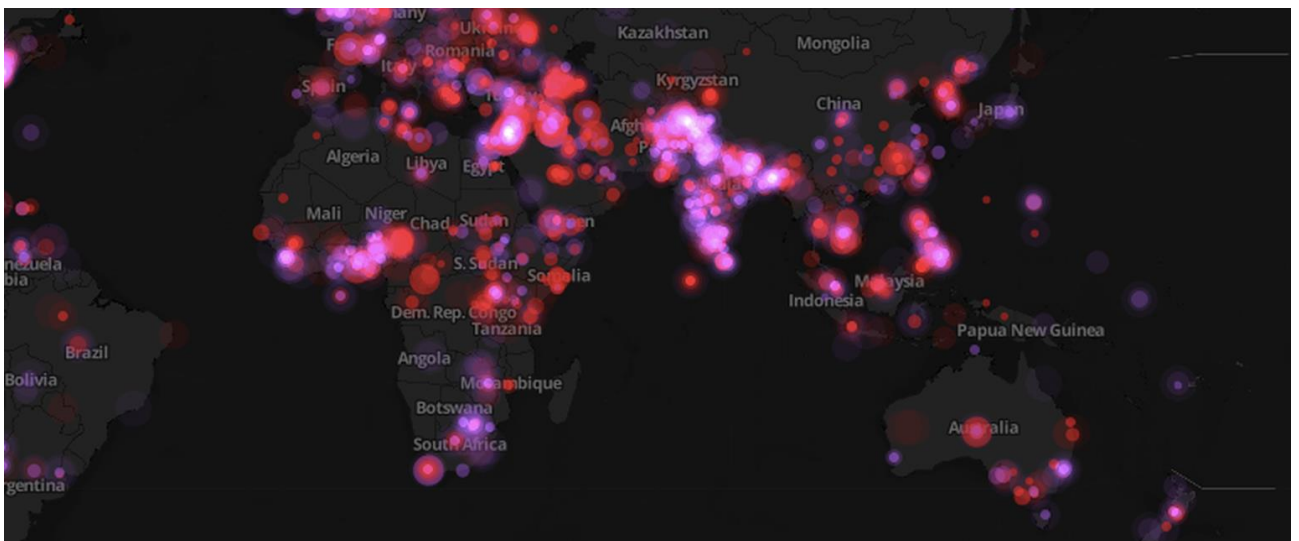


PROJET GDELT 2020

1. Présentation :

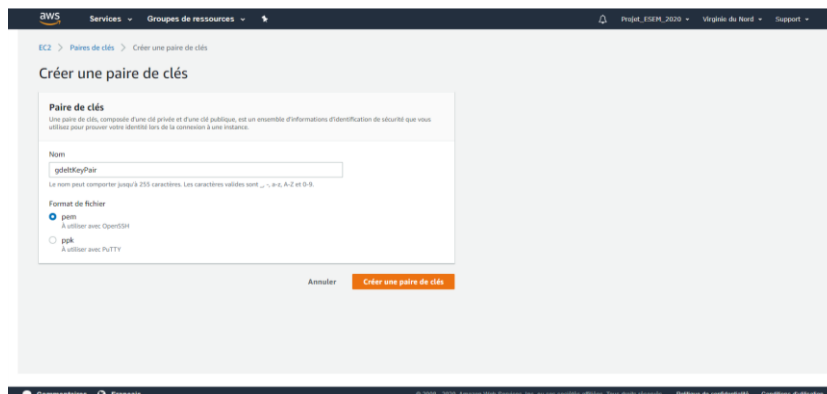
Le but du projet est d'analyser l'année 2020 via la base de données GDELT en utilisant le tone des articles dans les médias des divers pays du monde. "Le projet GDELT est une initiative visant à construire un catalogue de comportements et de croyances à l'échelle de la société humaine dans tous les pays du monde, reliant chaque personne, organisation, lieu, décompte, thème, source d'actualités et événement à travers la planète en un seul massif un réseau qui capture ce qui se passe dans le monde, quel est son contexte et qui est impliqué, et comment le monde en ressent chaque jour. "



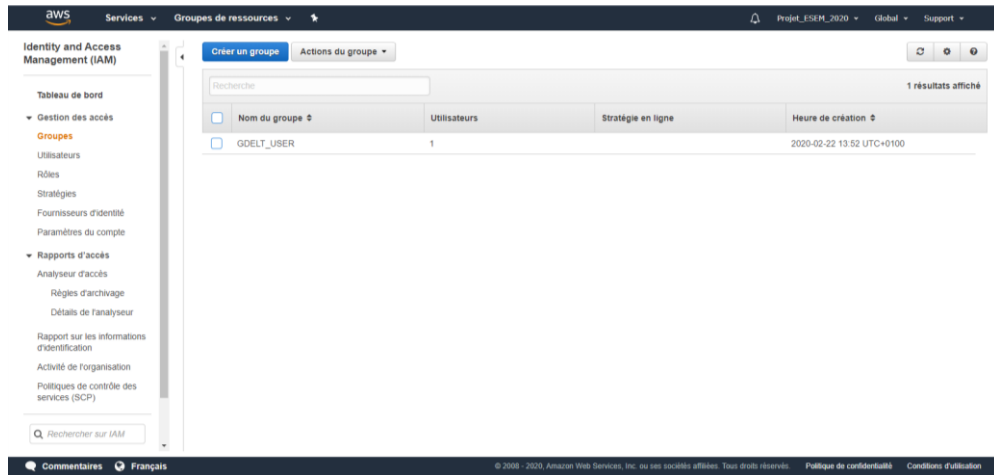
2. Création d'un cluster sur AWS EMR :

- On commence par la création d'une clé privée qui sera automatiquement sauvegardé dans un fichier `gdelKeyPair.pem`.
- On utilisera cette clé pour la connexion SSH.

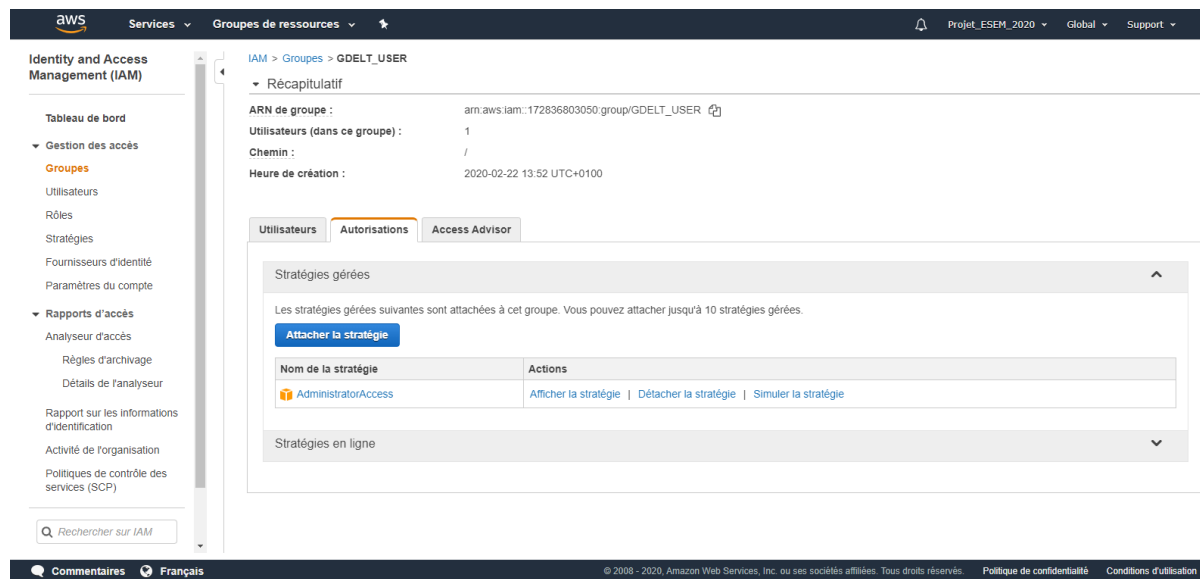
<https://console.aws.amazon.com/ec2/home?region=us-east-1#CreateKeyPair:>

Screenshot of the AWS Management Console 'Create a key pair' page. The 'Name' field is filled with 'gdelKeyPair'. The 'Format of file' section has 'pem' selected, with a note 'À utiliser avec OpenSSH'. The 'ppk' option is also visible with the note 'À utiliser avec PuTTY'. Buttons for 'Annuler' and 'Créer une paire de clés' are at the bottom.

- Création d'un groupe admin
<https://console.aws.amazon.com/iam/home?#/groups>



- Choix du type de stratégie : AdministratorAccess



- Ajout d'un utilisateur : GDELT_USER

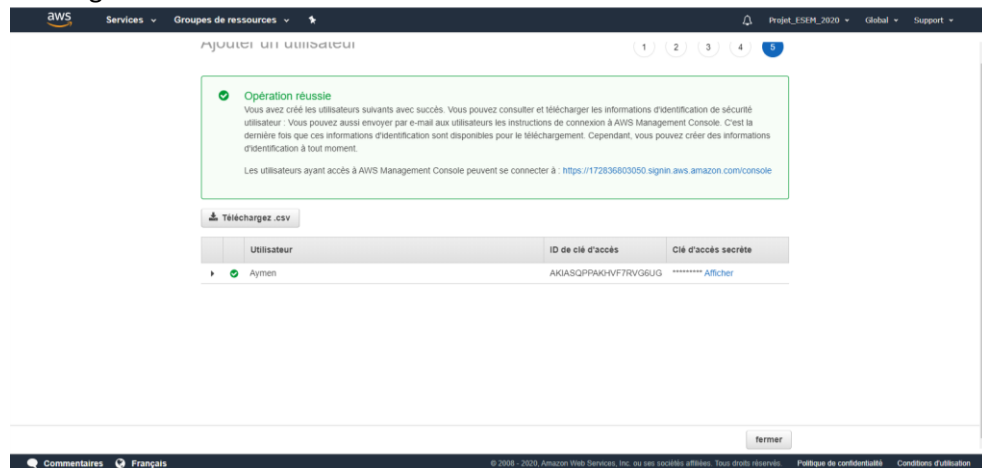
- Choix du type d'accès : « Accées par programmation » afin de créer un ID d'accées et une clé de sécurité.

The screenshot shows the AWS IAM console 'Ajouter un utilisateur' (Add user) page, step 2 of 5. The page title is 'Définissez les informations utilisateur' (Define user information). Below the title, it says 'Vous pouvez ajouter plusieurs utilisateurs en même temps avec les mêmes types et autorisations d'accès. [En savoir plus](#)' (You can add multiple users at the same time with the same types and access permissions. [En savoir plus](#)). The 'Nom d'utilisateur*' (User name*) field contains 'GDELT_USER'. Below this field is a link 'Ajoutez un autre utilisateur' (Add another user). The 'Sélectionnez un type d'accès AWS' (Select an AWS access type) section has two options: 'Accès par programmation' (Programmatic access) which is selected with a radio button, and 'Accès à AWS Management Console' (AWS Management Console access) which is not selected. The 'Accès par programmation' option has a description: 'Active un ID de clé d'accès et clé d'accès secrète pour AWS API, CLI, SDK et d'autres outils de développement.' (Activates an access key ID and secret access key for AWS API, CLI, SDK and other development tools). The 'Accès à AWS Management Console' option has a description: 'Active un mot de passe qui permet aux utilisateurs de vous connecter à l'AWS Management Console.' (Activates a password that allows users to connect to the AWS Management Console). At the bottom, there is a footer with '* Obligatoire' (Required), 'Annuler' (Cancel), and 'Suivant : Autorisations' (Next: Permissions) buttons.

- Ensuite, on met notre utilisateur dans le groupe administrateur et on valide la création de l'utilisateur.

The screenshot shows the AWS IAM console 'Ajouter un utilisateur' (Add user) page, step 3 of 5. The page title is 'Définir des autorisations' (Define permissions). Below the title, there are three buttons: 'Ajouter un utilisateur au groupe' (Add user to group), 'Copier les autorisations à partir de l'utilisateur existant' (Copy permissions from existing user), and 'Attacher directement les stratégies existantes' (Attach existing policies directly). Below these buttons, it says 'Ajoutez un utilisateur à un groupe existant ou créez-en un nouveau. L'utilisation des groupes est une façon respectant les bonnes pratiques de gérer les autorisations de l'utilisateur en fonction de ses activités professionnelles. [En savoir plus](#)' (Add a user to an existing group or create a new one. Using groups is a way to manage permissions in a best practice way based on the user's professional activities. [En savoir plus](#)). The 'Ajouter un utilisateur au groupe' (Add user to group) section has a search bar with 'Recherche' (Search) and 'Affichage de : 1 résultat' (Showing: 1 result). Below the search bar, there is a table with two columns: 'Groupe' (Group) and 'Stratégies attachées' (Attached policies). The table has one row with a checkmark in the 'Groupe' column and 'GDELT_USER' in the 'Stratégies attachées' column. At the bottom, there is a footer with 'Annuler' (Cancel), 'Précédent' (Previous), and 'Suivant : Réviser' (Next: Review) buttons.

- Après la validation de la création de l'utilisateur, on clique sur Téléchargez.csv afin de sauvegarder dans un fichier credentials.csv l'ID et la clé de sécurité.



- Démarrage du cluster sur AWS EMR : On crée le cluster en utilisant la console AWS EMR : <https://console.aws.amazon.com/elasticmapreduce/home?region=us-east-1#quick-create:>

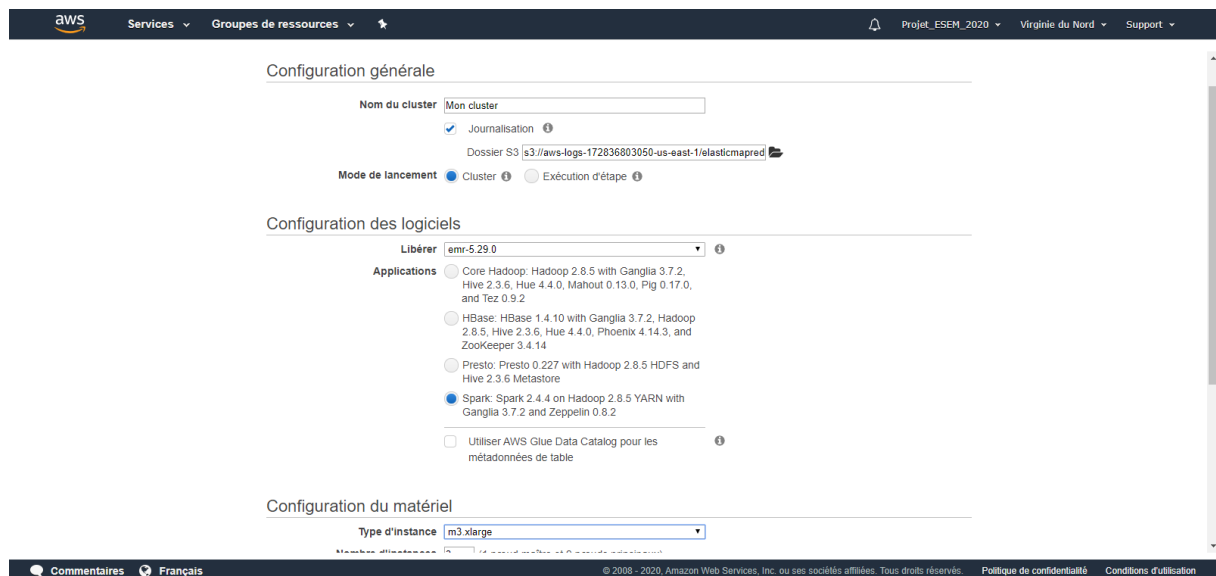
- On définit les paramètres adoptés :

Nom du cluster : Mon cluster

Applications : Spark

Type d'instance : m3.xlarge

Paire de clés EC2 : gdelKeyPair



Amazon EMR

Cluster : GDELT_ESEM_2020 Démarrage en cours

Récapitulatif Historique de l'application Surveillance Matériel Configurations Événements Étapes Actions d'amorçage

Connexions : --

DNS public principal : --

Balises : -- [Afficher tout/Modifier](#)

Récapitulatif

ID : j-3QVR4DIMSUGBP

Date de création : 22-02-2020 12:56 (UTC+1)

Temps écoulé : 0 secondes

Résiliation : Cluster waits automatique

Protection de la Désactivée [Modification](#)

résilience :

Détails de configuration

Étiquette de emr-5.19.1

version :

Distribution Amazon

Hadoop :

Applications : Ganglia 3.7.2, Spark 2.3.2, Zeppelin 0.8.0

URI de connexion : s3://aws-logs-172836803050-us-east-1/elasticmapreduce/

Vue Désactivée

cohérente EMRFS :

ID d'AMI --

personnalisée :

Réseau et matériel

Zone de disponibilité --

ID de sous-réseau [subnet-b2ab73bc](#) (subnet)

Maître : Mise en service 1 m1.large

Principal : Mise en service 2 m1.large

Tâche : --

Sécurité et accès

Nom de clé : GdelKeyPair

Profil EMR_EC2_DefaultRole

d'instance EC2

Commentaires Français

© 2008 - 2020, Amazon Web Services, Inc. ou ses sociétés affiliées. Tous droits réservés. Politique de confidentialité Conditions d'utilisation

GdelKeyPair.pem [Tout afficher](#)

- On rajoute une règle de firewall dans les groupes de sécurité pour le principale node pour permette la connexion SSH.

Amazon

Services Groupes de ressources

Projet_ESEM_2020 Virginie du Nord Support

New EC2 Experience Tell us what you think

Créer un groupe de sécurité Actions

search : sg-06df11976e483fb95 Ajouter filtre

Name	ID du groupe	Nom du groupe	ID de VPC	Propriétaire	Description
	sg-05113d80a312abdc	ElasticMapReduce-slave	vpc-20ac9f5a	172836803050	Slave group for Elastic MapReduce created on 2020-02-21T22:19:1
	sg-06df11976e483fb95	ElasticMapReduce-master	vpc-20ac9f5a	172836803050	Master group for Elastic MapReduce created on 2020-02-21T22:19:1

Supprimer le groupe de sécurité
Ajouter/Modifier des balises
Copier vers le nouveau
Modifier les règles entrantes
Modifier les règles sortantes

Groupe de sécurité: sg-06df11976e483fb95

Description Entrant Sortant Balises

Nom du groupe ElasticMapReduce-master

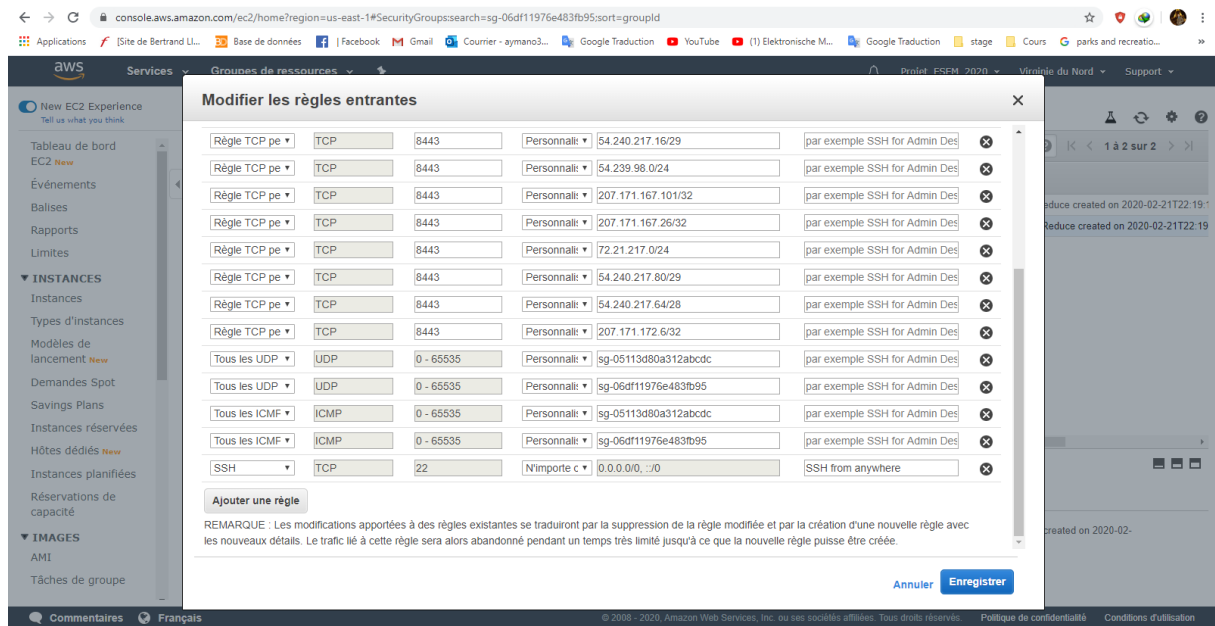
Description du groupe Master group for Elastic MapReduce created on 2020-02-21T22:19:12:200Z

ID du groupe sg-06df11976e483fb95

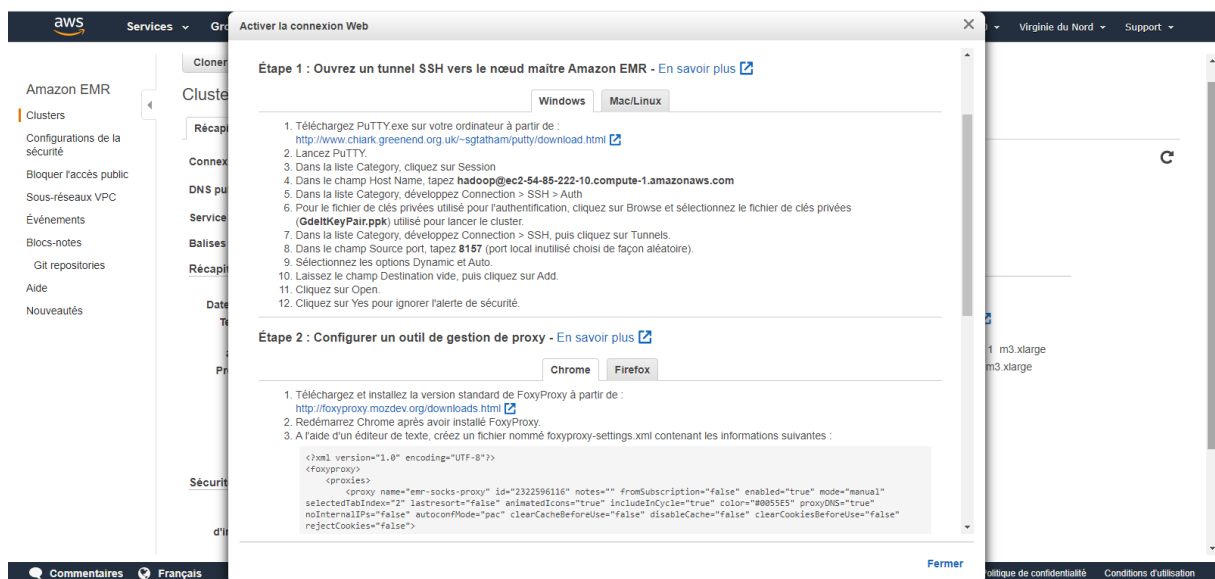
ID de VPC vpc-20ac9f5a

Commentaires Français

© 2008 - 2020, Amazon Web Services, Inc. ou ses sociétés affiliées. Tous droits réservés. Politique de confidentialité Conditions d'utilisation



- On a établi un tunnel SSH avec le nœud maître, afin d'accéder aux interfaces Web via <http://localhost:PORT>.
- On peut aussi utiliser un proxy web (FoxProxy) pour faire directement la translation master-public-dns-name :PORT => localhost:PORT.



Liste des services et les ports :

Name of interface	URI
Ganglia	http://master-public-dns-name/ganglia/
Hadoop HDFS NameNode	https://master-public-dns-name:50470/
Hadoop HDFS DataNode	https://coretask-public-dns-name:50475/
HBase	http://master-public-dns-name:16010/
Hue	http://master-public-dns-name:8888/
JupyterHub	https://master-public-dns-name:9443/
Livy	http://master-public-dns-name:8999/
Spark HistoryServer	http://master-public-dns-name:18080/
Tez	http://master-public-dns-name:8080/tez-ui
YARN NodeManager	http://coretask-public-dns-name:8042/
YARN ResourceManager	http://master-public-dns-name:8088/
Zeppelin	http://master-public-dns-name:8890/

- Lancement du cluster sur MobaXterm

The screenshot shows a Windows terminal window titled "hadoop@ip-172-31-5-224~". The terminal displays several SSH commands and their outputs:

```
[26/02/2020 13:45:46] /drives/c/Project2020
[Utilisateur_DESKTOP-61N1403] > ssh hadoop@ec2-3-228-2-142.compute-1.amazonaws.com -i
Warning: Permanently added 'ec2-3-228-2-142.compute-1.amazonaws.com' (RSA) to the list of known hosts.
Permission denied (publickey).

[26/02/2020 13:46:05] /drives/c/Project2020
[Utilisateur_DESKTOP-61N1403] > ssh hadoop@ec2-3-228-2-142.compute-1.amazonaws.com -i c:\Project2020\GdelKeyPair.pem
Warning: Identity file C:\Project2020\GdelKeyPair.pem not accessible: No such file or directory.
Permission denied (publickey).

[26/02/2020 13:46:21] /drives/c/Project2020
[Utilisateur_DESKTOP-61N1403] > ssh hadoop@ec2-3-228-2-142.compute-1.amazonaws.com -i C:\Project2020\GdelKeyPair.pem
Warning: Identity file C:\Project2020\GdelKeyPair.pem not accessible: No such file or directory.
Permission denied (publickey).

[26/02/2020 13:46:35] /drives/c/Project2020
[Utilisateur_DESKTOP-61N1403] > ssh hadoop@ec2-3-228-2-142.compute-1.amazonaws.com -i GdelKeyPair.pem
X11 forwarding request failed on channel 0
Last login: Wed Feb 26 12:33:22 2020
```

The terminal then displays the Amazon Linux logo and system information:

```
Amazon Linux AMI

https://aws.amazon.com/amazon-linux-ami/2019-03-release-notes/
43 package(s) needed for security, out of 75 available
Run "sudo yum update" to apply all updates.
```

A large ASCII art logo for Amazon Linux is displayed, followed by a detailed system status report:

```
EEEEEEEEEEEEEEEEEE M*****M ***** RRRRRRRRRRRRRR
E::::::::::::::::::E M:::MM M:::MM R:::RRRRRR:::R
EE:::EEEEEEEEEEE E M:::MM M:::MM R:::RRRRRR:::R
E:::E EEEEE M:::MM M:::MM RR:::R R:::R
E:::E M:::MM M:::MM R:::R R:::R
E:::EEEEEEEEEE M:::MM M:::MM R:::RRRRRR:::R
E:::MM M:::MM M:::MM R:::RR
E:::EEEEEEEEEE M:::MM M:::MM R:::RRRRRR:::R
E:::E M:::MM M:::MM R:::R R:::R
E:::E EEEEE M:::MM MM M:::MM R:::R
EE:::EEEEEEEEEEE M:::MM M:::MM R:::R
E::::::::::::::::::E M:::MM RR:::R R:::R
EEEEEEEEEEEEEEEEEE M*****M ***** RRRRRR RRRRRR
```

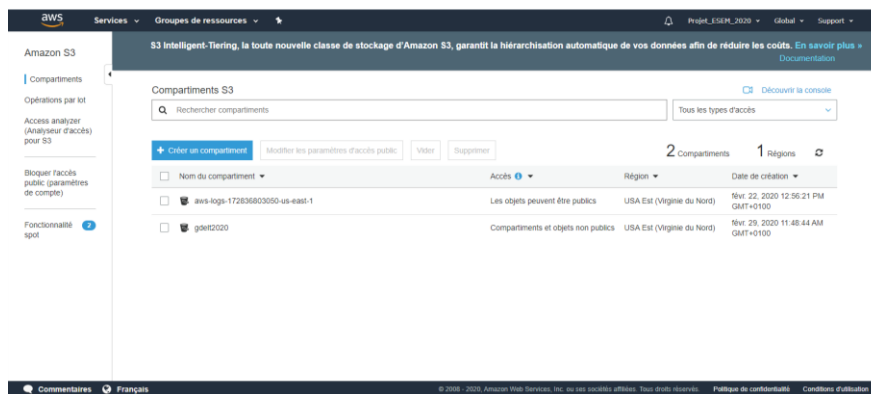
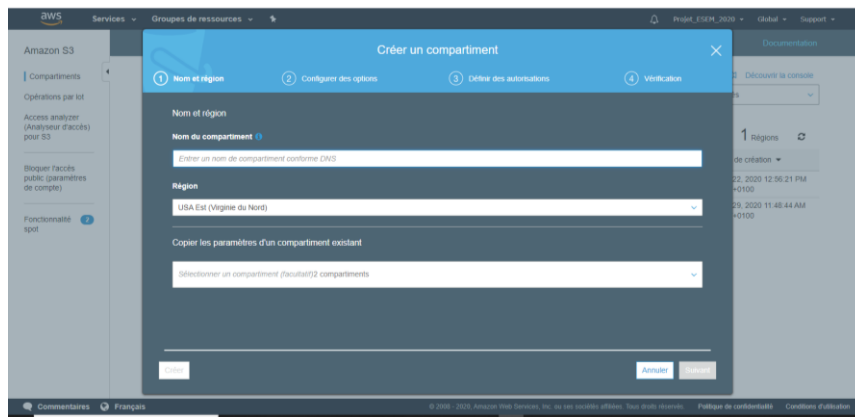
The prompt at the bottom indicates the user is now logged into the remote host:

```
r[hadoop@ip-172-31-5-224 ~]$
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

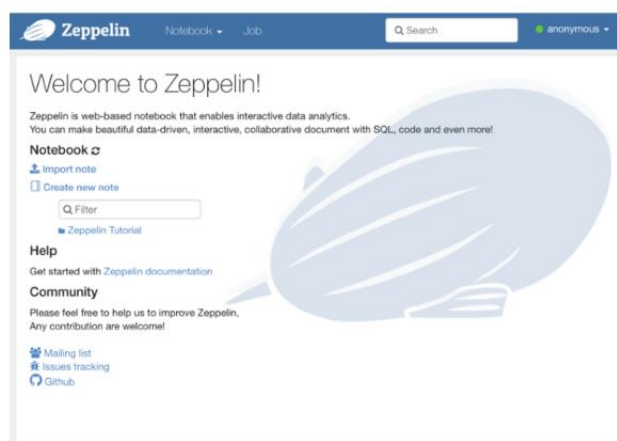
- Création d'un bucket AWS S3

- On va sur le lien : <https://s3.console.aws.amazon.com/s3/home?region=us-east-1>



- Connexion à l'interface Zeppelin :

Accéder au lien <http://master-public-dns-name:8890/> et importation du notebook « _esem_Gdelt_2020.json » dans Zeppelin.



3. ETL GDELT :

Dans notre notebook, on va télécharger les fichiers GDELT pour l'importation des packages

```
import sys.process._
import java.net.URL
import java.io.File
import java.io.File
import java.nio.file.{Files, StandardCopyOption}
import java.net.HttpURLConnection
import org.apache.spark.sql.functions._
import sqlContext.implicits._
import org.apache.spark.input.PortableDataStream
import java.util.zip.ZipInputStream
import java.io.BufferedReader
import java.io.InputStreamReader
import org.apache.spark.sql.SQLContext
import com.amazonaws.services.s3.AmazonS3Client
import com.amazonaws.auth.BasicAWSCredentials
import org.apache.spark.sql.types.IntegerType
import com.amazonaws.services.s3.AmazonS3Client
```

```
import sys.process._
import java.net.URL
import java.io.File
import java.io.File
import java.nio.file.{Files, StandardCopyOption}
import java.net.HttpURLConnection
import org.apache.spark.sql.functions._
import sqlContext.implicits._
import org.apache.spark.input.PortableDataStream
import java.util.zip.ZipInputStream
import java.io.BufferedReader
import java.io.InputStreamReader
import org.apache.spark.sql.SQLContext
import com.amazonaws.services.s3.AmazonS3Client
import com.amazonaws.auth.BasicAWSCredentials
import org.apache.spark.sql.types.IntegerType
import com.amazonaws.services.s3.AmazonS3Client
```

Took 5 sec. Last updated by anonymous at March 07 2020, 2:12:37 PM.

Téléchargement des fichiers masterfilelist.txt et masterfile_translation.txt

```
%pyspark
import urllib

print('Beginning file download with urllib2...')

url = 'http://data.gdeltproject.org/gdeltv2/masterfilelist.txt'
urllib.urlretrieve(url, '/tmp/masterfilelist.txt')
```

```
Beginning file download with urllib2...
('/tmp/masterfilelist.txt', <httplib.HTTPMessage instance at 0x7fb131b7cbd8>)
```

Took 2 sec. Last updated by anonymous at March 07 2020, 11:16:52 AM.

```
%pyspark
import urllib

print('Beginning file download with urllib2...')

url = 'http://data.gdeltproject.org/gdeltv2/masterfilelist-translation.txt', "/tmp/masterfilelist_translation.txt"
urllib.urlretrieve(url, '/tmp/masterfilelist_translation.txt')
```

```
Beginning file download with urllib2...
('/tmp/masterfilelist_translation.txt', <httplib.HTTPMessage instance at 0x7fb131b7c8c0>)
```

Took 1 sec. Last updated by anonymous at March 07 2020, 11:17:01 AM.

Récupération du fichier masterfile.txt et le spliter dans un RDD avec 3 colonnes qui sont size, hash et url

```
%pyspark
from pyspark.sql import Row
data_file = sc.textFile('s3://gdelt2020/masterfilelist.txt')
textSplit = data_file.map(lambda line : line.split(" "))
dataRDD = textSplit.map(lambda item : Row (size=item[0], hash=item[1], url=item[2]))
print("succes")
```

succes

Took 0 sec. Last updated by anonymous at March 07 2020, 11:17:12 AM.

```
%pyspark
from pyspark.sql import Row
data_file = sc.textFile('s3://gdelt2020/masterfilelist_translation.txt')
textSplit = data_file.map(lambda line : line.split(" "))
dataRDD_translation = textSplit.map(lambda item : Row (size=item[0], hash=item[1], url=item[2]))
print("succes")
```

succes

Took 0 sec. Last updated by anonymous at March 07 2020, 11:23:03 AM.

Affichage du fichier RDD

```
%pyspark
dataRDD.take(1)
```

SPARK JOB FINISHED

```
[Row(hash='297a16b493de7cf6ca809a7cc31d0b93', size='150383', url='http://data.gdeltproject.org/gdeltv2/20150218230000.export.CSV.zip'), Row(hash='bb27f78ba45f69a17ea6ed7755e9f8ff', size='318084', url='http://data.gdeltproject.org/gdeltv2/20150218230000.mentions.CSV.zip'), Row(hash='ea8dde0beb0ba98810a92db068c0ce99', size='10768507', url='http://data.gdeltproject.org/gdeltv2/20150218230000.gkg.csv.zip')]
```

Took 13 sec. Last updated by anonymous at March 07 2020, 2:57:20 PM

Conversion du fichier RDD to data frame

```
%pyspark
dataDF = dataRDD.toDF()
```

Took 19 sec. Last updated by anonymous at March 07 2020, 1:00:12 PM.

Affichage du data frame

```
%pyspark
dataDF.show(5, False)
```

hash	size	url
297a16b493de7cf6ca809a7cc31d0b93	150383	http://data.gdeltproject.org/gdeltv2/20150218230000.export.CSV.zip
bb27f78ba45f69a17ea6ed7755e9f8ff	318084	http://data.gdeltproject.org/gdeltv2/20150218230000.mentions.CSV.zip
ea8dde0beb0ba98810a92db068c0ce99	10768507	http://data.gdeltproject.org/gdeltv2/20150218230000.gkg.csv.zip
2a91041d7e72b0fc6a629e2ff867b240	149211	http://data.gdeltproject.org/gdeltv2/20150218231500.export.CSV.zip
dec3f427076b716a8112b9086c342523	339037	http://data.gdeltproject.org/gdeltv2/20150218231500.mentions.CSV.zip

only showing top 5 rows

Création d'un data frame avec une seule colonne « url »

```
%pyspark
urlDF = dataDF.select("url").show(5, False)

+-----+
|url|
+-----+
|http://data.gdeltproject.org/gdeltv2/20150218230000.export.CSV.zip|
|http://data.gdeltproject.org/gdeltv2/20150218230000.mentions.CSV.zip|
|http://data.gdeltproject.org/gdeltv2/20150218230000.gkg.csv.zip|
|http://data.gdeltproject.org/gdeltv2/20150218231500.export.CSV.zip|
|http://data.gdeltproject.org/gdeltv2/20150218231500.mentions.CSV.zip|
+-----+
only showing top 5 rows
```

Sélection de tous les urls « export »

```
%pyspark
from pyspark.sql.functions import col
newDf_export= dataDF.filter(col("url").like("%export%")).show(5, False)

+-----+-----+-----+
|hash|size|url|
+-----+-----+-----+
|297a16b493de7cf6ca809a7cc31d0b93|150383|http://data.gdeltproject.org/gdeltv2/20150218230000.export.CSV.zip|
|2a91041d7e72b0fc6a629e2ff867b240|149211|http://data.gdeltproject.org/gdeltv2/20150218231500.export.CSV.zip|
|12260e821823aae2da90882621feda18|149723|http://data.gdeltproject.org/gdeltv2/20150218233000.export.CSV.zip|
|a5298ce3c6df1a8a759c61b5c0b6f8bb|158842|http://data.gdeltproject.org/gdeltv2/20150218234500.export.CSV.zip|
|c4268d558bb22c02b3c132c17818c68b|362610|http://data.gdeltproject.org/gdeltv2/20150219000000.export.CSV.zip|
+-----+-----+-----+
only showing top 5 rows
```

Sélection de tous les urls « mentions »

```
%pyspark
from pyspark.sql.functions import col
newDf_mentions= dataDF.filter(col("url").like("%mentions%")).show(5, False)

+-----+-----+-----+
|hash|size|url|
+-----+-----+-----+
|bb27f78ba45f69a17ea6ed7755e9f8ff|318084|http://data.gdeltproject.org/gdeltv2/20150218230000.mentions.CSV.zip|
|dec3f427076b716a8112b9086c342523|339037|http://data.gdeltproject.org/gdeltv2/20150218231500.mentions.CSV.zip|
|744acad14559f2781a8db67715d63872|357229|http://data.gdeltproject.org/gdeltv2/20150218233000.mentions.CSV.zip|
|dd322c888f28311aca2c735468405551|374528|http://data.gdeltproject.org/gdeltv2/20150218234500.mentions.CSV.zip|
|e7f464a7a451ad2af6e9c8fa24f0ccea|287807|http://data.gdeltproject.org/gdeltv2/20150219000000.mentions.CSV.zip|
+-----+-----+-----+
only showing top 5 rows
```

Création d'une liste avec tous les urls de 2019

```
1 list_2019.select("url").repartition(100).foreach( r=> {
2     val URL = r.getAs[String](0)
3     val fileName = r.getAs[String](0).split("/").last
4     val dir = "/FileStore/tables/"
5     val localFileName = dir + fileName
6     fileDownloader(URL, localFileName)
7     val localFile = new File(localFileName)
8 })
```

Cmd 7

```
1 display(list_2019)
```

size	hash	url
149183	50e8534eece514f98fb184e6b36ecd94	http://data.gdeltproject.org/gdeltv2/20190101000000.export.CSV.zip
136583	cb61605430b6813d9ae2d60f72cf4c8f	http://data.gdeltproject.org/gdeltv2/20190101000000.mentions.CSV.zip
6415898	5837cd0fcdbfe0f37d528e262bb4ebe2	http://data.gdeltproject.org/gdeltv2/20190101000000.gkg.csv.zip
116538	b4869e9de5039dbd0db3841b826362cf	http://data.gdeltproject.org/gdeltv2/20190101001500.export.CSV.zip
139610	6dbea3ed9bfee3d824ebf84f3a33a140	http://data.gdeltproject.org/gdeltv2/20190101001500.mentions.CSV.zip
6769390	d8ef6f41fb4859456f0a8f5f51a47c4d	http://data.gdeltproject.org/gdeltv2/20190101001500.gkg.csv.zip
103092	4ed6e5d871807c2791710c2a5a3f0c07	http://data.gdeltproject.org/gdeltv2/20190101003000.export.CSV.zip
131103	2d3c9ffe3133666b5d20916b801310e4	http://data.gdeltproject.org/gdeltv2/20190101003000.mentions.CSV.zip
5744400	b27ae501e2ee320be545eb301e67156	http://data.gdeltproject.org/gdeltv2/20190101003000.gkg.csv.zip

Showing the first 1000 rows.

Command took 2.84 seconds -- by mansour.hmdn@gmail.com at 07/03/2020 à 12:14:30 on unknown cluster

Les urls « mentions » de l'année 2019

```
1 import org.apache.spark.sql.functions._
2 val list_2019_mentions = list_2019.where(col("url").like("%mentions%"))
3 display(list_2019_mentions)
```

list_2019_mentions: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [size: string, hash: string ... 1 more fields]

size	hash	url
136583	cb61605430b6813d9ae2d60f72cf4c8f	http://data.gdeltproject.org/gdeltv2/20190101000000.mentions.CSV.zip
139610	6dbea3ed9bfee3d824ebf84f3a33a140	http://data.gdeltproject.org/gdeltv2/20190101001500.mentions.CSV.zip
131103	2d3c9ffe3133666b5d20916b801310e4	http://data.gdeltproject.org/gdeltv2/20190101003000.mentions.CSV.zip
133070	90b62c33e6bbe736ffa5abfd6d57104d	http://data.gdeltproject.org/gdeltv2/20190101004500.mentions.CSV.zip
152346	e2cb3f8076677808f377f04411bdb556	http://data.gdeltproject.org/gdeltv2/20190101010000.mentions.CSV.zip
119871	fbadb5bdc7fa4c3074413e0ced156bcd	http://data.gdeltproject.org/gdeltv2/20190101011500.mentions.CSV.zip
150973	a596fd82ca05e9a72512502d6bd7cd85	http://data.gdeltproject.org/gdeltv2/20190101013000.mentions.CSV.zip
105685	befe08384614b2c48530d9d7fc8629df	http://data.gdeltproject.org/gdeltv2/20190101014500.mentions.CSV.zip
110436	04c08be29e1e54ee06e41050a1b30	http://data.gdeltproject.org/gdeltv2/20190101020000.mentions.CSV.zip

Showing the first 1000 rows.

```
1 import org.apache.spark.sql.functions._
2 val urldf_mentions = list_2019_mentions.select($"url")
```

Cancel ** Waiting to run...

Cmd 10

```
1 display(urldf_mentions)
```

url
http://data.gdeltproject.org/gdeltv2/20190101000000.mentions.CSV.zip
http://data.gdeltproject.org/gdeltv2/20190101001500.mentions.CSV.zip
http://data.gdeltproject.org/gdeltv2/20190101003000.mentions.CSV.zip
http://data.gdeltproject.org/gdeltv2/20190101004500.mentions.CSV.zip
http://data.gdeltproject.org/gdeltv2/20190101010000.mentions.CSV.zip
http://data.gdeltproject.org/gdeltv2/20190101011500.mentions.CSV.zip
http://data.gdeltproject.org/gdeltv2/20190101013000.mentions.CSV.zip
http://data.gdeltproject.org/gdeltv2/20190101014500.mentions.CSV.zip
http://data.gdeltproject.org/gdeltv2/20190101020000.mentions.CSV.zip

Showing the first 1000 rows.

Command took 1.51 seconds -- by mansour.hmdn@gmail.com at 07/03/2020 à 12:16:50 on unknown cluster

Création d'un alarme de consommation : On a créé une alarme de facturation qui envoie un message électronique lorsque les frais estimés pour AWS dépassent un seuil spécifié (100 euros)

The screenshot shows the AWS CloudWatch Alarms console. At the top, there are three notification banners: a blue one about user feedback, a green one stating 'Alarme Alarm_Conso créée avec succès.' (Alarm Alarm_Conso created successfully), and a blue one about SNS subscription confirmations. Below these, the 'Alarms (1)' section is visible. It includes a search bar, a 'Masquer les alarmes AutoScaling' checkbox, and a table of alarms. The table has columns for 'Nom', 'État', 'Conditions', and 'Actions'. One alarm, 'Alarm_Conso', is listed with the state 'Données insuffisantes' (Insufficient data) and the condition 'EstimatedCharges > 100 pour 1 points de données dans 6 heures'. The action for this alarm is 'Confirmation en attente' (Confirmation pending).

Nom	État	Conditions	Actions
Alarm_Conso	Données insuffisantes	EstimatedCharges > 100 pour 1 points de données dans 6 heures	Confirmation en attente

