EMR File System Client-side Encryption Using AWS KMS-managed Keys

**SPL-148 - Version 1.1.17**

© 2023 Amazon Web Services, Inc. or its affiliates. All rights reserved. This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited. All trademarks are the property of their owners.

Note: Do not include any personal, identifying, or confidential information into the lab environment. Information entered may be visible to others.

Corrections, feedback, or other questions? Contact us at [*AWS Training and Certification*](https://support.aws.amazon.com/#/contacts/aws-training).

**Lab Overview**

In this lab you will enable client-side at-rest encryption using AWS KMS-managed key for data stored in Amazon S3 with the EMR File System (EMRFS). Using Amazon EMR, you will create a security configuration to encrypt the object being written to S3 with client-side encryption using the AWS KMS-managed key specified by you, and decrypt objects with the same key that was used to encrypt them. This will allow you to more easily leverage frameworks like Apache Spark, Apache Tez, and Apache Hadoop MapReduce on Amazon EMR to run big data analytics, stream processing, machine learning, and ETL workloads on confidential data.

TOPICS COVERED

This lab will demonstrate how to:

* Create an Amazon S3 bucket
* Create a key using AWS KMS
* Create security configuration in EMR to enable client-side encryption using AWS KMS-managed key
* Launch an AWS Elastic Map Reduce(EMR) cluster using the AWS Management Console
* Read and write objects from and to S3 using AWS EMR File System (EMRFS)
* View EMR output data directly from Amazon S3

TECHNICAL KNOWLEDGE PREREQUISITES

To successfully complete this lab, you should be familiar with basics of Hadoop and Hadoop File System (HDFS).

You should also be familiar with basic Linux server administration and comfortable using the Linux command-line tools.

OTHER AWS SERVICES

AWS services other than those needed for this lab are disabled by IAM policy during your access time in this lab. In addition, the capabilities of the services used in this lab are limited to what’s required by the lab and in some cases are even further limited as an intentional aspect of the lab design. You should expect errors when accessing other services or performing actions beyond those provided in this lab guide.

WHAT IS AMAZON EMR?

Amazon EMR is a web service that makes it easy to quickly and cost-effectively process vast amounts of data. Amazon EMR simplifies big data processing, providing a managed Hadoop framework that makes it easy, fast, and cost-effective for you to distribute and process vast amounts of your data across dynamically scalable Amazon EC2 instances. You can also run other popular distributed frameworks such as Apache Spark and Presto in Amazon EMR, and interact with data in other AWS data stores such as Amazon S3 and Amazon DynamoDB. Amazon EMR securely and reliably handles your big data use cases, including log analysis, web indexing, data warehousing, machine learning, financial analysis, scientific simulation, and bio-informatics.

WHAT IS EMRFS?

EMRFS is an implementation of HDFS used for reading and writing regular files from Amazon EMR directly to Amazon S3. EMRFS provides the convenience of storing persistent data in Amazon S3 for use with Hadoop while also providing features like Amazon S3 server-side encryption, read-after-write consistency, and list consistency.

WHAT IS AWS KMS?

AWS Key Management Service (KMS) is a managed service that makes it easy for you to create and control the encryption keys used to encrypt your data, and uses Hardware Security Modules (HSMs) to protect the security of your keys. AWS Key Management Service is integrated with several other AWS services to help you protect the data you store with these services. AWS Key Management Service is also integrated with AWS CloudTrail to provide you with logs of all key usage to help meet your regulatory and compliance needs.

WHAT IS AMAZON S3?

Amazon Simple Storage Service (Amazon S3), provides developers and IT teams with secure, durable, highly-scalable cloud storage. Amazon S3 is easy to use object storage, with a simple web service interface to store and retrieve any amount of data from anywhere on the web. With Amazon S3, you pay only for the storage you actually use. On the AWS cloud, Amazon S3 is a good candidate for a data lake implementation to store large-scale data for big data analytics using Amazon EMR.

WHAT IS A SECURITY CONFIGURATION IN AMAZON EMR?

You can use a security configuration to encrypt data at-rest, data in-transit, or both. Each security configuration is stored in Amazon EMR rather than in cluster configuration objects, so you can easily reuse a configuration to specify encryption settings whenever a cluster is created.

**Start lab**

1. To launch the lab, at the top of the page, choose **Start lab**.

 You must wait for the provisioned AWS services to be ready before you can continue.

1. To open the lab, choose **Open Console**.

You are automatically signed in to the AWS Management Console in a new web browser tab.

**Do not change the Region unless instructed.**

COMMON SIGN-IN ERRORS

**Error: You must first sign out**



If you see the message, **You must first log out before logging into a different AWS account:**

* Choose the **click here** link.
* Close your **Amazon Web Services Sign In** web browser tab and return to your initial lab page.
* Choose **Open Console** again.

**Error: Choosing Start Lab has no effect**

In some cases, certain pop-up or script blocker web browser extensions might prevent the **Start Lab** button from working as intended. If you experience an issue starting the lab:

* Add the lab domain name to your pop-up or script blocker’s allow list or turn it off.
* Refresh the page and try again.

**Task 1: Create an Amazon S3 bucket**

You will create an Amazon S3 bucket to write encrypted objects (client-side using a KMS key) to the bucket from the EMR cluster as well as read objects from the bucket and decrypt them using the same KMS key.

1. At the top of the **AWS Management Console**, in the search bar, search for and choose

**S3**

 in a new browser tab.

1. Choose **Create bucket** then configure:

* **Bucket name:**

emr-NUMBER

.

* Replace **NUMBER** with a random number.

**Note:** Example bucket name should look like *emr-9192016*

* Copy the name of your bucket to your text editor.
* **Region:** Use the **Region** value located to the left of these instructions.
* Choose **Create bucket**.

**Task 2: Create an AWS KMS Key**

In this task, you will create an AWS KMS Key. You will use this KMS Key to encrypt objects (client side encryption) while writing to S3 using EMRFS and decrypt using the same key while reading the same objects back into the cluster.

1. At the top of the **AWS Management Console**, in the search bar, search for and choose

**Key Management Service**

 in a new browser tab.

1. Choose **Create key** then configure:

* **Key type:** *Symmetric*.
* Choose **Next**.

1. On the **Add labels** page, configure:

* **Alias:**

emr-seclab-key

* **Description:**

EMR key for use with encrypted clusters

* Choose **Next**

1. On the **Define key administrative permissions** page:

* Select  the user or role that you are logged in with. Your user or role is located at the top-right of the screen.
* Choose **Next**

**Key Administrators** are users or roles that will manage access to the encryption key.

1. On the **Define key usage permissions** page:

* Select  the user or role that you are logged in with.
* Choose **Next**

**Key Users** are the users or roles that will use the key to encrypt and decrypt data.

1. On the **Review** page:

* Review the key policy
* Choose **Finish**

**Task 3: Create a Security Configuration in Amazon EMR**

In this task, you will create a security configuration in Amazon EMR to enable client-side at-rest encryption using an AWS KMS-managed key for data stored in Amazon S3 with EMRFS.

1. At the top of the **AWS Management Console**, in the search bar, search for and choose

**EMR**

 in a new browser tab.

1. At the top-right of the screen, ensure that you are in the same region represented by the **Region** located to the left of these instructions. If you are not in the same region, change the region to match the region code located to the left of these instructions.
2. In the left navigation pane, expand the  **EMR on EC2** option and choose **Security configurations**.
3. On the **Security configurations** page, choose **Create** then configure:

* **Name:**

emr-cfg-NAME

* Replace **NAME** with your name

1. In the **Security configuration setup options** section:

* Select the radio button for the  *Choose custom settings* option.

1. In the **Encryption** section:

* Select  **Turn on at-rest encryption for data in Amazon S3**
* **Encryption mode:** *CSE-KMS*
* **Choose your AWS KMS key:** *emr-seclab-key*

1. Scroll to the bottom of the screen, then choose **Next**
2. Choose **Next**.
3. Choose **Create**.

**Task 4: Launching an Amazon Elastic MapReduce Cluster**

In this task, you will launch an EMR cluster.

1. In the left navigation pane, under  **EMR on EC2** option, choose **Clusters**.
2. Choose **Create cluster**.
3. For **Name:** enter

emr-sec

1. For **Cluster configuration**, configure.

* **Instance type for Primary:** *m4.large*.
* **Instance type for Core:** *m4.large*.
* **Instance type for Task 1 of 1:** *m4.large*.

1. For **Networking**, configure:

* **Virtual private cloud (VPC):** Choose **Browse** and select *Lab VPC* and then choose **Choose**.

1. For **Cluster logs - optional**, configure the following:

* **Amazon S3 location**: Choose **Browse S3**.
* Select the radio button  for the bucket that you created.
* Choose **Choose**.
* At the end of the **Amazon S3 location** name enter

/logs

.

**Note:** Your URL should look similar to *s3://emr-9192016/logs*.

1. In the **Security configuration and EC2 key pair** section, for **Security configuration**, select the security configuration that you created.

This is the security configuration that you created earlier to enable client-side encryption using AWS KMS-managed key.

1. For **Amazon EMR service role**, choose **Choose an existing service role**.
2. For **Service role**, select *EMR\_DefaultRole*.
3. For **EC2 instance profile for Amazon EMR**:

* Select **Choose an existing instance profile**.
* For **Instance profile**, choose *EMR\_EC2\_DefaultRole*.

1. Choose **Create cluster**.
2. Wait until the status of the EMR cluster you just launched with name changes to Waiting Cluster ready after last step completed.

The cluster creation will take approximately 10 minutes.

1. Choose the  **refresh** button every few minutes to check the status.
2. Copy the **Primary node public DNS** value to a text editor.

* **Note:** It should look something like **ec2-54-162-168-200.compute-1.amazonaws.com.**

**Task 5: Validate Client Side Encryption**

1. At the top of the **AWS Management Console**, in the search bar, search for and choose

**EC2**

 in a new browser tab.

1. In the left navigation pane, choose **Instances**.
2. In the search field, enter the **Primary node public DNS** value that you copied earlier and press **Enter**.
3. Choose the instance ID of the primary node.
4. Choose **Connect**.
5. With **Session Manager** selected, choose **Connect**.

WRITE AN ENCRYPTED OBJECT TO S3 FROM YOUR EMR CLUSTER

In this section, you’ll write an object to S3 from the EMR Cluster and encrypt it using client-side encryption.

1. In the terminal, type the following command to create a text file.

nano outputFile.txt

1. Type some text into text file:

This is my story

1. Save the file by:

* Pressing **Ctrl + X**
* Entering

Y

 then Pressing **Enter**

1. Write your file to your EMR cluster by:

* Entering

hadoop fs -put outputFile.txt s3://BUCKET/

* Replace **BUCKET** with the name of your S3 bucket
* **Note:** Make sure you add a **/** after your bucket name.

This will write the file as an encrypted object (client side using the AWS KMS Key) to the S3 bucket that you created as part of the lab.

READ THE ENCRYPTED OBJECT DIRECTLY FROM S3

In this section, you will verify if the object is encrypted when read directly from the S3 bucket.

1. In your Session Manager session:

* Enter

aws s3 cp s3://BUCKET/outputFile.txt encryptedOutputFile.txt

* Replace **BUCKET** with the name of your bucket
* Press **Enter**

This will download the encrypted object from your S3 bucket.

1. In your Session Manager session enter

more encryptedOutputFile.txt

.

This will allow you view your encrypted object.

1. Review the output. Can you read it? You should see encrypted data that is not readable.

READ THE ENCRYPTED OBJECT FROM S3 INTO THE EMR CLUSTER USING EMRFS TRANSPARENT ENCRYPTION

In this section, you will read the encrypted object from S3 into the EMR cluster and verify that its decrypted using EMRFS transparent encryption.

1. Enter

hadoop fs -cat s3://BUCKET/outputFile.txt

.

1. Replace **BUCKET** with the name of your bucket.
2. Press **Enter**.

This will read content of the encrypted object from S3 in the EMR cluster. You should be able to read the contents of the text file. Do you recognize the text you wrote earlier?

**Conclusion**

 Congratulations! You now have successfully:

* Created an Amazon S3 bucket and an encryption key using AWS Key Management System(KMS)
* Launched a Hadoop managed cluster with EMR that is enabled for client-side encryption using AWS KMS in EMRFS
* Using EMRFS transparent encryption, created an encrypted (client side using AWS KMS key) object in S3 through the EMR cluster as well as read the content of the encrypted object from S3 by decrypting the object with the same key used to encrypt them
* Verified that the content of the object in S3 was actually encrypted by directly reading the object from S3

**End lab**

Follow these steps to close the console and end your lab.

1. Return to the **AWS Management Console**.
2. At the upper-right corner of the page, choose **AWSLabsUser**, and then choose **Sign out**.
3. Choose **End lab** and then confirm that you want to end your lab.

**Additional Resources**

* [AWS Key Management System](https://aws.amazon.com/kms/)
* [EMR File System](http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-fs.html)
* [Hadoop](http://hadoop.apache.org/)
* [Hadoop Command Line](https://hadoop.apache.org/docs/r1.0.4/commands_manual.html)

For more information about AWS Training and Certification, see [*https://aws.amazon.com/training/*](https://aws.amazon.com/training/).

*Your feedback is welcome and appreciated.*  
If you would like to share any feedback, suggestions, or corrections, please provide the details in our [*AWS Training and Certification Contact Form*](https://support.aws.amazon.com/#/contacts/aws-training).