

Report - How to use AWS-CLI to create an EC2 instance?

Author: Sergio Bejarano

First, locate the hidden aws folder and edit the respective files to define the correct credentials and configuration.

```
PowerShell
Upgrade now, or check out the release page at:
https://aka.ms/PowerShell-Release?tag=v7.5.2

PS C:\Users\sergio.bejarano-r.LABINFO\Documents\sergio> aws configure
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
PS C:\Users\sergio.bejarano-r.LABINFO\Documents\sergio> cd .aws
Set-Location: Cannot find path 'C:\Users\sergio.bejarano-r.LABINFO\Documents\sergio\.aws' because it does not exist.
PS C:\Users\sergio.bejarano-r.LABINFO\Documents\sergio>
PS C:\Users\sergio.bejarano-r.LABINFO\Documents\sergio> cd..
PS C:\Users\sergio.bejarano-r.LABINFO\Documents\sergio> cd..
PS C:\Users\sergio.bejarano-r.LABINFO> dir

Directory: C:\Users\sergio.bejarano-r.LABINFO

Mode                LastWriteTime         Length Name
----                -
d-----          9/17/2025  9:03 AM                .aws
d-----          9/3/2025 10:25 AM             .docker
d-----         4/11/2024  9:55 AM             .dotnet
d-----          8/20/2025 10:36 AM                .m2
d-----          9/3/2025 10:41 AM               .ssh
d-----          8/20/2025 12:47 PM             .vscode
d-r-----          8/20/2025  8:28 AM             Contacts
d-r-----          8/20/2025  8:28 AM             Desktop
d-r-----          9/17/2025  9:00 AM             Documents
d-r-----          9/3/2025 10:39 AM             Downloads
d-r-----          8/20/2025  8:28 AM             Favorites
d-r-----          8/20/2025  8:28 AM              Links
d-r-----          8/20/2025  8:28 AM              Music
d-r-----          8/20/2025  8:29 AM             OneDrive
d-r-----          8/20/2025  1:59 PM             Pictures
d-r-----          8/20/2025  8:28 AM             Saved Games
d-r-----          8/20/2025  8:29 AM             Searches
d-r-----          8/20/2025  8:28 AM             Videos
-a-----          9/3/2025 11:03 AM              191 .bash_history
-a-----          8/20/2025  1:08 PM               80 .gitconfig
-a-----          8/20/2025  1:07 PM               20 .lessht
-a-----          8/20/2025  1:17 PM             1080 .viminfo

PS C:\Users\sergio.bejarano-r.LABINFO> cd .\.aws\
PS C:\Users\sergio.bejarano-r.LABINFO\.aws> ls

Directory: C:\Users\sergio.bejarano-r.LABINFO\.aws

Mode                LastWriteTime         Length Name
----                -
-a-----          9/17/2025  9:03 AM              46 config
-a-----          9/17/2025  9:03 AM             119 credentials
```

```
credentials
[default]
aws_access_key_id=ASIA3FLDYLUAL6MYCZP
aws_secret_access_key=0jsVFUyUK4VYzhFBKxIJ1ITwgbX1mtehTklS83j
aws_session_token=IQoJb3JpZ21uX2VjEC4aCXVzLXd1c3Q0tMlJHMEUCIE90qp3vDhfzH72zEcD4zc/78
+MEy9S2NzRtcYTRWp1fAiEA1Rbz0EN++/eN1N8Z8/SfbCCwJbq6F1rqaZQLF6BrYWEqyvIIP/////////ARAAGgw3NjczOTc3NDgzMjgiDNFUqWf3BK+WfirBRiqfAtg9k3Y1F19IwscEE5FUo9qINq1jW9o/FUI1a2y84NqVRPkh+
rJmR8
+KQ0d7XathoHfxm0yqg9sJ0VeehYgCIGoU/Ex3zyL3ufwD2TecqUQVrRmRqmE2uzxcK4KPh9obxBNymGjpEgiRJOMM2A2ueVAKmMptQIrZPGH2B4AOgCZi0k6G+LyZOWaSmbxMNDX750/uvRJz6EoI4YPw2a1B2Ve1jioEX700YBnDU
2cxbFvDX12FmsZft52gA0c2Y1Sgv52yx831Sg6vqMuH2AcV6q/Tg1I2WcIDxxNBAzToQtw3remcFqqkTqNpmsfQQ1D046rHy3uJbJYV+iHuz1Gsz/uBDxQQmCD8xDepc5
+/KHGLn/fmHhNk8Pzy5BhxTupPMV8qsYG0p08SegKyhFLOT0/Xv8UeBbPb14psFIAptxV70zvhV8tHWAe6wT5TGsq/Nyug+HGJ0/28qkEmfurGsrFeh81B3z1xXmd7RdrBj20RAT/qh/LkCrndMqjIg04NXyffCZH/wEx+/sPxSSMbe
SNW1QG2guC7/4H2mRv/CVUBqtXRmOnQ76wNCKvT3wH6j81oc9QbKbZJHF11I22vUpGzhDtW==

Ln 4, Col 819   930 characters   Plain text   100%   Windows (CRLF)   UTF-8   Session to end at:
```

```
credentials  config
File Edit View H1
[default]
region = us-east-1
output = json
```

Step 1: Create a Key Pair for EC2

```

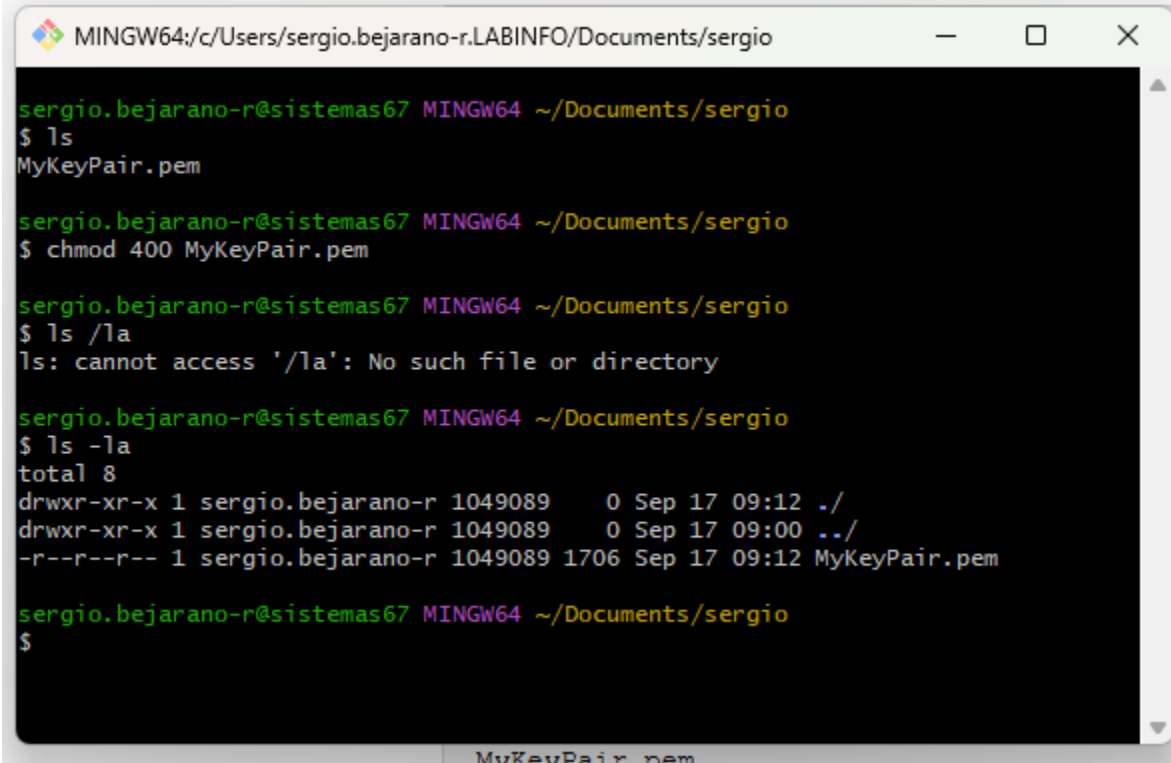
PS C:\Users\sergio.bejarano-r.LABINFO\Documents\sergio> aws ec2 create-key-pair --key-name MyKeyPair --query 'KeyMaterial' --output text > MyKeyPair.pem
PS C:\Users\sergio.bejarano-r.LABINFO\Documents\sergio> ls

Directory: C:\Users\sergio.bejarano-r.LABINFO\Documents\sergio

Mode                LastWriteTime         Length Name
----                -
-a---             9/17/2025   9:12 AM           1706 MyKeyPair.pem
PS C:\Users\sergio.bejarano-r.LABINFO\Documents\sergio> |

```

Make the private key readable only by myself:



```

MINGW64:/c:/Users/sergio.bejarano-r.LABINFO/Documents/sergio

sergio.bejarano-r@sisistemas67 MINGW64 ~/Documents/sergio
$ ls
MyKeyPair.pem

sergio.bejarano-r@sisistemas67 MINGW64 ~/Documents/sergio
$ chmod 400 MyKeyPair.pem

sergio.bejarano-r@sisistemas67 MINGW64 ~/Documents/sergio
$ ls /la
ls: cannot access '/la': No such file or directory

sergio.bejarano-r@sisistemas67 MINGW64 ~/Documents/sergio
$ ls -la
total 8
drwxr-xr-x 1 sergio.bejarano-r 1049089  0 Sep 17 09:12 ./
drwxr-xr-x 1 sergio.bejarano-r 1049089  0 Sep 17 09:00 ../
-r--r--r-- 1 sergio.bejarano-r 1049089 1706 Sep 17 09:12 MyKeyPair.pem

sergio.bejarano-r@sisistemas67 MINGW64 ~/Documents/sergio
$

```

Check the fingerprint:

```

sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ ls -la
total 8
drwxr-xr-x 1 sergio.bejarano-r 1049089  0 Sep 17 09:12 ./
drwxr-xr-x 1 sergio.bejarano-r 1049089  0 Sep 17 09:00 ../
-r--r--r-- 1 sergio.bejarano-r 1049089 1706 Sep 17 09:12 MyKeyPair.pem

sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ aws ec2 describe-key-pairs --key-name MyKeyPair
{
  "KeyPairs": [
    {
      "KeyPairId": "key-01faad2c8e72eba38",
      "KeyFingerprint": "f0:76:30:86:a2:64:dc:75:17:6f:a5:36:18:a2:64:fb:32:df:bf:fe",
      "KeyName": "MyKeyPair",
      "KeyType": "rsa",
      "Tags": [],
      "CreateTime": "2025-09-17T14:12:44.239000+00:00"
    }
  ]
}

sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ |

```

Step 2: Create a Security Group

First, check for VPCs configured in the account.

[Create VPC](#)
[Launch EC2 Instances](#)

Note: Your Instances will launch in the United States region.

Resources by Region

You are using the following Amazon VPC resources

[VPCs](#)
N. Virginia [1](#)

► See all regions

```

sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ aws ec2 create-security-group --group-name my-sg-cli --description "My security group" --vpc-id vpc-050895a691e406167
{
  "GroupId": "sg-0af0eb3614538d6f6"
}

sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$

```

List security groups:

```

sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ aws ec2 describe-security-groups --group-ids sg-0af0eb3614538d6f6
{
  "SecurityGroups": [
    {
      "Description": "My security group",
      "GroupName": "my-sg-cli",
      "IpPermissions": [],
      "OwnerId": "767397748328",
      "GroupId": "sg-0af0eb3614538d6f6",
      "IpPermissionsEgress": [
        {
          "IpProtocol": "-1",
          "IpRanges": [
            {
              "CidrIp": "0.0.0.0/0"
            }
          ],
          "Ipv6Ranges": [],
          "PrefixListIds": [],
          "UserIdGroupPairs": []
        }
      ],
      "VpcId": "vpc-050895a691e406167"
    }
  ]
}

sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio

```

Add Ingress Rules

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

```

sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ curl https://checkip.amazonaws.com
45.239.88.82

```

Allow RDP (port 3389):

```
sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ aws ec2 authorize-security-group-ingress --group-id sg-0af0eb3614538d6f6 --protocol tcp --port 3389 --cidr 0.0.0.0/0
{
  "Return": true,
  "SecurityGroupRules": [
    {
      "SecurityGroupRuleId": "sgr-087e10ee023267cbe",
      "GroupId": "sg-0af0eb3614538d6f6",
      "GroupOwnerId": "767397748328",
      "IsEgress": false,
      "IpProtocol": "tcp",
      "FromPort": 3389,
      "ToPort": 3389,
      "CidrIpv4": "0.0.0.0/0"
    }
  ]
}
```

Allow SSH (port 22):

```
sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ aws ec2 authorize-security-group-ingress --group-id sg-0af0eb3614538d6f6 --protocol tcp --port 22 --cidr 0.0.0.0/0
{
  "Return": true,
  "SecurityGroupRules": [
    {
      "SecurityGroupRuleId": "sgr-0381041e5629f01e3",
      "GroupId": "sg-0af0eb3614538d6f6",
      "GroupOwnerId": "767397748328",
      "IsEgress": false,
      "IpProtocol": "tcp",
      "FromPort": 22,
      "ToPort": 22,
      "CidrIpv4": "0.0.0.0/0"
    }
  ]
}
```

```
sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
```

Verifying:

The screenshot shows the AWS Management Console for an EC2 instance. The left sidebar contains navigation links for EC2, including Dashboard, EC2 Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, Key Pairs, and Network Interfaces.

The main content area shows the instance details for `i-010242e0ab14522f2`. The **Security** tab is selected, displaying the security group rules for the instance. The rules table shows two inbound rules:

Name	Security group rule ID	Port range	Protocol	Source	Security groups	Description
-	sgr-0381041e5629f01e3	22	TCP	0.0.0.0/0	my-sg-cl	-
-	sgr-087e10ee023267cbe	3389	TCP	0.0.0.0/0	my-sg-cl	-

The console also displays other instance details, including the Auto-assigned IP address (`13.217.90.197`), IAM Role, IMDSv2 status, Operator, VPC ID, Subnet ID, Instance ARN, and Launch time (`Wed Sep 17 2025 09:50:55 GMT-0500 (Colombia Standard Time)`).

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:

Resource map

CIDRs

Flow logs

Tags

Integrations

Resource map

Info

VPC

Your AWS virtual network

vpc-050895a691e406167

Subnets (6)

Subnets within this VPC

us-east-1a

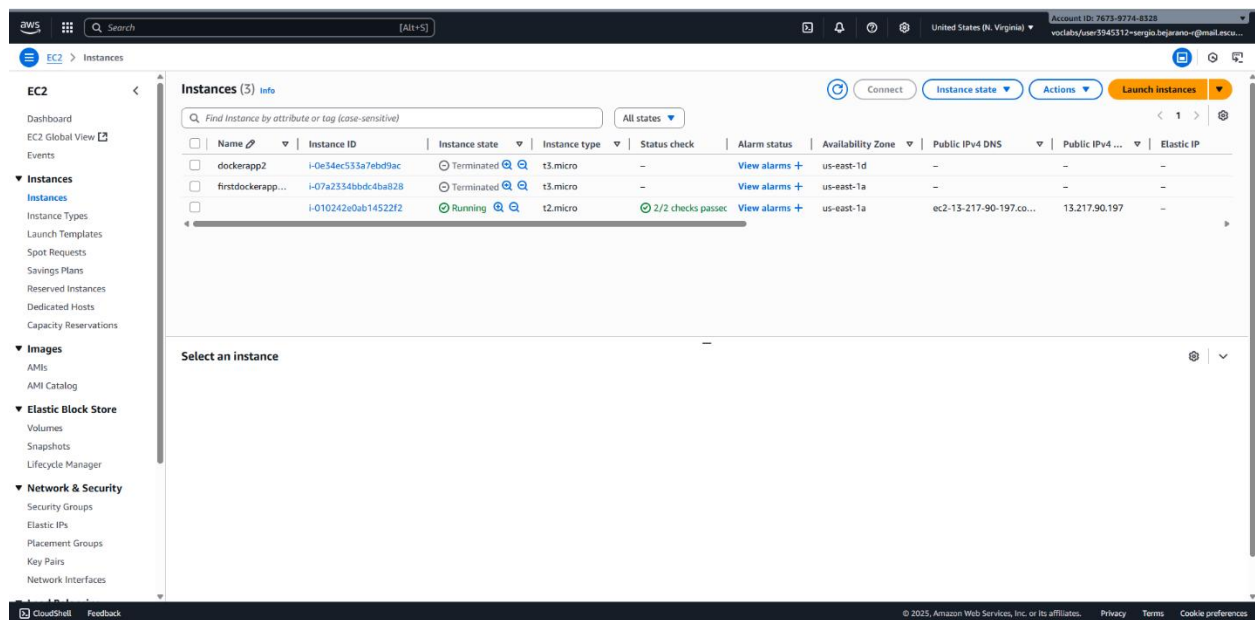
A subnet-003f86e84d49cab54

```

sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ aws ec2 run-instances --image-id ami-032930428bf1abbff --count 1 --instance-type t2.micro --key-name MyKeyPair --security-group-ids
sg-0af0eb3614538d6f6 --subnet-id subnet-003f86e84d49cab54
{
  "Groups": [],
  "Instances": [
    {
      "AmiLaunchIndex": 0,
      "ImageId": "ami-032930428bf1abbff",
      "InstanceId": "i-010242e0ab14522f2",
      "InstanceType": "t2.micro",
      "KeyName": "MyKeyPair",
      "LaunchTime": "2025-09-17T14:50:55+00:00",
      "Monitoring": {
        "State": "disabled"
      },
      "Placement": {
        "AvailabilityZone": "us-east-1a",
        "GroupName": "",
        "Tenancy": "default"
      },
      "PrivateDnsName": "ip-172-31-47-63.ec2.internal",
      "PrivateIpAddress": "172.31.47.63",
      "ProductCodes": [],
      "PublicDnsName": "",
      "State": {
        "Code": 0,
        "Name": "pending"
      },
      "StateTransitionReason": "",
      "SubnetId": "subnet-003f86e84d49cab54",
      "VpcId": "vpc-050895a691e406167",
      "Architecture": "x86_64",
      "BlockDeviceMappings": [],
      "ClientToken": "d95b5af4-deb3-41e1-9748-9b9f99eab116",
      "EbsOptimized": false,
      "EnaSupport": true,
      "Hypervisor": "xen",
      "NetworkInterfaces": [
        {
          "Attachment": {
            "AttachTime": "2025-09-17T14:50:55+00:00",
            "AttachmentId": "eni-attach-06721f36c623cd80c",
            "DeleteOnTermination": true,

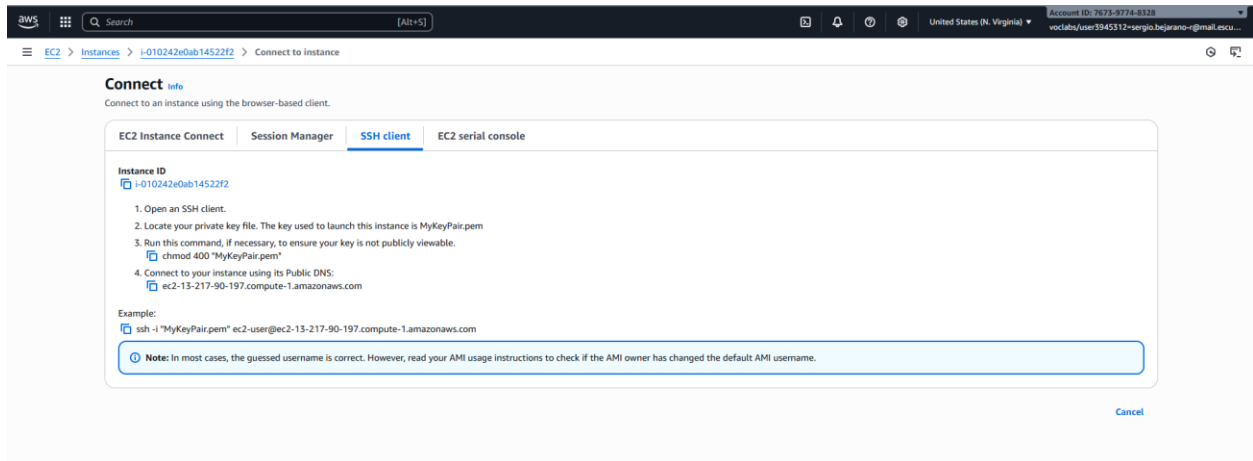
```

Step 4: Connect to the Instance



The screenshot shows the AWS Management Console interface for the 'Instances' page. The left sidebar contains navigation links for EC2, including Dashboard, EC2 Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, Elastic Block Store, and Network & Security. The main content area displays a table of instances with columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, Public IPv4, and Elastic IP. Three instances are listed: 'dockerapp2' (Terminated), 'firstdockerapp...' (Terminated), and 'i-010242e0ab14522f2' (Running). The 'Running' instance has a status check of '2/2 checks passed' and a public IPv4 address of '13.217.90.197'. A 'Connect' button is visible at the top right of the instance list.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4	Elastic IP
dockerapp2	i-0e34ec533a7ebd9ac	Terminated	t3.micro	-	View alarms +	us-east-1d	-	-	-
firstdockerapp...	i-07a2334bbdc4ba828	Terminated	t3.micro	-	View alarms +	us-east-1a	-	-	-
i-010242e0ab14522f2	i-010242e0ab14522f2	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2-13-217-90-197.co...	13.217.90.197	-



```
sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ ssh -i "MyKeyPair.pem" ec2-user@ec2-13-217-90-197.compute-1.amazonaws.com
The authenticity of host 'ec2-13-217-90-197.compute-1.amazonaws.com (13.217.90.197)' can't be established.
ED25519 key fingerprint is SHA256:4EqJ/EmpeirKKmJjc/jPJpffMvGoJUCd2F1+7r49zyY.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-13-217-90-197.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

  _ | _ | _ )
  _ | ( _ /   Amazon Linux AMI
 _ |\_|_|_|

https://aws.amazon.com/amazon-linux-ami/2018.03-release-notes/
27 package(s) needed for security, out of 44 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-47-63 ~]$
```

Step 5: List Your Instances

```
sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ aws ec2 describe-instances --filters "Name=instance-type,Values=t2.micro" --query "Reservations[].Instances[].InstanceId"
[
  "i-010242e0ab14522f2"
]

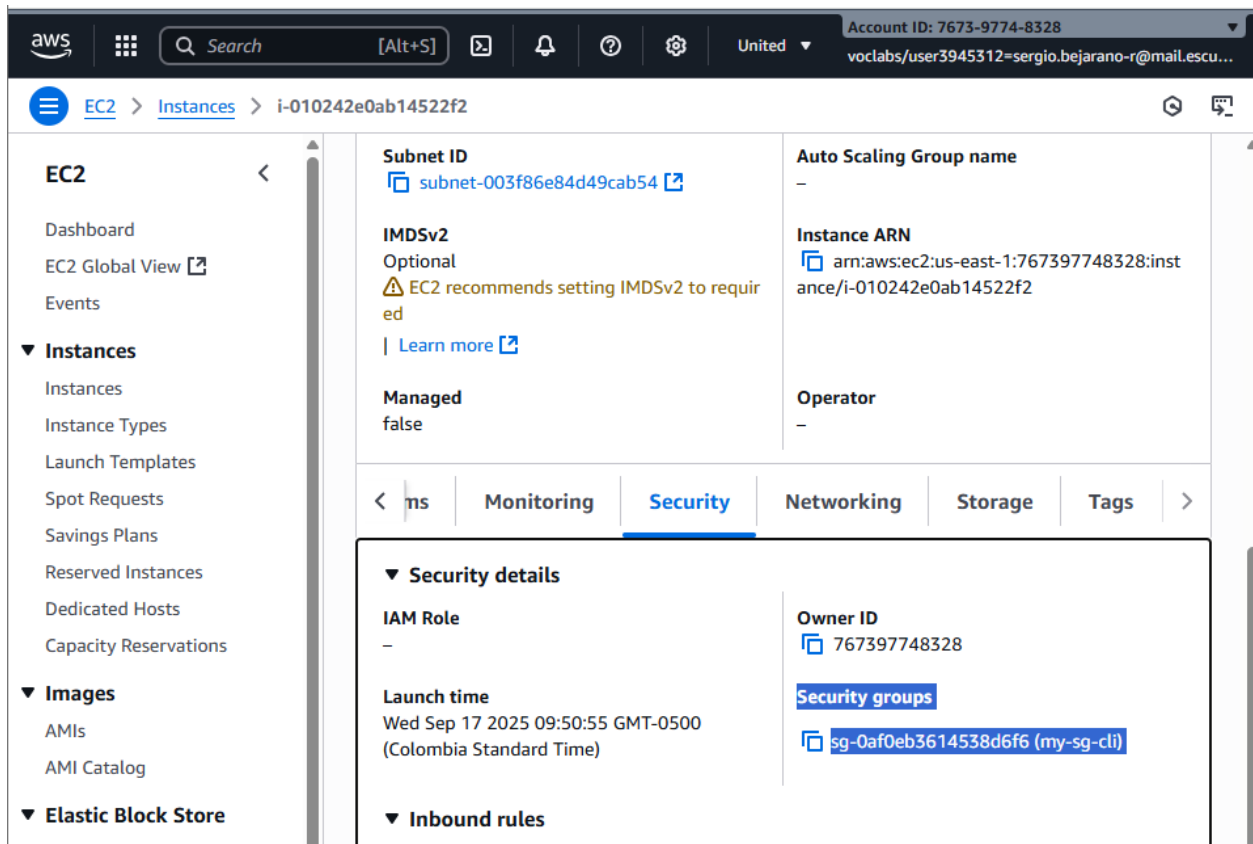
sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$
```

Step 6: Clean Up

Delete the key pair:

```
sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ aws ec2 delete-key-pair --key-name MyKeyPair
{
  "Return": true,
  "KeyPairId": "key-01faad2c8e72eba38"
}
```

Delete the security group:



The screenshot shows the AWS Management Console for an EC2 instance with ID i-010242e0ab14522f2. The left sidebar contains navigation links for EC2, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, and Elastic Block Store. The main content area shows the instance details, including Subnet ID, IMDSv2 status, Auto Scaling Group name, Instance ARN, and Operator. The 'Security' tab is selected, showing the 'Security details' section. The 'IAM Role' is empty, the 'Launch time' is Wed Sep 17 2025 09:50:55 GMT-0500 (Colombia Standard Time), and the 'Owner ID' is 767397748328. The 'Security groups' field is highlighted, showing 'sg-0af0eb3614538d6f6 (my-sg-cli)'. Below this, the 'Inbound rules' section is visible.

An error occurred:

```
sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ aws ec2 delete-security-group --group-id sg-0af0eb3614538d6f6

An error occurred (DependencyViolation) when calling the DeleteSecurityGroup operation: resource sg-0af0eb3614538d6f6 has a dependent object

sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ ^C
```

Delete the instance:

```
sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ aws ec2 terminate-instances --instance-ids i-010242e0ab14522f2
{
  "TerminatingInstances": [
    {
      "CurrentState": {
        "Code": 32,
        "Name": "shutting-down"
      },
      "InstanceId": "i-010242e0ab14522f2",
      "PreviousState": {
        "Code": 16,
        "Name": "running"
      }
    }
  ]
}
```

```
sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$
```

Now the security group is deleted:

```
sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$ aws ec2 delete-security-group --group-id sg-0af0eb3614538d6f6

sergio.bejarano-r@sistemas67 MINGW64 ~/Documents/sergio
$
```

Verifying:

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#instances:instanceState=running

us-east-1

Search

[Alt+S]

United States (N. Virginia)

Account ID: 7673-9774-8328
vocalis/user3945312=sergio.bajarano@gmail.com

EC2

Instances

EC2

Dashboard

EC2 Global View

Events

▼ Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

▼ Images

AMIs

AMI Catalog

▼ Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

▼ Network & Security

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Network Interfaces

Instances info

Find Instance by attribute or tag (case-sensitive)

All states

Instance state = running

Clear filters

Connect

Instance state

Actions

Launch instances

< 1 >

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
No matching instances found										

Select an instance

CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences