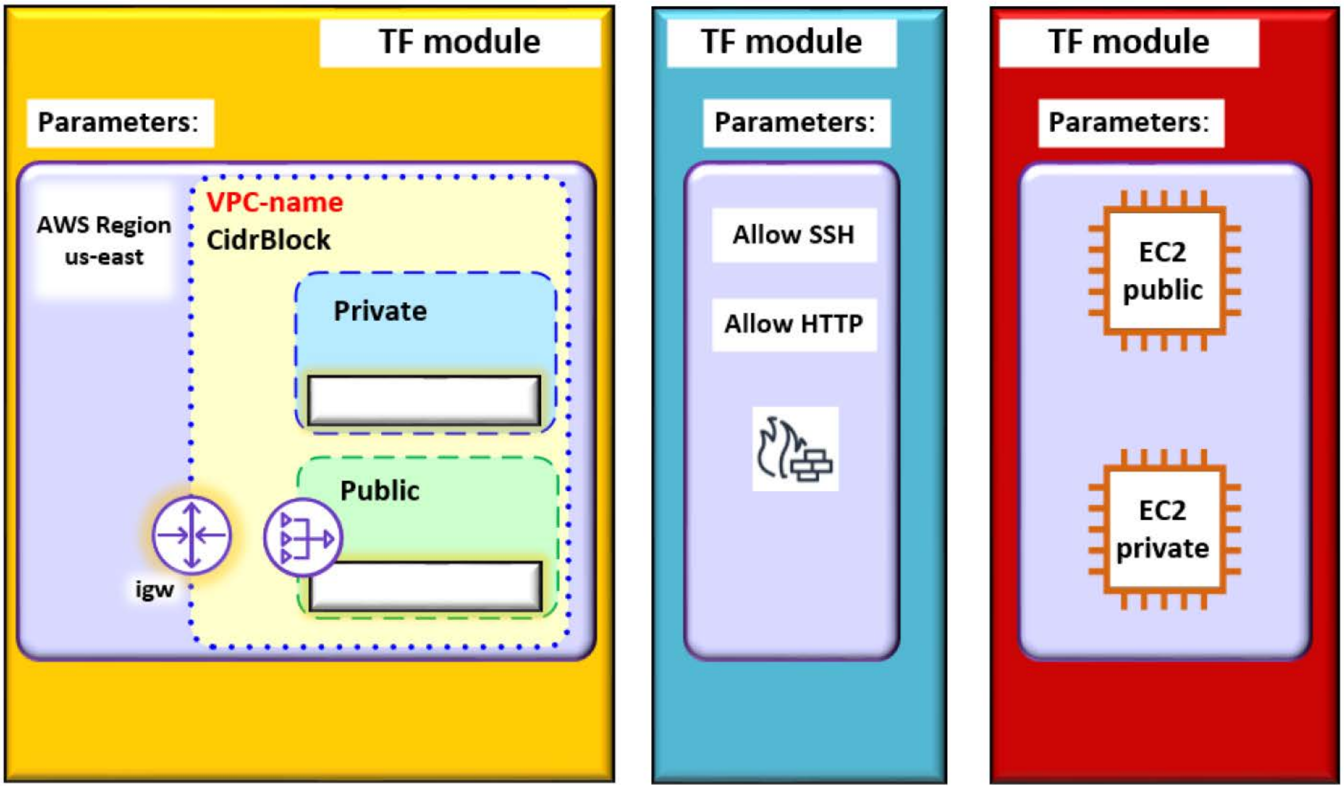
Author: Frank Ekwomadu. C

Tech Stack: AWS Terraform (Cloud9)

VISUAL OF THE INFRASTRUCTURE

Challenge: Using AWS's Terraform, provision the infrastructure below.



Challenge Requirements:

a. vpc/main.tf: Creates a VPC with **private** and **public subnets**. A **NAT Gateway** should be attached to the public subnet to enable the private subnet to access the internet.

b. sg/main.tf: Creates a security group with rules for **SSH** and **HTTP**. These rules should permit traffic from anywhere.

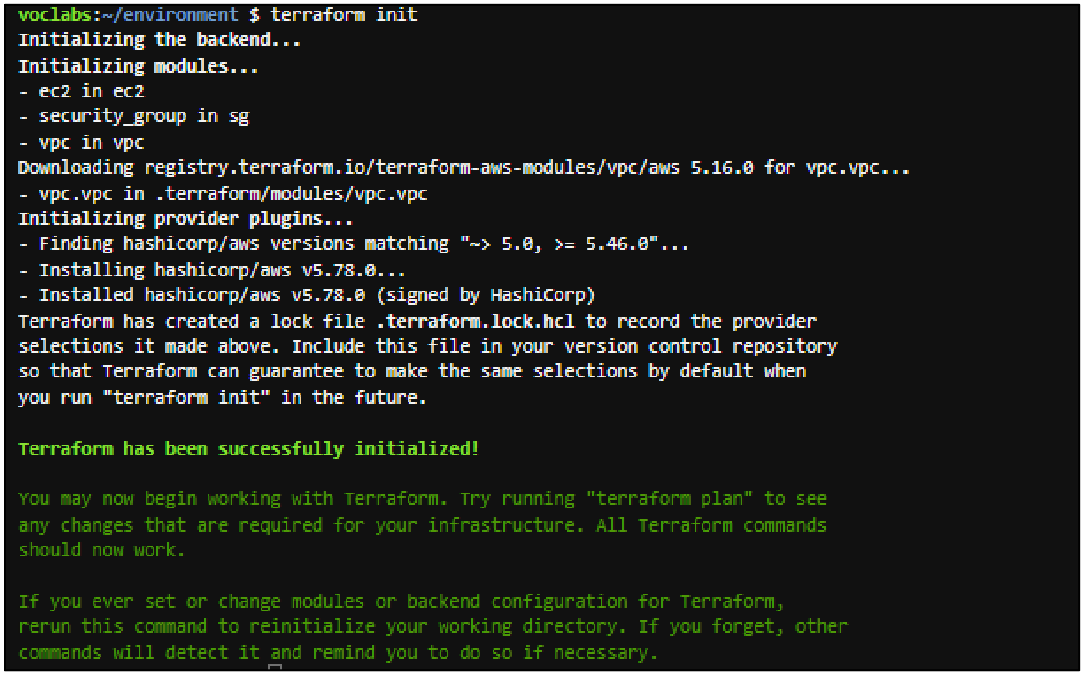
c. ec2/main.tf: Launches an EC2 instance in the private and public subnets each. These EC2 instances **should be webservers** and **make use of the VPC and SG created in (a) and (b)**.

d. main.tf: Wires the modules together by passing appropriate variables and dependencies.

MY SOLUTION VISUALS

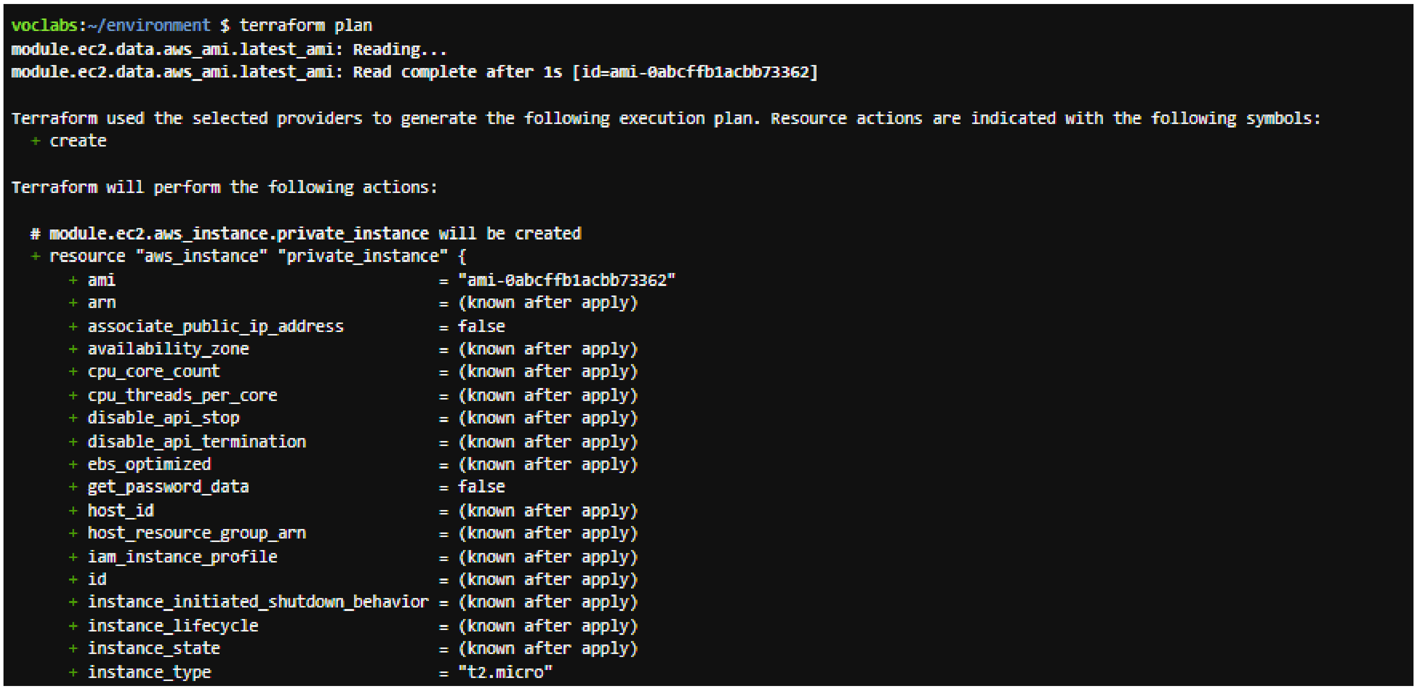
A. Within Cloud9 (Terraform)

Terraform init:



```
voclabs:~/environment $ terraform init
Initializing the backend...
Initializing modules...
- ec2 in ec2
- security_group in sg
- vpc in vpc
Downloading registry.terraform.io/terraform-aws-modules/vpc/aws 5.16.0 for vpc.vpc...
- vpc.vpc in .terraform/modules/vpc.vpc
Initializing provider plugins...
- Finding hashicorp/aws versions matching "~> 5.0, >= 5.46.0"...
- Installing hashicorp/aws v5.78.0...
- Installed hashicorp/aws v5.78.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

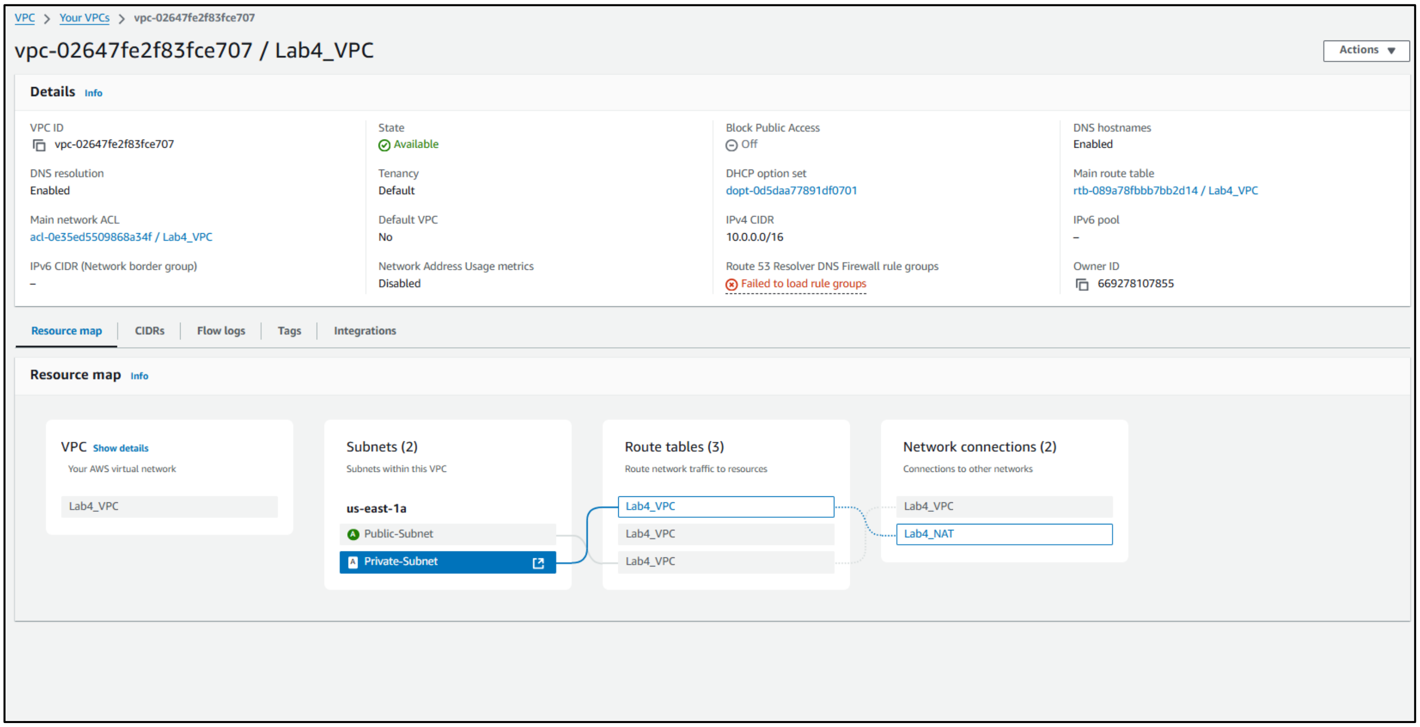
Terraform plan:



```
voclabs:~/environment $ terraform plan
module.ec2.data.aws_ami.latest_ami: Reading...
module.ec2.data.aws_ami.latest_ami: Read complete after 1s [id=ami-0abcffb1acbb73362]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # module.ec2.aws_instance.private_instance will be created
  + resource "aws_instance" "private_instance" {
      + ami                                  = "ami-0abcffb1acbb73362"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = false
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
```

B. VPC, IGW, and Public & Private Subnet with CIDR block

VPC:



VPC > Your VPCs > vpc-02647fe2f83fce707

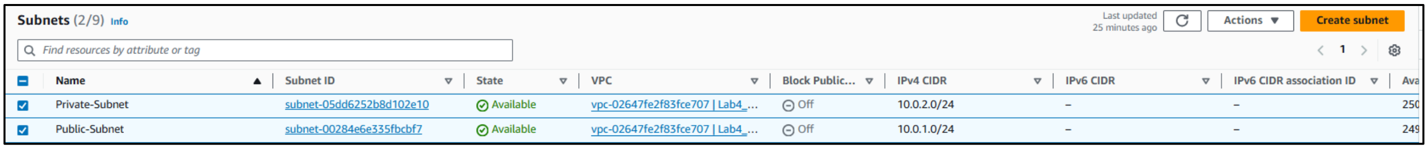
vpc-02647fe2f83fce707 / Lab4_VPC

Details Info

VPC ID vpc-02647fe2f83fce707	State Available	Block Public Access Off	DNS hostnames Enabled
DNS resolution Enabled	Tenancy Default	DHCP option set dopt-0d5daa77891df0701	Main route table rtb-089a78fbbb7bb2d14 / Lab4_VPC
Main network ACL acl-0e35ed5509868a34f / Lab4_VPC	Default VPC No	IPv4 CIDR 10.0.0.0/16	IPv6 pool –
IPv6 CIDR (Network border group) –	Network Address Usage metrics Disabled	Route 53 Resolver DNS Firewall rule groups Failed to load rule groups	Owner ID 669278107855

Resource map

Public & Private Subnet:

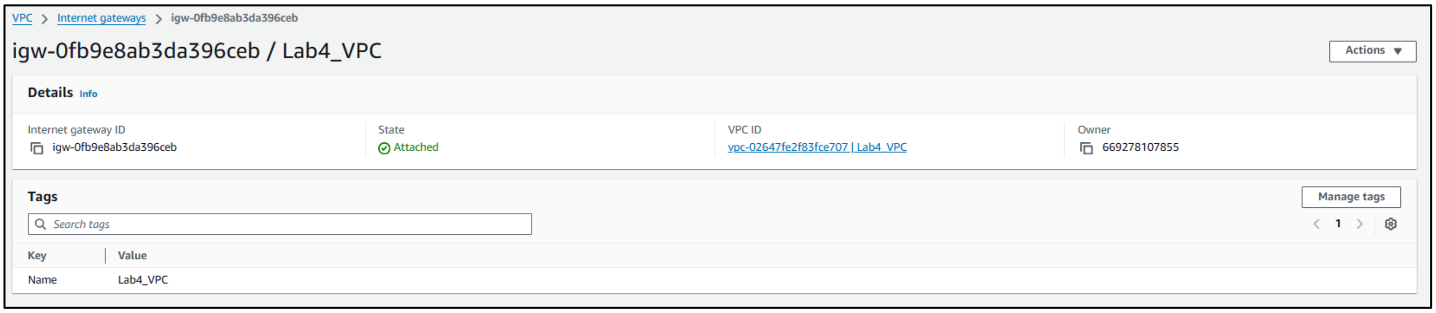


Subnets (2/9) Info

Last updated 25 minutes ago

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR	IPv6 CIDR	IPv6 CIDR association ID	Ava
Private-Subnet	subnet-05dd6252b8d102e10	Available	vpc-02647fe2f83fce707 | Lab4_...	Off	10.0.2.0/24	–	–	250
Public-Subnet	subnet-00284e6e335fbcbf7	Available	vpc-02647fe2f83fce707 | Lab4_...	Off	10.0.1.0/24	–	–	249

IGW:



VPC > Internet gateways > igw-0fb9e8ab3da396ceb

igw-0fb9e8ab3da396ceb / Lab4_VPC

Details Info

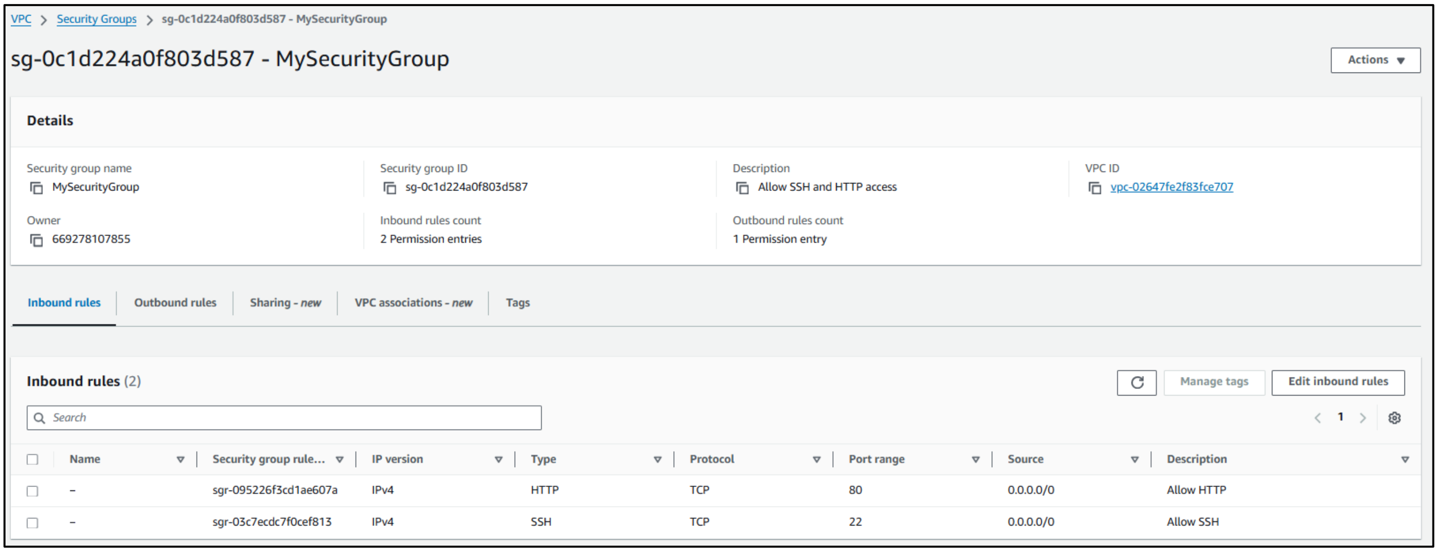
Internet gateway ID igw-0fb9e8ab3da396ceb	State Attached	VPC ID vpc-02647fe2f83fce707 | Lab4_VPC	Owner 669278107855

Tags

Key	Value
Name	Lab4_VPC

C. EC2 with the given SG configuration in the public subnet created

SG:



sg-0c1d224a0f803d587 - MySecurityGroup

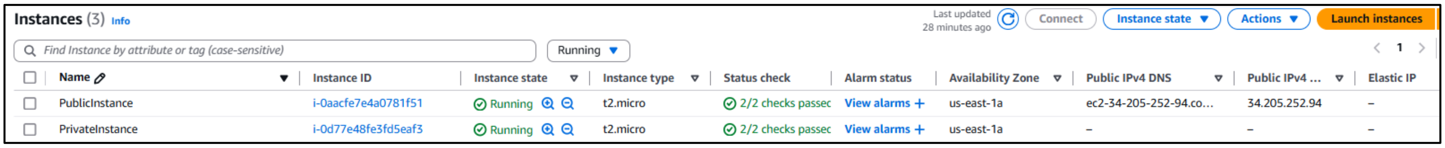
Details

Security group name MySecurityGroup	Security group ID sg-0c1d224a0f803d587	Description Allow SSH and HTTP access	VPC ID vpc-02647fe2f83fce707
Owner 669278107855	Inbound rules count 2 Permission entries	Outbound rules count 1 Permission entry	

Inbound rules (2)

□	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
□	–	sgr-095226f3cd1ae607a	IPv4	HTTP	TCP	80	0.0.0.0/0	Allow HTTP
□	–	sgr-03c7ecdc7f0cef813	IPv4	SSH	TCP	22	0.0.0.0/0	Allow SSH

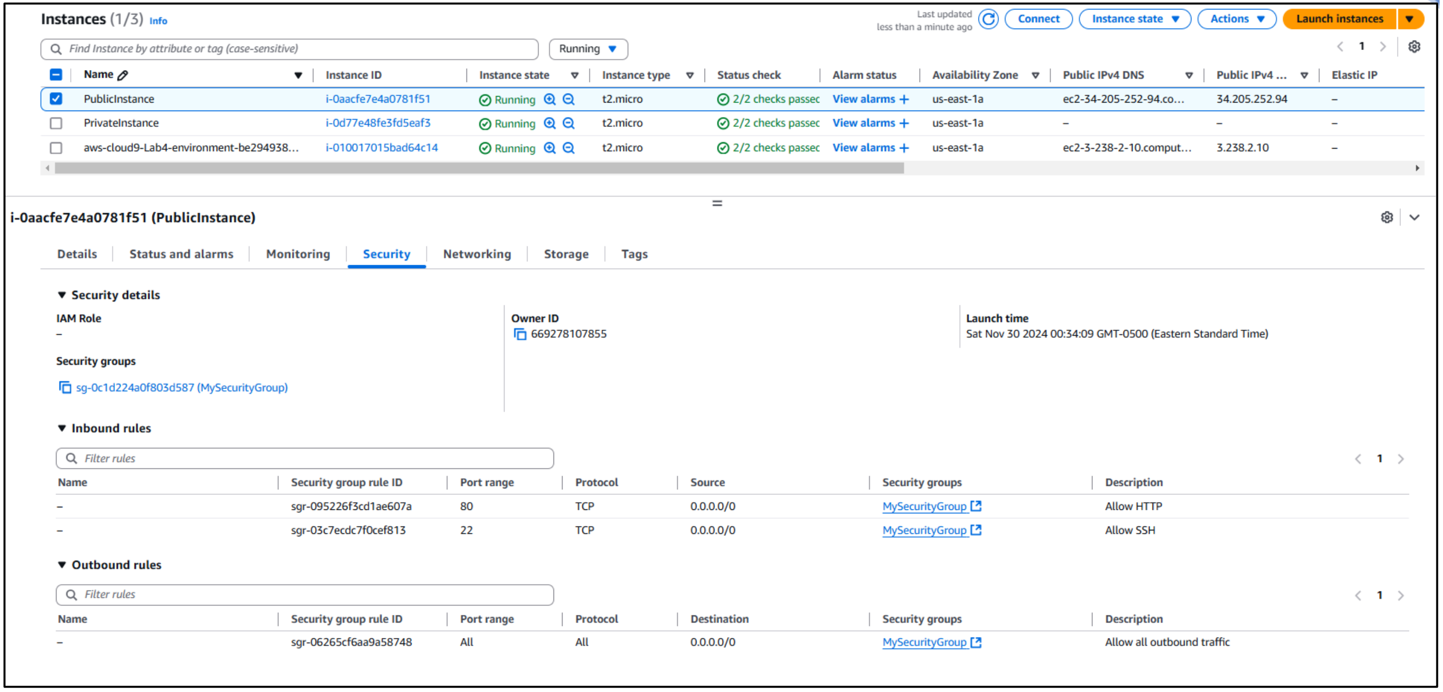
Running Instances:



Instances (3)

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
□	PublicInstance	i-0aacfe7e4a0781f51	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	us-east-1a	ec2-34-205-252-94.co...	34.205.252.94	–
□	PrivateInstance	i-0d77e48fe3fd5eaf3	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	us-east-1a	–	–	–

SG config in the public subnet:



Instances (1/3) Info

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
☑	PublicInstance	i-0aacfe7e4a0781f51	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1a	ec2-34-205-252-94.co...	34.205.252.94	–
□	PrivateInstance	i-0d77e48fe3fd5eaf3	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1a	–	–	–
□	aws-cloud9-Lab4-environment-be294938...	i-010017015bad64c14	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1a	ec2-3-238-2-10.comput...	3.238.2.10	–

i-0aacfe7e4a0781f51 (PublicInstance)

▼ Security details

IAM Role
–

Owner ID
669278107855

Launch time
Sat Nov 30 2024 00:34:09 GMT-0500 (Eastern Standard Time)

Security groups
sg-0c1d224a0f803d587 (MySecurityGroup)

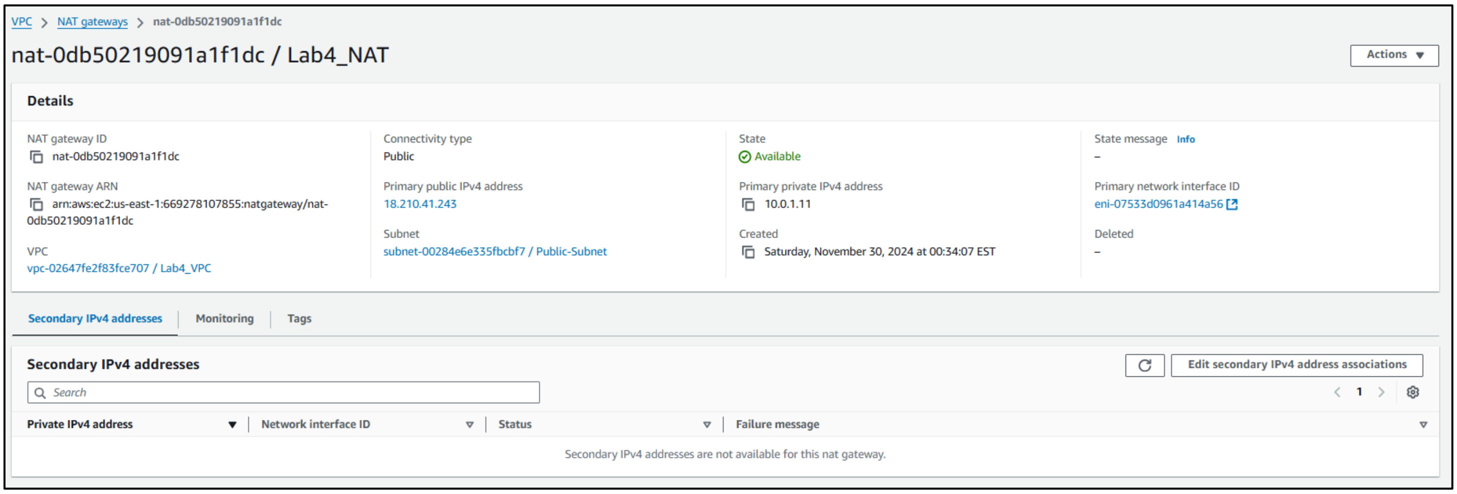
▼ Inbound rules

Name	Security group rule ID	Port range	Protocol	Source	Security groups	Description
–	sgr-095226f3cd1ae607a	80	TCP	0.0.0.0/0	MySecurityGroup	Allow HTTP
–	sgr-03c7ecdc7f0cef813	22	TCP	0.0.0.0/0	MySecurityGroup	Allow SSH

▼ Outbound rules

Name	Security group rule ID	Port range	Protocol	Destination	Security groups	Description
–	sgr-06265cf6aa9a58748	All	All	0.0.0.0/0	MySecurityGroup	Allow all outbound traffic

NAT Gateway



nat-0db50219091a1f1dc / Lab4_NAT

Details

NAT gateway ID
nat-0db50219091a1f1dc

NAT gateway ARN
arn:aws:ec2:us-east-1:669278107855:natgateway/nat-0db50219091a1f1dc

VPC
vpc-02647fe2f83fce707 / Lab4_VPC

Connectivity type
Public

Primary public IPv4 address
18.210.41.243

Subnet
subnet-00284e6e335fbcbf7 / Public-Subnet

State
Available

Primary private IPv4 address
10.0.1.11

Created
Saturday, November 30, 2024 at 00:34:07 EST

State message Info
–

Primary network interface ID
eni-07533d0961a414a56

Deleted
–

Secondary IPv4 addresses

Private IPv4 address	Network interface ID	Status	Failure message

Secondary IPv4 addresses are not available for this nat gateway.