Task:

* Create a VPC, subnet, security group, Key pair and an t2.micro instance using terraform
* The values should be passed as a variable.

Steps:

* First let try hardcoding the values
* **Create a VPC**

|  |
| --- |
| resource "aws\_vpc" "test\_vpc" {  cidr\_block = "10.0.0.0/26"  tags = {  Name = "terraform\_vpc"  }  } |

* **Create a subnet in the VPC**

|  |
| --- |
| resource "aws\_subnet" "test\_subnet" {  vpc\_id = aws\_vpc.test\_vpc.id  cidr\_block = "10.0.0.0/26" # 64 ips  availability\_zone = "ap-south-1a"  tags = {  Name = "terraform\_subnet"  }  } |

* **Create a security group**

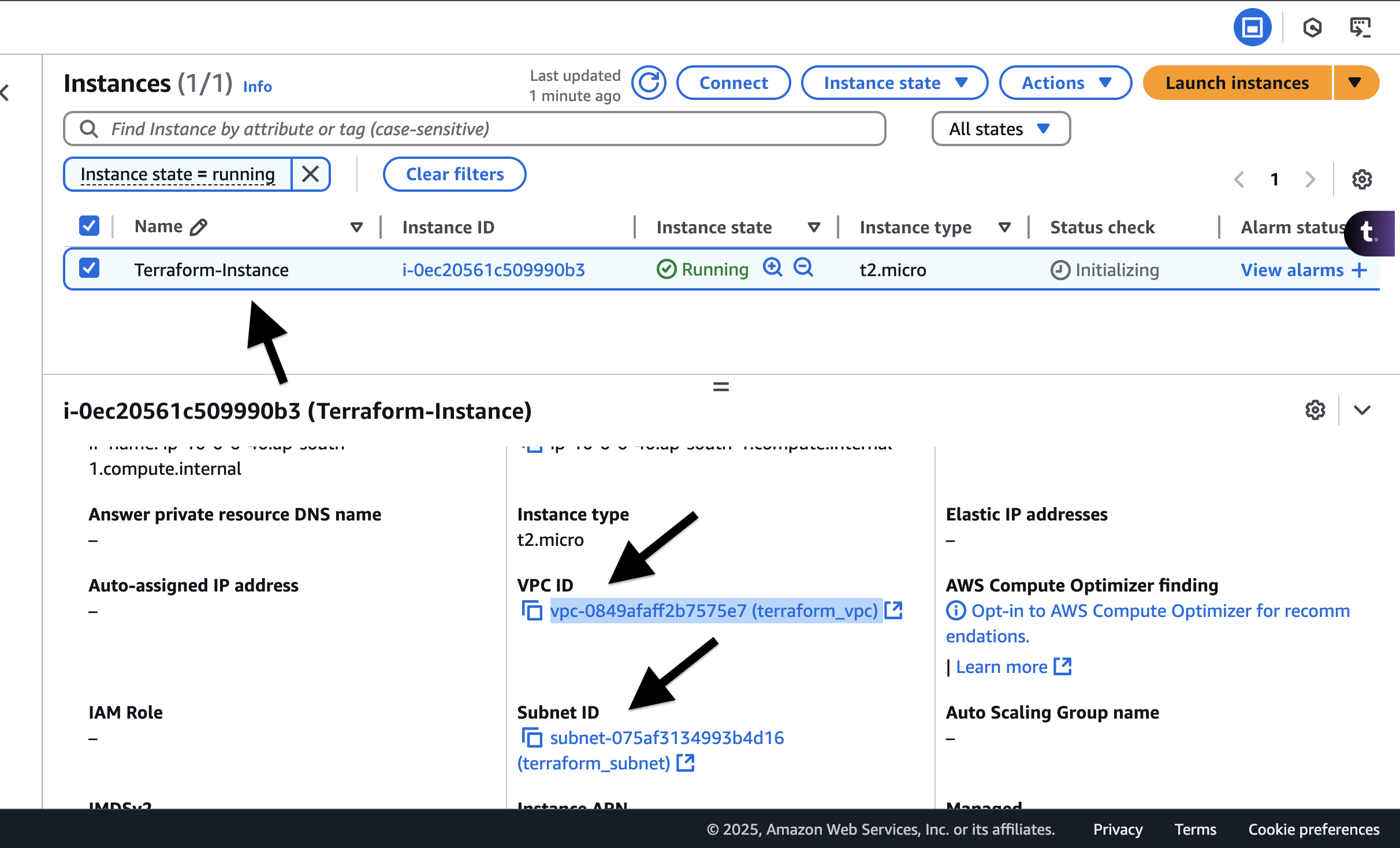
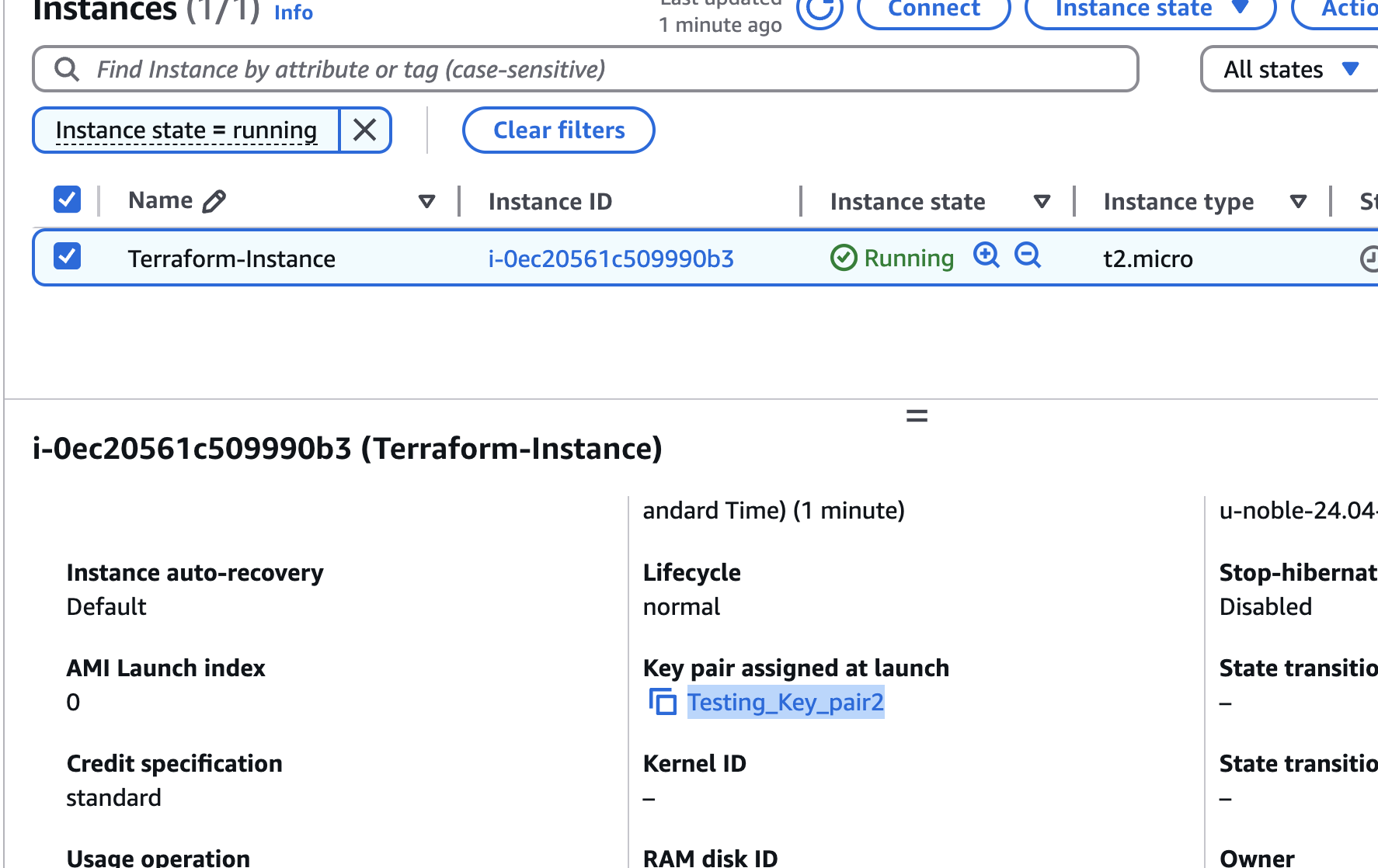
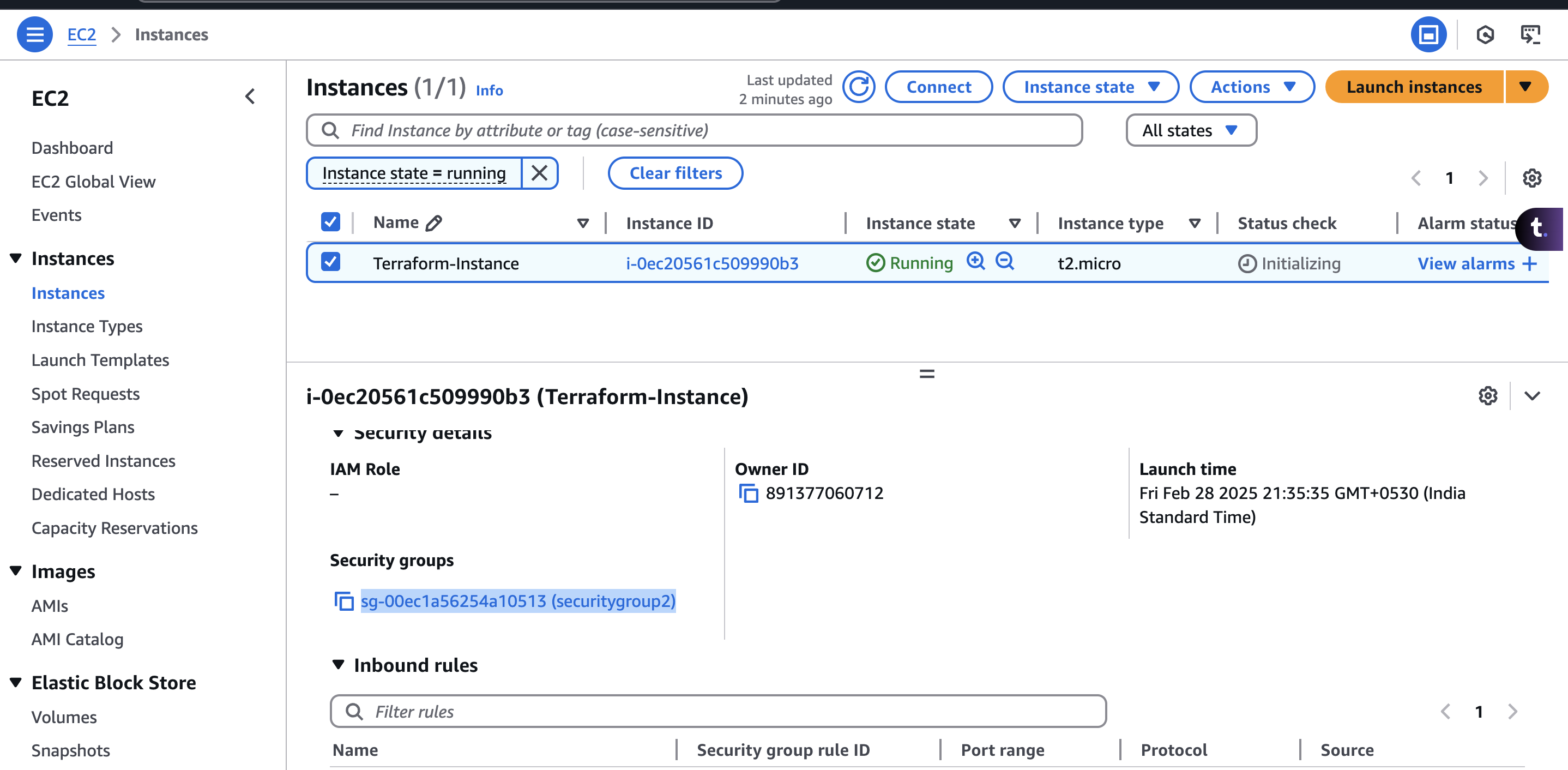
|  |
| --- |
| resource "aws\_security\_group" "test\_sg" {  name = "securitygroup1"  description = "Allow http and ssh ports"  vpc\_id = aws\_vpc.test\_vpc.id  tags = {  Name = "terrafrom\_securitygroup"  }  ingress {  from\_port = "22"  to\_port = "22"  protocol = "tcp"  cidr\_blocks = ["0.0.0.0/0"]  }  ingress { #allowing http port  from\_port = 80  to\_port = 80  protocol = "tcp"  cidr\_blocks = ["0.0.0.0/0"] #from anywhere  }  egress{  from\_port = 0  to\_port = 0  protocol = "-1" # any port  cidr\_blocks = ["0.0.0.0/0"] # anywhere  }  } |

* **Create Key\_pair**  
  Run the below command to generate the public key and store it in a file  
  **ssh-keygen -f terraform\_key\_pair**

|  |
| --- |
| resource "aws\_key\_pair" "test\_key\_pair" {  key\_name = "Testing\_Key\_pair2"  public\_key = "${file("terraform\_key\_pair.pub")}"  tags = {  Name = "terraform\_key\_pair"  }  } |

* **Create an Instance**

|  |
| --- |
| resource "aws\_instance" "test1" {  ami = "ami-00bb6a80f01f03502"  instance\_type = "t2.micro"  key\_name = "Testing\_Key\_pair2"  subnet\_id = aws\_subnet.test\_subnet.id  security\_groups = [aws\_security\_group.test\_sg.id]  availability\_zone = aws\_subnet.test\_subnet.availability\_zone  tags = {  Name = "Terraform-Instance" #this will be name of the ec2 instance  }    } |

* Done  
    
    
    
    
  
* Now, we need to do the same by using the variables.

Without variable:  
main.tf

|  |
| --- |
| provider "aws" {  region = "ap-south-1"  }  resource "aws\_vpc" "test\_vpc" {  cidr\_block = "10.0.0.0/25"  tags = {  Name = "terraform\_vpc"  }  }  resource "aws\_subnet" "test\_subnet" {  vpc\_id = aws\_vpc.test\_vpc.id  cidr\_block = "10.0.0.0/26" # 64 ips  tags = {  Name = "terraform\_subnet"  }  }  resource "aws\_security\_group" "test\_sg" {  name = "securitygroup2"  description = "Allow http and ssh ports"  vpc\_id = aws\_vpc.test\_vpc.id  tags = {  Name = "terrafrom\_securitygroup"  }  ingress {  from\_port = "22"  to\_port = "22"  protocol = "tcp"  cidr\_blocks = ["0.0.0.0/0"]  }  ingress { #allowing http port  from\_port = 80  to\_port = 80  protocol = "tcp"  cidr\_blocks = ["0.0.0.0/0"] #from anywhere  }  egress{  from\_port = 0  to\_port = 0  protocol = "-1" # any port  cidr\_blocks = ["0.0.0.0/0"] # anywhere  }  }  resource "aws\_key\_pair" "test\_key\_pair" {  key\_name = "Testing\_Key\_pair2"  public\_key = "${file("terraform\_key\_pair.pub")}"  tags = {  Name = "terraform\_key\_pair"  }  }  resource "aws\_instance" "test1" {  ami = "ami-00bb6a80f01f03502"  instance\_type = "t2.micro"  key\_name = "Testing\_Key\_pair2"  subnet\_id = aws\_subnet.test\_subnet.id  security\_groups = [aws\_security\_group.test\_sg.id]  tags = {  Name = "Terraform-Instance" #this will be name of the ec2 instance  }    } |

* With Variables:  
  **main.tf**

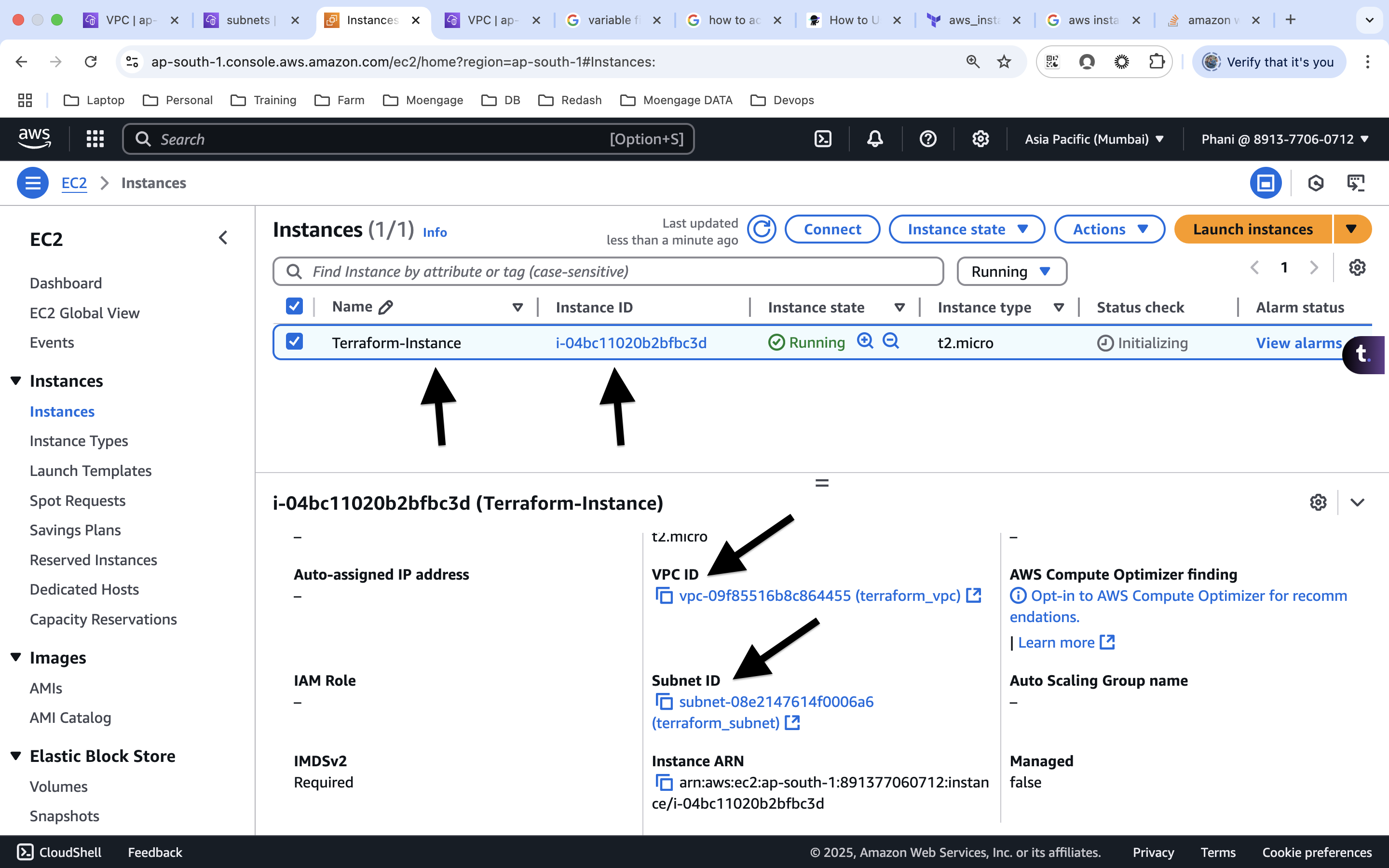
|  |
| --- |
| provider "aws" {  region = var.default\_region  }  resource "aws\_vpc" "test\_vpc" {  cidr\_block = var.vpc\_block  tags = {  Name = var.vpc\_name  }  }  resource "aws\_subnet" "test\_subnet" {  vpc\_id = aws\_vpc.test\_vpc.id  cidr\_block = var.subnet\_block  availability\_zone = var.availability\_zone  tags = {  Name = var.subnet\_name  }  }  resource "aws\_security\_group" "test\_sg" {  name = var.security\_groups\_group\_name  description = var.security\_groups\_desc  vpc\_id = aws\_vpc.test\_vpc.id  tags = {  Name = var.security\_groups\_name  }  ingress {  from\_port = var.ssh\_port  to\_port = var.ssh\_port  protocol = var.protocol  cidr\_blocks = var.traffic\_from\_anywhere  }  ingress { #allowing http port  from\_port = var.http\_port  to\_port = var.http\_port  protocol = var.protocol  cidr\_blocks = var.traffic\_from\_anywhere #from anywhere  }  egress{  from\_port = var.to\_from\_port  to\_port = var.to\_from\_port  protocol = var.any\_protocol # any port  cidr\_blocks = var.traffic\_from\_anywhere # anywhere  }  }  resource "aws\_key\_pair" "test\_key\_pair" {  key\_name = var.key\_name  public\_key = "${file("terraform\_key\_pair.pub")}"  tags = {  Name = var.key\_pair\_name  }  }  resource "aws\_instance" "test1" {  ami = var.ami\_id  instance\_type = var.instance\_type  key\_name = var.key\_name  subnet\_id = aws\_subnet.test\_subnet.id  security\_groups = [aws\_security\_group.test\_sg.id]  availability\_zone = aws\_subnet.test\_subnet.availability\_zone  tags = {  Name = var.instance\_name #this will be name of the ec2 instance  }    } |

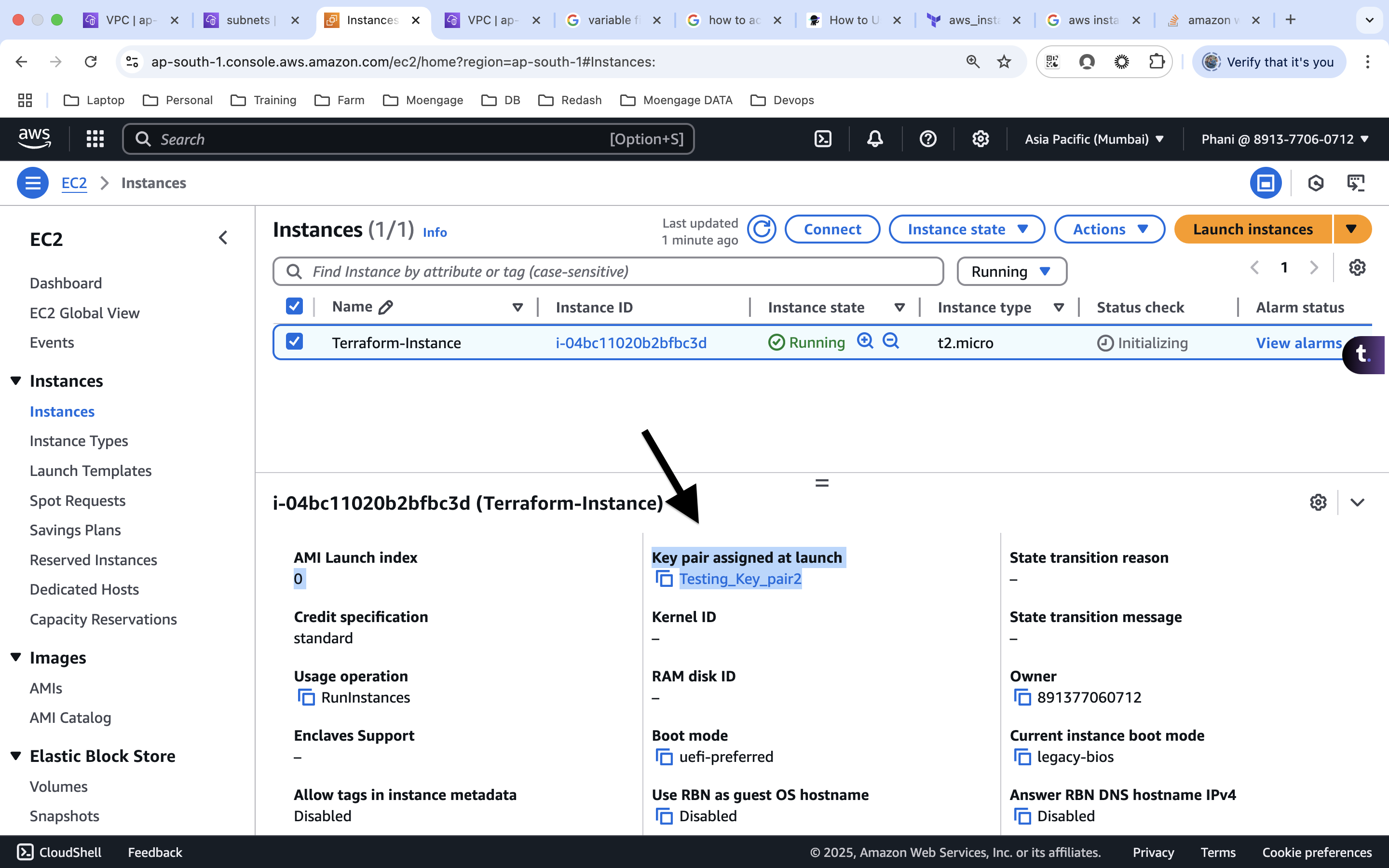
**variables.tf**

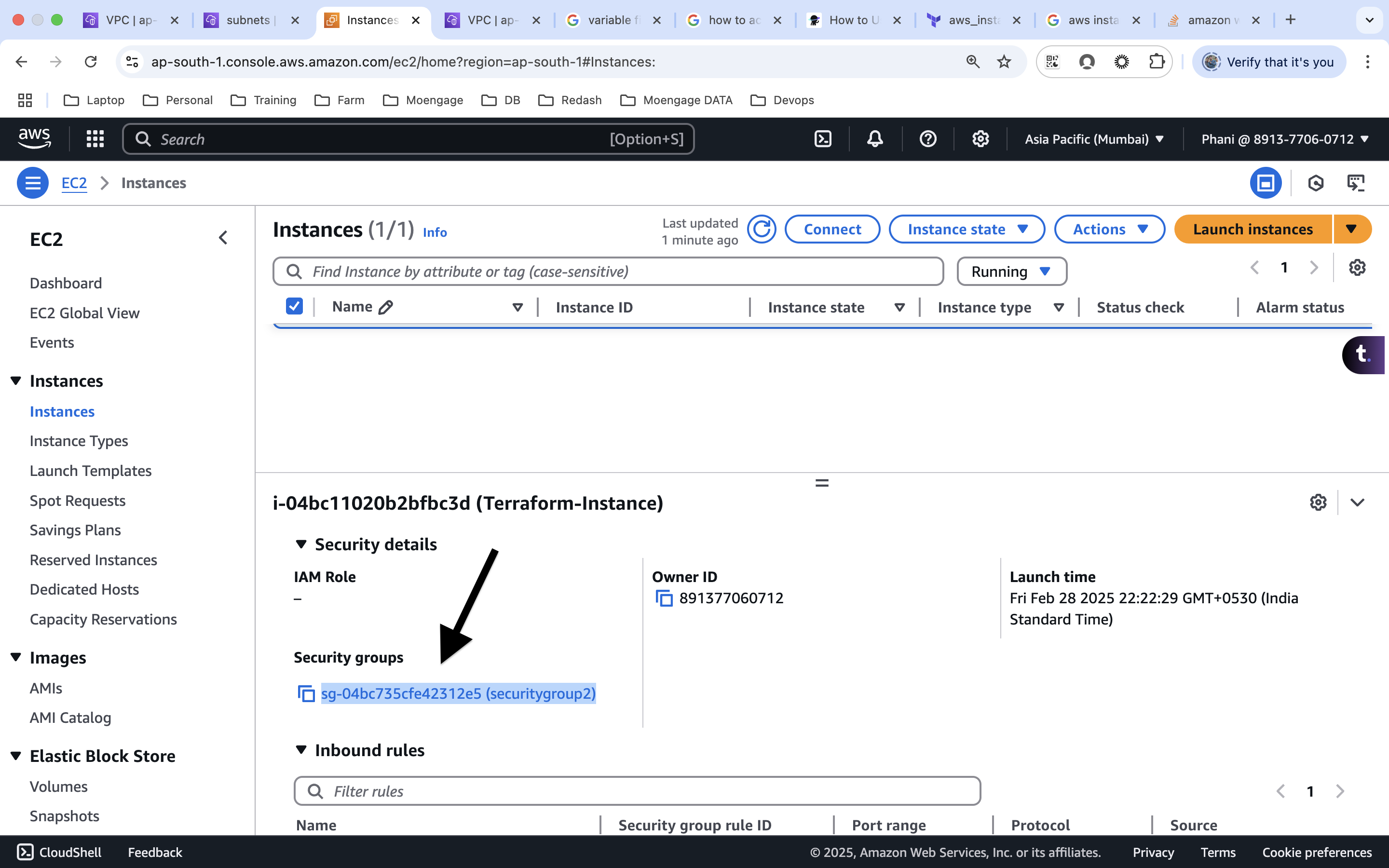
|  |
| --- |
| variable "default\_region" {  default = "ap-south-1"  description = "The default region"  }  variable "availability\_zone" {  default = "ap-south-1a"  }  variable "vpc\_block" {  default = "10.0.0.0/25"  description = "CIDR block"  }  variable "vpc\_name" {  default = "default\_VPC"  description = "Name of the VPC"  }  variable "subnet\_block" {  default = "10.0.0.0/25"  description = "CIDR block of the subnet"  }  variable "subnet\_name" {  default = "default\_VPC"  description = "Name of the VPC"  }  variable "security\_groups\_group\_name" {  default = "launch-wizard-1"  description = "security group name"  }  variable "security\_groups\_desc" {  default = "Allow http and ssh ports"  description = "security group name description"  }  variable "security\_groups\_name" {  default = ""  description = "security group name"  }  variable "ssh\_port" {  default = "22"  description = "SSH port"  }  variable "http\_port" {  default = "80"  description = "HTTP port"  }  variable "protocol" {  default = "tcp"  description = "tcp protocol"  }  variable "to\_from\_port" {  default = "0"  }  variable "any\_protocol" {  default = "-1"  description = "tcp protocol"  }  variable "traffic\_from\_anywhere" {  default = ["0.0.0.0/0"]  description = "Allow traffic from anywhere"  }  variable "key\_name" {  default = "Testing\_Key\_pair2"  description = "Key name"  }  variable "key\_pair\_name" {  default = "terraform\_key\_pair"  description = "Key pair name"  }  variable "ami\_id" {  default = "ami-00bb6a80f01f03502"  description = "AMI id"  }  variable "instance\_type" {  default = "t2.micro"  description = "The EC2 instance type"  }  variable "instance\_name" {  default = "Terraform-Instance"  description = "Name of the instance"  } |

**terraform.tfvars**

|  |
| --- |
| vpc\_name = "terraform\_vpc"  subnet\_name = "terraform\_subnet"  security\_groups\_group\_name = "securitygroup2"  security\_groups\_name = "terrafrom\_securitygroup" |







A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.