 12:39:54 (48.9 MB/s) - 'jdk-8u161-linux-x64.tar.gz' saved [189756259/189756259]

[root@ip-172-31-3-215 ~]#
```

5. Extract the java using the tar file to `/usr/java/`

```
[root@ip-172-31-3-215 ~]# tar zxvf jdk-8u161-linux-x64.tar.gz -C /usr/java/
```

```
[root@ip-172-31-3-215 ~]# tar zxvf jdk-8u161-linux-x64.tar.gz -C /usr/java/
jdk1.8.0_161/
jdk1.8.0_161/javafx-src.zip
jdk1.8.0_161/bin/
jdk1.8.0_161/bin/jmc
jdk1.8.0_161/bin/serialver
jdk1.8.0_161/bin/jmc.ini
jdk1.8.0_161/bin/jstack
jdk1.8.0_161/bin/rmiregistry
jdk1.8.0_161/bin/unpack200
jdk1.8.0_161/bin/jar
jdk1.8.0_161/bin/jps
jdk1.8.0_161/bin/bsimport
```

6. Exit from the root user, and download the **maven** from **ec2-user**.

```
[ec2-user@ip-172-31-3-215 ~]$ wget
http://www-us.apache.org/dist/maven/maven-3/3.5.4/binaries/apache-maven-3.5.4-bin.tar.gz
```

```
[ec2-user@ip-172-31-3-215 ~]$ wget http://www-us.apache.org/dist/maven/maven-3/3.5.4/binaries/apache-maven-3.5.4-bin.tar.gz
--2018-10-24 12:47:02-- http://www-us.apache.org/dist/maven/maven-3/3.5.4/binaries/apache-maven-3.5.4-bin.tar.gz
Resolving www-us.apache.org (www-us.apache.org)... 40.79.78.1
Connecting to www-us.apache.org (www-us.apache.org)|40.79.78.1|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 8842660 (8.4M) [application/x-gzip]
Saving to: 'apache-maven-3.5.4-bin.tar.gz'

100%[=====]
2018-10-24 12:47:02 (40.4 MB/s) - 'apache-maven-3.5.4-bin.tar.gz' saved [8842660/8842660]

[ec2-user@ip-172-31-3-215 ~]$
```

7. Untar the downloaded file

```
[ec2-user@ip-172-31-3-215 ~]$ tar -zxvf apache-maven-3.5.4-bin.tar.gz
```

```
[ec2-user@ip-172-31-3-215 ~]$ tar -zxvf apache-maven-3.5.4-bin.tar.gz
apache-maven-3.5.4/README.txt
apache-maven-3.5.4/LICENSE
apache-maven-3.5.4/NOTICE
apache-maven-3.5.4/lib/
apache-maven-3.5.4/lib/cdi-api.license
apache-maven-3.5.4/lib/commons-cli.license
apache-maven-3.5.4/lib/commons-io.license
apache-maven-3.5.4/lib/commons-lang3.license
apache-maven-3.5.4/lib/jcl-over-slf4j.license
apache-maven-3.5.4/lib/jsr250-api.license
```

8. Now add the java and maven path

```
vi .bashrc_profile
```

Once the file is opened add go to the end of the file, and insert below lines of code(**press i to enter**)

ec2-user@ip-172-31-3-215:~

```
# .bash_profile

# Set the aliases and Functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/.local/bin:$HOME/bin

export PATH
export JAVA_HOME=/usr/java/jdk1.8.0_161/
export JRE_HOME=/usr/java/jdk1.8.0_161/jre
export PATH=$PATH:/home/ec2-user/apache-maven-3.5.4/bin
export PATH=$JAVA_HOME/bin:$PATH
```

```
export JAVA_HOME=/usr/java/jdk1.8.0_161/
export JRE_HOME=/usr/java/jdk1.8.0_161/jre
export PATH=$PATH:/home/ec2-user/apache-maven-3.5.4/bin
export PATH=$JAVA_HOME/bin:$PATH
```

To exit, Press **Esc** > then type **:wq!** > and press **enter**

9. Enter the following command

```
source .bash_profile
```

10. Check the whether maven path is added or not, run folloing coommand

```
mvn
```

```
ec2-user@ip-172-31-3-215:~
[ec2-user@ip-172-31-3-215 ~]$ source .bash_profile
[ec2-user@ip-172-31-3-215 ~]$ mvn
[INFO] Scanning for projects...
[INFO] -----
[INFO] BUILD FAILURE
[INFO] -----
[INFO] Total time: 0.086 s
[INFO] Finished at: 2018-10-24T13:05:52Z
[INFO] -----
[ERROR] No goals have been specified for this build. You must specify a valid lifecycle phase or a goal in
n-artifact-id[:<plugin-version>]:<goal>. Available lifecycle phases are: validate, initialize, generate-sources,
, compile, process-classes, generate-test-sources, process-test-sources, generate-test-resources, process-test-resources,
are-package, package, pre-integration-test, integration-test, post-integration-test, verify, install, deploy
ite-deploy. -> [Help 1]
[ERROR]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.
[ERROR]
[ERROR] For more information about the errors and possible solutions, please read the following articles:
[ERROR] [Help 1] http://cwiki.apache.org/confluence/display/MAVEN/NoGoalSpecifiedException
[ec2-user@ip-172-31-3-215 ~]$
```

11. Now let's download the HBase-3.0.0

```
[ec2-user@ip-172-31-3-215 ~]$ git clone https://github.com/apache/hbase/
```

```
ec2-user@ip-172-31-3-215:~
[ec2-user@ip-172-31-3-215 ~]$ git clone https://github.com/apache/hbase/
Cloning into 'hbase'...
remote: Enumerating objects: 476, done.
remote: Counting objects: 100% (476/476), done.
remote: Compressing objects: 100% (253/253), done.
remote: Total 659506 (delta 212), reused 293 (delta 102), pack-reused 659030
Receiving objects: 100% (659506/659506), 293.96 MiB | 32.54 MiB/s, done.
Resolving deltas: 100% (309883/309883), done.
[ec2-user@ip-172-31-3-215 ~]$
```

12. We need to compile this repo first, follow the commands mentioned below.

```
[ec2-user@ip-172-31-3-215 ~]$ cd hbase
[ec2-user@ip-172-31-3-215 hbase]$ mvn -DskipTests=true package assembly:single
```



```
ec2-user@ip-172-31-3-215:~/hbase
```

```
[ec2-user@ip-172-31-3-215 hbase]$ mvn -DskipTests=true package assembly:single
```

This process will take some time(10-20 mins).

Once you see the build success, move ahead

```
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 08:29 min
[INFO] Finished at: 2018-10-24T13:20:01Z
[INFO] -----
[ec2-user@ip-172-31-3-215 hbase]$
```

After the compilation is successful, one tar file named **hbase-3.0.0-SNAPSHOT-bin.tar.gz** inside the target folder will be generated, this tar file contains the all the compiled jar of HBase.

13. So we copy this tar file to our home directory.

```
[ec2-user@ip-172-31-3-215 hbase]$ cp hbase-assembly/target/hbase-3.0.0-SNAPSHOT-bin.tar.gz /home/ec2-user/
```

```
[ec2-user@ip-172-31-3-215 hbase]$ cd
[ec2-user@ip-172-31-3-215 ~]$ ls
```

Now you see the hbase-3.0.0-SNAPSHOT-bin.tar.gz file the dome directory.

```
[ec2-user@ip-172-31-3-215 hbase]$ cp hbase-assembly/target/hbase-3.0.0-SNAPSHOT-bin.tar.gz /home/ec2-user/
[ec2-user@ip-172-31-3-215 hbase]$ ls
bin                hbase-assembly      hbase-common        hbase-http          hbase-procedure
CHANGES.txt       hbase-backup         hbase-endpoint      hbase-it            hbase-protocol
conf              hbase-build-configuration hbase-examples      hbase-mapreduce     hbase-protocol-
dev-support        hbase-build-support  hbase-external-blockcache hbase-metrics       hbase-replicati
hbase-annotations hbase-checkstyle     hbase-hadoop2-compat hbase-metrics-api   hbase-resource-
hbase-archetypes  hbase-client         hbase-hadoop-compat hbase-native-client hbase-rest
[ec2-user@ip-172-31-3-215 hbase]$ cd
[ec2-user@ip-172-31-3-215 ~]$ ls
apache-maven-3.5.4  apache-maven-3.5.4-bin.tar.gz  hbase  hbase-3.0.0-SNAPSHOT-bin.tar.gz
```



14. Let's tar the **hbase-3.0.0-SNAPSHOT-bin.tar.gz** file

```
[ec2-user@ip-172-31-3-215 ~]$ tar -xvf hbase-3.0.0-SNAPSHOT-bin.tar.gz
```

```
[ec2-user@ip-172-31-3-215 ~]$ tar -xvf hbase-3.0.0-SNAPSHOT-bin.tar.gz
hbase-3.0.0-SNAPSHOT/LICENSE.txt
hbase-3.0.0-SNAPSHOT/NOTICE.txt
hbase-3.0.0-SNAPSHOT/LEGAL
hbase-3.0.0-SNAPSHOT/README.txt
hbase-3.0.0-SNAPSHOT/conf/
hbase-3.0.0-SNAPSHOT/conf/hadoop-metrics2-hbase.properties
hbase-3.0.0-SNAPSHOT/conf/hbase-env.cmd
hbase-3.0.0-SNAPSHOT/conf/hbase-env.sh
hbase-3.0.0-SNAPSHOT/conf/hbase-policy.xml
```

15. Create a directory named **HbaseData**, this is where hbase will store data, in the local mode

```
[ec2-user@ip-172-31-3-215 ~]$ mkdir HbaseData
```

 ec2-user@ip-172-31-3-215:~

```
[ec2-user@ip-172-31-3-215 ~]$ mkdir HbaseData
[ec2-user@ip-172-31-3-215 ~]$
```

14. Now provide the path to this folder to the hbase-site.xml file

```
[ec2-user@ip-172-31-3-215 ~]$ vi hbase-3.0.0-SNAPSHOT/conf/hbase-site.xml
```

Once the file is opened to paste (press i) the following property, as shown the below image.

```
<property>
  <name>hbase.rootdir</name>
  <value>file:///home/ec2-user/HbaseData</value>
</property>
```



ec2-user@ip-172-31-3-215:~

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
/**
 * Licensed to the Apache Software Foundation (ASF) under one
 * or more contributor license agreements. See the NOTICE file
 * distributed with this work for additional information
 * regarding copyright ownership. The ASF licenses this file
 * to you under the Apache License, Version 2.0 (the
 * "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing,
 * distributed under the License is distributed on an "AS IS"
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express
 * See the License for the specific language governing permissions
 * and limitations under the License.
 */
-->
<configuration>
<property>
  <name>hbase.rootdir</name>
  <value>file:///home/ec2-user/HbaseData</value>
</property>
</configuration>
~
~
~
~
```

Since we have hbase-3.0 installed in local mode, to open hbase shell or run a jar file we need to execute the corresponding commands from the bin folder.

16. Now let's verify the successful installation of HBase.

```
[ec2-user@ip-172-31-3-215 ~]$ cd hbase-3.0.0-SNAPSHOT/bin
```

```
[ec2-user@ip-172-31-3-215 ~]$ cd hbase-3.0.0-SNAPSHOT/bin
[ec2-user@ip-172-31-3-215 bin]$ ls
considerAsDead.sh      hbase              hbase-config.cmd  hbase-jruby      master-backup.sh  replication        start-hbase.sh  zookeepers.sh
draining_servers.rb    hbase-cleanup.sh  hbase-config.sh   hmr.rb           region_mover.rb   rolling-restart.sh stop-hbase.cmd   stop-hbase.sh
get-active-master.rb    hbase.cmd          hbase-daemon.sh   local-master-backup.sh regionServers.sh  shutdown_regionserver.rb stop-hbase.sh
graceful_stop.sh        hbase-common.sh    hbase-daemons.sh local-regionServers.sh region_status.rb  start-hbase.cmd   test
```

17 . First, start the HMaster process,

```
[ec2-user@ip-172-31-3-215 bin]$ ./start-hbase.sh
```

Verify whether it is running

```
[ec2-user@ip-172-31-3-215 bin]$ jps
```

```
[ec2-user@ip-172-31-3-215 bin]$ ./start-hbase.sh
running master, logging to /home/ec2-user/hbase-3.0.0-SNAPSHOT/bin/../logs/hbase-ec2-user-master-ip-172-31-3-215.ec2.internal.out
[ec2-user@ip-172-31-3-215 bin]$ jps
30676 Jps
30383 HMaster
[ec2-user@ip-172-31-3-215 bin]$
```

18. Now, let's download Spark also

First, go to the home dir

```
[ec2-user@ip-172-31-3-215 bin]$ cd
```

```
[ec2-user@ip-172-31-3-215 bin]$ cd
[ec2-user@ip-172-31-3-215 ~]$
```

19. Now download spark-2.1.1

```
[ec2-user@ip-172-31-3-215 ~]$ wget
https://archive.apache.org/dist/spark/spark-2.1.1/spark-2.1.1-bin-hadoop2.7.tgz
```

20. Once the compressed file is downloaded, let's extract it

```
[ec2-user@ip-172-31-3-215 ~]$ tar -xvf spark-2.1.1-bin-hadoop2.7.tgz
```

```
[ec2-user@ip-172-31-3-215 ~]$ tar -xvf spark-2.1.1-bin-hadoop2.7.tgz
spark-2.1.1-bin-hadoop2.7/
spark-2.1.1-bin-hadoop2.7/NOTICE
spark-2.1.1-bin-hadoop2.7/jars/
spark-2.1.1-bin-hadoop2.7/jars/bonecp-0.8.0.RELEASE.jar
spark-2.1.1-bin-hadoop2.7/jars/commons-net-2.2.jar
spark-2.1.1-bin-hadoop2.7/jars/javax.servlet-api-3.1.0.jar
spark-2.1.1-bin-hadoop2.7/jars/hadoop-annotations-2.7.3.jar
spark-2.1.1-bin-hadoop2.7/jars/hadoop-hdfs-2.7.3.jar
spark-2.1.1-bin-hadoop2.7/jars/oro-2.0.8.jar
spark-2.1.1-bin-hadoop2.7/jars/xercesImpl-2.9.1.jar
spark-2.1.1-bin-hadoop2.7/jars/antlr-runtime-3.4.jar
```

21. Let's create a soft link which will be used by HBase-3.0

```
[ec2-user@ip-172-31-3-215 ~]$ ln -s spark-2.1.1-bin-hadoop2.7 spark
```

```
[ec2-user@ip-172-31-3-215 ~]$ ln -s spark-2.1.1-bin-hadoop2.7 spark
[ec2-user@ip-172-31-3-215 ~]$
```

22. Let's add the spark soft link created in the above step in the **hbase** starting script

First open the file in, vi editor

```
[ec2-user@ip-172-31-3-215 ~]$ vi hbase-3.0.0-SNAPSHOT/bin/hbase
```

Once the file is opened, type **:set nu >** press **Enter**

```
284 fi
285 for f in /home/ec2-user/spark/jars/*.jar; do
286     CLASSPATH=${CLASSPATH}:${f};
287 done
288 for f in "${HBASE_HOME}"/lib/client-facing-thirdparty/*.jar; do
289     if [[ ! "${f}" =~ ^.*\/htrace-core-3.*\.jar$ ]] && \
290        [ "${f}" != "htrace-core.jar" ] && \
291        [[ ! "${f}" =~ ^.*\/slf4j-log4j.*$ ]]; then
292         CLASSPATH="${CLASSPATH}:${f}"
```



Insert the following line of code in line number 285. (press i to paste)

```
for f in /home/ec2-user/spark/jars/*.jar; do
    CLASSPATH=${CLASSPATH}:$f;
done
```


Once you have copied the code to exit, Press Esc > type **:wq!** > press exit

23. Now let's Set the spark home path

```
[ec2-user@ip-172-31-3-215 ~]$ vi .bash_profile
```

Once the file is opened add the following line of code (press i to paste)

```
export SPARK_HOME=/home/ec2-user/spark
export PATH=$SPARK_HOME/bin:$PATH
```

 ec2-user@ip-172-31-3-215:~

```
# .bash_profile

# Set the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/.local/bin:$HOME/bin

export PATH
export JAVA_HOME=/usr/java/jdk1.8.0_161/
export JRE_HOME=/usr/java/jdk1.8.0_161/jre
export PATH=$PATH:/home/ec2-user/apache-maven-3.5.4/bin
export PATH=$JAVA_HOME/bin:$PATH
export SPARK_HOME=/home/ec2-user/spark
export PATH=$SPARK_HOME/bin:$PATH
```

24. Once you have copied the code to exit, Press Esc > type **:wq!** > press exit

Source the bash_profile file

```
[ec2-user@ip-172-31-3-215 ~]$ source .bash_profile
```

Verify the spark path

```
echo $SPARK_HOME
```




BITS Pilani
Pilani | Dubai | Goa | Hyderabad

UpGrad

```
[ec2-user@ip-172-31-3-215 ~]$ source .bash_profile  
[ec2-user@ip-172-31-3-215 ~]$ echo $SPARK_HOME  
/home/ec2-user/spark  
[ec2-user@ip-172-31-3-215 ~]$
```

That's it for the setup.