|  |  |
| --- | --- |
| Create a Virtual Private Cloud(VPC) with both private and public subnets |  |
|  |  |
| Create a second private subnet. Two subnets are required for RDS DB Subnet Group. |  |
| Create Security Group for a public web server that allows SSH from the Internet. |  |
| Security Group is created with correct ports and is linked to the VPC previously created. |  |
| Create a security group that has private access to an RDS Database Instance. |  |
| Create a Subnet group from previously created Subnets. These will be the designated Database subnets for a DB Instance created under the specified VPC. |  |
| Add the two private Subnets to the subnet group. Ensure that the availability zone is different for each. |  |
| Create a MySQL Database. |  |
| This DB will be created as “Dev/Test”. As described, this will serve as a development level Database but not intended for production use. |  |
| The DB instance will set to be large considering the amount of data intended to be loaded. A total of 150 gb of storage will be allocated.  Launch the DB as a multi-availability zone deployment with a standby instance. Connect the DB to previously created VPC and Subnet Group. |  |
|  |  |
| Launch an Ubuntu Instance for DB Access. This instance will be extra large as it will facilitate the loading of very large datasets into the MySQL database. |  |
| Attach Ubuntu Instance to an Auto Scaling Group |  |
| Modify volume attached to instance to 200 gb |  |
| Using the following commands, view the distribution of memory across the instance. In order to utilize the additional storage, it must be distributed to the working directory of the instance. “Grow” the xvda1 directory to absorb additional memory  **lsblk**  **df -h**  **sudo growpart /dev/xvda 1**  **sudo resize2fs /dev/xvda1** |  |
| GDELT uploads files daily starting in 2013. We will use the WGET command to download every daily upload csv into our Ubuntu Instance. |  |
| Create a new directory to hold GDELT data. Use WGET to download every zip file in the web directory. |  |
| A total of 2553 files are uploaded, for a total of 26 gb in zip files |  |
| Unzip every file and delete the original |  |

|  |  |
| --- | --- |
| After unzipping, the events folder houses nearly 150 gb of data |  |
| Create a shell script that accesses the Database and create a table for every year of events in our events directory. |  |
| Next, create a shell script that upload all ~2500 csv files into the Database tables by year |  |
|  |  |
| Download all regional curated files from acled and store in local directory |  |
| Create an S3 bucket to store ACLED Data. |  |
| Upload all ACLED Data |  |
| Install Fuse in the Instance, access S3 bucket and download ACLED Data into instance |  |
| Create shell script that creates table in ACLED database to store all events |  |
| Create shell script to upload all ACLED data to |  |
| We now have a database that holds ~ 750,000 ACLED events |  |
| Create IAM roles for group member access |  |
| Create user access to EC2 Instance |  |
| Grant password authentication on EC2 Instance |  |
|  |  |
| Create a new table from events\_2019 where the main actor is the United States |  |
|  |  |
| Show number of US events by Month |  |
| Create a new table from US events in 2019 where Afghanistan is the second actor country.  There are ~31,000 interactions captured between the United States and Afghanistan in 2019 with a pretty even monthly distribution throughout the year. |  |
| Check out the average Goldstein Scale per month for the United States/Afghanistan interactions. |  |