AWS CLI AND PYTHON MODULE BOTO

WHAT WE'LL COVER

In this section we will learn other methods to access to AWS. **AWS CLI** which stands for **Command-Line Interface,** and Python module named **boto3**. So far, we have used mainly the web console which is the **GUI** (**Graphical User Interface**) of AWS. The **AWS CLI** and Python module **boto3** give us more power and capabilities to interact with AWS and also script commands.

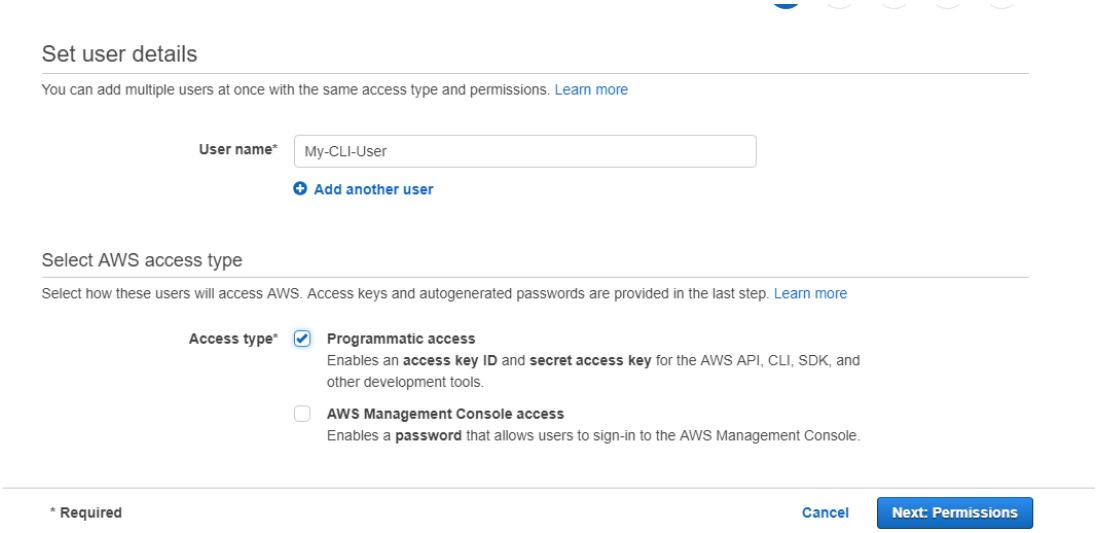
AWS CLI

You can install the AWS Tools both on a Laptop, Desktop, EC2 instances. The command line is really powerful. The AWS CLI is pre-installed on the AWS Linux AMI.

LAB

Let's start by creating a user called **myCliUser** and attach **AdministratorAccess** policy to it.

Go to IAM, Users, Add User, name it **My-CLI-User**, choose **Programmatic access** and hit **Next**



In the next screen you can just add it to the **system-admins** group, but for practicing purposes go to **Attach Existing Policies Directly** and choose **AdministratorAccess,** hit **Next** and then **Create User.**

You will get **Access ID** and **Secret access key**. You can save them by pressing **Download .csv** or just by pressing show and copying the keys. We now have a user the has administrative access to our AWS account.

1. Connect to the instance public IP address using SSH
2. Then elevate yourself to user root by running **sudo su –**
3. Type yes if needed.
4. Run **aws s3 ls**

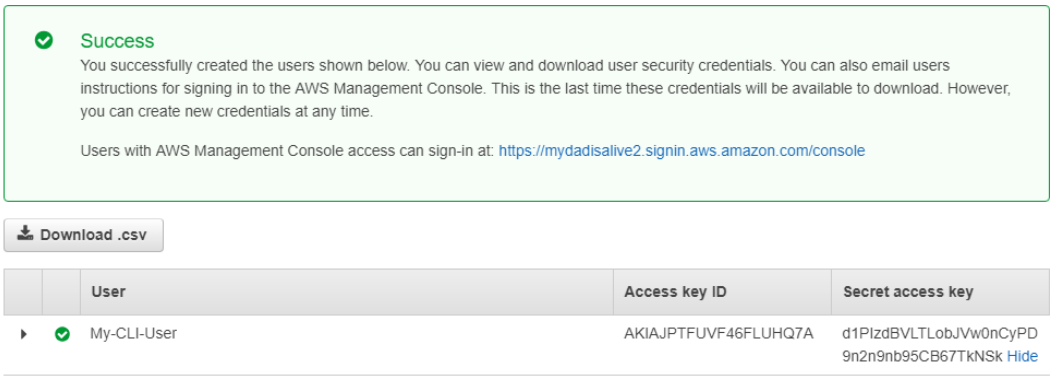
5. You will get an error that it cannot locate your credentials. Run **aws configure**

```
avicii@Xubuntu-Ansible:~$ ssh -i /media/sf_Xubuntu4/MyNVKeyPair.pem ec2-user@3.89.112.92
Last login: Thu Mar  7 13:26:34 2019 from 77.126.36.13

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-83-211 ~]$ sudo su -
Last login: Thu Mar  7 13:26:39 UTC 2019 on pts/0
[root@ip-172-31-83-211 ~]# aws s3 ls
Unable to locate credentials. You can configure credentials by running "aws configure".
[root@ip-172-31-83-211 ~]# aws configure
AWS Access Key ID [None]:
```

6. Paste you **Access Key ID** and **Secret Key** of the new user you created. They were given to you on this screen or you have downloaded them to .csv file or saved them in your notepad. For region enter **us-east-1** and for output format just type enter.




```
[root@ip-172-31-83-211 ~]# aws configure
AWS Access Key ID [None]: AKIAJPTFUVF46FLUHQ7A
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]:
[root@ip-172-31-83-211 ~]#
```

7. Now if you type **aws s3 ls** you should be able to see you S3 bucket list

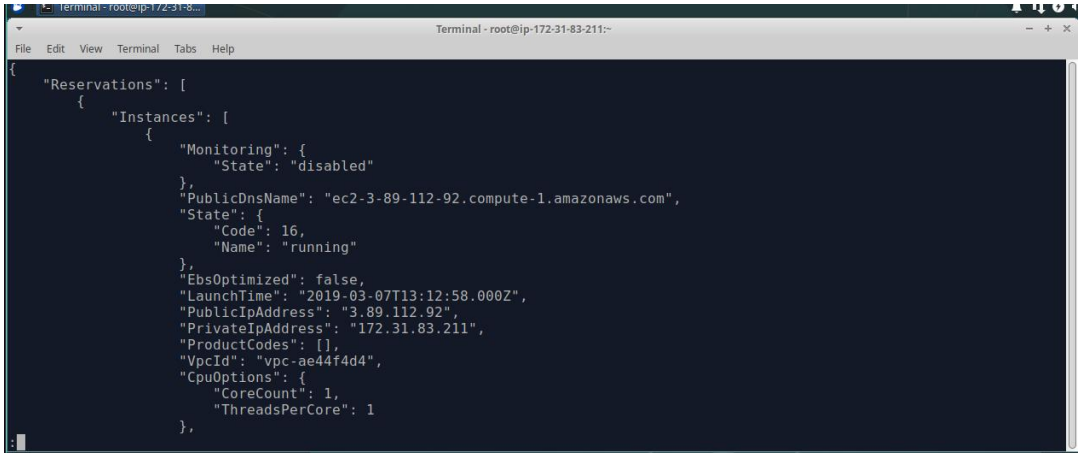
```
Default output format [None]:
[root@ip-172-31-83-211 ~]# aws s3 ls
2019-03-03 16:42:53 aws-codestar-us-east-1-792584129020
2019-03-03 16:43:31 aws-codestar-us-east-1-792584129020-test-pipe
2019-01-30 20:15:44 cf-templates-ia8a6ddbb8gi-us-east-1
2019-02-03 21:44:21 elasticbeanstalk-us-east-1-792584129020
2019-01-13 13:02:15 mydadisalive-accelerated
2019-01-12 17:20:26 mydadisalive-glacier
2019-03-05 13:22:28 mydadisalive-mysydneybucket
2019-01-23 15:27:59 mydadisalive-polly-mp3s
2019-01-23 15:27:33 mydadisalive-polly-website
2019-01-12 17:13:19 mydadisalive-test
2019-03-05 10:18:04 mydadisalive-testbucket
2019-01-12 17:43:51 mydadisalive-tokyo
2019-01-13 13:08:30 mydadisalive-website
2019-01-22 15:17:59 mydadisalive-website-bucket
2019-01-31 15:17:57 mydadisalive-wpcode
2019-01-31 15:18:34 mydadisalive-wpmedia
2019-03-05 15:06:25 mydadisalive.com
2019-03-05 15:55:43 www.mydadisalive.com
[root@ip-172-31-83-211 ~]#
```

8. If you run **aws s3 help** you will get help screen on the command
9. So you can see for example available commands

10. Type **q** or **ctrl-c** to exit that screen
11. You will see you have new two file inside your home directory inside. aws subdirectory which contain the information you entered using the **aws configure**

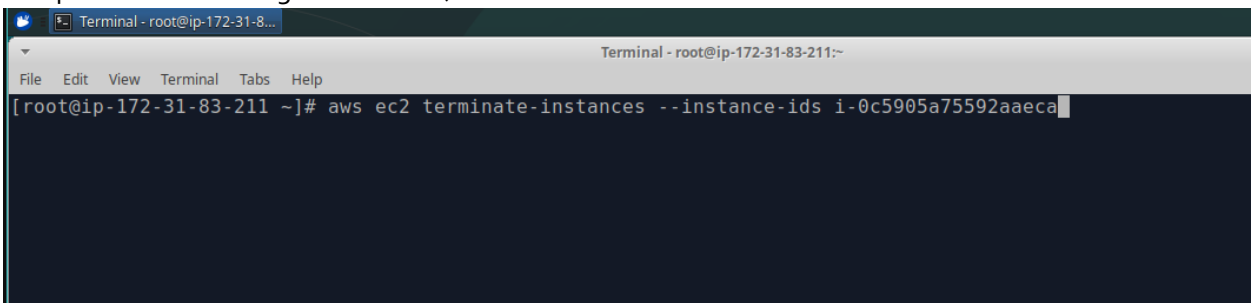
```
[root@ip-172-31-83-211 ~]# cat .aws/config
[default]
region = us-east-1
[root@ip-172-31-83-211 ~]# cat .aws/credentials
[default]
aws_access_key_id = AKIAJPTFUVF46FLUHQ7A
aws_secret_access_key = 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
[root@ip-172-31-83-211 ~]#
```

12. Now run **aws ec2 describe-instances**
13. You will get a list describing the instances run on your AWS account with many details about them in JSON format

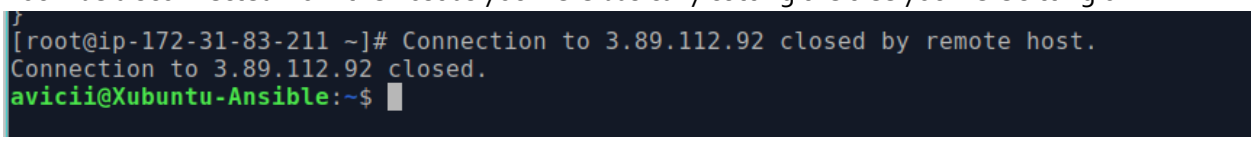


14. Try to locate the instance-id of your instance in this information. If you get confused try using **grep** or less and then search it.

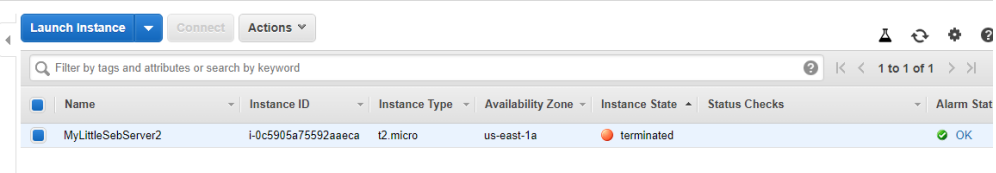
15. Now let's try to terminate the instance using the CLI
16. Run **aws ec2 terminate-instances --instance-ids i-0c5905a75592aaeca** (you can use tab completion with the arguments here)



17. You'll be disconnected from the host as you were basically cutting the tree you were sitting on



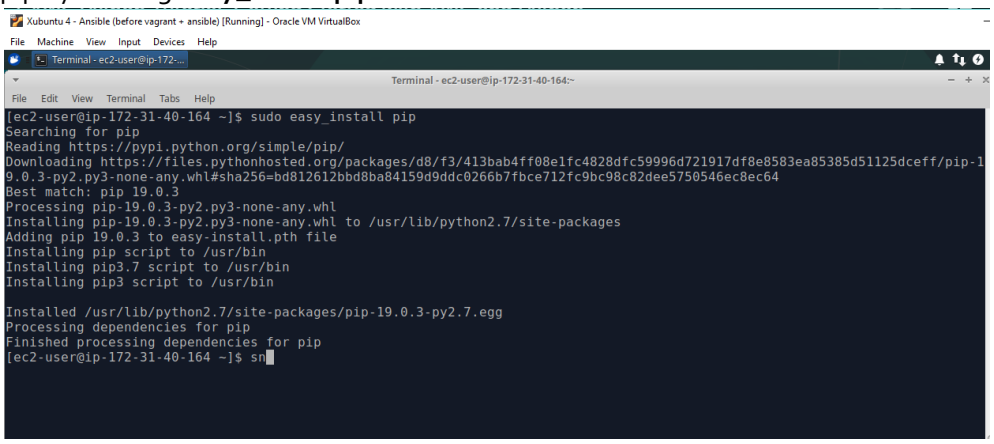
18. And if we go to the AWS console, we'll see indeed that the instance is Terminated



AWS PYTHON MODULE BOTO3

To use AWS via python you need to install and then import a module called **boto**. It uses the same mechanism of **Access Key** and **Secret Key** that is inside the .**aws** directory that **aws configure** created.

1. Run **sudo su -**
2. Install pip by running **easy_install pip**



3. Run **pip install boto3**
4. Create a script named **list_s3_bucket.py** with the following code

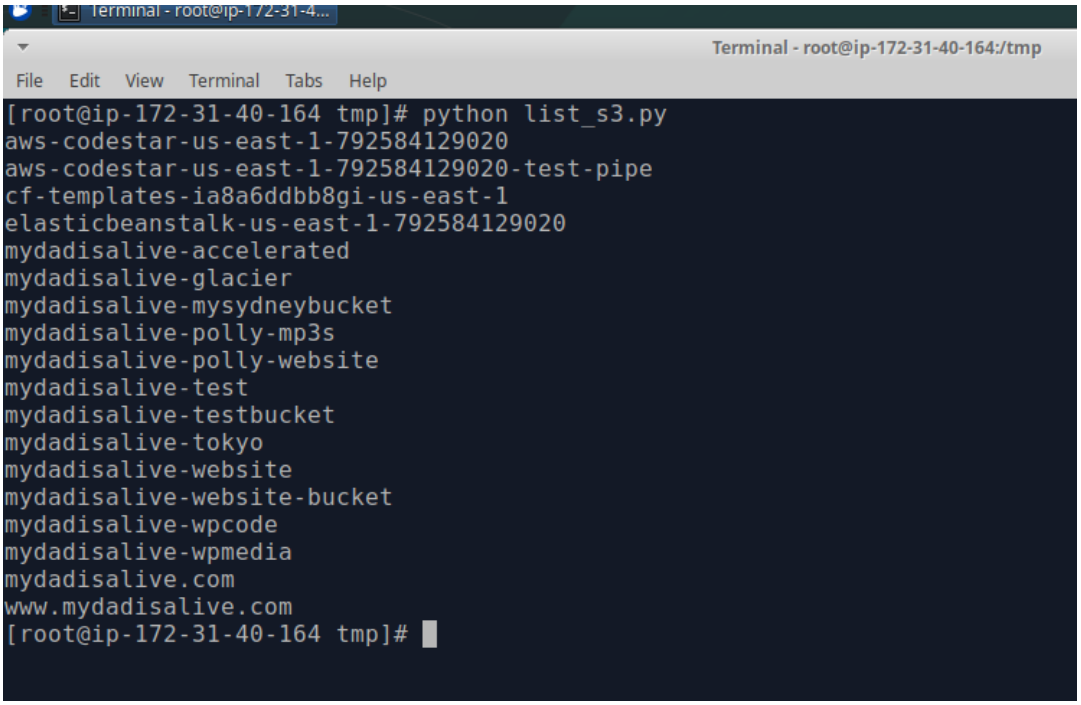
```
#!/bin/python

import boto3

s3=boto3.resource('s3')
for bucket in s3.buckets.all():
        print(bucket.name)
```

5. Run it by running **python list_s3_bucket.py**

6. You should see an output with all of your bucket names



EXERCISE: LIST ALL OF YOU EC2 INSTANCES

Try writing a similar code to list all of your EC2 instances.

SUMMARY

In this section we have learned other about accessing AWS through the CLI and also through python module boto3. We have seen way to list buckets and instances. Controlling those two methods will brings us power in automating things in the AWS environment.