



INSTITUTE FOR APPLIED
COMPUTATIONAL SCIENCE
AT HARVARD UNIVERSITY



HARVARD
School of Engineering
and Applied Sciences

Guide: Hadoop Cluster on AWS

Ignacio M. Llorente, Simon Warchol

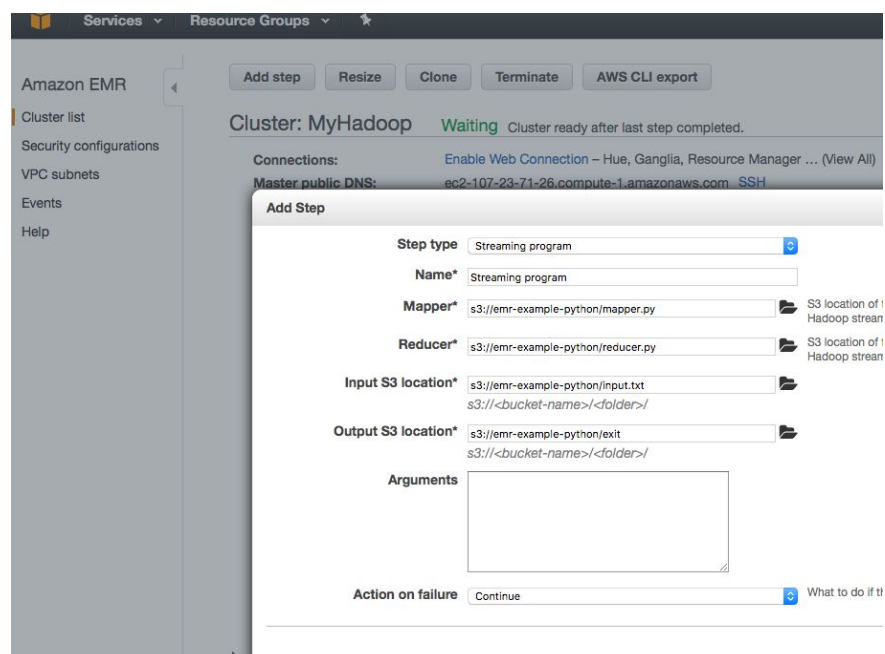
v3.0 - 14 March 2021

Abstract

This is a screenshot document of how to run EMR Hadoop cluster and run MapReduce jobs on AWS environment.

Requirements

- **First you should have followed the Guide “First Access to AWS”**. It is assumed you already have an AWS account and a key pair, and you are familiar with the AWS EC2 environment.
- We strongly recommend cluster instances with at least 4 vCPUs (**m4.xlarge**) to be able to evaluate parallel implementation within each node. m4 instances are optimized for Amazon EBS, the block



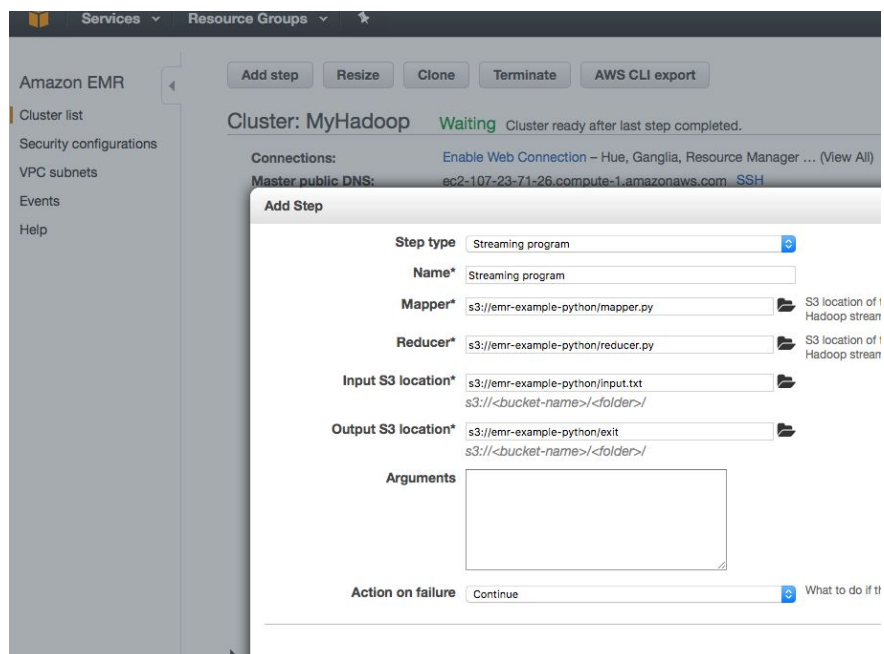


storage for ec2 instances, and are best when distributing large files amongst your cluster. They are the preferred instance type for Hadoop clusters.

- The files needed to do the exercises are available for download from **Canvas**.

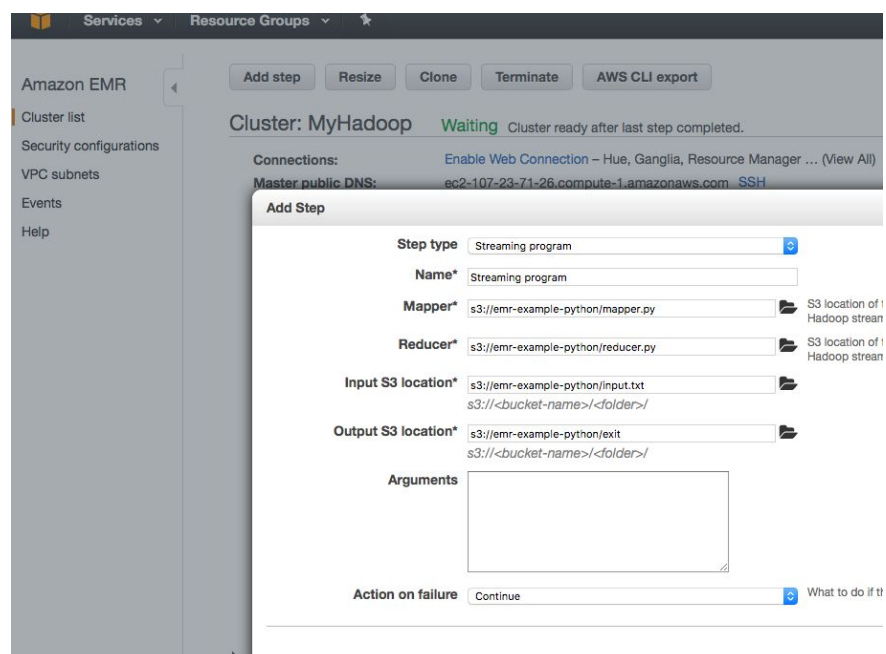
Acknowledgments

The author is grateful for constructive comments and suggestions from David Sondak, Charles Liu, Matthew Holman, Keshavamurthy Indireskumar, Kar Tong Tan, Zudi Lin, Nick Stern, Dylan Randle, Hayoun Oh, Zhiying Xu and Zijie Zhao.



1. Launch Hadoop EMR cluster

- Go to the EMR dashboard (<https://console.aws.amazon.com/elasticmapreduce/home>) and click "Create cluster". We recommend the following configuration
 - ClusterName: MyHadoop
 - Launch mode "Cluster"
 - Release: 5.32.0
 - Applications: Core Hadoop
 - Instance type: m4.xlarge
 - Number of Instances: 3
 - Key pair: course-key (or any other key you want to use, see Guide "First Access to AWS")





CS205: Computing Foundations for Computational Science, Spring 2021

Clone Terminate AWS CLI export

Cluster: MyHadoop **Waiting** Cluster ready after last step completed.

Summary Application user interfaces Monitoring Hardware Configurations Events Steps Bootstrap actions

Summary

ID: j-68YXCT086CEK
Creation date: 2021-03-14 20:11 (UTC-4)
Elapsed time: 29 minutes
After last step completes: Cluster waits
Termination protection: Off [Change](#)
Tags: -- [View All](#) / [Edit](#)
Master public DNS: ec2-18-234-211-252.compute-1.amazonaws.com [Connect to the Master Node Using SSH](#)

Configuration details

Release label: emr-5.32.0
Hadoop distribution: Amazon 2.10.1
Applications: Hive 2.3.7, Hue 4.8.0, Mahout 0.13.0, Pig 0.17.0, Tez 0.9.2
Log URI: s3://aws-logs-337392631707-us-east-1/elasticmapreduce/ [View](#)
EMRFS consistent view: Disabled
Custom AMI ID: --

Network and hardware

Availability zone: us-east-1a
Subnet ID: [subnet-0ff55742](#) [View](#)
Master: **Running** 1 m4.xlarge
Core: **Running** 2 m4.xlarge
Task: --
Cluster scaling: Not enabled

Security and access

Key name: CS205-key
EC2 instance profile: EMR_EC2_DefaultRole
EMR role: EMR_DefaultRole
Visible to all users: All [Change](#)
Security groups for Master: [sg-04397bfc4f1a3c5df](#) [View](#) (ElasticMapReduce-master)
Security groups for Core & Task: [sg-0c5fca61d4fb2f31](#) [View](#) (ElasticMapReduce-slave)

- Click on “Create Cluster”
- Wait for the cluster to be ready. This may take 5-10 minutes. This is a good opportunity to briefly call a loved one, take out the trash, or ask other questions of the esteemed TF leading this lab. The cluster is ready when its state is “Waiting” and the Master and Core under the **Networks and**

Services Resource Groups

Amazon EMR

Cluster list Security configurations VPC subnets Events Help

Add step Resize Clone Terminate AWS CLI export

Cluster: MyHadoop **Waiting** Cluster ready after last step completed.

Connections: [Enable Web Connection – Hue, Ganglia, Resource Manager ... \(View All\)](#)

Master public DNS: ec2-107-23-71-26.compute-1.amazonaws.com [SSH](#)

Add Step

Step type: Streaming program

Name*: Streaming program

Mapper*: s3://emr-example-python/mapper.py [View](#) S3 location of 1 Hadoop stream

Reducer*: s3://emr-example-python/reducer.py [View](#) S3 location of 1 Hadoop stream

Input S3 location*: s3://emr-example-python/input.txt [View](#)
s3://<bucket-name>/<folder>/

Output S3 location*: s3://emr-example-python/exit [View](#)
s3://<bucket-name>/<folder>/

Arguments

Action on failure: Continue [What to do if ti](#)

hardware section are both in “Running” state

Amazon EMR

EMR on EC2

Clusters

Notebooks

Git repositories

Security configurations

Block public access

VPC subnets

Events

EMR on EKS

Virtual clusters

Help

What's new

Cluster: My cluster **Running** Running step

Summary Application user interfaces Monitoring Hardware Configurations Events Steps Bootstrap actions

Summary

ID: j-68YXCT086CEK
 Creation date: 2021-03-14 20:11 (UTC-4)
 Elapsed time: 9 minutes
 After last step completes: Cluster waits
 Termination protection: Off [Change](#)
 Tags: -- [View All](#) / [Edit](#)
 Master public DNS: ec2-18-234-211-252.compute-1.amazonaws.com [Connect to the Master Node Using SSH](#)

Configuration details

Release label: emr-5.32.0
 Hadoop distribution: Amazon 2.10.1
 Applications: Hive 2.3.7, Hue 4.8.0, Mahout 0.13.0, Pig 0.17.0, Tez 0.9.2
 Log URI: s3://aws-logs-337392631707-us-east-1/elasticmapreduce/
 EMRFS consistent view: Disabled
 Custom AMI ID: --

Network and hardware

Availability zone: us-east-1a
 Subnet ID: subnet-0ff55742 [View](#)
 Master: **Running** 1 m4.xlarge
 Core: **Running** 2 m4.xlarge
 Task: --
 Cluster scaling: Not enabled

Security and access

Key name: CS205-key
 EC2 instance profile: EMR_EC2_DefaultRole
 EMR role: EMR_DefaultRole
 Visible to all users: All [Change](#)
 Security groups for Master: sg-04397bfc4f1a3c5df [View](#) (ElasticMapReduce-master)
 Security groups for Core & Task: sg-0c5fca61d4fb2f31 [View](#) (ElasticMapReduce-slave)

2. Login to the cluster

This section is for illustrative purposes to show how EMR is a Hadoop cluster automatically installed and configured on-demand on EC2 instances. You can skip this section to complete this guide because, as it is described in Section 3, you can submit basic MapReduce jobs from the AWS web interface

- You can SSH into master with the Master public DNS address listed above. For instance

```
ssh -i your/course/ssh-key.pem hadoop@your-master-public-dns
```

- If SSH fails, you may need to open port 22 on the master security group. Click the link to the security group next to **Security groups for Master**, click the Master security group and add an SSH

Services Resource Groups

Amazon EMR

Cluster list

Security configurations

VPC subnets

Events

Help

Cluster: MyHadoop **Waiting** Cluster ready after last step completed.

Connections: [Enable Web Connection - Hue, Ganglia, Resource Manager ... \(View All\)](#)

Master public DNS: ec2-107-23-71-26.compute-1.amazonaws.com [SSH](#)

Add Step

Step type: Streaming program

Name*: Streaming program

Mapper*: s3://emr-example-python/mapper.py [S3 location of Hadoop stream](#)

Reducer*: s3://emr-example-python/reducer.py [S3 location of Hadoop stream](#)

Input S3 location*: s3://emr-example-python/input.txt [S3 location of Hadoop stream](#)
 s3://<bucket-name>/<folder>/

Output S3 location*: s3://emr-example-python/exit [S3 location of Hadoop stream](#)
 s3://<bucket-name>/<folder>/

Arguments

Action on failure: Continue [What to do if it fails](#)

rule with port 22 and source 0.0.0.0/0.

- SSH should now work if it didn't already.

```
| => ssh -i ~/.ssh/CS205-key.pem hadoop@ec2-18-234-211-252.compute-1.amazonaws.com
Last login: Mon Mar 15 00:21:17 2021

  __|  __|_ )
 _| (  /   Amazon Linux 2 AMI
---|\___|___|

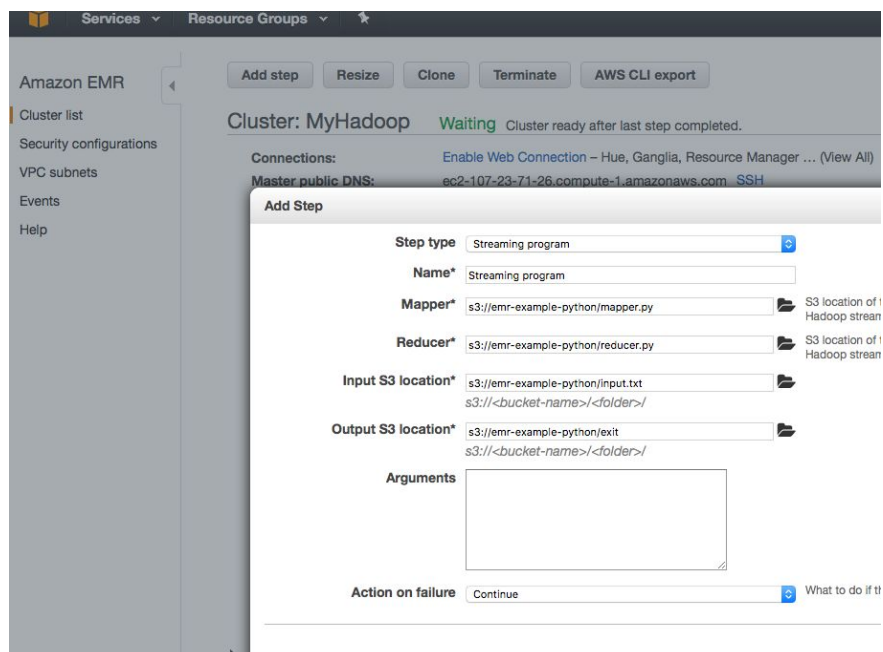
https://aws.amazon.com/amazon-linux-2/
38 package(s) needed for security, out of 76 available
Run "sudo yum update" to apply all updates.

EEEEEEEEEEEEEEEEEEEE MMMMMMMM MMMMMMMM RRRRRRRRRRRRRRRR
E::::::::::::::::::::E M::::::::M M::::::::M R::::::::::::R
EE::::EEEEEEEE::::E M::::::::M M::::::::M R::::RRRRRR::::R
 E::::E EEEEE M::::::::M M::::::::M RR::R R::R
 E::::E M::::M M::M M::M M::M R::R R::R
 E::::EEEEEEEE M::::M M::M M::M M::M R::RRRRRR::::R
 E::::::::::::E M::::M M::M M::M M::M R:::::::::RR
 E::::EEEEEEEE M::::M M::M M::M M::M R::RRRRRR::R
 E::::E M::::M M::M M::M M::M R::R R::R
 E::::E EEEEE M::::M MMM M::::M R::R R::R
 EE::::EEEEEEEE::::E M::::M M::::M R::R R::R
 E::::::::::::E M::::M M::::M RR::R R::R
 EEEEEEEEEEEEEEEEE MMMMMMMM MMMMMMMM RRRRRRR RRRRRR

[hadoop@ip-172-31-17-242 ~]$
```

3. Submit a MapReduce job

Hadoop Streaming is a utility that comes with Hadoop that enables you to develop MapReduce executables in languages other than Java. A Streaming application reads input from standard input and then runs a script or executable (called a mapper) against each input. The result from each of the inputs



is saved locally on a Hadoop Distributed File System (HDFS) partition. After all the input is processed by the mapper, a second script or executable (called a reducer) processes the mapper results. The results from the reducer are sent to standard output.

- Upload [mapper](#), [reducer](#) and [input](#) files to a new S3 bucket via the AWS interface. Create a S3 bucket, I named it `emr-example-python`. Remember this name should be unique. Moreover, because of Hadoop requirements, S3 bucket names used with Amazon EMR have the following constraints: must contain only lowercase letters, numbers, periods (.), and hyphens (-); and cannot end in numbers. This is a great opportunity to express your creativity!
 - Both mapper and reducer assume that lines are fed in through `sys.stdin`. Good sources of available text to play with are in Project Gutenberg.

Upload

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose **Add files**, or **Add folders**.

Files and folders (3 Total, 1.2 MB)

All files and folders in this table will be uploaded.

Find by name

<

1

>

<input type="checkbox"/>	Name	▲	Folder	▼	Type	▼	Size	▼
<input type="checkbox"/>	input.txt		-		text/plain		1.2 MB	
<input type="checkbox"/>	mapper.py		-		text/x-python-script		209.0 B	
<input type="checkbox"/>	reducer.py		-		text/x-python-script		333.0 B	

The screenshot shows the Amazon EMR console interface. On the left is a sidebar with navigation links: Amazon EMR, Cluster list, Security configurations, VPC subnets, Events, and Help. The main area displays the 'Cluster: MyHadoop' status as 'Waiting' with a note 'Cluster ready after last step completed.' Below this, there are links for 'Connections' and 'Master public DNS'. An 'Add Step' modal dialog is open in the foreground. The dialog contains the following fields:

- Step type:** Streaming program
- Name:** Streaming program
- Mapper:** s3://emr-example-python/mapper.py
- Reducer:** s3://emr-example-python/reducer.py
- Input S3 location:** s3://emr-example-python/input.txt
- Output S3 location:** s3://emr-example-python/exit
- Arguments:** (Empty text area)
- Action on failure:** Continue

- Go to the Hadoop cluster dashboard's **Steps** tab and click on "Add Step" with the following configuration
 - Step type: Streaming program
 - Name: MyHadoopJob
 - Mapper: Complete path to uploaded mapper
 - Reducer: Complete path to uploaded reducer
 - Input: Complete path to uploaded input
 - Output: Complete path to new folder to be created with the output (**it should not exist**)
- Wait for the "step" to be "completed"
- After "completed" you can check the execution time in the `controller` log file

Filter: All steps 2 steps (all loaded)					
ID	Name	Status	Start time (UTC-4)	Elapsed time	Log files
s-2T2QQSXTDEIL	MyHadoopJob	Completed	2021-03-14 20:44 (UTC-4)	1 minute	controller syslog stderr stdout
JAR location : command-runner.jar Main class : None Arguments : hadoop-streaming -files s3://simon-simon-simon-simon/mapper.py,s3://simon-simon-simon-simon/reducer.py -mapper mapper.py -reducer reducer.py -input s3://simon-simon-simon-simon/input.txt -output s3://simon-simon-simon-simon/output.txt Action on failure: Continue					

```
INFO total process run time: 78 seconds
2021-03-15T00:46:01.995Z INFO Step created jobs: job_1615767477849_0001
2021-03-15T00:46:01.996Z INFO Step succeeded with exitCode 0 and took 78 seconds
```

- If the job is not successfully "completed", you can check the logging files for further information
- Finally, check the results in the bucket, Hadoop creates one output file for each executed reducer task

The screenshot shows the Amazon EMR console interface. On the left is a sidebar with navigation links: Amazon EMR, Cluster list, Security configurations, VPC subnets, Events, and Help. The main panel shows the 'Cluster: MyHadoop' page with a status of 'Waiting' and a message 'Cluster ready after last step completed.' Below this, there are buttons for 'Add step', 'Resize', 'Clone', 'Terminate', and 'AWS CLI export'. A modal window titled 'Add Step' is open, displaying the following configuration fields:

- Step type:** Streaming program
- Name:** Streaming program
- Mapper:** s3://jemr-example-python/mapper.py
- Reducer:** s3://jemr-example-python/reducer.py
- Input S3 location:** s3://jemr-example-python/input.txt
- Output S3 location:** s3://jemr-example-python/exit
- Arguments:** (Empty text area)
- Action on failure:** Continue



CS205: Computing Foundations for Computational Science, Spring 2021

Viewing 1 to 8				
<input type="checkbox"/> Name ▾	Last modified ▾	Size ▾	Storage class ▾	
<input type="checkbox"/> _SUCCESS	Mar 3, 2020 7:18:26 PM GMT+0100	0 B	Standard	
<input type="checkbox"/> part-00000	Mar 3, 2020 7:18:16 PM GMT+0100	24.7 KB	Standard	
<input type="checkbox"/> part-00001	Mar 3, 2020 7:18:17 PM GMT+0100	25.4 KB	Standard	
<input type="checkbox"/> part-00002	Mar 3, 2020 7:18:25 PM GMT+0100	25.7 KB	Standard	
<input type="checkbox"/> part-00003	Mar 3, 2020 7:18:25 PM GMT+0100	25.0 KB	Standard	
<input type="checkbox"/> part-00004	Mar 3, 2020 7:18:24 PM GMT+0100	25.7 KB	Standard	
<input type="checkbox"/> part-00005	Mar 3, 2020 7:18:24 PM GMT+0100	25.8 KB	Standard	
<input type="checkbox"/> part-00006	Mar 3, 2020 7:18:21 PM GMT+0100	26.1 KB	Standard	

Terminate the cluster when you are sure you are done for the day to avoid incurring charges

The screenshot shows the Amazon EMR console interface. On the left is a sidebar with navigation links: Amazon EMR, Cluster list, Security configurations, VPC subnets, Events, and Help. The main area displays the 'Cluster: MyHadoop' page, which is in a 'Waiting' state. Above the cluster details are buttons for 'Add step', 'Resize', 'Clone', 'Terminate', and 'AWS CLI export'. The 'Add Step' dialog is open, showing the following fields:

- Step type:** Streaming program
- Name:** Streaming program
- Mapper:** s3://jemr-example-python/mapper.py
- Reducer:** s3://jemr-example-python/reducer.py
- Input S3 location:** s3://jemr-example-python/input.txt
- Output S3 location:** s3://jemr-example-python/exit
- Arguments:** (Empty text area)
- Action on failure:** Continue

Helpful text on the right side of the dialog indicates that the S3 locations are for Hadoop streams.