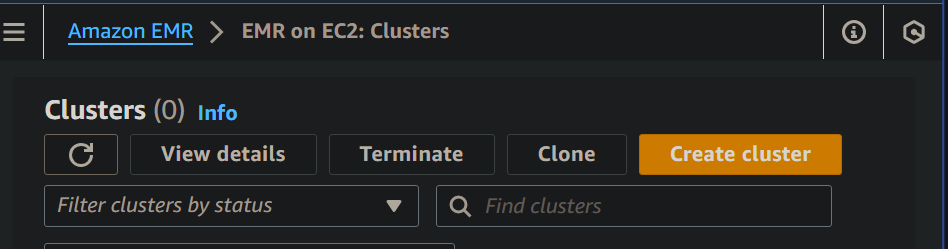
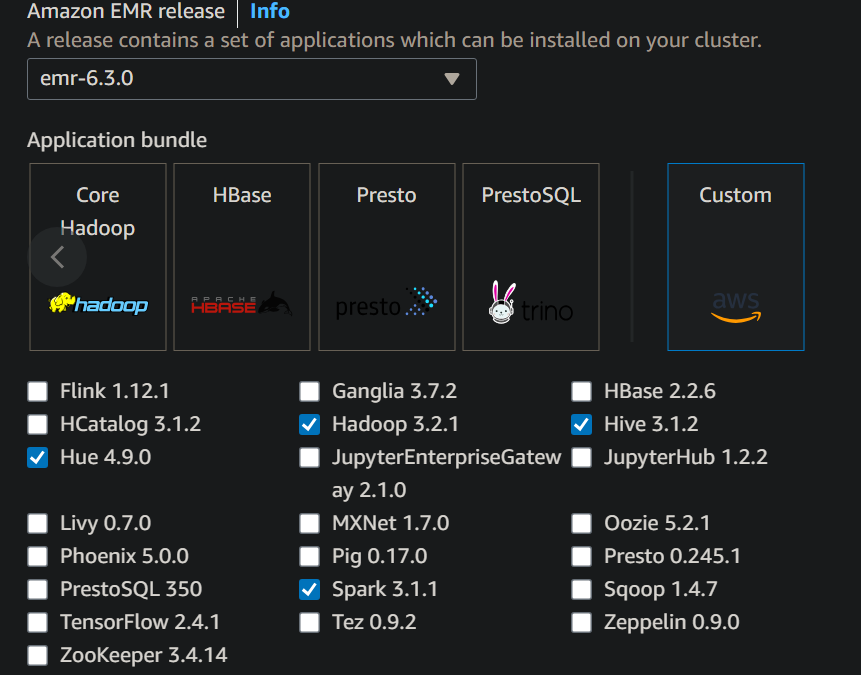
# Lab 4 - Interactive Demos

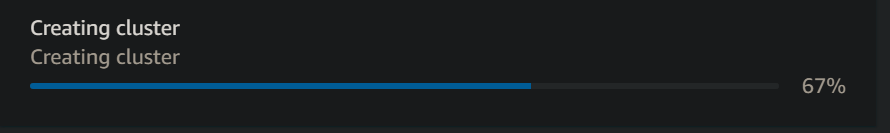
**Demo 1: Launching an Amazon EMR cluster**





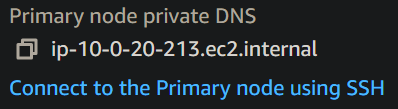
|  |
| --- |
|  |

| Summary Info  Name and applications - required  Name  mycluster  Amazon EMR release  emr-6.3.0  Application bundle  Custom (Hadoop 3.2.1, Hive 3.1.2, Hue 4.9.0, Spark 3.1.1)  Cluster configuration - required  Uniform instance groups  Primary (m4.large), Core (m4.large), Task (m4.4xlarge)  Cluster scaling and provisioning - required  Provisioning configuration  Core size: 1 instance  Task size: 1 instance  Networking - required  VPC  vpc-09f46284b...  Subnet  subnet-051109...  Cluster termination  Cluster termination  Manually terminate cluster  Cluster logs  Amazon S3 location  s3://aws-logs...  Security configuration and EC2 key pair  Security configuration  emr-cfg-34495...  Amazon EC2 key pair  EMRKey  Identity and Access Management (IAM) roles - required  Service role  EMRDefaultRole  Instance profile  EMR\_EC2\_DefaultRole |
| --- |



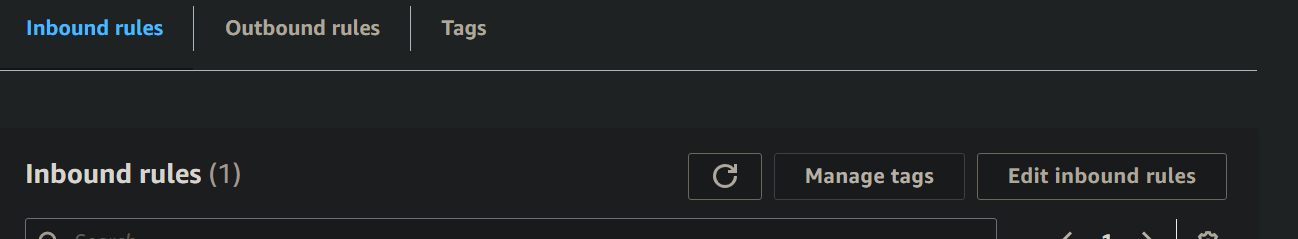
**Demo 2: Connect to an EMR cluster**

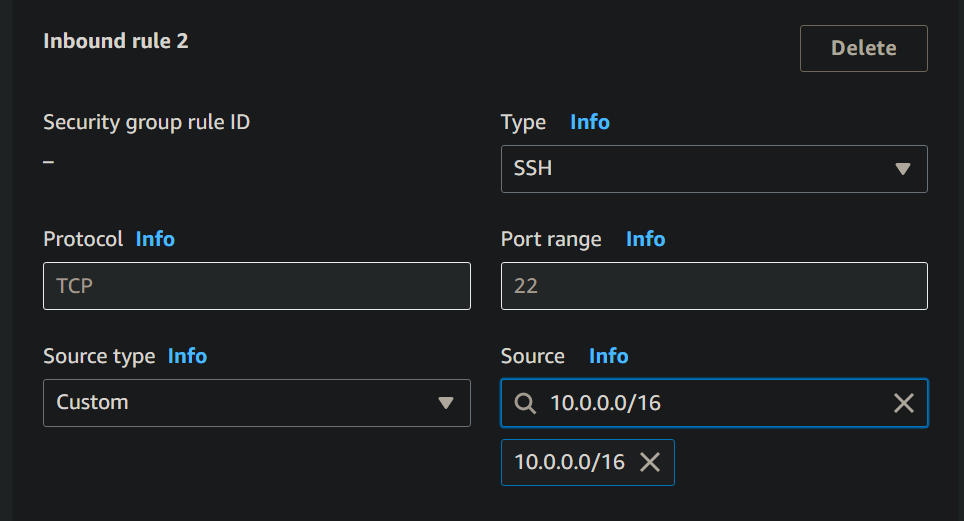
2.1 using SSH



2.2 go to security (has a link to SG, which opens in EC2) and add a rule to this security group







2.3 use link to open CLI, and connect using

*# Get EMR cluster ID and export to the Environment.*

*export ID=$(aws emr list-clusters | jq '.Clusters[0].Id' | tr -d '"')*

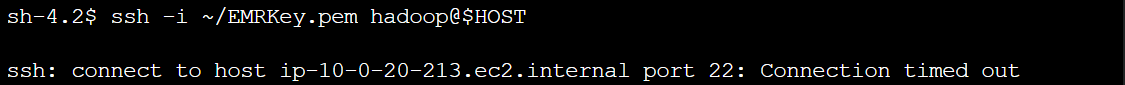
*# Use the ID to get the PublicDNS name of the EMR cluster*

*# and export to the Environment.*

*export HOST=$(aws emr describe-cluster --cluster-id $ID | jq '.Cluster.MasterPublicDnsName' | tr -d '"')*

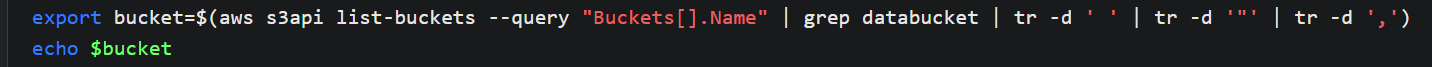
*# SSH to the EMR cluster*

*ssh -i ~/EMRKey.pem hadoop@$HOST*

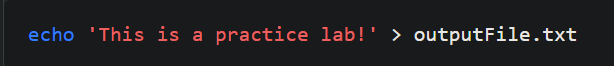


**Demo 3: Client-side encryption with EMRFS**

3.1 S3 bucket



3.2 enter text in a file in S3



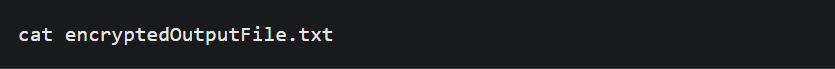


Auto-encryption - When EMRFS is configured to use server-side encryption with S3, it automatically encrypts data before it is stored persistently in S3. This means that when you use EMRFS to write data (outputFile.txt in this case) to your S3 bucket (${bucket}), the data is encrypted before being stored in S3.

Now, we will get the encrypted file from S3, and in next step, decrypt it



Here, decryption does not occur; the file remains encrypted.



Decrypt -

EMRFS, handles the decryption of the object automatically before making it available for reading within the EMR cluster.

