

LEXISNEXIS RISK SOLUTIONS

Setup of HPCC System on AWS

Timothy L Humphrey

4/27/2018

Table of Contents

1. Introduction	2
1.1 Document Conventions	2
2. Clone https://github.com/tlhumphrey/EasyFastHPCCoAWS	3
3. Make an S3 Bucket and Copy Scripts and Ssh Pem File	3
3.1 Make an S3 Bucket	3
3.2 Copy Scripts and Ssh Pem File to S3 Bucket	3
4. Using CloudFormation on AWS Console to Configure and Deploy an HPCC System	4
4.1 Navigate to CloudFormation Web Page	4
4.2 Notes About Some Parameters of MultiThorHPCCCloudFormationTemplate.json	12
4.3 Start Creation Process	12
4.4 Getting Public IP of Master(or Support)	18
Appendix A. Make a Ssh Pem File	20
Appendix B. Making IAM Super-Power-Group	24
Appendix C. Using awscli to Configure and Deploy an HPCC System on AWS	25

Setup of HPCC System on AWS

1. Introduction

This document tells how to configure and deploy an HPCC System on AWS using either of these CloudFormation templates: `MyHPCCCloudFormationTemplate.json` or `MultiThorHPCCCloudFormationTemplate.json`. The first template stands-up an HPCC System that has one THOR and/or one ROXIE. While the second template stands-up an HPCC System that (optionally) has more than one THOR and/or more than one ROXIE.

Configuring and deploying an HPCC System on AWS from linux is a three-step process when using the HPCC CloudFormation template and accompanying scripts found on github at <https://github.com/tlhumphrey/EasyFastHPCCoAWS>. The three-step process is:

1. Clone the github repository <https://github.com/tlhumphrey/EasyFastHPCCoAWS>
2. Copy the scripts in the folder, `AWSInstanceFiles` (or `MultiThorAWSInstanceFiles` if using the cloudformation template, `MultiThorHPCCCloudFormationTemplate.json`), and your ssh pem file to an S3 bucket.
3. Build a stack, using CloudFormation on the AWS console, that does the rest.

When you get to section 4, “Using CloudFormation ...”, you need:

1. An ssh key pair. Appendix A gives detailed instructions on how to make the key pair on the AWS console.
2. If you aren’t the administrator of your AWS account, then you may not have the permissions to build the stack that launches your HPCC System. So, ask your AWS administrator to follow the instructions in Appendix B, “Making IAM Super-Power-Group”, to make the Super-Power-Group and add you to it.



1.1 Document Conventions

Throughout this document we have commands that you must enter at the command line of a bash shell. These commands are shown in this document in a blue background, like the following command.

```
ls -l
```

Also, throughout this document are screenshots that show current topic web pages. I don’t always show the whole page. Often the right side of the web page has been chopped off. In addition, often you’ll see a green arrow that points to something of interest. These green arrows look like:



Also, in this document you will find helpful tips in boxes marked with this symbol:  And, warnings or cautions marked with this symbol: 

2. Clone <https://github.com/tlhumphrey/EasyFastHPCCoAWS>

You can get the contents of the EasyFastHPCCoAWS repository by cloning it using 'git'. So, first cd into the directory where you want the cloned repository. Then do the following command.

```
git clone https://github.com/tlhumphrey/EasyFastHPCCoAWS.git
```



There are directions for installing 'git' on linux here:
<https://gist.github.com/derhuerst/1b15ff4652a867391f03>

You will need an account on github.

3. Make an S3 Bucket and Copy Scripts and Ssh Pem File

Here is a summary of this process:

1. Make an S3 bucket to copy scripts (contents of AWSInstanceFiles or MultiThorAWSInstanceFiles directory) and ssh pem file (**by the way, the pem file MUST end with ".pem"**).
2. Copy scripts (contents of one of the directories mentioned above) and ssh pem file into the created S3 bucket.

You need aws cli installed on your linux machine. Installation instructions are given here
<http://docs.aws.amazon.com/cli/latest/userguide/installing.html>.

Here is the url for the awscli reference guide: <http://docs.aws.amazon.com/cli/latest/index.html>

3.1 Make an S3 Bucket

First, you need an s3 bucket to copy the scripts and ssh pem file. The following commands make an s3 bucket called BuildHPCCScripts.

```
aws s3 mb s3://BuildHPCCScripts
```

Next, cd into the folder containing all scripts (either AWSInstanceFiles or MultiThorAWSInstanceFiles if using the cloudformation template, MultiThorHPCCCloudFormationTemplate.json). And, copy your ssh pem file into this folder. **Don't forget to copy your ssh pem file into the folder.**

3.2 Copy Scripts and Ssh Pem File to S3 Bucket

The following command copies everything in the current folder into the s3 bucket, BuildHPCCScripts.

```
aws s3 cp . s3://BuildHPCCScripts --recursive
```

You can check to see if the scripts and ssh pem file are in the s3 bucket using the following command.

```
aws s3 ls s3://BuildHPCCScripts --recursive
```

4. Using CloudFormation on AWS Console to Configure and Deploy an HPCC System

This section will show you how to configure and deploy an HPCC System from the AWS console.



Instead of using the AWS console, Appendix C shows you how to launch the CloudFormation template using awscli

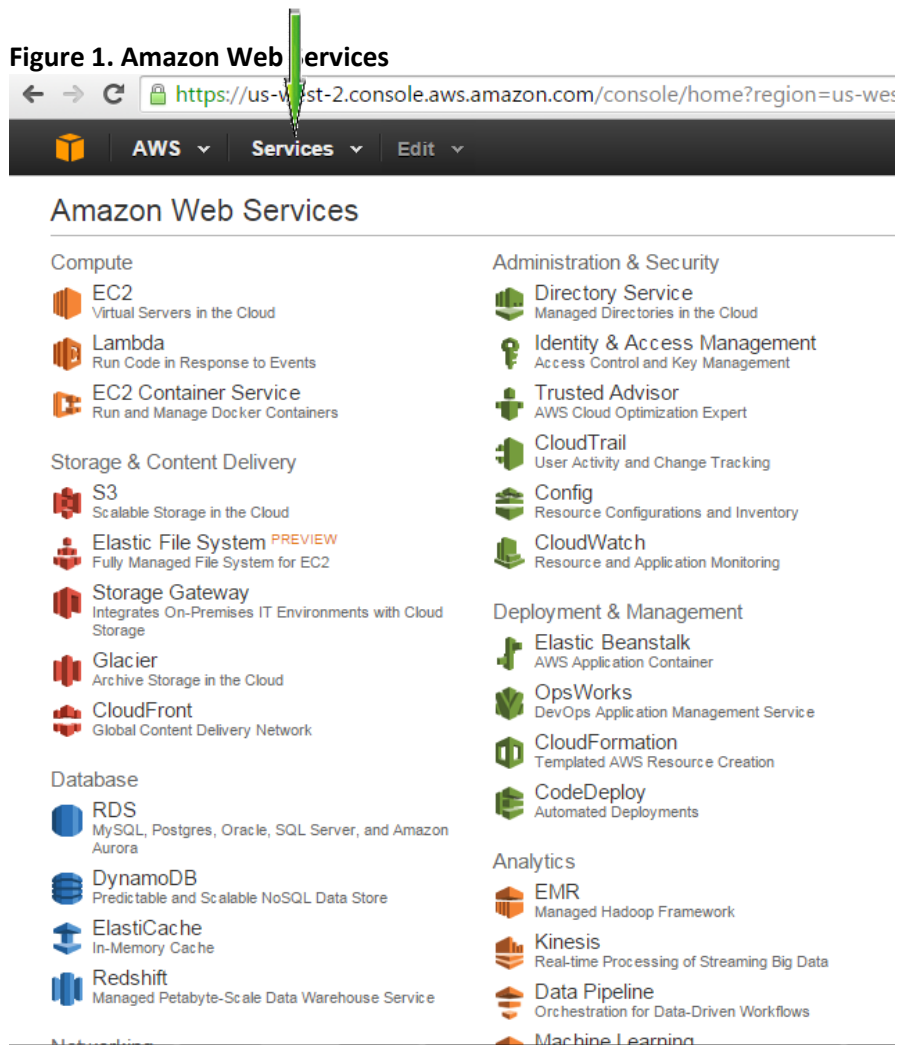
Here is a summary of this process:

1. Navigate to the CloudFormation (CF) web page on the AWS console.
2. Give the path to the HPCC CF template, i.e. MyHPCCCloudFormationTemplate.json or MultiThorHPCCCloudFormationTemplate.json.
3. Enter a Unique Name for the stack being created and fill in the HPCC CF template parameters.
4. Click the “Create” button to start the process of creating the HPCC System on AWS.

4.1 Navigate to CloudFormation Web Page

After logging into the AWS console (from your browser), you should see a web page that looks like Figure 1, below. Click on “Services”, (pointed to by the green arrow in Figure 1).

Figure 1. Amazon Web Services



On the next web page, which is a list of all AWS services, click on CloudFormation (pointed to by the green arrow of Figure 2).

Figure 2. All AWS Services

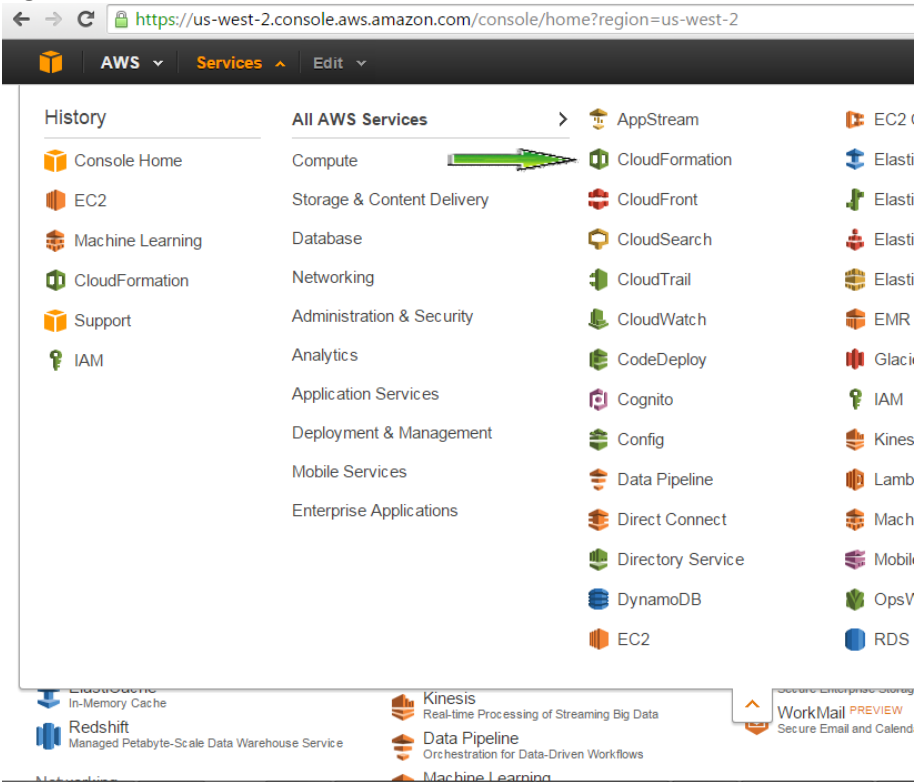
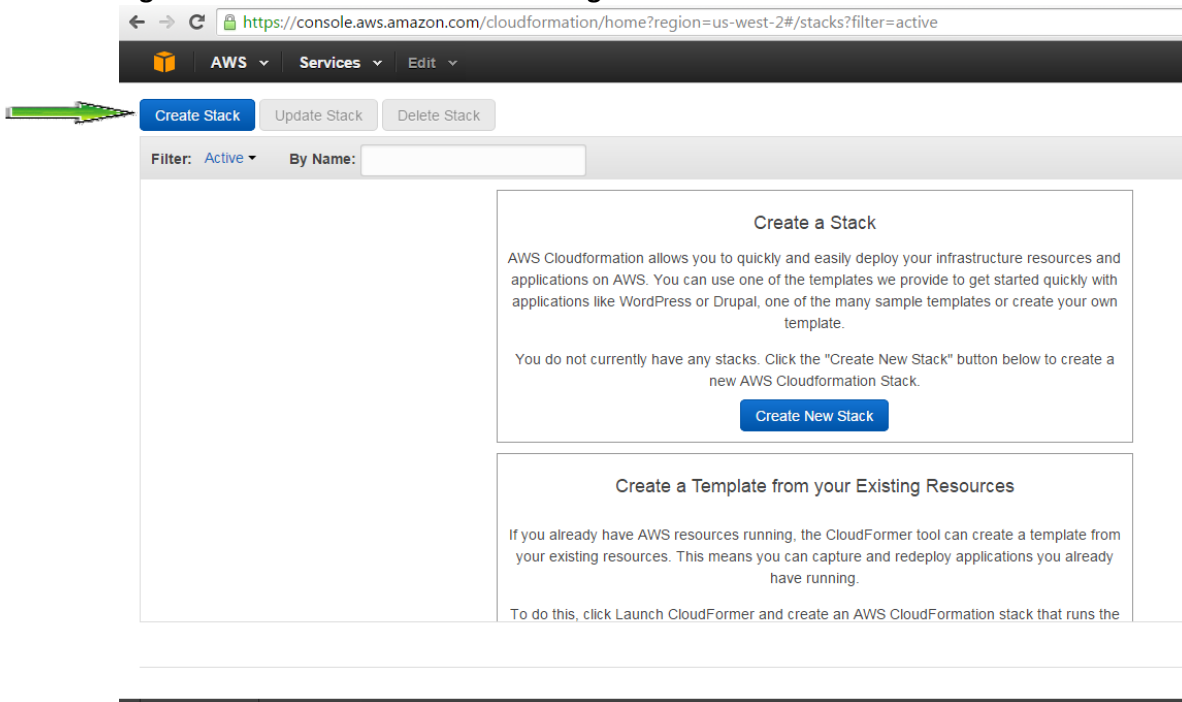
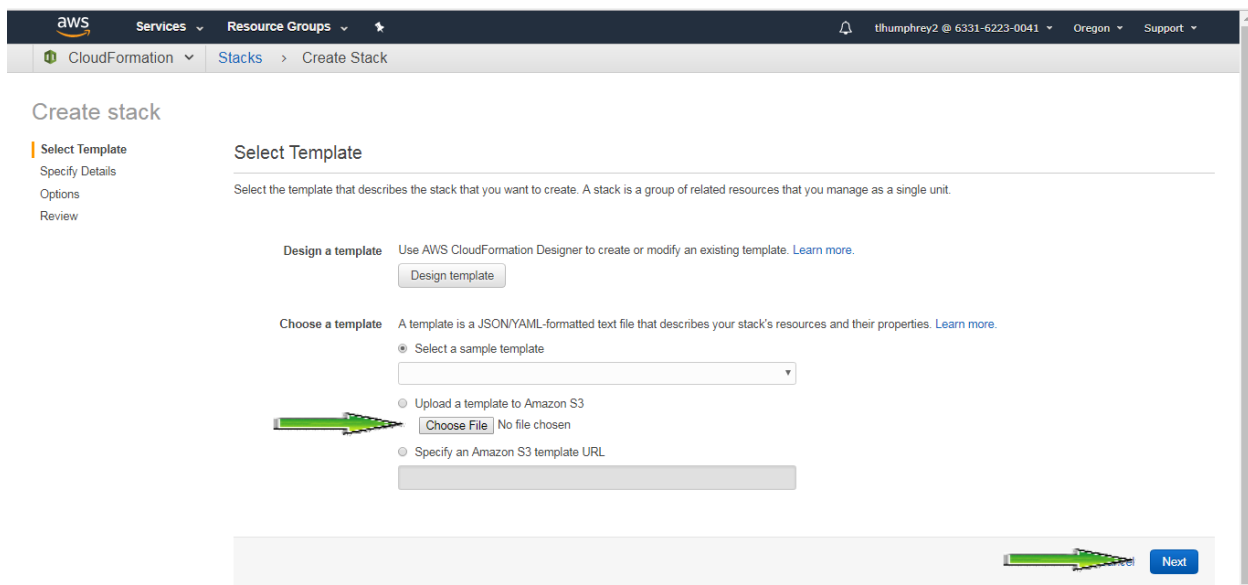


Figure 3 shows the first page of the CloudFormation section (if other cloudformation stacks have been created, they will be listed here). Click on “Create Stack” to begin the process of configuring and deploying an HPCC System to AWS. (“Create Stack” is pointed to by the green arrow of Figure 3).

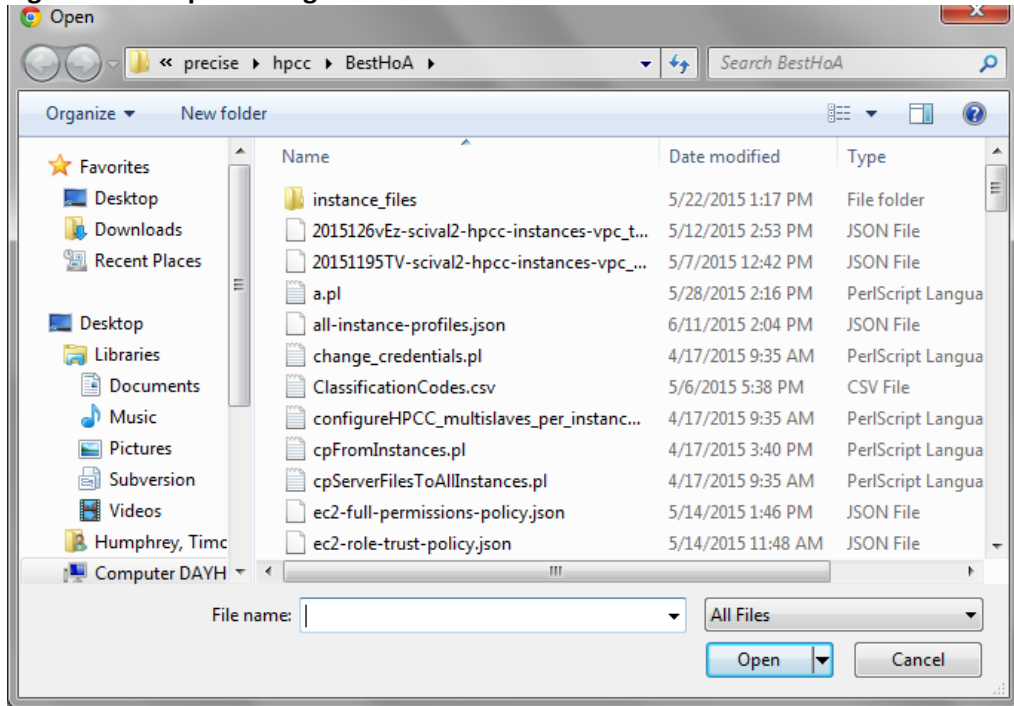
Figure 3. First CloudFormation Web Page

When you click on “Create Stack”, the next page you see is the “Select Template” page shown in Figure 4. On this page you a) select the cloudformation template that configures and deploys an HPCC System and b) click on “Next” to go to the next page.

Figure 4. Select Template Web Page

When you click on “Choose File” (pointed to by the left most green arrow of Figure 4), you get an open-file dialog box (like that of Figure 5, below). Use it to find and select the cloudformation template, i.e. MyHPCCCloudFormationTemplate.json or MultiThorHPCCCloudFormationTemplate.json. Note: “Choose File” may have a different name, like “Browse”. But, the button you click on will be under “Upload a template to Amazon s3”.

Figure 5. File Open Dialog Box



Use the open-file dialog box to find the HPCC CF template and then click on open. The name of the HPCC CF template is: MyHPCCCloudFormationTemplate.json or MultiThorHPCCCloudFormationTemplate.json. After selecting the template click on “Next” (pointed to by the right most green arrow in Figure 4). The next page you see should look like Figure 6a (or Figure 6b for multi-thor CF template), Specify Details, shown below.

Figure 6a. Specify Details, for Template, MyHPCCCloudFormationTemplate.json

The screenshot shows the 'Specify Details' page for the template `MyHPCCCloudFormationTemplate.json`. The page is titled 'Specify Details' and includes a sub-header 'Specify a stack name and parameter values. You can use or change the default parameter values, which are defined in the AWS CloudFormation template. [Learn more.](#)

The 'Stack name' field is highlighted with a green arrow. The 'Parameters' section includes the following fields:

- HPCCPlatform:** HPCC-Platform-6.2.22-1 (The version of the HPCC Platform to install.)
- InstallCassandra:** NO (Optional) YES or NO to Install Cassandra
- KeyPair:** Search (The EC2 Key Pair to allow SSH access to the instances.)
- MasterInstanceType:** t2.xlarge (HPCC Thor Master EC2 Instance type)
- NumberOfRoxieNodes:** 0 (Number of Roxie nodes in deployed HPCC System)
- NumberOfSlaveInstances:** 6 (Number of Slave instances to be launched)
- NumberOfSlavesPerNode:** 12 (Number of THOR slave nodes per slave instance)
- RoxieInstanceType:** t2.xlarge (HPCC Roxie EC2 Instance type)
- ScriptsS3BucketFolder:** (S3 bucket folder that contains the scripts that are executed in the UserData section.)
- SlaveInstanceType:** t2.xlarge (HPCC Thor Slave EC2 Instance type)
- UserNameAndPassword:** (Optional) Enter like: username:password Used to log into ECL Watch and ECL IDE.

The 'Next' button is highlighted with a green arrow.

Figure 6b. Specify Details for Template, MultiThorHPCCCloudFormationTemplate.json

The screenshot shows the 'Specify Details' page for the template `MultiThorHPCCCloudFormationTemplate.json`. The page is titled 'Specify Details' and includes a sub-header 'Specify a stack name and parameter values. You can use or change the default parameter values, which are defined in the AWS CloudFormation template. [Learn more.](#)

The 'Stack name' field is highlighted with a green arrow. The 'Parameters' section includes the following fields:

- HPCCPlatform:** HPCC-Platform-6.4.10-1 (The version of the HPCC Platform to install.)
- Slave Servers/THOR:** 3,2 (Comma separated list of slave instances for each THOR.)
- Roxie Servers/ROXIE:** 2 (Comma separated list of roxie instances for each ROXIE.)
- SlaveNodes/Server/THOR:** 1 (Comma separated list of slave nodes per instance for each THOR.)
- Support Instance Number:** 0 (Number of support instances. These would be used for support components.)
- ClusterInventoryFile:** NONE (CURRENTLY NO SUPPORTED)
- UserNameAndPassword:** NONE (Optional) Enter like: username:password Used to log into ECL Watch and ECL IDE.
- KeyPair:** Search (The EC2 Key Pair to allow SSH access to the instances.)
- InstallCassandra:** NO (Optional) YES or NO to Install Cassandra
- ScriptsS3BucketFolder:** (S3 bucket folder that contains the scripts that are executed in the UserData section.)
- SSHUserName:** ec2-user (The user name used to ssh into an instance)
- InstanceType:** r3.xlarge (HPCC Thor Master EC2 Instance type)

The 'Next' button is highlighted with a green arrow.

On the Specify Details page, you will enter a "Stack name" and enter or modify the other parameters on the page.

The "Stack name" textbox is pointed to by the left most green arrow. **You MUST specify a unique name for the "Stack name" (a name that has never been used).** Guidance for creating the "Stack name" can be found here: <http://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/cfn-using-console->

create-stack-parameters.html.

You will notice that some of the parameters that configure the HPCC System already have values and some do not. Plus, some have dropdown menus and some do not.

Tables 1a and 1b show parameters that already have values for MyHPCCCloudFormationTemplate.json and MultiThorHPCCCloudFormationTemplate.json, respectively.

Table 1a. MyHPCCCloudFormationTemplate.json Parameters that already of Values

MyHPCCCloudFormationTemplate.json Parameters that already of Values		
Parameter Name	Description	Values
SSHUserName	User name used for ssh	ec2-user
HPCCPlatform	HPCC System Platform to install	HPCC-Platform-6.2.22-1
InstallCassandra	YES or NO do you want Cassandra installed	NO
[Master, Roxie, Slave]InstanceTypes	Instance type for Master, Roxies or Slaves	All 3 are i2.8xlarge
NumberOf[Roxie,Slave]Instances	Number of Roxie or Slave Instances	0, 12 respectively
NumberOfSlavesPerInstance	Number of slave nodes on each instance	6

Table 1b. MultiThorHPCCCloudFormationTemplate.json Parameters that already of Values

Parameter Name	Description	Values
SSHUserName	User name used for ssh	ec2-user
HPCCPlatform	HPCC System Platform to install	HPCC-Platform-6.4.10-1
InstallCassandra	YES or NO do you want Cassandra installed	NO
InstanceType	Instance type for Master, Roxies Slaves, and Support	r3.xlarge
Slave Servers/THOR	Number of slave server for each THOR	3,2
Roxie Servers/ROXIE	Number of Roxie servers for each ROXIE	2
SlaveNodes/Server/THOR	Number of slave nodes on each server for each THOR	1
Support Instance Number	Number of support instances	0
ClusterInventoryFile	Name of Inventory File	NONE
UserNameAndPassword	Username/password to access cluster	NONE

Tables 2a and 2b show parameters that do not have values for MyHPCCCloudFormationTemplate.json and MultiThorHPCCCloudFormationTemplate.json, respectively.

Table 2a. MyHPCCCloudFormationTemplate.json Parameters that Need Values

Parameter Name	Description
KeyPair	Private key used for ssh
ScriptsS3BucketFolder	S3 Bucket where instance scripts exist
UserNameAndPassword	User name and password for ECL Watch or IDE.

Table 2b. MultiThorHPCCCloudFormationTemplate.json Parameters that Need Values

Parameter Name	Description
KeyPair	Private key used for ssh
ScriptsS3BucketFolder	S3 Bucket where instance scripts exist

Tables 3a and 3b show parameters that have dropdown menus for MyHPCCCloudFormationTemplate.json and MultiThorHPCCCloudFormationTemplate.json, respectively. So you can select the parameter values instead of typing them in.

Table 3a. MyHPCCCloudFormationTemplate.json Parameters with Dropdown Menus

Parameter Name	Description
SSHUserName	User name used for ssh
HPCCPlatform	HPCC System Platform to install
InstallCassandra	YES or NO do you want Cassandra installed
[Master,RoxieSlave] InstanceTypes	Instance type for Master, Roxies or Slaves
KeyPair	Private key used for ssh

Table 3b. MultiThorHPCCCloudFormationTemplate.json Parameters with Dropdown Menus

Parameter Name	Description
SSHUserName	User name used for ssh
HPCCPlatform	HPCC System Platform to install
InstallCassandra	YES or NO do you want Cassandra installed
InstanceType	Instance type for all instances
KeyPair	Private key used for ssh

Tables 4a and 4b show parameters that you type in their values for MyHPCCCloudFormationTemplate.json and MultiThorHPCCCloudFormationTemplate.json, respectively.

Table 4a. MyHPCCCloudFormationTemplate.json Parameters you type in their values

Parameter Name	Description
Number of [Roxie,Slave] Instances	Number of Roxie or Slave Instances
NumberOfSlavesPerInstance	Number of slave nodes on each instance
ScriptsS3BucketFolder	S3 Bucket where instance scripts exist
UserNameAndPassword	User name and password for ECL Watch or IDE.

Table 4b. MultiThorHPCCCloudFormationTemplate.json Parameters you type in their values

Parameter Name	Description
Slave Servers/THOR	Number of slave server for each THOR
Roxie Servers/ROXIE	Number of Roxie servers for each ROXIE
SlaveNodes/Server/THOR	Number of slave nodes on each server for each THOR
Support Instance Number	Number of support instances
ScriptsS3BucketFolder	S3 Bucket where instance scripts exist
UserNameAndPassword	User name and password for ECL Watch or IDE.

One parameter that must be typed in is ScriptsS3BucketFolder. If your s3 bucket is the same as the one we created in Section 3, the value for ScriptsS3BucketFolder will be:

s3://BuildHPCCScripts

To reduce the cost during the testing of the CloudFormation template, a) I changed the instance types for the master, roxie, and slaves to something that didn't cost much per hour, e.g. the c3.large instance type currently cost only \$0.21 per hour; and b) I changed the number of roxie and slave instances launched.

4.2 Notes About Some Parameters of MultiThorHPCCCloudFormationTemplate.json

The 3 parameters: Slave Servers/THOR, Roxie Servers/ROXIE and SlaveNodes/Server/THOR are comma separated lists of numbers. Each number in the comma separated list tells the number of servers for one of the Slaves or Roxies. Or for the parameter, SlaveNodes/Server/THOR, each number in its comma separated list tells the number of slave nodes on each of that THOR's servers.


For example, if you want 3 THORs one with 2 servers, the 2nd with 3 servers and the 3rd with 1 server then Slave Servers/THOR should have the following value:

2,3,1

4.3 Start Creation Process

Once you have the parameters the way you want them, you click on "Next" (where the right most green arrow points in Figure 6a or Figure 6b, above). The next web page looks like Figure 7, below.

Figure 7. Stack Options Web Page

AWS

Services

Edit

Timothy HumphreyOregonSupport

Select Template

Specify Parameters

Options

Review

Options

Tags

You can specify tags (key-value pairs) for resources in your stack. You can add up to 10 unique key-value pairs for each stack. [Learn more.](#)

	Key (127 characters maximum)	Value (255 characters maximum)	
1	<input type="text"/>	<input type="text"/>	<div>+</div>

Advanced

You can set additional options for your stack, like notification options and a stack policy. [Learn more.](#)

Cancel

Previous

Next

There is nothing to do on this web page. So, just click the “Next” button to continue to the next web page which looks like Figure 8, below (Note. Figure 8 is what you see if the Cloudformation template is MyHPCCCloudFormationTemplate.json).

13

Figure8. Review Parameters Web Page

The screenshot shows the 'Review Parameters' web page in the AWS CloudFormation console. The page has a dark header with the AWS logo and navigation tabs for 'AWS', 'Services', and 'Edit'. On the left, there is a sidebar with links: 'Select Template', 'Specify Parameters', 'Options', and 'Review' (which is highlighted with an orange bar). The main content area is titled 'Review' and contains two sections: 'Template' and 'Parameters'.

Template

Name	FastHPCC-150615-1
Template URL	https://s3-us-west-2.amazonaws.com/cf-templates-19ljwti8ng5rf-us-west-2/2015
Description	Need: Add descriptions of the various resources.
Estimate cost	Cost

Parameters

HPCCPlacementGroup	pg-tilh-best
KeyPair	tilh_keys_us_west_2
MasterInstanceType	c3.2xlarge
NumberOfRoxieNodes	0
NumberOfSlaveInstances	1
NumberOfSlavesPerNode	2
RoxieInstanceType	c3.2xlarge
ScriptsS3BucketFolder	s3://BuildHPCCScripts
SlaveInstanceType	c3.2xlarge
Create IAM resources	False

This page shows all your settings, so you can verify they are what you want before starting the stack building process.

The next screenshot, Figure 9, shows the same web page as above, except I've scrolled down so you can see the "Create" button. You click on the "Create" button to start the stack building process.

But, before you do, you must click on the acknowledge checkbox (pointed to by the green arrow of Figure 9, below) to indicate that you know that the template comes from a trusted source. Why? Because the template creates AWS resources that you have to pay for and also the template gives permission to the instances created to access AWS resources such as S3 buckets.

Figure9. Bottom of Review Parameters Web Page

Options

Tags

No tags provided

Advanced

Notification Timeout

Rollback on failure

none
Yes

Capabilities

i

The following resource(s) require capabilities: [AWS::IAM::InstanceProfile, AWS::IAM::Role]

This template might include Identity and Access Management (IAM) resources, which can include groups, IAM users, and IAM roles with certain permissions. Ensure that the template you are using is from a trusted source. [Learn more.](#)

I acknowledge that this template might cause AWS CloudFormation to create IAM resources.

Cancel

Previous

Create

The next web page you see should look like Figure 10, below.

It is divided into 2 parts. The top part shows the stacks (in this case only one). And what the bottom part shows depends on which tab you have selected, i.e. Overview or Outputs or Resources or Events or Template or Parameters or Tags or Stack Policy or Change Set. In Figures 10 through 12, I have selected the Events tab because it tells you the progress of the “stack build” process.


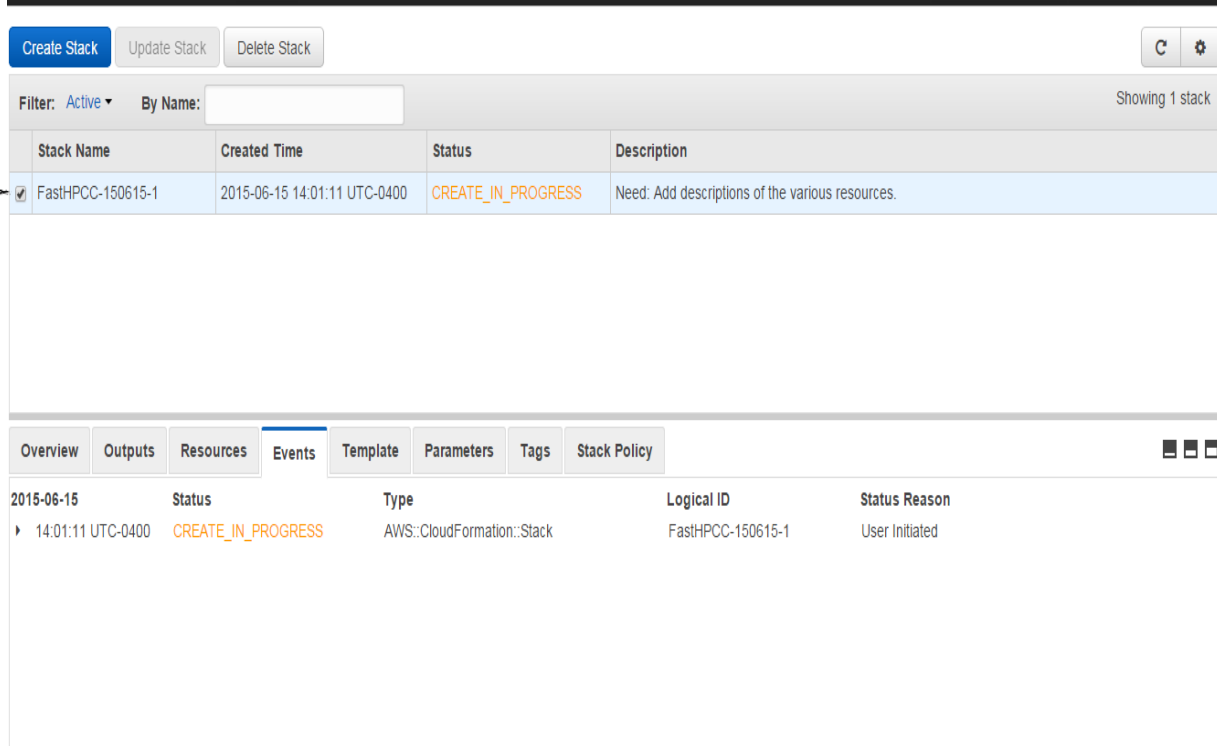
	<p>The security group (firewall) created for the launched HPCC System has port 8010 and port 22 open to the world. You may want to change these to specific IPs or IP ranges. Click on this link, http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/authorizing-access-to-an-instance.html, for information about how to change inbound traffic to these ports.</p>
---	--

Figure 10. Show Stacks Web Page



The screenshot shows the AWS CloudFormation 'Show Stacks' web page. At the top, there are buttons for 'Create Stack', 'Update Stack', and 'Delete Stack'. Below these are filters: 'Filter: Active' and 'By Name:'. The main table lists stacks, with one stack selected: 'FastHPCC-150615-1'. The stack's status is 'CREATE_IN_PROGRESS'. Below the table, there are tabs for 'Overview', 'Outputs', 'Resources', 'Events', 'Template', 'Parameters', 'Tags', and 'Stack Policy'. The 'Events' tab is selected, showing a list of events for the stack, including '14:01:11 UTC-0400' with status 'CREATE_IN_PROGRESS' and reason 'User Initiated'.

Stack Name	Created Time	Status	Description
FastHPCC-150615-1	2015-06-15 14:01:11 UTC-0400	CREATE_IN_PROGRESS	Need: Add descriptions of the various resources.

2015-06-15	Status	Type	Logical ID	Status Reason
14:01:11 UTC-0400	CREATE_IN_PROGRESS	AWS::CloudFormation::Stack	FastHPCC-150615-1	User Initiated

If you refresh the web page, you will see an update of the progress of the “stack build” process (like Figure 11, below).

Figure 11. Updated Show Stacks Web Page

The screenshot shows the same AWS CloudFormation 'Show Stacks' web page, but with the 'Events' tab selected. The stack 'FastHPCC-150615-1' is still in 'CREATE_IN_PROGRESS' status. The 'Events' tab shows a list of events, including '14:04:54 UTC-0400' with status 'CREATE_COMPLETE' and '14:03:35 UTC-0400' with status 'CREATE_IN_PROGRESS'. The 'Status Reason' column shows 'Resource creation Initiated' for several events.

2015-06-15	Status	Type	Logical ID	Status Reason
14:04:54 UTC-0400	CREATE_COMPLETE	AWS::AutoScaling::AutoScalingGroup	SlaveASG	
14:03:37 UTC-0400	CREATE_COMPLETE	AWS::AutoScaling::AutoScalingGroup	RoxieASG	
14:03:36 UTC-0400	CREATE_IN_PROGRESS	AWS::AutoScaling::AutoScalingGroup	MasterASG	Resource creation Initiated
14:03:36 UTC-0400	CREATE_IN_PROGRESS	AWS::AutoScaling::AutoScalingGroup	RoxieASG	Resource creation Initiated
14:03:36 UTC-0400	CREATE_IN_PROGRESS	AWS::AutoScaling::AutoScalingGroup	SlaveASG	Resource creation Initiated
14:03:35 UTC-0400	CREATE_IN_PROGRESS	AWS::AutoScaling::AutoScalingGroup	MasterASG	
14:03:35 UTC-0400	CREATE_IN_PROGRESS	AWS::AutoScaling::AutoScalingGroup	RoxieASG	
14:03:35 UTC-0400	CREATE_IN_PROGRESS	AWS::AutoScaling::AutoScalingGroup	SlaveASG	



During the stack build process, you may get an error message that contains the words “not authorized to perform:iam:CreateRole”, which means you don’t have the proper permissions to create the stack. So, ask your AWS administrator to follow the instructions in Appendix B, “Making IAM Super-Power-Group”, to make the Super-Power-Group and add you to it.

The stack build process is done (meaning the HPCC System is setup) when “CREATE_COMPLETE” is the Status of the stack building process (where the green arrow points in Figure 12, below).

Figure 12. Finished Stack Creation

The screenshot shows the AWS CloudFormation console. At the top, there are buttons for 'Create Stack', 'Update Stack', and 'Delete Stack'. Below these are filters for 'Active' and 'By Name'. A table lists the stacks, with one stack 'FastHPCC-150615-1' shown. The 'Status' column for this stack is 'CREATE_COMPLETE', which is highlighted by a green arrow. Below the stack list, the 'Events' tab is selected, showing a list of events for the stack. The events include 'CREATE_COMPLETE' and 'CREATE_IN_PROGRESS' for various resources like MasterASG, SlaveASG, and RoxieASG.

Stack Name	Created Time	Status	Description
FastHPCC-150615-1	2015-06-15 14:01:11 UTC-0400	CREATE_COMPLETE	Need: Add descriptions of the various resources.

Overview	Outputs	Resources	Events	Template	Parameters	Tags	Stack Policy
2015-06-15							
		Status	Type	Logical ID		Status Reason	
▶ 14:09:52 UTC-0400		CREATE_COMPLETE	AWS::CloudFormation::Stack	FastHPCC-150615-1			
▶ 14:09:48 UTC-0400		CREATE_COMPLETE	AWS::AutoScaling::AutoScalingGroup	MasterASG			
▶ 14:09:46 UTC-0400		CREATE_IN_PROGRESS	AWS::AutoScaling::AutoScalingGroup	MasterASG		Received SUCCESS signal with Uniqueld i-da9c6412	
▶ 14:04:54 UTC-0400		CREATE_COMPLETE	AWS::AutoScaling::AutoScalingGroup	SlaveASG			
▶ 14:03:37 UTC-0400		CREATE_COMPLETE	AWS::AutoScaling::AutoScalingGroup	RoxieASG			
▶ 14:03:36 UTC-0400		CREATE_IN_PROGRESS	AWS::AutoScaling::AutoScalingGroup	MasterASG		Resource creation Initiated	
▶ 14:03:36 UTC-0400		CREATE_IN_PROGRESS	AWS::AutoScaling::AutoScalingGroup	RoxieASG		Resource creation Initiated	
▶ 14:03:36 UTC-0400		CREATE_IN_PROGRESS	AWS::AutoScaling::AutoScalingGroup	SlaveASG		Resource creation Initiated	

When the stack build process has completed, go to the “Instances” page on the AWS console to get the IP address of the Master, which is needed to access ECL Watch and to setup the ECL IDE. If you are using template, MultiThorHPCCCloudFormationTemplate.json, and do not have a master instance then you need the public IP of the Support instance.

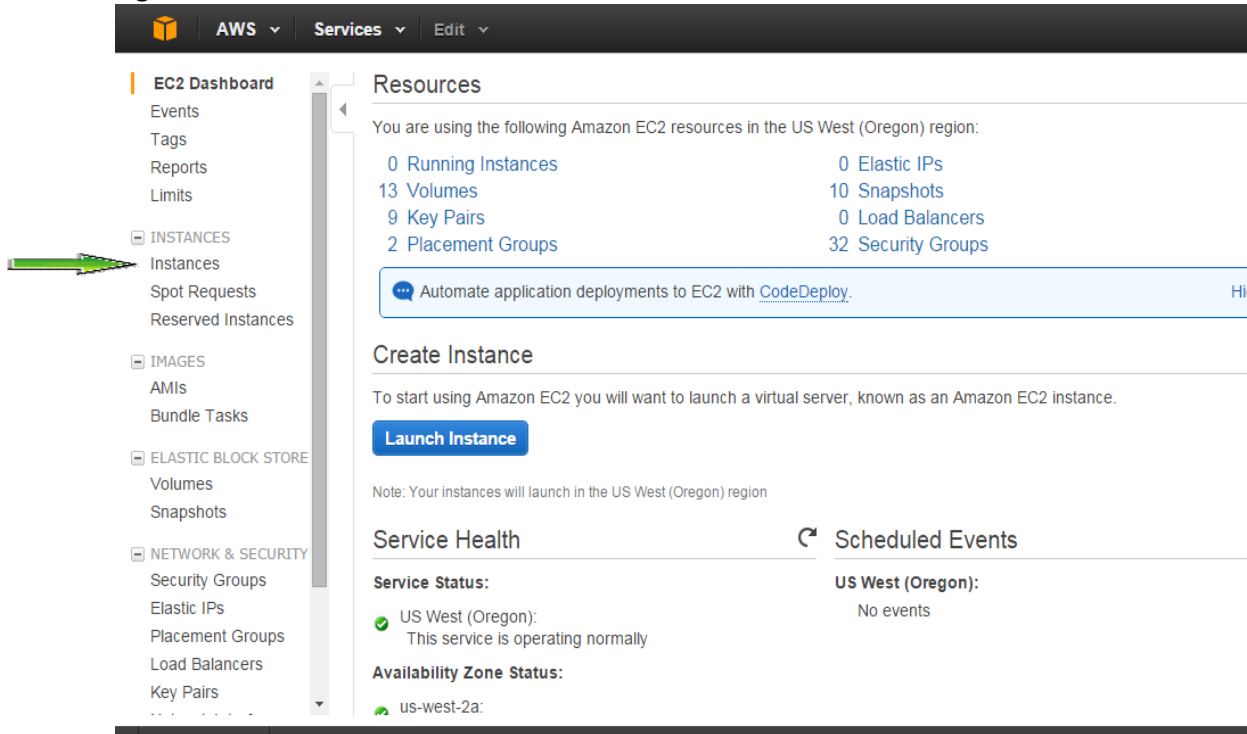


Once you have finished with the deployed HPCC System, delete the stack so all resources are terminated and AWS charges won’t continue.

4.4 Getting Public IP of Master(or Support)

From the EC2 Dashboard, click on “Instances” (pointed to by the green arrow of Figure 13, below).

Figure 13. EC2 Dashboard with Instances Marked



Next, you will see the Instance page which should look like Figure 14, below. The Master instance has been selected by clicking on the button just to the left of the instance’s name (pointed to by the top-left green arrow in Figure 14). And because this instance is selected, its Public IP address is shown where the bottom-right green arrow points. You can swipe across this Public IP address and save it. Then, paste it into your browser’s address bar and append “:8010” to access ECL Watch.

Figure 14. Instances Page with Master Marked

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Spot Requests

Reserved Instances

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Volumes

Snapshots

NETWORK & SECURITY

Security Groups

Elastic IPs

Placement Groups

Load Balancers

Launch Instance

Connect

Actions

Filter by tags and attributes or search by keyword

	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status
<input checked="" type="checkbox"/>	FastHPCC-150615-1--Master	i-41cf3589	c3.2xlarge	us-west-2c	running	
<input type="checkbox"/>	FastHPCC-150615-1--Slave	i-e2cf352a	c3.2xlarge	us-west-2c	running	

Instance: **i-41cf3589 (FastHPCC-150615-1--Master)** Public IP: 52.26.215.137

Description

Status Checks

Monitoring

Tags

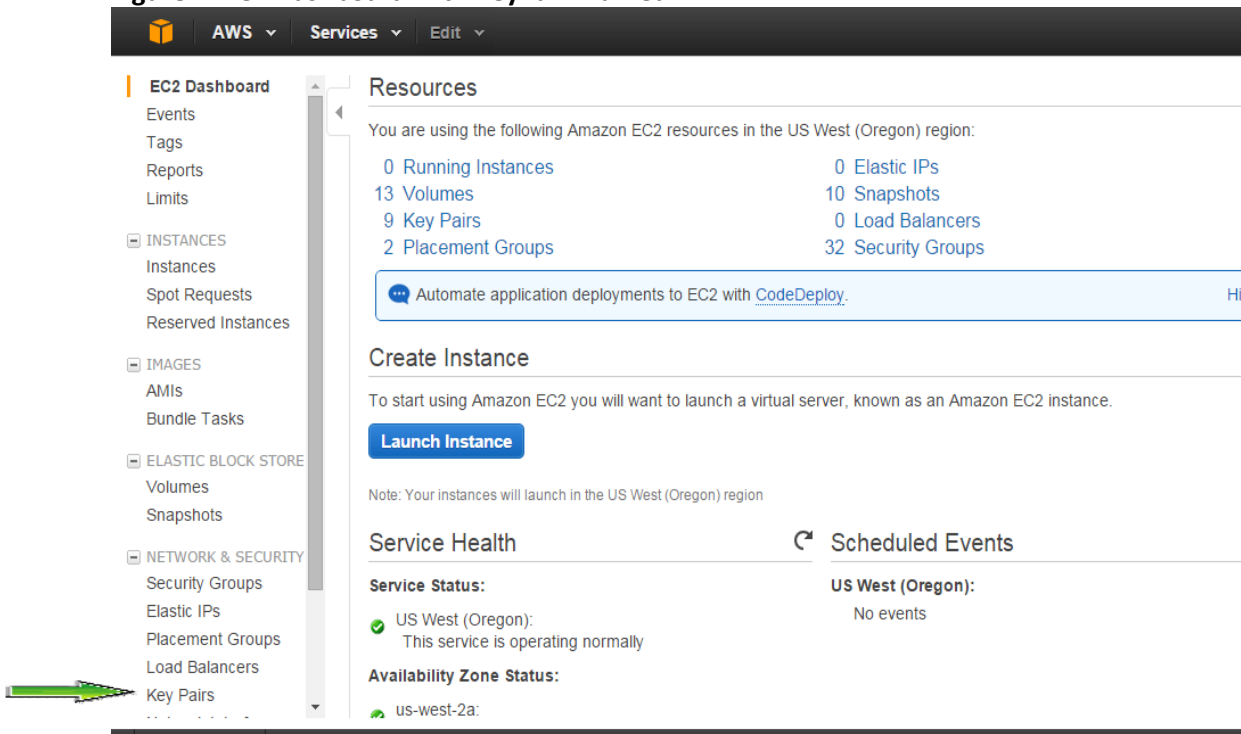
Instance ID	i-41cf3589	Public DNS	-
Instance state	running	Public IP	52.26.215.137

Appendix A. Make a Ssh Pem File

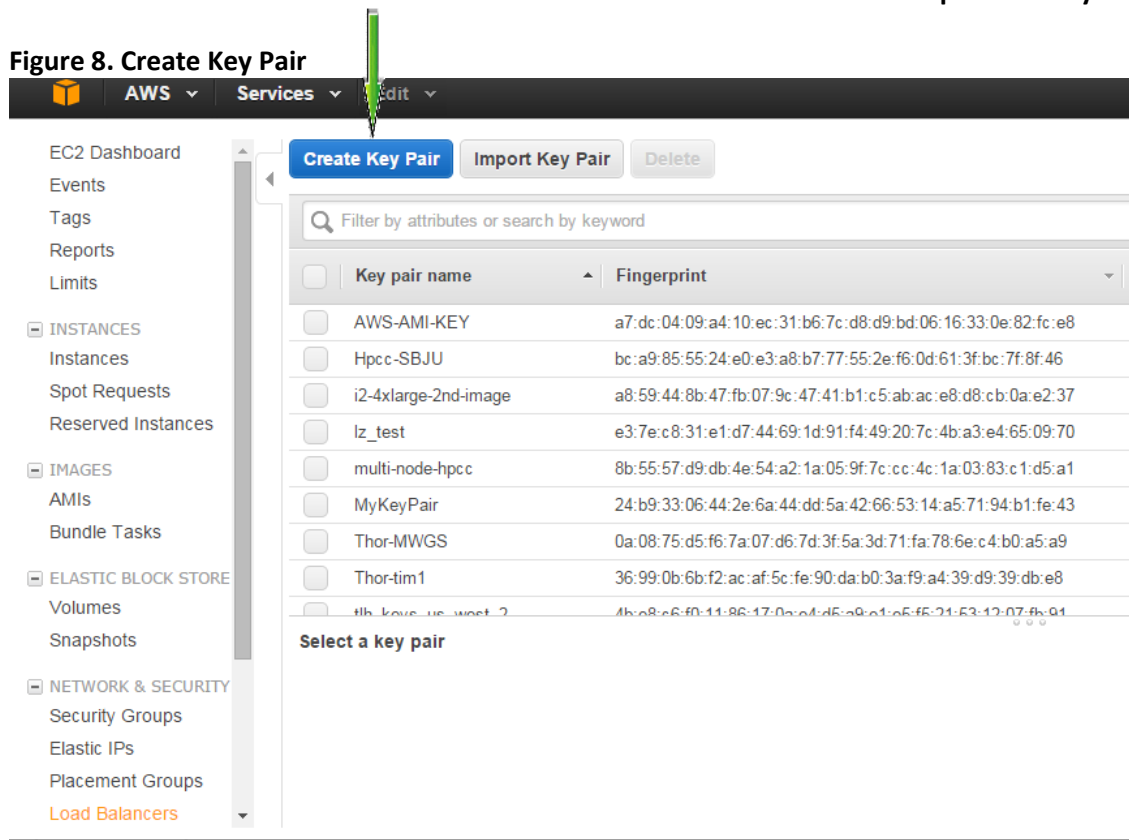
An ssh key pair is used by the nodes of your deployed HPCC System to access other nodes of the system. The following gives details on how to create an ssh key pair and have its pem file downloaded to your linux computer.

First, go to the EC2 Dashboard and select the region you will be using the key pair (select from dropdown menu on the right side of the dark ribbon at the top of the web page). And second, click on Key Pair (pointed to by the green arrow of Figure 25).

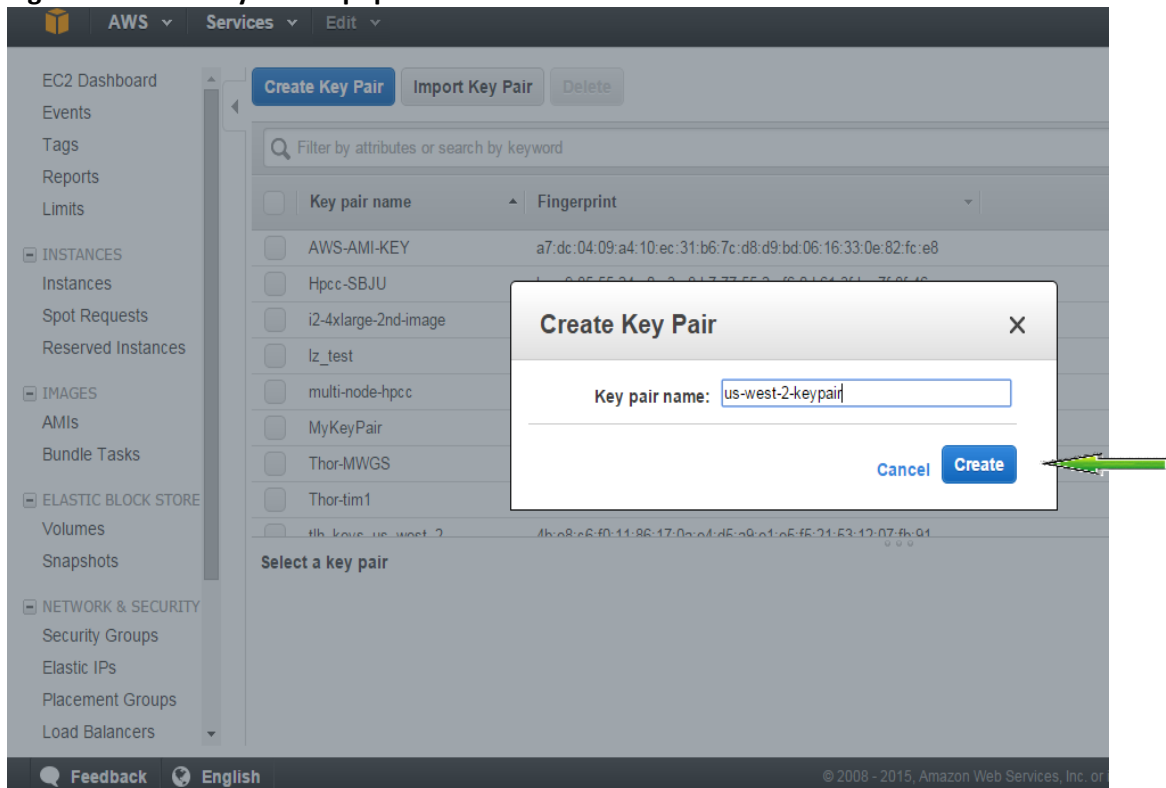
Figure 7. EC2 Dashboard with KeyPair Marked



Next, you will see the page where you can create a new key pair. It should look like Figure 26, below. Click on “Create Key Pair” (pointed to by the green arrow of Figure 26).

Figure 8. Create Key Pair

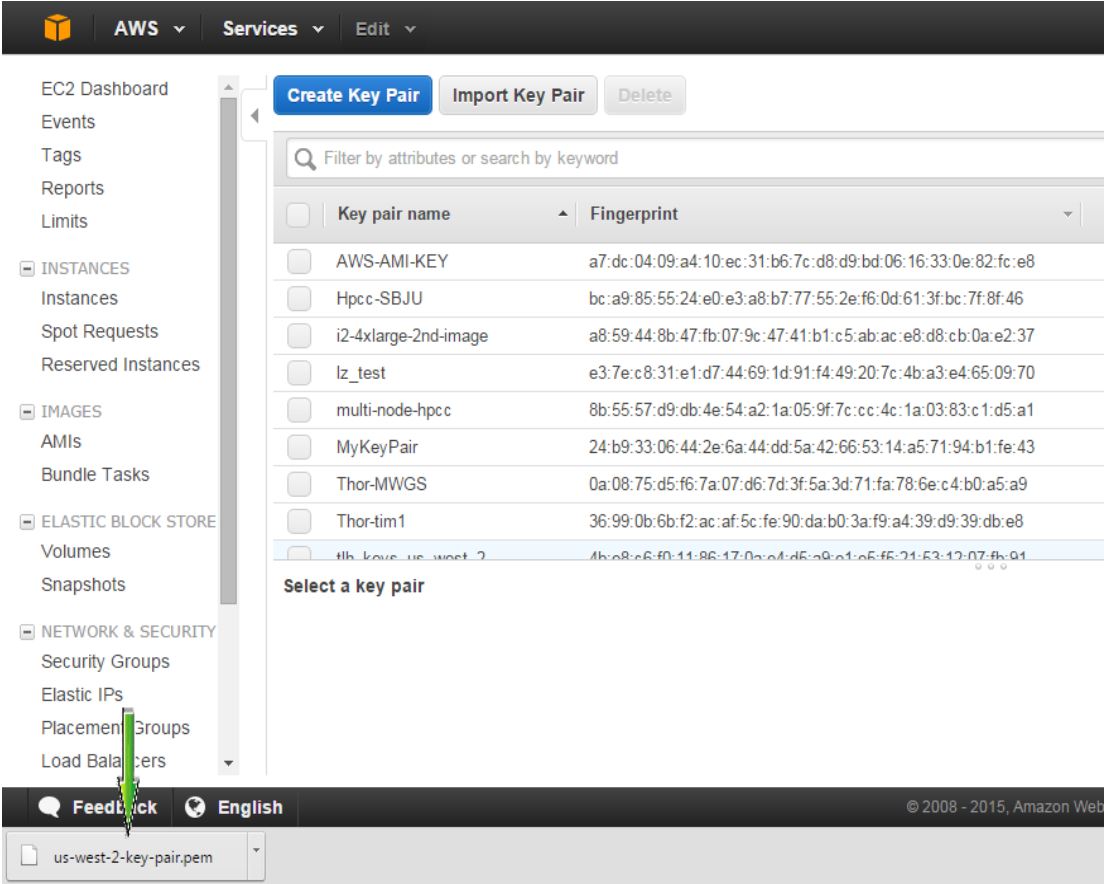
Next, you will see a popup with the title “Create Key Pair”. It should look like Figure 27. Enter the name of your new key pair and then click on “Create” (pointed to by the green arrow of Figure 27). You will notice that in Figure 27, I entered the name “us-west-2-keypair”.

Figure 9. Create Key Pair Popup

After clicking on the “Create” button, the next thing you will see is a page that looks like Figure 28, below. And, you should get an indication that the pem file for the newly created key pair was downloaded to your linux computer (pointed to by the green arrow of Figure 28, below).

Copy this file into the folder, InstanceFilesForConfigureAndStartHPCC, of the cloned repository (see Section 2 for the location of this folder).

Figure 10. Create Key Pair with Pem File Downloaded



Appendix B. Making IAM Super-Power-Group

This must be done by the administrator of your AWS account. And, it only needs to be done if you get an error during stack creation saying you are “not authorized to perform:iam:CreateRole”.

First the administrator of your AWS account will create a new IAM group which I call Super-Power-Group using the policy in the file super-user-iam-policy.json of the cloned repository. Then, the administrator will place you in that group.

The following AWS cli command creates the Super-Power-Group (This can also be done on the AWS console using the instructions at

http://docs.aws.amazon.com/IAM/latest/UserGuide/Using_CreatingAndListingGroups.html).

```
aws iam put-group-policy --group-name Super-Power-Group --policy-document file://super-user-iam-policy.json --policy-name super-power-policy
```

To add a user to the Super-Power-Group:

1. Sign into the AWS console and open the IAM console, i.e. select IAM from Services.
2. Open the Users section by clicking on “Users” in the menu on the left.
3. Select the user you want to add to the group.
4. From the User Actions select Add User to Group
5. Select the group and click on Add to Groups (bottom right corner).

Appendix C. Using awscli to Configure and Deploy an HPCC System on AWS

Below is an example showing how to launch an HPCC System using the aws command-line interpreter, aws-cli, and the cloudformation template described in this document. This example assumes you are on a linux machine.

In this example, for the argument, --template-url, notice that the URL is the path to the S3 bucket that contains the scripts for the instances of the system. That is, this example expects to find the cloudformation template in this S3 bucket. Also, notice, in this example, that the KeyPair is the one created in Appendix A in region us-west-2.

```
aws cloudformation create-stack --capabilities CAPABILITY_IAM \  
--stack-name FastHPCC-150615-1 \  
--region us-west-2 \  
--template-url https://s3.amazonaws.com/ BuildHPCCScripts /MyHPCCCloudFormationTemplate.json \  
--parameters \  
ParameterKey=SSHUserName,ParameterValue=ec2-user \  
ParameterKey=HPCCPlatform,ParameterValue=HPCC-Platform-6.2.22-1 \  
ParameterKey=KeyPair,ParameterValue=us-west-2-keypair \  
ParameterKey=MasterInstanceType,ParameterValue=c3.large \  
ParameterKey=RoxieInstanceType,ParameterValue=c3.large \  
ParameterKey=SlaveInstanceType,ParameterValue=c3.large \  
ParameterKey=NumberOfRoxieInstances,ParameterValue=0 \  
ParameterKey=NumberOfSlaveInstances,ParameterValue=1 \  
ParameterKey=NumberOfSlavesPerInstance,ParameterValue=1 \  
ParameterKey=ScriptsS3BucketFolder,ParameterValue=s3://BestHoA/instance_files
```