Hortonworks HDP on Amazon Web Services

1 INTRODUCTION

1.1 HORTONWORKS HDP

Hortonworks Data Platform (HDP) provides an enterprise ready data Hadoop platform that enables organizations to deploy the next generation Hadoop enterprise data platform.

Hortonworks Data Platform integrates with, and augments, your existing applications and systems so that you can take advantage of Hadoop with only minimal change to existing data architectures and skillsets. Deploy HDP in-cloud, on-premise or from an appliance across both Linux and Windows.

For more details, visit: http://hortonworks.com/hdp/

1.1.1 Ambari

Ambari makes Hadoop management simpler by providing a consistent, secure platform for operational control. Ambari provides an intuitive Web UI as well as a robust REST API, which is particularly useful for automating cluster operations.

For more details, visit: http://hortonworks.com/hadoop/ambari/

1.2 AMAZON AWS

Amazon Web Services (AWS) provides on-demand computing resources and services in the cloud, with pay-as-you-go pricing.

For more details, visit: http://docs.aws.amazon.com/gettingstarted/latest/awsgsg-intro/gsg-aws-intro.html

1.2.1 AWS EC2

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers.

Benefits of EC2 services include:

- Elastic Web-Scale Computing
- Completely Controlled
- Flexible Cloud Hosting Services
- Designed for use with other Amazon Web Services
- Reliable, Secure and Inexpensive
- Easy to Start

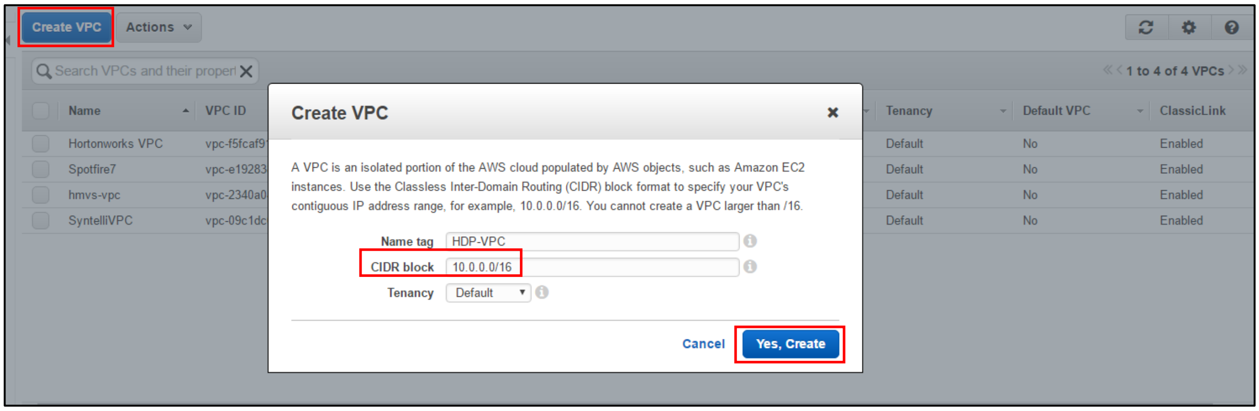
For more details, visit: https://aws.amazon.com/ec2/

2 PREPARING YOUR INSTANCE

2.1 PREPARING VPC FOR INSTANCE AND CLUSTER

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the Amazon Web Services (AWS) cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.

For more details, visit: https://aws.amazon.com/vpc/



Name	VPC ID	Tenancy	Default VPC	ClassicLink
Hortonworks VPC	vpc-f5fcaf9	Default	No	Enabled
Spotfire7	vpc-e19283	Default	No	Enabled
hmvs-vpc	vpc-2340a0	Default	No	Enabled
SyntelliVPC	vpc-09c1dc	Default	No	Enabled

Create VPC

A VPC is an isolated portion of the AWS cloud populated by AWS objects, such as Amazon EC2 instances. Use the Classless Inter-Domain Routing (CIDR) block format to specify your VPC's contiguous IP address range, for example, 10.0.0.0/16. You cannot create a VPC larger than /16.

Name tag: HDP-VPC

CIDR block: 10.0.0.0/16

Tenancy: Default

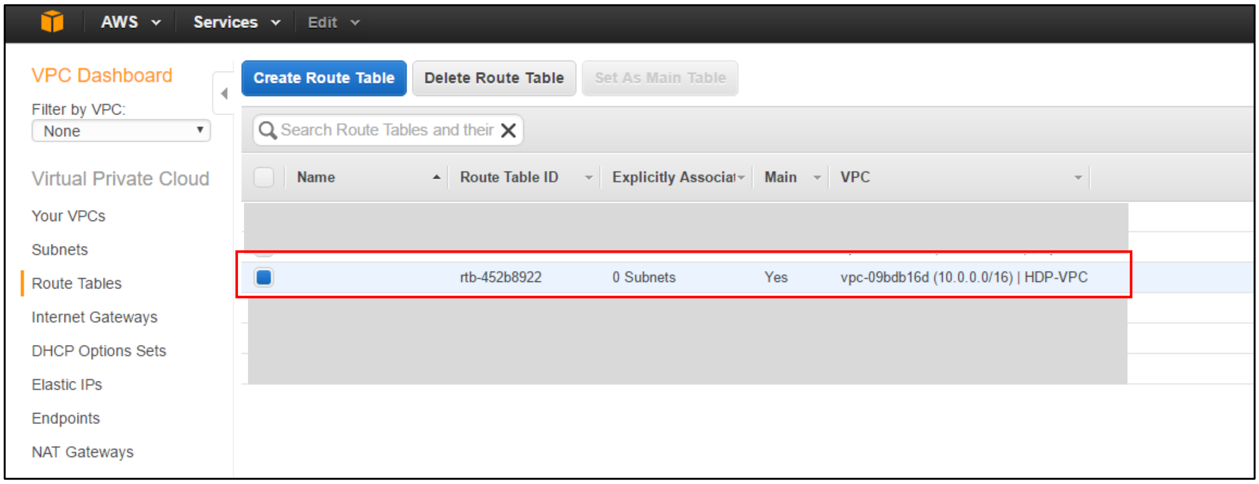
Figure 2.1.1

When I was creating a VPC, I had already created an AWS account. Once you have access to AWS Management Console, you can navigate to Networking. Under Networking, you will find VPC.

You can create a VPC using VPC Wizard or create one manually.

To create a VPC manually, follow the steps mentioned below:

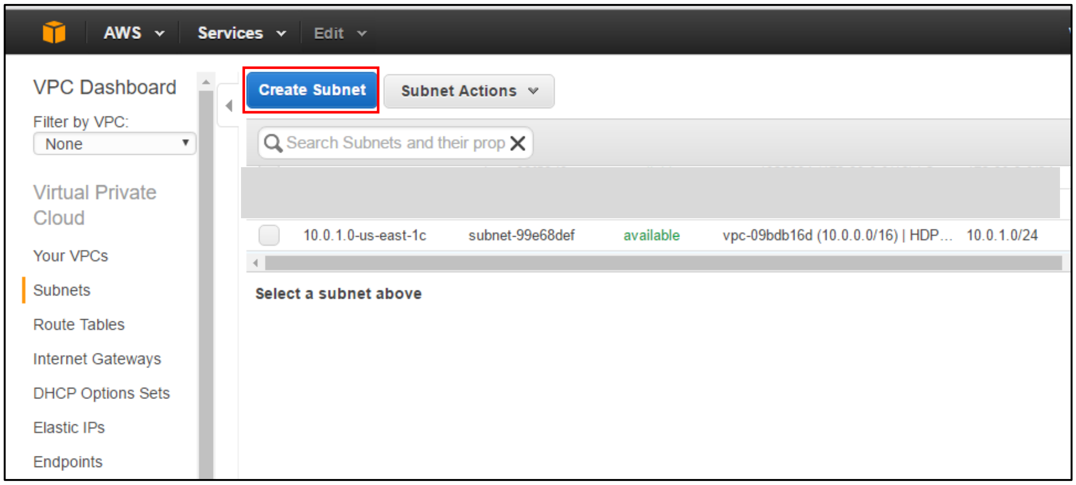
1. Navigate to "Your VPCs" on the VPC dashboard.
2. Click "Create VPC" as highlighted, in Figure 2.1.1.
3. Provide a name for your VPC.
4. Next, provide a CIDR block. In my case, I entered (10.0.0.0/16)
   To learn more about CIDR, visit: https://en.wikipedia.org/wiki/Classless_Inter-Domain_Routing
5. Leave the Tenancy as "Default". If choose "Dedicated", then charges apply.
   To learn more about Dedicated, visit:
   http://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/dedicated-instance.html
6. Finally click, "Yes, Create".



Name	Route Table ID	Explicitly Associat	Main	VPC
	rtb-452b8922	0 Subnets	Yes	vpc-09bdb16d (10.0.0.0/16) | HDP-VPC

Figure 2.1.2

In Figure 2.1.2, if you navigate to Route Tables under VPC Dashboard, you will notice that a new Route Table is created along the VPC.



10.0.1.0-us-east-1c	subnet-99e68def	available	vpc-09bdb16d (10.0.0.0/16) | HDP...	10.0.1.0/24

Select a subnet above

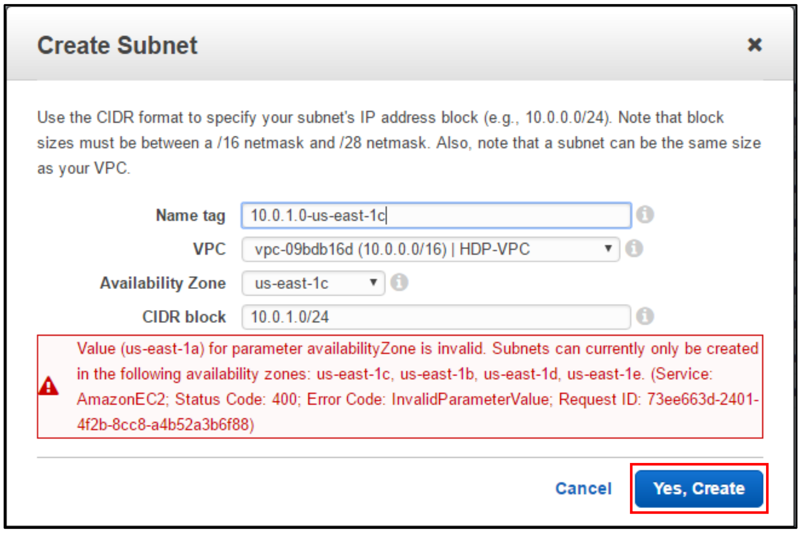
Figure 2.1.3

After creating a VPC, you can add one or more subnets in each Availability Zone. Each subnet must reside entirely within one Availability Zone and cannot span zones. Availability Zones are distinct locations that are engineered to be isolated from failures in other Availability Zones.

Navigate to Subnets under VPC Dashboard, and click “Create Subnet” as highlighted in Figure 2.1.3.

- Add a name-tag.
- Select the VPC to which the subnet will be linked to.
- Choose an availability zone, where the EC2 instances will be launched.
  To read more about availability zone in AWS, visit:
  http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html
- Choose a CIDR block. In my case, I entered (10.0.1.0/24)
- Click “Yes, Create” as highlighted in Figure 2.1.4

After a successful subnet creation, you should be able to verify it, as shown in Figure 2.1.5



Create Subnet

Use the CIDR format to specify your subnet's IP address block (e.g., 10.0.0.0/24). Note that block sizes must be between a /16 netmask and /28 netmask. Also, note that a subnet can be the same size as your VPC.

Name tag: 10.0.1.0-us-east-1c

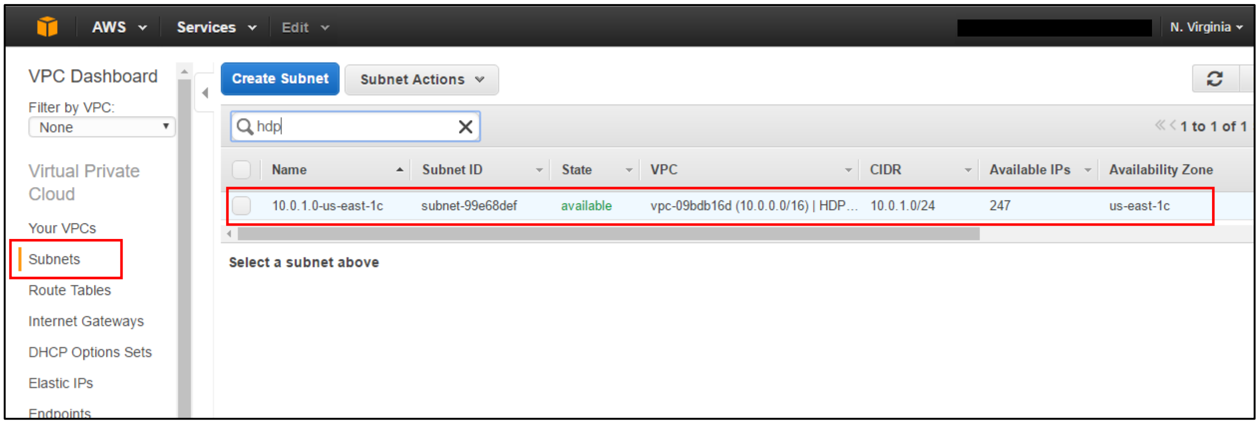
VPC: vpc-09bdb16d (10.0.0.0/16) | HDP-VPC

Availability Zone: us-east-1c

CIDR block: 10.0.1.0/24

Value (us-east-1a) for parameter availabilityZone is invalid. Subnets can currently only be created in the following availability zones: us-east-1c, us-east-1b, us-east-1d, us-east-1e. (Service: AmazonEC2; Status Code: 400; Error Code: InvalidParameterValue; Request ID: 73ee663d-2401-4f2b-8cc8-a4b52a3b6f88)

Figure 2.1.4



[redacted]

Search: hdp

Name	Subnet ID	State	VPC	CIDR	Available IPs	Availability Zone
10.0.1.0-us-east-1c	subnet-99e68def	available	vpc-09bdb16d (10.0.0.0/16) | HDP...	10.0.1.0/24	247	us-east-1c

Select a subnet above

Figure 2.1.5

An Internet gateway is a horizontally scaled, redundant, and highly available VPC component that allows communication between instances in your VPC and the Internet. It therefore imposes no availability risks or bandwidth constraints on your network traffic.

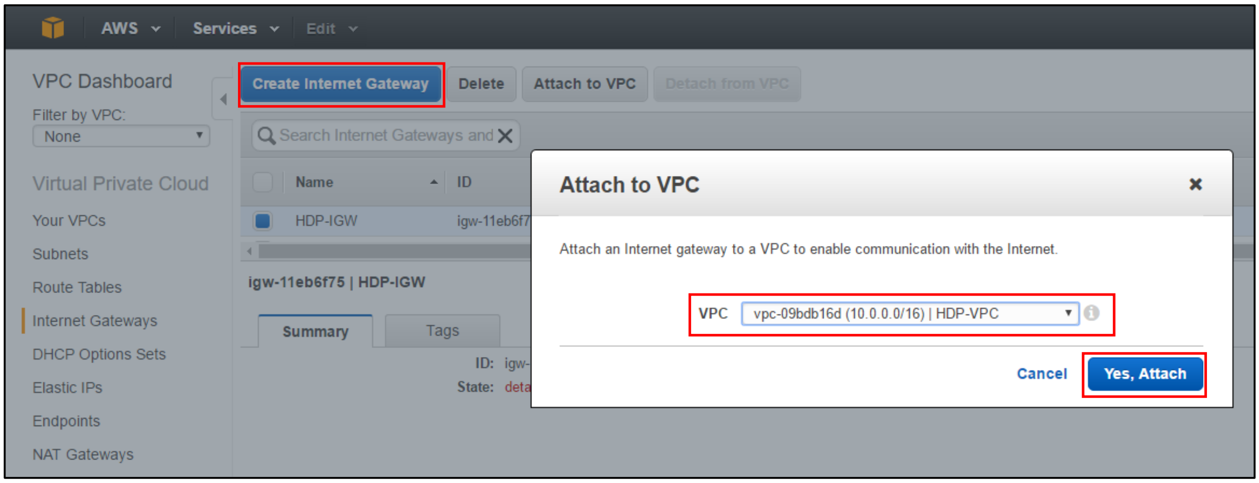
An Internet gateway serves two purposes:

- Provide a target in your VPC route tables for Internet-routable traffic
- Perform network address translation (NAT) for instances that have been assigned public IP addresses.

Navigate to Internet Gateways under VPC Dashboard and click “Create Internet Gateway” as highlighted in Figure 2.1.6.

Provide a name for the Internet Gateway and click “Yes, Create” as highlighted in Figure 2.1.7.

Choose VPC to which, the newly created Internet Gateway will be attached to. Next, click “Yes, Attach” as highlighted in Figure 2.1.6.

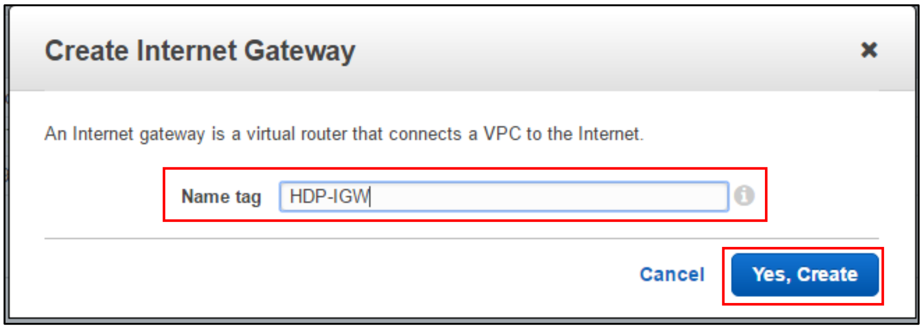


Attach to VPC

Attach an Internet gateway to a VPC to enable communication with the Internet.

VPC: vpc-09bdb16d (10.0.0.0/16) | HDP-VPC

Figure 2.1.6



Create Internet Gateway

An Internet gateway is a virtual router that connects a VPC to the Internet.

Name tag: HDP-IGW

Figure 2.1.7

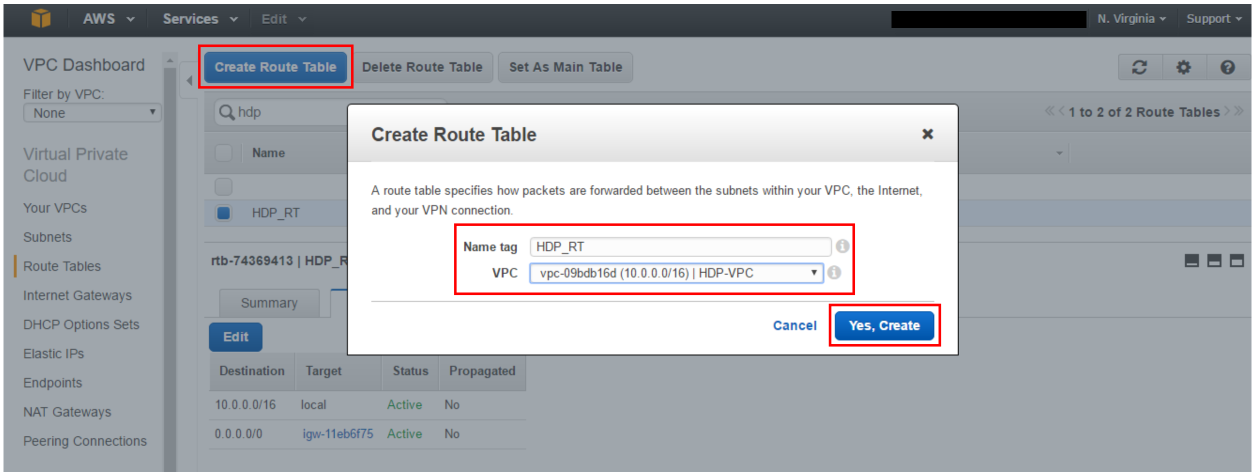
A route table contains a set of rules, called routes, that are used to determine where network traffic is directed.

Each subnet in your VPC must be associated with a route table; the table controls the routing for the subnet. A subnet can only be associated with one route table at a time, but you can associate multiple subnets with the same route table.

Navigate to Route Tables under VPC Dashboard and click “Create Route Table” as highlighted in Figure 2.1.8.

Provide a name for the Route Table and attach it to the VPC, that was created earlier. Click “Yes, Create”.

After the Route Table is created, you need to add an Internet route to it.



[redacted]

Create Route Table

A route table specifies how packets are forwarded between the subnets within your VPC, the Internet, and your VPN connection.

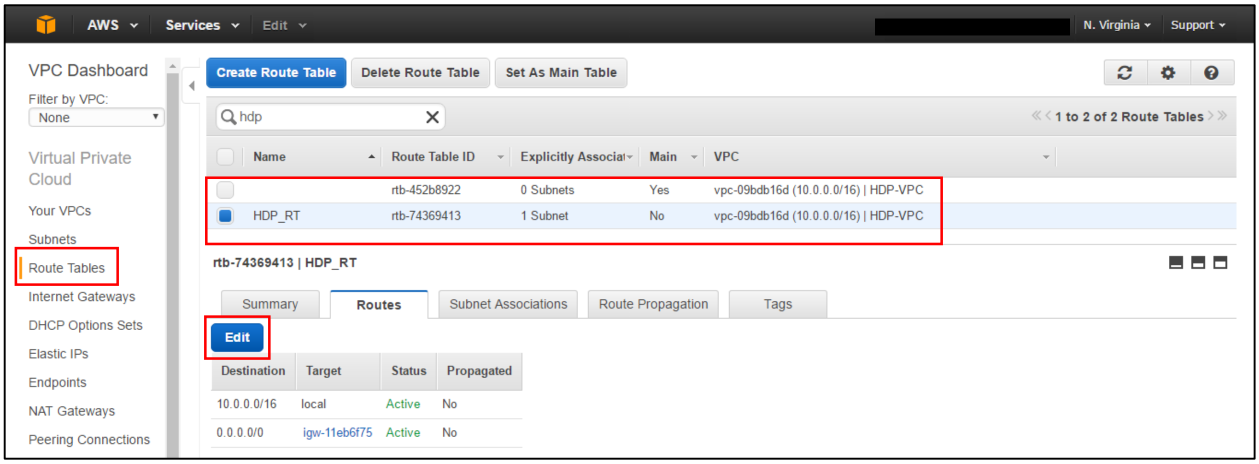
Name tag: HDP_RT

VPC: vpc-09bdb16d (10.0.0.0/16) | HDP-VPC

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	igw-11eb6f75	Active	No

Figure 2.1.8

You can add an Internet Gateway route to your Route Table as highlighted in Figure 2.1.9. Select the route table and then select "Routes" tab as shown in Figure 2.1.9.



[redacted]

Name	Route Table ID	Explicitly Associat	Main	VPC
	rtb-452b8922	0 Subnets	Yes	vpc-09bdb16d (10.0.0.0/16) | HDP-VPC
HDP_RT	rtb-74369413	1 Subnet	No	vpc-09bdb16d (10.0.0.0/16) | HDP-VPC

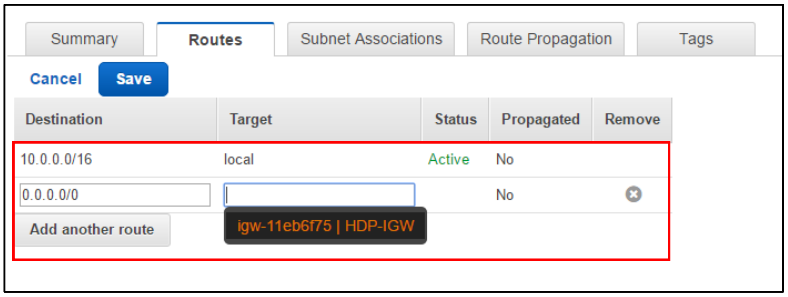
rtb-74369413 | HDP_RT

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	igw-11eb6f75	Active	No

Figure 2.1.9

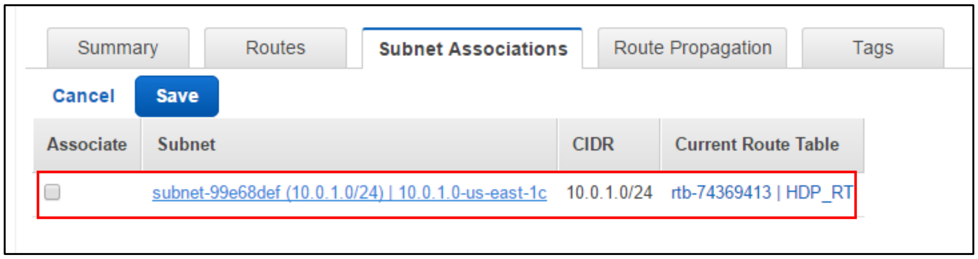
Click "Edit" and CIDR information. I entered (0.0.0.0/0) in the destination field, because I wanted the all the instances have Internet access as highlighted in Figure 2.1.10.

The Internet Gateway which was previous configured, was attached as a target. Click "Save".



Destination	Target	Status	Propagated	Remove
10.0.0.0/16	local	Active	No	
0.0.0.0/0			No	⊗

Figure 2.1.10

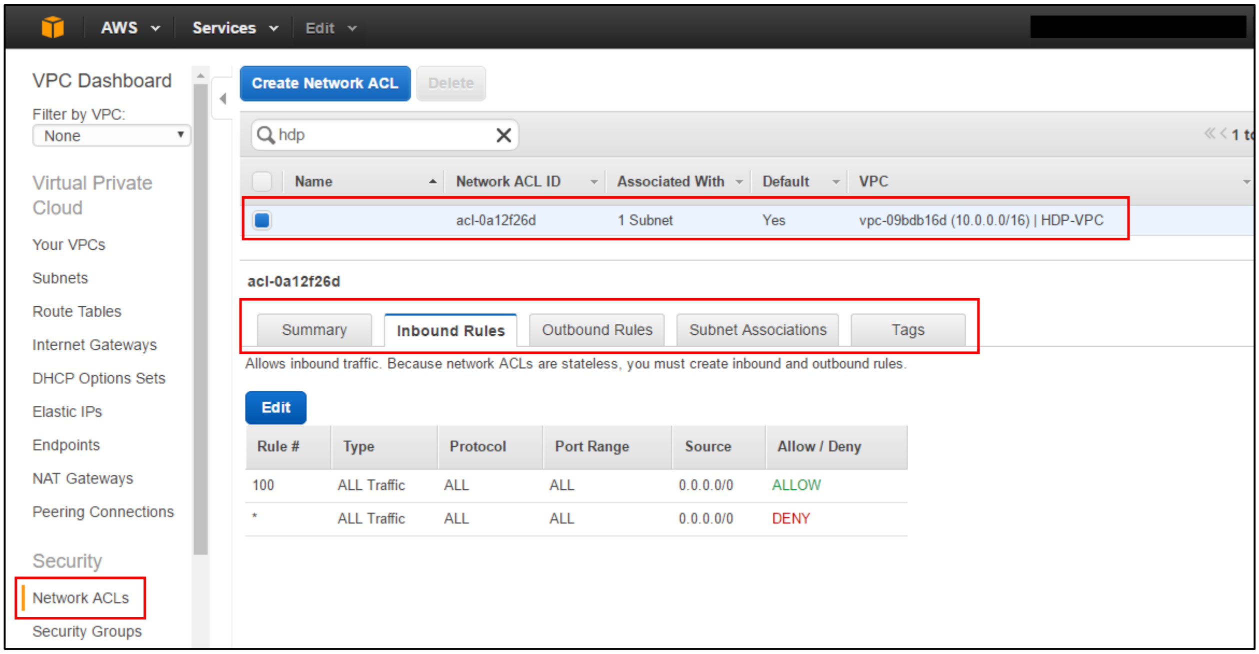


Associate	Subnet	CIDR	Current Route Table
☐	subnet-99e68def (10.0.1.0/24) | 10.0.1.0-us-east-1c	10.0.1.0/24	rtb-74369413 | HDP_RT

Figure 2.1.11

Next, Click “Subnet Association” tab and assign the subnet you created earlier to your Route Table. “Check” the subnet and click “Save” as highlighted in Figure 2.1.11

If you have associated the subnet to your Route Table, you will able to see it as shown in Figure 2.1.9, which states “1 Subnet” explicitly associated.



[redacted]

	Name	Network ACL ID	Associated With	Default	VPC
■		acl-0a12f26d	1 Subnet	Yes	vpc-09bdb16d (10.0.0.0/16) | HDP-VPC

acl-0a12f26d

Allows inbound traffic. Because network ACLs are stateless, you must create inbound and outbound rules.

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
100	ALL Traffic	ALL	ALL	0.0.0.0/0	ALLOW
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY

Figure 2.1.12

When a VPC is created, a Network ACL is also created for the VPC by default as highlighted in Figure 2.1.12. The Inbound and Outbound Rules are pre-defined. We can add rules if needed.

If you select the “Subnet Association” tab, you will notice that the subnet you created earlier is attached to the Network ACL.

To learn more about Network ACL, visit:
http://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/VPC_ACLs.html

Once you have completed all the steps to creating a VPC, you can proceed to launch an instance on AWS EC2.

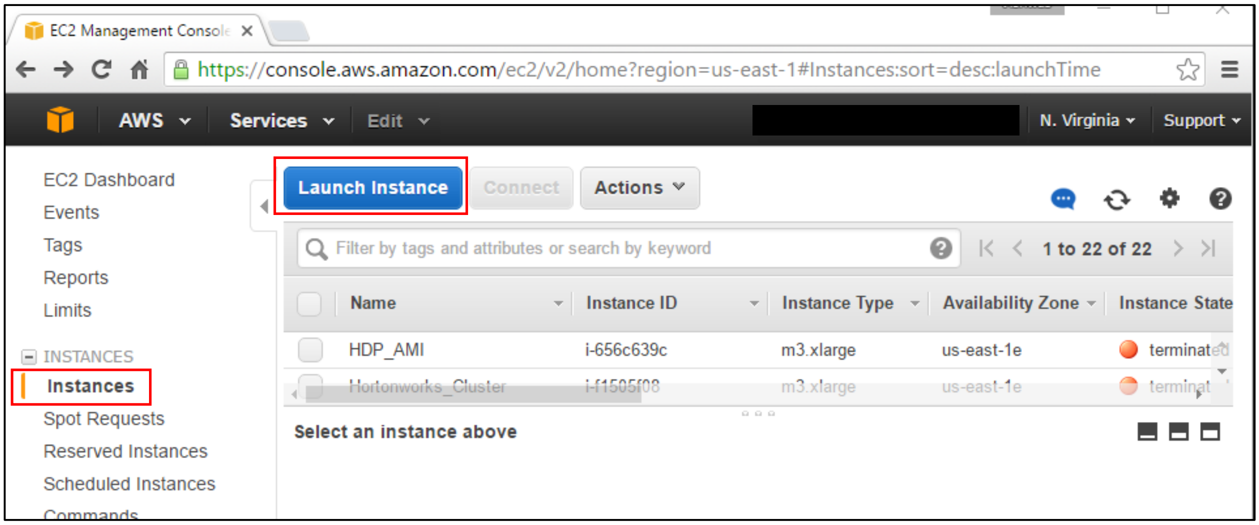
2.2 Preparing Your Instance

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases.

Instance types comprise varying combinations of CPU, memory, storage, and networking capacity and give you the flexibility to choose the appropriate mix of resources for your applications. Each instance type includes one or more instance sizes, allowing you to scale your resources to the requirements of your target workload.

I chose M3 instance type. M3 instance types and provides a balance of compute, memory, and network resources, and it is a good choice for many applications.

For more details about EC2 instance types, visit: https://aws.amazon.com/ec2/instance-types/



https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#Instances:sort=desc:launchTime

[redacted]

	Name	Instance ID	Instance Type	Availability Zone	Instance State
□	HDP_AMI	i-656c639c	m3.xlarge	us-east-1e	● terminated
□	Hortonworks_Cluster	i-f1505f08	m3.xlarge	us-east-1e	● terminat

Select an instance above

Figure 2.2.1

Navigate to EC2 Management Console from AWS Home screen.

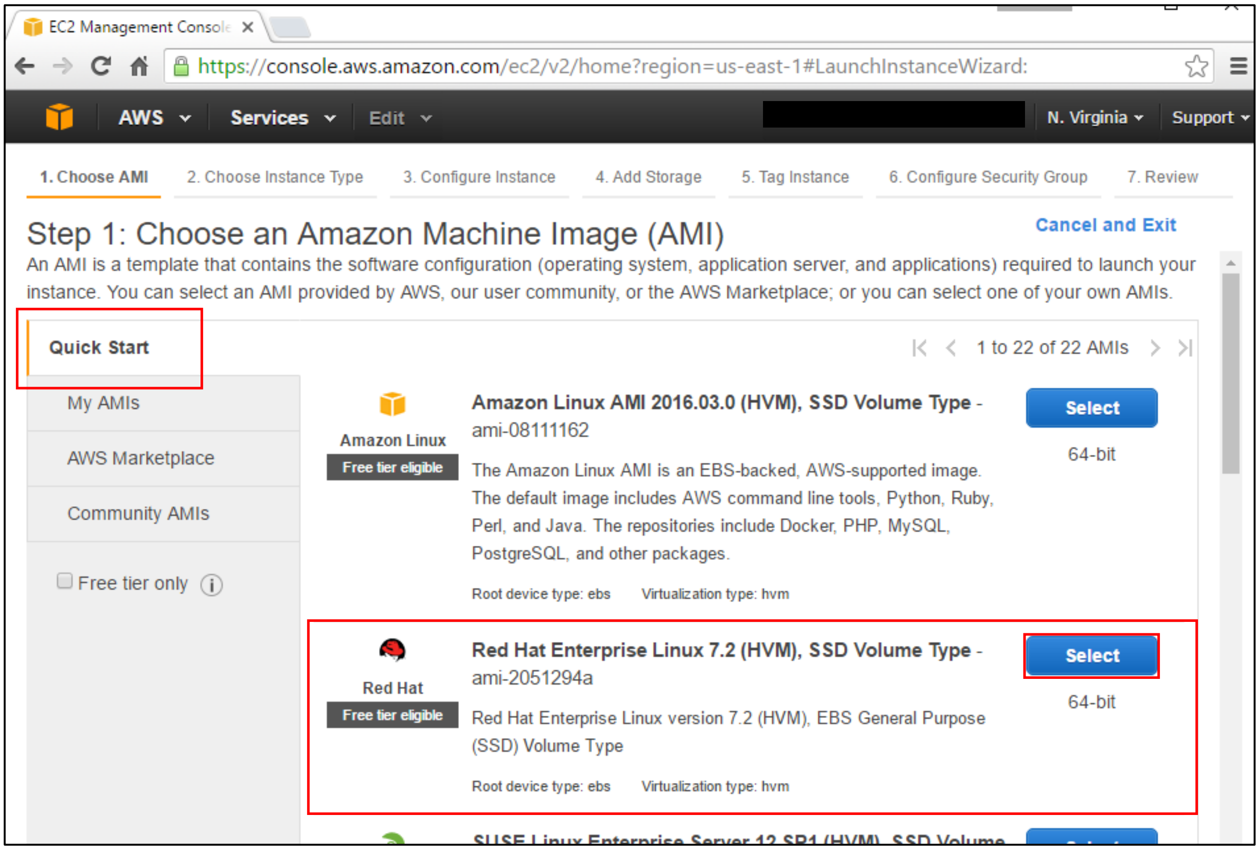
Under EC2 Dashboard, click “Instances” and click “Launch Instance” as highlighted in Figure 2.2.1.

The next screen is shown Figure 2.2.2. We need to choose an Amazon Machine Image.

An Amazon Machine Image (AMI) provides the information required to launch an instance, which is a virtual server in the cloud. You specify an AMI when you launch an instance, and you can launch as many instances from the AMI as you need.

An AMI includes the following:

- A template for the root volume for the instance (for example, an operating system, an application server, and applications)
- Launch permissions that control which AWS accounts can use the AMI to launch instances.
- A block device mapping that specifies the volumes to attach to the instance when it's launched



EC2 Management Console

https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#LaunchInstanceWizard:

AWS | Services | Edit | [redacted] | N. Virginia | Support

1. Choose AMI | 2. Choose Instance Type | 3. Configure Instance | 4. Add Storage | 5. Tag Instance | 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.

Quick Start | My AMIs | AWS Marketplace | Community AMIs

☐ Free tier only

1 to 22 of 22 AMIs

Amazon Linux — Free tier eligible

Amazon Linux AMI 2016.03.0 (HVM), SSD Volume Type - ami-08111162

The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.

Root device type: ebs   Virtualization type: hvm

Select — 64-bit

Red Hat — Free tier eligible

Red Hat Enterprise Linux 7.2 (HVM), SSD Volume Type - ami-2051294a

Red Hat Enterprise Linux version 7.2 (HVM), EBS General Purpose (SSD) Volume Type

Root device type: ebs   Virtualization type: hvm

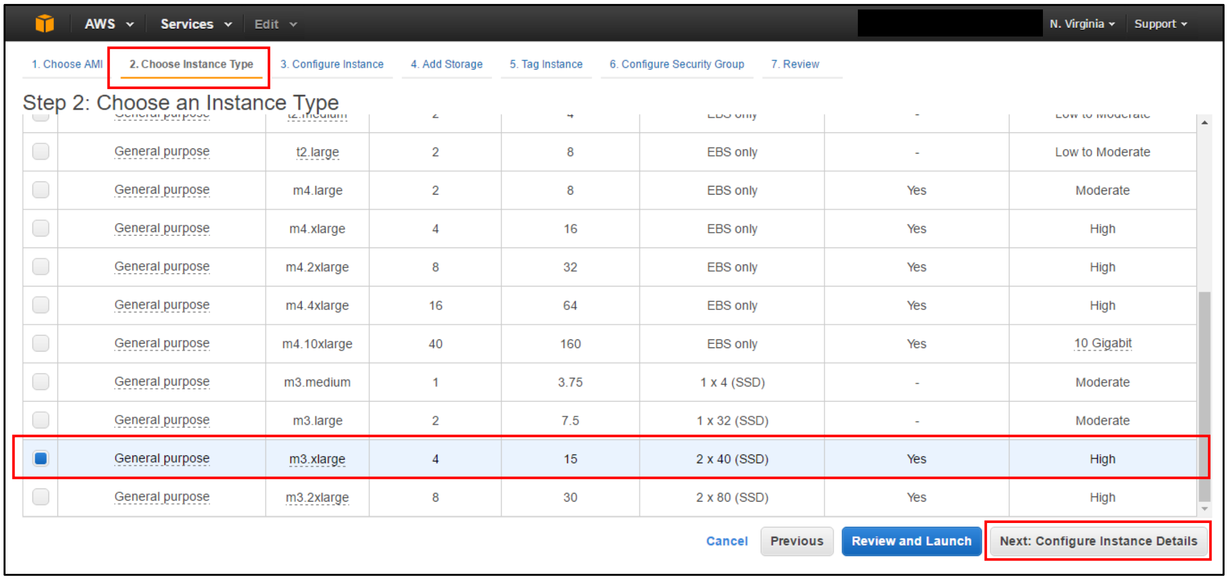
Select — 64-bit

SUSE Linux Enterprise Server 12 SP1 (HVM), SSD Volume

Figure 2.2.2

I chose Red Hat Enterprise Linux (RHEL) 7.2 as highlighted in Figure 2.2.2. Most of the generic AMIs are present in “Quick Start”. Click “Select”

The next screen will ask you to choose an instance type as shown in Figure 2.2.3.



AWS | Services | Edit | [redacted] | N. Virginia | Support

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

Step 2: Choose an Instance Type

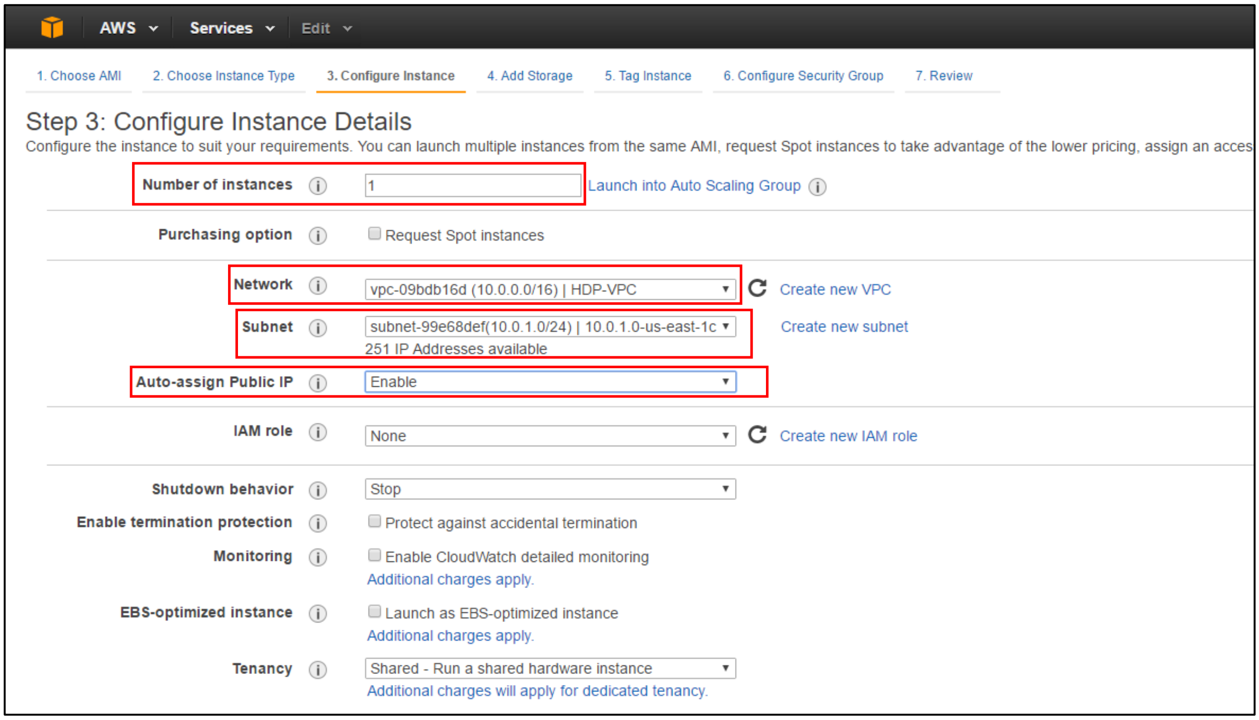
	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate
☐	General purpose	t2.large	2	8	EBS only	-	Low to Moderate
☐	General purpose	m4.large	2	8	EBS only	Yes	Moderate
☐	General purpose	m4.xlarge	4	16	EBS only	Yes	High
☐	General purpose	m4.2xlarge	8	32	EBS only	Yes	High
☐	General purpose	m4.4xlarge	16	64	EBS only	Yes	High
☐	General purpose	m4.10xlarge	40	160	EBS only	Yes	10 Gigabit
☐	General purpose	m3.medium	1	3.75	1 x 4 (SSD)	-	Moderate
☐	General purpose	m3.large	2	7.5	1 x 32 (SSD)	-	Moderate
☑	General purpose	m3.xlarge	4	15	2 x 40 (SSD)	Yes	High
☐	General purpose	m3.2xlarge	8	30	2 x 80 (SSD)	Yes	High

Cancel | Previous | Review and Launch | Next: Configure Instance Details

Figure 2.2.3

I chose "m3.xlarge" with 4 virtual CPUs, 15 GB of RAM and an instance storage of 2 X 40 GB SSD (EBS Optimized). The instance also provides high network performance.

Click "Next: Configure Instance Details" to proceed.



Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an acces

Number of instances: 1 — Launch into Auto Scaling Group

Purchasing option: □ Request Spot instances

Network: vpc-09bdb16d (10.0.0.0/16) | HDP-VPC — Create new VPC

Subnet: subnet-99e68def(10.0.1.0/24) | 10.0.1.0-us-east-1c
251 IP Addresses available — Create new subnet

Auto-assign Public IP: Enable

IAM role: None — Create new IAM role

Shutdown behavior: Stop

Enable termination protection: □ Protect against accidental termination

Monitoring: □ Enable CloudWatch detailed monitoring
Additional charges apply.

EBS-optimized instance: □ Launch as EBS-optimized instance
Additional charges apply.

Tenancy: Shared - Run a shared hardware instance
Additional charges will apply for dedicated tenancy.

Figure 2.2.4

In Configure Instance Details, we specify the number of instances, type of network and subnet we want to launch. Shown in Figure 2.2.4

I chose one instance, since I need to configure the instance for launching HDP on AWS.

For network, I selected the VPC that I had created in the earlier section.

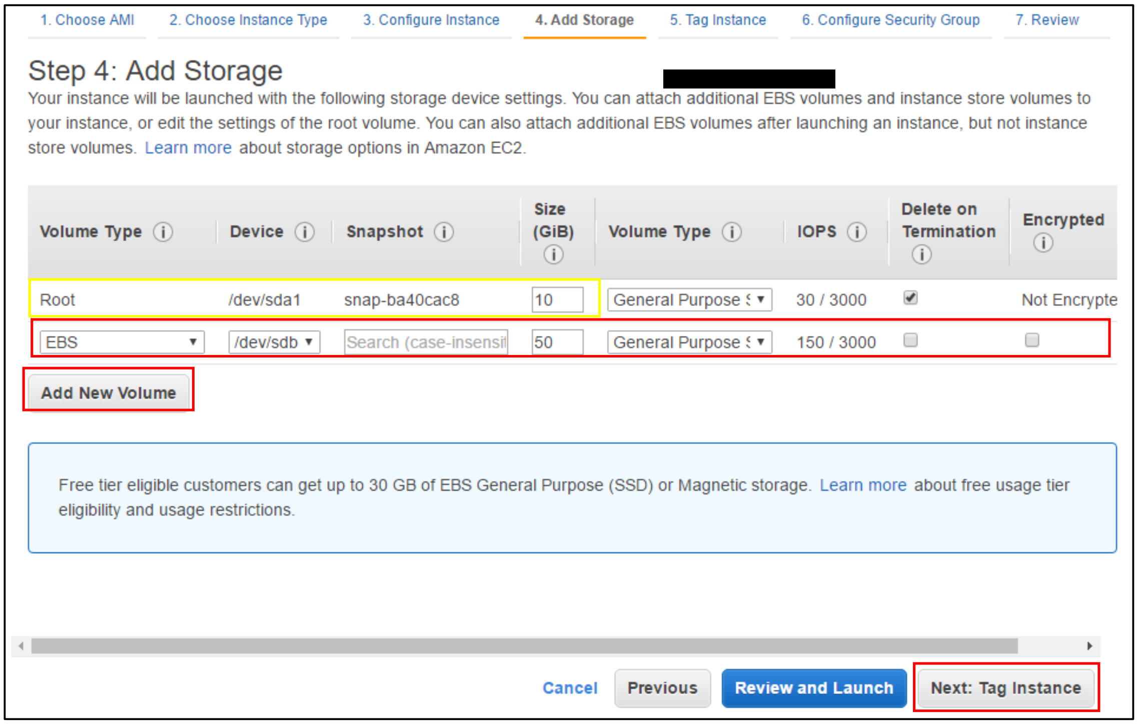
I attached the subnet, which I created in the previous section. Also, choose "Enable" for "Auto-assign Public IP". This will assign a Public IP, which will be used to SSH into the instance.

Next step in launching an instance is adding storage. The instance selected has a default volume 10 GB, which is set for "Linux Root", highlighted in yellow.

Click "Add New Volume".

A new list populates. Select "EBS" as the volume type. The device, is the path to the disk space as highlighted in Figure 2.2.5. I selected my path to "/dev/sdb". I entered my volume size as 50 GB.

To learn more about AWS EBS, visit: https://aws.amazon.com/ebs/



Step 4: Add Storage [redacted]

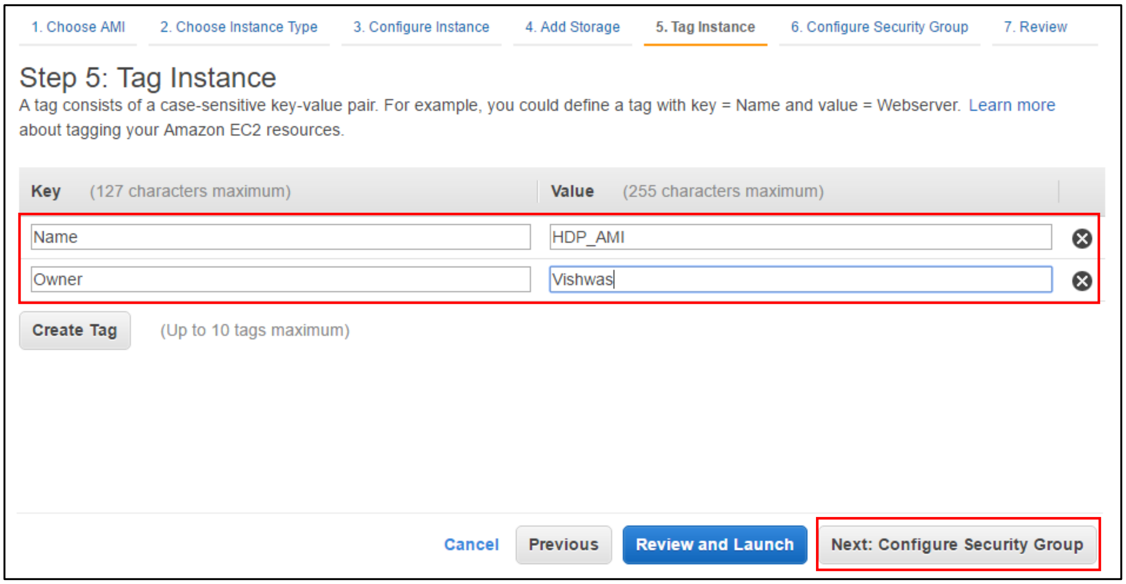
Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes.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

Volume Type ⓘ	Device ⓘ	Snapshot ⓘ	Size (GiB) ⓘ	Volume Type ⓘ	IOPS ⓘ	Delete on Termination ⓘ	Encrypted ⓘ
Root	/dev/sda1	snap-ba40cac8	10	General Purpose (▾	30 / 3000	☑	Not Encrypte
EBS ▾	/dev/sdb ▾	Search (case-insensit	50	General Purpose (▾	150 / 3000	□	□

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.

Figure 2.2.5

Next step is to tag your instance. Click “Next: Configure Security Group”



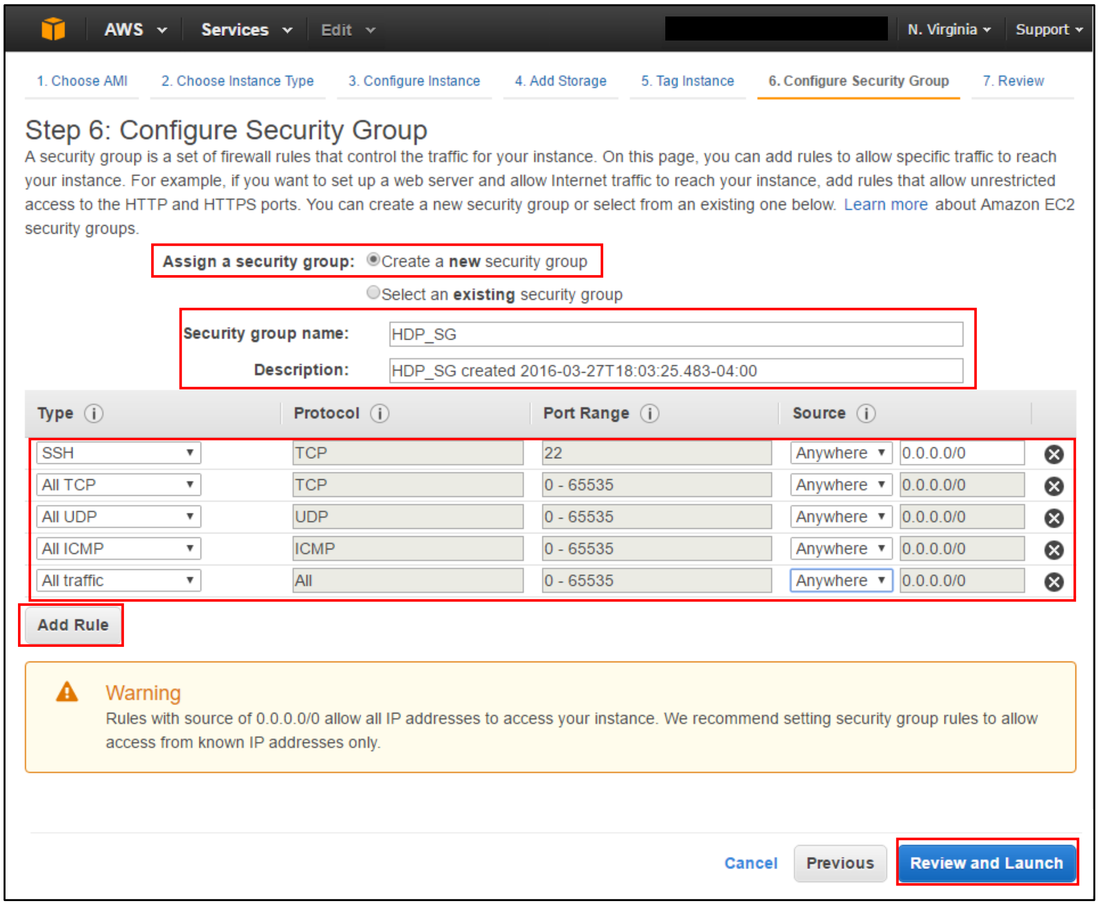
Step 5: Tag Instance

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. Learn more about tagging your Amazon EC2 resources.

Key (127 characters maximum)	Value (255 characters maximum)
Name	HDP_AMI
Owner	Vishwas

(Up to 10 tags maximum)

Figure 2.2.6



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 name: HDP_SG
Description: HDP_SG created 2016-03-27T18:03:25.483-04:00

Type	Protocol	Port Range	Source	
SSH	TCP	22	Anywhere	0.0.0.0/0
All TCP	TCP	0 - 65535	Anywhere	0.0.0.0/0
All UDP	UDP	0 - 65535	Anywhere	0.0.0.0/0
All ICMP	ICMP	0 - 65535	Anywhere	0.0.0.0/0
All traffic	All	0 - 65535	Anywhere	0.0.0.0/0

Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Figure 2.2.7

In this step, you will need to create a new security group or choose an existing security group.

A security group acts as a virtual firewall that controls the traffic for one or more instances. When you launch an instance, you associate one or more security groups with the instance.

To learn more about Security Groups in AWS, visit:

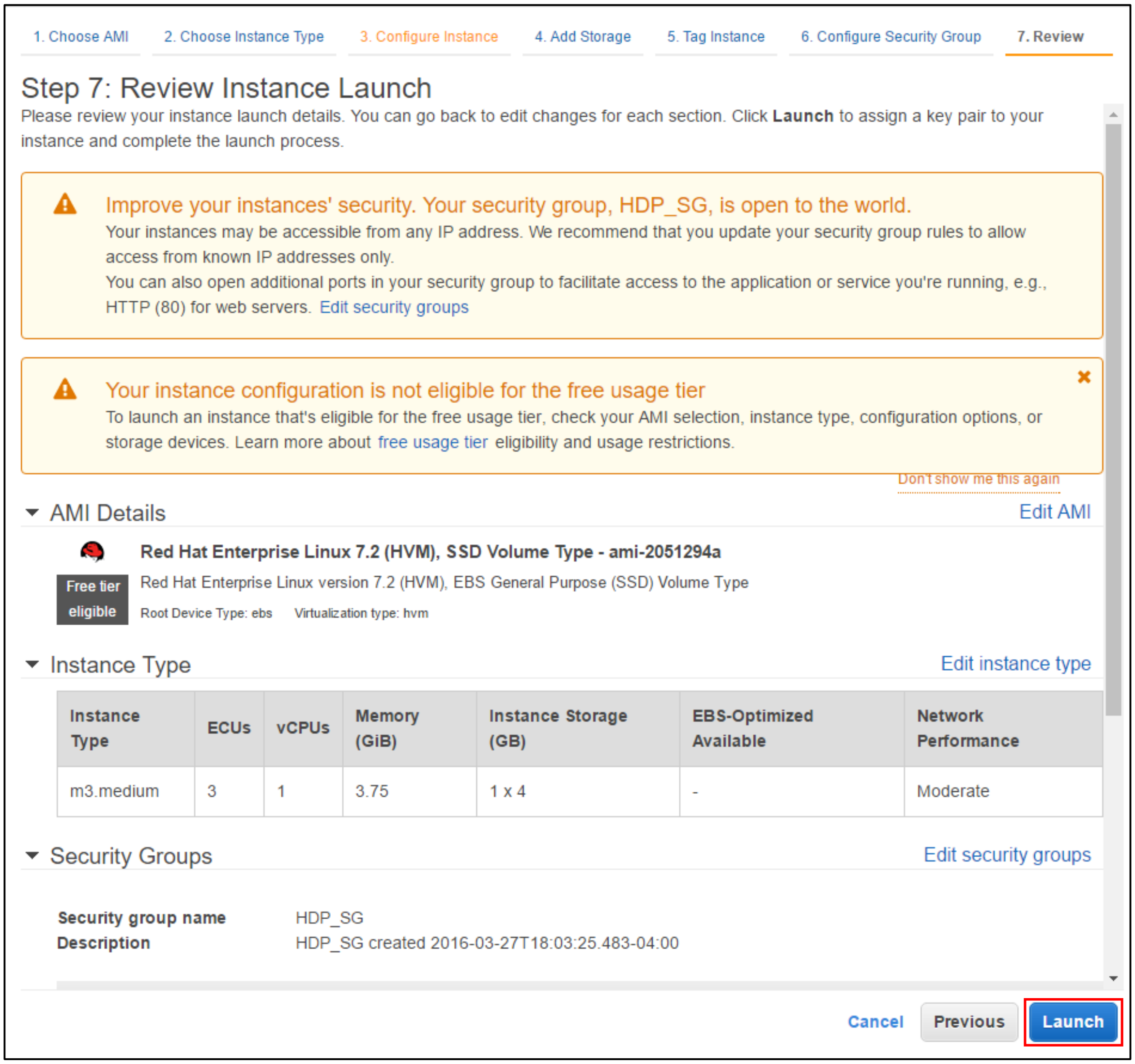
http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-network-security.html

Provide a name for your security group. You need to add rules, which will allow you to access your instance using several protocols.

Source defines the user access to the instance. Setting the source (0.0.0.0/0) makes your instance accessible from any device, when the obtain the Public IP address of the instance. The above link about security group explains, how your instance access can be made more secure.

For Proof of Concept (POC), the above configurations shown in Figure 2.2.7 will be helpful.

Next, click “Review and Launch”.



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.

Improve your instances' security. Your security group, HDP_SG, is open to the world.

Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only.

You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. Edit security groups

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 — Edit AMI

Red Hat Enterprise Linux 7.2 (HVM), SSD Volume Type - ami-2051294a

Free tier eligible

Red Hat Enterprise Linux version 7.2 (HVM), EBS General Purpose (SSD) Volume Type

Root Device Type: ebs   Virtualization type: hvm

Instance Type — Edit instance type

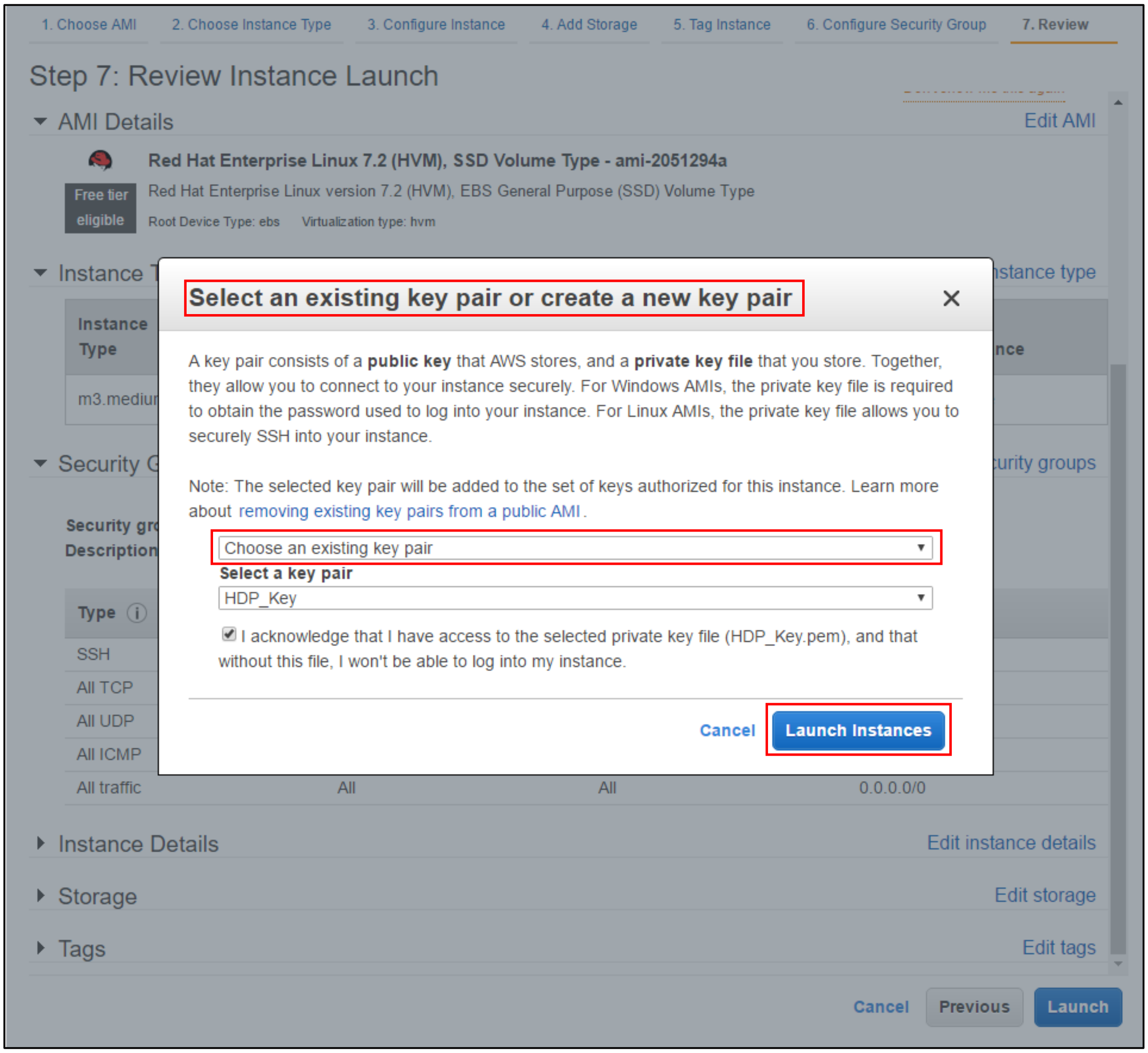
Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
m3.medium	3	1	3.75	1 x 4	-	Moderate

Security Groups — Edit security groups

Security group name HDP_SG

Description HDP_SG created 2016-03-27T18:03:25.483-04:00

Figure 2.2.8



Step 7: Review Instance Launch

AMI Details

Red Hat Enterprise Linux 7.2 (HVM), SSD Volume Type - ami-2051294a

Red Hat Enterprise Linux version 7.2 (HVM), EBS General Purpose (SSD) Volume Type

Root Device Type: ebs Virtualization type: hvm

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

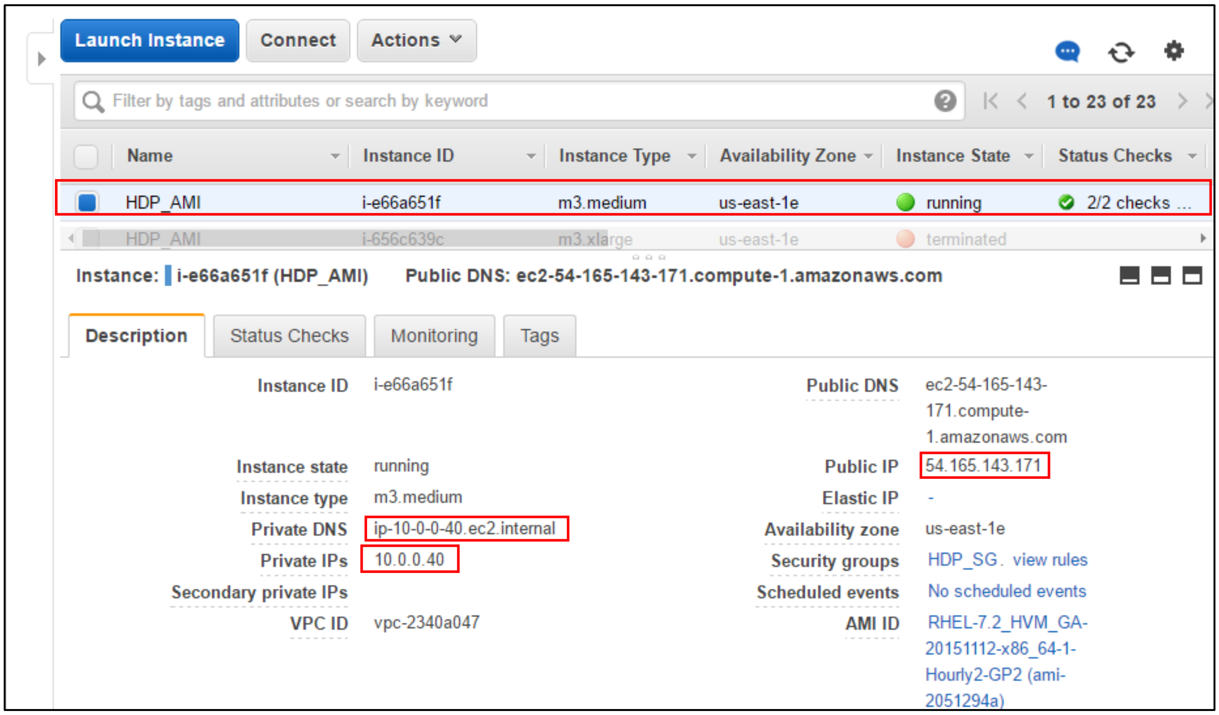
Select a key pair

HDP_Key

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

Figure 2.2.9

As you approach, the last step to launch your instance, you will be asked to select a key-pair. You can either create a new one, or select an existing key-pair as highlighted in Figure 2.2.9.



Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks
HDP_AMI	i-e66a651f	m3.medium	us-east-1e	● running	✓ 2/2 checks ...
HDP_AMI	i-656c639c	m3.xlarge	us-east-1e	● terminated	

Instance: i-e66a651f (HDP_AMI) Public DNS: ec2-54-165-143-171.compute-1.amazonaws.com

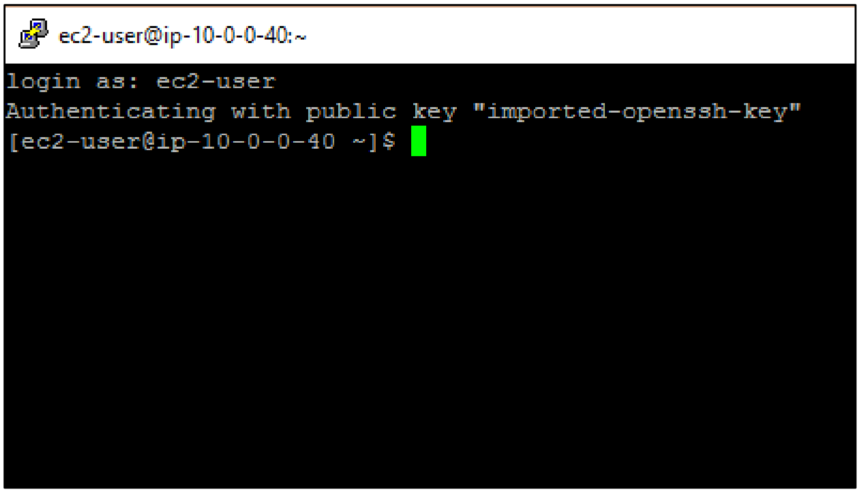
Instance ID	i-e66a651f	Public DNS	ec2-54-165-143-171.compute-1.amazonaws.com
Instance state	running	Public IP	54.165.143.171
Instance type	m3.medium	Elastic IP	-
Private DNS	ip-10-0-0-40.ec2.internal	Availability zone	us-east-1e
Private IPs	10.0.0.40	Security groups	HDP_SG. view rules
Secondary private IPs		Scheduled events	No scheduled events
VPC ID	vpc-2340a047	AMI ID	RHEL-7.2_HVM_GA-20151112-x86_64-1-Hourly2-GP2 (ami-2051294a)

Figure 2.2.10

Figure 2.2.10 highlights the instance has launched successfully and is running.

3 CONFIGURING YOUR INSTANCE FOR HDP INSTALLATION

3.1 INSTALL APACHE



```
login as: ec2-user
Authenticating with public key "imported-openssh-key"
[ec2-user@ip-10-0-0-40 ~]$
```

Figure 3.1.1

Once your instance is ready and running as shown in Figure 2.2.10 (previous section), then you should be able log into the machine by SSH. Login as “ec2-user”.

To learn more about connecting to Linux instances on AWS, visit:

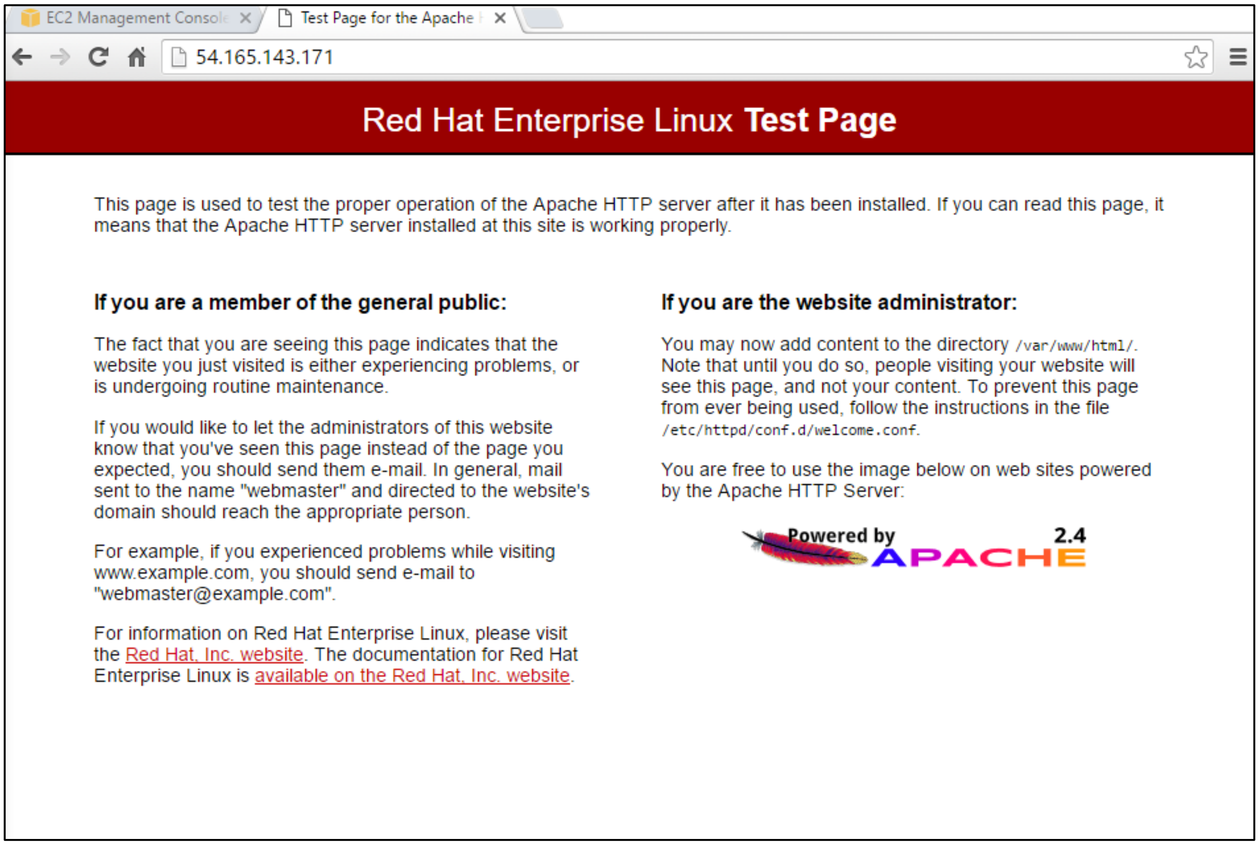
http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstances.html

Just to make sure, the instance public IP address is visible, install Apache.

To install Apache, follow instructions in the link below:

http://www.cyberciti.biz/faq/howto-install-linux-apache-mariadb-php-lamp-stack-on-centos7-rhel7/

If Apache is installed successfully, you should see the following webpage online accessible on your instance. Shown in Figure 3.1.2.



Red Hat Enterprise Linux **Test Page**

This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.

If you are a member of the general public:

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting www.example.com, you should send e-mail to "webmaster@example.com".

For information on Red Hat Enterprise Linux, please visit the Red Hat, Inc. website. The documentation for Red Hat Enterprise Linux is available on the Red Hat, Inc. website.

If you are the website administrator:

You may now add content to the directory `/var/www/html/`. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file `/etc/httpd/conf.d/welcome.conf`.

You are free to use the image below on web sites powered by the Apache HTTP Server:

Figure 3.1.2

3.2 Attach Additional Volume (EBS) To Your Instance

Remember, we attached an EBS of 50 GB for our instance. Now, we need to verify, if this storage is available.

```
$ df -h
```

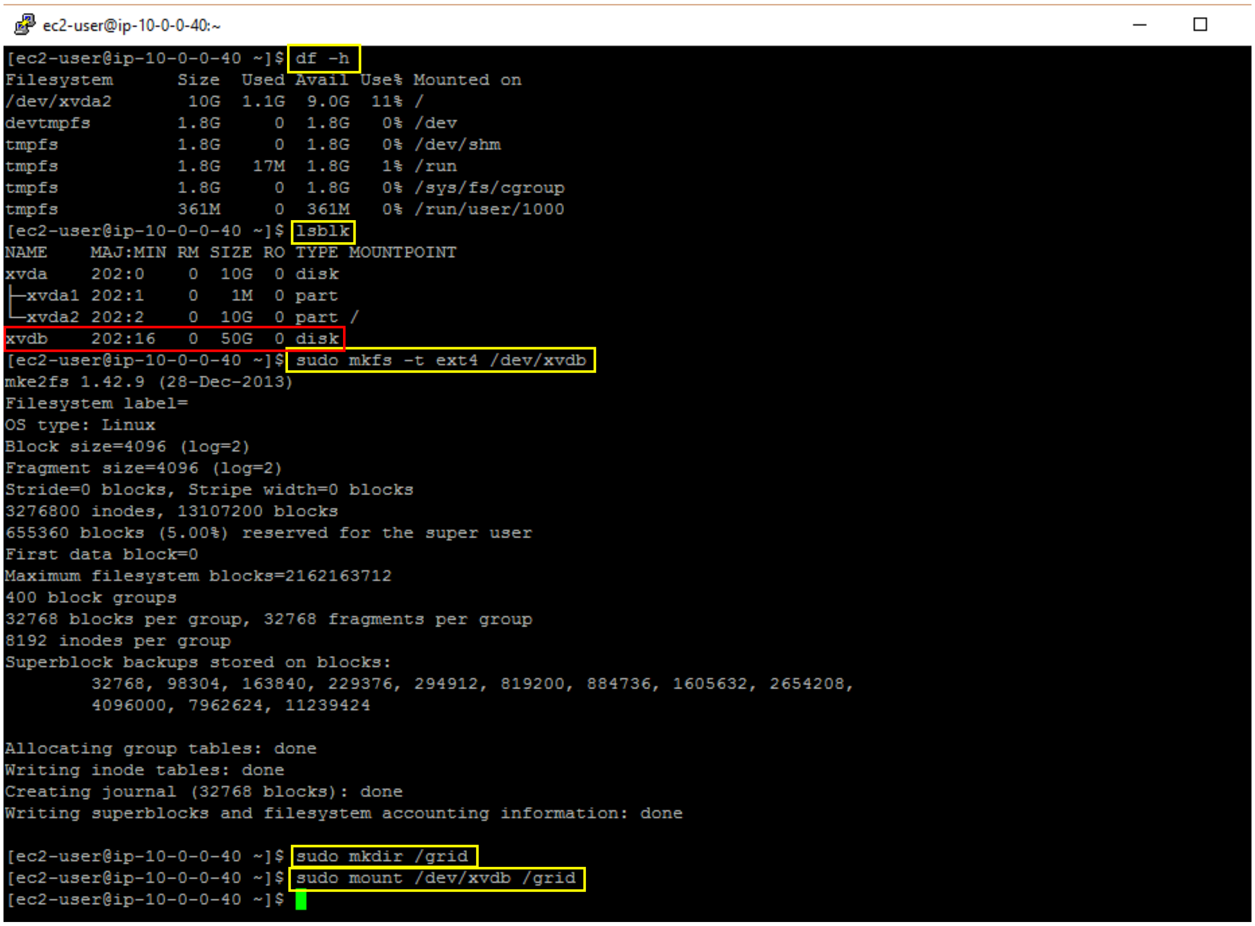
The **df** command provides an option to display sizes in Human Readable formats by using '**-h**' (prints the results in human readable format), highlighted in Figure 3.2.1.

To learn more about disk file system, visit: http://www.tecmint.com/how-to-check-disk-space-in-linux/

```
$ lsblk
```

The **lsblk** command allows you to display a list of available block devices. Highlighted in Figure 3.2.1.

For each listed block device, the lsblk command displays the device name (NAME), major and minor device number (MAJ:MIN), if the device is removable (RM), what is its size (SIZE), if the device is read-only (RO), what type is it (TYPE), and where the device is mounted (MOUNTPOINT).



```
[ec2-user@ip-10-0-0-40 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2       10G  1.1G  9.0G  11% /
devtmpfs        1.8G     0  1.8G   0% /dev
tmpfs           1.8G     0  1.8G   0% /dev/shm
tmpfs           1.8G   17M  1.8G   1% /run
tmpfs           1.8G     0  1.8G   0% /sys/fs/cgroup
tmpfs           361M     0  361M   0% /run/user/1000
[ec2-user@ip-10-0-0-40 ~]$ lsblk
NAME    MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda    202:0    0  10G  0 disk
├─xvda1 202:1    0   1M  0 part
└─xvda2 202:2    0  10G  0 part /
xvdb    202:16   0  50G  0 disk
[ec2-user@ip-10-0-0-40 ~]$ sudo mkfs -t ext4 /dev/xvdb
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
3276800 inodes, 13107200 blocks
655360 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2162163712
400 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624, 11239424

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

[ec2-user@ip-10-0-0-40 ~]$ sudo mkdir /grid
[ec2-user@ip-10-0-0-40 ~]$ sudo mount /dev/xvdb /grid
[ec2-user@ip-10-0-0-40 ~]$
```

Figure 3.2.1

To learn more about **lsblk**, visit:

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s1-sysinfo-filesystems.html

The lsblk command, displays the attached 50 GB EBS as “xvdb”, highlighted in red, in Figure 3.2.1.

Once you add a new disk drive to the system, you may want to partition the drive and use the **ext4** file system.

```
$ sudo mkfs -t ext4 /dev/xvdb
```

Highlighted in yellow, in Figure 3.2.1.

To learn more about make file system, refer:

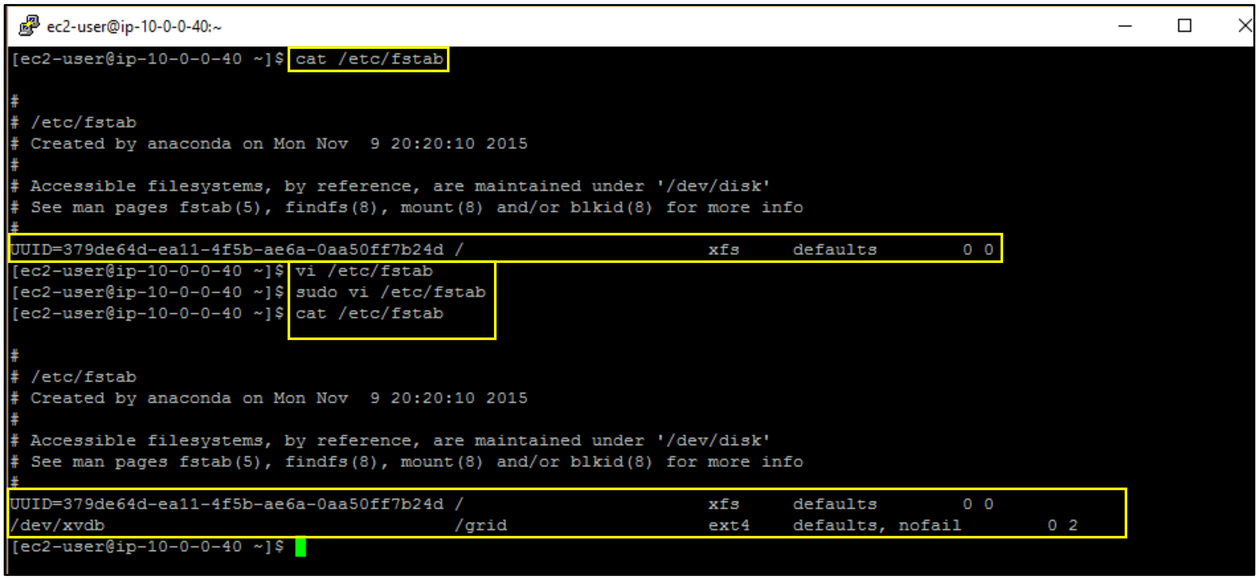
https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/5/html/Deployment_Guide/s1-filesystem-ext4-create.html

Next, we need to create a directory “**grid**” in the **root directory**.

```
$ sudo mkdir /grid
```

```
$ sudo mount /dev/xvdb /grid
```

We can now mount the new storage to “grid” directory on the AWS EC2 instance.



```
[ec2-user@ip-10-0-0-40 ~]$ cat /etc/fstab

#
# /etc/fstab
# Created by anaconda on Mon Nov  9 20:20:10 2015
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=379de64d-ea11-4f5b-ae6a-0aa50ff7b24d /                       xfs     defaults        0 0
[ec2-user@ip-10-0-0-40 ~]$ vi /etc/fstab
[ec2-user@ip-10-0-0-40 ~]$ sudo vi /etc/fstab
[ec2-user@ip-10-0-0-40 ~]$ cat /etc/fstab

#
# /etc/fstab
# Created by anaconda on Mon Nov  9 20:20:10 2015
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=379de64d-ea11-4f5b-ae6a-0aa50ff7b24d /                       xfs     defaults        0 0
/dev/xvdb                                /grid                    ext4    defaults, nofail        0 2
[ec2-user@ip-10-0-0-40 ~]$
```

Figure 3.2.2

The **/etc/fstab** file is used to control what file systems are mounted when the system boots, as well as to supply default values for other file systems that may be mounted manually from time to time.

As highlighted in Figure 3.2.2, I listed the contents present in “**fstab**” file.

Next, you need to edit the “**fstab**” file and add content as highlighted in Figure 3.4.

```
/dev/xvdb                /grid                ext4        defaults, nofail        0 2
```

To learn more about **fstab**, visit:

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/4/html/Introduction_To_System_Administration/s2-storage-mount-fstab.html

Finally **reboot** your instance.

3.3 Disable SELINUX and Enable NTP

Security-Enhanced Linux (**SELinux**) is an implementation of a mandatory access control mechanism in the Linux kernel, checking for allowed operations after standard discretionary access controls are checked.

SELinux can enforce rules on files and processes in a Linux system, and on their actions, based on defined policies.

To learn more about SELinux, visit:

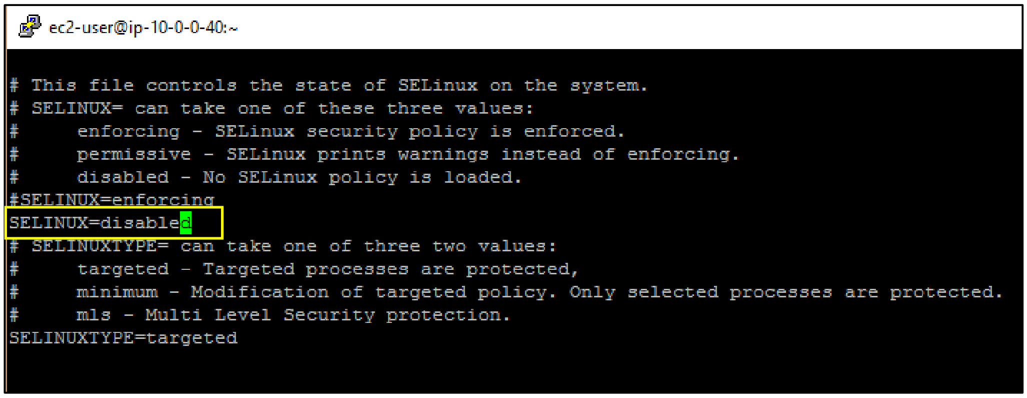
https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/SELinux_Users_and_Administrators_Guide/chap-Security-Enhanced_Linux-Introduction.html

When SELinux is disabled, SELinux policy is not loaded at all; it is not enforced.

To permanently disable SELinux, follow the procedure below:

```
$ sudo vi /etc/sysconfig/selinux
```

Configure **SELINUX=disabled** in the **/etc/selinux/config** file, highlighted in Figure 3.3.1.



```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
#SELINUX=enforcing
SELINUX=disabled
# SELINUXTYPE= can take one of three two values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected processes are protected.
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Figure 3.3.1

Next, you need to disable "**firewall daemon**" in your instance.

```
$ sudo yum install firewalld
```

If the instance, throws a message stating the service is unavailable, then you need to install firewall service.

```
$ sudo yum install firewalld
```

After installation completes, you can disable "firewalld" using "**sudo yum install firewalld**".

Next, stop the "firewalld" service.

```
$ sudo service firewalld stop
```

NTP (**N**etwork **T**ime **P**rotocol) is a protocol to keep servers time synchronized: one or several master servers provide time to client servers that can themselves provide time to other client servers (notion of stratus).

You need NTP service on your instance. You need install and enable NTP on start-up.

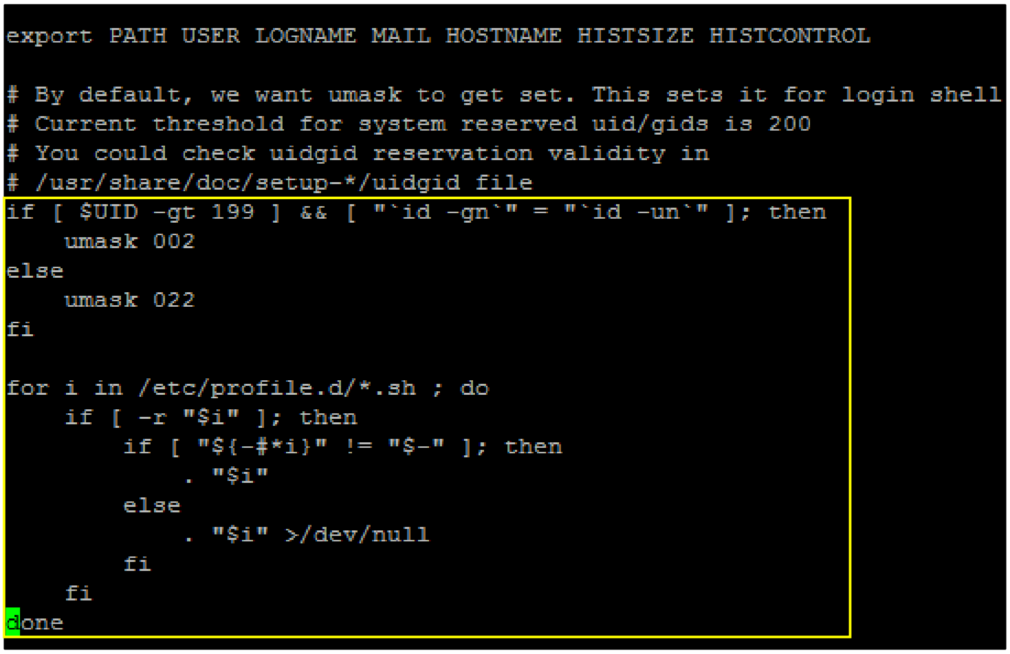
```
$ sudo yum install ntp

$ sudo systemctl enable ntpd

$ sudo systemctl start ntpd

$ sudo systemctl is-enabled ntpd
```

3.4 UMASK

UMASK (**User Mask** or User file creation MASK) is the default permission or base permissions given when a new file (even folder too, as Linux treats everything as files) is created on a Linux machine. Most of the Linux distros give 022 (**0022**) as default UMASK. In other words, it is a system default permission for newly created files/folders in the machine.

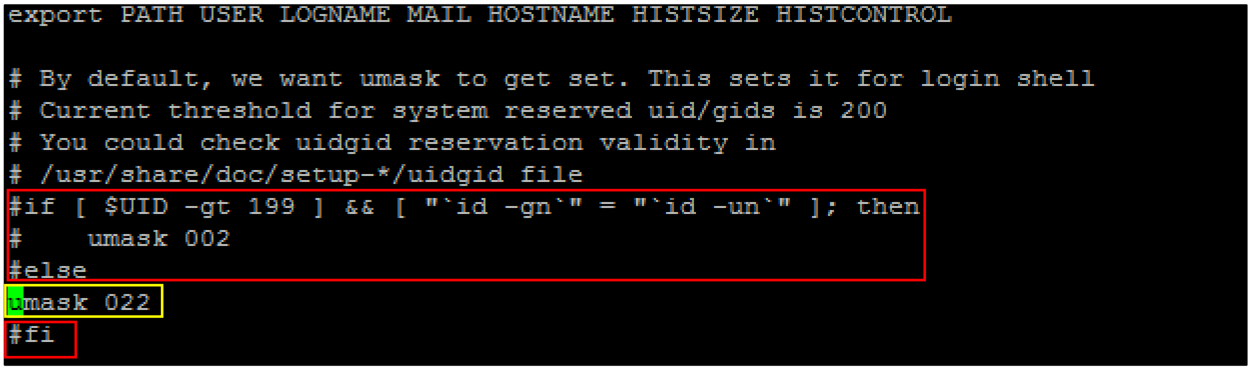
Modify the "/etc/profile" file. Remove the "**umask**" condition as highlighted in Figure 3.4.1 and set it to "**umask 022**" as highlighted in Figure 3.4.2.

```
$ umask

$ sudo vi /etc/profile
```



```
export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE HISTCONTROL

# By default, we want umask to get set. This sets it for login shell
# Current threshold for system reserved uid/gids is 200
# You could check uidgid reservation validity in
# /usr/share/doc/setup-*/uidgid file
if [ $UID -gt 199 ] && [ "`id -gn`" = "`id -un`" ]; then
    umask 002
else
    umask 022
fi

for i in /etc/profile.d/*.sh ; do
    if [ -r "$i" ]; then
        if [ "${-#*i}" != "$-" ]; then
            . "$i"
        else
            . "$i" >/dev/null
        fi
    fi
done
```

Figure 3.4.1



```
export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE HISTCONTROL

# By default, we want umask to get set. This sets it for login shell
# Current threshold for system reserved uid/gids is 200
# You could check uidgid reservation validity in
# /usr/share/doc/setup-*/uidgid file
#if [ $UID -gt 199 ] && [ "`id -gn`" = "`id -un`" ]; then
#    umask 002
#else
umask 022
#fi
```

Figure 3.4.2

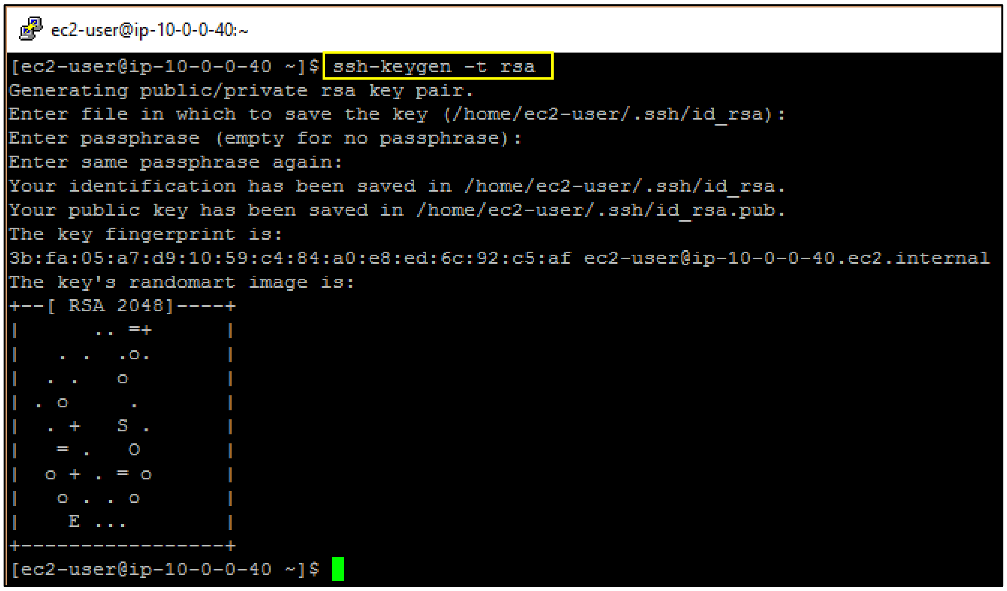
As seen in Figure 3.4.2, the red-highlighted boxes are commented out. The yellow box, shows the "**umask**", set to "**umask 022**".

3.5 Create SSH Keygen

SSH (Secure Shell) is a protocol which facilitates secure communications between two systems using a client-server architecture and allows users to log in to server host systems remotely. Unlike other remote communication protocols, such as FTP or Telnet, SSH encrypts the login session, rendering the connection difficult for intruders to collect unencrypted passwords.

For SSH to be truly effective, using insecure connection protocols should be prohibited. Otherwise, a user's password may be protected using SSH for one session, only to be captured later while logging in using Telnet.

```
$ ssh-keygen -t rsa
```



ec2-user@ip-10-0-0-40:~

```
[ec2-user@ip-10-0-0-40 ~]$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ec2-user/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ec2-user/.ssh/id_rsa.
Your public key has been saved in /home/ec2-user/.ssh/id_rsa.pub.
The key fingerprint is:
3b:fa:05:a7:d9:10:59:c4:84:a0:e8:ed:6c:92:c5:af ec2-user@ip-10-0-0-40.ec2.internal
The key's randomart image is:
+--[ RSA 2048]----+
|      .. =+      |
|   . .  .o.      |
|  . .   o        |
| . o     .       |
|  . +   S .      |
|   = .   O       |
|  o + . = o      |
|   o . . o       |
|    E ...        |
+-----------------+
[ec2-user@ip-10-0-0-40 ~]$
```

Figure 3.5.1

Reboot the instance to apply the changes made to the instance.

3.6 Verify UMASK and NTP

Run the following command mentioned below to verify, if UMASK and NTP are enabled.

```
$ umask

$ sudo systemctl is-enabled ntpd
```

Next, move the public key to “authorized_keys” file.

The public key was created during “Create SSh Keygen” step.

```
$ ls .ssh/

$ cd .ssh

$ $ cat id_rsa.pub >> authorized_keys
```

To verify, a password-less authentication is successful, SSH into the same instance using your instance’s

Private IP or Private DNS

For example:

My private IP is **10.0.0.40**

My private DNS is **ip-10-0-0-40.ec2.internal**

```
$ ssh ip-10-0-0-40.ec2.internal
```

When you enter the above command, you should login, without entering any passwords.

If your login was successful, that means, you have configured password-less SSH correctly.

To read more about password-less SSH, visit:

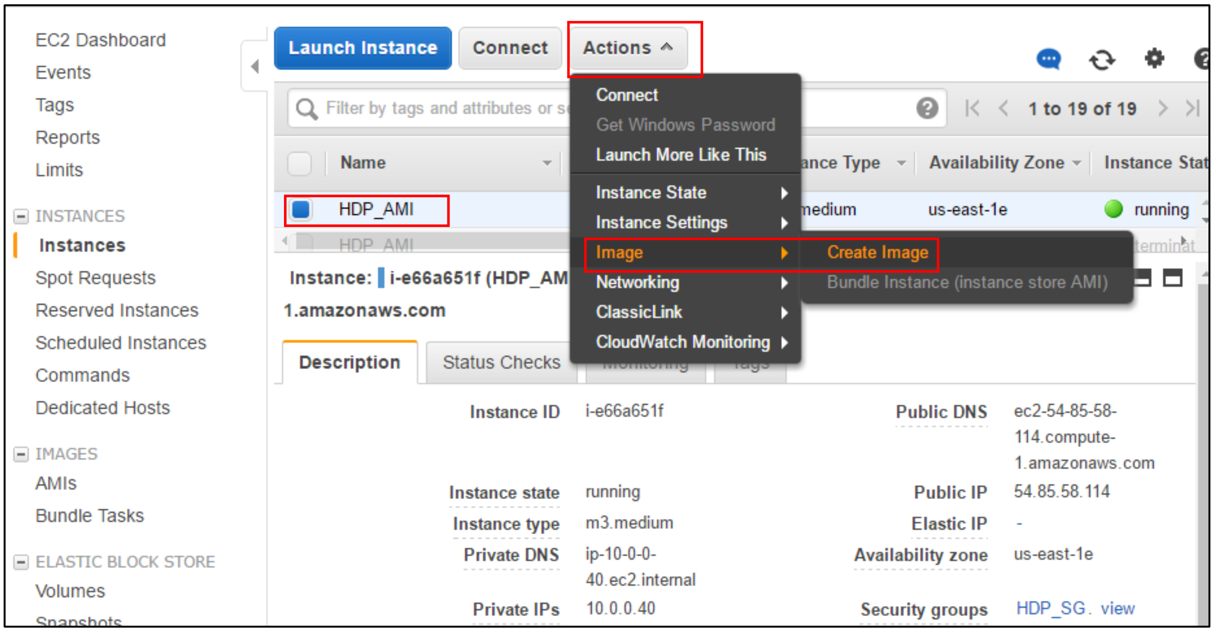
https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System_Administrators_Guide/s1-ssh-configuration.html

4 Create a Base Image on AWS Management Console

After your instance has been configured for installing HDP, we need to make a base image.

This image will be later utilized to create a HDP cluster.

The base image helps you to replicate same configuration on several host machines, avoiding all the steps performed in Section 2.



Name	ance Type	Availability Zone	Instance Stat
HDP_AMI	medium	us-east-1e	running
HDP_AMI			terminat

Instance: i-e66a651f (HDP_AM
1.amazonaws.com

Instance ID	i-e66a651f	Public DNS	ec2-54-85-58-114.compute-1.amazonaws.com
Instance state	running	Public IP	54.85.58.114
Instance type	m3.medium	Elastic IP	-
Private DNS	ip-10-0-0-40.ec2.internal	Availability zone	us-east-1e
Private IPs	10.0.0.40	Security groups	HDP_SG. view

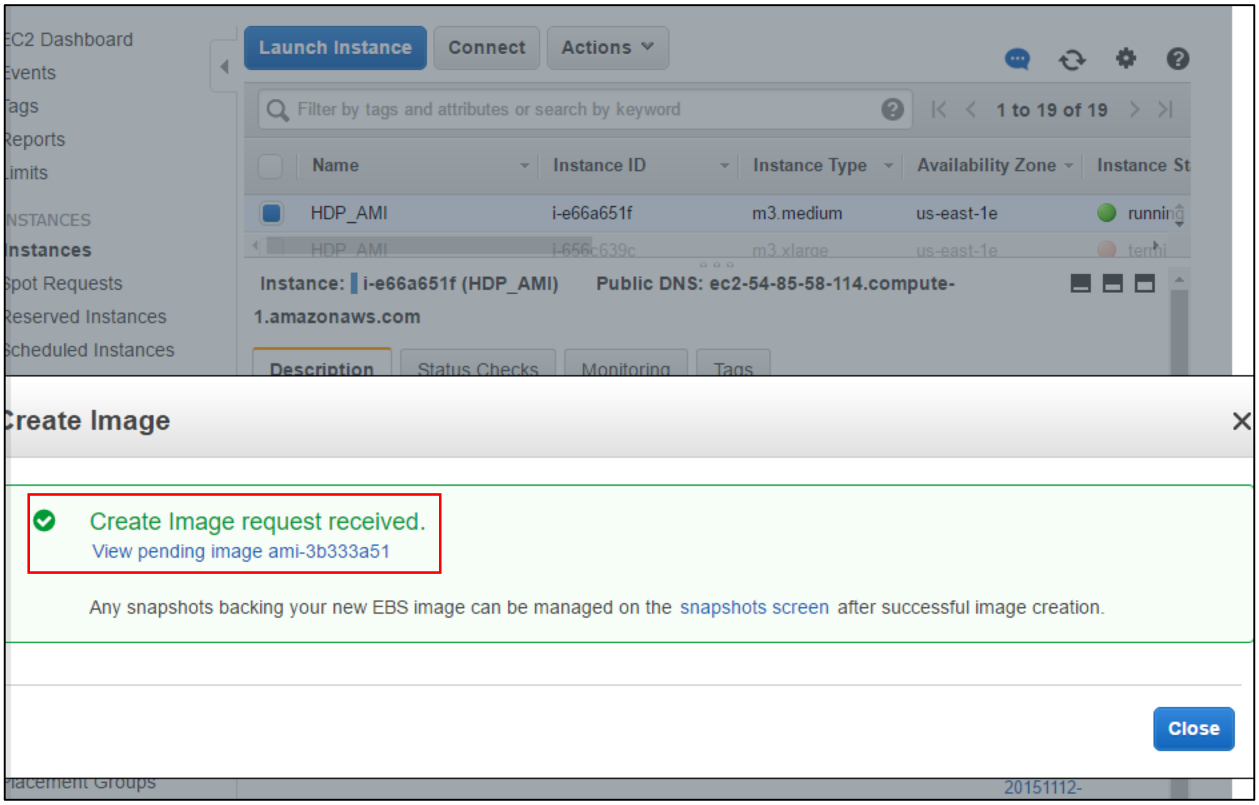
Figure 4.1

Navigate to Instances on EC2 Dashboard and select the configured instance as highlighted in Figure 4.1.

Click “Actions” and navigate to “Image”. Then click on “Create Image”.

You will get an alert dialog box, sayaing “Create image request received” as highlighted in Figure 4.2

Once the base image has been created, navigate to AMIs under EC2 Dashboard as highlighted in Figure 4.3. You will notice, that a new AMI is created and available for use.



Name	Instance ID	Instance Type	Availability Zone	Instance St
HDP_AMI	i-e66a651f	m3.medium	us-east-1e	running
HDP_AMI	i-656c639c	m3.xlarge	us-east-1e	termi

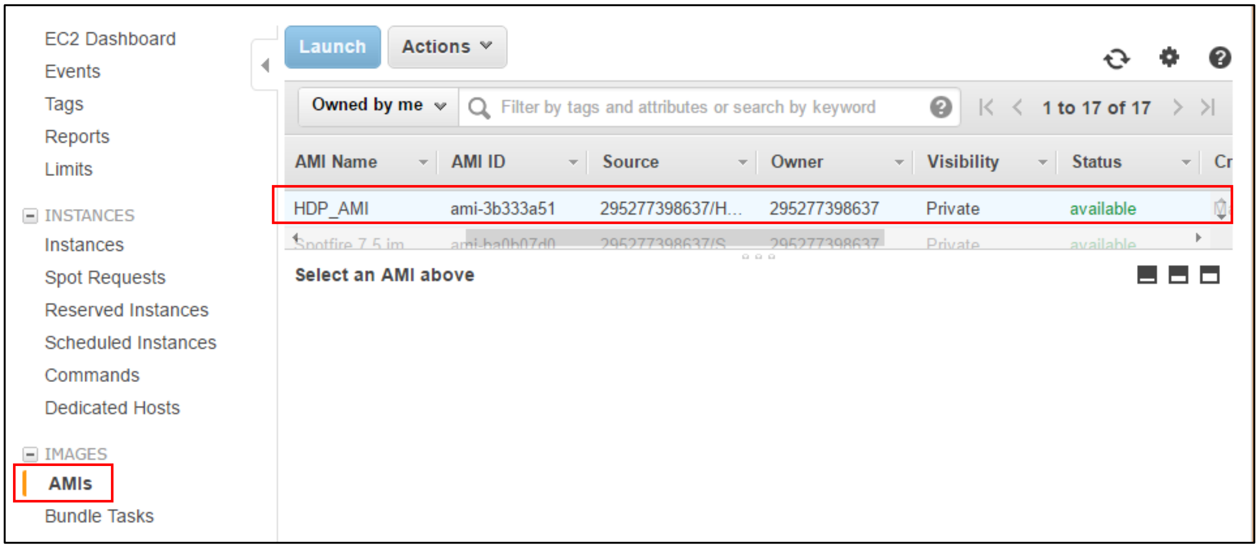
Instance: i-e66a651f (HDP_AMI) Public DNS: ec2-54-85-58-114.compute-1.amazonaws.com

Create Image

Create Image request received.
View pending image ami-3b333a51

Any snapshots backing your new EBS image can be managed on the snapshots screen after successful image creation.

Figure 4.2



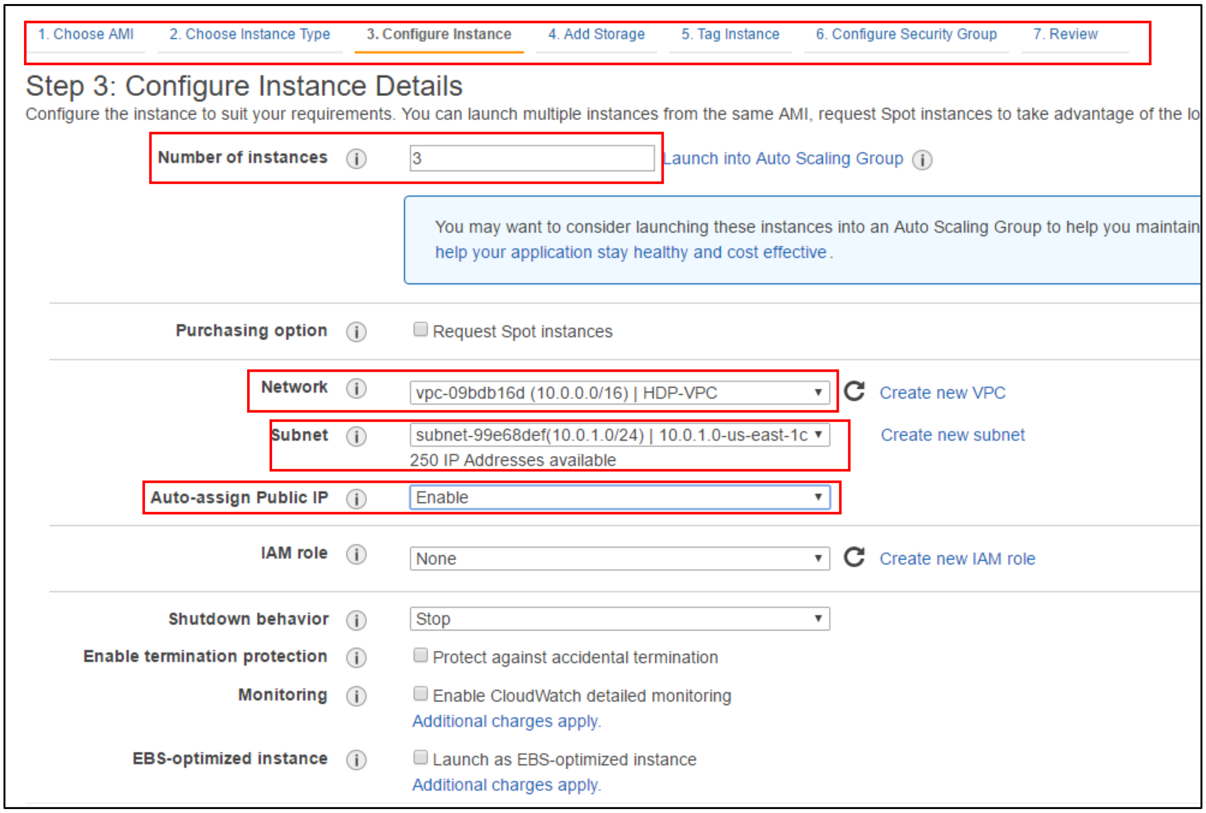
AMI Name	AMI ID	Source	Owner	Visibility	Status	Cr
HDP_AMI	ami-3b333a51	295277398637/H...	295277398637	Private	available	
Spotfire 7.5 im...	ami-ba0b07d0	295277398637/S...	295277398637	Private	available	

Select an AMI above

Figure 4.3

5 CONFIGURE CLUSTER USING BASE IMAGE

We will create a HDP cluster, with the help of Base Image created earlier in Section 4.



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

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lo

Number of instances ⓘ 3 Launch into Auto Scaling Group ⓘ

You may want to consider launching these instances into an Auto Scaling Group to help you maintain
help your application stay healthy and cost effective.

Purchasing option ⓘ □ Request Spot instances

Network ⓘ vpc-09bdb16d (10.0.0.0/16) | HDP-VPC ▾ Create new VPC

Subnet ⓘ subnet-99e68def(10.0.1.0/24) | 10.0.1.0-us-east-1c ▾ Create new subnet
250 IP Addresses available

Auto-assign Public IP ⓘ Enable ▾

IAM role ⓘ None ▾ Create new IAM role

Shutdown behavior ⓘ Stop ▾

Enable termination protection ⓘ □ Protect against accidental termination

Monitoring ⓘ □ Enable CloudWatch detailed monitoring
Additional charges apply.

EBS-optimized instance ⓘ □ Launch as EBS-optimized instance
Additional charges apply.

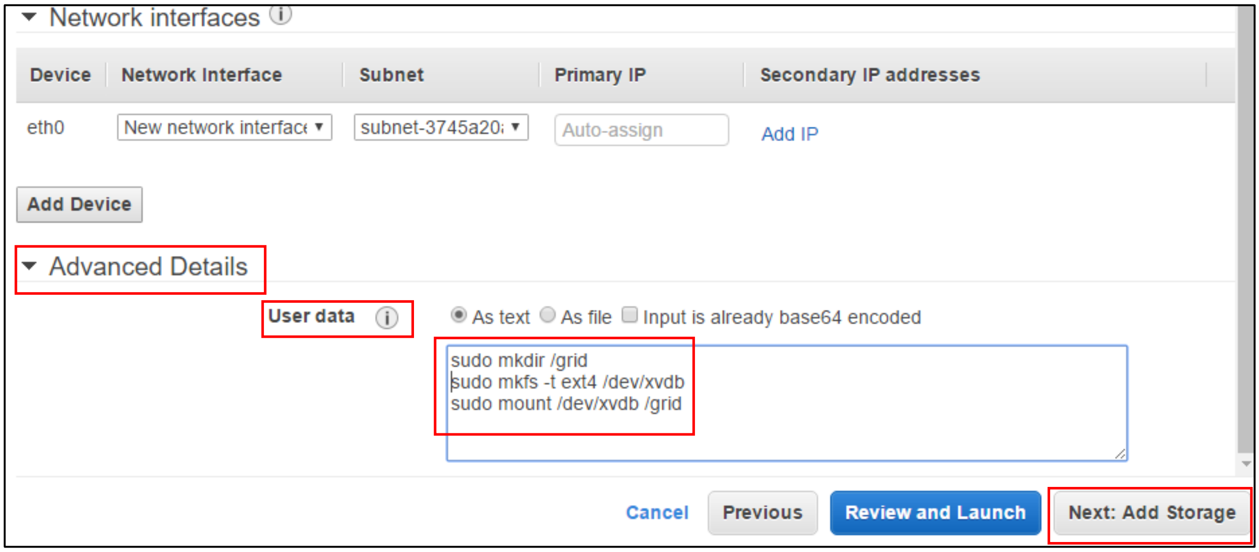
Figure 5.1

- From Figure 4.3 in Section 4, we click "Launch" to launch the HDP_AMI base image.
- Choose "m3.xlarge" instance type with 4 vCPUs and 15 GB of RAM.
- Next, you will need to configure your cluster instances.
- Enter "3" in "Number of Instances" field. Then, select HDP_VPC as the preferred Network to configure networking for your cluster.
- For the subnet field, select the subnet you created in Section 2.1 from the drop down.
- Select "Enable" from the drop down in "Auto-assign Public IP" as highlighted in Figure 5.1.

In the same page, "Configure Instance Details" scroll down to "Advanced Details".

In "User Data" text field, type the following:

```
sudo mkdir /grid

sudo mkfs -t ext4 /dev/xvdb

sudo mount /dev/xvdb /grid
```

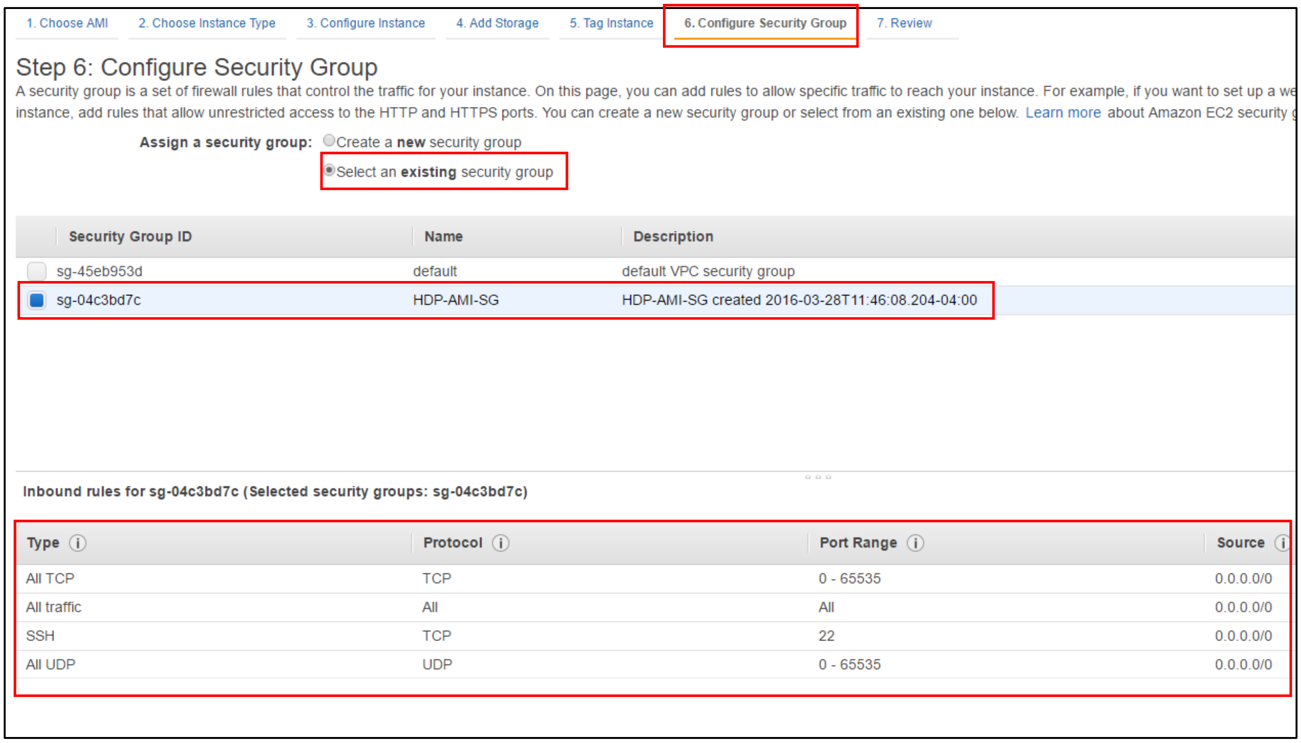


Device	Network Interface	Subnet	Primary IP	Secondary IP addresses
eth0	New network interface	subnet-3745a20	Auto-assign	Add IP

User data: ● As text ○ As file □ Input is already base64 encoded

```
sudo mkdir /grid
sudo mkfs -t ext4 /dev/xvdb
sudo mount /dev/xvdb /grid
```

Figure 5.2



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 we 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 g

Assign a security group: ○ Create a **new** security group
● Select an **existing** security group

	Security Group ID	Name	Description
□	sg-45eb953d	default	default VPC security group
■	sg-04c3bd7c	HDP-AMI-SG	HDP-AMI-SG created 2016-03-28T11:46:08.204-04:00

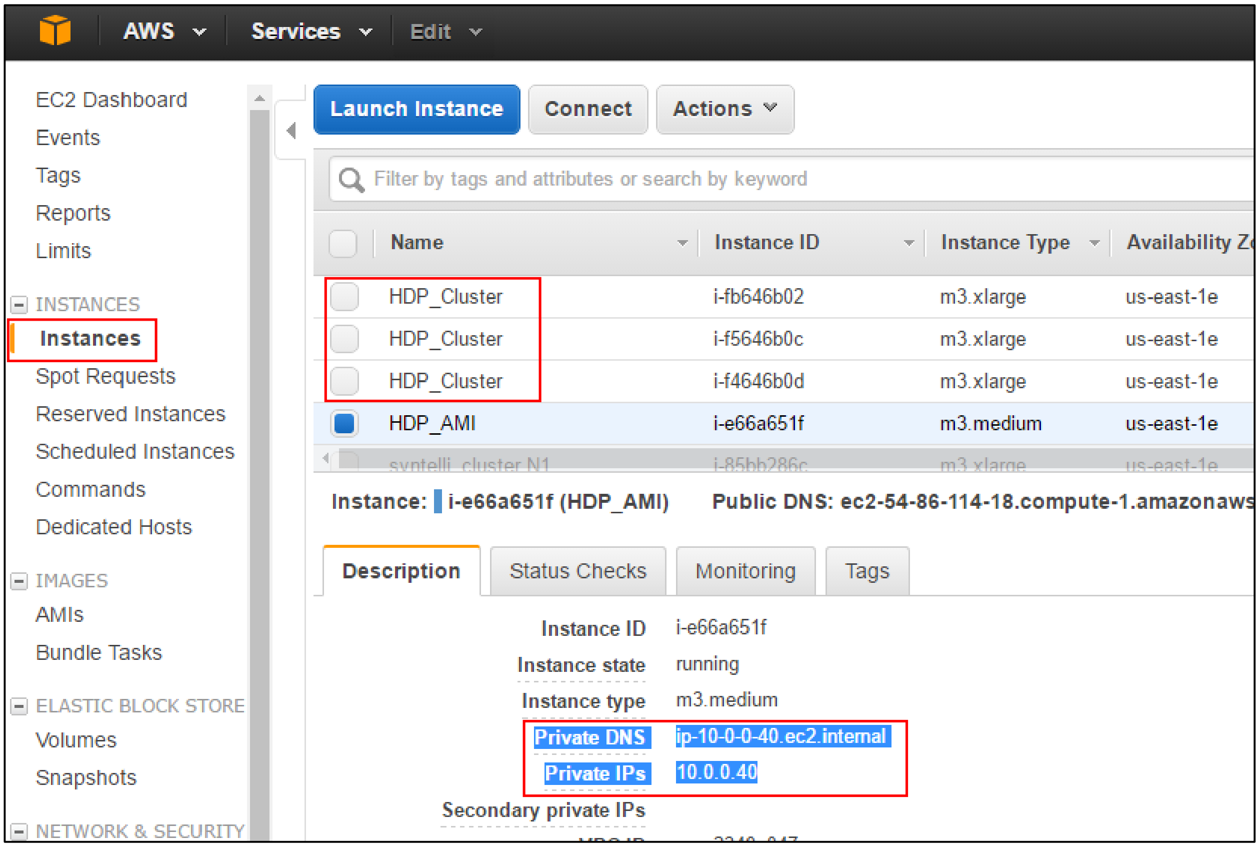
Inbound rules for sg-04c3bd7c (Selected security groups: sg-04c3bd7c)

Type	Protocol	Port Range	Source
All TCP	TCP	0 - 65535	0.0.0.0/0
All traffic	All	All	0.0.0.0/0
SSH	TCP	22	0.0.0.0/0
All UDP	UDP	0 - 65535	0.0.0.0/0

Figure 5.3

Next, you need to configure “Security Group” for the cluster. You had already created a security group, when you prepared the instance in Section 2.2.

You can re-use the same, instead of configuring a new security group, as highlighted in Figure 5.3.



Name	Instance ID	Instance Type	Availability Z
HDP_Cluster	i-fb646b02	m3.xlarge	us-east-1e
HDP_Cluster	i-f5646b0c	m3.xlarge	us-east-1e
HDP_Cluster	i-f4646b0d	m3.xlarge	us-east-1e
HDP_AMI	i-e66a651f	m3.medium	us-east-1e
syntelli_cluster N1	i-85bb286c	m3.xlarge	us-east-1e

Instance: i-e66a651f (HDP_AMI) Public DNS: ec2-54-86-114-18.compute-1.amazonaws

Instance ID	i-e66a651f
Instance state	running
Instance type	m3.medium
Private DNS	ip-10-0-0-40.ec2.internal
Private IPs	10.0.0.40
Secondary private IPs	

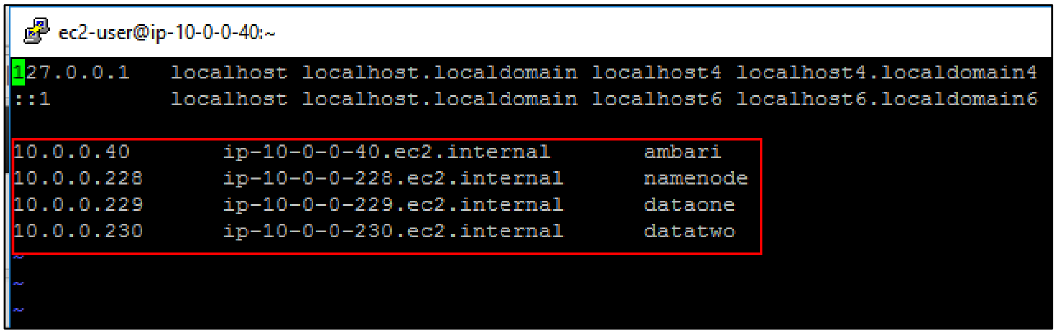
Figure 5.4

NOTE:

- In Figure 5.4, you will notice that HDP_AMI is an instance of type “m3.medium”. I had created two AMI with different instance type “m3.medium” and “m3.xlarge”.
- I realized that “m3.xlarge” instance type was more effective.

Once, the instances are launched, you can navigate to “Instances” under EC2 Dashboard and you will notice 3 new instances in “**running**” state.

Make a note of the **Private DNS** and **Private IPs** for 3 newly created HDP instances, as highlighted in Figure 5.4.


ec2-user@ip-10-0-0-40:~

```
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6

10.0.0.40       ip-10-0-0-40.ec2.internal       ambari
10.0.0.228      ip-10-0-0-228.ec2.internal      namenode
10.0.0.229      ip-10-0-0-229.ec2.internal      dataone
10.0.0.230      ip-10-0-0-230.ec2.internal      datatwo
~
~
~
```

Figure 5.5

SSH as “ec2-user” into HDP_AMI which will serve as Ambari Server.

Ambari makes Hadoop management simpler by providing a consistent, secure platform for operational control. Ambari provides an intuitive Web UI as well as a robust REST API, which is particularly useful for automating cluster operations.

To learn more about Ambari, visit: http://hortonworks.com/hadoop/ambari/

Edit “/etc/hosts” file on all four instances.

```
$ sudo vi /etc/hosts
```

The “**hosts**” file’s contents looks similar to Figure 5.5. You will need to add the Private IP addresses and Private DNS of all four instances:

a. HDP_AMI – ambari
b. HDP_Cluster – namenode
c. HDP_Cluster – dataone
d. HDP_Cluster – datatwo

I named the HDP_Cluster instances namenode, dataone and datatwo based on the ascending numbers of the instance’s Private IP address.

NOTE:

- Your instance IP addresses will differ from the one shown in Figures.
- Make sure, you add the Private IP addresses and Private DNS of all the four instances in “**/etc/hosts**” file for all four instances.

If you try to SSH to one of the data instances from your ambari instance, you should be able login into the instances password-less.

This is possible only, if you have configured the “/etc/hosts” file right.

Also, the SSH Keygen that was created in Section 3.5, helps you log-in to instances on the cluster, not having you to enter password. This is very important for HDP cluster deployment.

To have Ambari Server automatically install Ambari Agents on all your cluster hosts, you must set up password-less SSH connections between the Ambari Server host and all other hosts in the cluster. The Ambari Server host uses SSH public key authentication to remotely access and install the Ambari Agent.

To read more about, password-less SSH on Ambari cluster, visit:

https://docs.hortonworks.com/HDPDocuments/Ambari-2.2.0.0/bk_Installing_HDP_AMB/content/_set_up_password-less_ssh.html

6 AMBARI SERVER INSTALLATION AND SETUP

6.1 INSTALL WGET

Install “wget” tool on your Ambari instance. This instance will serve as your Ambari Server.

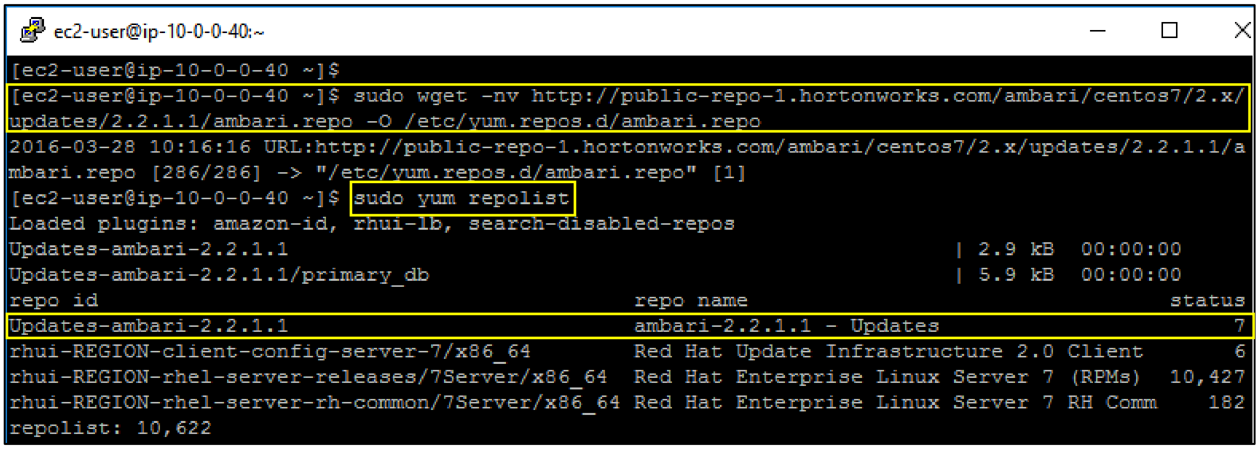
GNU Wget is a free utility for non-interactive download of files from the Web. It supports HTTP, HTTPS, and FTP protocols, as well as retrieval through HTTP proxies.

```
$ sudo yum install wget
```

Next, you need to download the Ambari Repository onto your Ambari Server host.

Make sure, when you copy the above text, it’s a single line execution as highlighted in Figure 6.1.1.

```
$ sudo wget -nv http://public-repo-
1.hortonworks.com/ambari/centos7/2.x/updates/2.2.1.1/ambari.repo -O
/etc/yum.repos.d/ambari.repo
```



```
[ec2-user@ip-10-0-0-40 ~]$
[ec2-user@ip-10-0-0-40 ~]$ sudo wget -nv http://public-repo-1.hortonworks.com/ambari/centos7/2.x/
updates/2.2.1.1/ambari.repo -O /etc/yum.repos.d/ambari.repo
2016-03-28 10:16:16 URL:http://public-repo-1.hortonworks.com/ambari/centos7/2.x/updates/2.2.1.1/a
mbari.repo [286/286] -> "/etc/yum.repos.d/ambari.repo" [1]
[ec2-user@ip-10-0-0-40 ~]$ sudo yum repolist
Loaded plugins: amazon-id, rhui-lb, search-disabled-repos
Updates-ambari-2.2.1.1                                                 | 2.9 kB  00:00:00
Updates-ambari-2.2.1.1/primary_db                                      | 5.9 kB  00:00:00
repo id                                          repo name                                    status
Updates-ambari-2.2.1.1                           ambari-2.2.1.1 - Updates                          7
rhui-REGION-client-config-server-7/x86_64        Red Hat Update Infrastructure 2.0 Client          6
rhui-REGION-rhel-server-releases/7Server/x86_64  Red Hat Enterprise Linux Server 7 (RPMs)     10,427
rhui-REGION-rhel-server-rh-common/7Server/x86_64 Red Hat Enterprise Linux Server 7 RH Comm       182
repolist: 10,622
```

Figure 6.1.1

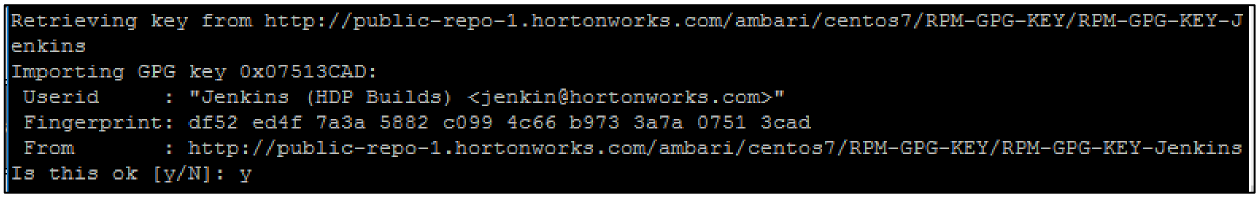
You can verify, if the Ambari Repository has been successfully added to your system repository by issuing the following command.

```
$ sudo yum repolist
```

Now, you need to install the Ambari Server on your “Ambari Server” host. Issue the following command:

```
$ sudo yum install ambari-server
```

When you install the Ambari Server, you will get a token that verifies the installation as shown in Figure 6.1.2.



```
Retrieving key from http://public-repo-1.hortonworks.com/ambari/centos7/RPM-GPG-KEY/RPM-GPG-KEY-J
enkins
Importing GPG key 0x07513CAD:
 Userid     : "Jenkins (HDP Builds) <jenkin@hortonworks.com>"
 Fingerprint: df52 ed4f 7a3a 5882 c099 4c66 b973 3a7a 0751 3cad
 From       : http://public-repo-1.hortonworks.com/ambari/centos7/RPM-GPG-KEY/RPM-GPG-KEY-Jenkins
Is this ok [y/N]: y
```

Figure 6.1.2

6.2 Setup Ambari Server

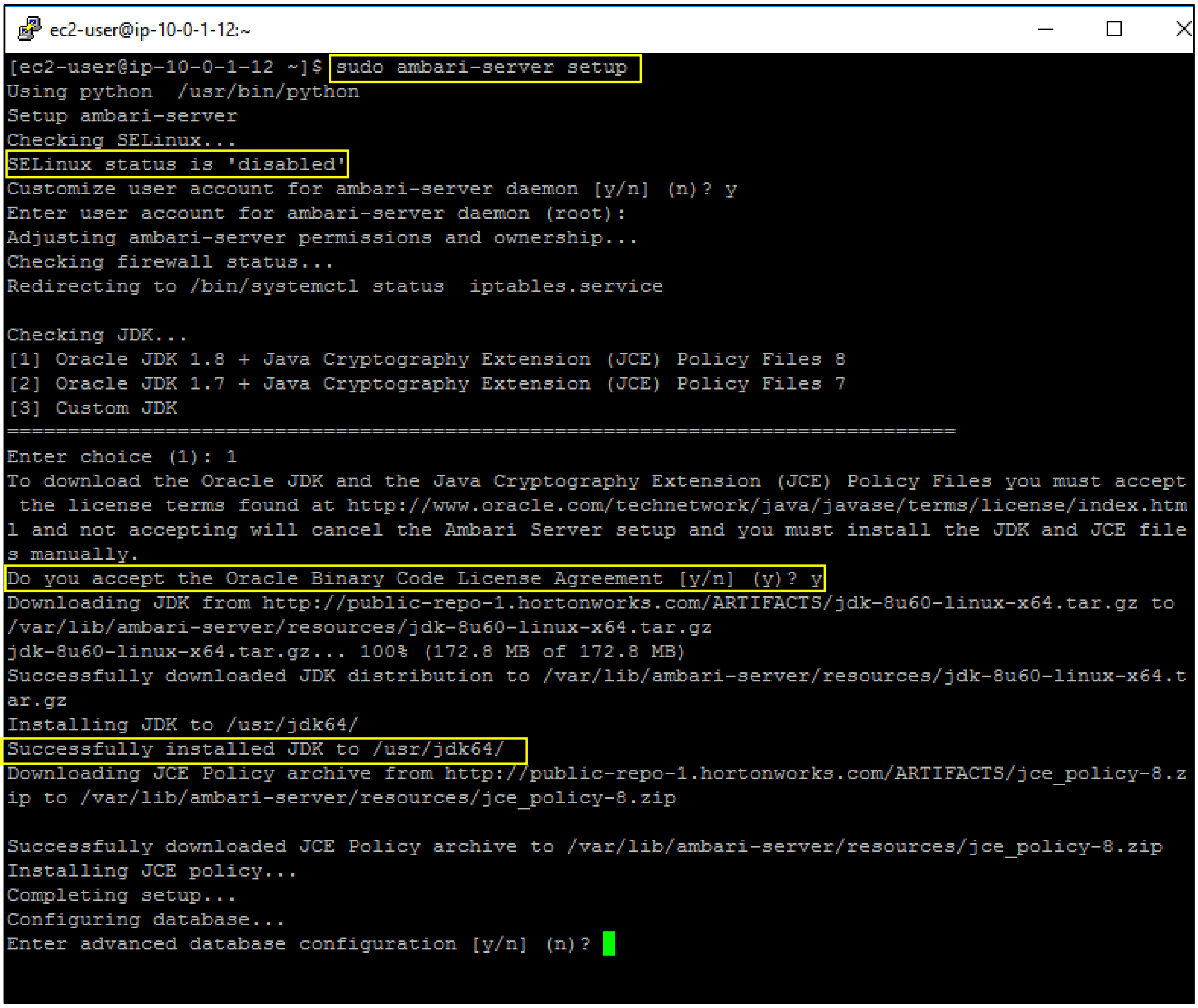
Before starting the Ambari Server, you must set up the Ambari Server. Setup configures Ambari to talk to the Ambari database, installs the JDK and allows you to customize the user account the Ambari Server daemon will run as. The ambari-server setup command manages the setup process. Run the following command on the Ambari server host to start the setup process.

To learn more about ambari server setup, visit:

http://docs.hortonworks.com/HDPDocuments/Ambari-2.2.1.1/bk_Installing_HDP_AMB/content/_set_up_the_ambari_server.html

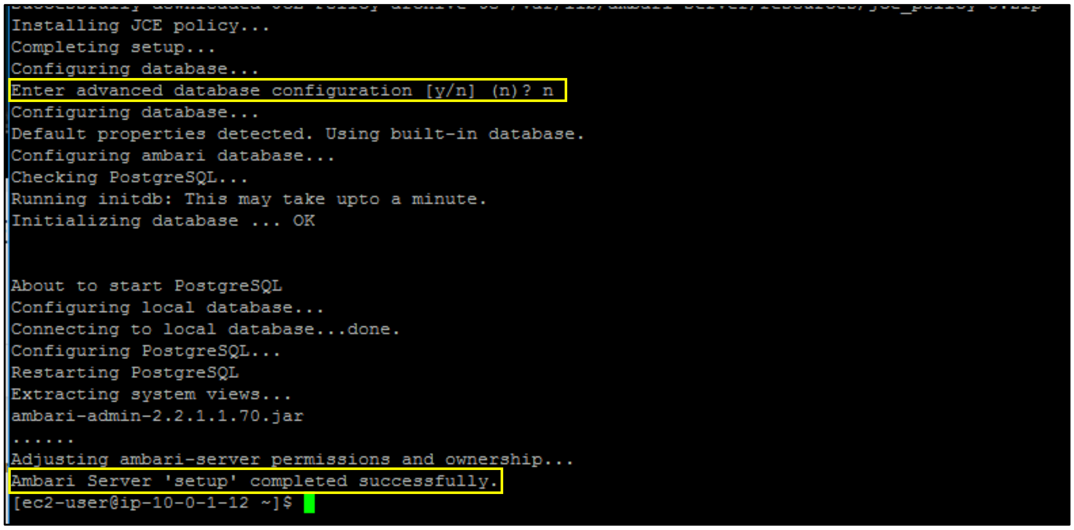
```
$ sudo ambari-server setup
```

Figure 6.2.1 and 6.2.2 highlight the ambari-server setup.



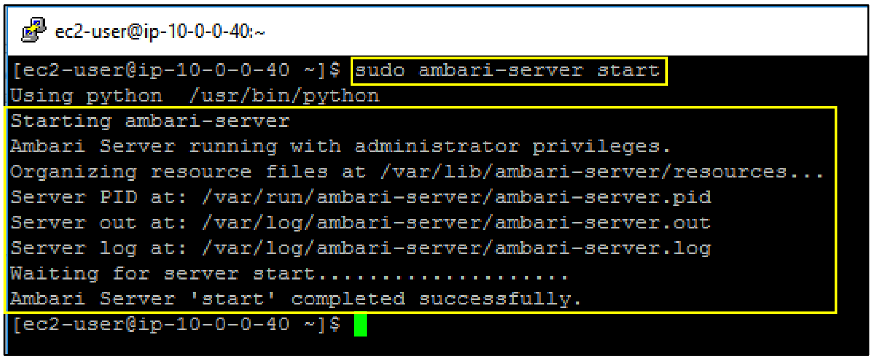
```
[ec2-user@ip-10-0-1-12 ~]$ sudo ambari-server setup
Using python  /usr/bin/python
Setup ambari-server
Checking SELinux...
SELinux status is 'disabled'
Customize user account for ambari-server daemon [y/n] (n)? y
Enter user account for ambari-server daemon (root):
Adjusting ambari-server permissions and ownership...
Checking firewall status...
Redirecting to /bin/systemctl status  iptables.service

Checking JDK...
[1] Oracle JDK 1.8 + Java Cryptography Extension (JCE) Policy Files 8
[2] Oracle JDK 1.7 + Java Cryptography Extension (JCE) Policy Files 7
[3] Custom JDK
==============================================================================
Enter choice (1): 1
To download the Oracle JDK and the Java Cryptography Extension (JCE) Policy Files you must accept
 the license terms found at http://www.oracle.com/technetwork/java/javase/terms/license/index.htm
l and not accepting will cancel the Ambari Server setup and you must install the JDK and JCE file
s manually.
Do you accept the Oracle Binary Code License Agreement [y/n] (y)? y
Downloading JDK from http://public-repo-1.hortonworks.com/ARTIFACTS/jdk-8u60-linux-x64.tar.gz to
/var/lib/ambari-server/resources/jdk-8u60-linux-x64.tar.gz
jdk-8u60-linux-x64.tar.gz... 100% (172.8 MB of 172.8 MB)
Successfully downloaded JDK distribution to /var/lib/ambari-server/resources/jdk-8u60-linux-x64.t
ar.gz
Installing JDK to /usr/jdk64/
Successfully installed JDK to /usr/jdk64/
Downloading JCE Policy archive from http://public-repo-1.hortonworks.com/ARTIFACTS/jce_policy-8.z
ip to /var/lib/ambari-server/resources/jce_policy-8.zip

Successfully downloaded JCE Policy archive to /var/lib/ambari-server/resources/jce_policy-8.zip
Installing JCE policy...
Completing setup...
Configuring database...
Enter advanced database configuration [y/n] (n)?
```

Figure 6.2.1



```
Successfully downloaded JCE Policy archive to /var/lib/ambari-server/resources/jce_policy-8.zip
Installing JCE policy...
Completing setup...
Configuring database...
Enter advanced database configuration [y/n] (n)? n
Configuring database...
Default properties detected. Using built-in database.
Configuring ambari database...
Checking PostgreSQL...
Running initdb: This may take upto a minute.
Initializing database ... OK


About to start PostgreSQL
Configuring local database...
Connecting to local database...done.
Configuring PostgreSQL...
Restarting PostgreSQL
Extracting system views...
ambari-admin-2.2.1.1.70.jar
......
Adjusting ambari-server permissions and ownership...
Ambari Server 'setup' completed successfully.
[ec2-user@ip-10-0-1-12 ~]$
```

Figure 6.2.2



ec2-user@ip-10-0-0-40:~

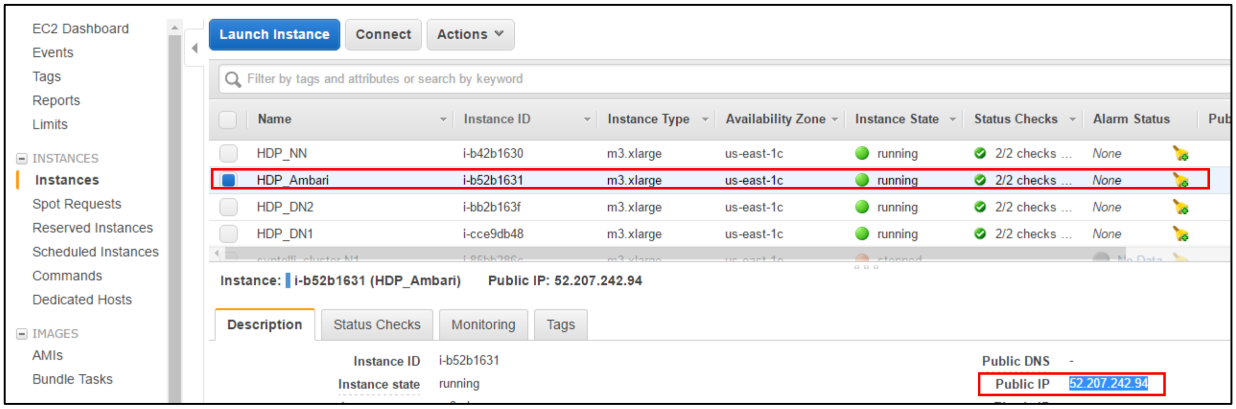
```
[ec2-user@ip-10-0-0-40 ~]$ sudo ambari-server start
Using python  /usr/bin/python
Starting ambari-server
Ambari Server running with administrator privileges.
Organizing resource files at /var/lib/ambari-server/resources...
Server PID at: /var/run/ambari-server/ambari-server.pid
Server out at: /var/log/ambari-server/ambari-server.out
Server log at: /var/log/ambari-server/ambari-server.log
Waiting for server start..................
Ambari Server 'start' completed successfully.
[ec2-user@ip-10-0-0-40 ~]$
```

Figure 6.2.3

Once you successfully completed, ambari-server setup, you can start the server and proceed to installation process on using the Ambari Web based installation.

```
$ sudo ambari-server start
```

The ambari-server will start on the host, and display few log messages as highlighted in Figure 6.2.3.



Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Pub
HDP_NN	i-b42b1630	m3.xlarge	us-east-1c	● running	✓ 2/2 checks ...	None	
HDP_Ambari	i-b52b1631	m3.xlarge	us-east-1c	● running	✓ 2/2 checks ...	None	
HDP_DN2	i-bb2b163f	m3.xlarge	us-east-1c	● running	✓ 2/2 checks ...	None	
HDP_DN1	i-cce9db48	m3.xlarge	us-east-1c	● running	✓ 2/2 checks ...	None	

Instance: i-b52b1631 (HDP_Ambari)   Public IP: 52.207.242.94

Instance ID i-b52b1631
Instance state running
Public DNS -
Public IP 52.207.242.94

Figure 6.2.4

7 Installing, Configuring, and Deploying a HDP Cluster

Use the Ambari Install Wizard running in your browser to install, configure, and deploy your cluster, as follows:

7.1 Log In to Apache Ambari

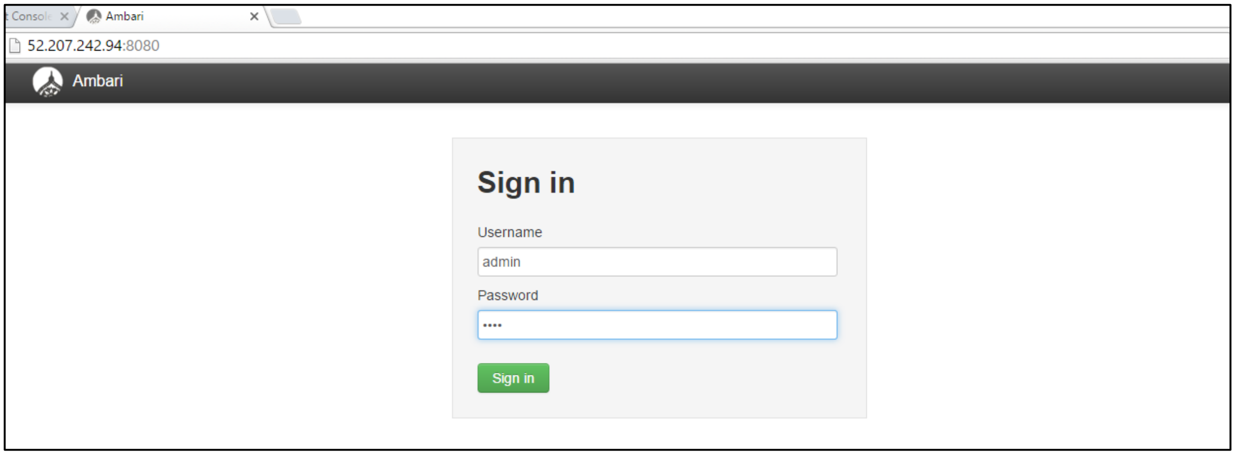
After starting the Ambari service, open Ambari Web using a web browser.

Point your browser to http://<your.ambari.server.external.ip>:8080,where

<your.ambari.server.external.ip> is the name of your ambari server host.

Log in to the Ambari Server using the default user name/password: admin/admin. You can change these credentials later. Highlighted in Figure 7.1.1.

For a new cluster, the Ambari install wizard displays a Welcome page from which you launch the Ambari Install wizard.



52.207.242.94:8080

Sign in

Username: admin

Password: ••••

Figure 7.1.1

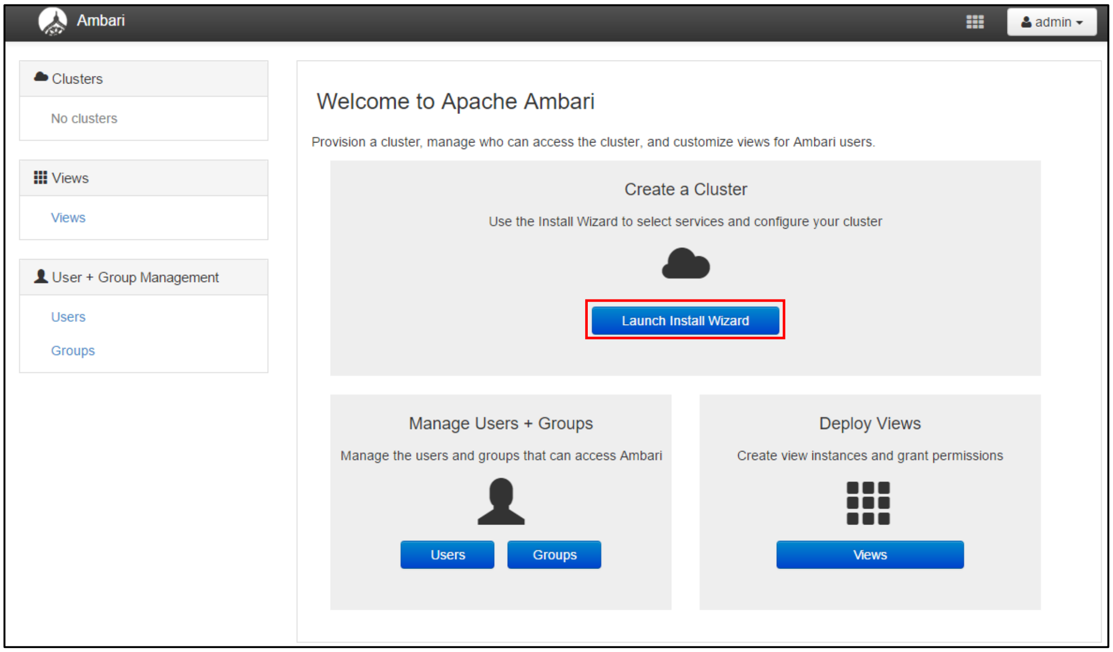
7.2 Name Your Cluster

Once, you sign into the Ambari Install Wizard, then you will be presented to a screen shown in Figure 7.2.1.

Click “Launch Install Wizard” to create a new cluster.

You will be navigated to “Get Started” screen, where you need to provide a name for your cluster.

Provide a name for your cluster, and click “Next” as highlighted in Figure 7.2.2.



Welcome to Apache Ambari

Provision a cluster, manage who can access the cluster, and customize views for Ambari users.

Create a Cluster

Use the Install Wizard to select services and configure your cluster

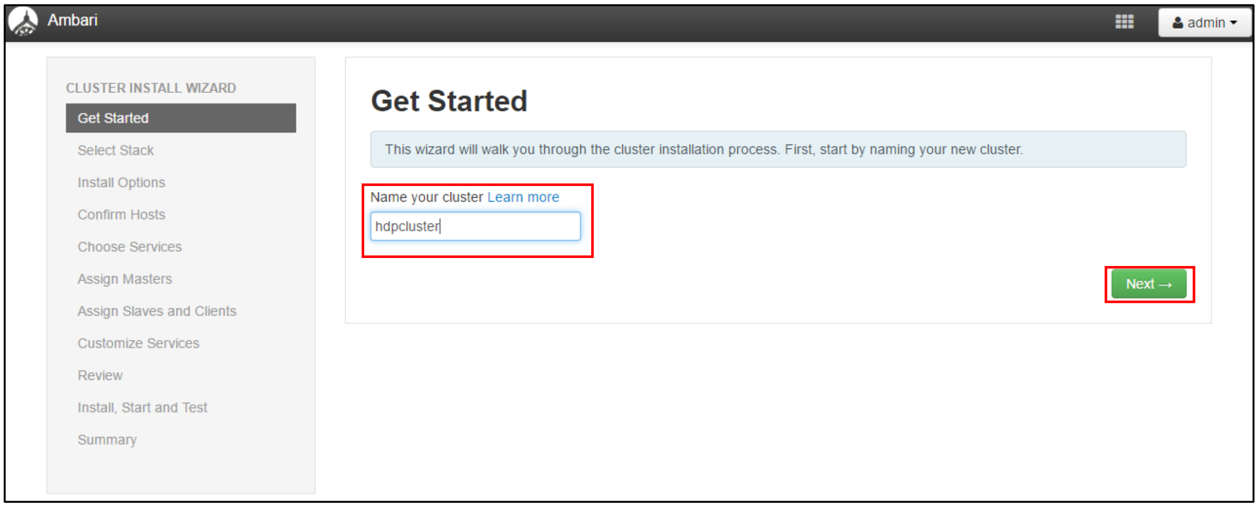
Manage Users + Groups

Manage the users and groups that can access Ambari

Deploy Views

Create view instances and grant permissions

Figure 7.2.1



Get Started

This wizard will walk you through the cluster installation process. First, start by naming your new cluster.

Name your cluster Learn more

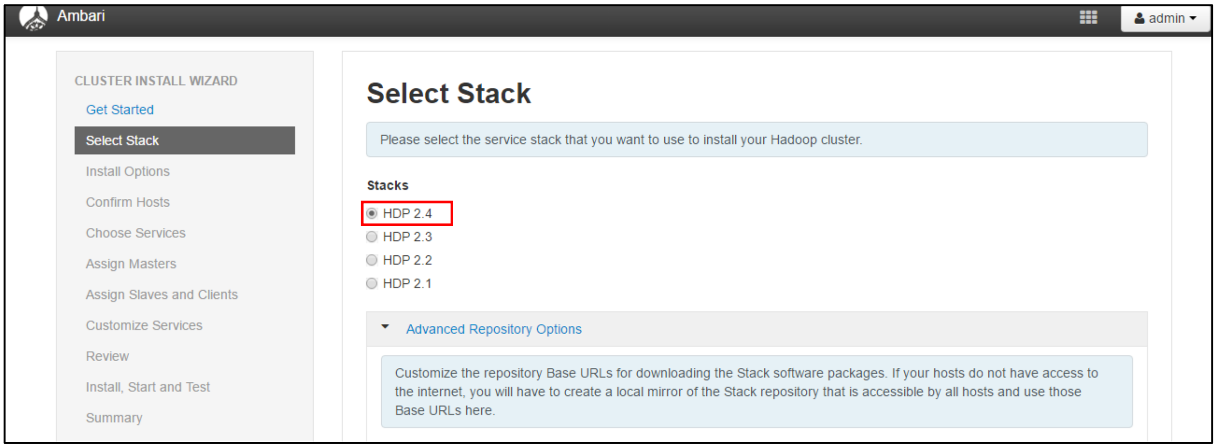
hdpcluster

Figure 7.2.2

7.3 Select Stack

Select HDP stack version, which you would like to install on your cluster.

I selected the latest version to be installed on my AWS cluster, as highlighted in Figure 7.3.1



Select Stack

Please select the service stack that you want to use to install your Hadoop cluster.

Stacks

● HDP 2.4
○ HDP 2.3
○ HDP 2.2
○ HDP 2.1

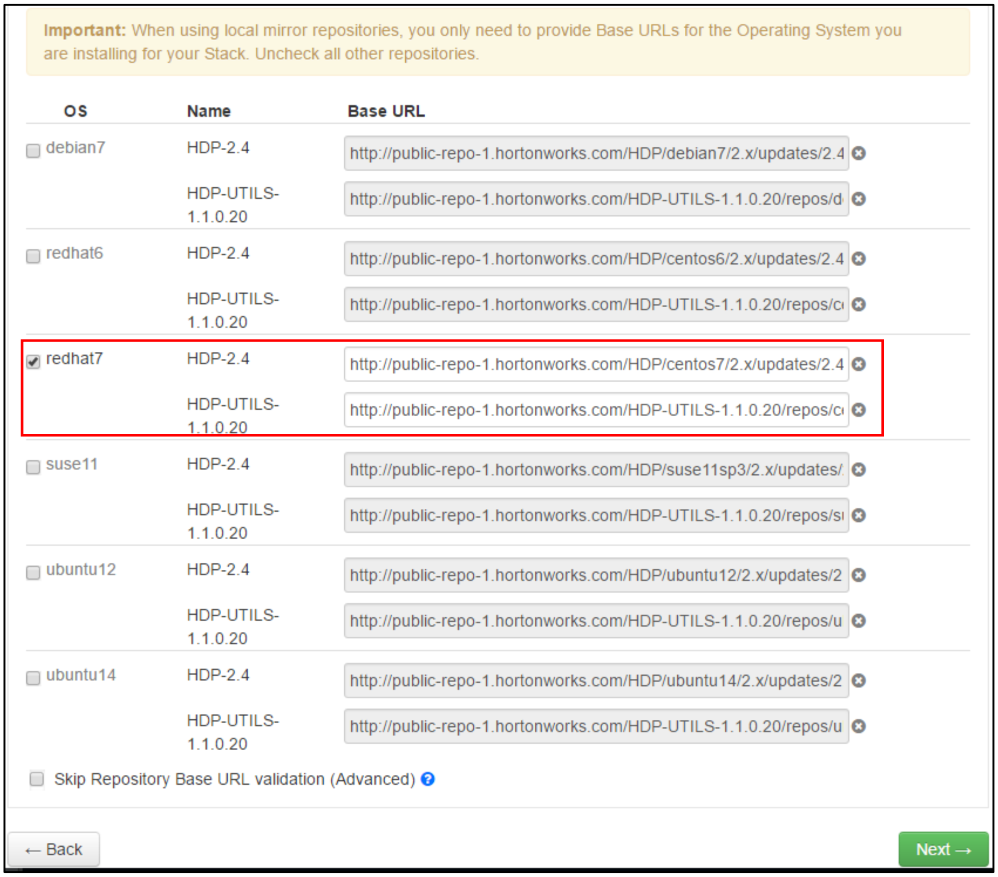
Advanced Repository Options

Customize the repository Base URLs for downloading the Stack software packages. If your hosts do not have access to the internet, you will have to create a local mirror of the Stack repository that is accessible by all hosts and use those Base URLs here.

Figure 7.3.1

7.4 Install Options

Select the HDP version for your OS type. I selected redhat7 repositories as highlighted in Figure 7.4.1.

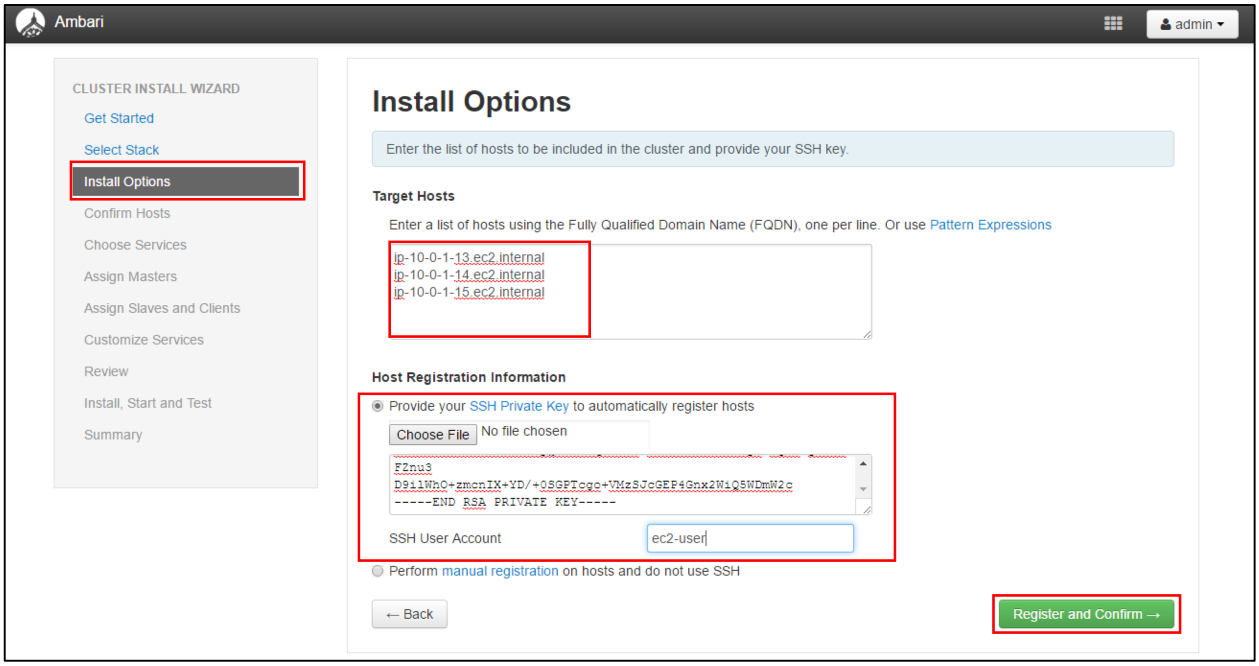


Important: When using local mirror repositories, you only need to provide Base URLs for the Operating System you are installing for your Stack. Uncheck all other repositories.

OS	Name	Base URL
□ debian7	HDP-2.4	http://public-repo-1.hortonworks.com/HDP/debian7/2.x/updates/2.4
	HDP-UTILS-1.1.0.20	http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.20/repos/d
□ redhat6	HDP-2.4	http://public-repo-1.hortonworks.com/HDP/centos6/2.x/updates/2.4
	HDP-UTILS-1.1.0.20	http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.20/repos/c
☑ redhat7	HDP-2.4	http://public-repo-1.hortonworks.com/HDP/centos7/2.x/updates/2.4
	HDP-UTILS-1.1.0.20	http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.20/repos/c
□ suse11	HDP-2.4	http://public-repo-1.hortonworks.com/HDP/suse11sp3/2.x/updates/
	HDP-UTILS-1.1.0.20	http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.20/repos/s
□ ubuntu12	HDP-2.4	http://public-repo-1.hortonworks.com/HDP/ubuntu12/2.x/updates/2
	HDP-UTILS-1.1.0.20	http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.20/repos/u
□ ubuntu14	HDP-2.4	http://public-repo-1.hortonworks.com/HDP/ubuntu14/2.x/updates/2
	HDP-UTILS-1.1.0.20	http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.20/repos/u

□ Skip Repository Base URL validation (Advanced)

Figure 7.4.1



Install Options

Enter the list of hosts to be included in the cluster and provide your SSH key.

Target Hosts

Enter a list of hosts using the Fully Qualified Domain Name (FQDN), one per line. Or use Pattern Expressions

```
ip-10-0-1-13.ec2.internal
ip-10-0-1-14.ec2.internal
ip-10-0-1-15.ec2.internal
```

Host Registration Information

● Provide your SSH Private Key to automatically register hosts

Choose File No file chosen

```
FZnu3
D9ilWhO+zmcnIX+YD/+0SGPTcgo+VMzSJcGEP4Gnx2WiQ5WDmW2c
-----END RSA PRIVATE KEY-----
```

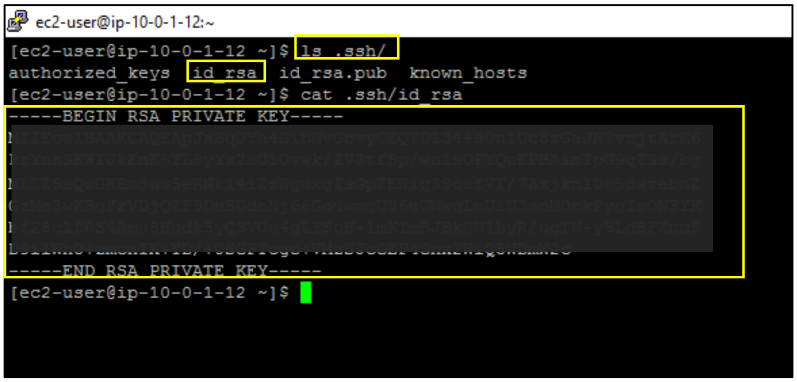
SSH User Account: ec2-user

○ Perform manual registration on hosts and do not use SSH

Figure 7.4.2

Next, enter the Private DNS of instances of namenode, dataone and datatwo. One instance will be configured as Name Node and the other two will be configured as Data Nodes.

Copy the private key from “**id_rsa**” file on you Ambari Server host. This file is located in “.ssh” directory.



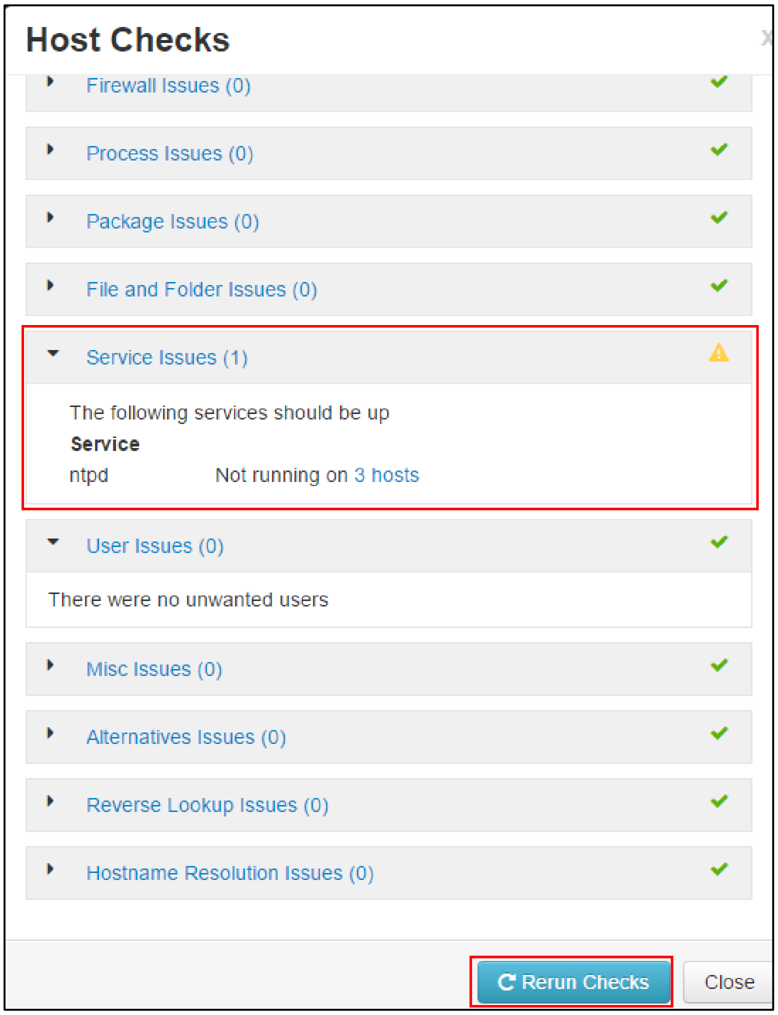
```
ec2-user@ip-10-0-1-12:~
[ec2-user@ip-10-0-1-12 ~]$ ls .ssh/
authorized_keys  id_rsa  id_rsa.pub  known_hosts
[ec2-user@ip-10-0-1-12 ~]$ cat .ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
[redacted]
-----END RSA PRIVATE KEY-----
[ec2-user@ip-10-0-1-12 ~]$
```

Figure 7.4.3

Copy it from “---BEGIN RSA PRIVATE KEY--” to “----END RSA PRIVATE KEY---” as highlighted in Figure 7.4.3.

Finally, click “Register and Confirm”.

7.5 Confirm Hosts



Host Checks

Firewall Issues (0) ✓

Process Issues (0) ✓

Package Issues (0) ✓

File and Folder Issues (0) ✓

Service Issues (1) ⚠

The following services should be up

Service

ntpd Not running on 3 hosts

User Issues (0) ✓

There were no unwanted users

Misc Issues (0) ✓

Alternatives Issues (0) ✓

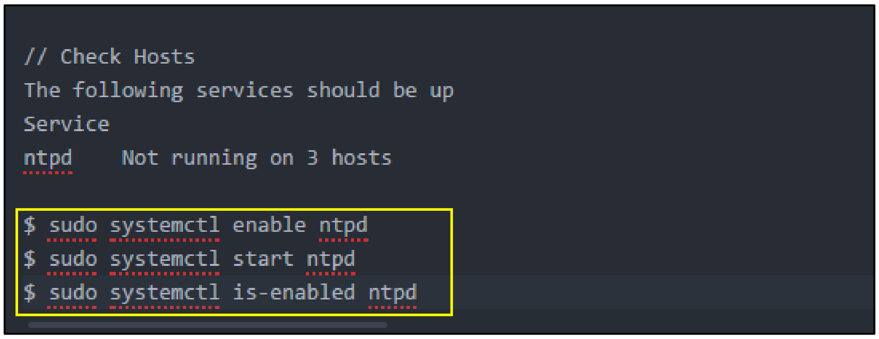
Reverse Lookup Issues (0) ✓

Hostname Resolution Issues (0) ✓

Figure 7.5.1

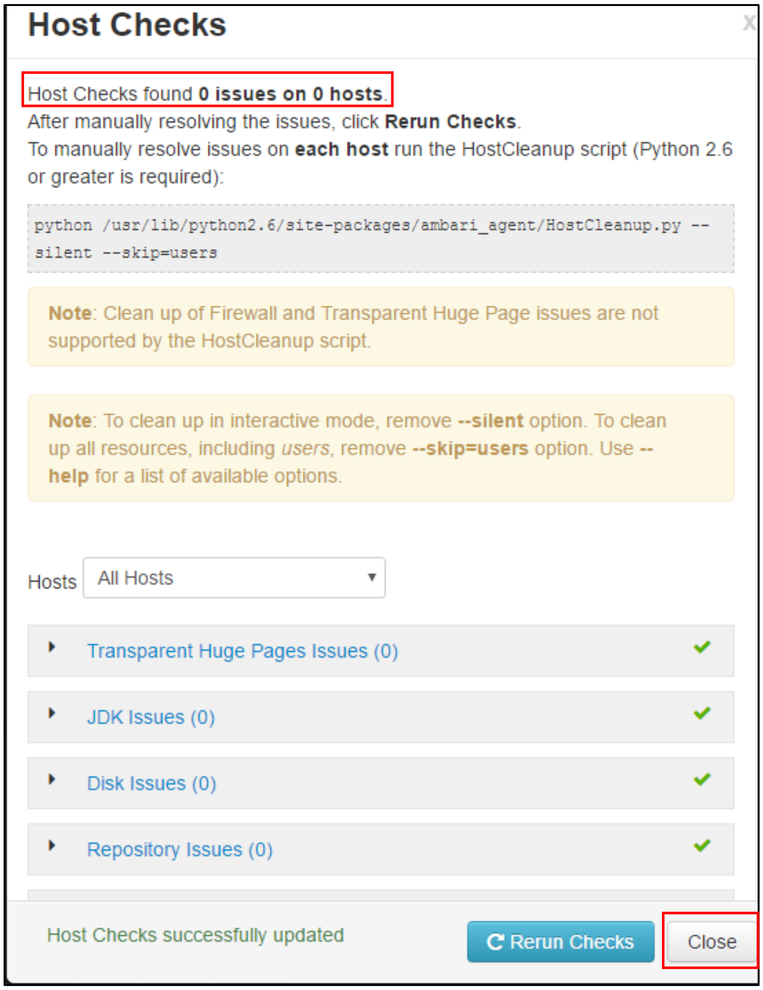
If the hosts have any issue, Ambari installation wizard will highlight it to the user as shown in Figure 7.5.1

In my case, I had to enable “ntpd” (NTP daemon). I ran the following commands shown in Figure 7.5.2.



```
// Check Hosts
The following services should be up
Service
ntpd    Not running on 3 hosts

$ sudo systemctl enable ntpd
$ sudo systemctl start ntpd
$ sudo systemctl is-enabled ntpd
```

Figure 7.5.2



Host Checks

Host Checks found **0 issues on 0 hosts**.
After manually resolving the issues, click **Rerun Checks**.
To manually resolve issues on **each host** run the HostCleanup script (Python 2.6 or greater is required):

```
python /usr/lib/python2.6/site-packages/ambari_agent/HostCleanup.py --
silent --skip=users
```

Note: Clean up of Firewall and Transparent Huge Page issues are not supported by the HostCleanup script.

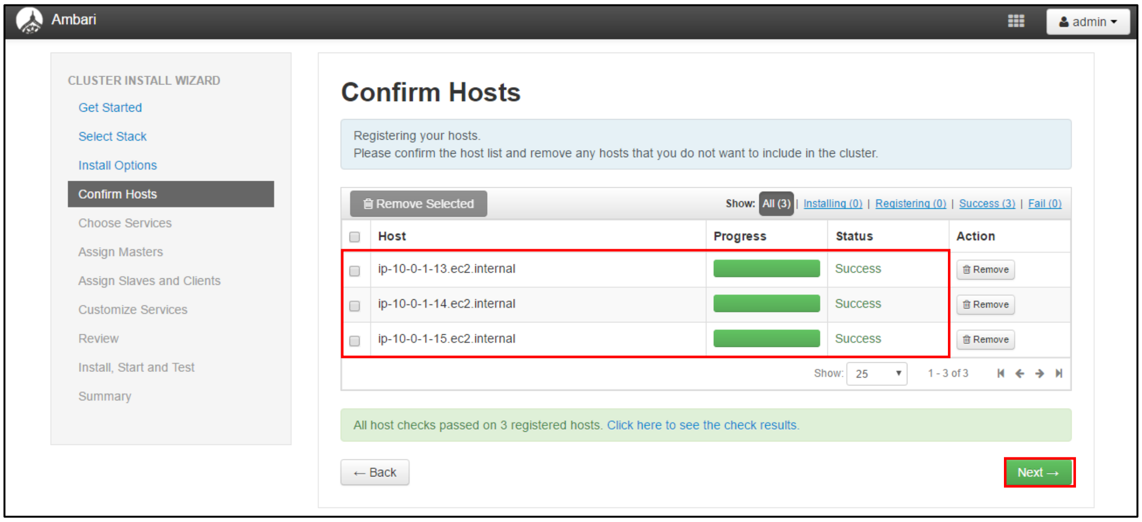
Note: To clean up in interactive mode, remove **--silent** option. To clean up all resources, including *users*, remove **--skip=users** option. Use **--help** for a list of available options.

Hosts: All Hosts

- Transparent Huge Pages Issues (0) ✓
- JDK Issues (0) ✓
- Disk Issues (0) ✓
- Repository Issues (0) ✓

Host Checks successfully updated

Figure 7.5.3



Confirm Hosts

Registering your hosts.
Please confirm the host list and remove any hosts that you do not want to include in the cluster.

Remove Selected   Show: All (3) | Installing (0) | Registering (0) | Success (3) | Fail (0)

□	Host	Progress	Status	Action
□	ip-10-0-1-13.ec2.internal	🟩	Success	Remove
□	ip-10-0-1-14.ec2.internal	🟩	Success	Remove
□	ip-10-0-1-15.ec2.internal	🟩	Success	Remove

Show: 25   1 - 3 of 3

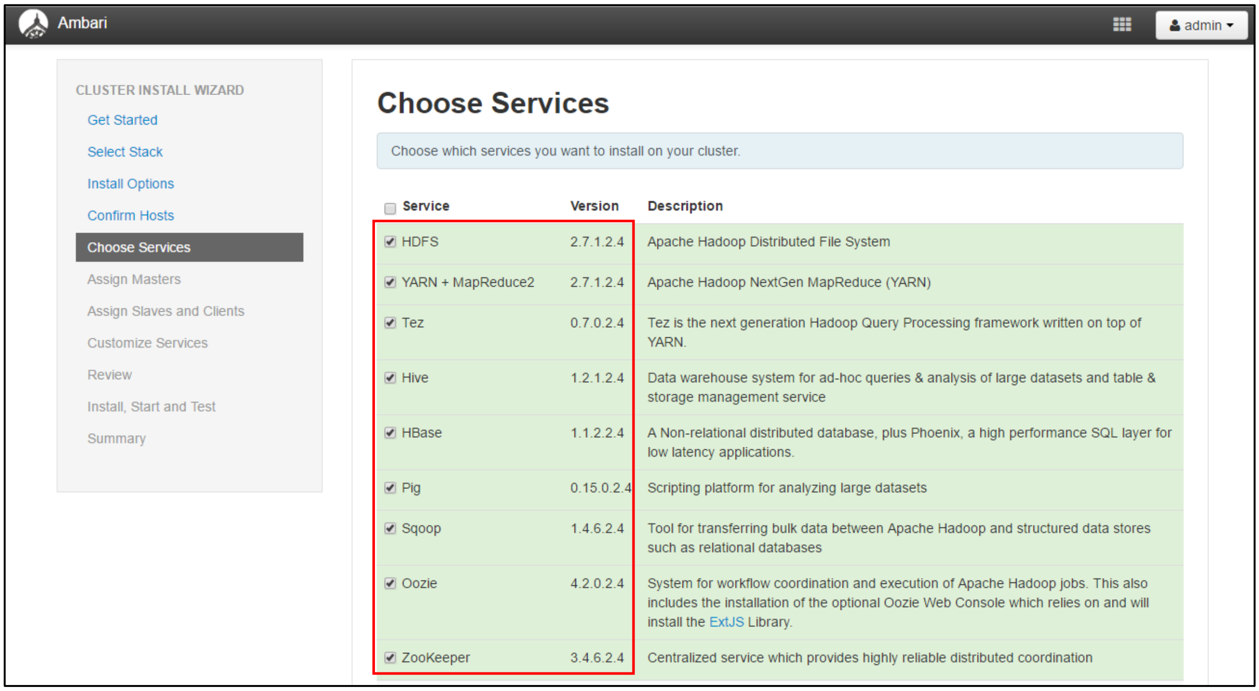
All host checks passed on 3 registered hosts. Click here to see the check results.

Figure 7.5.4

If the Ambari Installation Wizard, does not find any issues with the hosts, it report zero errors as highlighted in Figure 7.5.3.

Also successful registration of the hosts by the Ambari Server is checked “green” as highlighted in Figure 7.5.4.

7.6 CHOOSE SERVICES

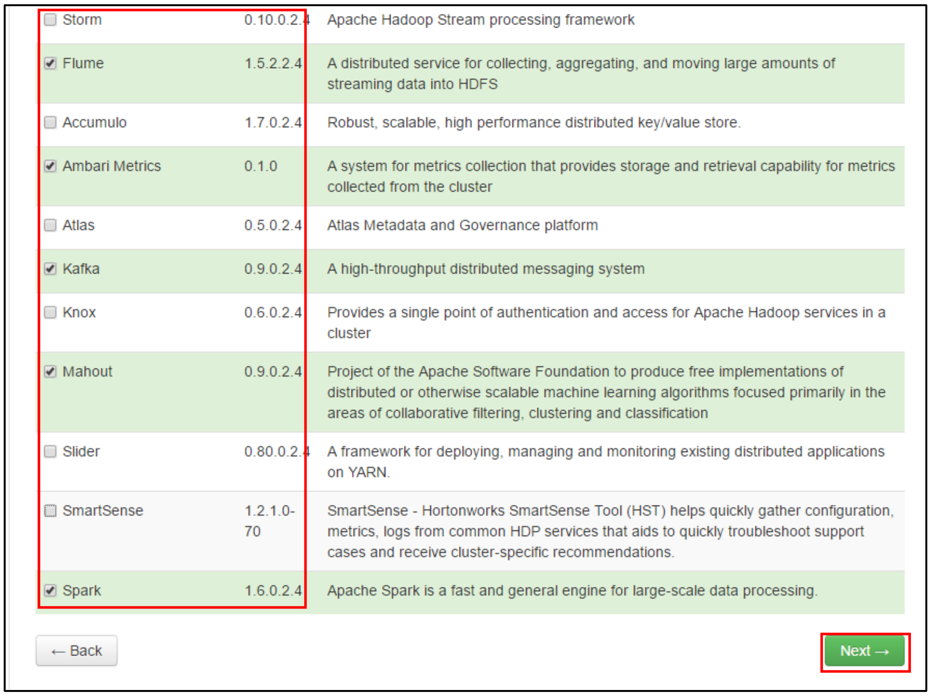


Choose Services

Choose which services you want to install on your cluster.

□ Service	Version	Description
☑ HDFS	2.7.1.2.4	Apache Hadoop Distributed File System
☑ YARN + MapReduce2	2.7.1.2.4	Apache Hadoop NextGen MapReduce (YARN)
☑ Tez	0.7.0.2.4	Tez is the next generation Hadoop Query Processing framework written on top of YARN.
☑ Hive	1.2.1.2.4	Data warehouse system for ad-hoc queries & analysis of large datasets and table & storage management service
☑ HBase	1.1.2.2.4	A Non-relational distributed database, plus Phoenix, a high performance SQL layer for low latency applications.
☑ Pig	0.15.0.2.4	Scripting platform for analyzing large datasets
☑ Sqoop	1.4.6.2.4	Tool for transferring bulk data between Apache Hadoop and structured data stores such as relational databases
☑ Oozie	4.2.0.2.4	System for workflow coordination and execution of Apache Hadoop jobs. This also includes the installation of the optional Oozie Web Console which relies on and will install the ExtJS Library.
☑ ZooKeeper	3.4.6.2.4	Centralized service which provides highly reliable distributed coordination

Figure 7.6.1



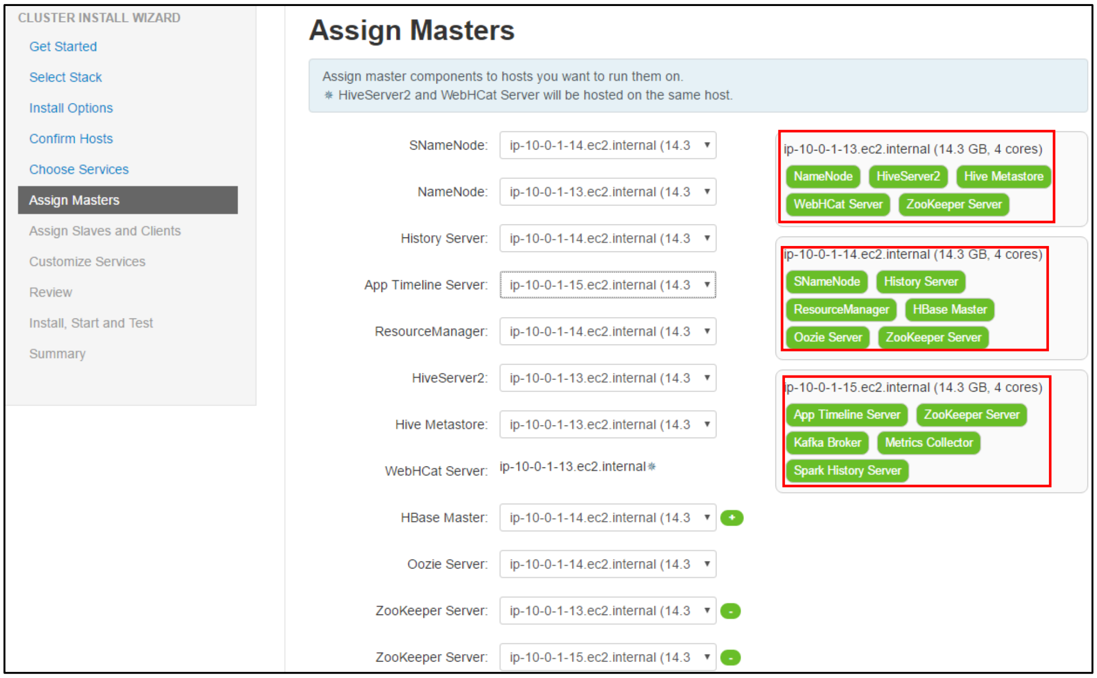
	Service	Version	Description
☐	Storm	0.10.0.2.4	Apache Hadoop Stream processing framework
☑	Flume	1.5.2.2.4	A distributed service for collecting, aggregating, and moving large amounts of streaming data into HDFS
☐	Accumulo	1.7.0.2.4	Robust, scalable, high performance distributed key/value store.
☑	Ambari Metrics	0.1.0	A system for metrics collection that provides storage and retrieval capability for metrics collected from the cluster
☐	Atlas	0.5.0.2.4	Atlas Metadata and Governance platform
☑	Kafka	0.9.0.2.4	A high-throughput distributed messaging system
☐	Knox	0.6.0.2.4	Provides a single point of authentication and access for Apache Hadoop services in a cluster
☑	Mahout	0.9.0.2.4	Project of the Apache Software Foundation to produce free implementations of distributed or otherwise scalable machine learning algorithms focused primarily in the areas of collaborative filtering, clustering and classification
☐	Slider	0.80.0.2.4	A framework for deploying, managing and monitoring existing distributed applications on YARN.
☐	SmartSense	1.2.1.0-70	SmartSense - Hortonworks SmartSense Tool (HST) helps quickly gather configuration, metrics, logs from common HDP services that aids to quickly troubleshoot support cases and receive cluster-specific recommendations.
☑	Spark	1.6.0.2.4	Apache Spark is a fast and general engine for large-scale data processing.

Figure 7.6.2

I chose services and tools for my cluster. You can choose, the tools you would like to test on your cluster.

Click “Next”.

7.7 Assign Masters



Assign Masters

Assign master components to hosts you want to run them on.
✻ HiveServer2 and WebHCat Server will be hosted on the same host.

SNameNode:	ip-10-0-1-14.ec2.internal (14.3
NameNode:	ip-10-0-1-13.ec2.internal (14.3
History Server:	ip-10-0-1-14.ec2.internal (14.3
App Timeline Server:	ip-10-0-1-15.ec2.internal (14.3
ResourceManager:	ip-10-0-1-14.ec2.internal (14.3
HiveServer2:	ip-10-0-1-13.ec2.internal (14.3
Hive Metastore:	ip-10-0-1-13.ec2.internal (14.3
WebHCat Server:	ip-10-0-1-13.ec2.internal✻
HBase Master:	ip-10-0-1-14.ec2.internal (14.3
Oozie Server:	ip-10-0-1-14.ec2.internal (14.3
ZooKeeper Server:	ip-10-0-1-13.ec2.internal (14.3
ZooKeeper Server:	ip-10-0-1-15.ec2.internal (14.3

ip-10-0-1-13.ec2.internal (14.3 GB, 4 cores)
NameNode, HiveServer2, Hive Metastore, WebHCat Server, ZooKeeper Server

ip-10-0-1-14.ec2.internal (14.3 GB, 4 cores)
SNameNode, History Server, ResourceManager, HBase Master, Oozie Server, ZooKeeper Server

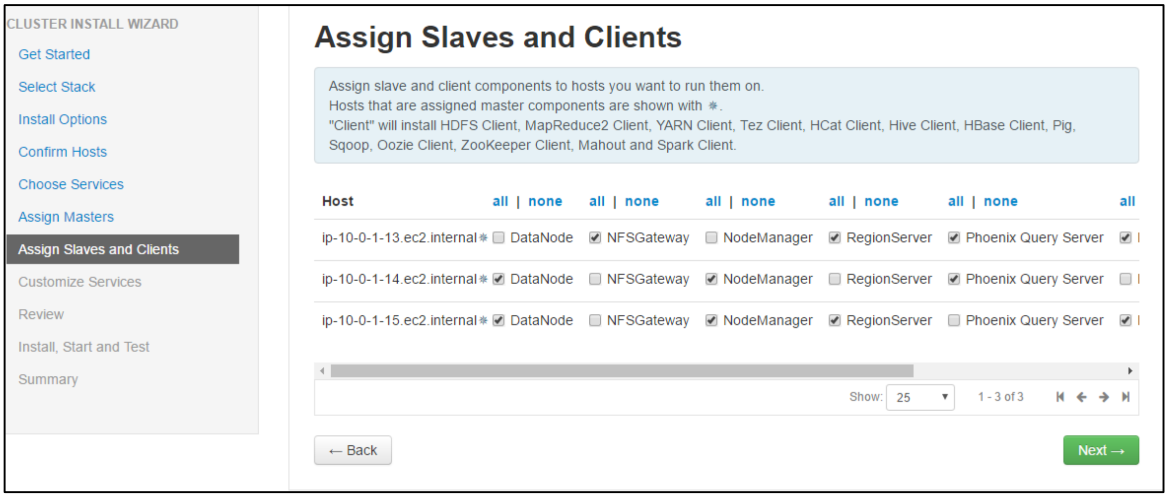
ip-10-0-1-15.ec2.internal (14.3 GB, 4 cores)
App Timeline Server, ZooKeeper Server, Kafka Broker, Metrics Collector, Spark History Server

Figure 7.7.1

Assign masters, as to which instance will host, which service.

I spread my Hadoop services as shown in Figure 7.7.1. You can plan your services on the hosts in the same manner.

7.8 Assign Slaves and Clients

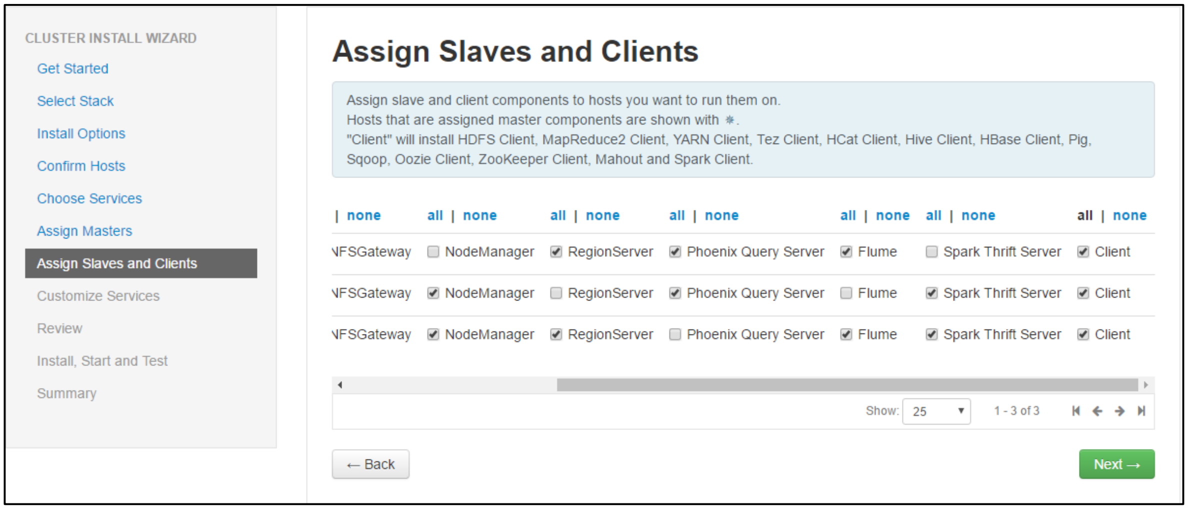


Assign Slaves and Clients

Assign slave and client components to hosts you want to run them on.
Hosts that are assigned master components are shown with ✱.
"Client" will install HDFS Client, MapReduce2 Client, YARN Client, Tez Client, HCat Client, Hive Client, HBase Client, Pig, Sqoop, Oozie Client, ZooKeeper Client, Mahout and Spark Client.

Host	all | none	all | none	all | none	all | none	all | none	all
ip-10-0-1-13.ec2.internal✱	□ DataNode	☑ NFSGateway	□ NodeManager	☑ RegionServer	☑ Phoenix Query Server	☑ I
ip-10-0-1-14.ec2.internal✱	☑ DataNode	□ NFSGateway	☑ NodeManager	□ RegionServer	☑ Phoenix Query Server	□ I
ip-10-0-1-15.ec2.internal✱	☑ DataNode	□ NFSGateway	☑ NodeManager	☑ RegionServer	□ Phoenix Query Server	☑ I

Figure 7.8.1



Assign Slaves and Clients

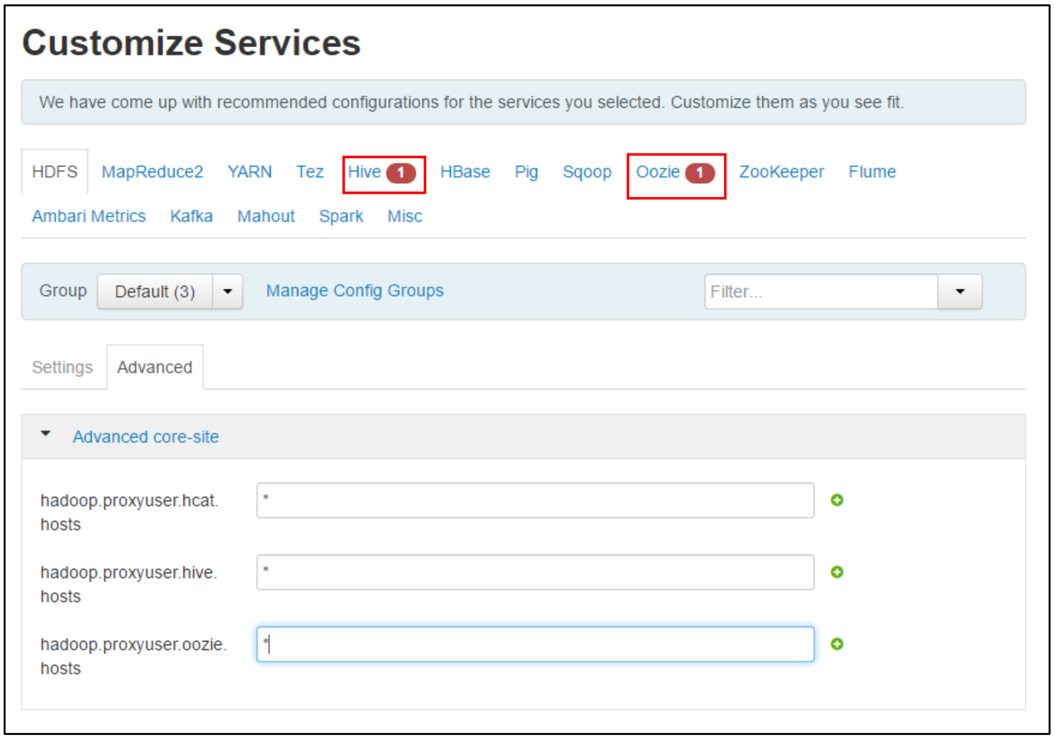
Assign slave and client components to hosts you want to run them on.
Hosts that are assigned master components are shown with ✱.
"Client" will install HDFS Client, MapReduce2 Client, YARN Client, Tez Client, HCat Client, Hive Client, HBase Client, Pig, Sqoop, Oozie Client, ZooKeeper Client, Mahout and Spark Client.

| none	all | none	all | none	all | none	all | none	all | none	all | none
NFSGateway	□ NodeManager	☑ RegionServer	☑ Phoenix Query Server	☑ Flume	□ Spark Thrift Server	☑ Client
NFSGateway	☑ NodeManager	□ RegionServer	☑ Phoenix Query Server	□ Flume	☑ Spark Thrift Server	☑ Client
NFSGateway	☑ NodeManager	☑ RegionServer	□ Phoenix Query Server	☑ Flume	☑ Spark Thrift Server	☑ Client

Figure 7.8.2

Decide, the data node instances on your cluster and accordingly install the services and clients required.

7.9 Customize Services



Customize Services

We have come up with recommended configurations for the services you selected. Customize them as you see fit.

HDFS MapReduce2 YARN Tez Hive 1 HBase Pig Sqoop Oozie 1 ZooKeeper Flume
Ambari Metrics Kafka Mahout Spark Misc

Group Default (3) Manage Config Groups Filter...

Settings Advanced

Advanced core-site

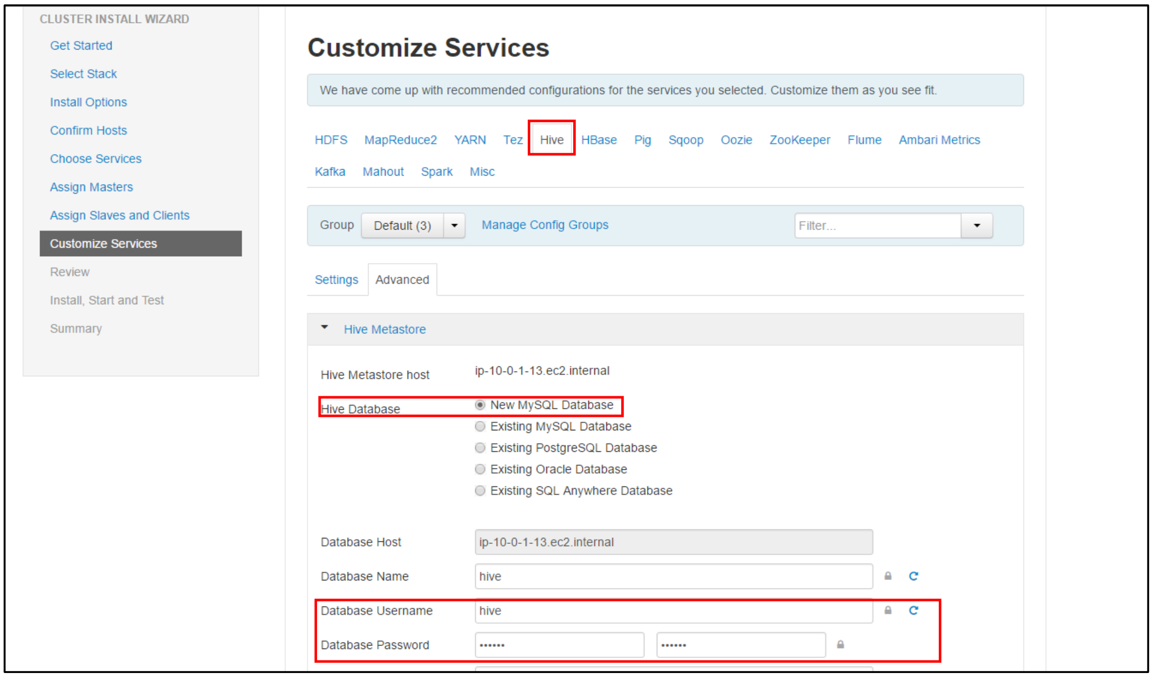
hadoop.proxyuser.hcat.hosts: *

hadoop.proxyuser.hive.hosts: *

hadoop.proxyuser.oozie.hosts: *

Figure 7.9.1

Figure 7.9.1 shows services highlighted, I had to attend and resolve. You might come across similar issue.



CLUSTER INSTALL WIZARD
- Get Started
- Select Stack
- Install Options
- Confirm Hosts
- Choose Services
- Assign Masters
- Assign Slaves and Clients
- Customize Services
- Review
- Install, Start and Test
- Summary

Customize Services

We have come up with recommended configurations for the services you selected. Customize them as you see fit.

HDFS MapReduce2 YARN Tez Hive HBase Pig Sqoop Oozie ZooKeeper Flume Ambari Metrics
Kafka Mahout Spark Misc

Group Default (3) Manage Config Groups Filter...

Settings Advanced

Hive Metastore

Hive Metastore host: ip-10-0-1-13.ec2.internal

Hive Database:
- ● New MySQL Database
- ○ Existing MySQL Database
- ○ Existing PostgreSQL Database
- ○ Existing Oracle Database
- ○ Existing SQL Anywhere Database

Database Host: ip-10-0-1-13.ec2.internal

Database Name: hive

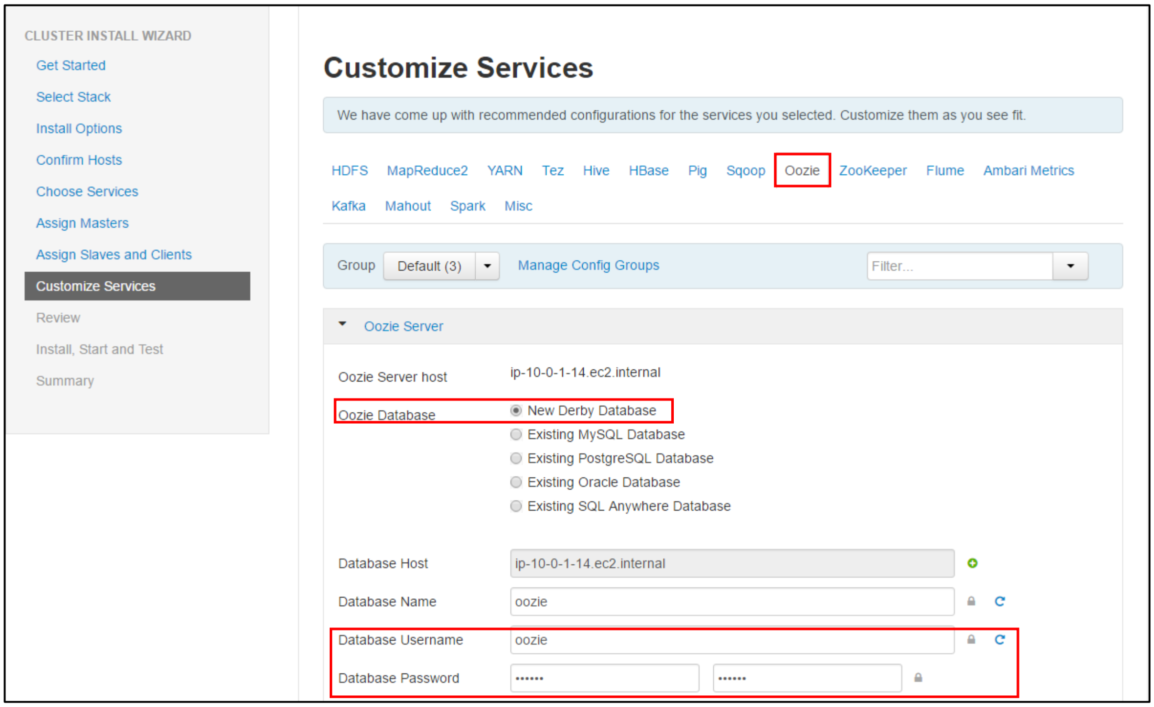
Database Username: hive

Database Password: •••••• ••••••

Figure 7.9.2

The Hive Database needed configuration. I created a new MySQL Database for Hive metastore.

Provide the username and password. Make sure, you document or note down the username and password.



Customize Services

We have come up with recommended configurations for the services you selected. Customize them as you see fit.

Oozie Server host: ip-10-0-1-14.ec2.internal

Oozie Database:
● New Derby Database
○ Existing MySQL Database
○ Existing PostgreSQL Database
○ Existing Oracle Database
○ Existing SQL Anywhere Database

Database Host: ip-10-0-1-14.ec2.internal

Database Name: oozie

Database Username: oozie

Database Password: •••••• ••••••

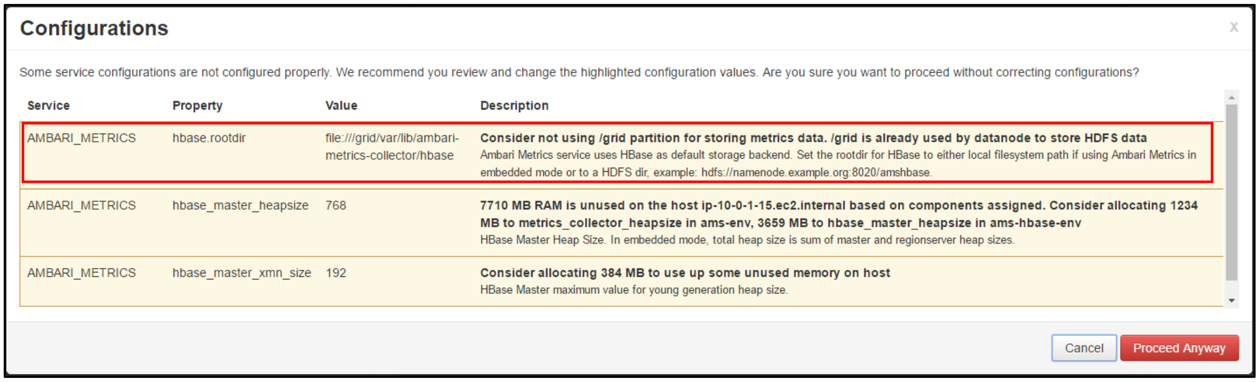
Figure 7.9.3

The Oozie Database needed configuration. I created a new Derby Database for Oozie DB.

Provide the username and password. Make sure, you document or note down the username and password.

NOTE:

- Derby DB is not recommended for production use.



Configurations

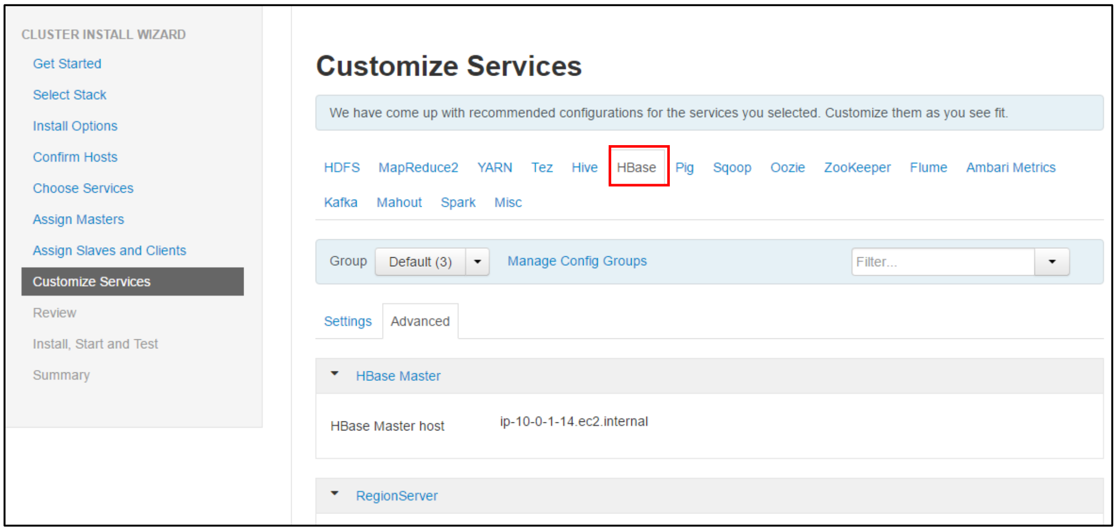
Some service configurations are not configured properly. We recommend you review and change the highlighted configuration values. Are you sure you want to proceed without correcting configurations?

Service	Property	Value	Description
AMBARI_METRICS	hbase.rootdir	file:///grid/var/lib/ambari-metrics-collector/hbase	**Consider not using /grid partition for storing metrics data. /grid is already used by datanode to store HDFS data** Ambari Metrics service uses HBase as default storage backend. Set the rootdir for HBase to either local filesystem path if using Ambari Metrics in embedded mode or to a HDFS dir, example: hdfs://namenode.example.org:8020/amshbase.
AMBARI_METRICS	hbase_master_heapsize	768	**7710 MB RAM is unused on the host ip-10-0-1-15.ec2.internal based on components assigned. Consider allocating 1234 MB to metrics_collector_heapsize in ams-env, 3659 MB to hbase_master_heapsize in ams-hbase-env** HBase Master Heap Size. In embedded mode, total heap size is sum of master and regionserver heap sizes.
AMBARI_METRICS	hbase_master_xmn_size	192	**Consider allocating 384 MB to use up some unused memory on host** HBase Master maximum value for young generation heap size.

Figure 7.9.4

When you resolve all warnings and click “Proceed”, you might get few more warning with respect to Ambari Metrics memory allocation as shown in Figure 7.9.4.

Ambari Installation Wizard will suggest you to assign Ambari Metrics root directory. You can set it by locating, where the HBase root directory is located as highlighted in Figure 7.9.5

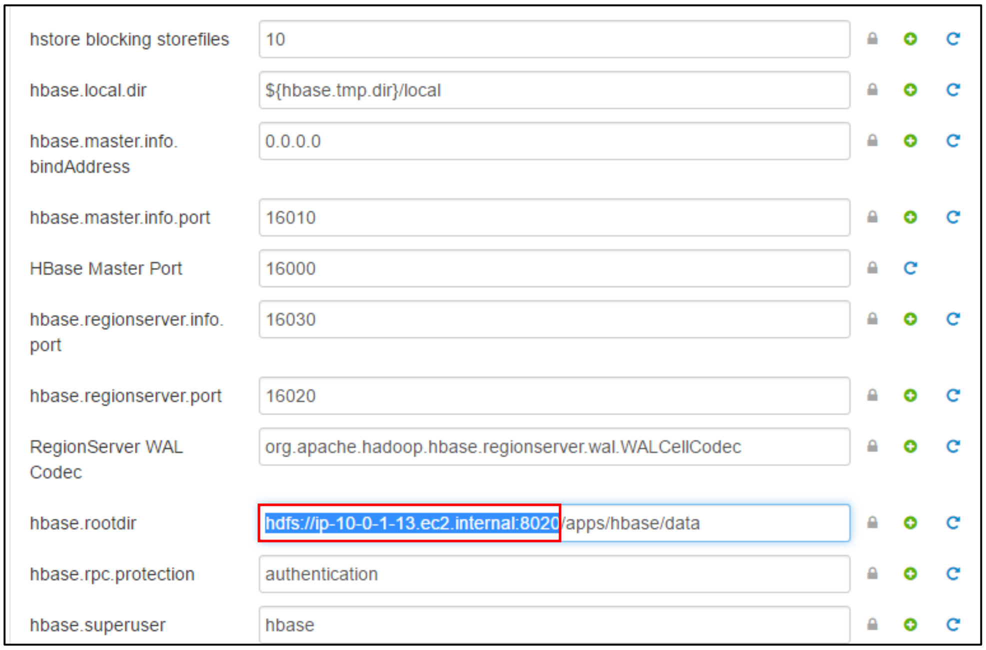


Customize Services

We have come up with recommended configurations for the services you selected. Customize them as you see fit.

HBase Master host ip-10-0-1-14.ec2.internal

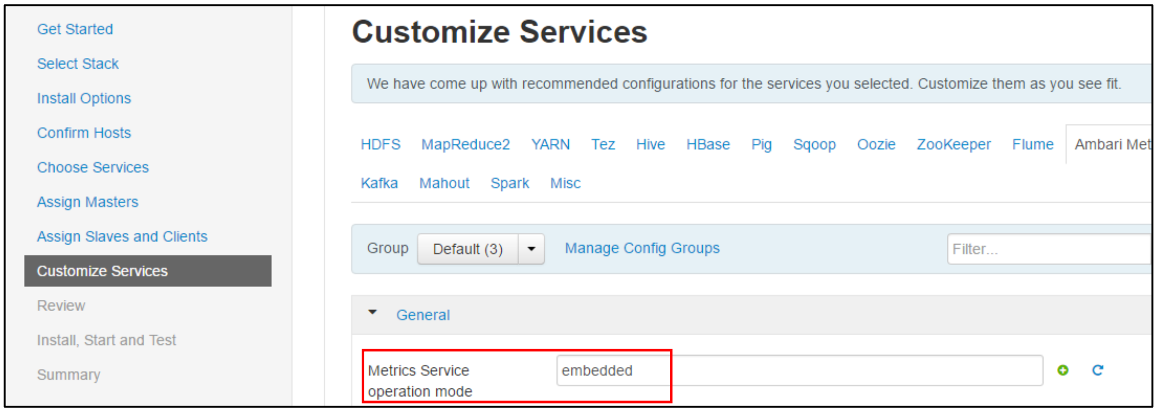
Figure 7.9.5



hstore blocking storefiles	10
hbase.local.dir	${hbase.tmp.dir}/local
hbase.master.info.bindAddress	0.0.0.0
hbase.master.info.port	16010
HBase Master Port	16000
hbase.regionserver.info.port	16030
hbase.regionserver.port	16020
RegionServer WAL Codec	org.apache.hadoop.hbase.regionserver.wal.WALCellCodec
hbase.rootdir	hdfs://ip-10-0-1-13.ec2.internal:8020/apps/hbase/data
hbase.rpc.protection	authentication
hbase.superuser	hbase

Figure 7.9.6

You can copy the path of your Hbase root directory as highlighted in Figure 7.9.6.

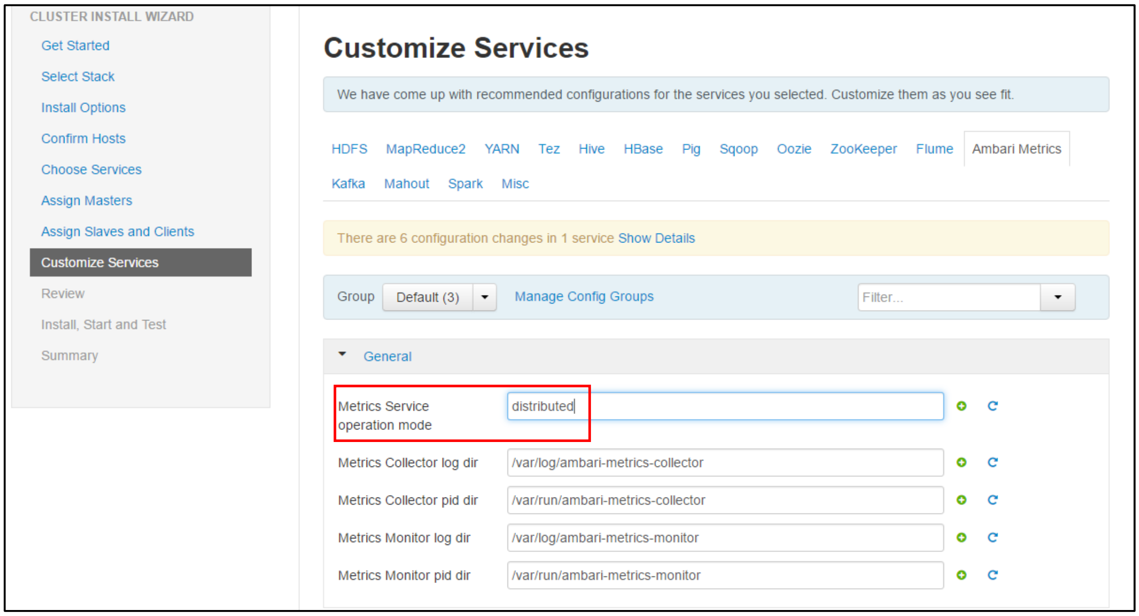


Customize Services

We have come up with recommended configurations for the services you selected. Customize them as you see fit.

Metrics Service operation mode: embedded

Figure 7.9.7



Customize Services

We have come up with recommended configurations for the services you selected. Customize them as you see fit.

There are 6 configuration changes in 1 service Show Details

Metrics Service operation mode: distributed

Metrics Collector log dir: /var/log/ambari-metrics-collector

Metrics Collector pid dir: /var/run/ambari-metrics-collector

Metrics Monitor log dir: /var/log/ambari-metrics-monitor

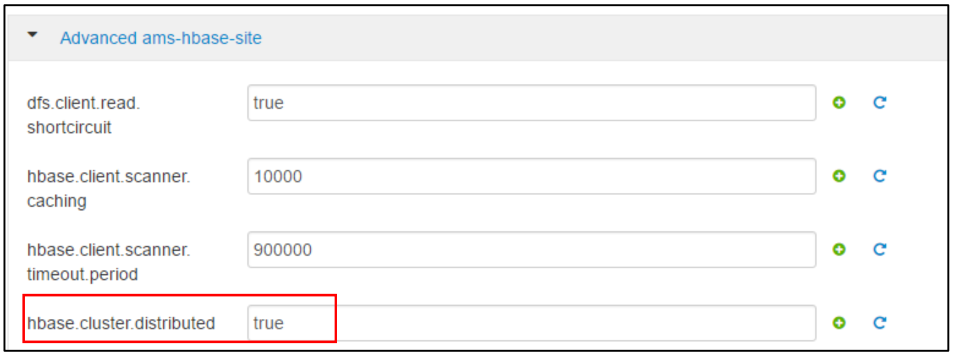
Metrics Monitor pid dir: /var/run/ambari-metrics-monitor

Figure 7.9.8

Change the value for the field “Metrics Service operation mode” from “embedded” to “distributed” highlighted in Figure 7.9.7 to Figure 7.9.8.

Also set “hbase.cluster.distributed” field to “true” as highlighted in Figure 7.9.9.

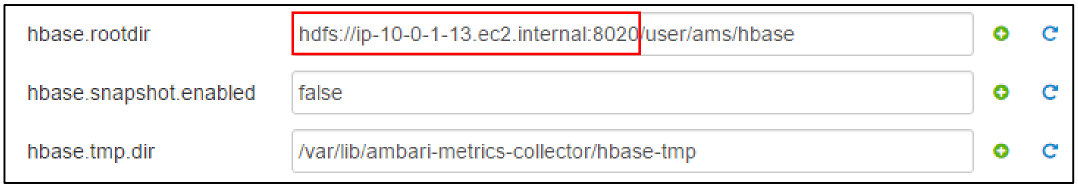
You will notice ambari metrics root has similar directory path as Hbase root directory, highlighted in Figure 7.9.10.



Advanced ams-hbase-site

dfs.client.read.shortcircuit	true
hbase.client.scanner.caching	10000
hbase.client.scanner.timeout.period	900000
hbase.cluster.distributed	true

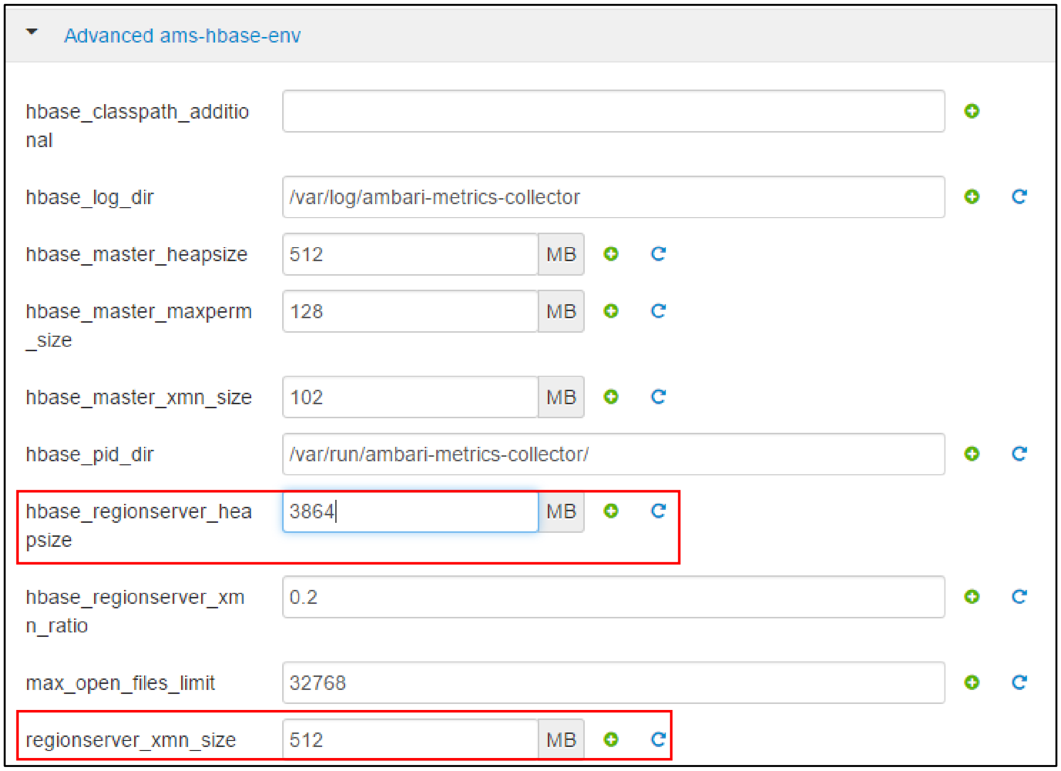
Figure 7.9.9



hbase.rootdir	hdfs://ip-10-0-1-13.ec2.internal:8020/user/ams/hbase
hbase.snapshot.enabled	false
hbase.tmp.dir	/var/lib/ambari-metrics-collector/hbase-tmp

Figure 7.9.10

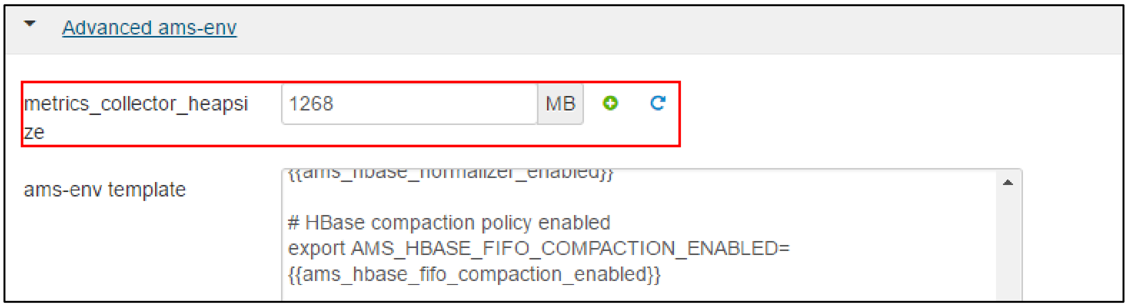
Set the “hbase_regionserver_heapsize” and “regionserver_xmn_size” to the recommended value



Advanced ams-hbase-env

hbase_classpath_additional	
hbase_log_dir	/var/log/ambari-metrics-collector
hbase_master_heapsize	512 MB
hbase_master_maxperm_size	128 MB
hbase_master_xmn_size	102 MB
hbase_pid_dir	/var/run/ambari-metrics-collector/
hbase_regionserver_heapsize	3864 MB
hbase_regionserver_xmn_ratio	0.2
max_open_files_limit	32768
regionserver_xmn_size	512 MB

Figure 7.9.11

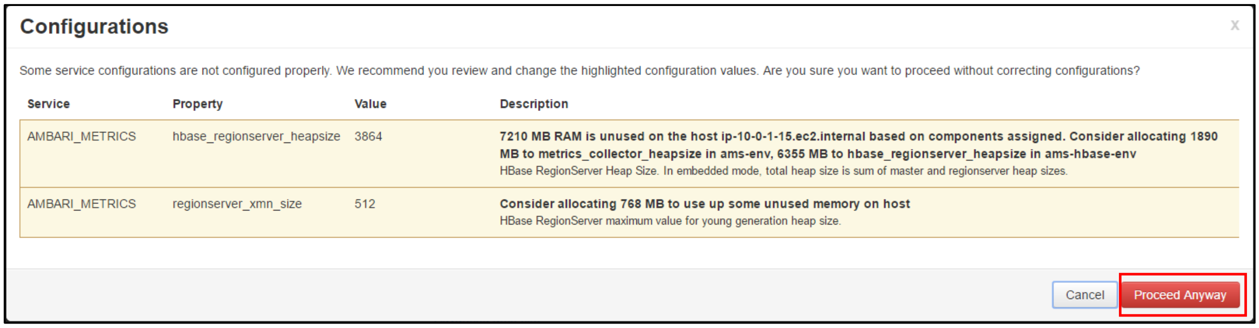


metrics_collector_heapsize 1268 MB

ams-env template

```
{{ams_hbase_normalizer_enabled}}

# HBase compaction policy enabled
export AMS_HBASE_FIFO_COMPACTION_ENABLED=
{{ams_hbase_fifo_compaction_enabled}}
```

Figure 7.9.12



Configurations

Some service configurations are not configured properly. We recommend you review and change the highlighted configuration values. Are you sure you want to proceed without correcting configurations?

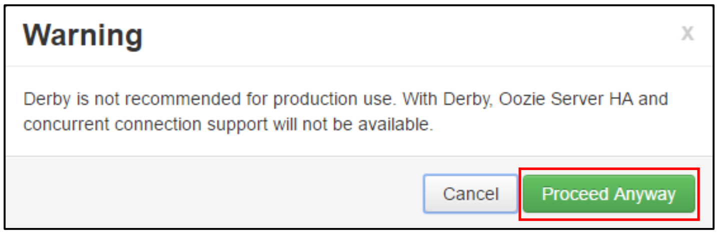
Service	Property	Value	Description
AMBARI_METRICS	hbase_regionserver_heapsize	3864	**7210 MB RAM is unused on the host ip-10-0-1-15.ec2.internal based on components assigned. Consider allocating 1890 MB to metrics_collector_heapsize in ams-env, 6355 MB to hbase_regionserver_heapsize in ams-hbase-env** HBase RegionServer Heap Size. In embedded mode, total heap size is sum of master and regionserver heap sizes.
AMBARI_METRICS	regionserver_xmn_size	512	**Consider allocating 768 MB to use up some unused memory on host** HBase RegionServer maximum value for young generation heap size.

Figure 7.9.13

I clicked “Proceed Anyway” after configuring “metrics_collector_heapsize” to “1268” highlighted in Figure 7.9.12.

Warnings may still appear. You can look for help by looking at Hortonworks Community forum.

Visit: http://hortonworks.com/community/



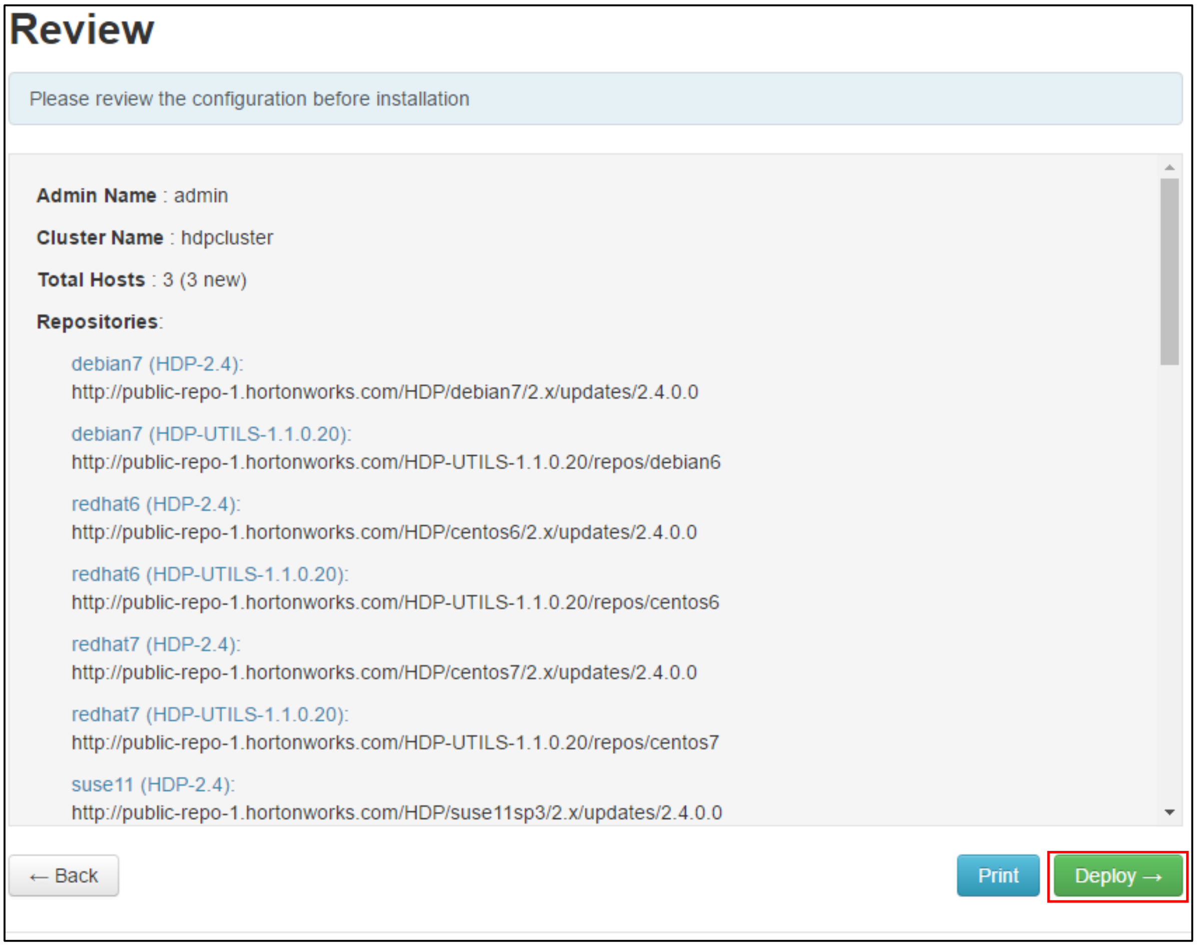
Warning

Derby is not recommended for production use. With Derby, Oozie Server HA and concurrent connection support will not be available.

Figure 7.9.14

I mentioned earlier, stating Derby DB is not recommended for production use. When you install, you will also be provided with a similar warning. Click “Procced Anyway”.

7.10 Review



Review

Please review the configuration before installation

Admin Name : admin

Cluster Name : hdpcluster

Total Hosts : 3 (3 new)

Repositories:

debian7 (HDP-2.4):
http://public-repo-1.hortonworks.com/HDP/debian7/2.x/updates/2.4.0.0

debian7 (HDP-UTILS-1.1.0.20):
http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.20/repos/debian6

redhat6 (HDP-2.4):
http://public-repo-1.hortonworks.com/HDP/centos6/2.x/updates/2.4.0.0

redhat6 (HDP-UTILS-1.1.0.20):
http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.20/repos/centos6

redhat7 (HDP-2.4):
http://public-repo-1.hortonworks.com/HDP/centos7/2.x/updates/2.4.0.0

redhat7 (HDP-UTILS-1.1.0.20):
http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.20/repos/centos7

suse11 (HDP-2.4):
http://public-repo-1.hortonworks.com/HDP/suse11sp3/2.x/updates/2.4.0.0

Figure 7.10.1

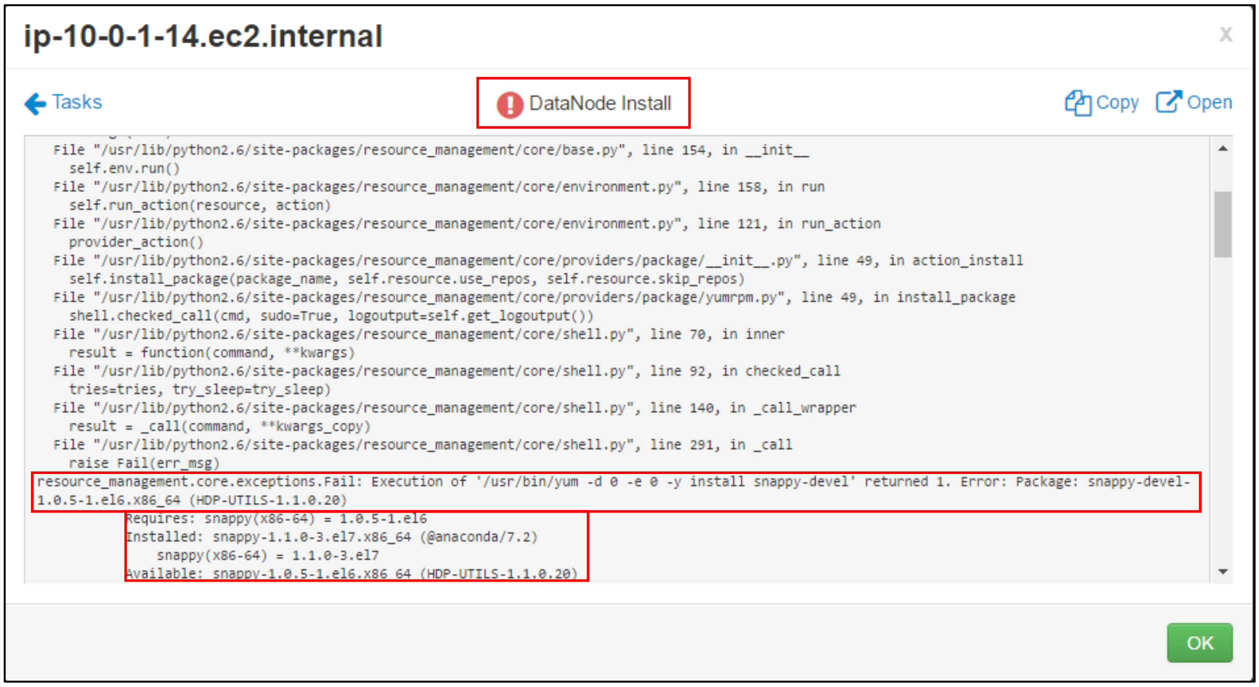
Review the services and clients, that will be launched on your cluster on AWS.

7.11 Install, Start and Test

Installation of Hadoop services and Clients is fairly easy, if you have right libraries and binaries.

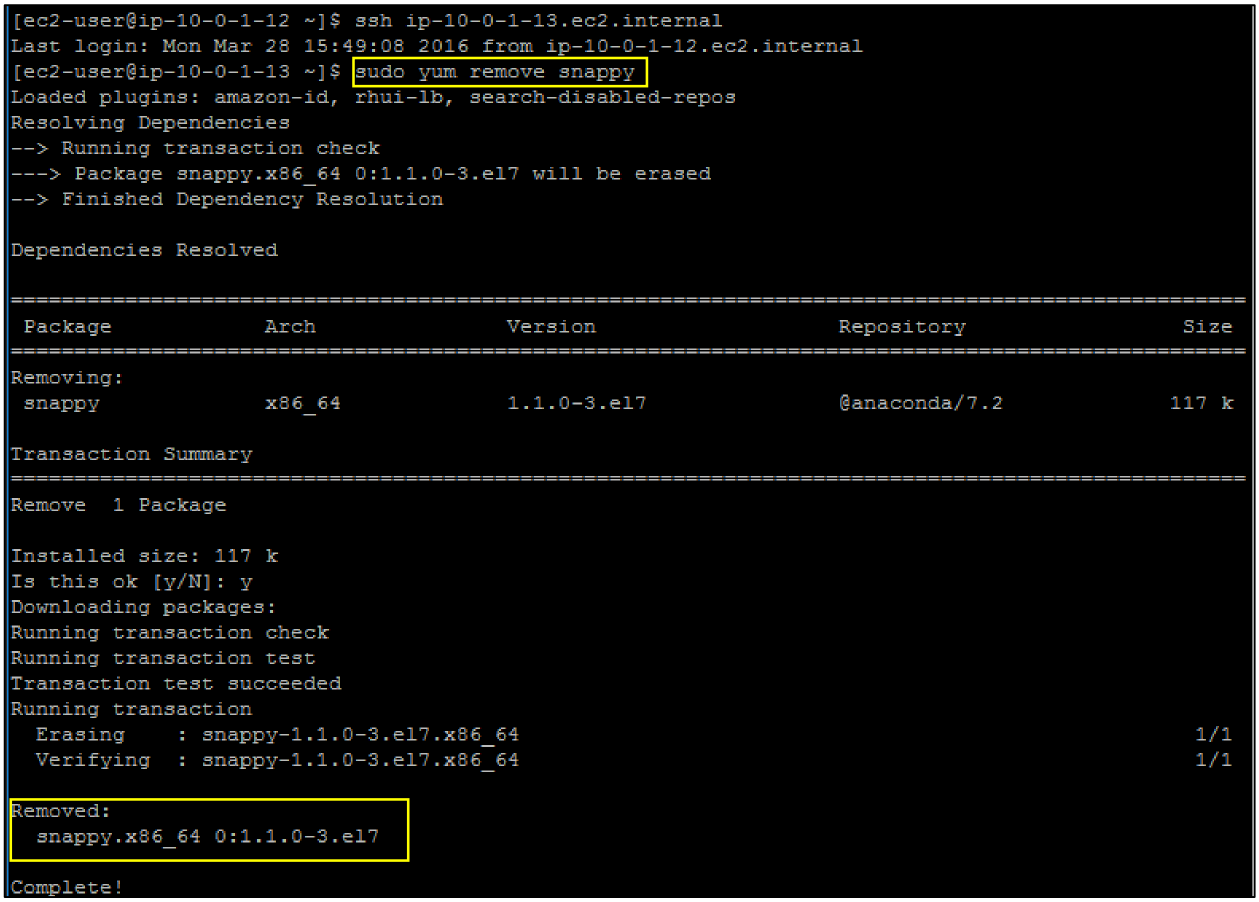
In my installation process, I had faced the problem of my hosts missing “snappy-devel” libraries.

When, I did my research as to why my installation failed, I realized that HDP repo-list did not install “snappy-devel” library but just “snappy” library. Refer Figure 7.11.1.



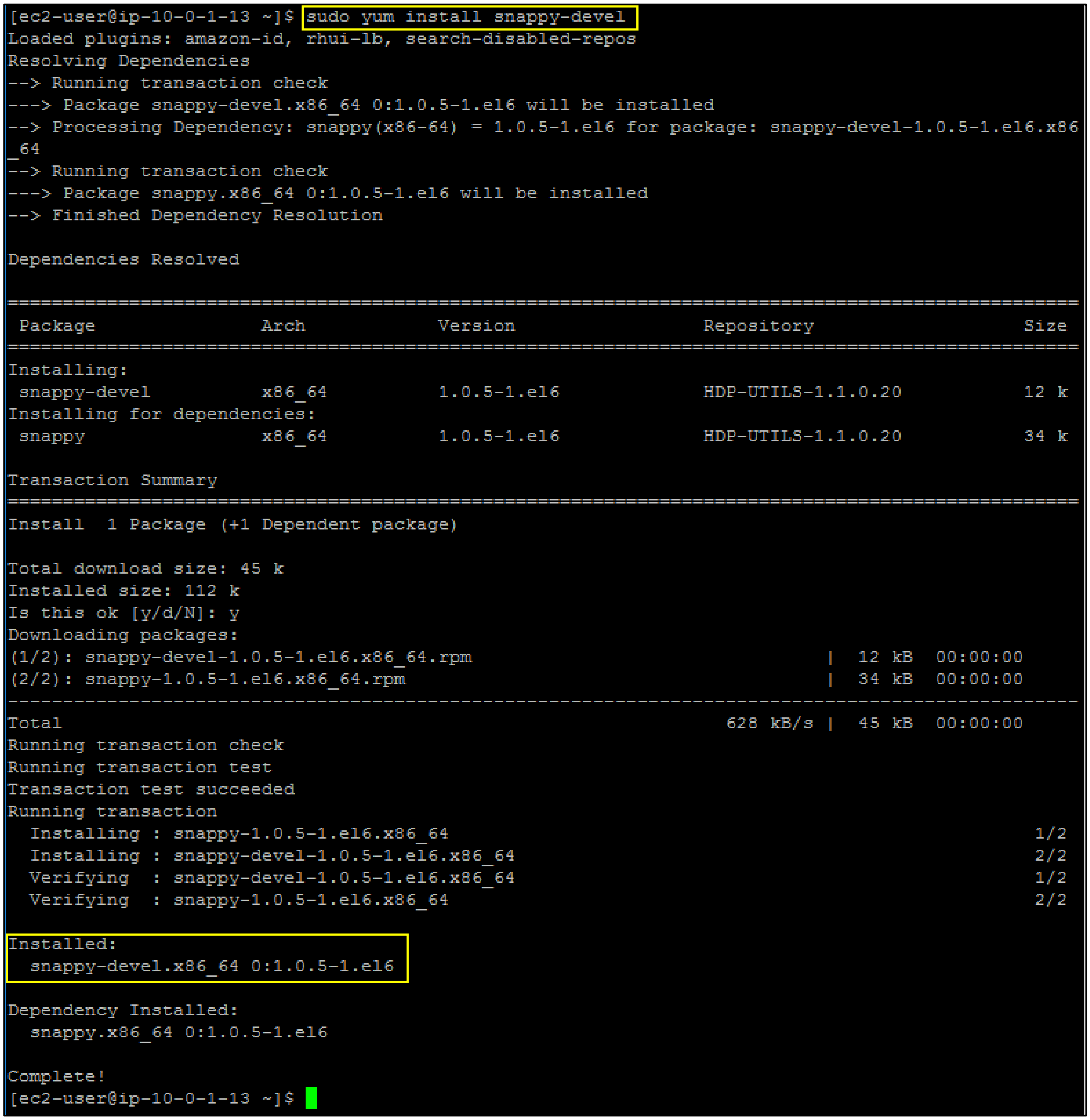
```
  File "/usr/lib/python2.6/site-packages/resource_management/core/base.py", line 154, in __init__
    self.env.run()
  File "/usr/lib/python2.6/site-packages/resource_management/core/environment.py", line 158, in run
    self.run_action(resource, action)
  File "/usr/lib/python2.6/site-packages/resource_management/core/environment.py", line 121, in run_action
    provider_action()
  File "/usr/lib/python2.6/site-packages/resource_management/core/providers/package/__init__.py", line 49, in action_install
    self.install_package(package_name, self.resource.use_repos, self.resource.skip_repos)
  File "/usr/lib/python2.6/site-packages/resource_management/core/providers/package/yumrpm.py", line 49, in install_package
    shell.checked_call(cmd, sudo=True, logoutput=self.get_logoutput())
  File "/usr/lib/python2.6/site-packages/resource_management/core/shell.py", line 70, in inner
    result = function(command, **kwargs)
  File "/usr/lib/python2.6/site-packages/resource_management/core/shell.py", line 92, in checked_call
    tries=tries, try_sleep=try_sleep)
  File "/usr/lib/python2.6/site-packages/resource_management/core/shell.py", line 140, in _call_wrapper
    result = _call(command, **kwargs_copy)
  File "/usr/lib/python2.6/site-packages/resource_management/core/shell.py", line 291, in _call
    raise Fail(err_msg)
resource_management.core.exceptions.Fail: Execution of '/usr/bin/yum -d 0 -e 0 -y install snappy-devel' returned 1. Error: Package: snappy-devel-
1.0.5-1.el6.x86_64 (HDP-UTILS-1.1.0.20)
           Requires: snappy(x86-64) = 1.0.5-1.el6
           Installed: snappy-1.1.0-3.el7.x86_64 (@anaconda/7.2)
               snappy(x86-64) = 1.1.0-3.el7
           Available: snappy-1.0.5-1.el6.x86_64 (HDP-UTILS-1.1.0.20)
```

Figure 7.11.1



```
[ec2-user@ip-10-0-1-12 ~]$ ssh ip-10-0-1-13.ec2.internal
Last login: Mon Mar 28 15:49:08 2016 from ip-10-0-1-12.ec2.internal
[ec2-user@ip-10-0-1-13 ~]$ sudo yum remove snappy
Loaded plugins: amazon-id, rhui-lb, search-disabled-repos
Resolving Dependencies
--> Running transaction check
---> Package snappy.x86_64 0:1.1.0-3.el7 will be erased
--> Finished Dependency Resolution

Dependencies Resolved

=================================================================================================
 Package                Arch                Version                   Repository            Size
=================================================================================================
Removing:
 snappy                 x86_64              1.1.0-3.el7               @anaconda/7.2        117 k

Transaction Summary
=================================================================================================
Remove  1 Package

Installed size: 117 k
Is this ok [y/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Erasing    : snappy-1.1.0-3.el7.x86_64                                                      1/1
  Verifying  : snappy-1.1.0-3.el7.x86_64                                                      1/1

Removed:
  snappy.x86_64 0:1.1.0-3.el7

Complete!
```

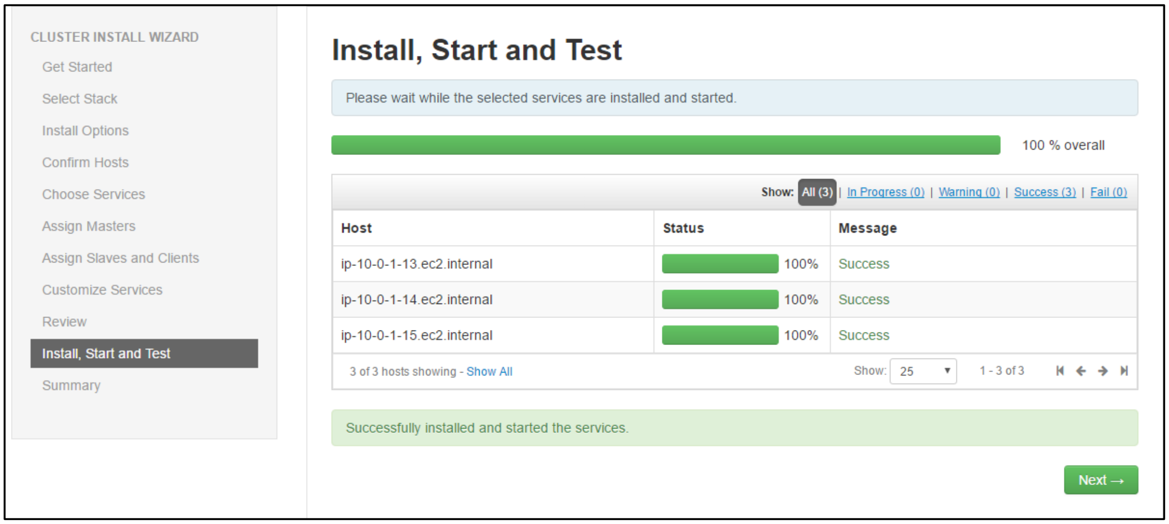
Figure 7.11.2



```
[ec2-user@ip-10-0-1-13 ~]$ sudo yum install snappy-devel
Loaded plugins: amazon-id, rhui-lb, search-disabled-repos
Resolving Dependencies
--> Running transaction check
---> Package snappy-devel.x86_64 0:1.0.5-1.el6 will be installed
--> Processing Dependency: snappy(x86-64) = 1.0.5-1.el6 for package: snappy-devel-1.0.5-1.el6.x86
_64
--> Running transaction check
---> Package snappy.x86_64 0:1.0.5-1.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=================================================================================================
 Package               Arch            Version                 Repository                   Size
=================================================================================================
Installing:
 snappy-devel          x86_64          1.0.5-1.el6             HDP-UTILS-1.1.0.20           12 k
Installing for dependencies:
 snappy                x86_64          1.0.5-1.el6             HDP-UTILS-1.1.0.20           34 k

Transaction Summary
=================================================================================================
Install  1 Package (+1 Dependent package)

Total download size: 45 k
Installed size: 112 k
Is this ok [y/d/N]: y
Downloading packages:
(1/2): snappy-devel-1.0.5-1.el6.x86_64.rpm                                |  12 kB  00:00:00
(2/2): snappy-1.0.5-1.el6.x86_64.rpm                                      |  34 kB  00:00:00
-------------------------------------------------------------------------------------------------
Total                                                            628 kB/s |  45 kB  00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : snappy-1.0.5-1.el6.x86_64                                                     1/2
  Installing : snappy-devel-1.0.5-1.el6.x86_64                                               2/2
  Verifying  : snappy-devel-1.0.5-1.el6.x86_64                                               1/2
  Verifying  : snappy-1.0.5-1.el6.x86_64                                                     2/2

Installed:
  snappy-devel.x86_64 0:1.0.5-1.el6

Dependency Installed:
  snappy.x86_64 0:1.0.5-1.el6

Complete!
[ec2-user@ip-10-0-1-13 ~]$
```

Figure 7.11.3

Follow the steps higlighted in Figure 7.11.2 and Figure 7.11.3 to resolve the “snappy-devel” issue.



Install, Start and Test

Please wait while the selected services are installed and started.

100 % overall

Show: All (3) | In Progress (0) | Warning (0) | Success (3) | Fail (0)

Host	Status	Message
ip-10-0-1-13.ec2.internal	100%	Success
ip-10-0-1-14.ec2.internal	100%	Success
ip-10-0-1-15.ec2.internal	100%	Success

3 of 3 hosts showing - Show All   Show: 25   1 - 3 of 3

Successfully installed and started the services.

Figure 7.11.4

Once the issue was resolved, I was able successfully install and start all services, as seen Figure 7.11.4.

7.12 Complete

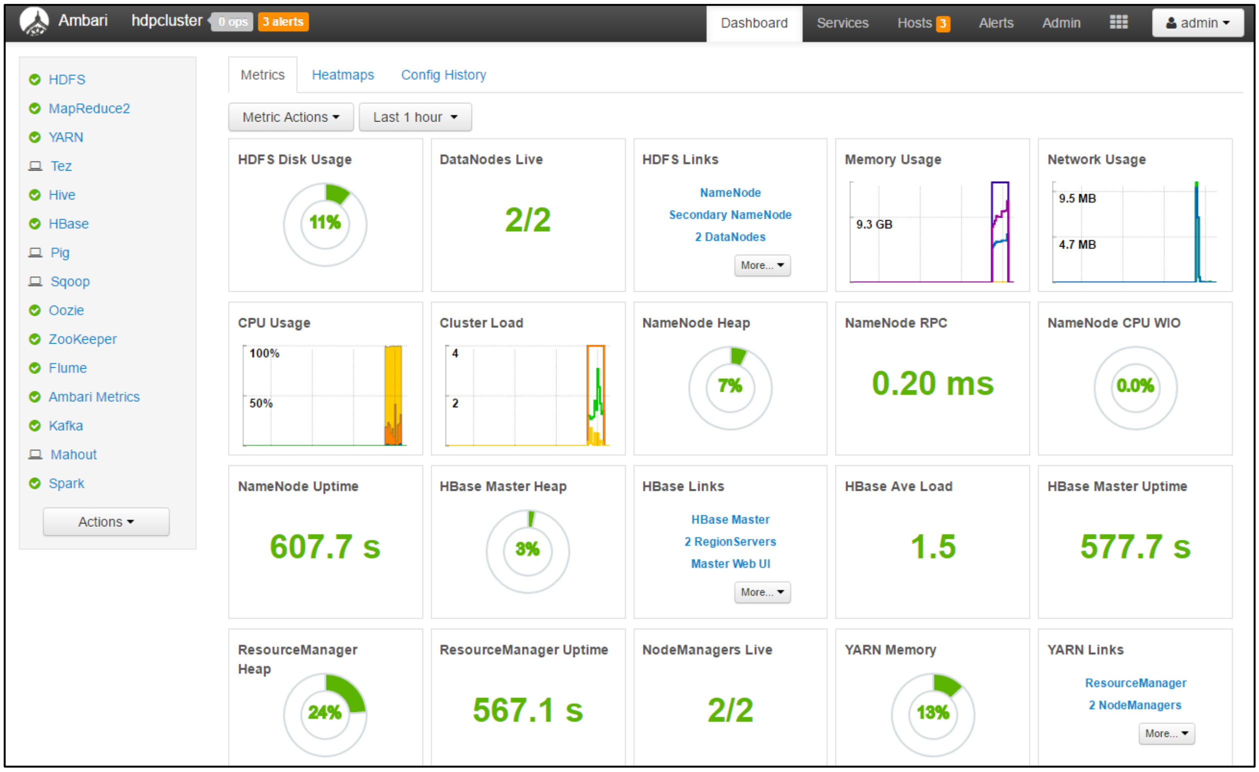
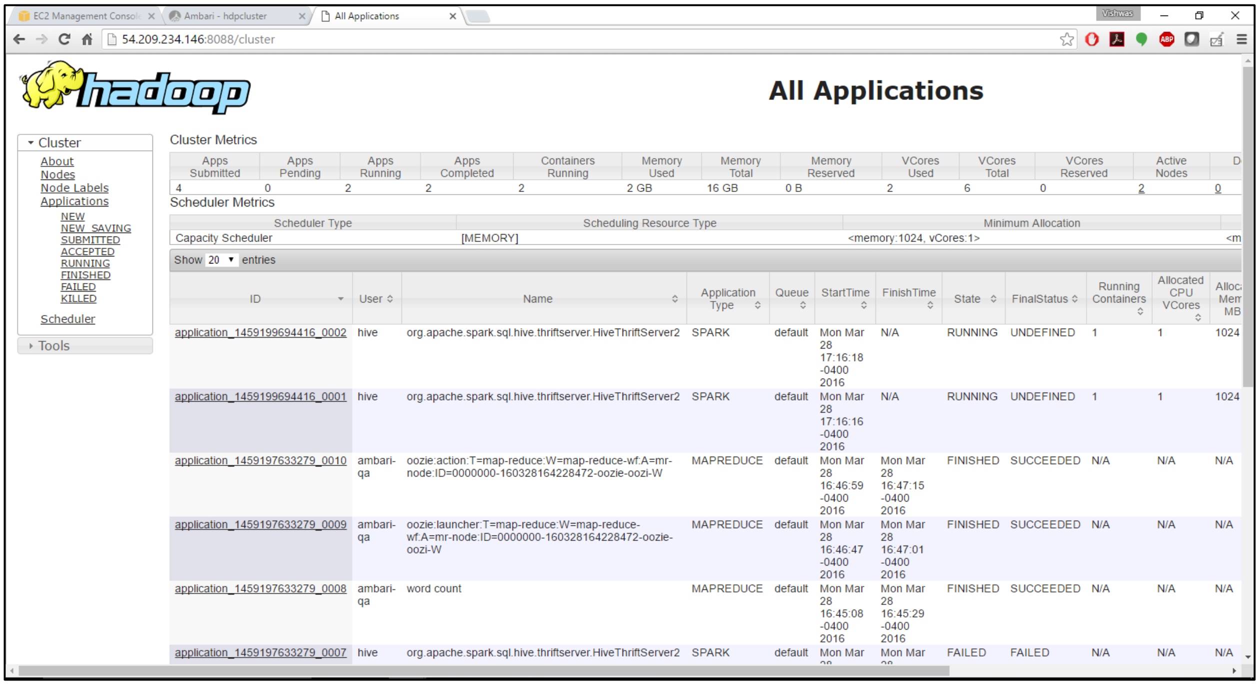


Figure 7.12.1

You Ambari dashboard will look very similar to Figure 7.12.1, once all your services are up and running.



All Applications

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	D
4	0	2	2	2	2 GB	16 GB	0 B	2	6	0	2	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	
Capacity Scheduler	[MEMORY]	<memory:1024, vCores:1>	<m

Show 20 entries

ID	User	Name	Application Type	Queue	StartTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU VCores	Alloc Mem MB
application_1459199694416_0002	hive	org.apache.spark.sql.hive.thriftserver.HiveThriftServer2	SPARK	default	Mon Mar 28 17:16:18 -0400 2016	N/A	RUNNING	UNDEFINED	1	1	1024
application_1459199694416_0001	hive	org.apache.spark.sql.hive.thriftserver.HiveThriftServer2	SPARK	default	Mon Mar 28 17:16:16 -0400 2016	N/A	RUNNING	UNDEFINED	1	1	1024
application_1459197633279_0010	ambari-qa	oozie:action:T=map-reduce:W=map-reduce-wf:A=mr-node:ID=0000000-160328164228472-oozie-oozi-W	MAPREDUCE	default	Mon Mar 28 16:46:59 -0400 2016	Mon Mar 28 16:47:15 -0400 2016	FINISHED	SUCCEEDED	N/A	N/A	N/A
application_1459197633279_0009	ambari-qa	oozie:launcher:T=map-reduce:W=map-reduce-wf:A=mr-node:ID=0000000-160328164228472-oozie-oozi-W	MAPREDUCE	default	Mon Mar 28 16:46:47 -0400 2016	Mon Mar 28 16:47:01 -0400 2016	FINISHED	SUCCEEDED	N/A	N/A	N/A
application_1459197633279_0008	ambari-qa	word count	MAPREDUCE	default	Mon Mar 28 16:45:08 -0400 2016	Mon Mar 28 16:45:29 -0400 2016	FINISHED	SUCCEEDED	N/A	N/A	N/A
application_1459197633279_0007	hive	org.apache.spark.sql.hive.thriftserver.HiveThriftServer2	SPARK	default	Mon Mar	Mon Mar	FAILED	FAILED	N/A	N/A	N/A

Figure 7.12.2

Figure 7.12.2 – Resource Manager Web UI for the HDP cluster