Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

Objetivo

Este documento tem como objetivo de descrever o provisionamento do ambiente para prática de processamento de dados.

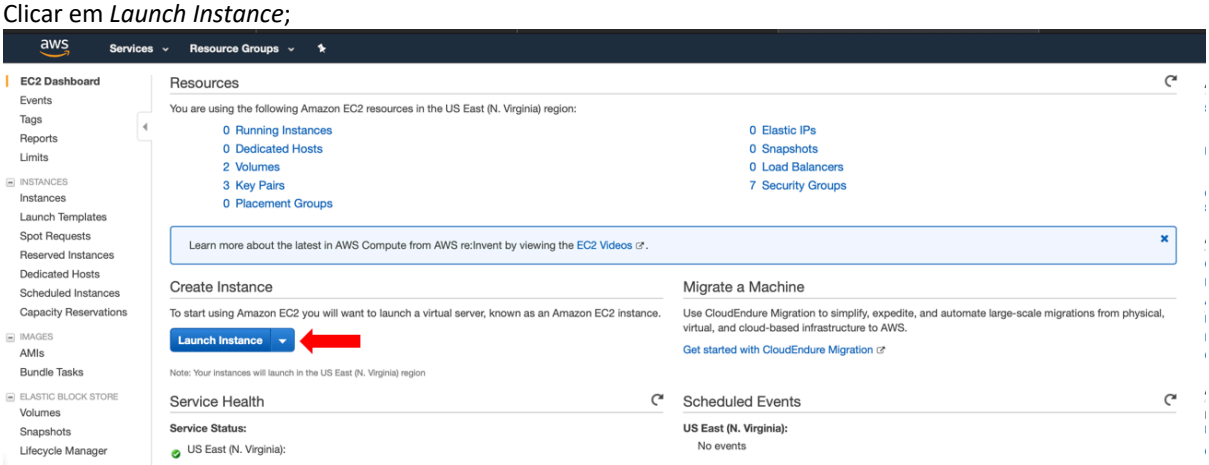
Atividades:

- Provisionamento Red Hat Enterprise Linux 8
  - StreamSets
- Provisionamento Windows Server 2019
  - SQL Server 2017
- Provisionamento Amazon Linux
  - Kafka
- Provisionamento EMR – Elastic MapReduce
  - Hive 2.3.5
  - Spark 2.4.4
  - Hue 4.4.0
  - Livy 0.6.0
  - Tez 0.9.2
  - Zeppelin 0.8.1
  - JupyterHub 1.0.0
- Configuração do ambiente para ingestão de dados ao Data Lake

Red Hat Enterprise Linux 8 – StreamSets

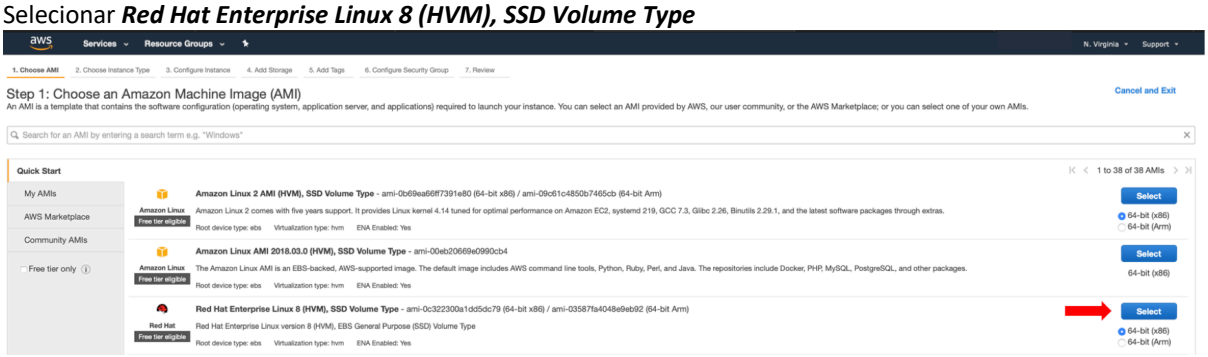
Provisionamento de instância EC2 (Elastic Compute Cloud):

1. Realizar *login* na console AWS;
2. Acessar *Services > Compute > EC2*;
3. Clicar em *Launch Instance*;

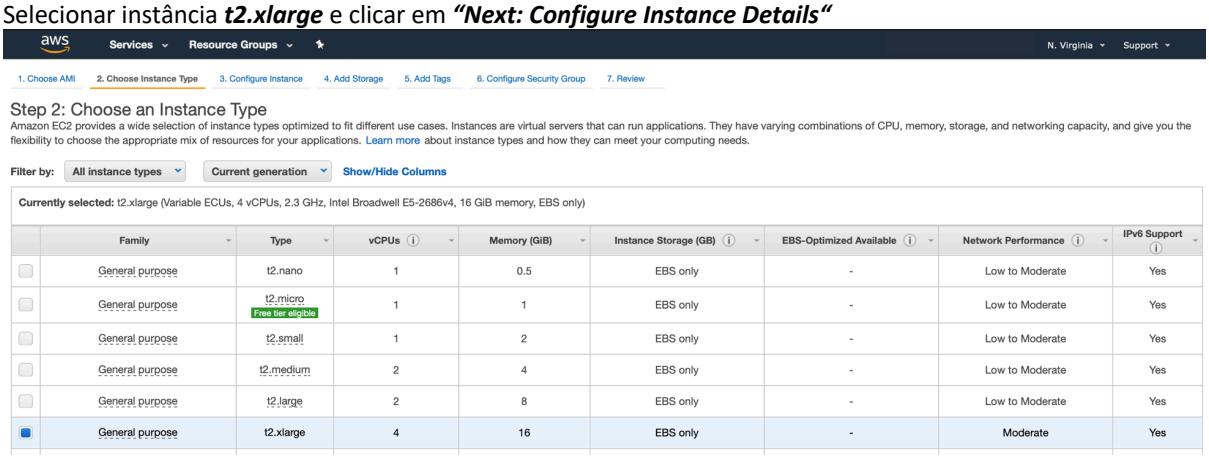


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

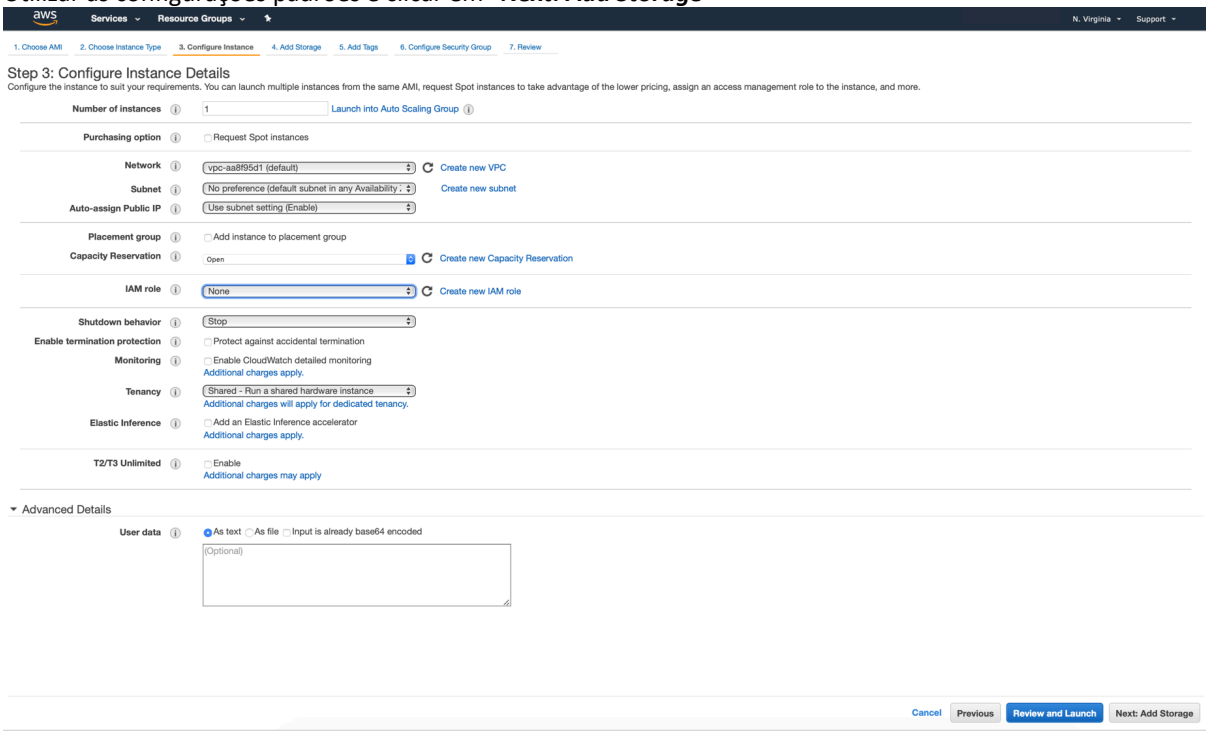
4. Selecionar ***Red Hat Enterprise Linux 8 (HVM), SSD Volume Type***



5. Selecionar instância ***t2.xlarge*** e clicar em ***"Next: Configure Instance Details"***

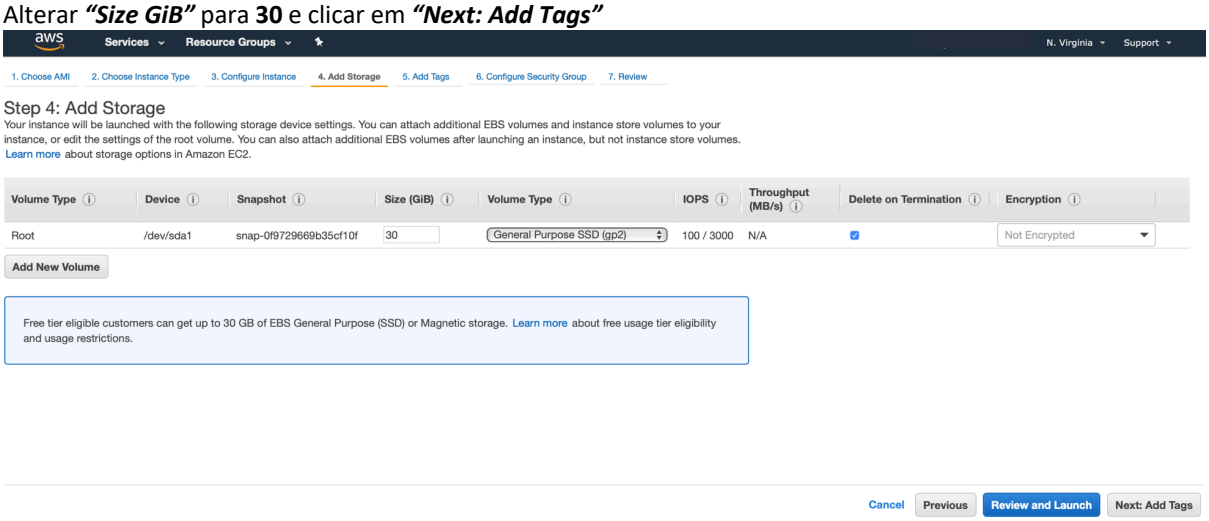


6. Utilizar as configurações padrões e clicar em ***"Next: Add Storage"***



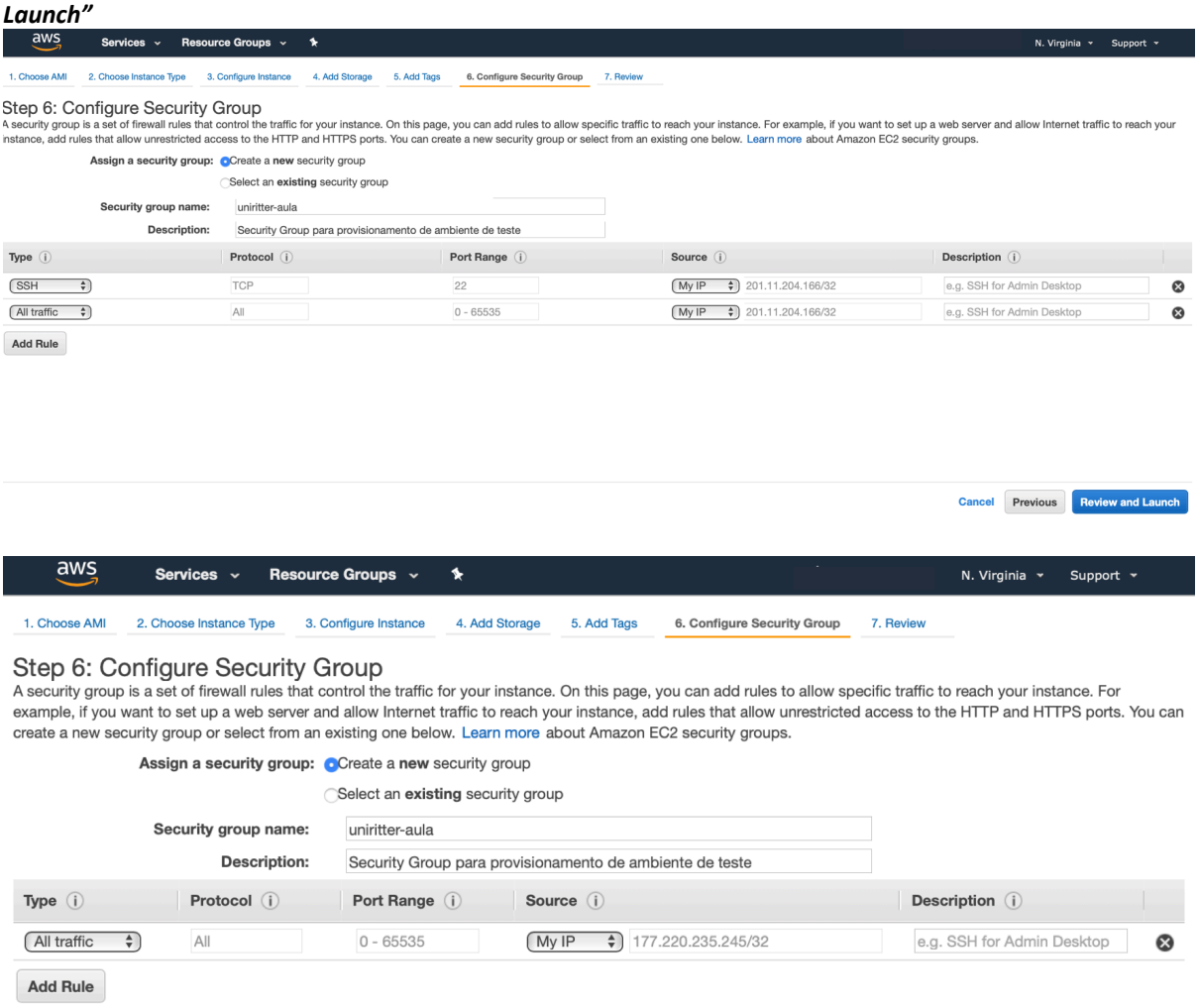
Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

7. Alterar ***"Size GiB"*** para **30** e clicar em ***"Next: Add Tags"***



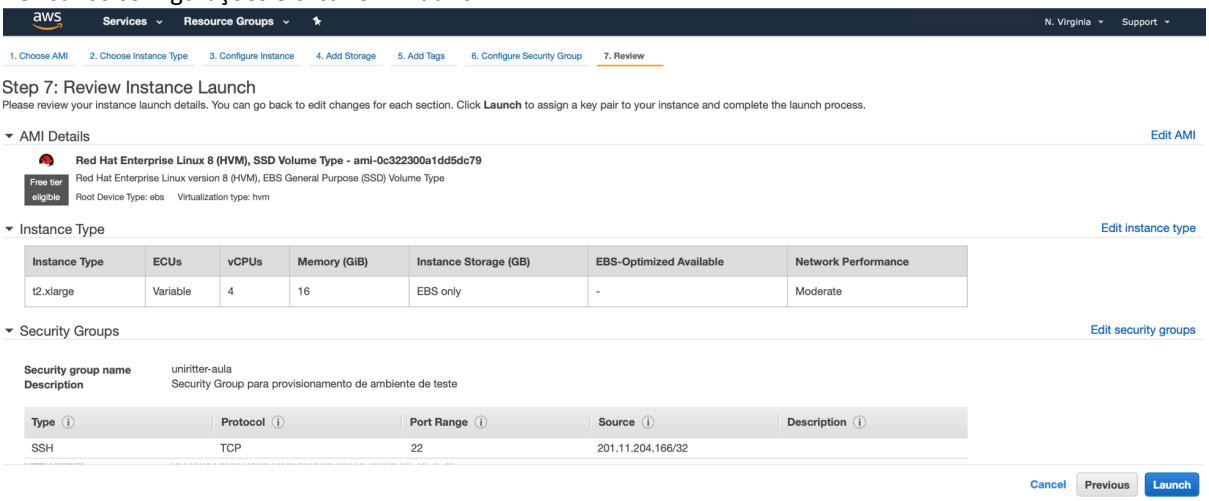
8. Não adicionar tag e clicar em ***"Next: Configure Security Group"***

9. Preencher nome, descrição, liberação para acesso externo da sua máquina local e clicar ***"Review and Launch"***

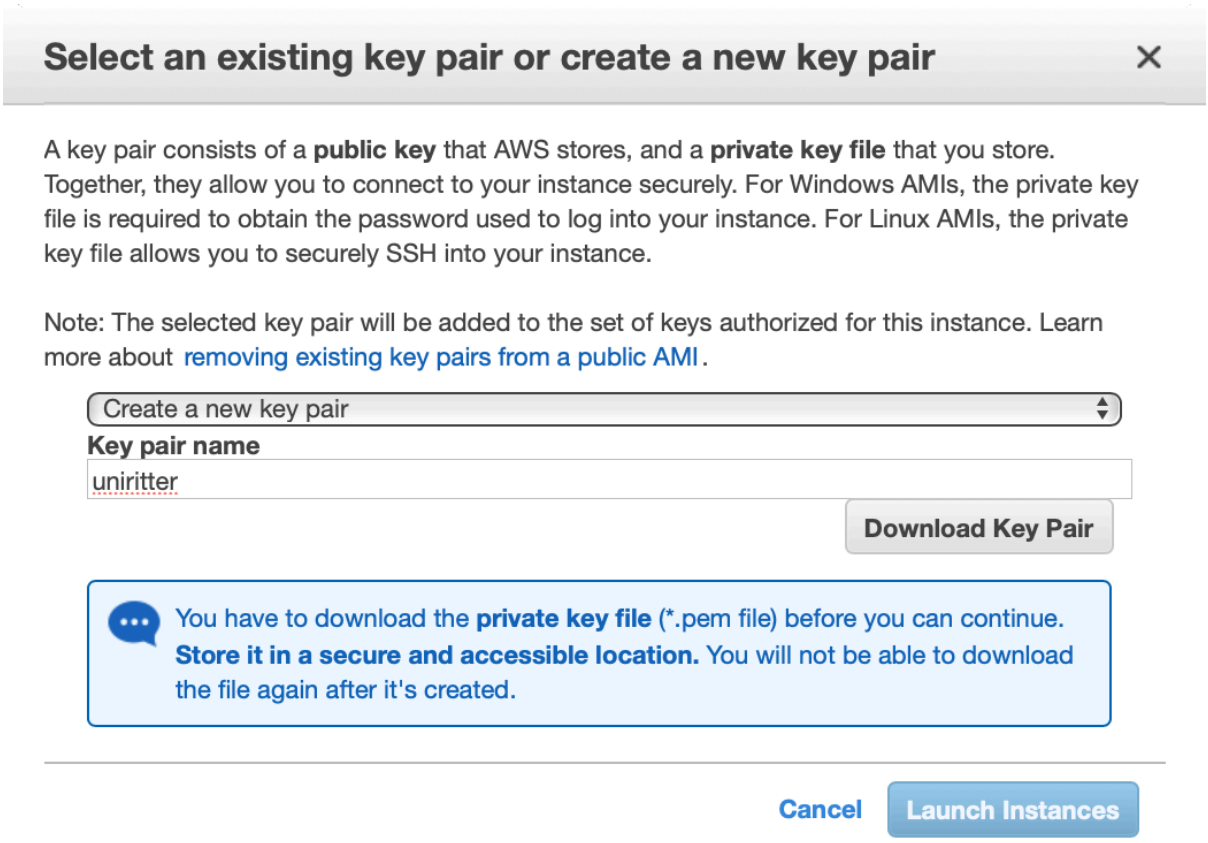


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

10. Revisar as configurações e clicar em ***"Launch"***

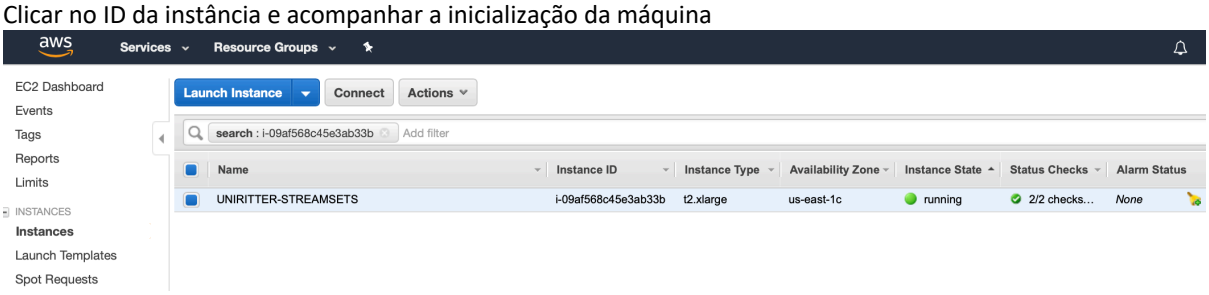


11. Criar uma chave ou utilizar uma existente
    - Informar nome
    - Realizar download da chave
    - Clicar em ***"Launch Instances"***



Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

12. Clicar no ID da instância e acompanhar a inicialização da máquina



13. Ajustar as permissões da chave para acesso via SSH

```
sh-3.2# pwd
/Users/roberto.silva/Documents/keys/uniritter
sh-3.2#
sh-3.2# ls -l
total 8
-rw-r--r--@ 1 roberto.silva  894878024  1692 Oct 16 08:38 uniritter.pem
sh-3.2#
sh-3.2#
sh-3.2# chmod 400 uniritter.pem
sh-3.2#
sh-3.2#
sh-3.2# ls -l
total 8
-r--------@ 1 roberto.silva  894878024  1692 Oct 16 08:38 uniritter.pem
sh-3.2#
```

14. Acessar a instância provisionada

```
sh-3.2#
sh-3.2# ssh -i uniritter.pem ec2-user@34.201.140.253
The authenticity of host '34.201.140.253 (34.201.140.253)' can't be established.
ECDSA key fingerprint is SHA256:iXcwSoKx9LkHchZ6AL4E6VE+vulcOMIu8QByYm+81K8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '34.201.140.253' (ECDSA) to the list of known hosts.
[ec2-user@ip-172-31-46-168 ~]$
```

15. Instalar o Java

```
[ec2-user@ip-172-31-46-168 ~]$
[ec2-user@ip-172-31-46-168 ~]$ sudo su
[root@ip-172-31-46-168 ec2-user]#
[root@ip-172-31-46-168 ec2-user]# yum -y install java-1.8.0-openjdk
Last metadata expiration check: 0:10:20 ago on Wed 16 Oct 2019 11:51:51 AM UTC.
Dependencies resolved.
================================================================================
 Package                                Arch                          Version
================================================================================
Installing:
 java-1.8.0-openjdk                     x86_64                        1:1.8.0.222.b1
Installing dependencies:
 libdatrie                              x86_64                        0.2.9-7.el8
 giflib                                 x86_64                        5.1.4-3.el8
 libXext                                x86_64                        1.3.3-9.el8
 libXrandr                              x86_64                        1.5.1-7.el8
 graphite2                              x86_64                        1.3.10-10.el8
```

Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

16. Verificar a instalação do Java

```
[root@ip-172-31-46-168 ec2-user]# java -version
openjdk version "1.8.0_222"
OpenJDK Runtime Environment (build 1.8.0_222-b10)
OpenJDK 64-Bit Server VM (build 25.222-b10, mixed mode)
[root@ip-172-31-46-168 ec2-user]#
```

17. Instalar o WGET

```
[root@ip-172-31-46-168 ec2-user]#
[root@ip-172-31-46-168 ec2-user]# yum install wget
Last metadata expiration check: 0:33:29 ago on Wed 16 Oct 2019 11:51:51 AM UTC.
Dependencies resolved.
================================================================================
 Package                    Arch                    Version
================================================================================
Installing:
 wget                       x86_64                  1.19.5-7.el8_0.1

Transaction Summary
================================================================================
Install  1 Package

Total download size: 734 k
Installed size: 2.8 M
Is this ok [y/N]: y
Downloading Packages:
wget-1.19.5-7.el8_0.1.x86_64.rpm
--------------------------------------------------------------------------------
Total
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :
  Installing       : wget-1.19.5-7.el8_0.1.x86_64
  Running scriptlet: wget-1.19.5-7.el8_0.1.x86_64
  Verifying        : wget-1.19.5-7.el8_0.1.x86_64

Installed:
  wget-1.19.5-7.el8_0.1.x86_64

Complete!
```

18. Baixar o StreamSets
    Comando: wget https://s3-us-west-2.amazonaws.com/archives.streamsets.com/datacollector/3.11.0/rpm/el7/streamsets-datacollector-3.11.0-el7-all-rpms.tar

```
[root@ip-172-31-46-168 ec2-user]#
[root@ip-172-31-46-168 ec2-user]# wget https://s3-us-west-2.amazonaws.com/archives.streamsets.com/datacollector/3.11.0/rpm/el7/streamsets-datacollector-3.11.0-el7-all-rpms.tar
--2019-10-16 12:26:26--  https://s3-us-west-2.amazonaws.com/archives.streamsets.com/datacollector/3.11.0/rpm/el7/streamsets-datacollector-3.11.0-el7-all-rpms.tar
Resolving s3-us-west-2.amazonaws.com (s3-us-west-2.amazonaws.com)... 52.218.220.56
Connecting to s3-us-west-2.amazonaws.com (s3-us-west-2.amazonaws.com)|52.218.220.56|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5544171520 (5.2G) [application/x-tar]
Saving to: 'streamsets-datacollector-3.11.0-el7-all-rpms.tar'

streamsets-datacollector-3.11.0-el7-all-rpms.tar            7%[========>
```

19. Descompactar o StreamSets
    Comando: tar -xvf streamsets-datacollector-3.11.0-el7-all-rpms.tar

```
[root@ip-172-31-46-168 ec2-user]#
[root@ip-172-31-46-168 ec2-user]#
[root@ip-172-31-46-168 ec2-user]# tar -xvf streamsets-datacollector-3.11.0-el7-all-rpms.tar
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-aerospike-lib-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-apache-kafka_1_0-lib-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-apache-kafka_1_1-lib-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-apache-kafka_2_0-lib-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-apache-kudu_1_3-lib-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-apache-kudu_1_4-lib-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-apache-kudu_1_5-lib-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-apache-kudu_1_6-lib-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-apache-kudu_1_7-lib-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-apache-pulsar_2-lib-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-apache-solr_6_1_0-lib-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-aws-lib-3.11.0-1.noarch.rpm
streamsets-datacollector-3.11.0-el7-all-rpms/streamsets-datacollector-aws-secrets-manager-credentialstore-lib-3
```

20. Instalar o StreamSets
    Comando: yum localinstall streamsets*.rpm

```
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]#
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]#
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]# pwd
/home/ec2-user/streamsets-datacollector-3.11.0-el7-all-rpms
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]# yum localinstall streamsets*.rpm
Last metadata expiration check: 0:44:38 ago on Wed 16 Oct 2019 11:51:51 AM UTC.
Dependencies resolved.
================================================================================================================
 Package                                                                                        Arch
================================================================================================================
Installing:
 streamsets-datacollector                                                                       noarch
 streamsets-datacollector-aerospike-lib                                                         noarch
 streamsets-datacollector-apache-kafka_1_0-lib                                                  noarch
 streamsets-datacollector-apache-kafka_1_1-lib                                                  noarch
 streamsets-datacollector-apache-kafka_2_0-lib                                                  noarch
 streamsets-datacollector-apache-kudu_1_3-lib                                                   noarch
```

21. Alterar o owner do diretório de instalação
    Comando:
    chown -R sdc:sdc /etc/sdc/
    chown -R sdc:sdc /opt/streamsets-datacollector/

```
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]#
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]#
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]#
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]# chown -R sdc:sdc /etc/sdc/
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]# chown -R sdc:sdc /opt/streamsets-datacollector/
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]#
```

Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

22. Iniciar o StreamSets

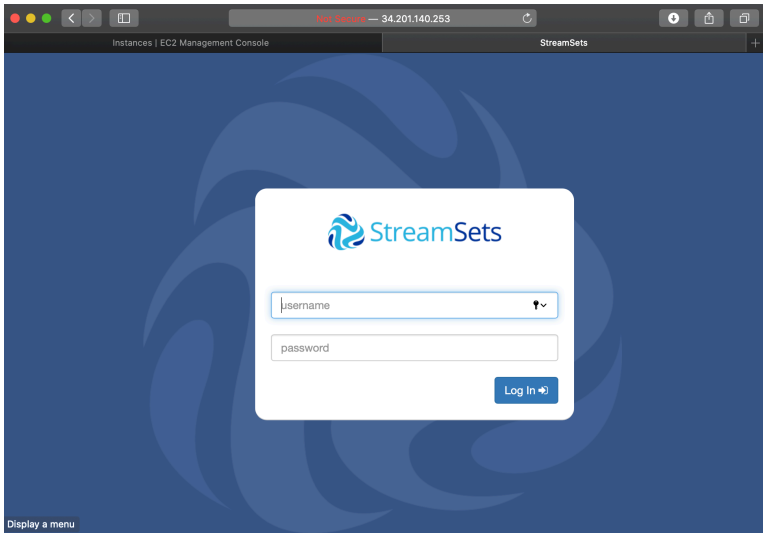
```
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]# systemctl start sdc
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]#
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]# systemctl status sdc
● sdc.service - StreamSets Data Collector (SDC)
   Loaded: loaded (/usr/lib/systemd/system/sdc.service; disabled; vendor preset: disabled)
   Active: active (running) since Wed 2019-10-16 12:43:01 UTC; 12s ago
 Main PID: 14504 (_sdc)
    Tasks: 19 (limit: 26213)
   Memory: 680.5M
   CGroup: /system.slice/sdc.service
           ├─14504 /bin/bash /opt/streamsets-datacollector/libexec/_sdc -verbose
           └─14550 /usr/bin/java -classpath /opt/streamsets-datacollector/libexec/bootstrap-libs/main/streamsets-datacollector-bootstrap-3.11.0.jar:

Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   BOOTSTRAP_CLASSPATH           : /opt/streamsets-datacollector/libexec/bootstrap-
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   API_CLASSPATH                : /opt/streamsets-datacollector/api-lib/*.jar
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   CONTAINER_CLASSPATH           : /etc/sdc:/opt/streamsets-datacollector/container
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   LIBS_COMMON_LIB_DIR           : /opt/streamsets-datacollector/libs-common-lib
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   STREAMSETS_LIBRARIES_DIR      : /opt/streamsets-datacollector/streamsets-libs
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   STREAMSETS_LIBRARIES_EXTRA_DIR : /opt/streamsets-datacollector/streamsets-libs-ex
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   USER_LIBRARIES_DIR            : /opt/streamsets-datacollector/user-libs
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   JAVA OPTS                    : -Djava.security.manager -Djava.security.policy=f
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   MAIN CLASS                   : com.streamsets.datacollector.main.DataCollectorM
Oct 16 12:43:03 ip-172-31-46-168.ec2.internal streamsets[14504]: Logging initialized @1390ms to org.eclipse.jetty.util.log.Slf4jLog
lines 1-20/20 (END)
```

Aguardar o serviço subir.

```
[root@ip-172-31-46-168 streamsets-datacollector-3.11.0-el7-all-rpms]# systemctl status sdc
● sdc.service - StreamSets Data Collector (SDC)
   Loaded: loaded (/usr/lib/systemd/system/sdc.service; disabled; vendor preset: disabled)
   Active: active (running) since Wed 2019-10-16 12:43:01 UTC; 1min 4s ago
 Main PID: 14504 (_sdc)
    Tasks: 33 (limit: 26213)
   Memory: 926.0M
   CGroup: /system.slice/sdc.service
           ├─14504 /bin/bash /opt/streamsets-datacollector/libexec/_sdc -verbose
           └─14550 /usr/bin/java -classpath /opt/streamsets-datacollector/libexec/bootstrap-libs/main/streamsets-datacollector-bootstrap-3.

Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   API_CLASSPATH                : /opt/streamsets-datacollector/api-lib/*
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   CONTAINER_CLASSPATH           : /etc/sdc:/opt/streamsets-datacollector/
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   LIBS_COMMON_LIB_DIR           : /opt/streamsets-datacollector/libs-comm
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   STREAMSETS_LIBRARIES_DIR      : /opt/streamsets-datacollector/streamset
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   STREAMSETS_LIBRARIES_EXTRA_DIR : /opt/streamsets-datacollector/streamset
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   USER_LIBRARIES_DIR            : /opt/streamsets-datacollector/user-libs
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   JAVA OPTS                    : -Djava.security.manager -Djava.security
Oct 16 12:43:01 ip-172-31-46-168.ec2.internal streamsets[14504]:   MAIN CLASS                   : com.streamsets.datacollector.main.DataC
Oct 16 12:43:03 ip-172-31-46-168.ec2.internal streamsets[14504]: Logging initialized @1390ms to org.eclipse.jetty.util.log.Slf4jLog
Oct 16 12:43:26 ip-172-31-46-168.ec2.internal streamsets[14504]: Running on URI : 'http://ip-172-31-46-168.ec2.internal:18630'
lines 1-20/20 (END)
```

- Acessar a console do StreamSets

  Utilizar o IP publico.

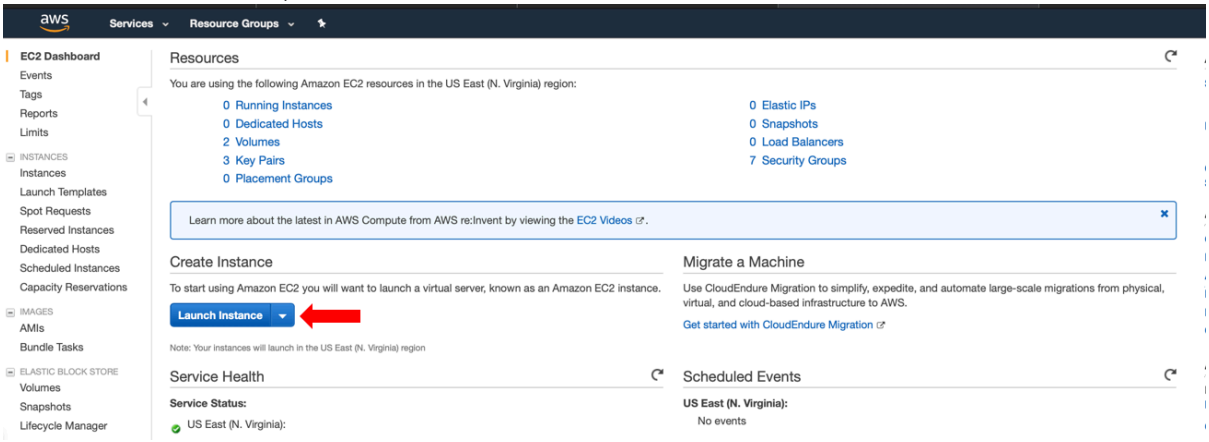


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

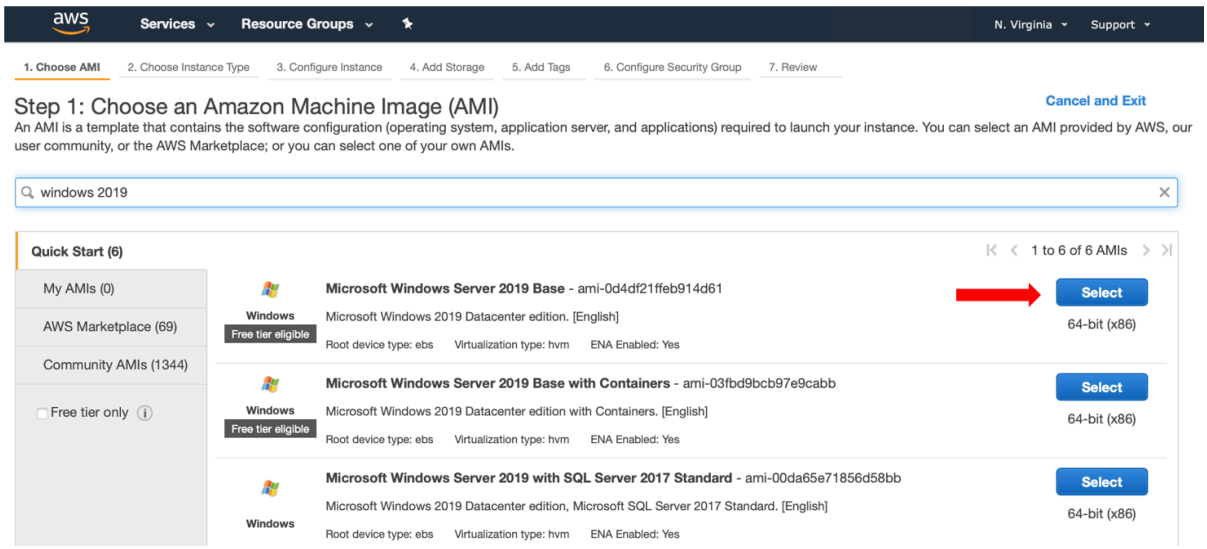
Windows Server 2019 – SQL Server 2017

Provisionamento de instância EC2 (Elastic Computer Cloud):

1. Realizar *login* na console AWS;
2. Acessar *Services > Compute > EC2*;
3. Clicar em *Launch Instance*;

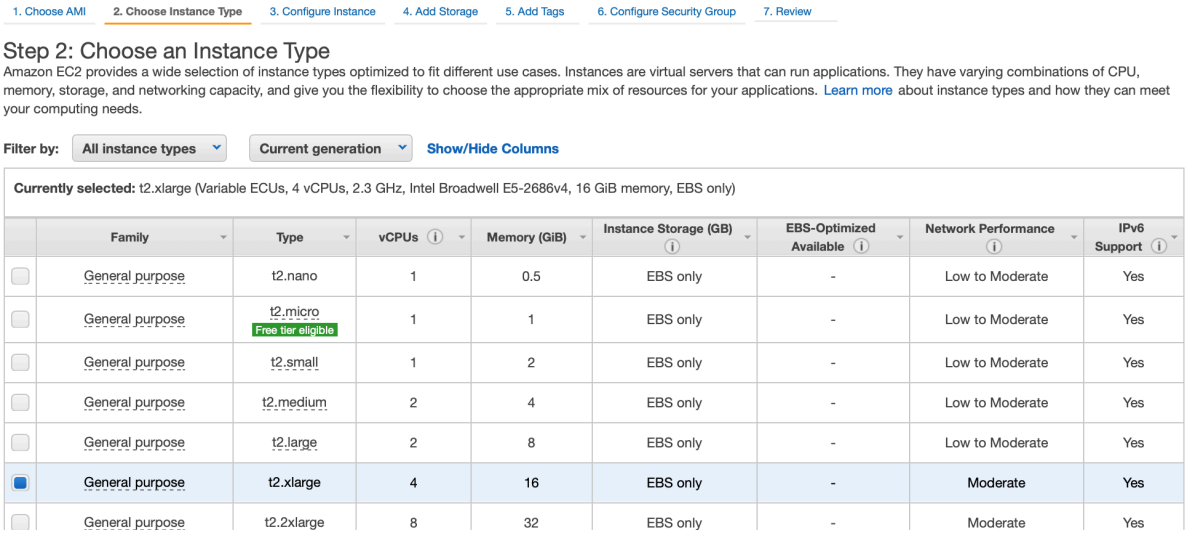


4. Selecionar Microsoft Windows Server 2019

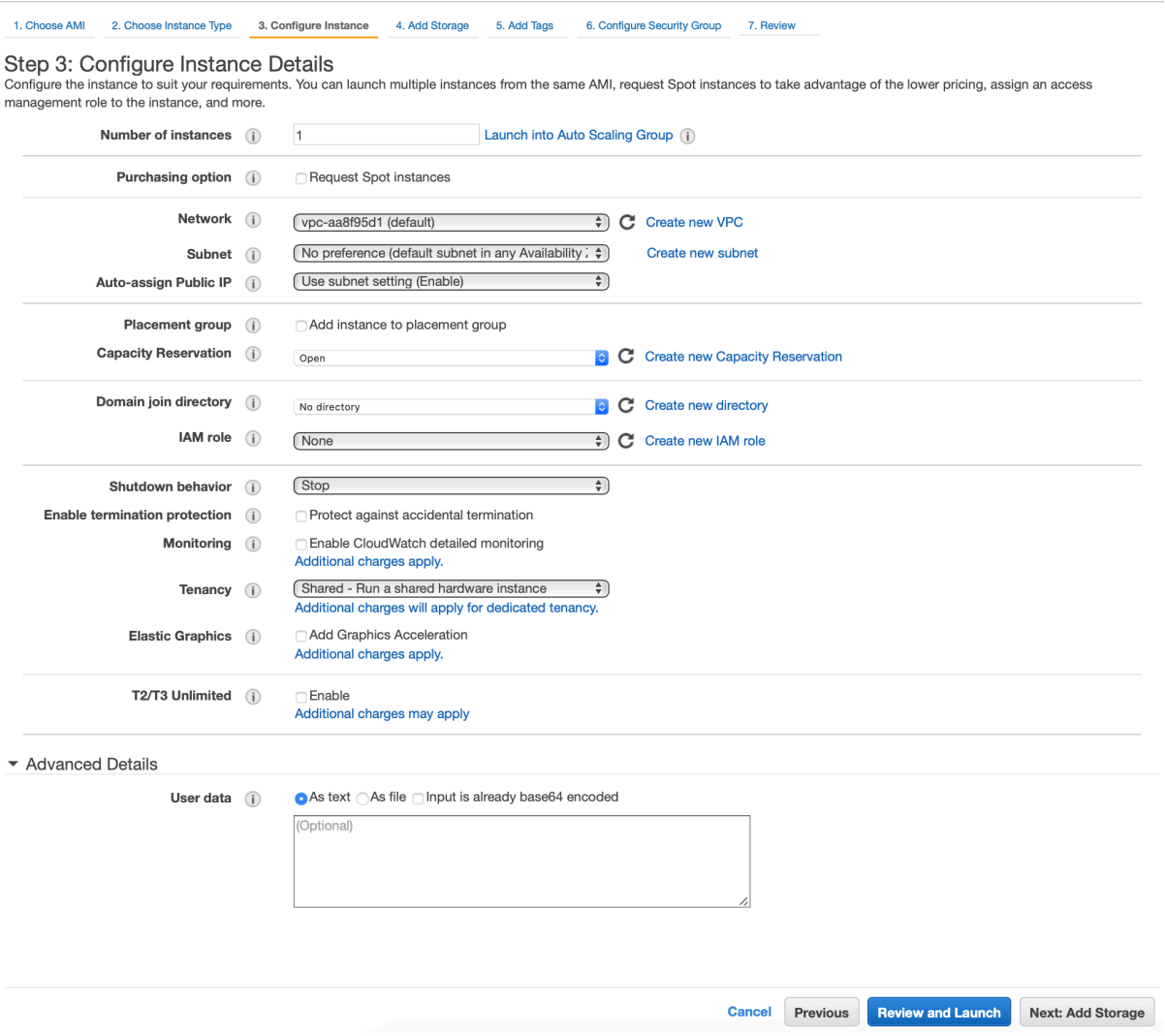


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

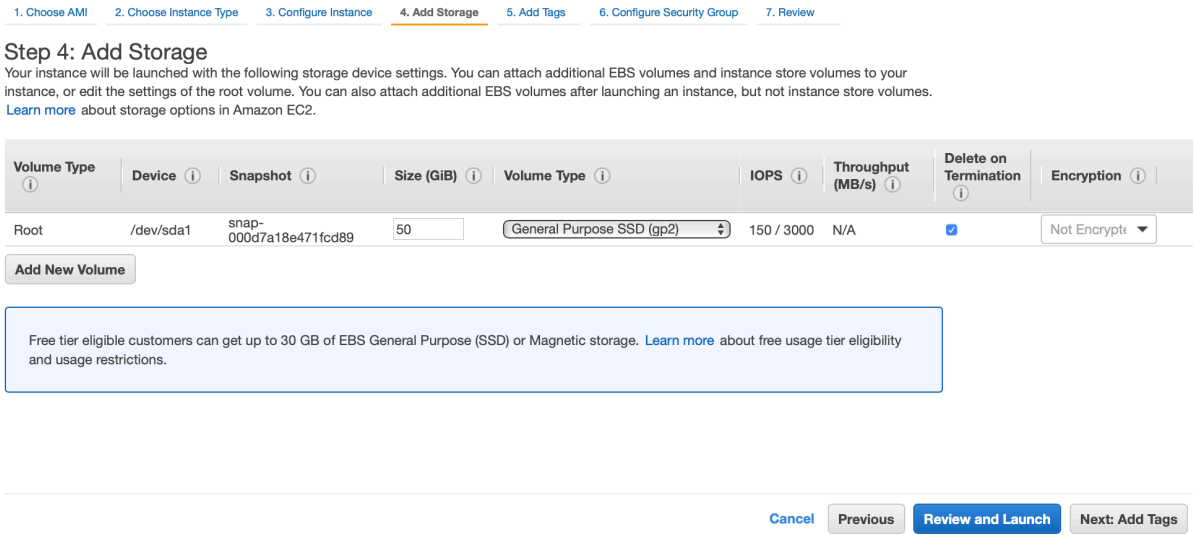
5. Selecionar o tipo de máquina t2.xlarge e clicar em ***"Next: Configure Instance Details"***



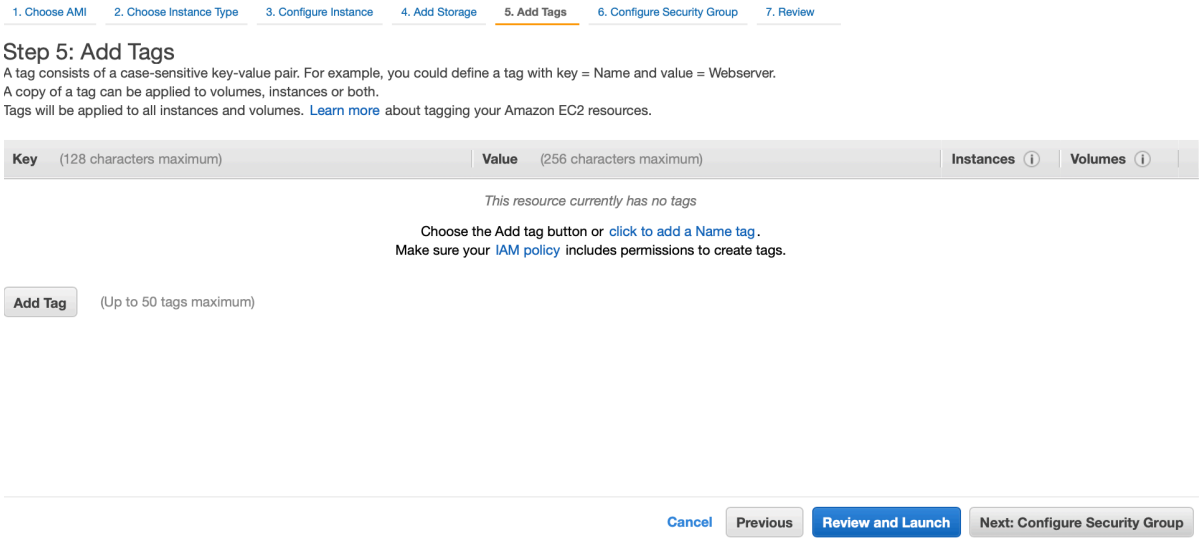
6. Utilizar as configurações padrões e clicar em ***"Next: Add Storage"***



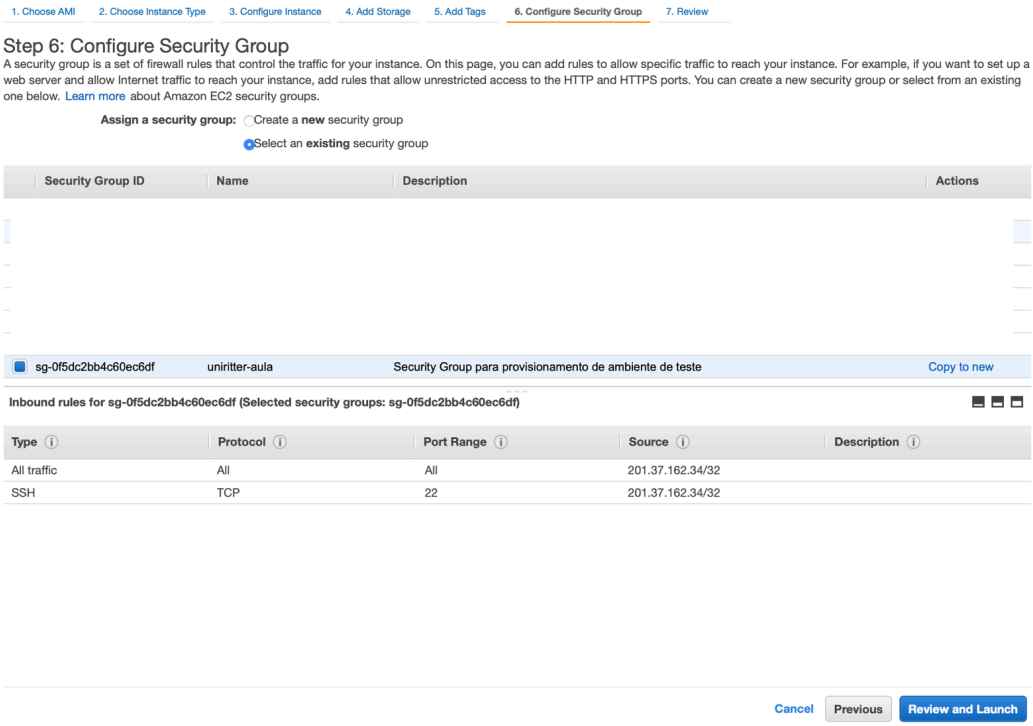
7. Ajustar o tamanho do disco se necessário para volumetria do teste e clicar em ***"Next: Add Tags"***



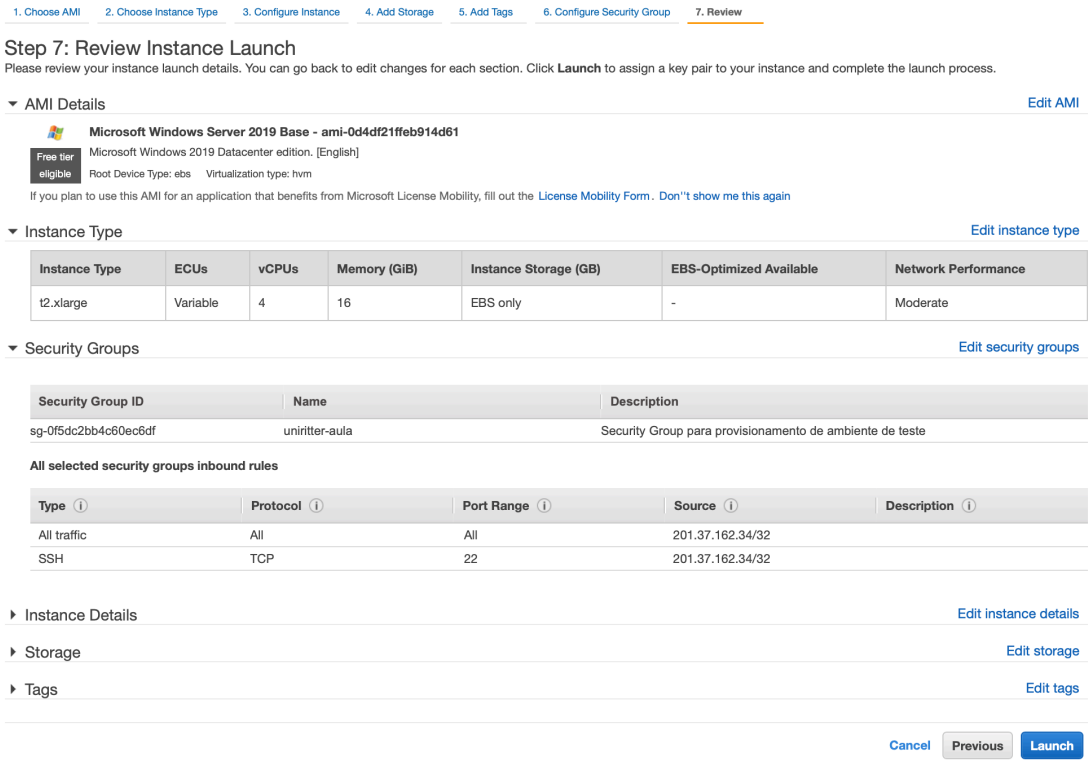
8. Não utilizaremos Tags. Clicar em ***"Next Configure Security Group"***



9. Selecionar o *Security Group* existente que criamos no provisionamento anterior e clicar em ***"Review and Launch"***

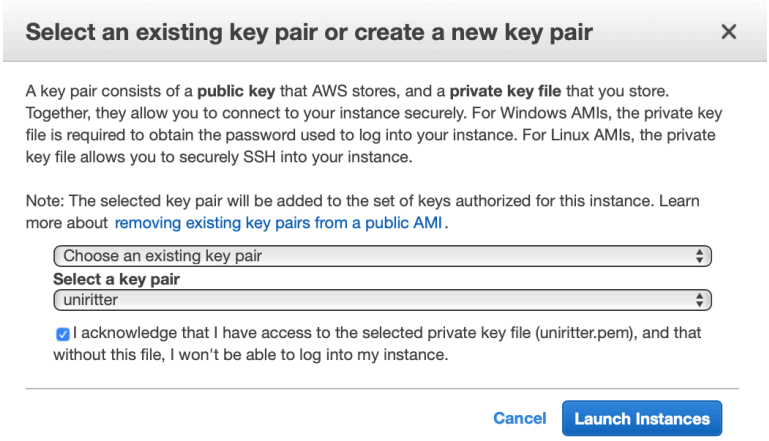


10. Revisar as configurações e clicar em *"Launch"*

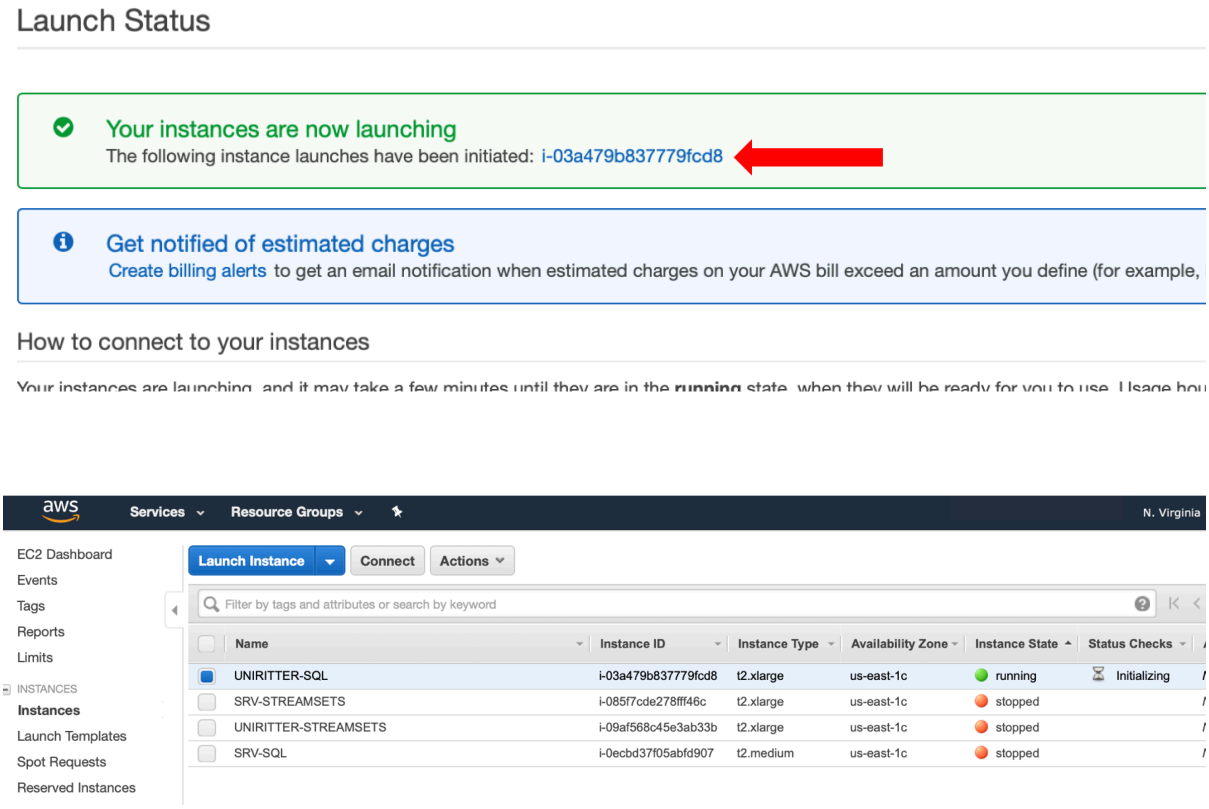


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

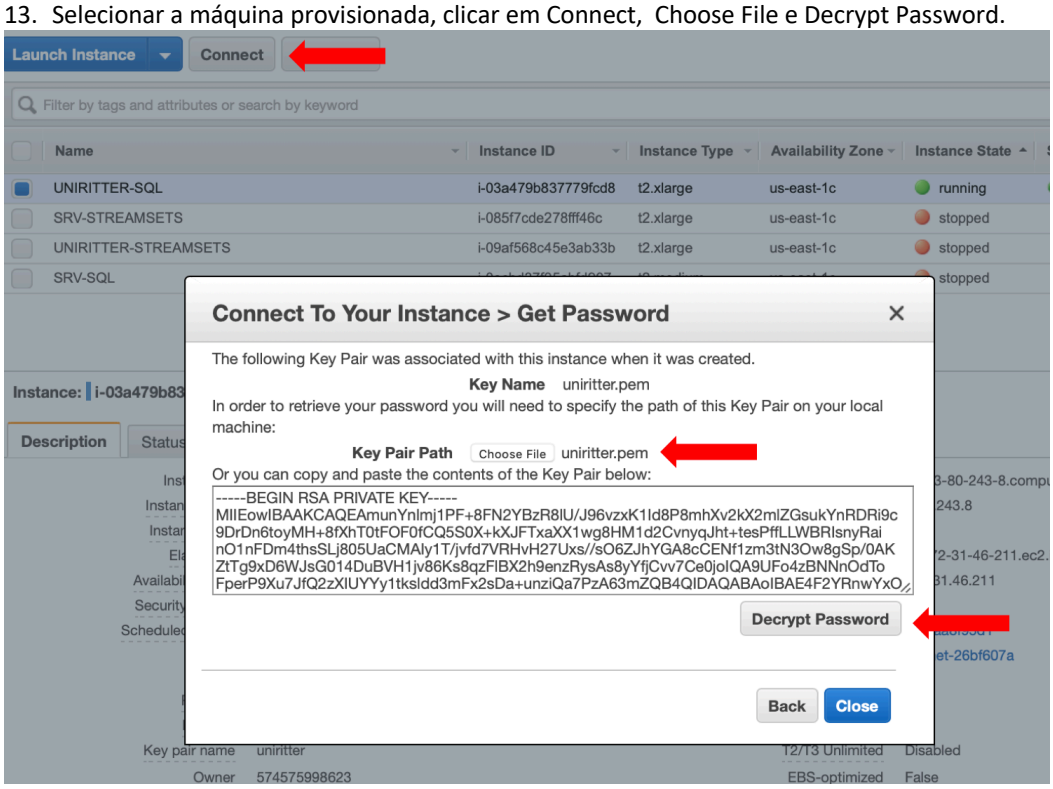
11. Selecionar a chave que criamos no provisionamento anterior e clicar em ***"Launch"***



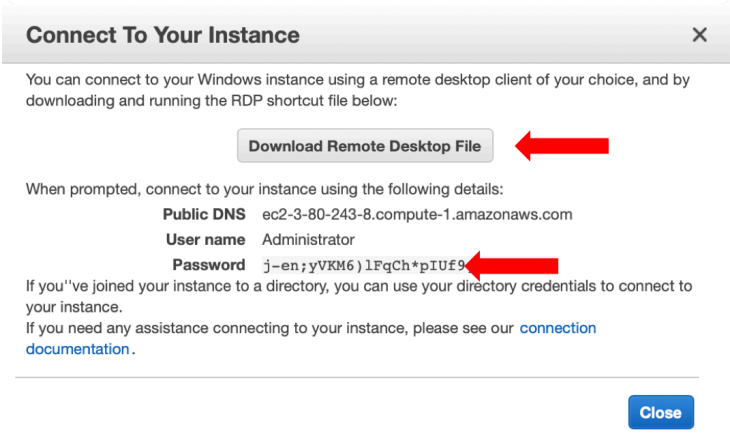
12. Clicar no ID da instância e aguardar o provisionamento da máquina



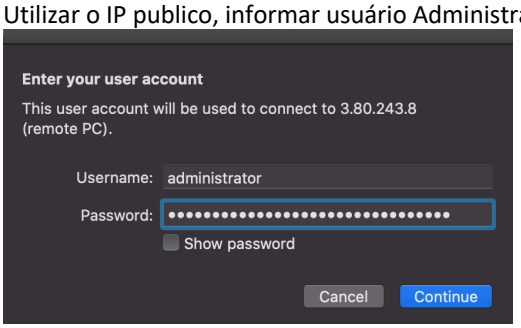
13. Selecionar a máquina provisionada, clicar em Connect, Choose File e Decrypt Password.



14. Copiar o Password e se necessário realize o download do Remote Desktop para acesso remoto ao servidor

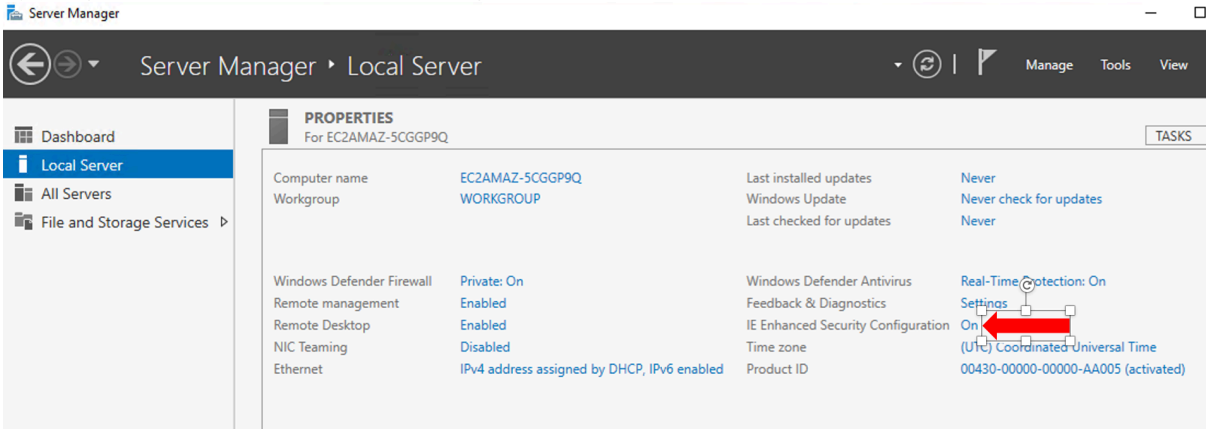


15. Utilizar o IP publico, informar usuário Administrator e o Password copiado anteriormente

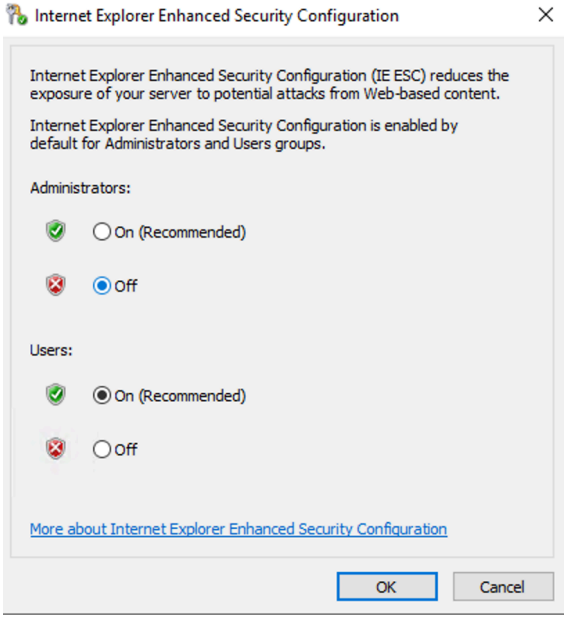


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

16. Após login ao servidor acessar Server Manager, Local Server e clicar em ON no marcado abaixo.



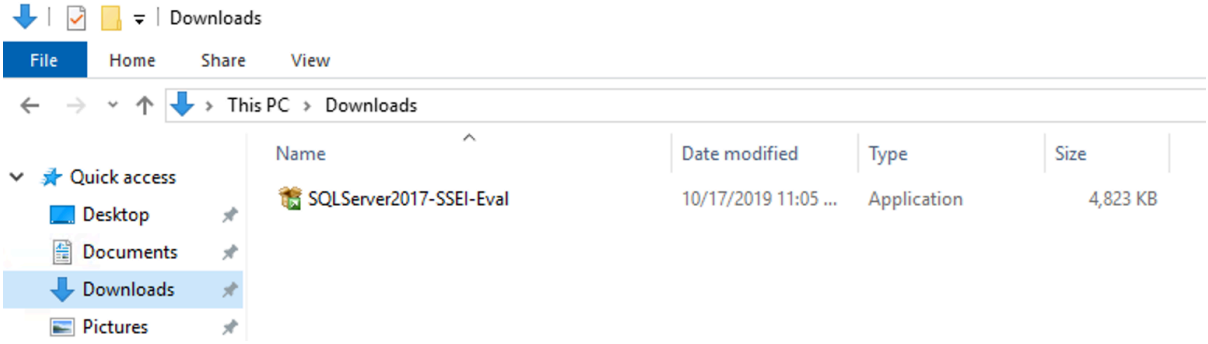
17. Alterar para OFF em Administrators e reiniciar o servidor.



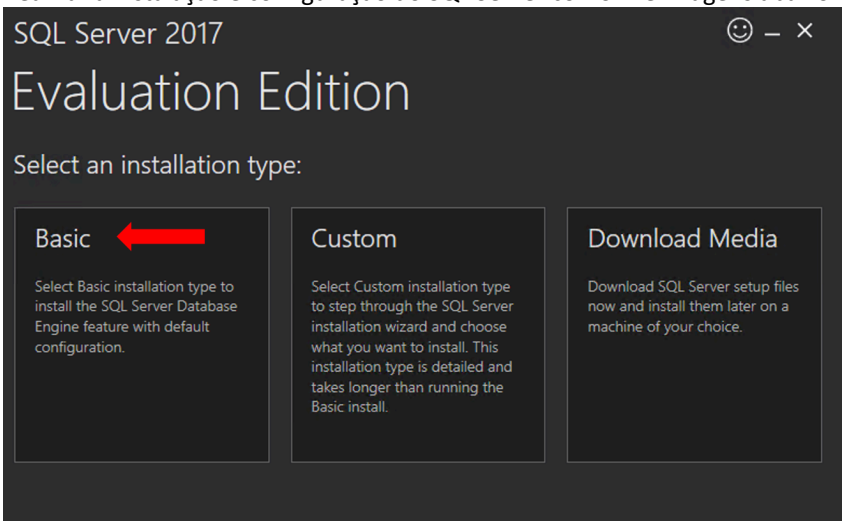
18. Realizar download do SQL Server Trial
Link: https://www.microsoft.com/en-us/sql-server/sql-server-downloads

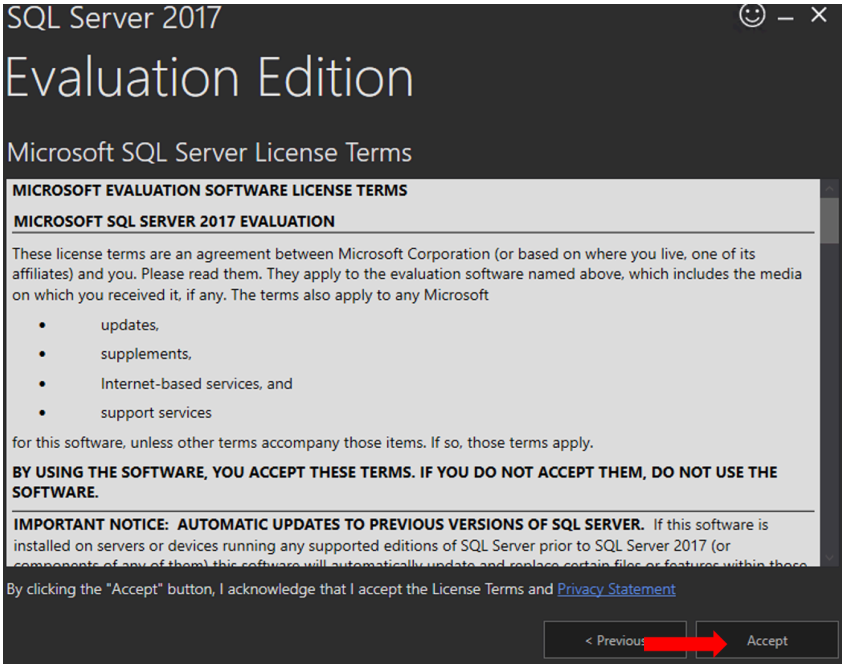
Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

19. Executar o instalador do SQL Server

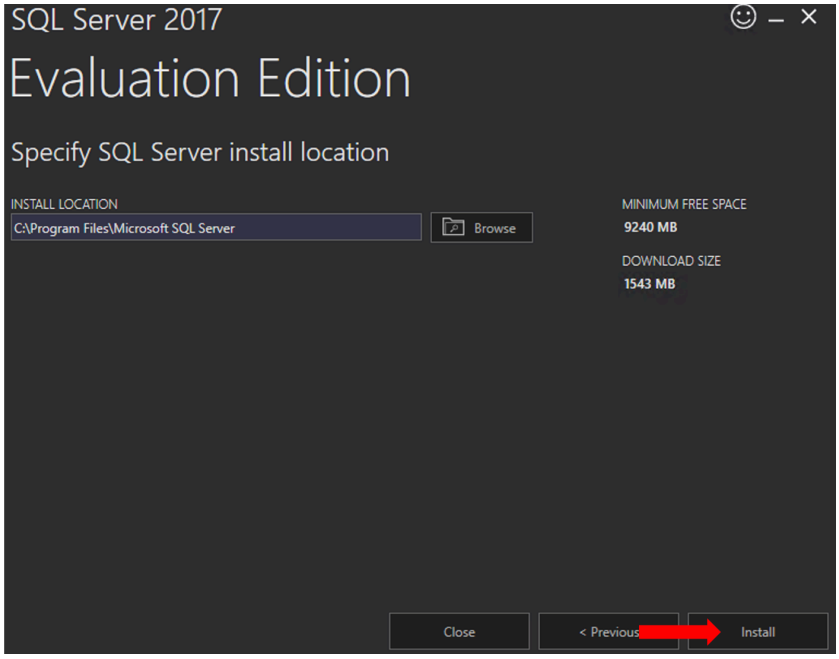


20. Realizar a instalação e configuração do SQL Server conforme imagens abaixo.

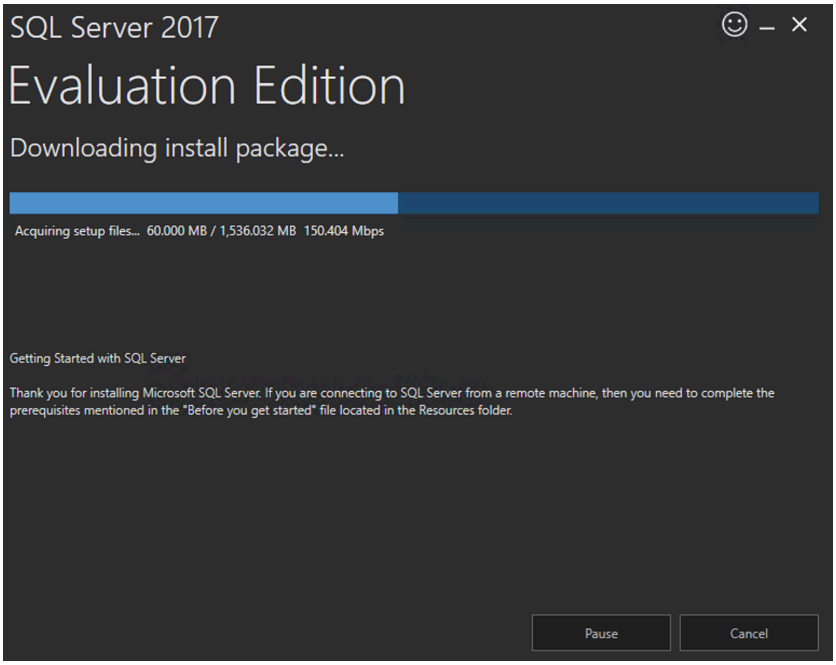




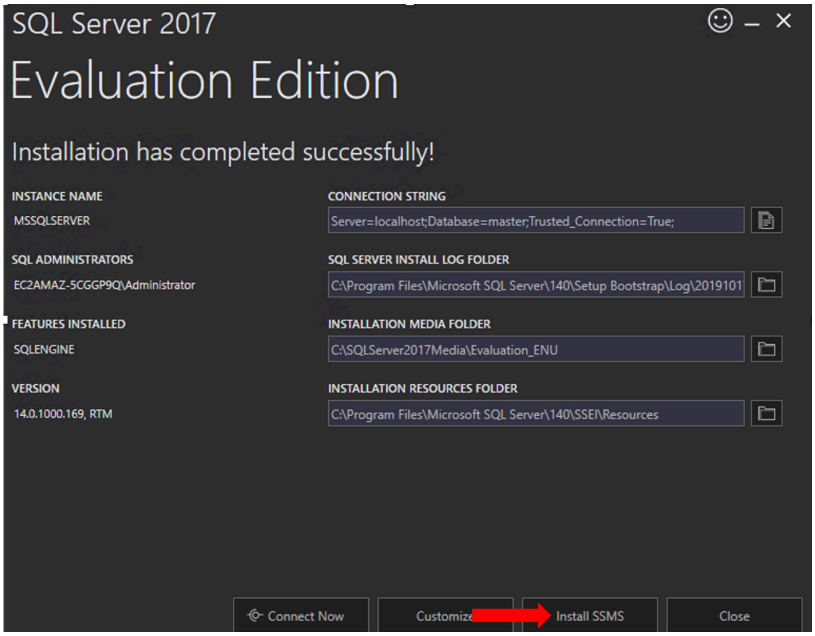
Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

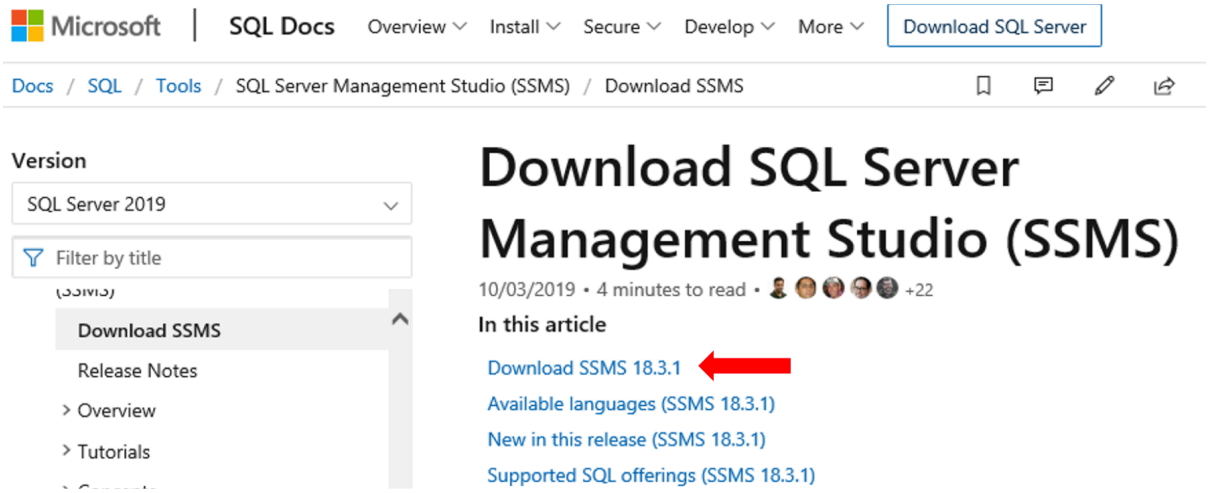


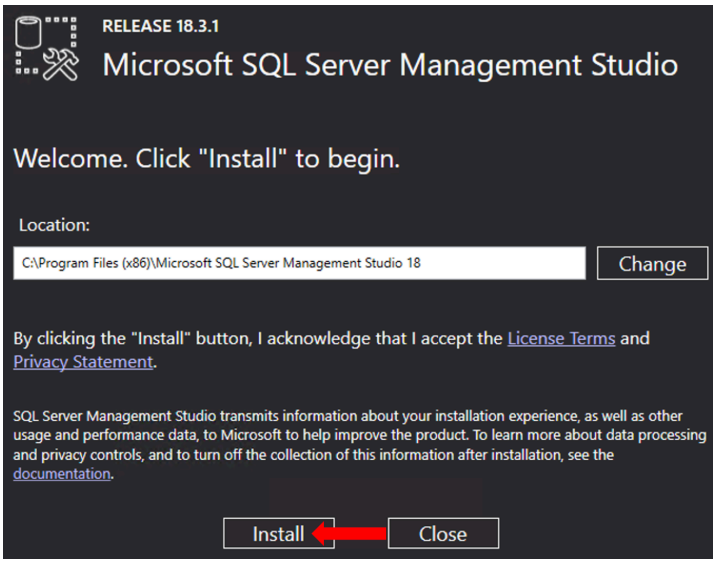
*** Aguardar o download

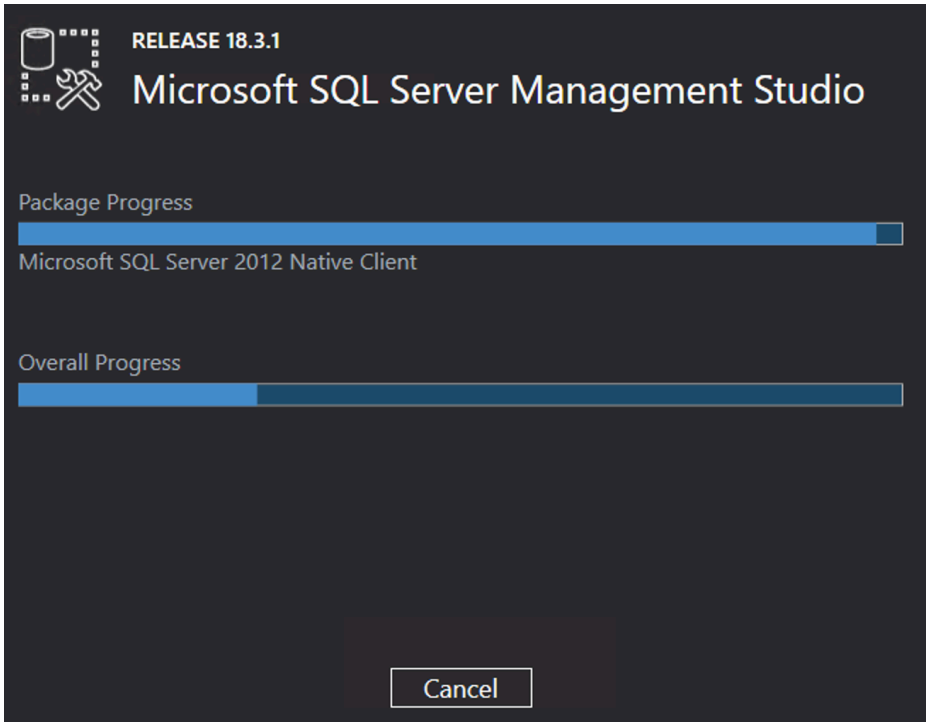


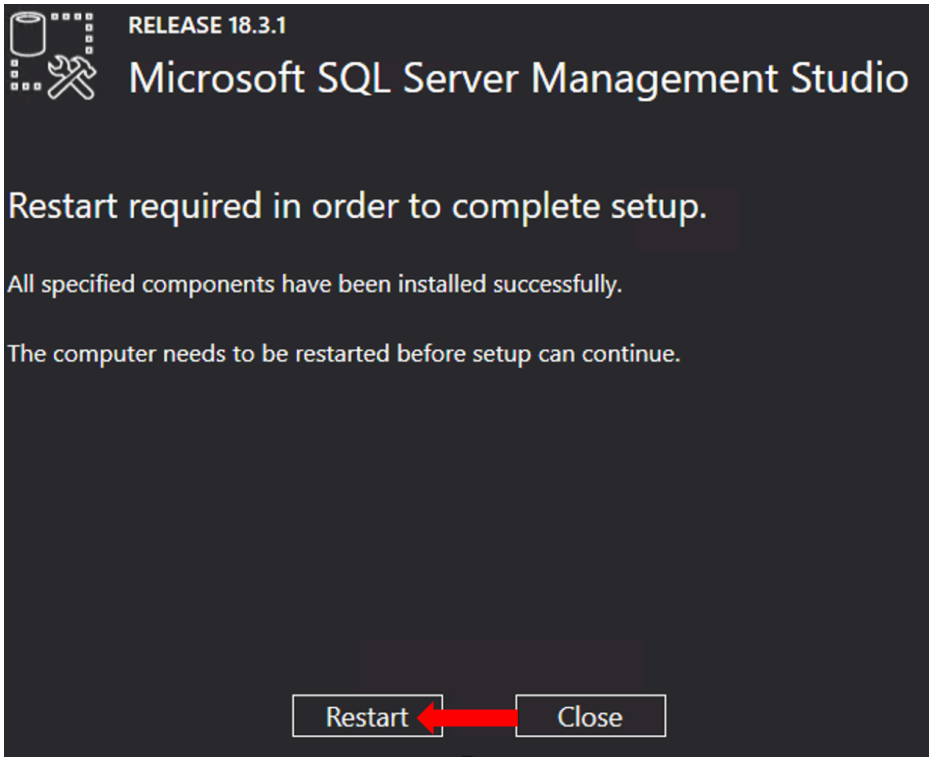
Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão







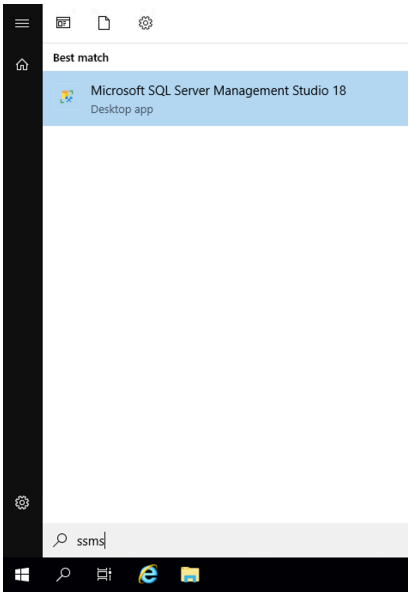




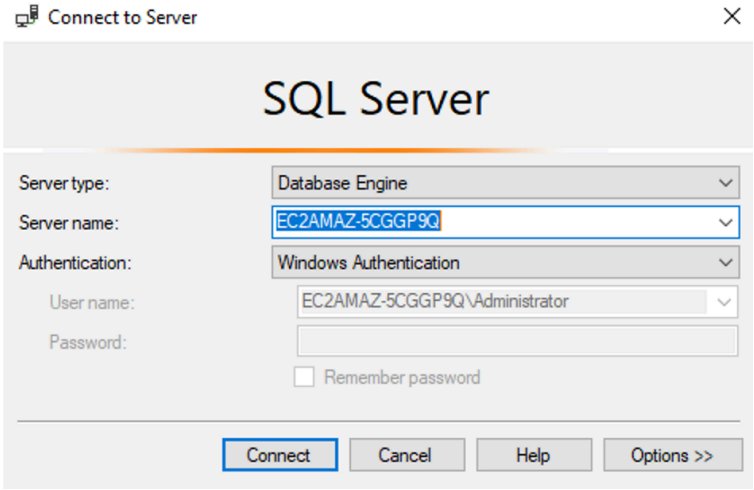
21. Desativar o Firewall do Windows Server.

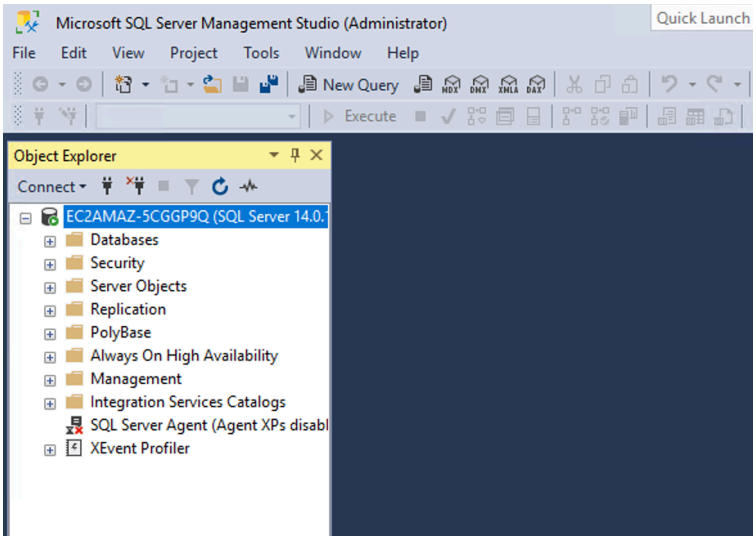
Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

22. Realizar teste de acesso ao SQL Server com SSMS



23. Clicar em Connect



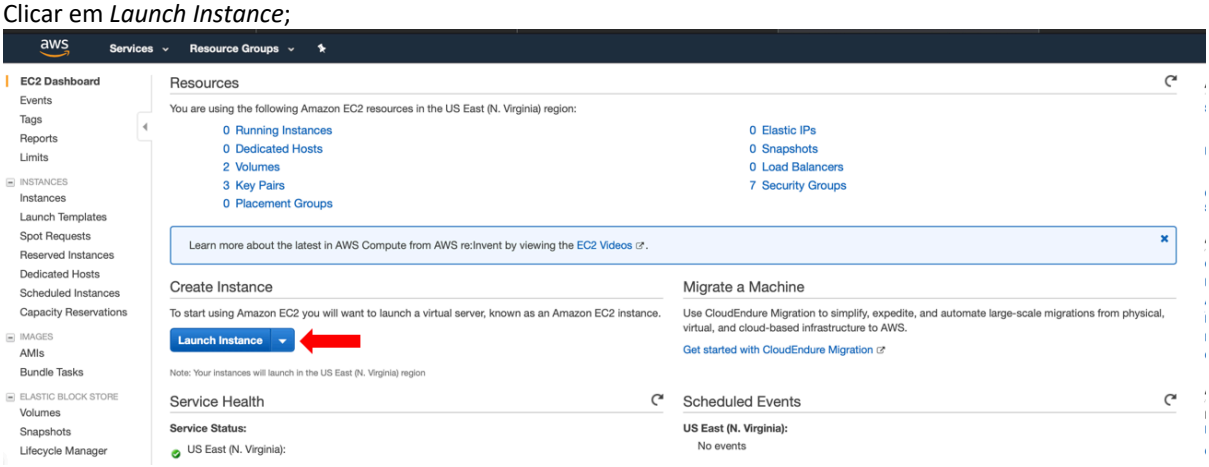


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

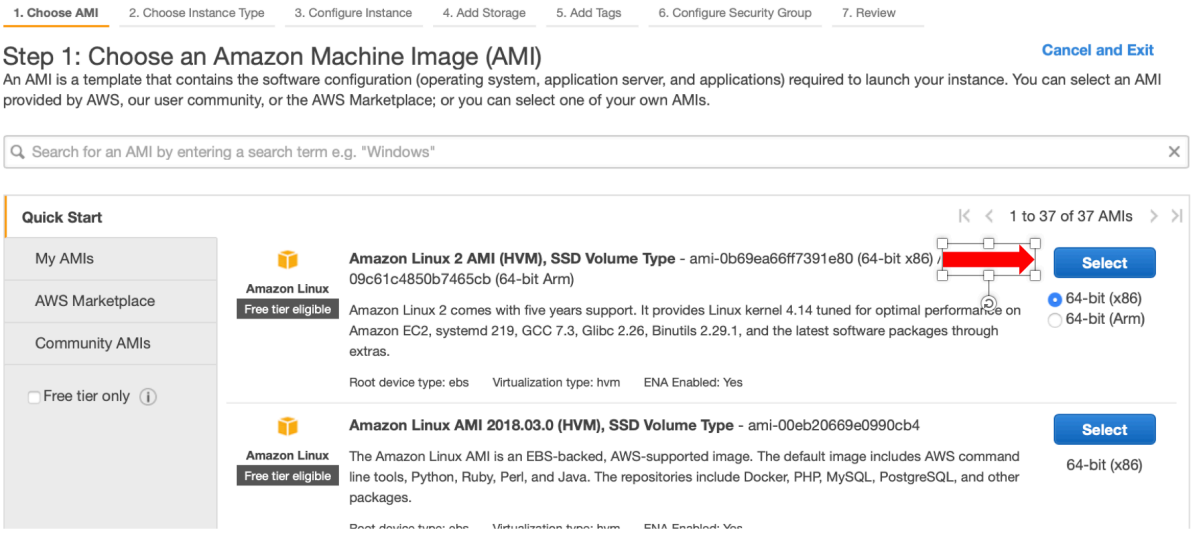
Amazon Linux 2 – Kafka

Provisionamento de instância EC2 (Elastic Computer Cloud):

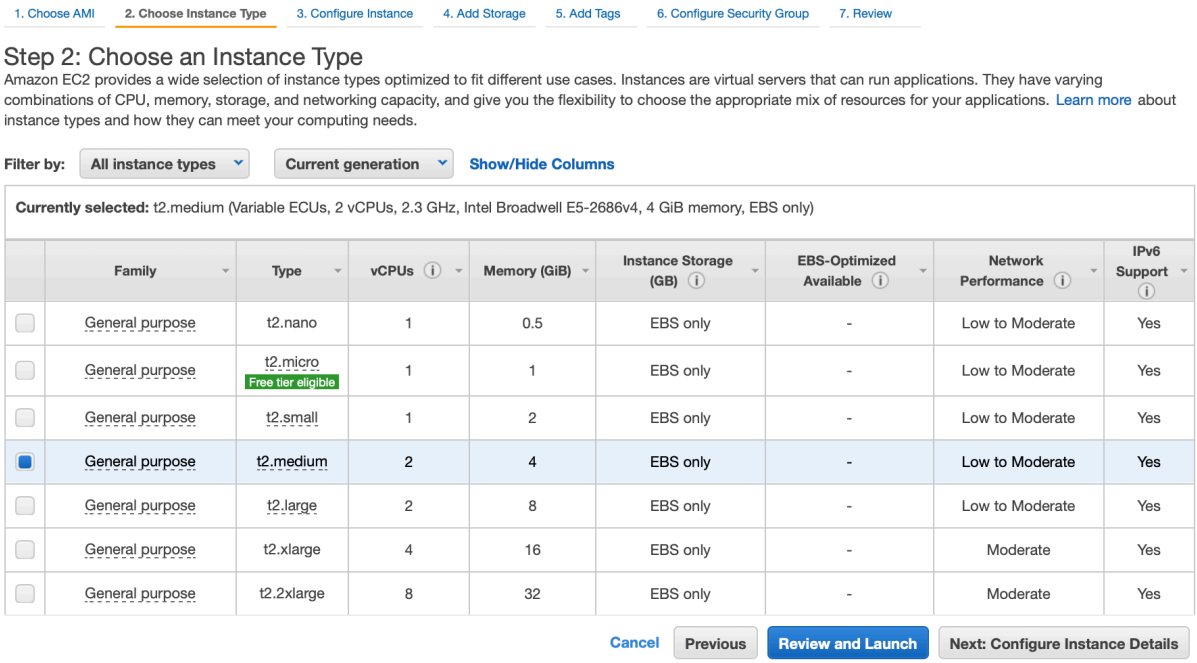
1. Realizar *login* na console AWS;
2. Acessar *Services > Compute > EC2*;
3. Clicar em *Launch Instance*;



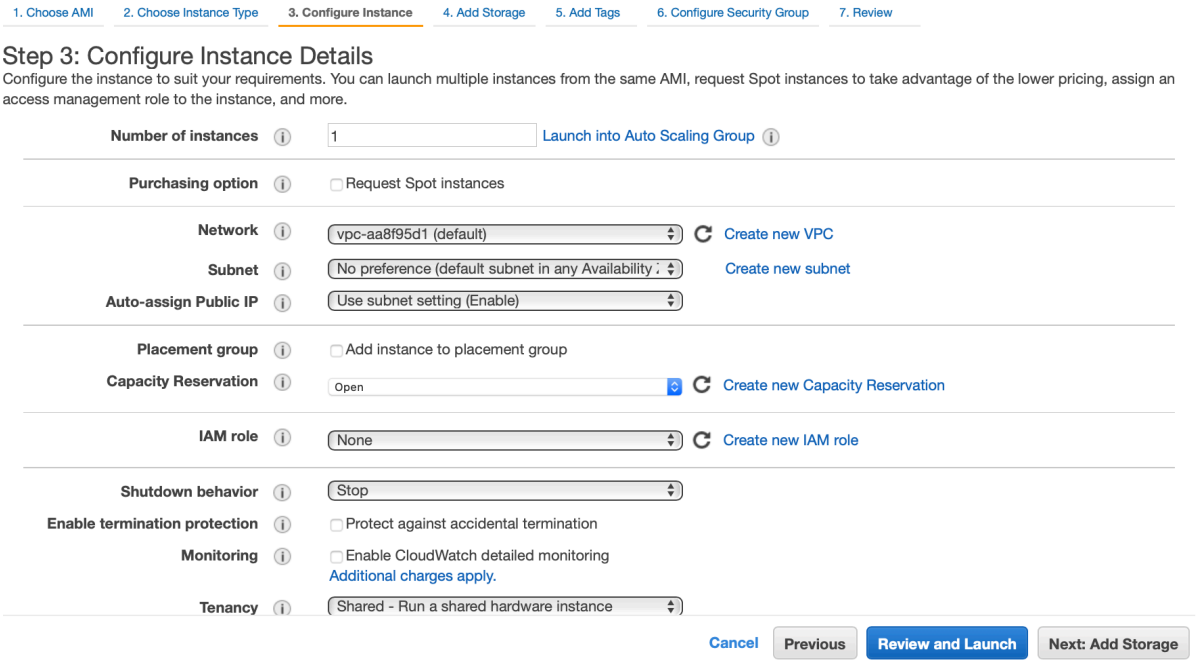
4. Seleciona Amazon Linux 2



5. Selecionar o tipo de máquina t2.medium e clicar em ***"Next: Configure Instance Details"***

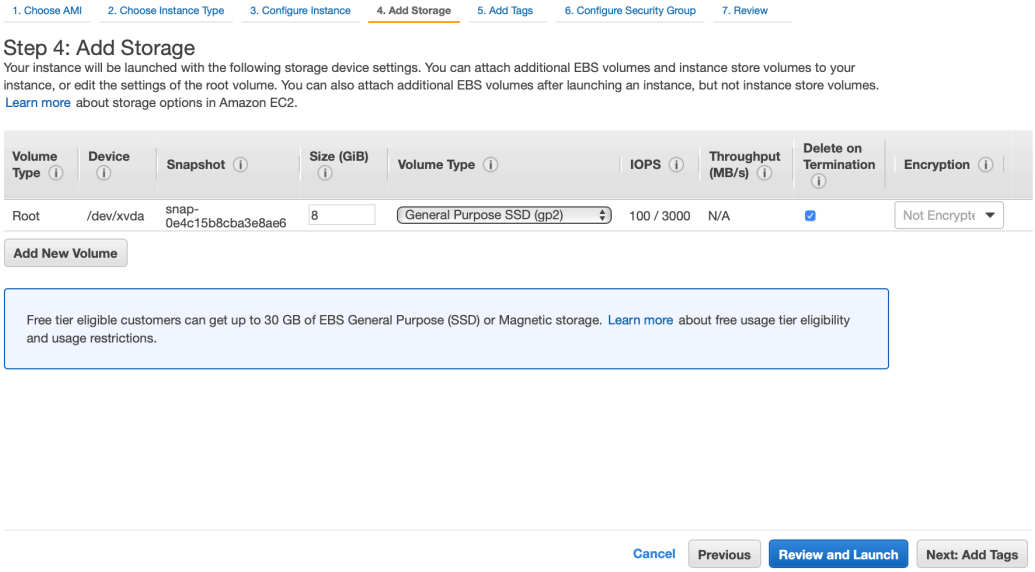


6. Utilizar as configurações padrões e clicar em ***"Next: Add Storage"***

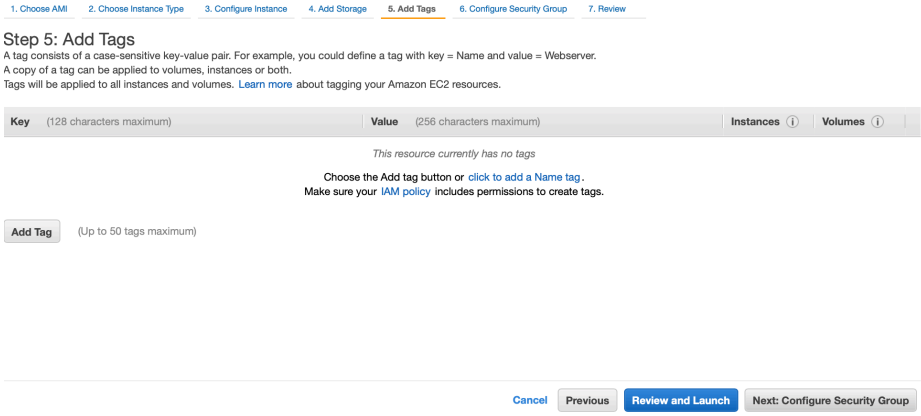


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

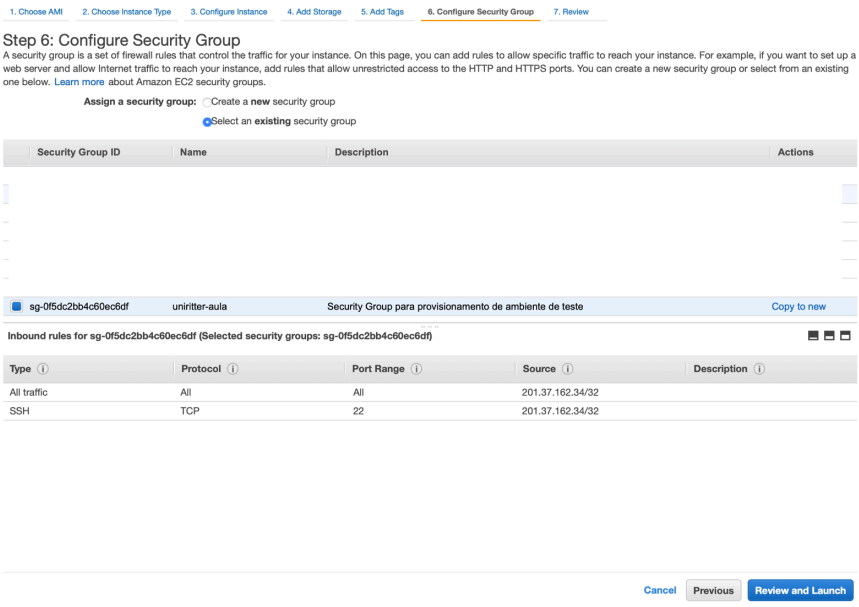
7. Ajustar o tamanho do disco se necessário para volumetria do teste e clicar em ***"Next: Add Tags"***



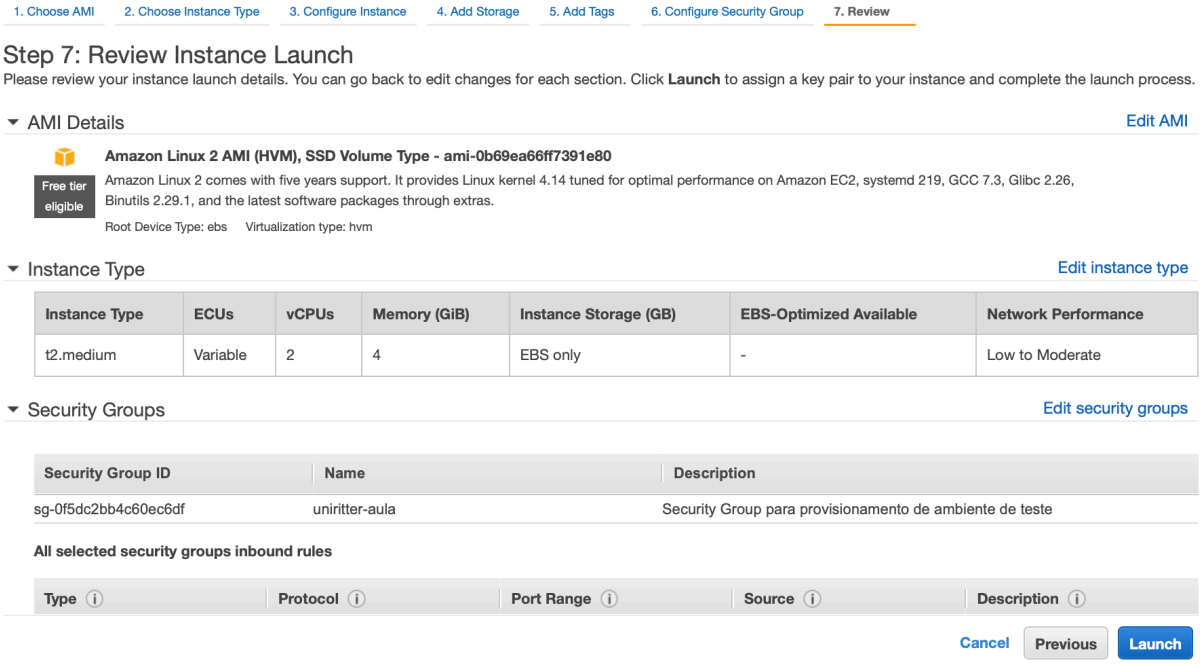
8. Não utilizaremos Tags. Clicar em ***"Next Configure Security Group"***



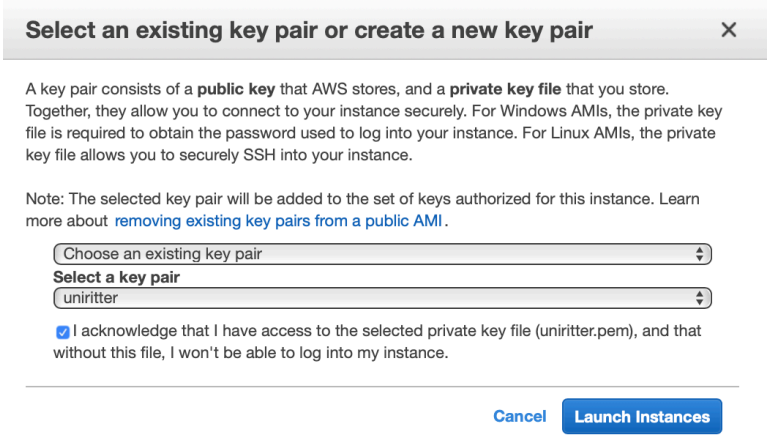
9. Selecionar o *Security Group* existente que criamos no provisionamento anterior e clicar em ***"Review and Launch"***



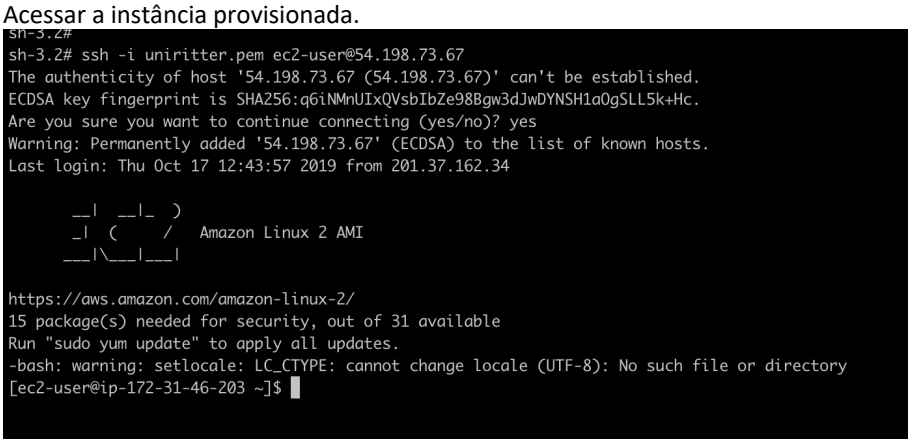
10. Revisar as configurações e clicar em *"Launch"*



11. Selecionar a chave que criamos no provisionamento anterior e clicar em ***"Launch"***



12. Acessar a instância provisionada.



Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

13. Instalar o Java 8
    Comando: sudo yum -y install java-1.8.0-openjdk

```
[ec2-user@ip-172-31-46-203 ~]$ sudo yum -y install java-1.8.0-openjdk
Failed to set locale, defaulting to C
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package java-1.8.0-openjdk.x86_64 1:1.8.0.222.b10-0.amzn2.0.1 will be installed
--> Processing Dependency: java-1.8.0-openjdk-headless(x86-64) = 1:1.8.0.222.b10-0.amzn2.0.1 for package: 1:java-1
--> Processing Dependency: xorg-x11-fonts-Type1 for package: 1:java-1.8.0-openjdk-1.8.0.222.b10-0.amzn2.0.1.x86_64
--> Processing Dependency: libpng15.so.15(PNG15_0)(64bit) for package: 1:java-1.8.0-openjdk-1.8.0.222.b10-0.amzn2.
```

14. Realizar download do Kafka
    Comando: wget http://ftp.unicamp.br/pub/apache/kafka/2.3.0/kafka_2.12-2.3.0.tgz

```
[ec2-user@ip-172-31-46-203 ~]$ wget http://ftp.unicamp.br/pub/apache/kafka/2.3.0/kafka_2.12-2.3.0.tgz
--2019-10-17 12:47:39--  http://ftp.unicamp.br/pub/apache/kafka/2.3.0/kafka_2.12-2.3.0.tgz
Resolving ftp.unicamp.br (ftp.unicamp.br)... 143.106.10.149
Connecting to ftp.unicamp.br (ftp.unicamp.br)|143.106.10.149|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 57215197 (55M) [application/x-gzip]
Saving to: 'kafka_2.12-2.3.0.tgz'

100%[==========================================================================================================

2019-10-17 12:47:45 (9.64 MB/s) - 'kafka_2.12-2.3.0.tgz' saved [57215197/57215197]

[ec2-user@ip-172-31-46-203 ~]$
```

15. Descompactar o Kafka
    Comando: tar -xzf kafka_2.12-2.3.0.tgz

```
[ec2-user@ip-172-31-46-203 ~]$ tar -xzf kafka_2.12-2.3.0.tgz
[ec2-user@ip-172-31-46-203 ~]$ ls -l
total 55876
drwxr-xr-x 6 ec2-user ec2-user       89 Jun 19 20:44 kafka_2.12-2.3.0
-rw-rw-r-- 1 ec2-user ec2-user 57215197 Jun 25 00:38 kafka_2.12-2.3.0.tgz
[ec2-user@ip-172-31-46-203 ~]$
```

16. Iniciar o Zookeeper
    Configurar variável de ambiente para 50% da memoria provisionada no servidor.
    Comando: vi .bashrc
    Incluir o comando: export KAFKA_HEAP_OPTS="-Xmx2048M -Xms2048M"

```
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
        . /etc/bashrc
fi
export KAFKA_HEAP_OPTS="-Xmx2048M -Xms2048M"
# Uncomment the following line if you don't like systemctl's auto-paging feature:
# export SYSTEMD_PAGER=

# User specific aliases and functions
~
~
```

    Executar o comando: source .bashrc

    Acessar o diretório do Kafka:

```
[ec2-user@ip-172-31-46-203 ~]$ cd kafka_2.12-2.3.0/
[ec2-user@ip-172-31-46-203 kafka_2.12-2.3.0]$
[ec2-user@ip-172-31-46-203 kafka_2.12-2.3.0]$ pwd
/home/ec2-user/kafka_2.12-2.3.0
[ec2-user@ip-172-31-46-203 kafka_2.12-2.3.0]$
```

Executar o comando: nohup bin/zookeeper-server-start.sh config/zookeeper.properties > ~/zookeeper-logs &

```
[ec2-user@ip-172-31-46-203 kafka_2.12-2.3.0]$ nohup bin/zookeeper-server-start.sh config/zookeeper.properties > ~/zookeeper-logs &
[1] 3936
[ec2-user@ip-172-31-46-203 kafka_2.12-2.3.0]$ nohup: ignoring input and redirecting stderr to stdout

[ec2-user@ip-172-31-46-203 kafka_2.12-2.3.0]$
```

17. Iniciar o Kafka
    Comando: nohup bin/kafka-server-start.sh config/server.properties > ~/kafka-logs &

```
[ec2-user@ip-172-31-46-203 kafka_2.12-2.3.0]$ nohup bin/kafka-server-start.sh config/server.properties > ~/kafka-logs &
[2] 4250
[ec2-user@ip-172-31-46-203 kafka_2.12-2.3.0]$ nohup: ignoring input and redirecting stderr to stdout

[ec2-user@ip-172-31-46-203 kafka_2.12-2.3.0]$
```

Criar tópico de teste:
Comando: bin/kafka-topics.sh --create --zookeeper localhost:2181 --replication-factor 1 --partitions 13 --topic tst-topic

```
[ec2-user@ip-172-31-32-240 kafka_2.12-2.3.0]$ bin/kafka-topics.sh --create --zookeeper localhost:2181 --replication-factor 1 --partitions 13 --topic tst-topic
Created topic tst-topic.
[ec2-user@ip-172-31-32-240 kafka_2.12-2.3.0]$
```

Listar tópico:
Comando: bin/kafka-topics.sh --list --zookeeper localhost:2181

```
[ec2-user@ip-172-31-32-240 kafka_2.12-2.3.0]$ bin/kafka-topics.sh --list --zookeeper localhost:2181
tst-topic
[ec2-user@ip-172-31-32-240 kafka_2.12-2.3.0]$
```

Postar mensagem no tópico de teste:
Comando: bin/kafka-console-producer.sh --broker-list localhost:9092 --topic tst-topic

```
[ec2-user@ip-172-31-32-240 kafka_2.12-2.3.0]$ bin/kafka-console-producer.sh --broker-list localhost:9092 --topic tst-topic
>mensagem-teste
>^C[ec2-user@ip-172-31-32-240 kafka_2.12-2.3.0]$
[ec2-user@ip-172-31-32-240 kafka_2.12-2.3.0]$
```

Verificar mensagem postada anteriormente:
Comando: bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic tst-topic --from-beginning

```
[ec2-user@ip-172-31-32-240 kafka_2.12-2.3.0]$ bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic tst-topic --from-beginning
mensagem-teste
^CProcessed a total of 1 messages
[ec2-user@ip-172-31-32-240 kafka_2.12-2.3.0]$
```

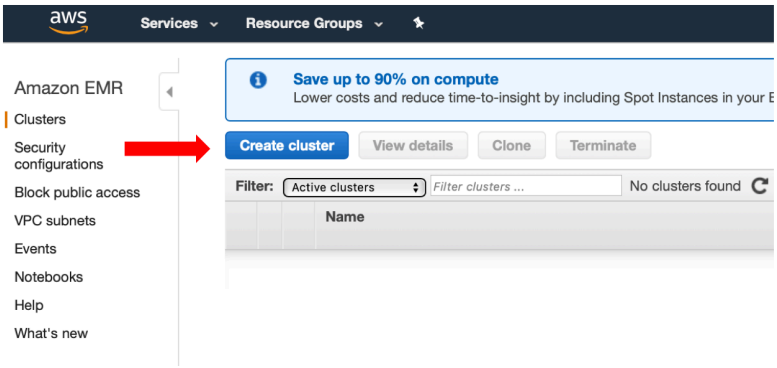
Para Kafka e Zookeeper:
bin/kafka-server-stop.sh
bin/zookeeper-server-stop.sh

Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

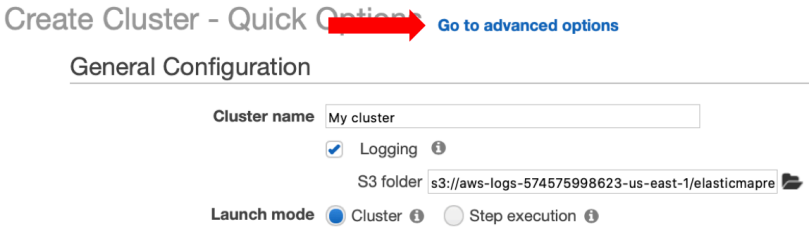
EMR – Elastic MapReduce

Provisionamento de instância EC2 (Elastic Computer Cloud):

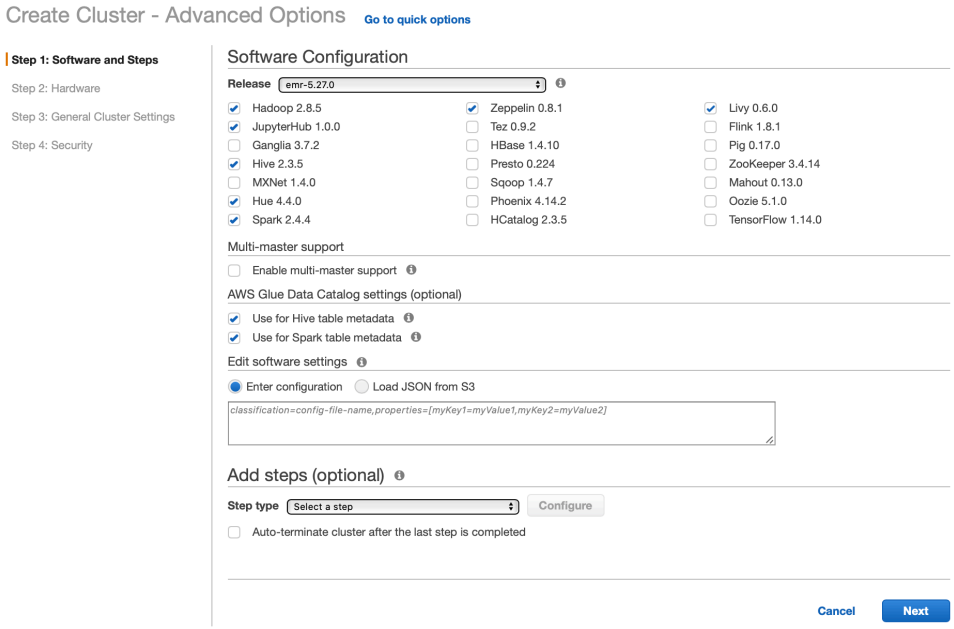
1. Realizar *login* na console AWS;
2. Acessar *Services > EMR*
3. Clicar em “Create Cluster”



4. Clicar em “Go to advanced options”

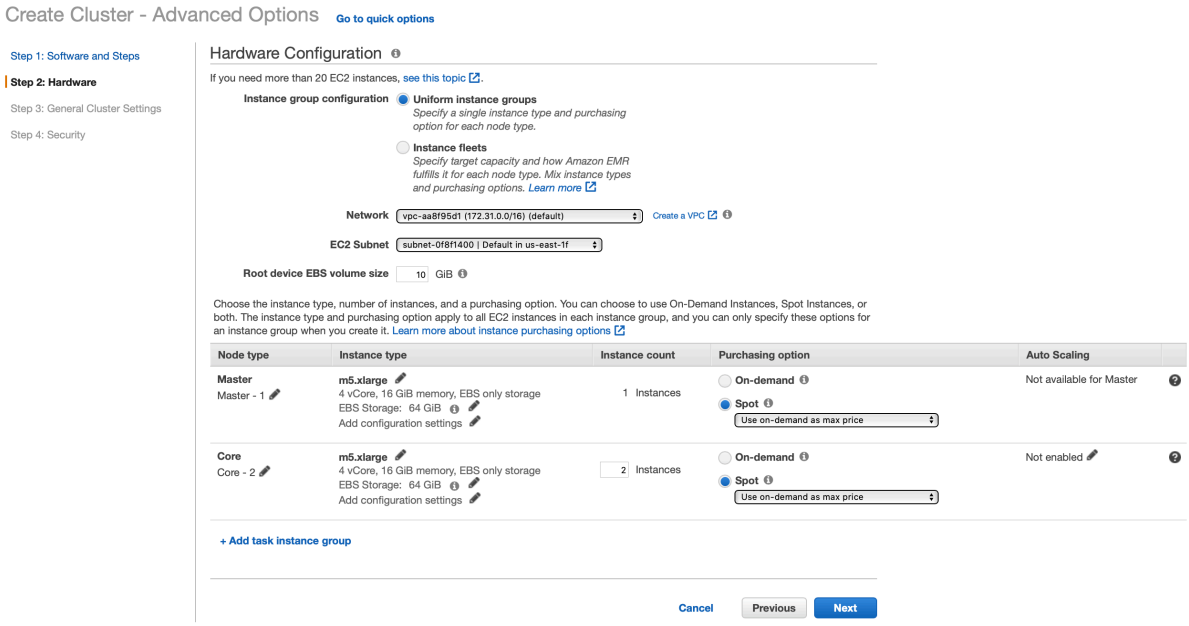


5. Selecionar os componentes conforme abaixo e clicar em “Next”.

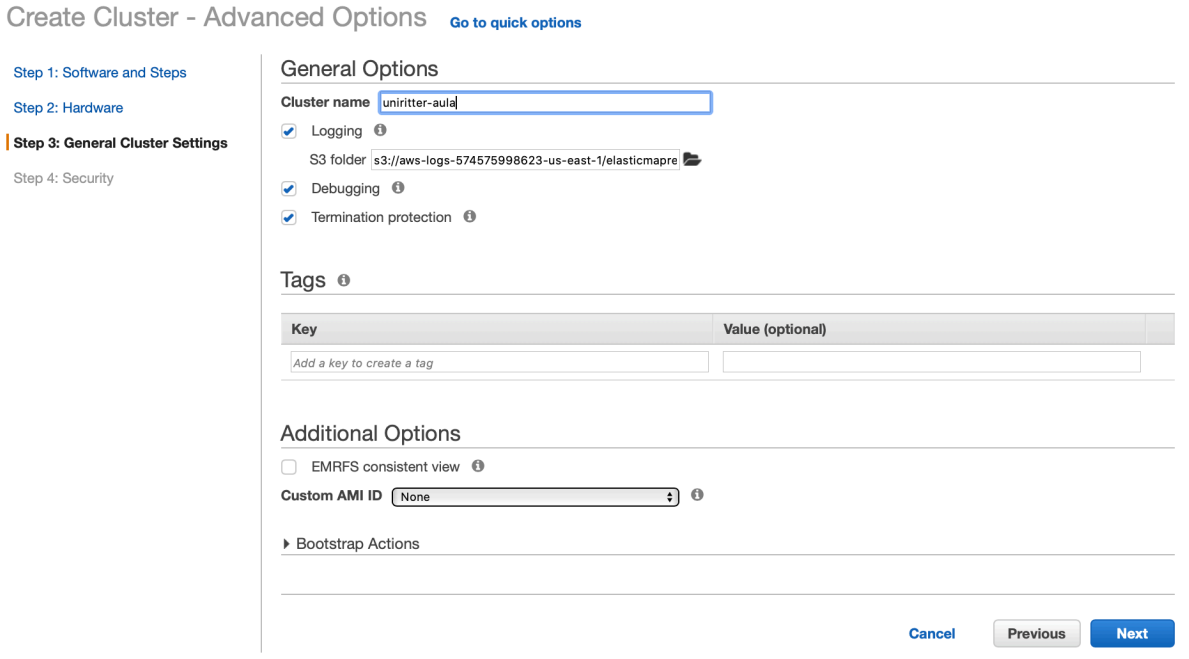


6. Selecionar os componentes conforme abaixo e clicar em “Next”.

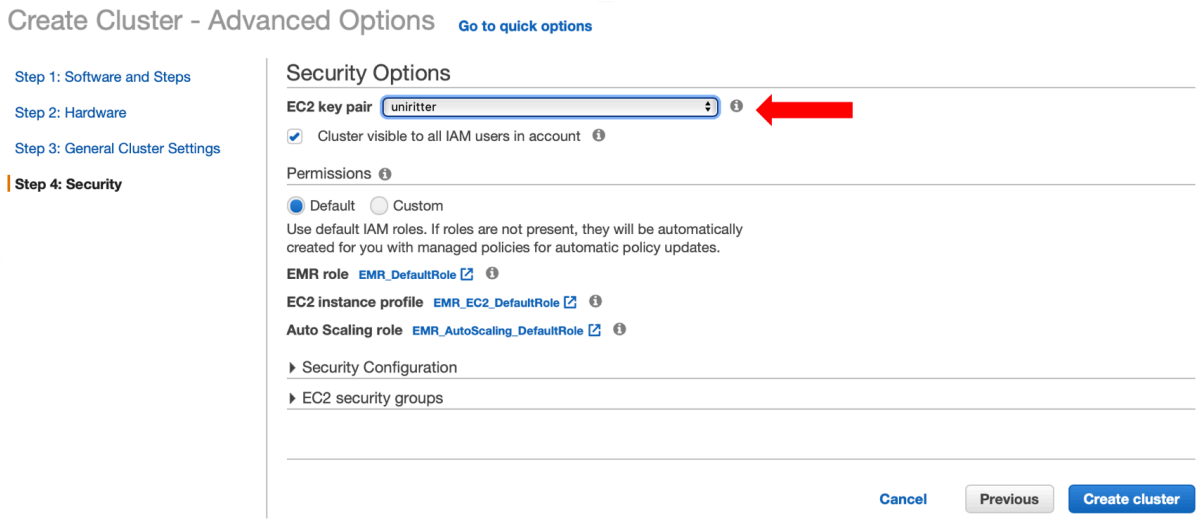
   *** Recomenda-se utilizar Spot instances para estes testes devido ao custo.
   *** Configurações de Network e EC2 Subnet conforme sua conta.
   *** tempo aproximado para iniciar o cluster é de 15 minutos.



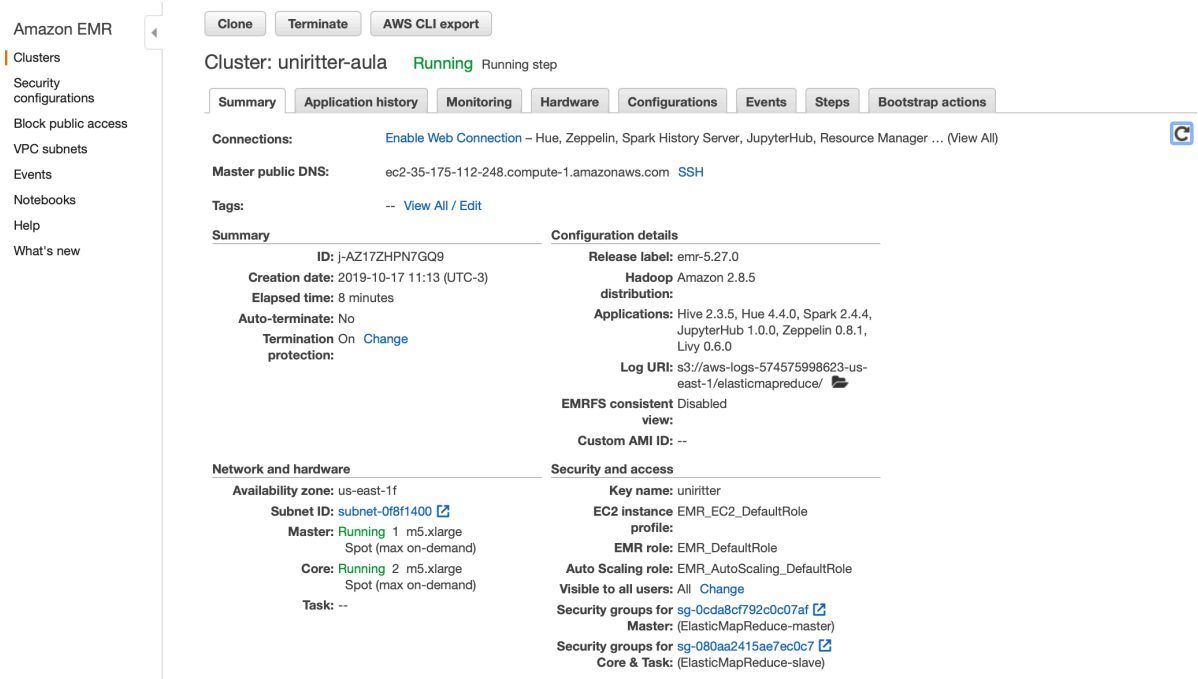
7. Informa o nome do cluster.



8. Selecionar a chave e clicar em “Create Cluster”.



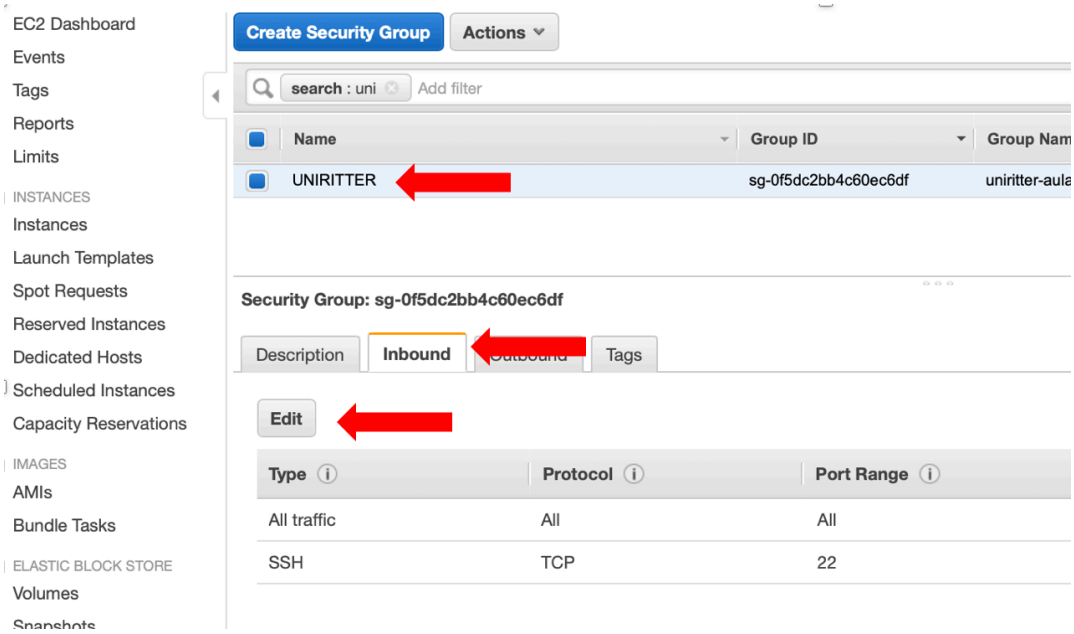
9. Aguardar o provisionamento do Cluster.
10. Após finalizar o provisionamento os status devem estar conforme abaixo.



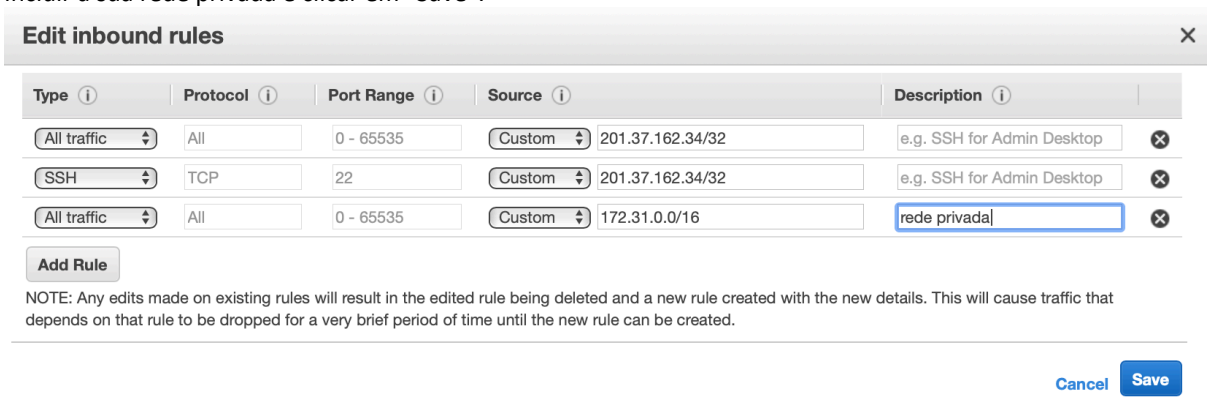
Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

Configuração do ambiente para ingestão de dados ao Data Lake

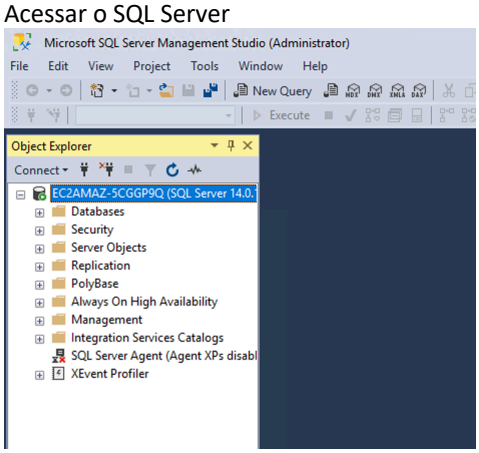
1. Acessar Security Group unirriter-aula e liberar acesso as portas/protocolos na rede privada. Clicar no botão “Edit”.



2. Incluir a sua rede privada e clicar em “Save”.

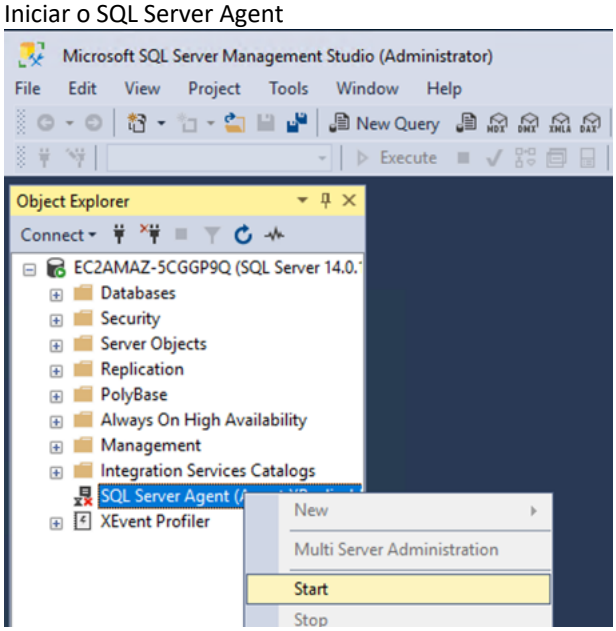


3. Acessar o SQL Server



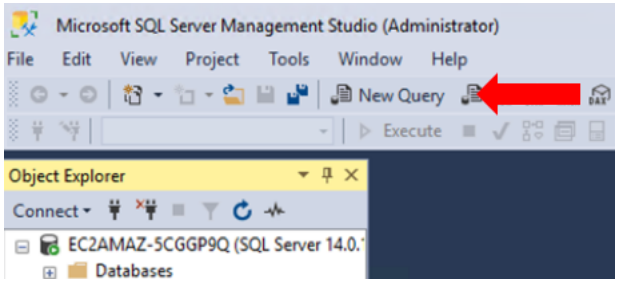
Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

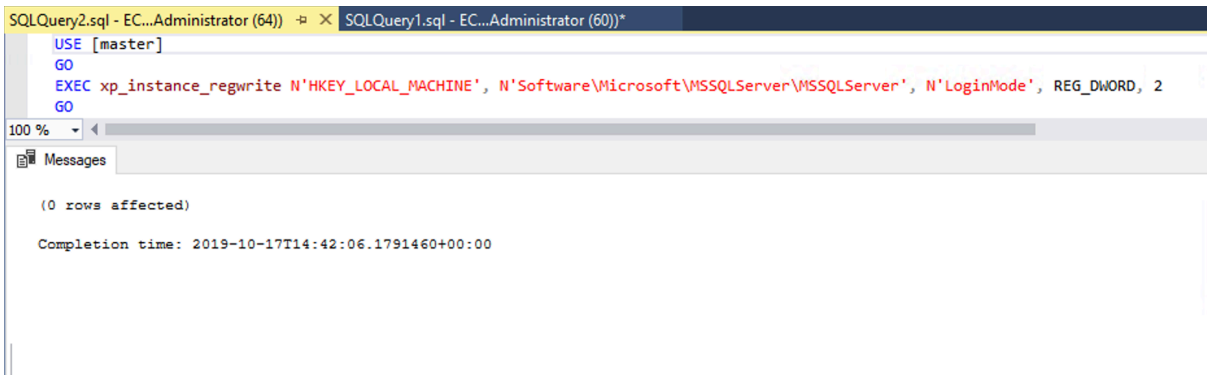
4. Iniciar o SQL Server Agent



5. Selecionar “New Query” e executar o comando abaixo para alteração da instância para modo de autenticação mista (Windows e SQL).
Comando:

```
USE [master]
GO
EXEC xp_instance_regwrite N'HKEY_LOCAL_MACHINE',
N'Software\Microsoft\MSSQLServer\MSSQLServer', N'LoginMode', REG_DWORD, 2
GO
```

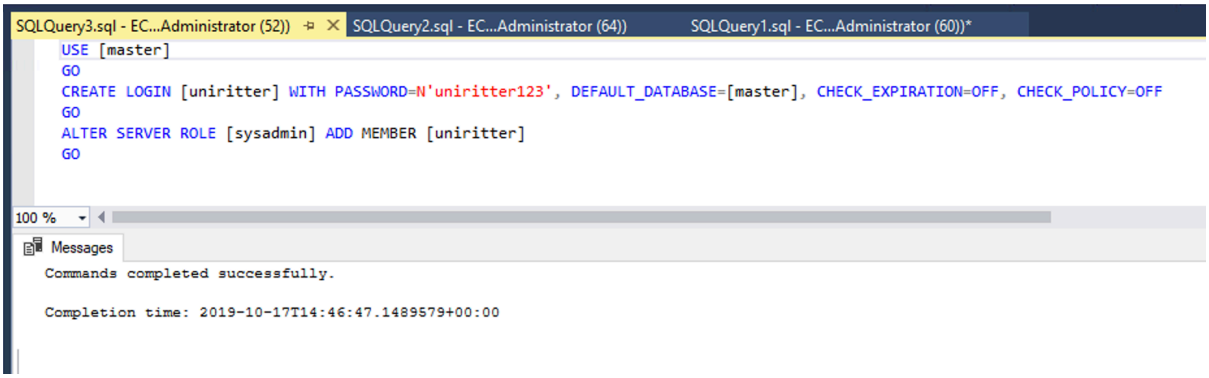




Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

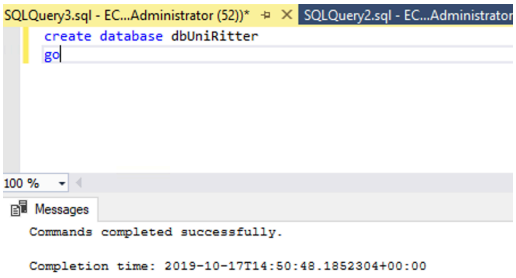
6. Criar usuário para os testes.
   Comando:

```
USE [master]
GO
CREATE LOGIN [uniritter] WITH PASSWORD=N'uniritter123',
DEFAULT_DATABASE=[master], CHECK_EXPIRATION=OFF, CHECK_POLICY=OFF
GO
ALTER SERVER ROLE [sysadmin] ADD MEMBER [uniritter]
GO
```



7. Criar database.
   Comando:

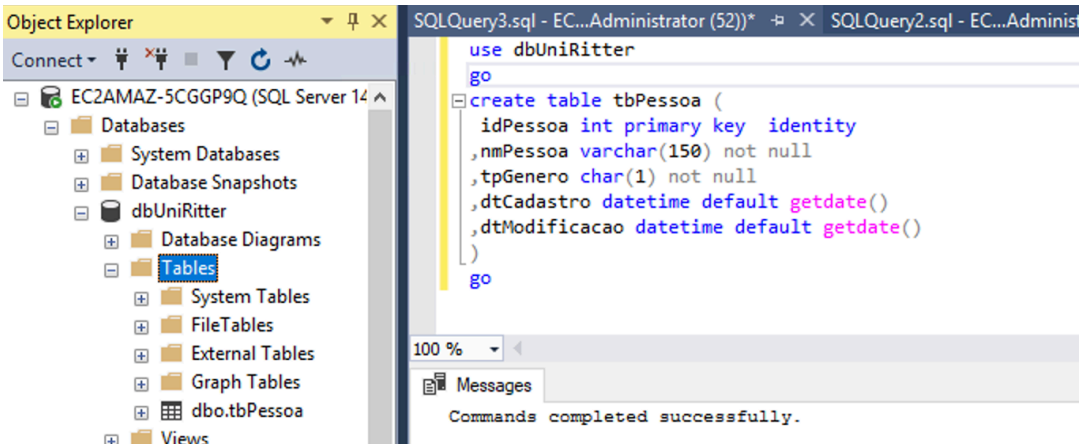
```
create database dbUniRitter
go
```



8. Criar tabela tbPessoa
   Comando:

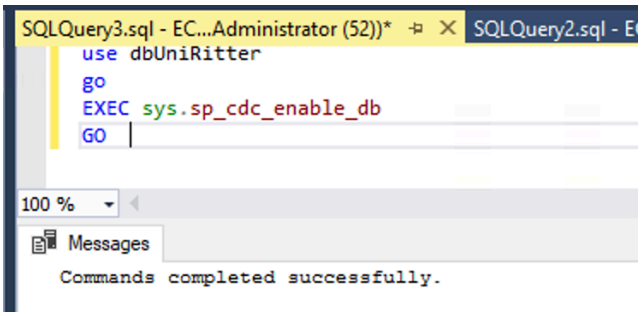
```
use dbUniRitter
go
create table tbPessoa (
 idPessoa int primary key  identity
,nmPessoa varchar(150) not null
,tpGenero char(1) not null
,dtCadastro datetime default getdate()
,dtModificacao datetime default getdate()
)
```

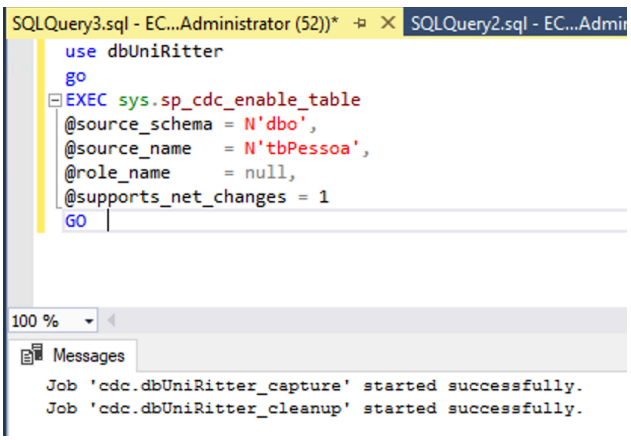
```
go
```



9. Habilitar CDC – Change Data Capture
   Comandos:

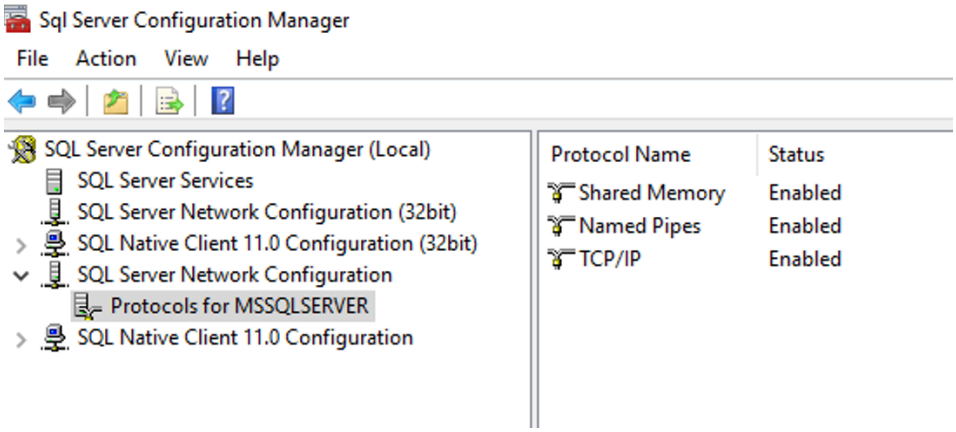
```
use dbUniRitter
go
EXEC sys.sp_cdc_enable_db
GO

use dbUniRitter
go
EXEC sys.sp_cdc_enable_table
@source_schema = N'dbo',
@source_name   = N'tbPessoa',
@role_name     = null,
@supports_net_changes = 1
GO
```



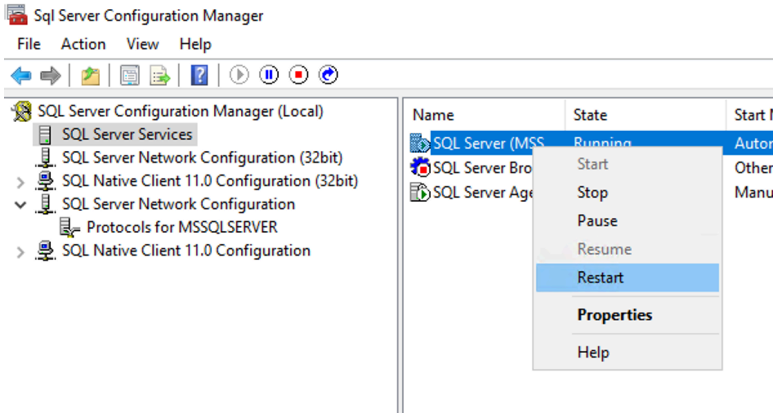


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

10. Acessar o SQL Configuration Manager e habilitar os protocolos conforme abaixo.



11. Reiniciar o SQL Server.



12. Acessar o servidor do Kafka por SSH.
    ssh -i uniritter.pem ec2-user@<ip do servidor>

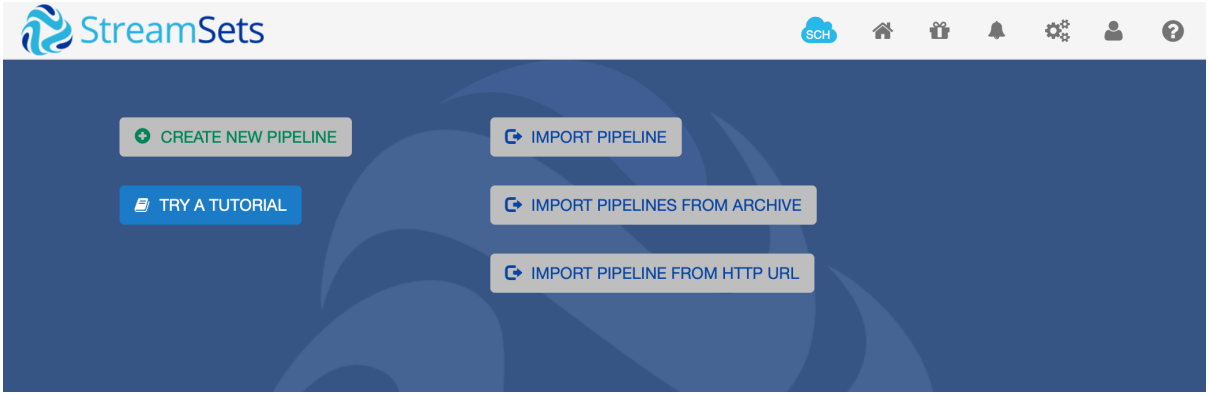
13. Acessar o diretório do Kafka.
    Comando:
    cd kafka_2.12-2.3.0/

14. Criar tópico para receber as mensagens da tabela de pessoa.
    Comando:
    bin/kafka-topics.sh --create --zookeeper localhost:2181 --replication-factor 1 --partitions 13 --topic kf-tbpessoa

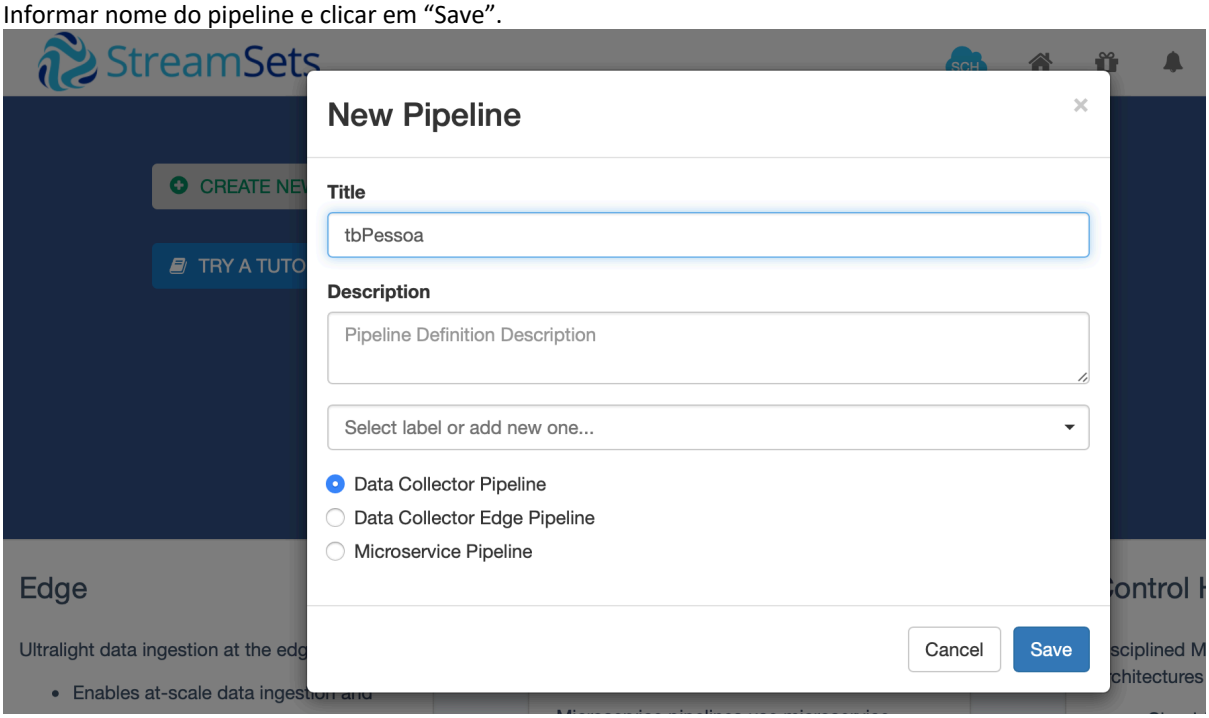
Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

15. Acessar o StreamSets
    URL: http://<ip do servidor>:18630

    Usuário: admin
    Senha: admin

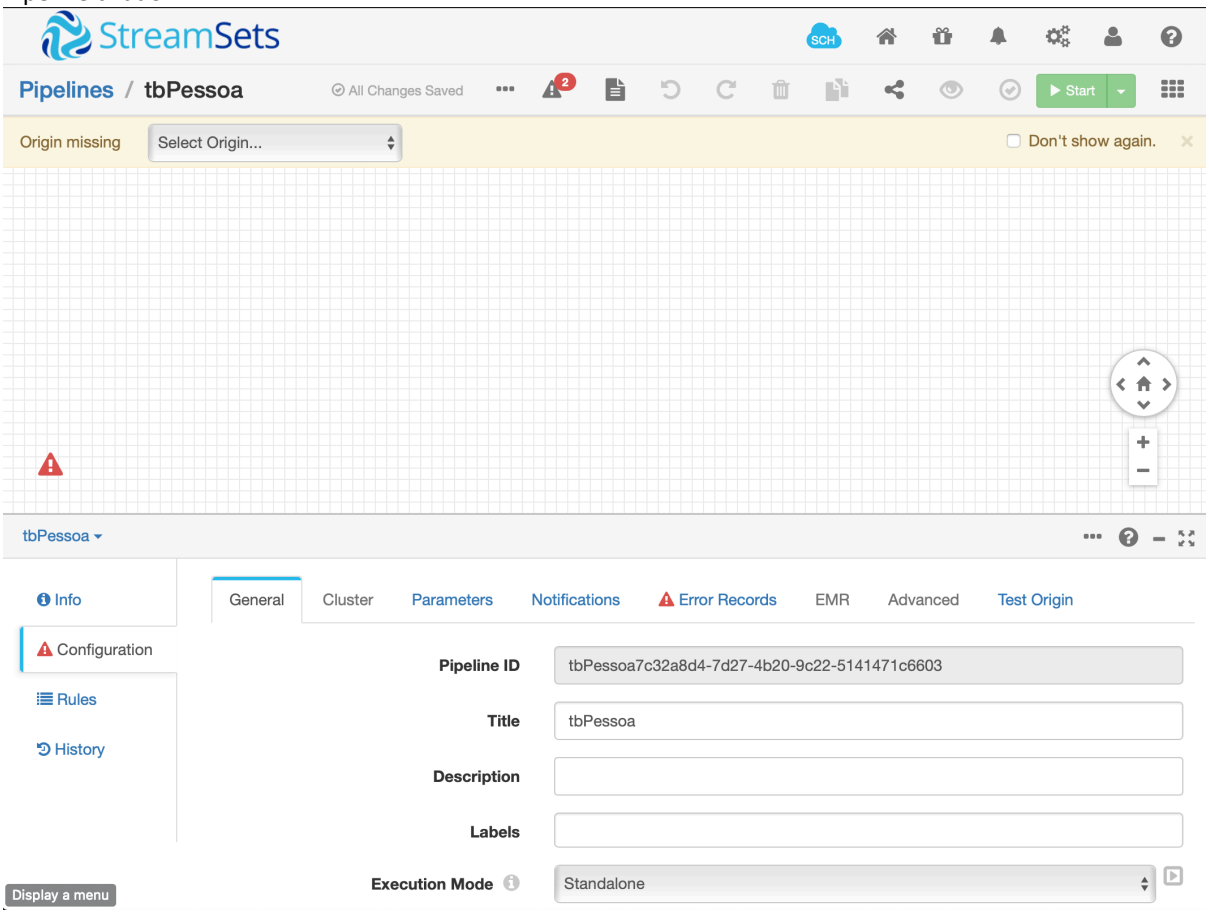


16. Clicar em Create New Pipeline.
17. Informar nome do pipeline e clicar em "Save".

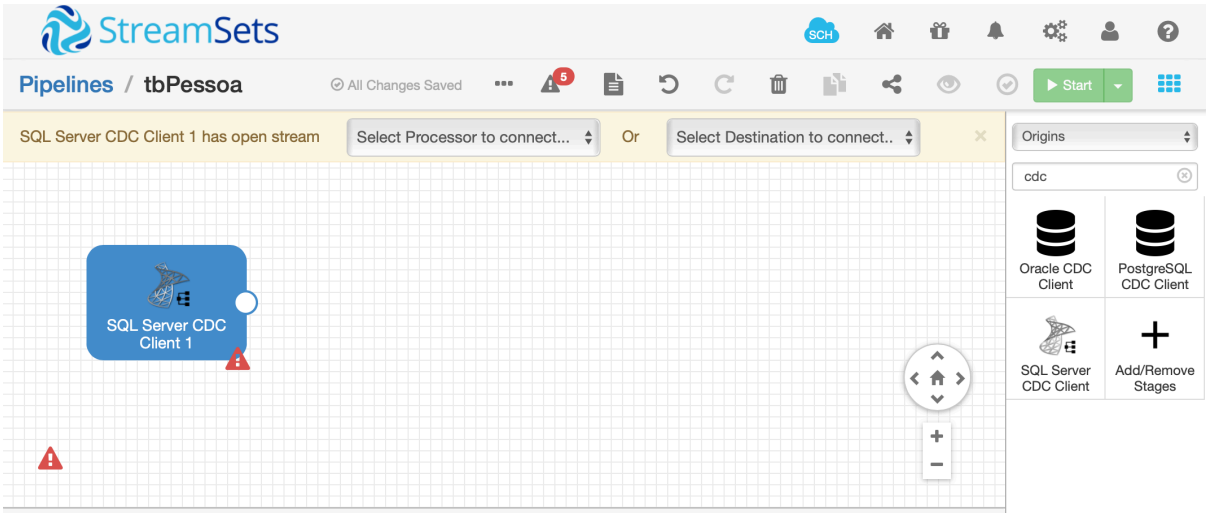


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

18. Pipeline criado.

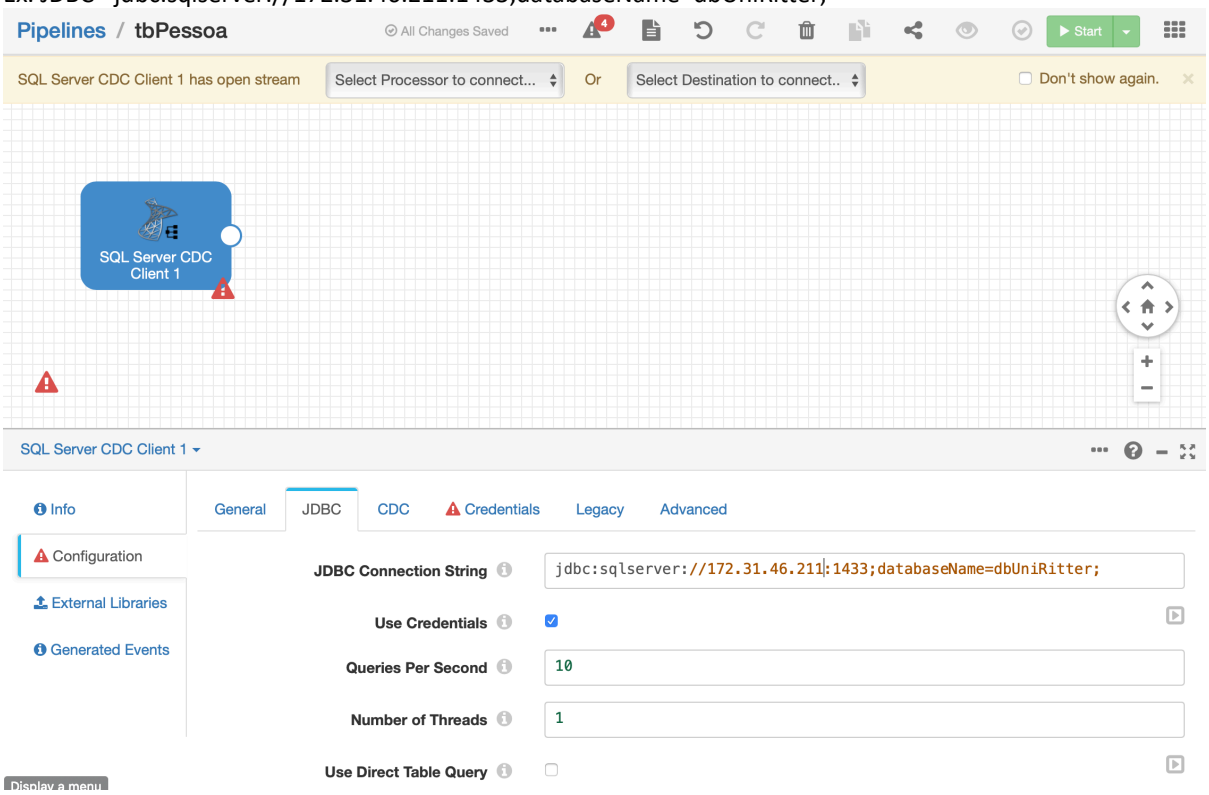


19. Selecionar SQL Server CDC Client.

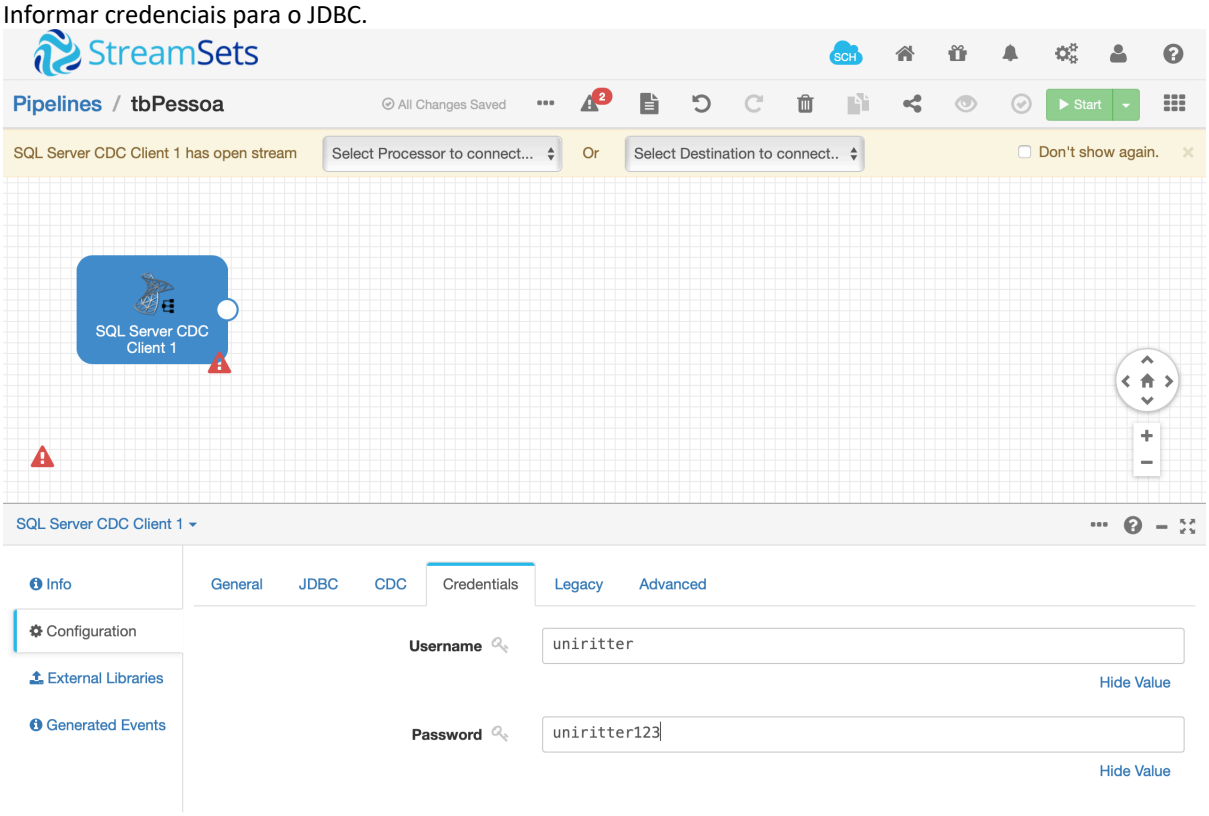


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

20. Configurar JDBC.
    Ex. JDBC - jdbc:sqlserver://172.31.46.211:1433;databaseName=dbUniRitter;

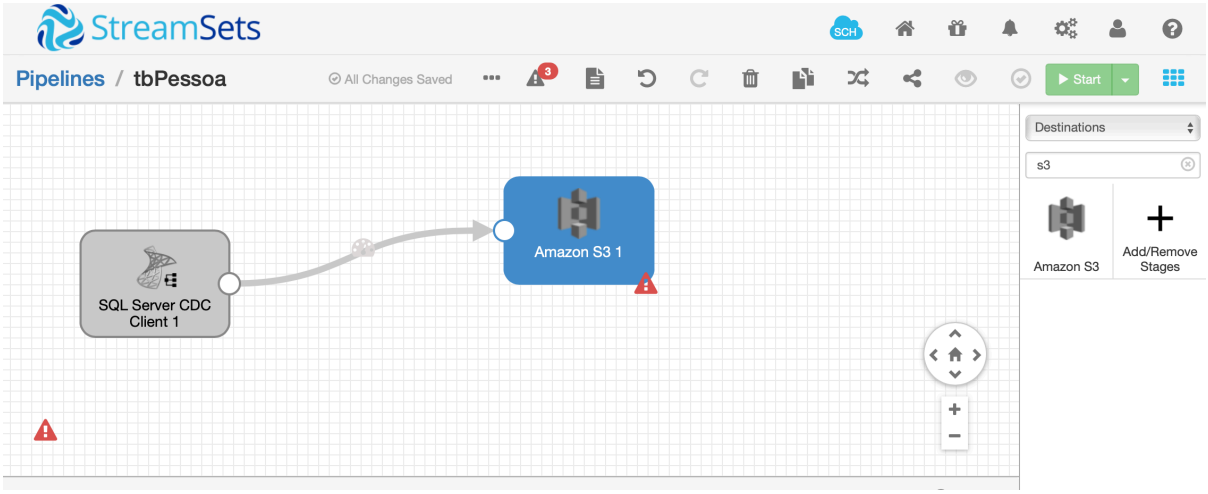


21. Informar credenciais para o JDBC.

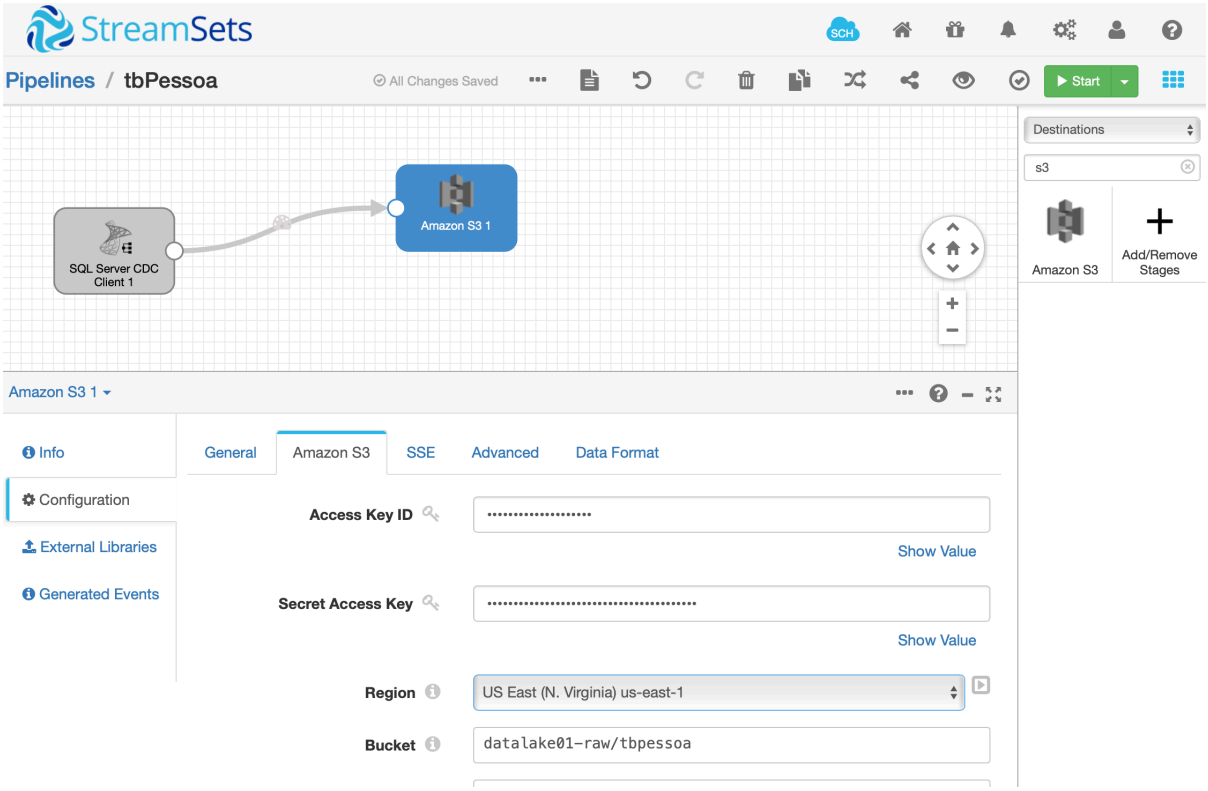


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

22. Selecionar destino Amazon S3.

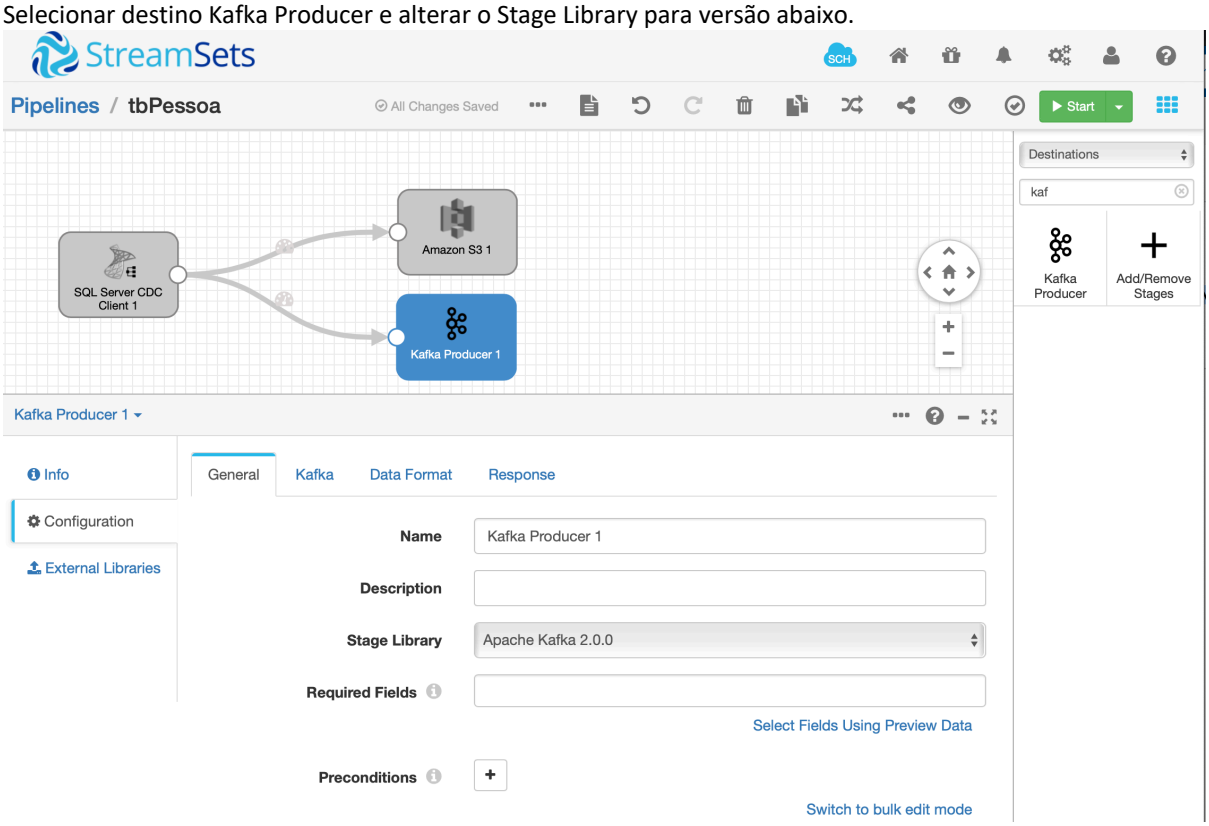


23. Configurar credenciais de acesso ao S3, região e bucket.

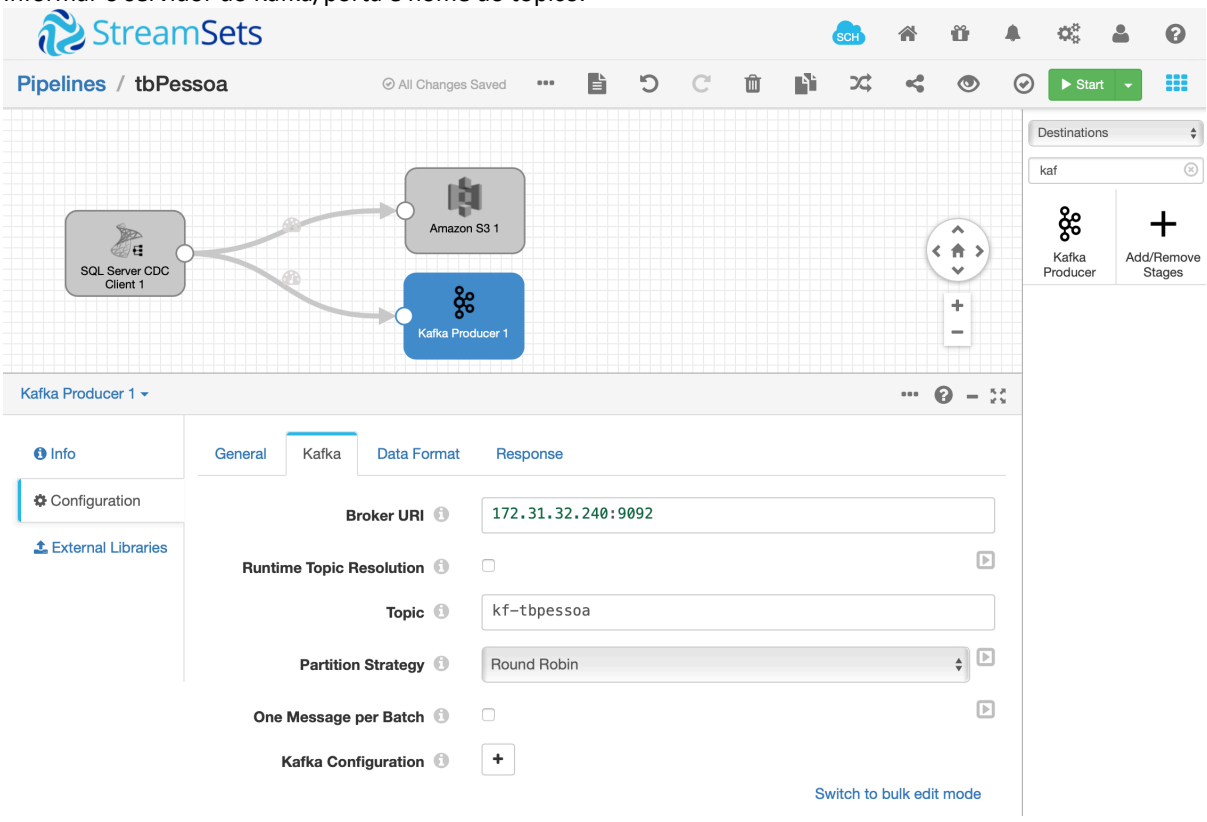


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

24. Selecionar destino Kafka Producer e alterar o Stage Library para versão abaixo.

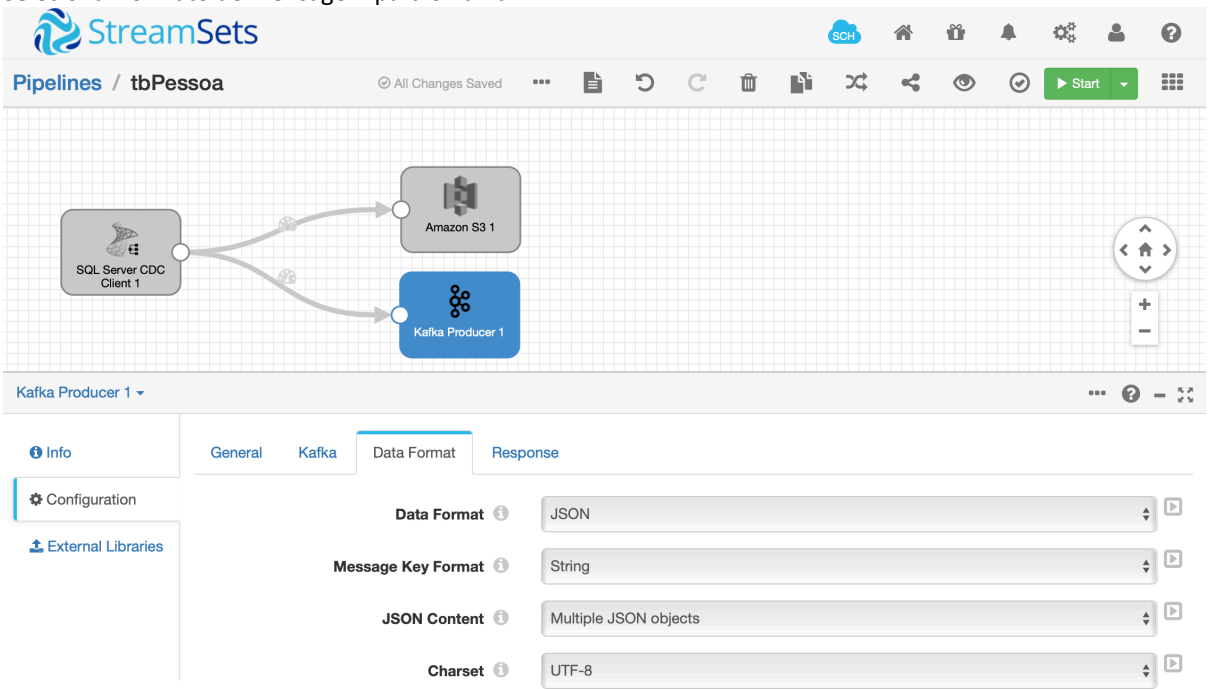


25. Informar o servidor do Kafka/porta e nome do tópico.



Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

26. Selecionar formato de mensagem para o Kafka.

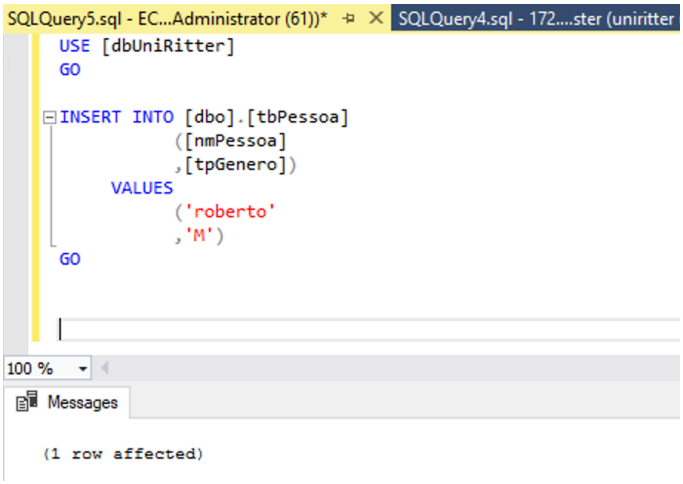


27. Iniciar o pipeline.

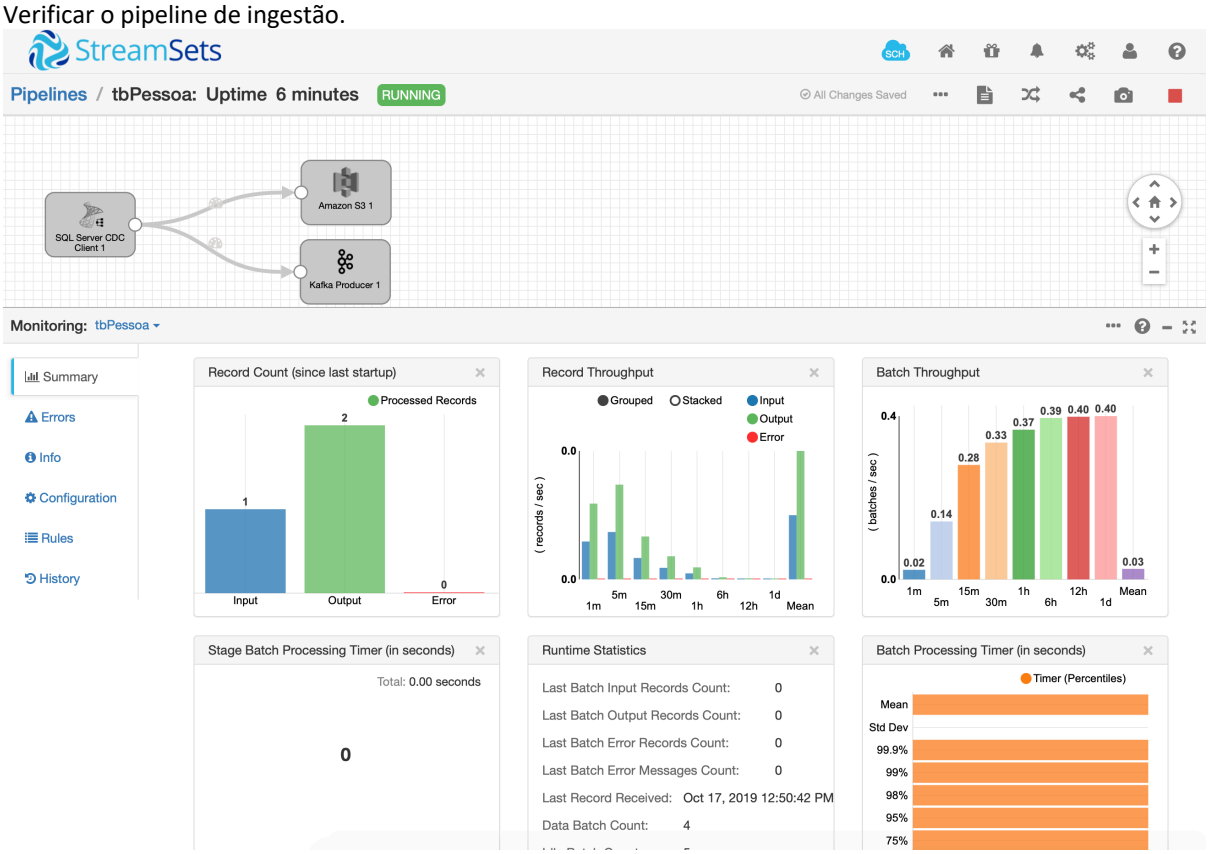
Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

28. Realizar teste de fluxo de ingestão inserindo registro na tabela de pessoa no SQL Server.
    Comando:

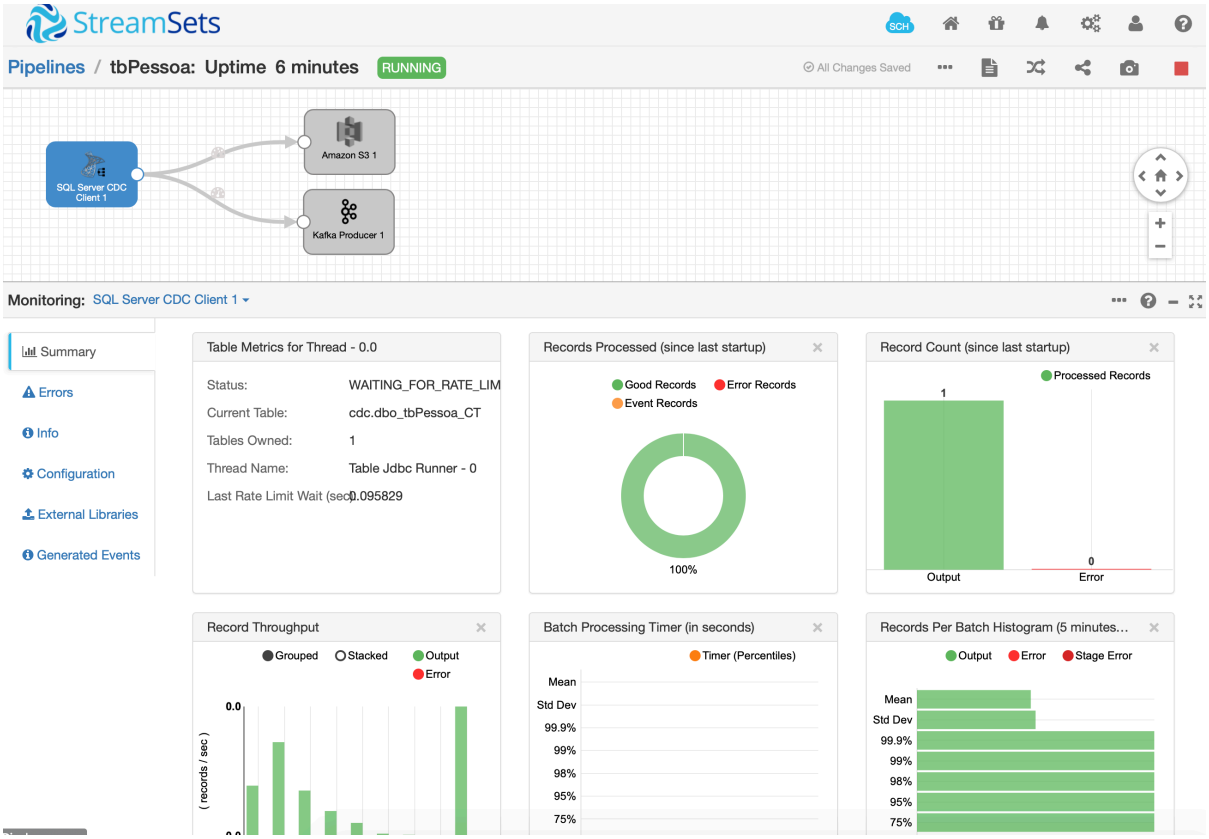
```
USE [dbUniRitter]
GO

INSERT INTO [dbo].[tbPessoa]
           ([nmPessoa]
           ,[tpGenero])
     VALUES
           ('roberto'
           ,'M')
GO
```

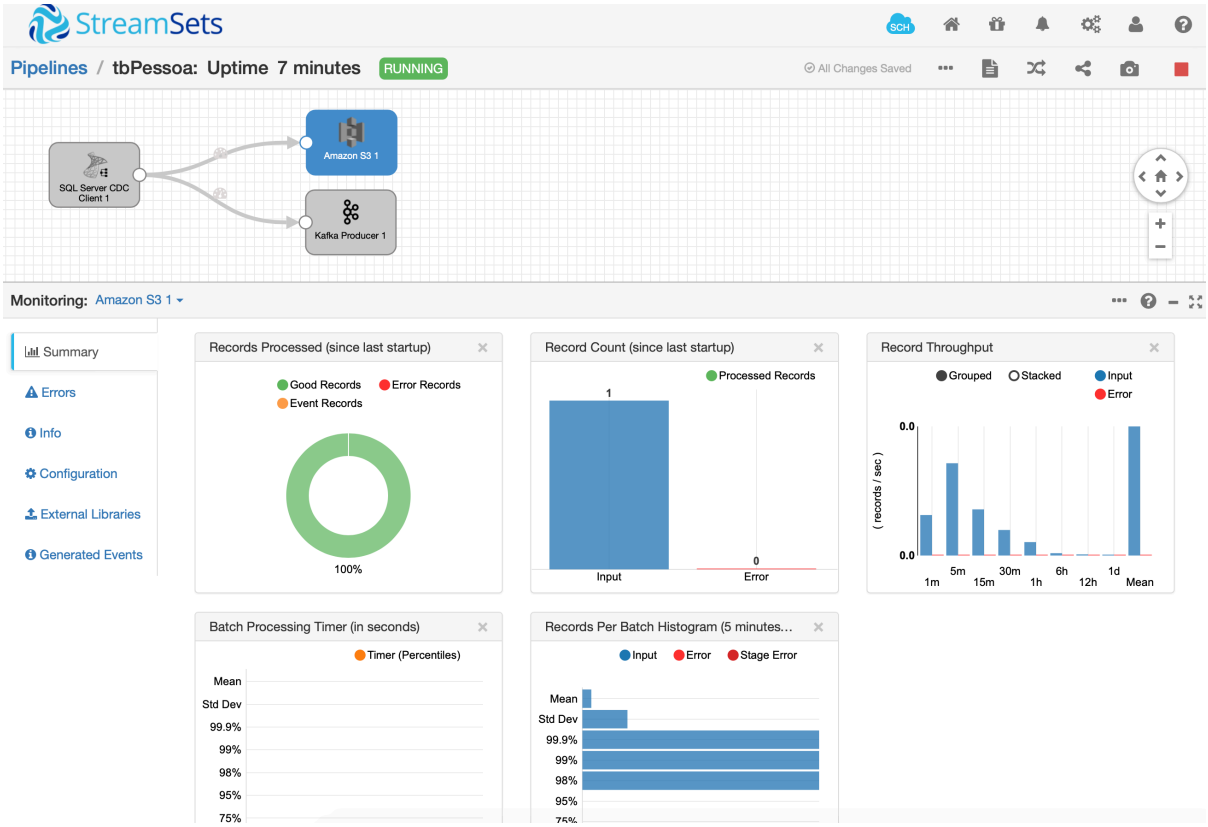


29. Verificar o pipeline de ingestão.

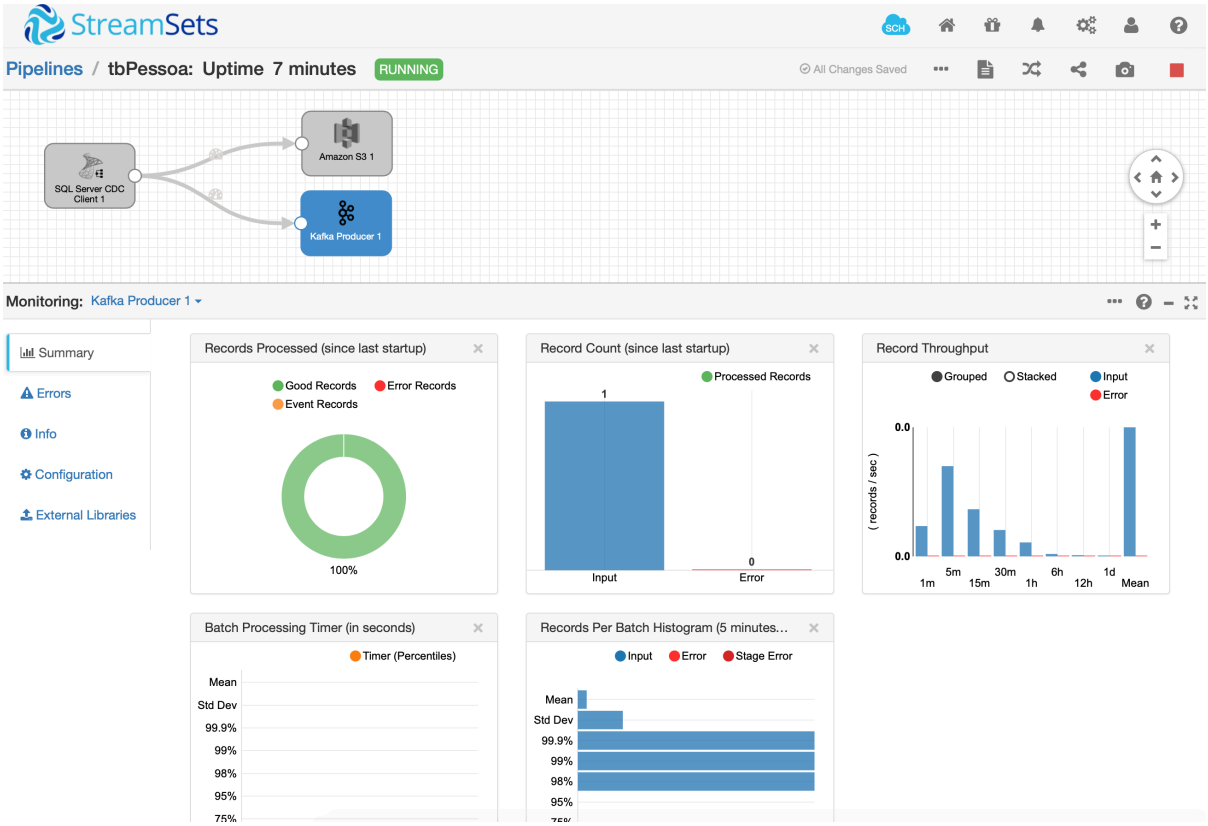


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

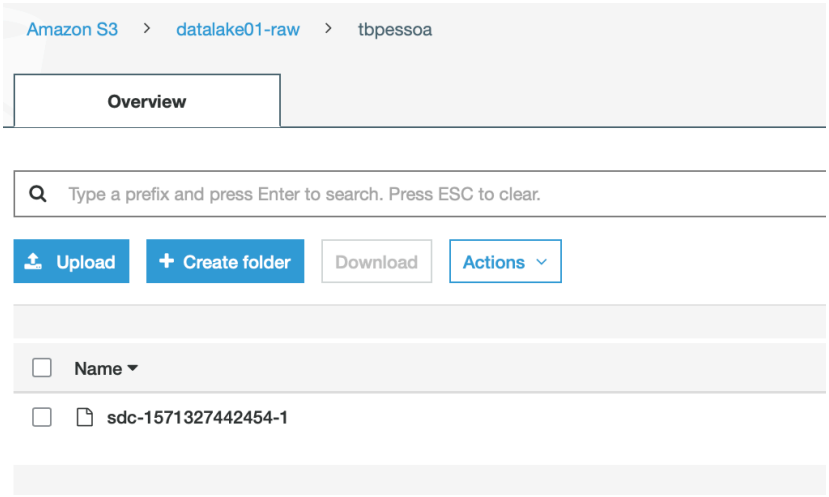




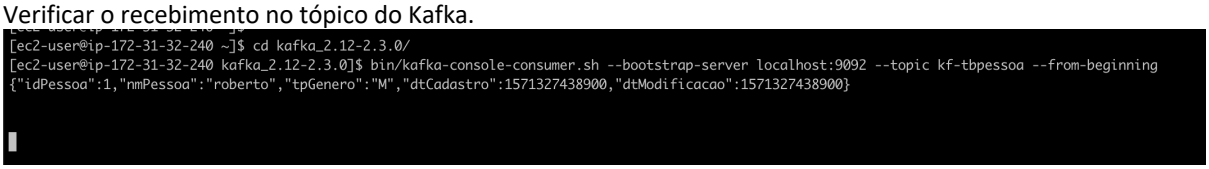
Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão



30. Verificar o recebimento no bucket S3.



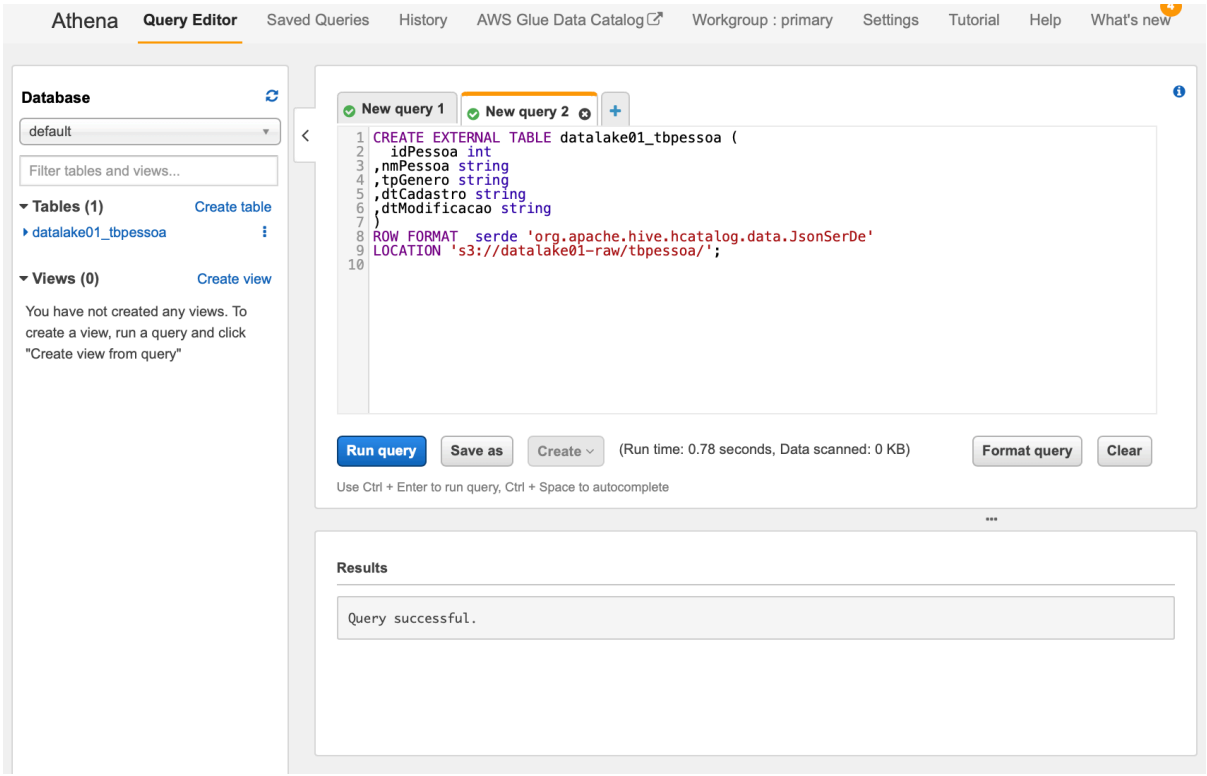
31. Verificar o recebimento no tópico do Kafka.



Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

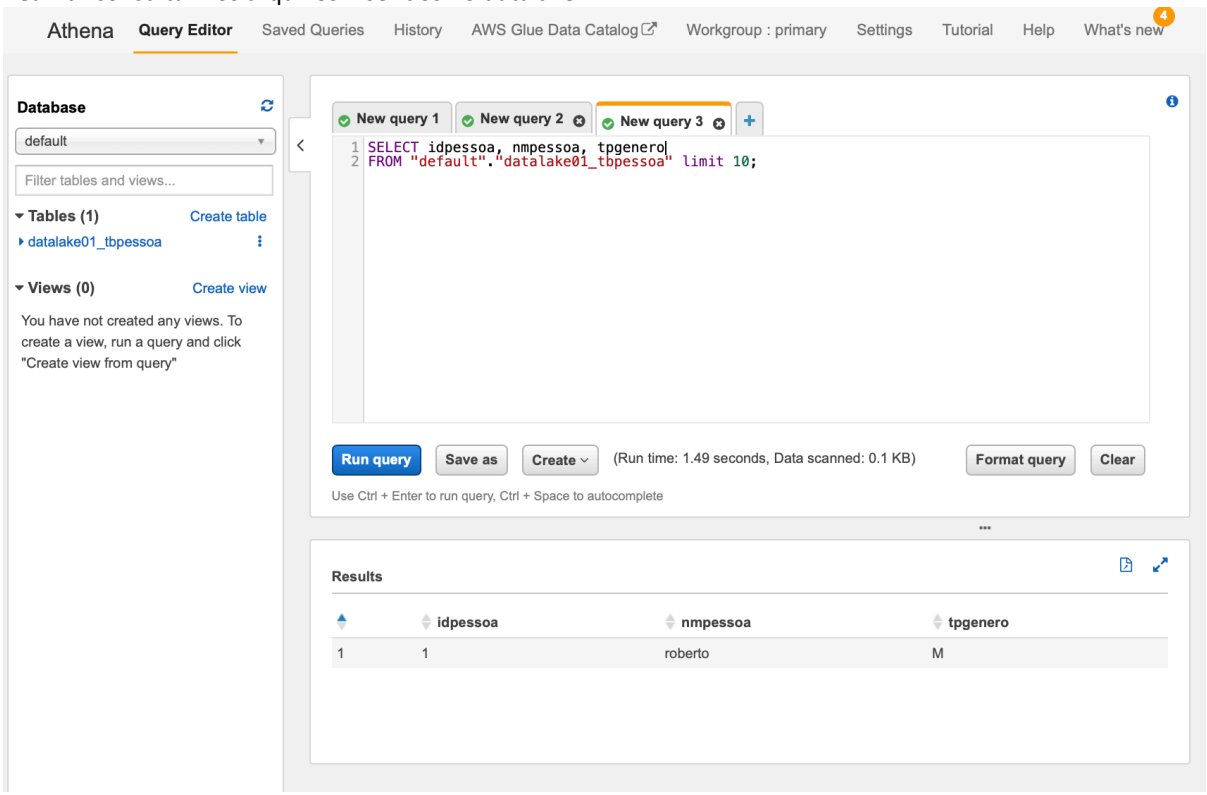
32. Criar tabela no AWS Athena para realizar consulta nos dados.
    Acessar Services > Athena
    Comando:

```
CREATE EXTERNAL TABLE datalake01_tbpessoa (
  idPessoa int
,nmPessoa string
,tpGenero string
,dtCadastro string
,dtModificacao string
)
ROW FORMAT  serde 'org.apache.hive.hcatalog.data.JsonSerDe'
LOCATION 's3://datalake01-raw/tbpessoa/';
```

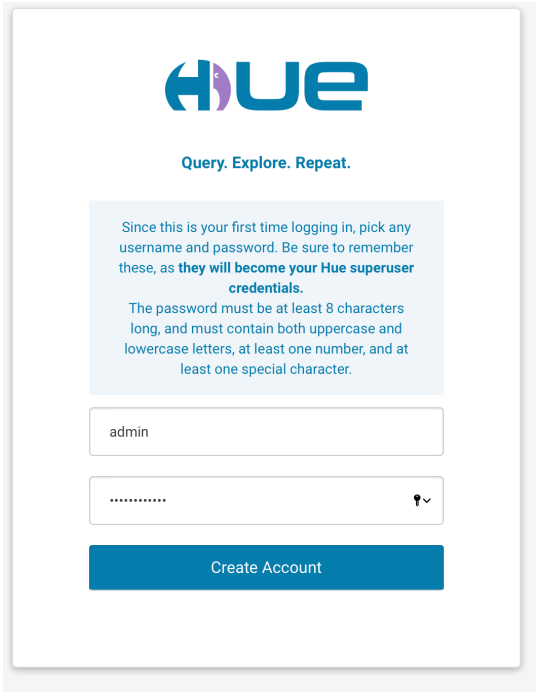


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

33. Realizar consultar nos arquivos inseridos no datalake.

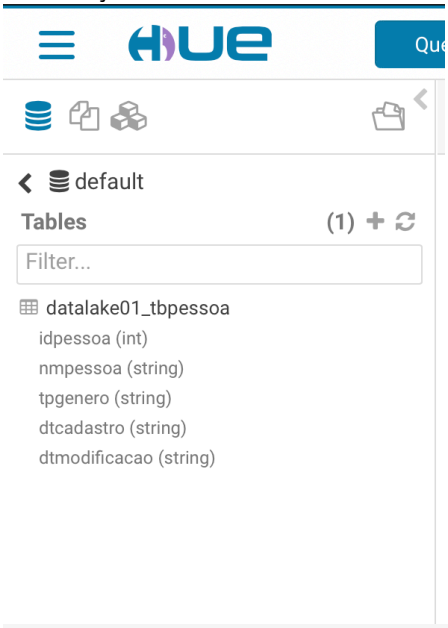


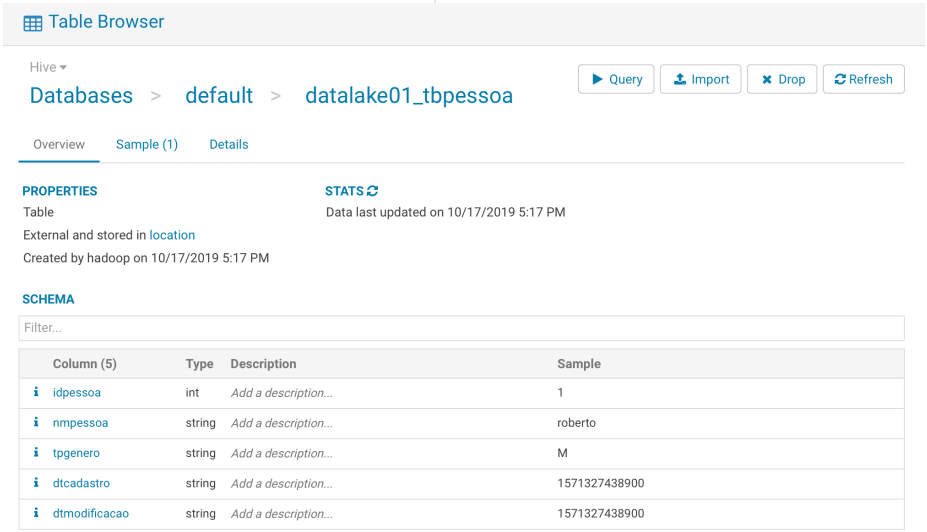
34. Realizar acesso através do self-service (HUE).
No primeiro acesso é necessário definir usuário e senha.
Link: http://<ip publico>:8888



Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

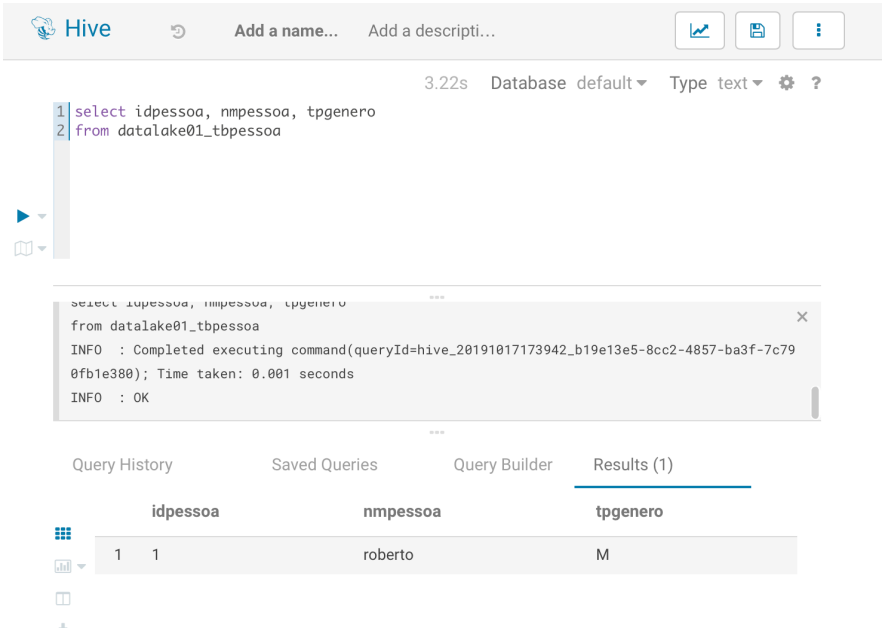
35. Visualização da estrutura de dados.



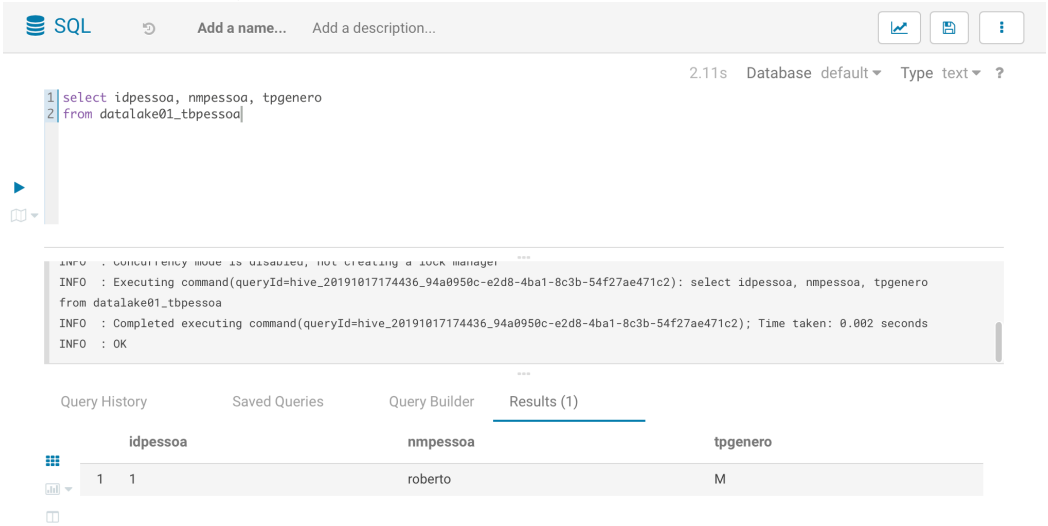


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

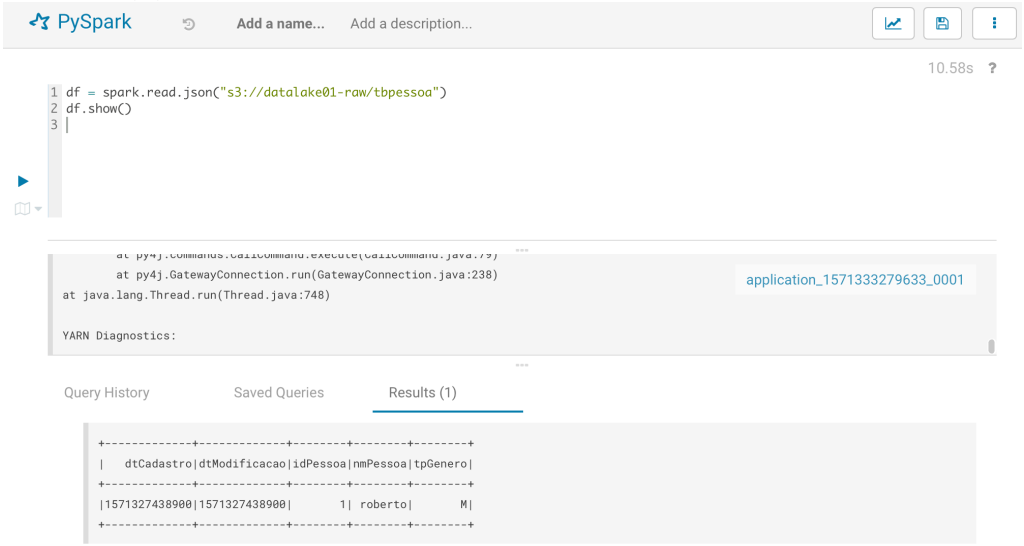
36. Consulta através do HIVE.



37. Consulta através do SparkSQL.



38. Notebook PySpark.



Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

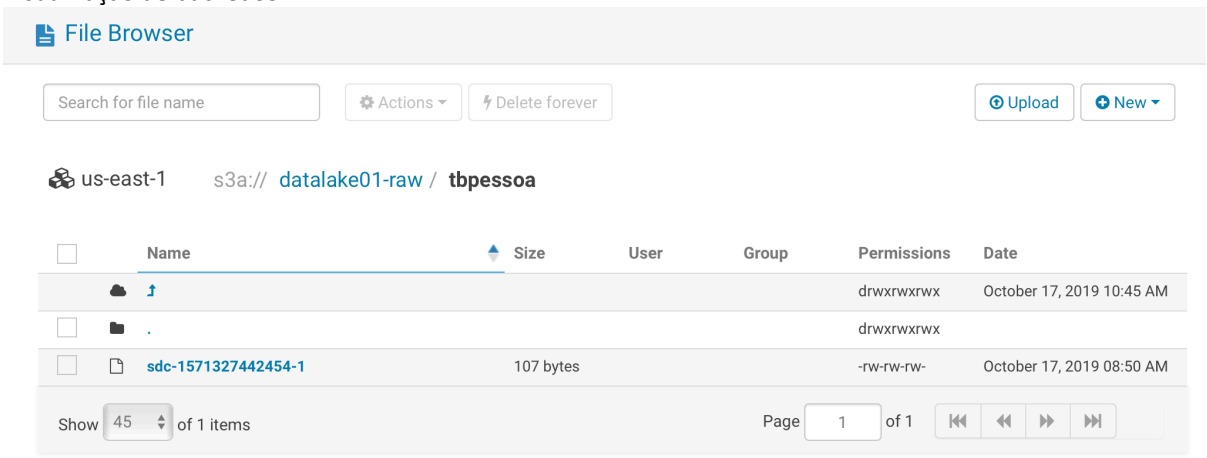
39. Notebook Scala.

40. Acesso ao contexto do HIVE através do PySpark.

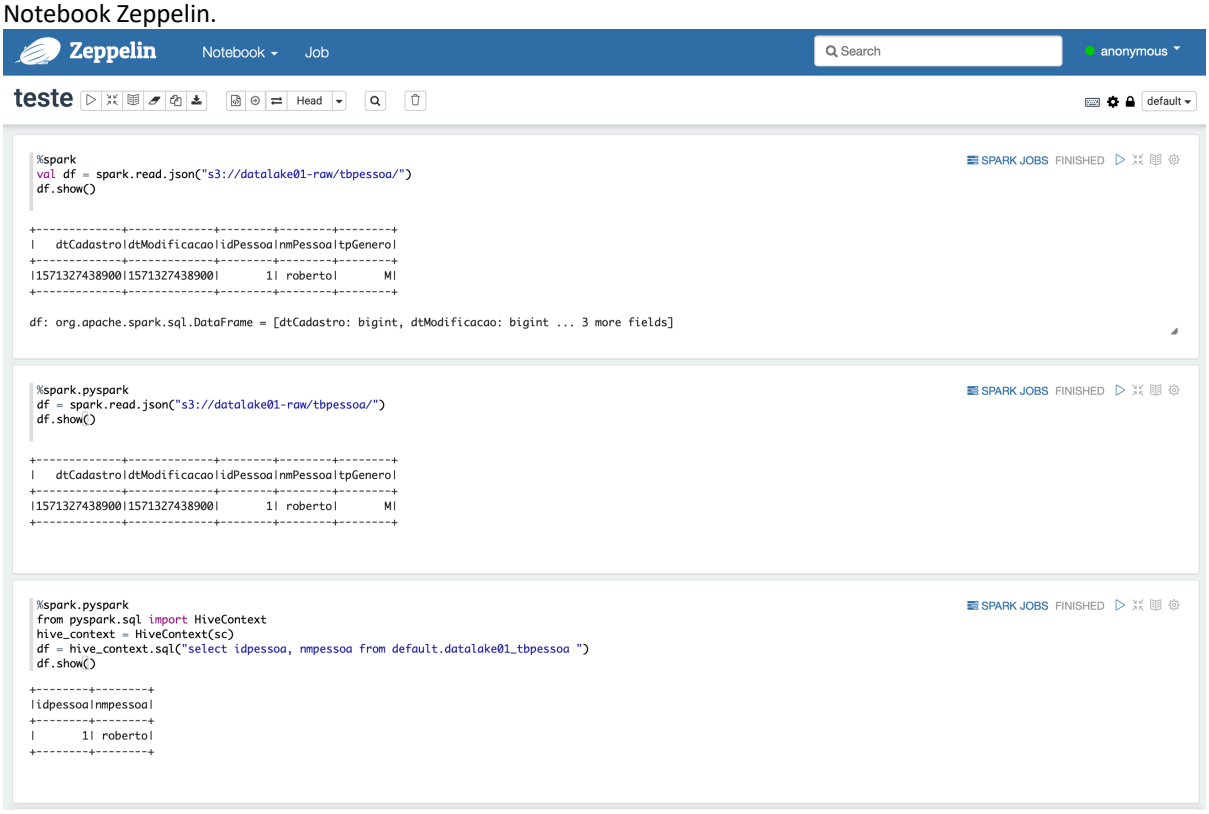
```
from pyspark.sql import HiveContext
hive_context = HiveContext(sc)
df = hive_context.sql("select idpessoa, nmpessoa from default.datalake01_tbpessoa ")
df.show()
```

Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

41. Visualização de bucket S3.

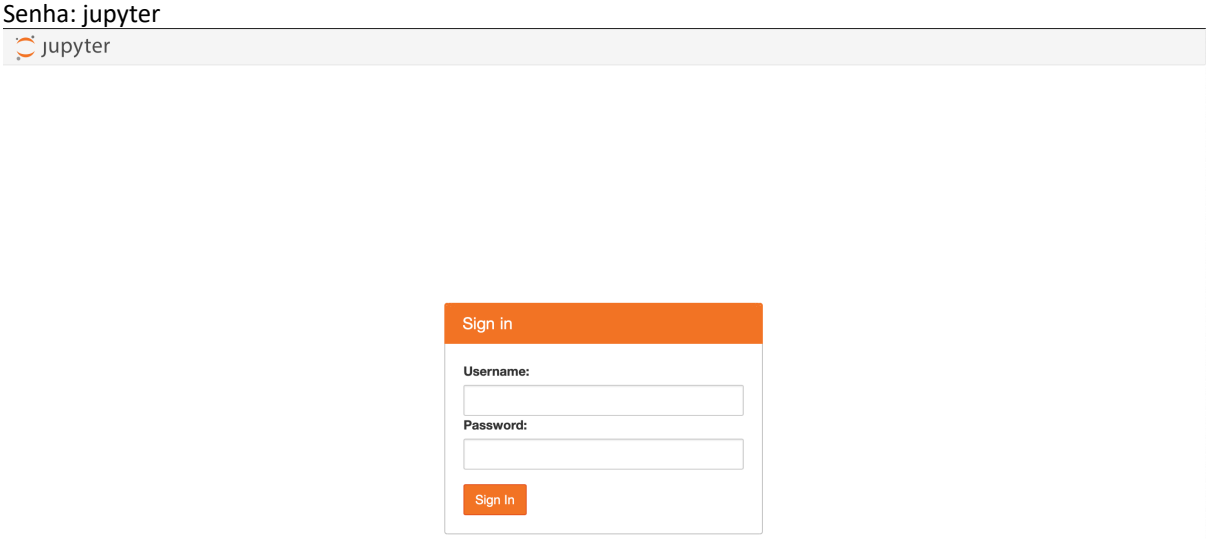


42. Notebook Zeppelin.

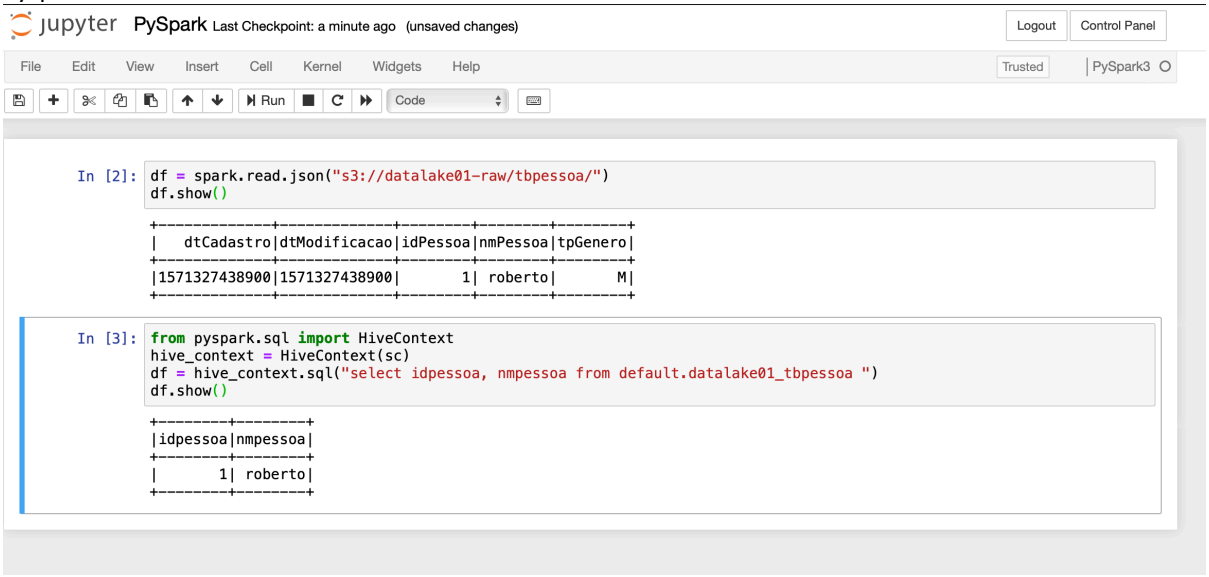


Disciplina: Processamento de Grandes Volumes de Dados
Professor: Roberto Galvão

43. Notebook JupyterHub (EMR)
    https://<ip publico>:9443
    Usuário default: jovyan
    Senha: jupyter



44. Pyspark



45. Spark – Scala