EMR Serverless Setup with EMR Studio

1. Setup Permissions for IAM user or group or role using the follow JSON, modify the permission if users are not supposed to make new studio or URL etc.

*{*

*"Version": "2012-10-17",*

*"Statement": [*

*{*

*"Sid": "EMRStudioCreate",*

*"Effect": "Allow",*

*"Action": [*

*"elasticmapreduce:CreateStudioPresignedUrl",*

*"elasticmapreduce:DescribeStudio",*

*"elasticmapreduce:CreateStudio",*

*"elasticmapreduce:ListStudios"*

*],*

*"Resource": "\*"*

*},*

*{*

*"Sid": "EMRServerlessFullAccess",*

*"Effect": "Allow",*

*"Action": [*

*"emr-serverless:\*"*

*],*

*"Resource": "\*"*

*},*

*{*

*"Sid": "AllowEC2ENICreationWithEMRTags",*

*"Effect": "Allow",*

*"Action": [*

*"ec2:CreateNetworkInterface"*

*],*

*"Resource": [*

*"arn:aws:ec2:\*:\*:network-interface/\*"*

*],*

*"Condition": {*

*"StringEquals": {*

*"aws:CalledViaLast": "ops.emr-serverless.amazonaws.com"*

*}*

*}*

*},*

*{*

*"Sid": "AllowEMRServerlessServiceLinkedRoleCreation",*

*"Effect": "Allow",*

*"Action": "iam:CreateServiceLinkedRole",*

*"Resource": "arn:aws:iam::\*:role/aws-service-role/\*"*

*}*

*]*

*}*

1. Setup permissions for EMR (runtime role) – First create a S3 bucket to use to store EMR scripts, logs and outputs.
2. Create a role called EMRServerlessS3RuntimeRole

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Principal": {

"Service": "emr-serverless.amazonaws.com"

},

"Action": "sts:AssumeRole"

}

]

}

1. Create a policy called EMRServerlessS3AndGlueAccessPolicy and attach to the role created above. Replace the bucket name with your bucket name.

*{*

*"Version": "2012-10-17",*

*"Statement": [*

*{*

*"Sid": "ReadAccessForEMRSamples",*

*"Effect": "Allow",*

*"Action": [*

*"s3:GetObject",*

*"s3:ListBucket"*

*],*

*"Resource": [*

*"arn:aws:s3:::\*.elasticmapreduce",*

*"arn:aws:s3:::\*.elasticmapreduce/\*"*

*]*

*},*

*{*

*"Sid": "FullAccessToOutputBucket",*

*"Effect": "Allow",*

*"Action": [*

*"s3:PutObject",*

*"s3:GetObject",*

*"s3:ListBucket",*

*"s3:DeleteObject"*

*],*

*"Resource": [*

*"arn:aws:s3:::DOC-EXAMPLE-BUCKET",*

*"arn:aws:s3:::DOC-EXAMPLE-BUCKET/\*"*

*]*

*},*

*{*

*"Sid": "GlueCreateAndReadDataCatalog",*

*"Effect": "Allow",*

*"Action": [*

*"glue:GetDatabase",*

*"glue:CreateDatabase",*

*"glue:GetDataBases",*

*"glue:CreateTable",*

*"glue:GetTable",*

*"glue:UpdateTable",*

*"glue:DeleteTable",*

*"glue:GetTables",*

*"glue:GetPartition",*

*"glue:GetPartitions",*

*"glue:CreatePartition",*

*"glue:BatchCreatePartition",*

*"glue:GetUserDefinedFunctions"*

*],*

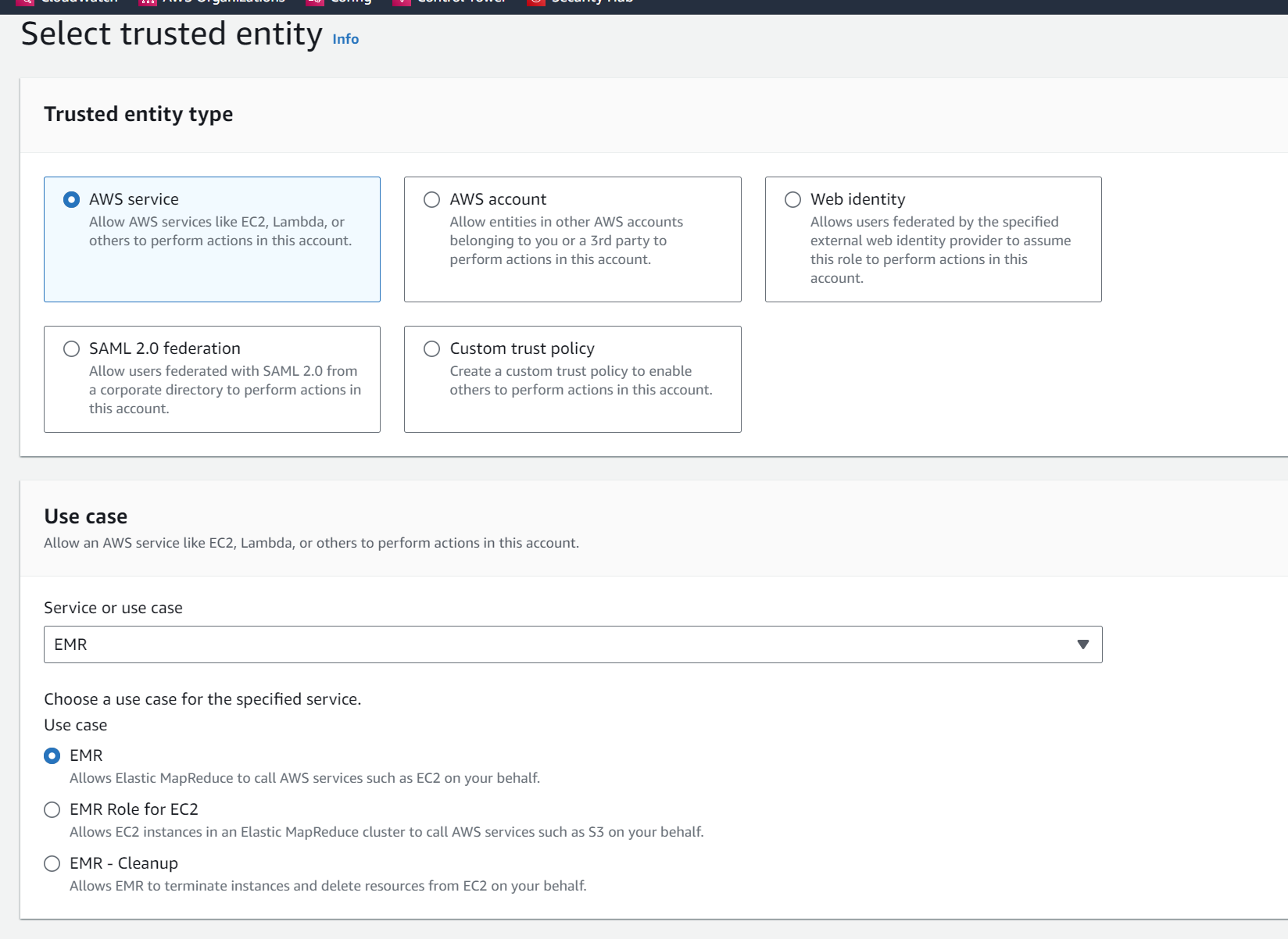
*"Resource": ["\*"]*

*}*

*]*

*}*

1. Create role for service role EMR Studio
   1. Create Role, choose AWS service, search for EMR, choose EMR from drop down menu, and choose EMR the first option.
   2. Attach the above created policy “EMRServerlessS3AndGlueAccessPolicy” to this role you made above.



1. Create EMR Studio
   1. Go to EMR then EMR Studio > then Studio
   2. Click Create studio
   3. Give meanigul name, choose VPC and subnets, default security groups, choose the IAM role we made from step 3, and the S3 bucket you intend to use.
   4. Click create.
   5. Provide Name, descrcription, VPC, subnet, security group, leave reset default
   6. For IAM service role choose AWSElasticMapReduce Role we made in step 3
   7. Choose the bucket you intend to use
2. Create Application
   1. click on the Studio Access URL, a new page will open up
   2. under the left side menu choose application, create application
   3. Provide Name, choose between Spark or Hive, and which version of EMR and architecture
   4. If need choose custom setting otherwise stick to defaults. Custom setting will allow to modify and setup pre-intialized capacity in CPU and memory as a warm pool ready to run job immediately. Set limits on the amount of maximum CPU, memory and disk. Retnetion time and location for logs. Set idle time for the application before the job shuts down, 15 min is default. Change networking setting etc.
   5. Click create application to complete
3. Create a job
   1. Click on the application made in previous step
   2. Click submit job
   3. Provide Name, choose the role we made “EMRServerlessS3RuntimeRole” we made in step 2
   4. Choos the location in the S3 bucket where the script is stored, choose the script it self
   5. For script argument enter the path for the output from the script like so ["s3://DOC-EXAMPLE-BUCKET/emr-serverless-spark/output"] include the square brackets
   6. For spark properties enter the needed value for CPU RAM networking etc or you can modify and copy text only like so

--conf spark.executor.cores=1 --conf spark.executor.memory=4g --conf spark.driver.cores=1 --conf spark.driver.memory=4g --conf spark.executor.instances=1

* 1. Submit the job and it should auto run
  2. See the output in the S3 under the outputs folder