**AWS-CLI Primer Workshop**

**Student:** Cristian David Polo Garrido

In this workshop, you will learn how to use AWS-CLI to create an EC2 instance.

**Prerequisites**

1. Install the AWS-CLI application.

A screenshot of a computer

AI-generated content may be incorrect.

1. Configure the AWS-CLI application:

aws configure

Example configuration:

AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE

AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY

Default region name [None]: us-west-2

Default output format [None]: json

Screens screenshots of a computer screen

AI-generated content may be incorrect.

**Step 1: Create a Key Pair for EC2**

aws ec2 create-key-pair --key-name MyKeyPair --query 'KeyMaterial' --output text > MyKeyPair.pem



ls

Output:

MyKeyPair.pem

A computer screen shot of a black screen

AI-generated content may be incorrect.

Make the private key readable only by you:

chmod 400 MyKeyPair.pem

ls -la

Output:

-r-------- 1 user staff 1675 Oct 9 20:39 MyKeyPair.pem

Check the fingerprint:

aws ec2 describe-key-pairs --key-name MyKeyPair

**Step 2: Create a Security Group**

First, check for VPCs configured in your account.

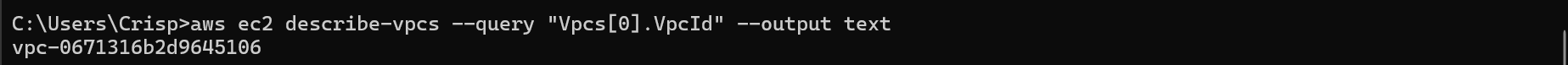
aws ec2 create-security-group --group-name my-sg-cli --description "My security group" --vpc-id vpc-xxxxxxxx

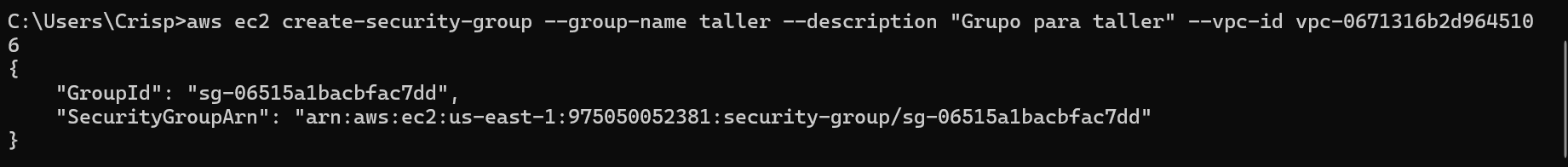
Example output:

{

"GroupId": "sg-01f4c77b81e9dc434"

}





List security groups:

aws ec2 describe-security-groups --group-ids sg-01f4c77b81e9dc434

**Add Ingress Rules**

Check your public IP address (optional for restricted access):

curl https://checkip.amazonaws.com

Example output:

186.96.109.58

Allow RDP (port 3389):

aws ec2 authorize-security-group-ingress --group-id sg-01f4c77b81e9dc434 --protocol tcp --port 3389 --cidr 0.0.0.0/0

A computer screen with white text

AI-generated content may be incorrect.

Allow SSH (port 22):

aws ec2 authorize-security-group-ingress --group-id sg-01f4c77b81e9dc434 --protocol tcp --port 22 --cidr 0.0.0.0/0

A black screen with white text

AI-generated content may be incorrect.

**Step 3: Create the Instance**

Before creating the instance, ensure you have a subnet configured.

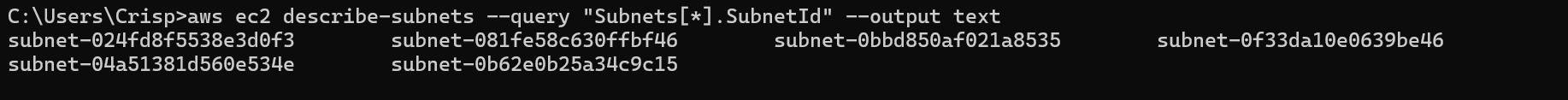
Run the following command to launch a **t2.micro** instance:

aws ec2 run-instances --image-id ami-032930428bf1abbff --count 1 --instance-type t2.micro --key-name MyKeyPair --security-group-ids sg-01f4c77b81e9dc434 --subnet-id subnet-1175cf1d

A computer screen with white text

AI-generated content may be incorrect.

To know the subnets, I typed this command:



Now I list the instances that are running, seeing that indeed the one I just created is up and running:

A black screen with white text

AI-generated content may be incorrect.

**Step 4: Connect to the Instance**

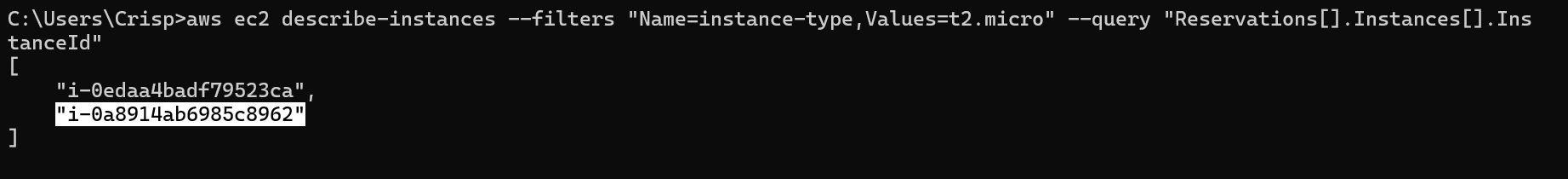
ssh -i "MyKeyPair.pem" [ec2-user@ec2-34-204-197-22.compute-1.amazonaws.com](mailto:ec2-user@ec2-34-204-197-22.compute-1.amazonaws.com)

A computer screen with white text

AI-generated content may be incorrect.

**Step 5: List Your Instances**

aws ec2 describe-instances --filters "Name=instance-type,Values=t2.micro" --query "Reservations[].Instances[].InstanceId"



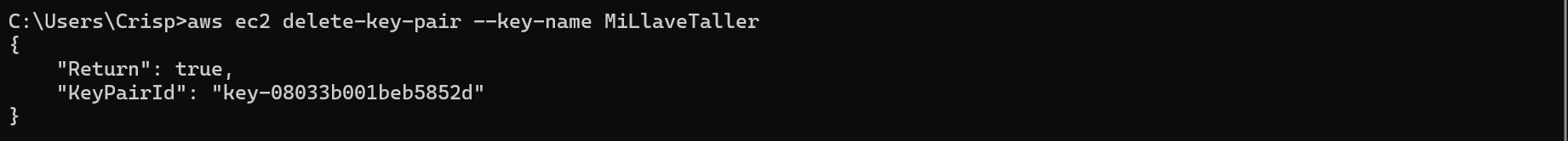
A screenshot of a computer

AI-generated content may be incorrect.

**Step 6: Clean Up**

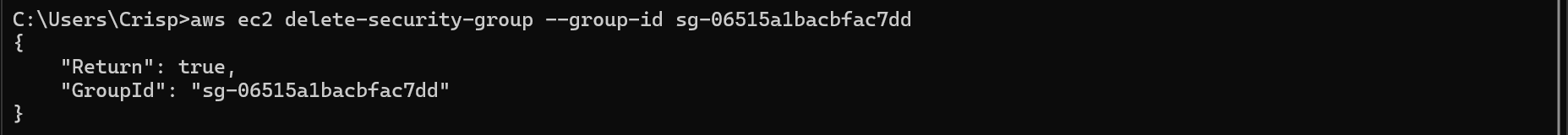
Delete the key pair:

aws ec2 delete-key-pair --key-name MyKeyPair



Delete the security group:

aws ec2 delete-security-group --group-id sg-903004f8



Terminate the instance:

aws ec2 terminate-instances --instance-ids i-07d0ddb36ea3e65a4

A screen shot of a computer

AI-generated content may be incorrect.

**Conclusion**

Congratulations! You just learned how to automatically deploy an EC2 instance on AWS.

**References**

* [AWS CLI User Guide](https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-welcome.html)