Camilo Andrés Quintero Rodríguez

Primero realizamos la configuración inicial con las llaves de acceso, la región y formato:

```
Microsoft Windows [Versión 10.0.26100.6584]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Milo>aws configure
AWS Access Key ID [****************PHW4]:
AWS Secret Access Key [****************tF7+]:
WTtUZMDModFdG
Default region name [us-east-1]:
Default output format [json]:
```

Step 1: Create a Key Pair for EC2

```
C:\Users\Milo\.aws>aws ec2 create-key-pair --key-name MyKeyPair --query '
KeyMaterial' --output text > MyKeyPair.pem
```

Podemos ver que las llaves se han creado:

```
 Directorio de C:\Users\Milo\.aws

22/09/2025  09:46p. m.    <DIR>          .
18/09/2025  03:14p. m.    <DIR>          ..
22/09/2025  09:38p. m.                46 config
22/09/2025  09:44p. m.               941 credentials
22/09/2025  09:47p. m.                13 MyKeyPair.pem
               3 archivos          1.000 bytes
               2 dirs  14.258.839.552 bytes libres
```

Step 2: Create a Security Group

Primero miramos nuestros VPCs:

```
C:\Users\Milo\.aws>aws ec2 describe-vpcs
{
    "Vpcs": [
        {
            "OwnerId": "767398145456",
            "InstanceTenancy": "default",
            "CidrBlockAssociationSet": [
                {
                    "AssociationId": "vpc-cidr-assoc-079a3ea54274bd937",
                    "CidrBlock": "172.31.0.0/16",
                    "CidrBlockState": {
                        "State": "associated"
                    }
                }
            ],
            "IsDefault": true,
            "BlockPublicAccessStates": {
                "InternetGatewayBlockMode": "off"
            },
            "VpcId": "vpc-01c3ca8322fbb43dc",
            "State": "available",
            "CidrBlock": "172.31.0.0/16",
            "DhcpOptionsId": "dopt-08634978fd76fcc99"
        }
    ]
}
```

Creamos el grupo de seguridad:

```
C:\Users\Milo\.aws>aws ec2 create-security-group --group-name my-sg-cli --
-description "My security group" --vpc-id vpc-01c3ca8322fbb43dc
{
    "GroupId": "sg-0ab162d2371c289b1",
    "SecurityGroupArn": "arn:aws:ec2:us-east-1:767398145456:security-grou
p/sg-0ab162d2371c289b1"
}
```

Listamos el grupo creado:

```
C:\Users\Milo\.aws>aws ec2 describe-security-groups --group-ids sg-0ab162
d2371c289b1
{
    "SecurityGroups": [
        {
            "GroupId": "sg-0ab162d2371c289b1",
            "IpPermissionsEgress": [
                {
                    "IpProtocol": "-1",
                    "UserIdGroupPairs": [],
                    "IpRanges": [
                        {
                            "CidrIp": "0.0.0.0/0"
                        }
                    ],
                    "Ipv6Ranges": [],
                    "PrefixListIds": []
                }
            ],
            "VpcId": "vpc-01c3ca8322fbb43dc",
            "SecurityGroupArn": "arn:aws:ec2:us-east-1:767398145456:secur
ity-group/sg-0ab162d2371c289b1",
            "OwnerId": "767398145456",
            "GroupName": "my-sg-cli",
            "Description": "My security group",
            "IpPermissions": []
        }
    ]
}
```

Add Ingress Rules

```
C:\Users\Milo\.aws>curl https://checkip.amazonaws.com
204.199.126.185
```

Permitimos RDP por el puerto 3389:

```
C:\Users\Milo\.aws>aws ec2 authorize-security-group-ingress --group-id sg
-0ab162d2371c289b1 --protocol tcp --port 3389 --cidr 0.0.0.0/0
{
    "Return": true,
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-07e5d1fe528c407b9",
            "GroupId": "sg-0ab162d2371c289b1",
            "GroupOwnerId": "767398145456",
            "IsEgress": false,
            "IpProtocol": "tcp",
            "FromPort": 3389,
            "ToPort": 3389,
            "CidrIpv4": "0.0.0.0/0",
            "SecurityGroupRuleArn": "arn:aws:ec2:us-east-1:767398145456:s
ecurity-group-rule/sgr-07e5d1fe528c407b9"
        }
    ]
}
```

Permitimos SSH por el puerto 22:

```
C:\Users\Milo\.aws>aws ec2 authorize-security-group-ingress --group-id sg
-0ab162d2371c289b1 --protocol tcp --port 22 --cidr 0.0.0.0/0
{
    "Return": true,
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-0dc6cc0b8819d55c0",
            "GroupId": "sg-0ab162d2371c289b1",
            "GroupOwnerId": "767398145456",
            "IsEgress": false,
            "IpProtocol": "tcp",
            "FromPort": 22,
            "ToPort": 22,
            "CidrIpv4": "0.0.0.0/0",
            "SecurityGroupRuleArn": "arn:aws:ec2:us-east-1:767398145456:s
ecurity-group-rule/sgr-0dc6cc0b8819d55c0"
        }
    ]
}
```

Step 3: Create the Instance

Consultamos primero las AMIs disponibles:

```
C:\Users\Milo\.aws>aws ec2 describe-images --owners amazon --filters "Nam
e=name,Values=amzn2-ami-hvm-*-x86_64-gp2" --query "Images[*].[ImageId,Nam
e]" --output table
-------------------------------------------------------------------------
|                           DescribeImages                              |
+-----------------------+-----------------------------------------------+
|  ami-0023921b4fcd5382b |  amzn2-ami-hvm-2.0.20250902.3-x86_64-gp2  |
|  ami-0254b2d5c4c472488 |  amzn2-ami-hvm-2.0.20250915.0-x86_64-gp2  |
|  ami-083e865b97bdf1c1b |  amzn2-ami-hvm-2.0.20250721.2-x86_64-gp2  |
|  ami-0871b7e0b83ae16c4 |  amzn2-ami-hvm-2.0.20250707.0-x86_64-gp2  |
|  ami-0ad253013fad0a42a |  amzn2-ami-hvm-2.0.20250728.1-x86_64-gp2  |
|  ami-0c12c782c6284b66c |  amzn2-ami-hvm-2.0.20250804.1-x86_64-gp2  |
|  ami-0e95a5e2743ec9ec9 |  amzn2-ami-hvm-2.0.20250818.2-x86_64-gp2  |
|  ami-0e2c86481225d3c51 |  amzn2-ami-hvm-2.0.20250808.1-x86_64-gp2  |
+-----------------------+-----------------------------------------------+
```

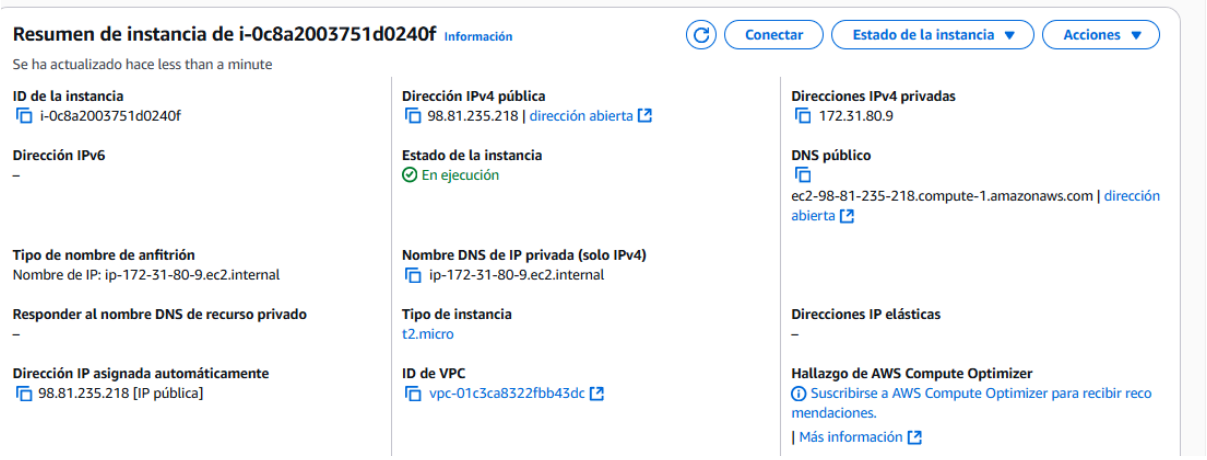
Consultamos el id de la subnet:

```
C:\Users\Milo\.aws>aws ec2 describe-subnets --query "Subnets[*].[SubnetId
,VpcId,CidrBlock,AvailabilityZone]" --output table
-------------------------------------------------------------------------
-------------
|                                DescribeSubnets
            |
+--------------------------+-------------------------+-----------------+
-------------+
|  subnet-00cb88909a70ef17a |  vpc-01c3ca8322fbb43dc  |  172.31.80.0/20 |
  us-east-1c |
|  subnet-083be637256a8f93d |  vpc-01c3ca8322fbb43dc  |  172.31.64.0/20 |
  us-east-1f |
|  subnet-035208338957d0c58 |  vpc-01c3ca8322fbb43dc  |  172.31.32.0/20 |
  us-east-1a |
|  subnet-0241ed5dc6c604764 |  vpc-01c3ca8322fbb43dc  |  172.31.0.0/20  |
  us-east-1b |
|  subnet-09647e875060674fa |  vpc-01c3ca8322fbb43dc  |  172.31.48.0/20 |
  us-east-1e |
|  subnet-0ef139f32b590ff62 |  vpc-01c3ca8322fbb43dc  |  172.31.16.0/20 |
  us-east-1d |
+--------------------------+-------------------------+-----------------+
-------------+
```

Creamos la instancia seleccionando el id del grupo de seguridad, la subnet y el AMI:

```
C:\Users\Milo\.aws>aws ec2 run-instances --image-id ami-0254b2d5c4c472488
 --count 1 --instance-type t2.micro --key-name MyKeyPair --security-group
-ids sg-0ab162d2371c289b1 --subnet-id subnet-00cb88909a70ef17a
```

Veremos que se ha creado correctamente:



Nos conectamos mediante ssh:

```
C:\Users\Milo\.aws>ssh -i "MyNewKeyPair.pem" ec2-user@ec2-98-81-235-218.c
ompute-1.amazonaws.com
The authenticity of host 'ec2-98-81-235-218.compute-1.amazonaws.com (98.8
1.235.218)' can't be established.
ED25519 key fingerprint is SHA256:9ktAvbrFKv5J26J71GN2jLV7+T9kWR1zH7nJBrD
RqMI.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-98-81-235-218.compute-1.amazonaws.com' (E
D25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15
.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-80-9 ~]$
```

Listamos las instancias:

```
C:\Users\Milo\.aws>aws ec2 describe-instances --filters "Name=instance-ty
pe,Values=t2.micro" --query "Reservations[].Instances[].InstanceId"
[
    "i-0c8a2003751d0240f",
    "i-0564a57ae0dfce692",
    "i-07800b36110fa466d"
]
```

Step 6: Clean Up

Eliminamos las llaves:

```
C:\Users\Milo\.aws>aws ec2 delete-key-pair --key-name MyNewKeyPair
{
    "Return": true,
    "KeyPairId": "key-08095448547f2df13"
}
```

Eliminamos la instancia:

```
C:\Users\Milo\.aws>aws ec2 terminate-instances --instance-ids i-0c8a20037
51d0240f
{
    "TerminatingInstances": [
        {
            "InstanceId": "i-0c8a2003751d0240f",
            "CurrentState": {
                "Code": 32,
                "Name": "shutting-down"
            },
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
```

Eliminamos el grupo de seguridad:

```
C:\Users\Milo\.aws>aws ec2 delete-security-group --group-id sg-0ab162d237
1c289b1
{
    "Return": true,
    "GroupId": "sg-0ab162d2371c289b1"
}
```