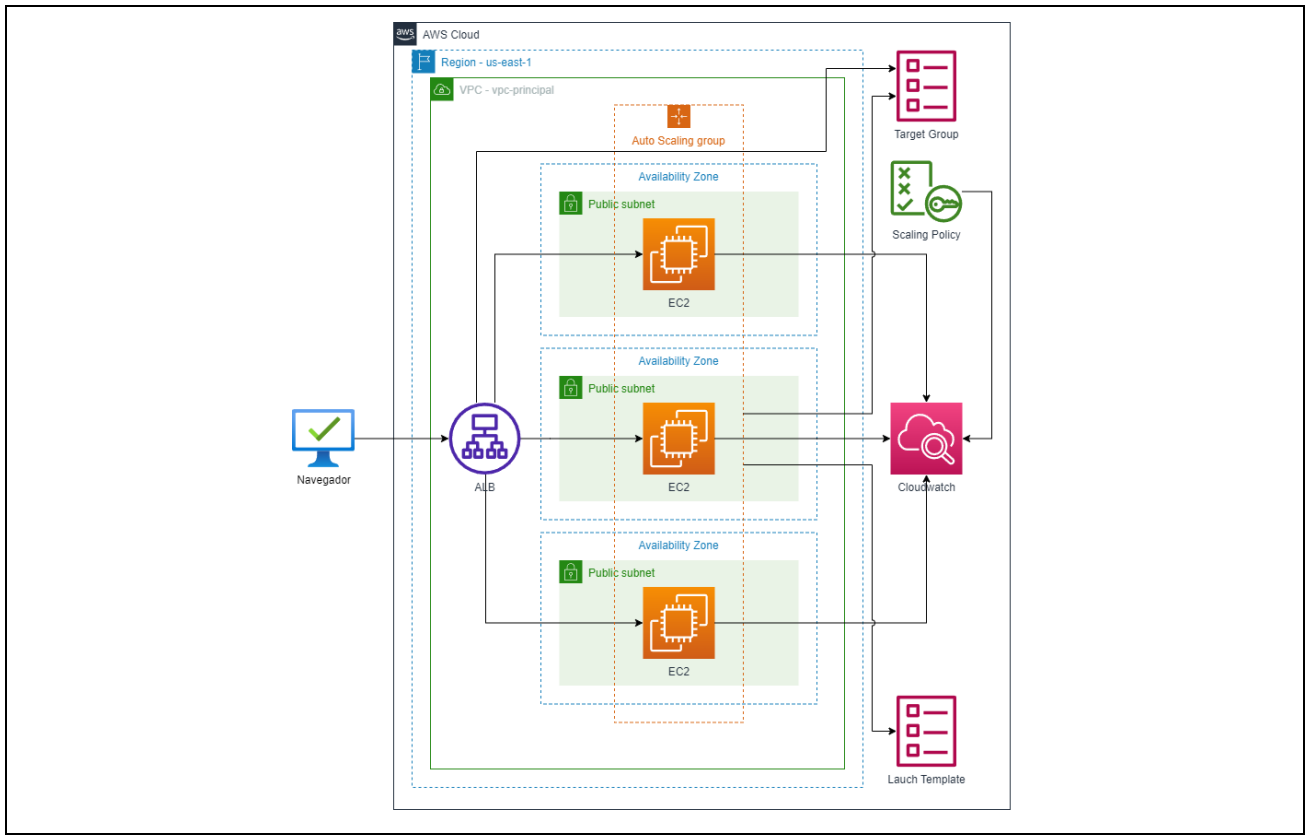
Implantando um Application Load Balancer (ALB) na AWS com EC2

Prof. Thomás da Costa - https://thomasdacosta.com.br

Objetivo:

Implantar um Application Load Balancer na AWS direcionando o tráfego para as instâncias EC2 que possuem o servidor Apache HTTP instalado. Neste tutorial, iremos demonstrar como criar o ALB, Auto Scaling Group, Target Groupe o Launch Template.

Desenho da Solução:



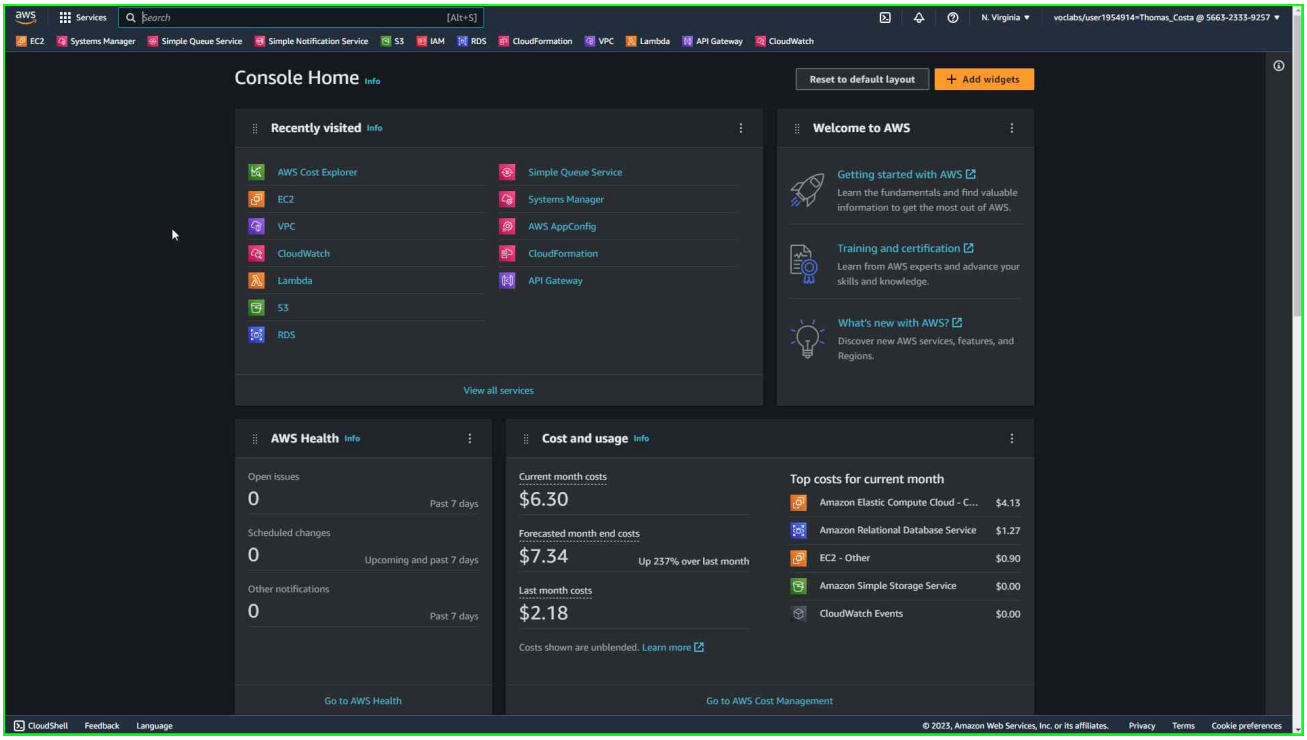
Premissas:

- Possuir um **Security Group** com **acesso de entrada** para a **porta HTTP 80** para todos os endereços chamado **SG-webserver**;
- Possuir uma **IAM Role** chamada **LabRole** com as permissões e **Policy** necessárias para a execução dos recursos;
- Conhecer previamente o **CloudWatch Logs** e seu funcionamento.

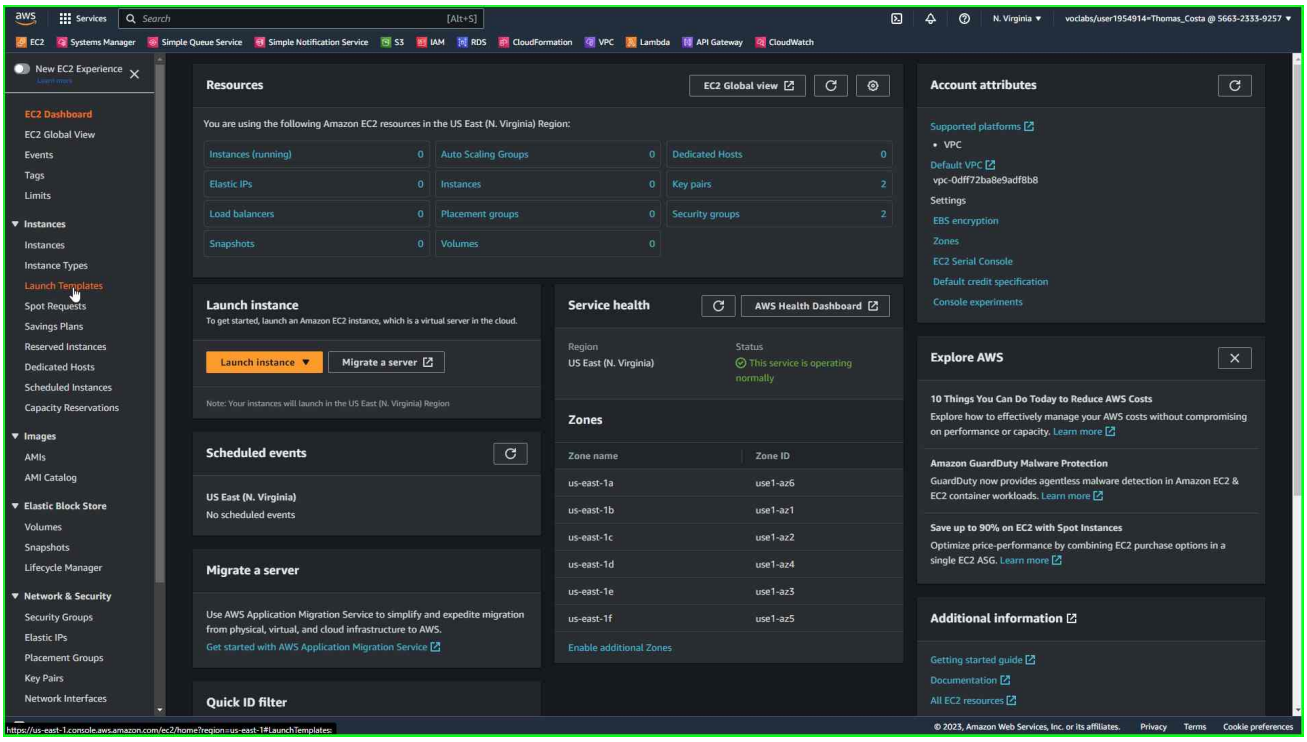
Implantando um Application Load Balancer (ALB) na AWS com EC2

Parte 1 – Criando o Launch Template

Na tela principal, pesquisar a opção **EC2**:

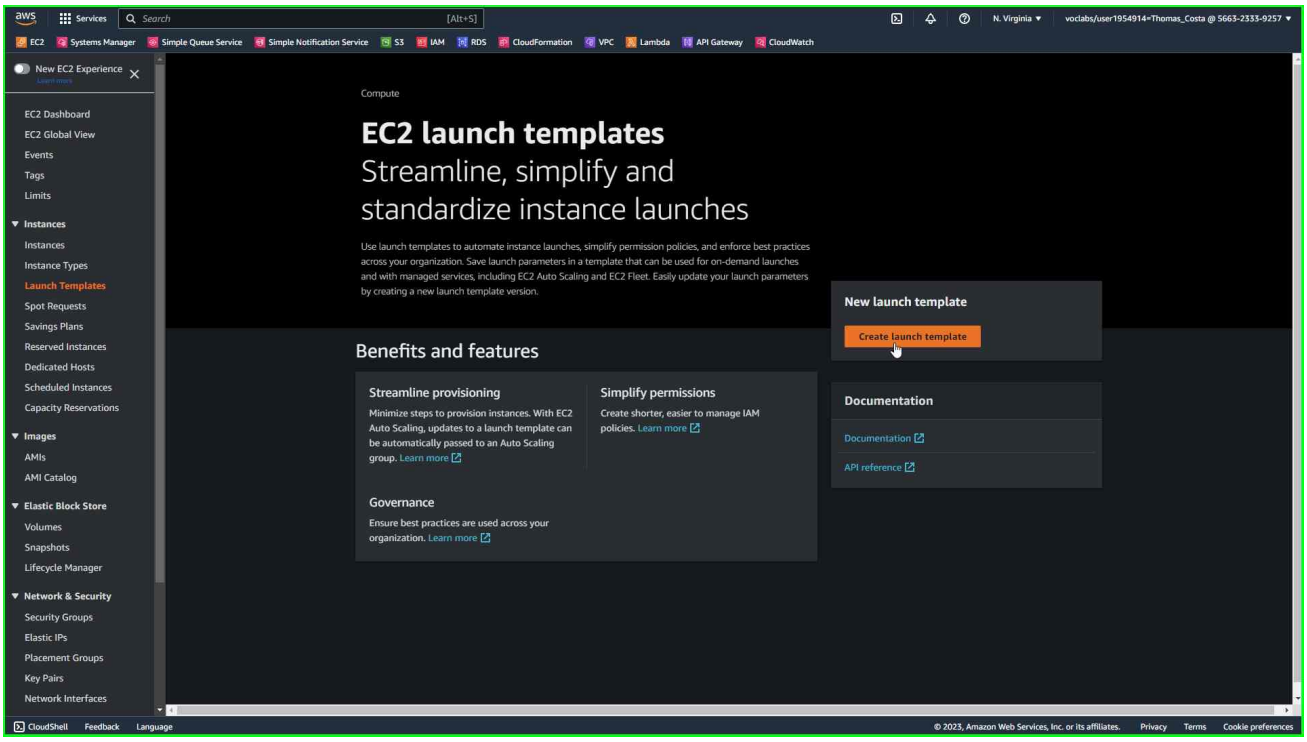


Selecione a opção **Launch Templates:**

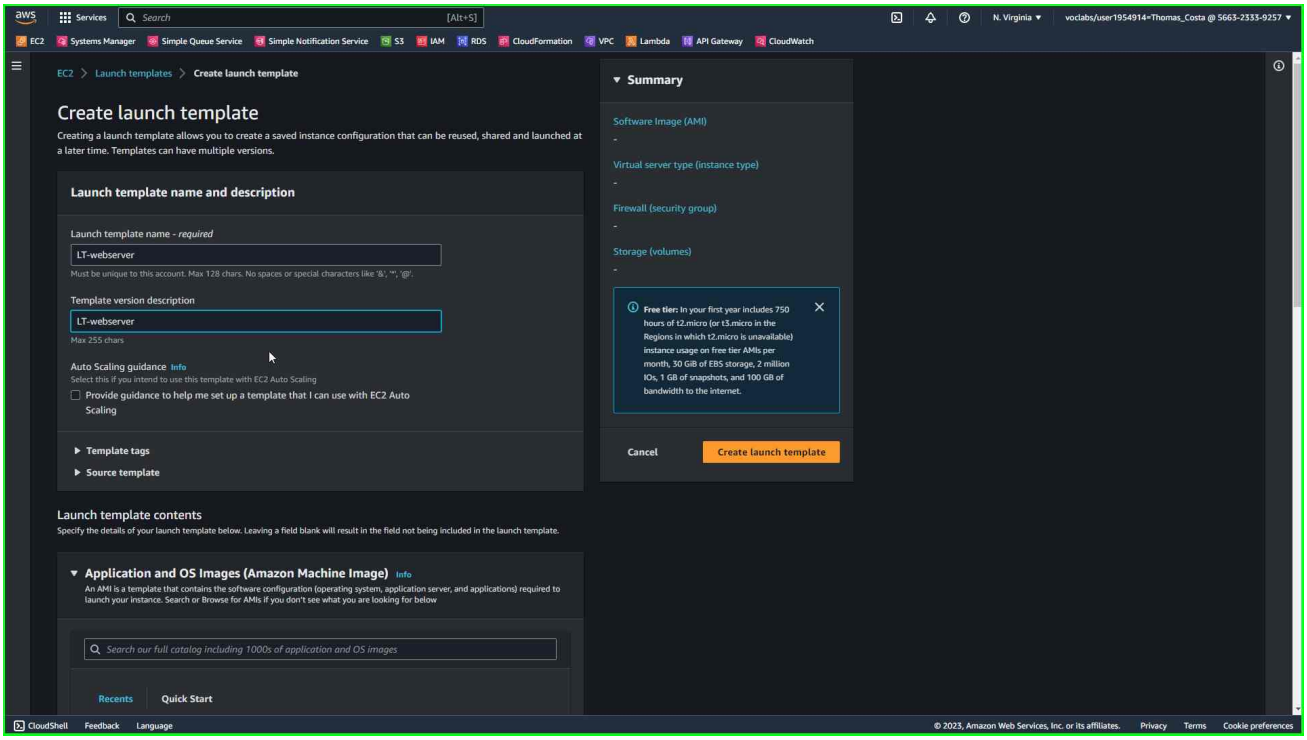


Implantando um Application Load Balancer (ALB) na AWS com EC2

Selecione o botão **Create launch template:**

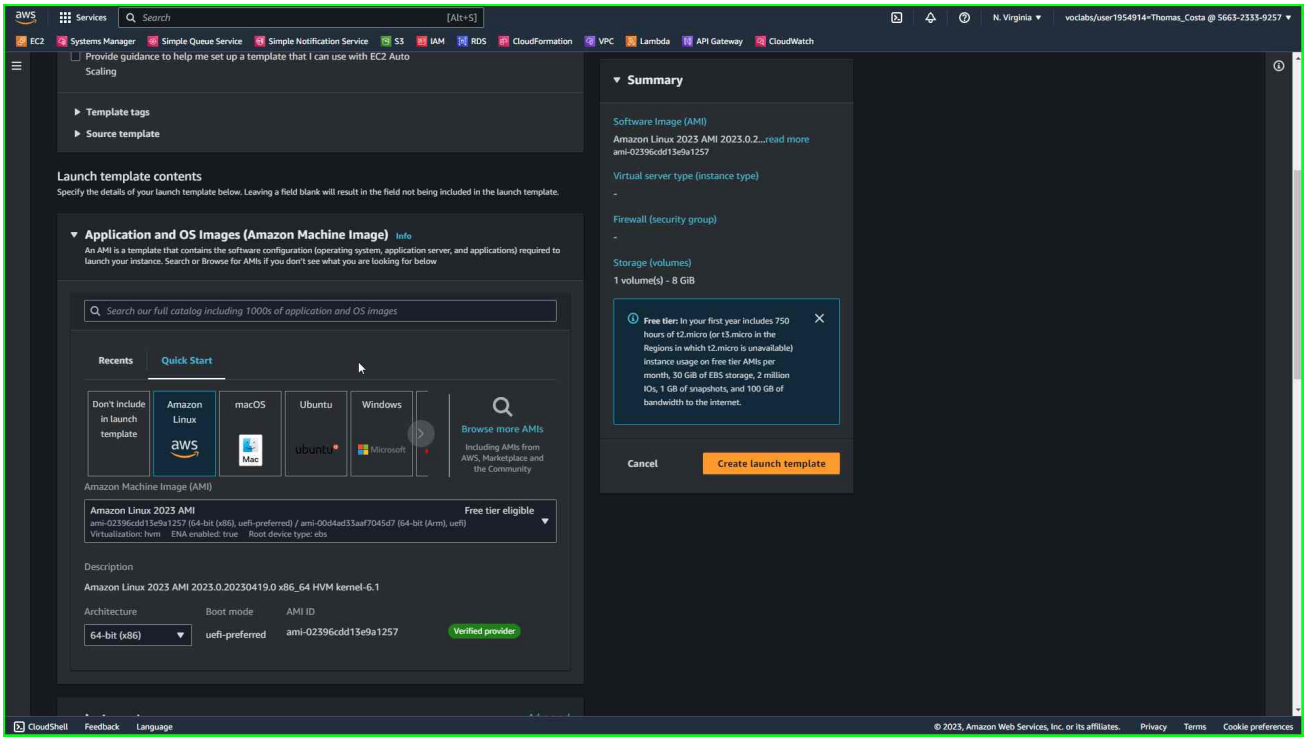


Incluir o nome do template, No nosso caso **LT-webserver**:

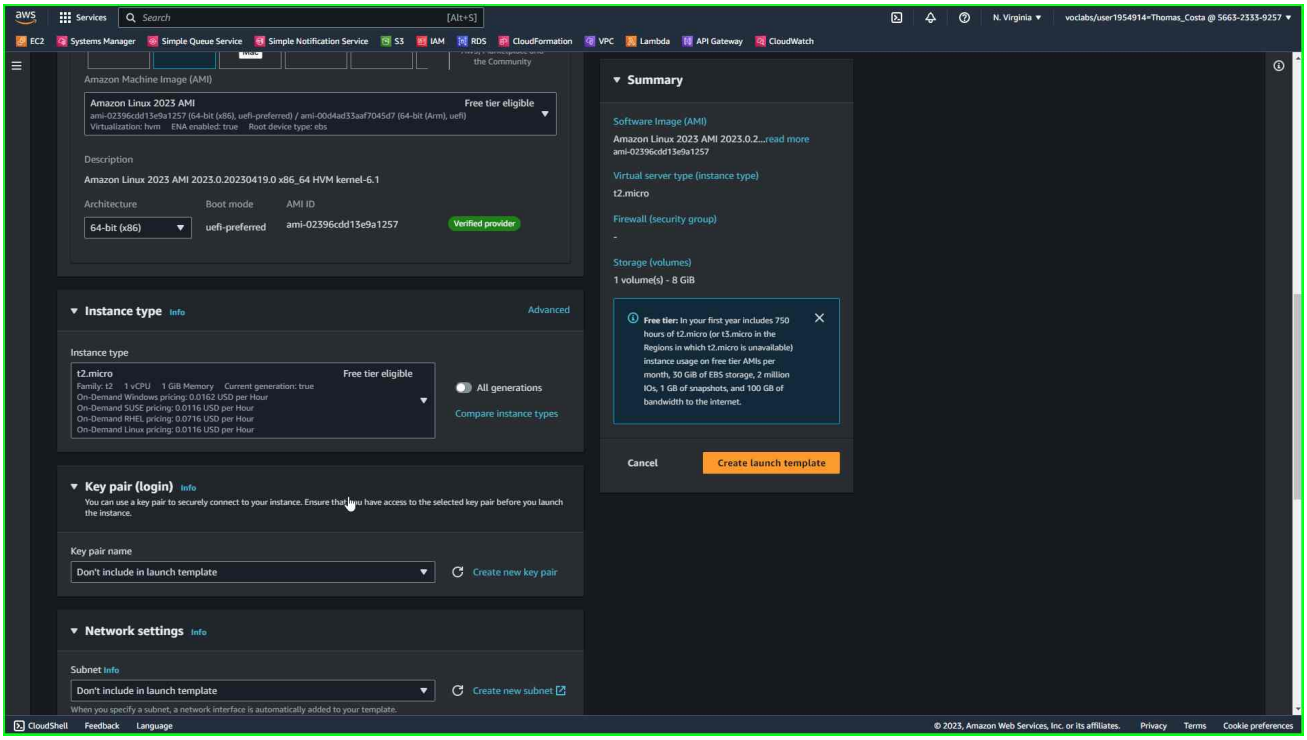


Implantando um Application Load Balancer (ALB) na AWS com EC2

Selecionar a imagem do Amazon Linux:

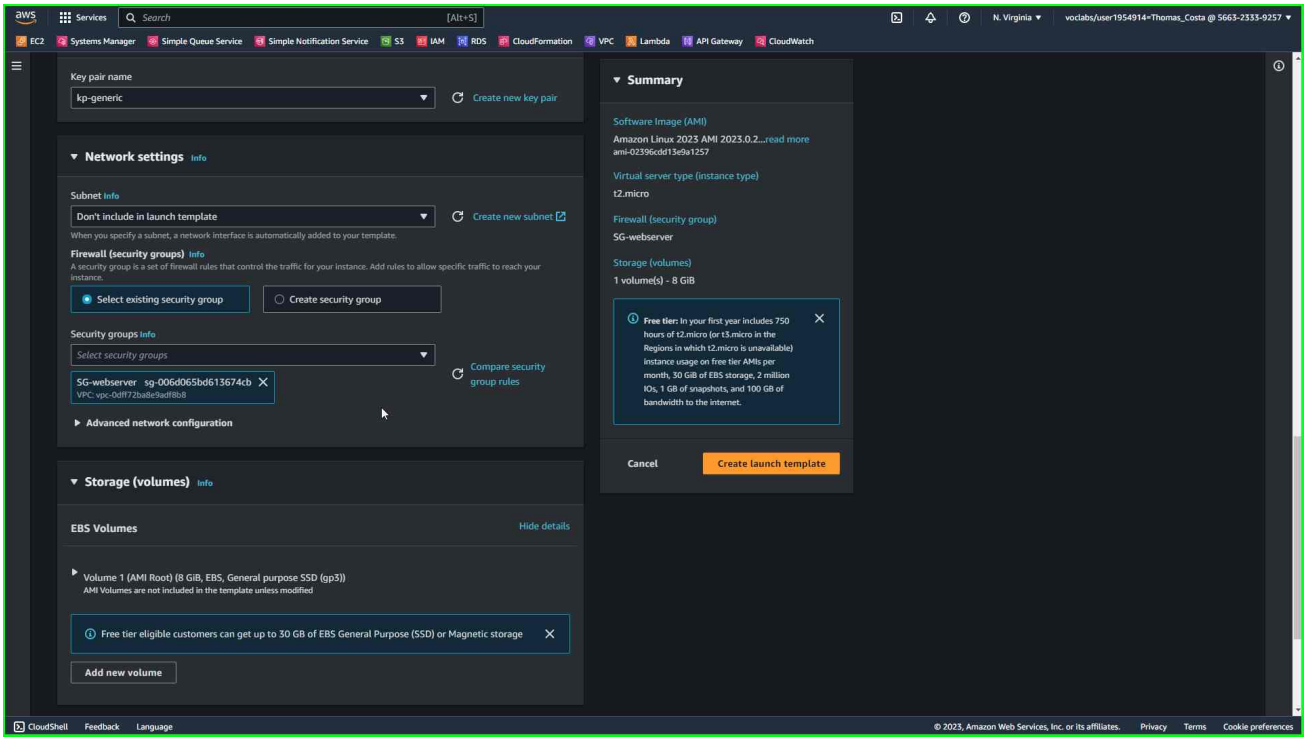


Escolher a instância do tipo t2.micro:



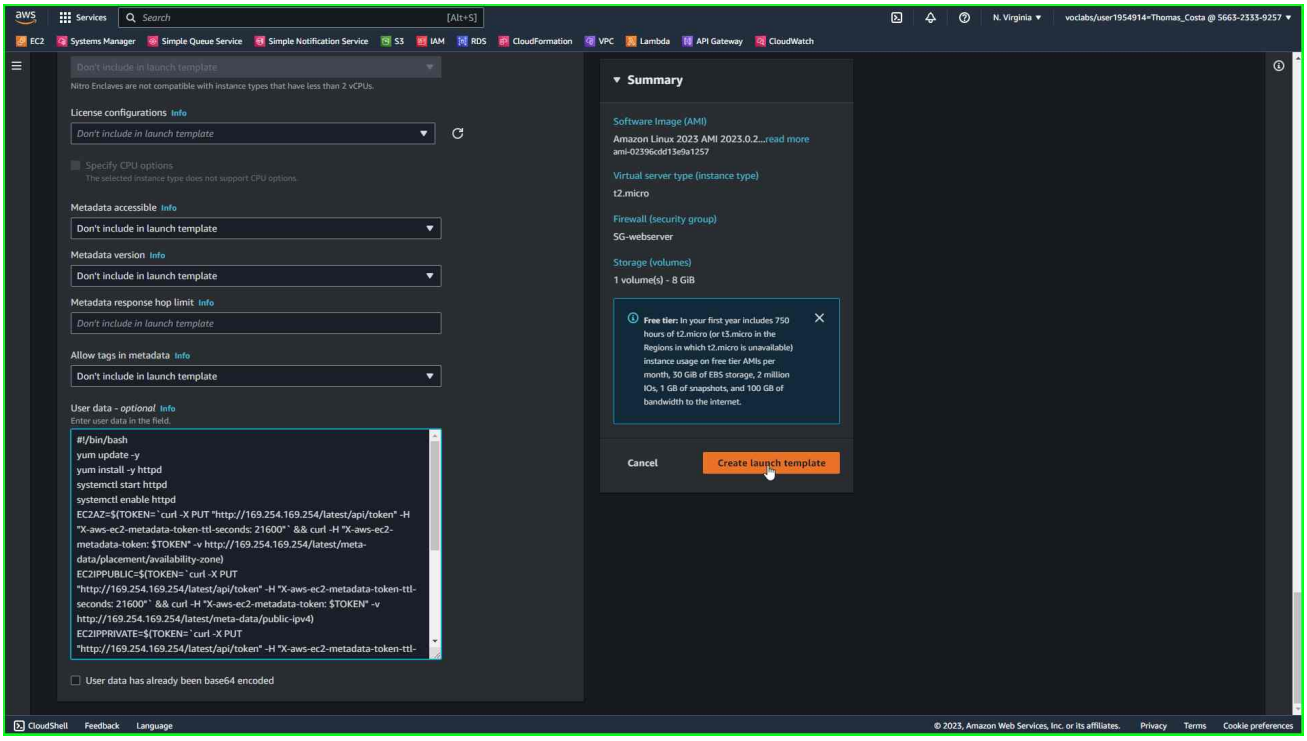
Implantando um Application Load Balancer (ALB) na AWS com EC2

Selecionar o **Security Group** chamado **SG-webserver**:



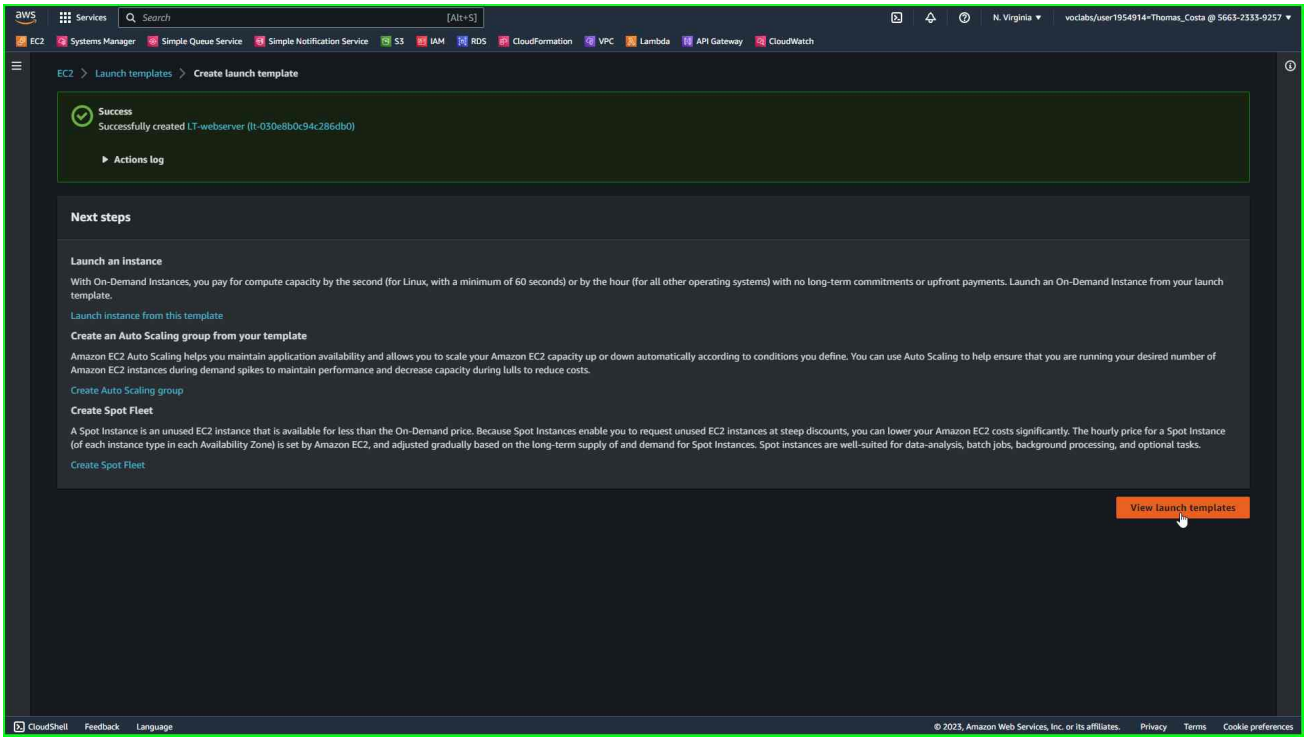
Incluir o script de inicialização no **User data** e clicar no botão **Create launch template**.

O script se encontra no final desse documento:

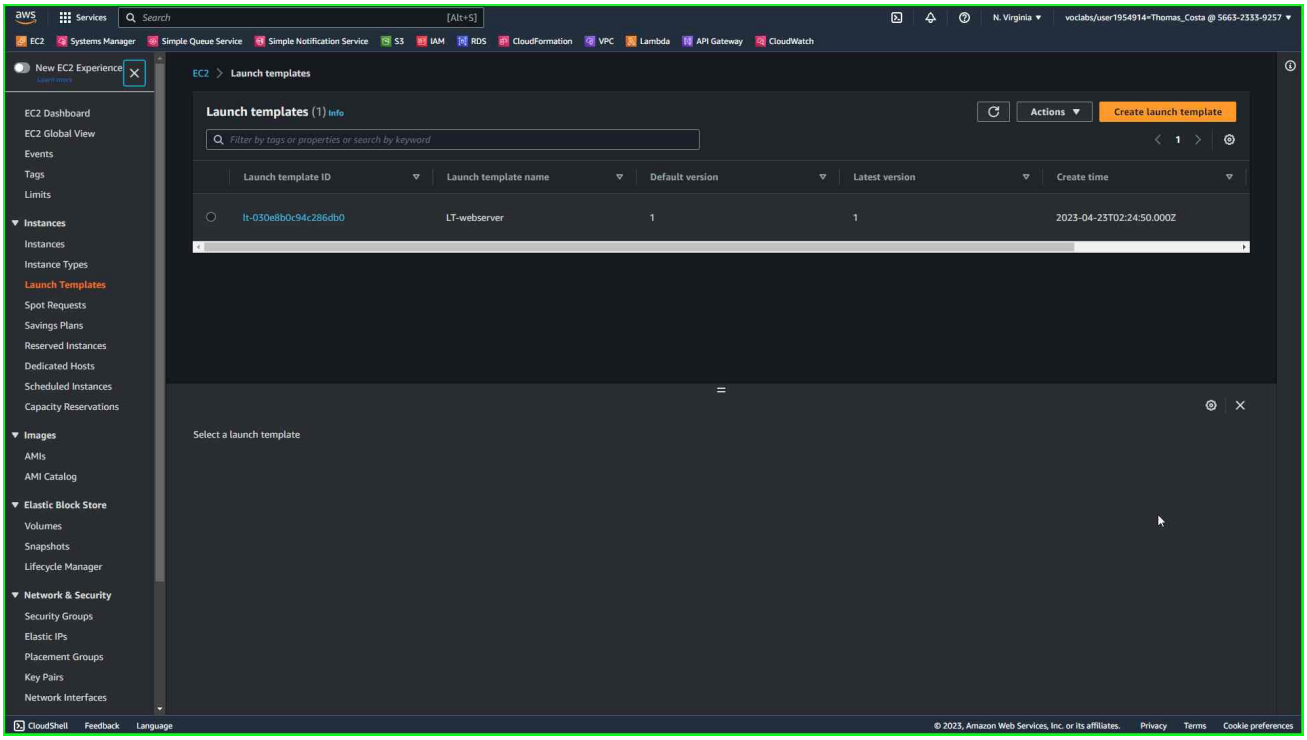


Implantando um Application Load Balancer (ALB) na AWS com EC2

Launch Template criado com sucesso:

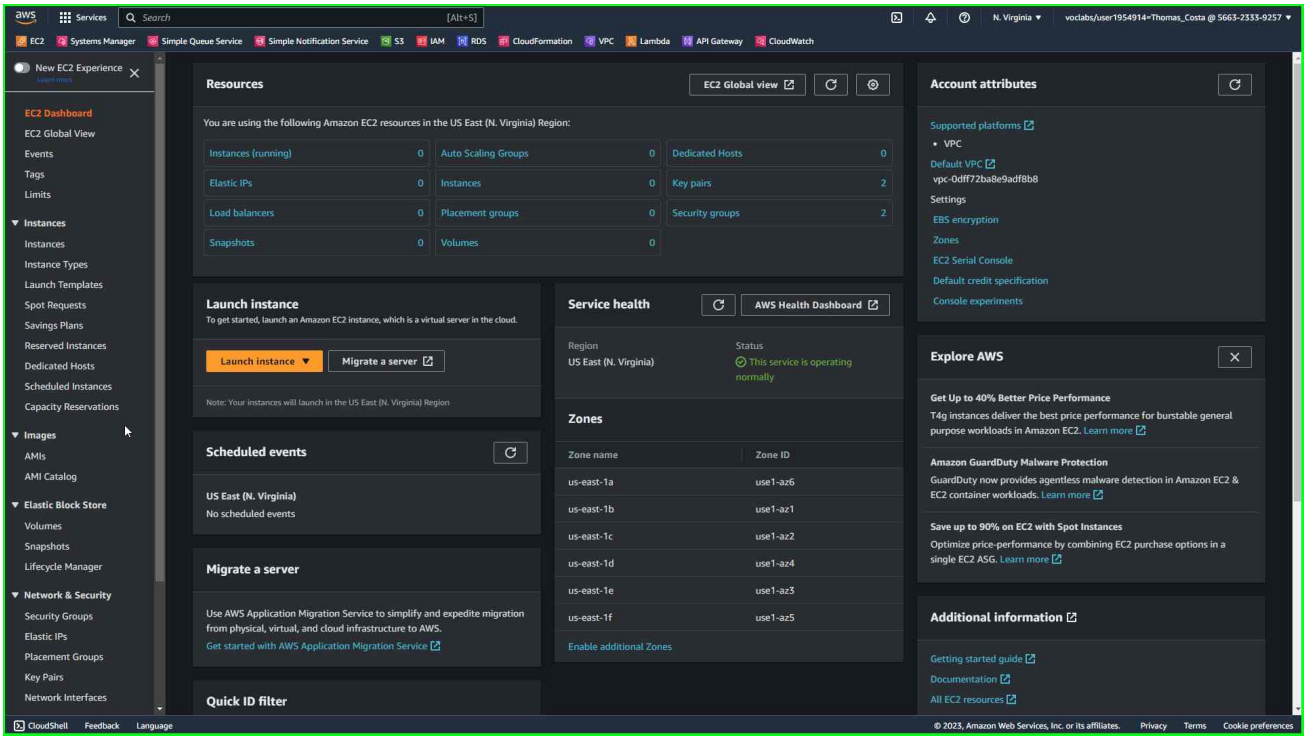


Entre na tela inicial do **Launch templates** e temos o template criado:

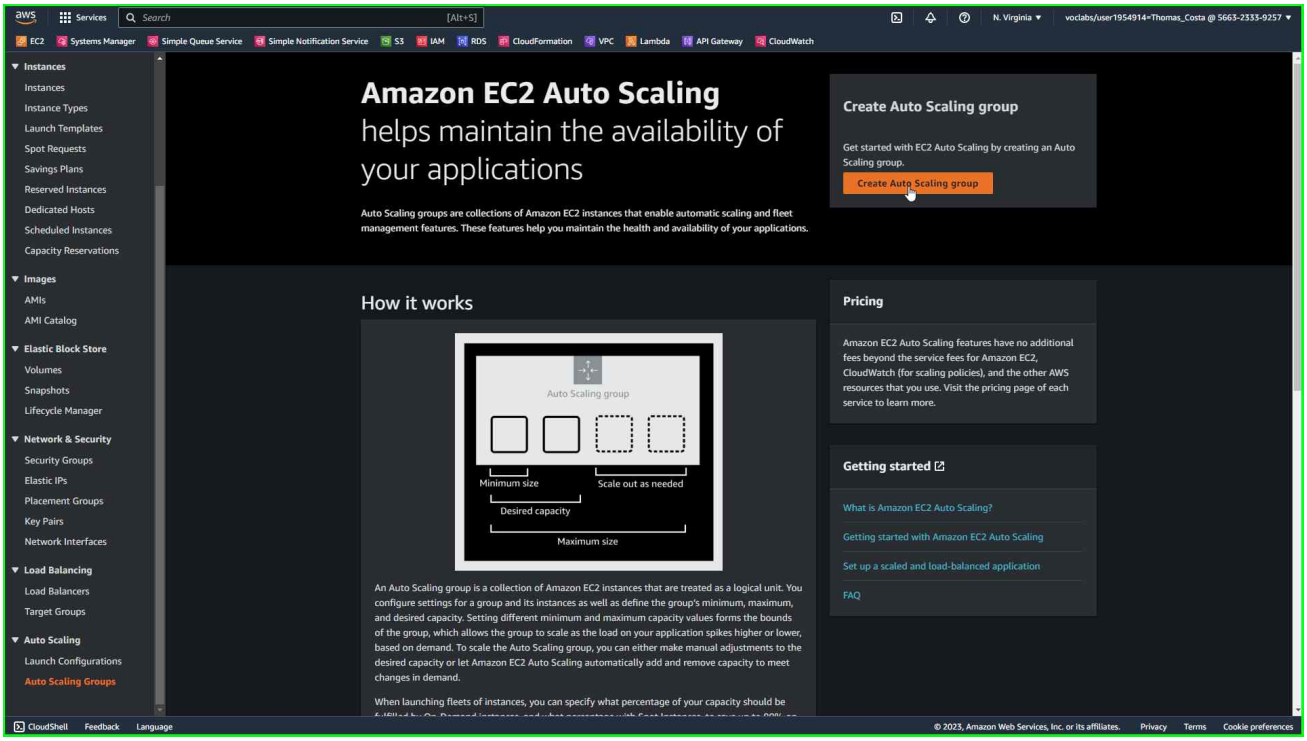


Parte 2 – Criando o EC2 Auto Scaling Group (ASG)

Selecione a opção **Auto Scaling Groups**:

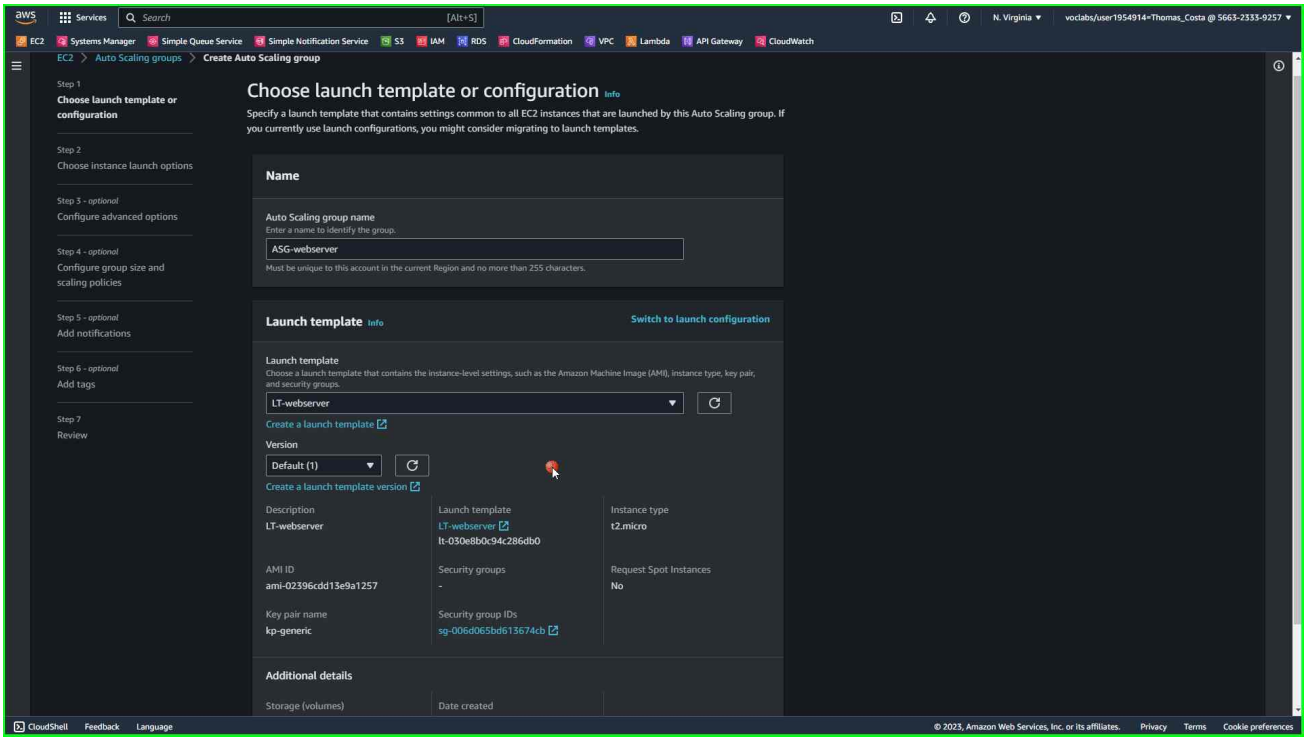


Clique no botão **Create Auto Scaling group**:

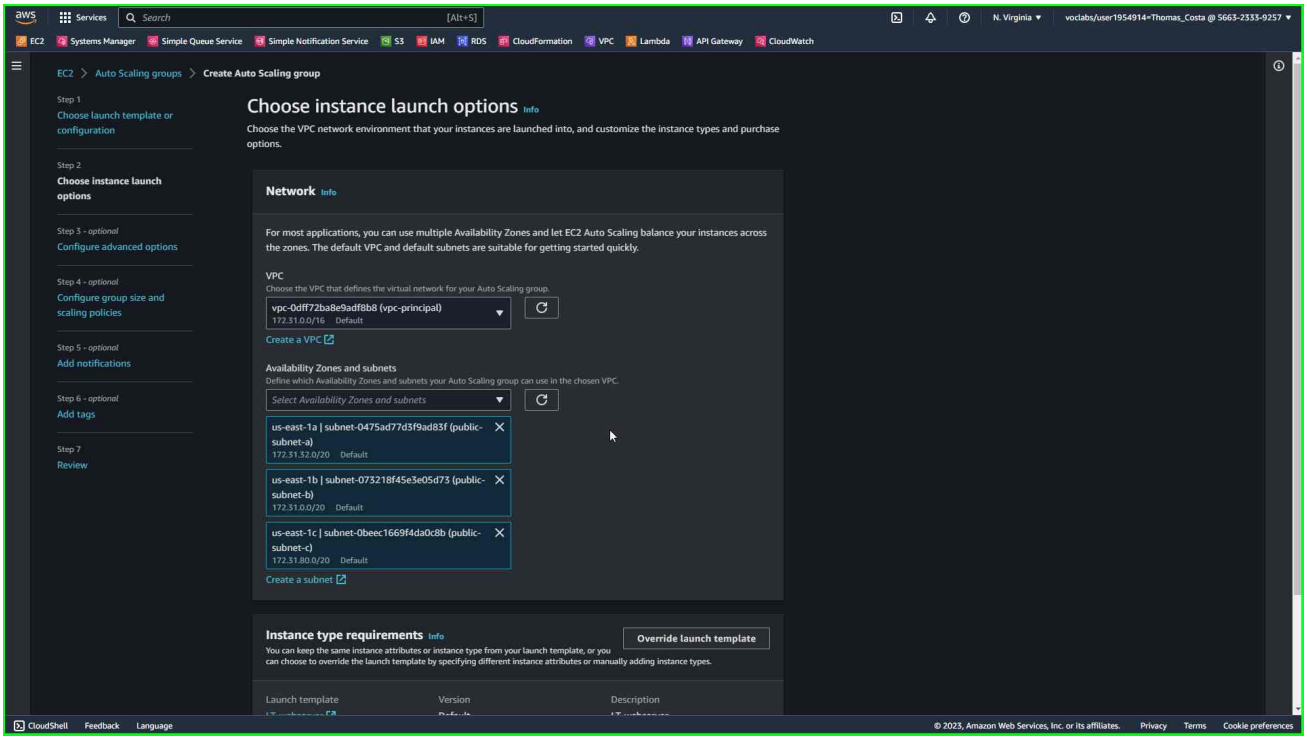


Implantando um Application Load Balancer (ALB) na AWS com EC2

Inclua o nome do ASG. No nosso exemplo chama-se **ASG-webserver**:

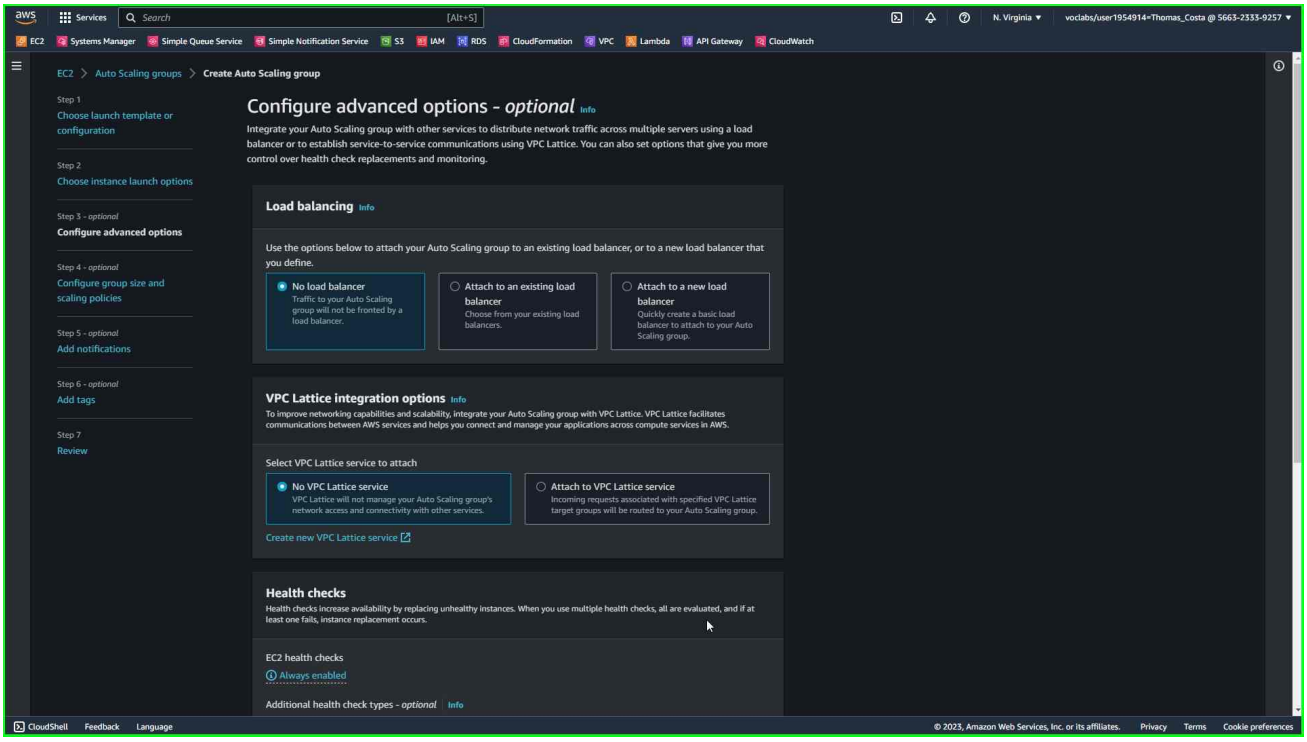


Selecione no mínimo 3 AZ para o ASG:

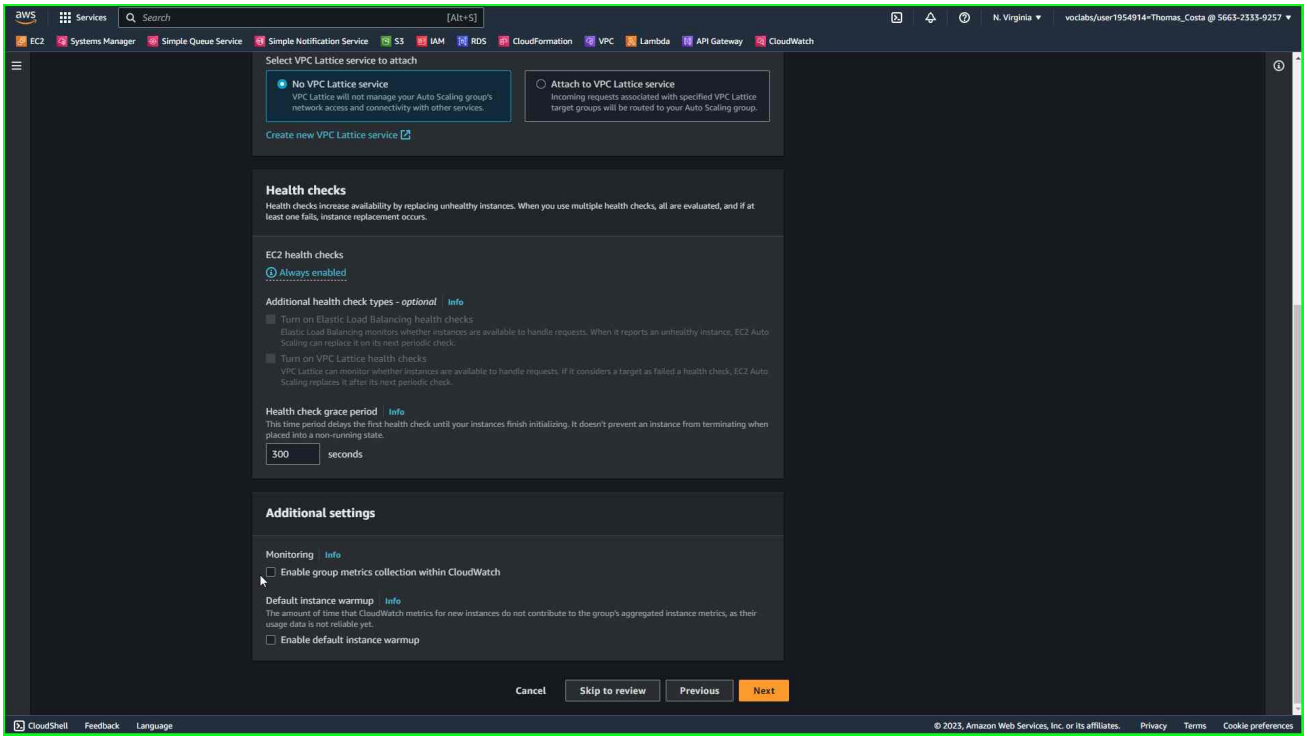


Implantando um Application Load Balancer (ALB) na AWS com EC2

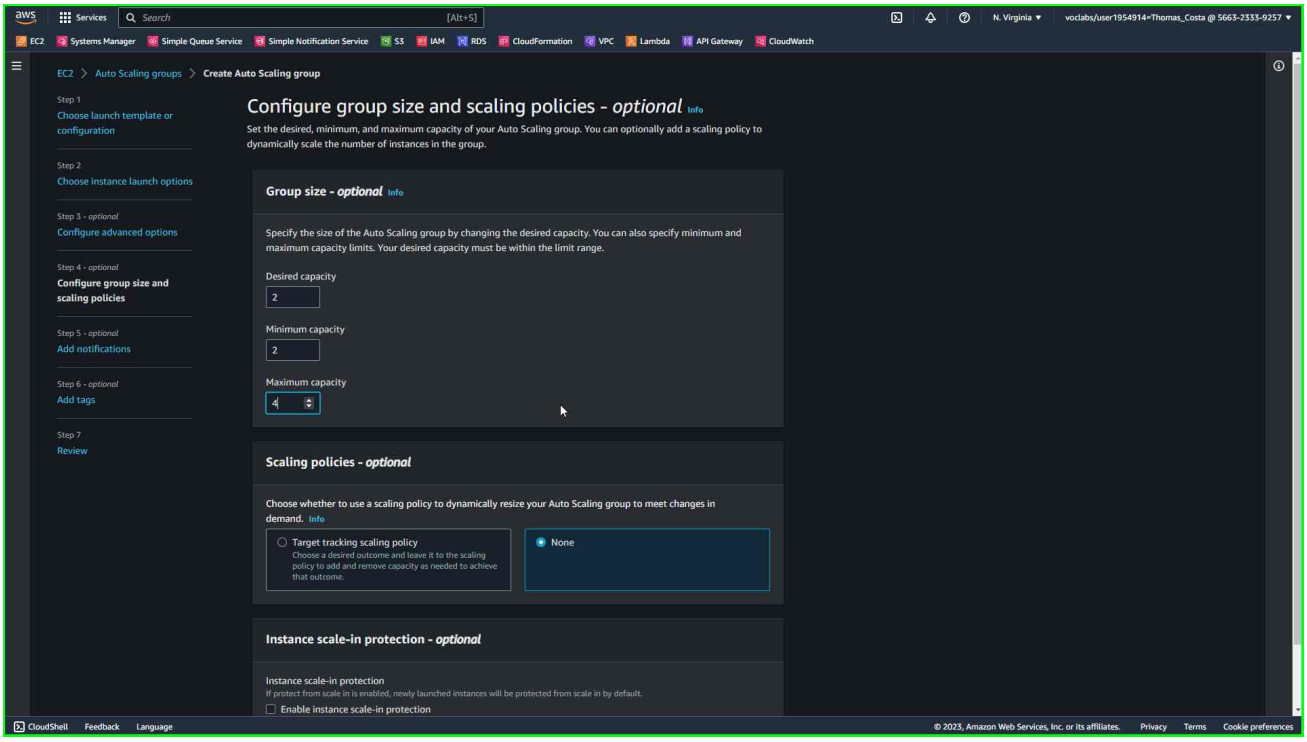
Selecione a opção **No load balancer**:



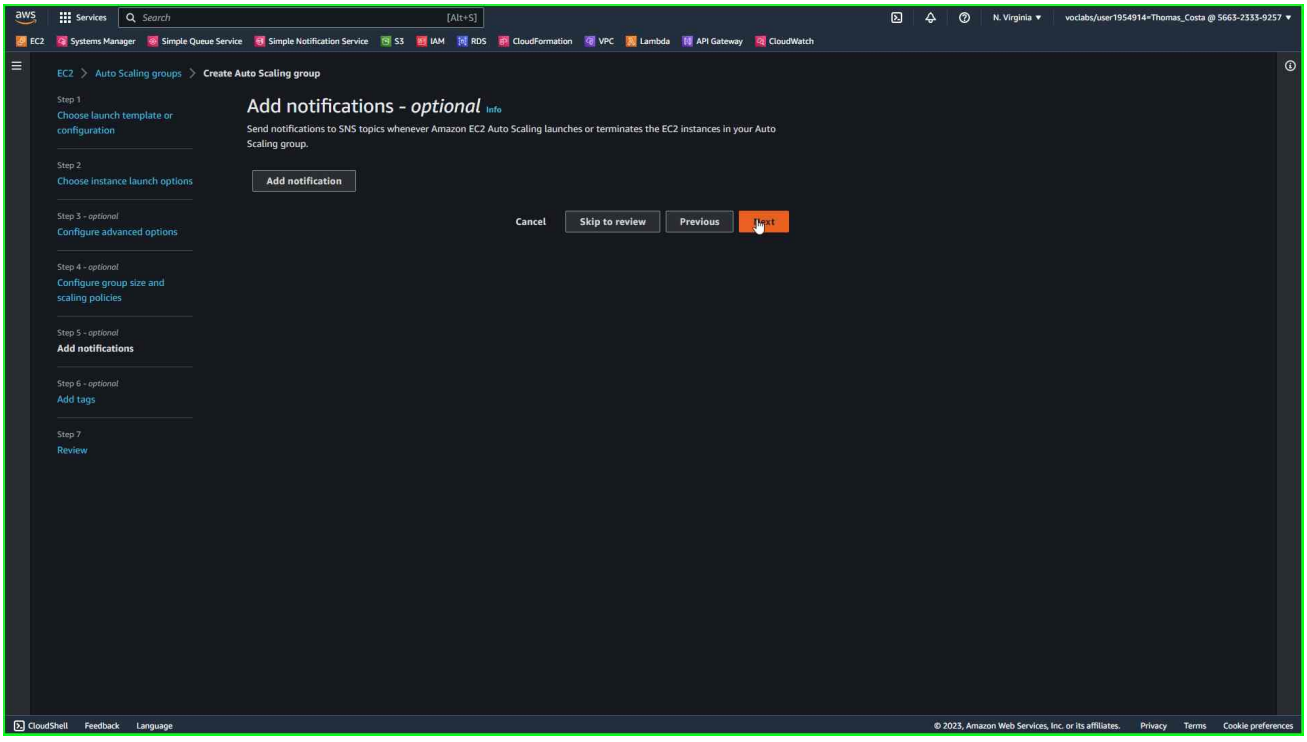
Selecione a opção **Enable group metrics collection within CloudWatch**:



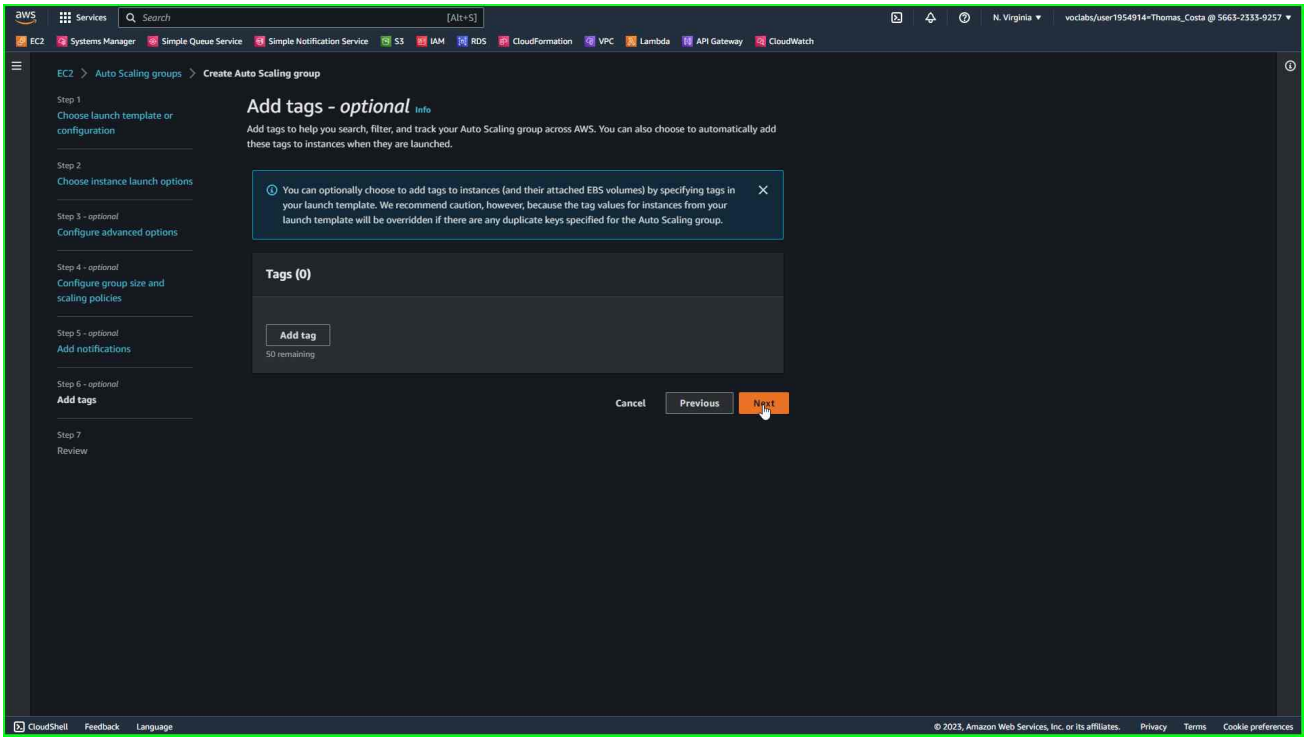
Inclua os seguintes valores para os servidores: **Desired Capacity e Minimum Capacity é igual a 2 Maximum Capacity é igual a 4**:



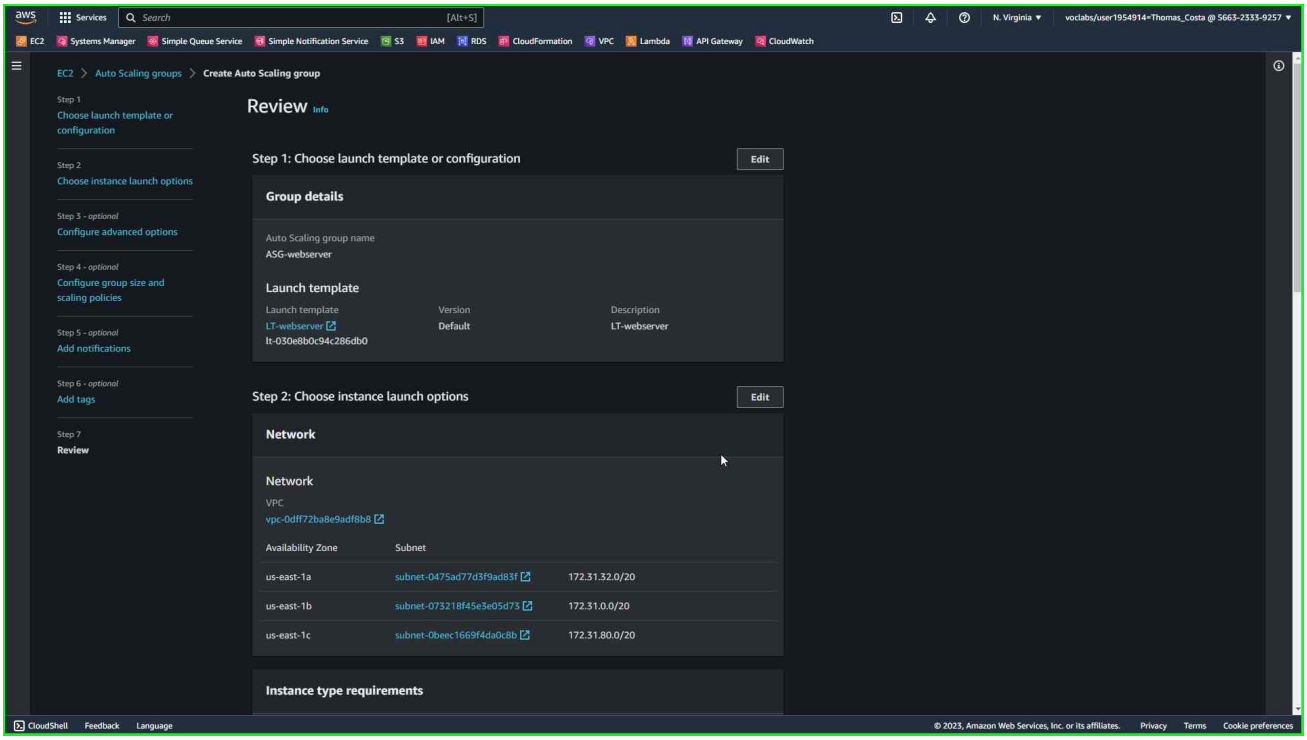
Na próxima tela não tem ação somente clique em **Next**:



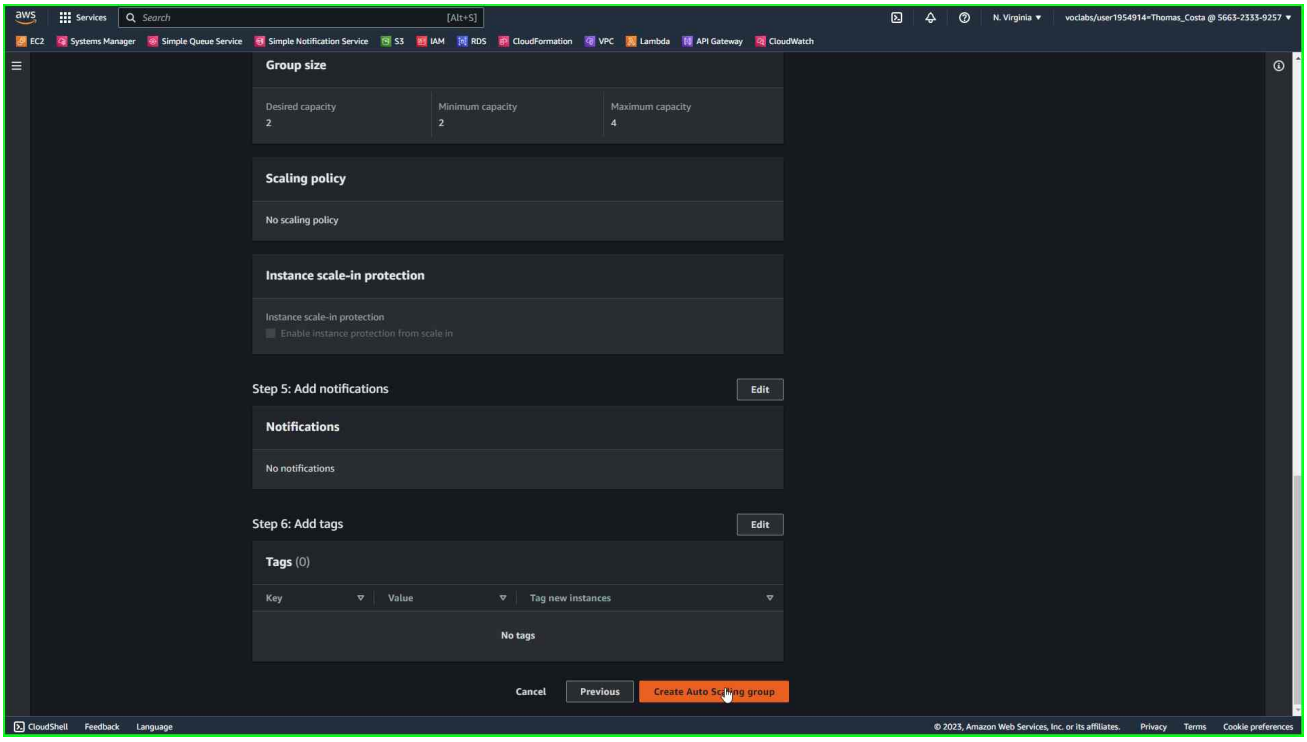
Na próxima tela não tem ação somente clique em **Next**:



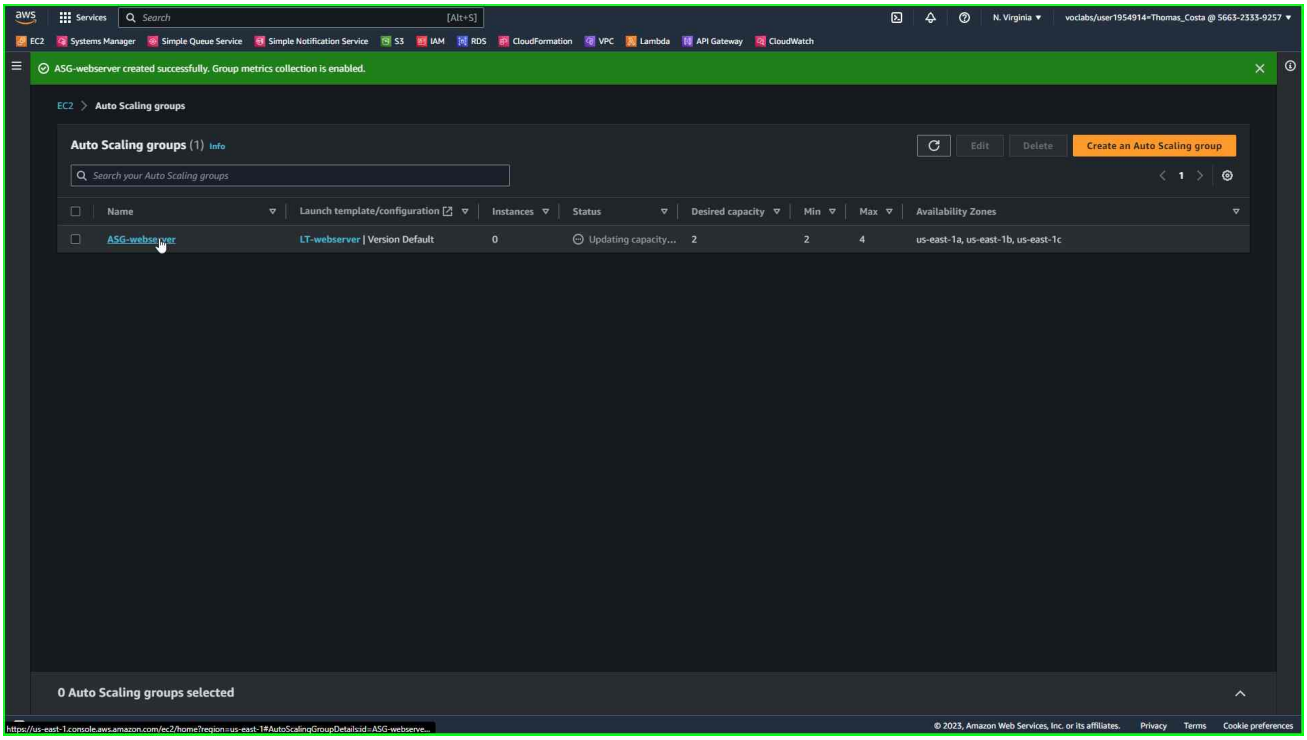
Faça o **Review** da criação do ASG:



Confirme a criação:

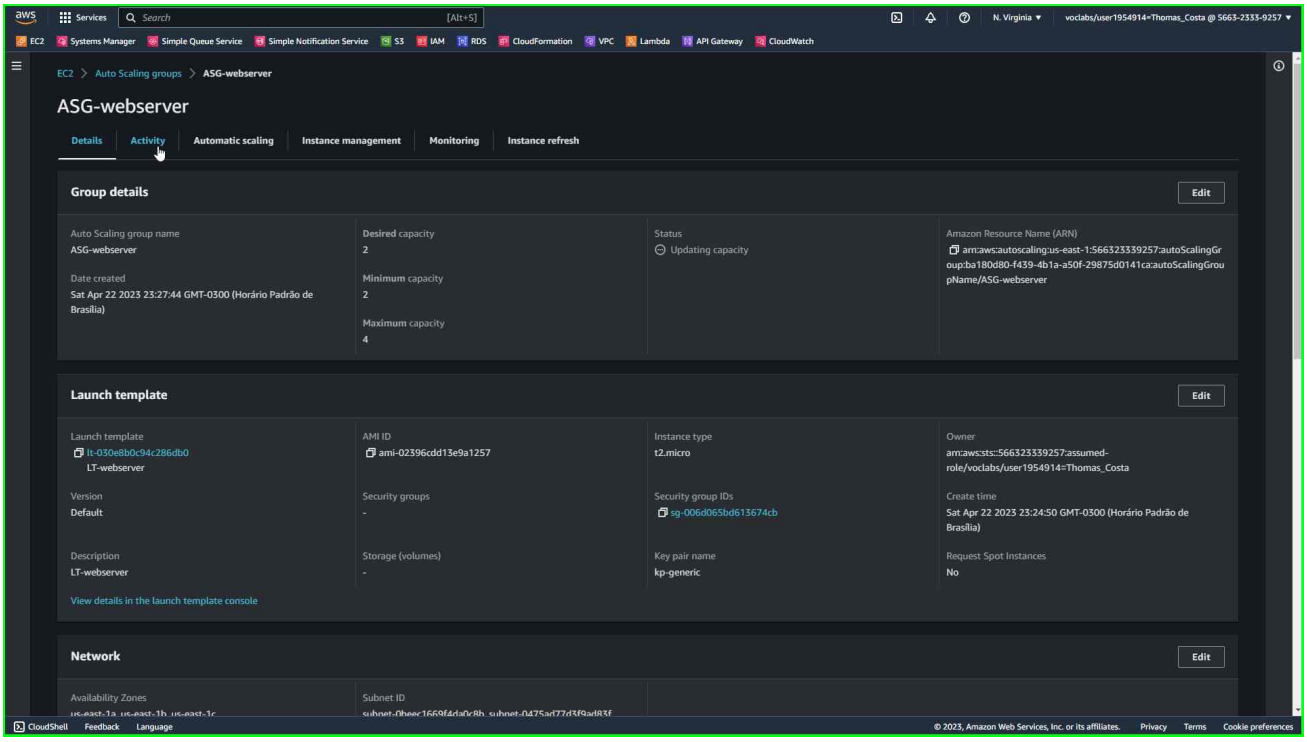


Com ASG criado com sucesso, selecione para ver suas propriedades:

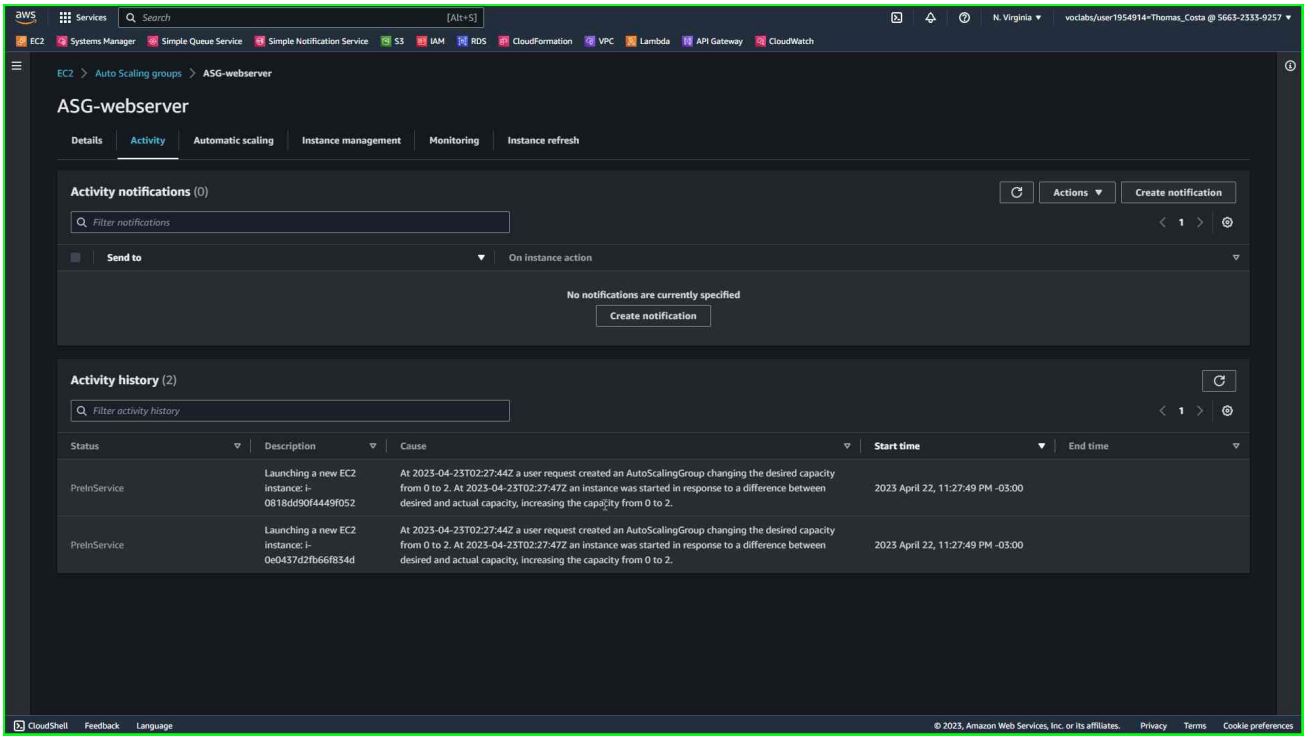


Implantando um Application Load Balancer (ALB) na AWS com EC2

Selecione a aba **Activity**:

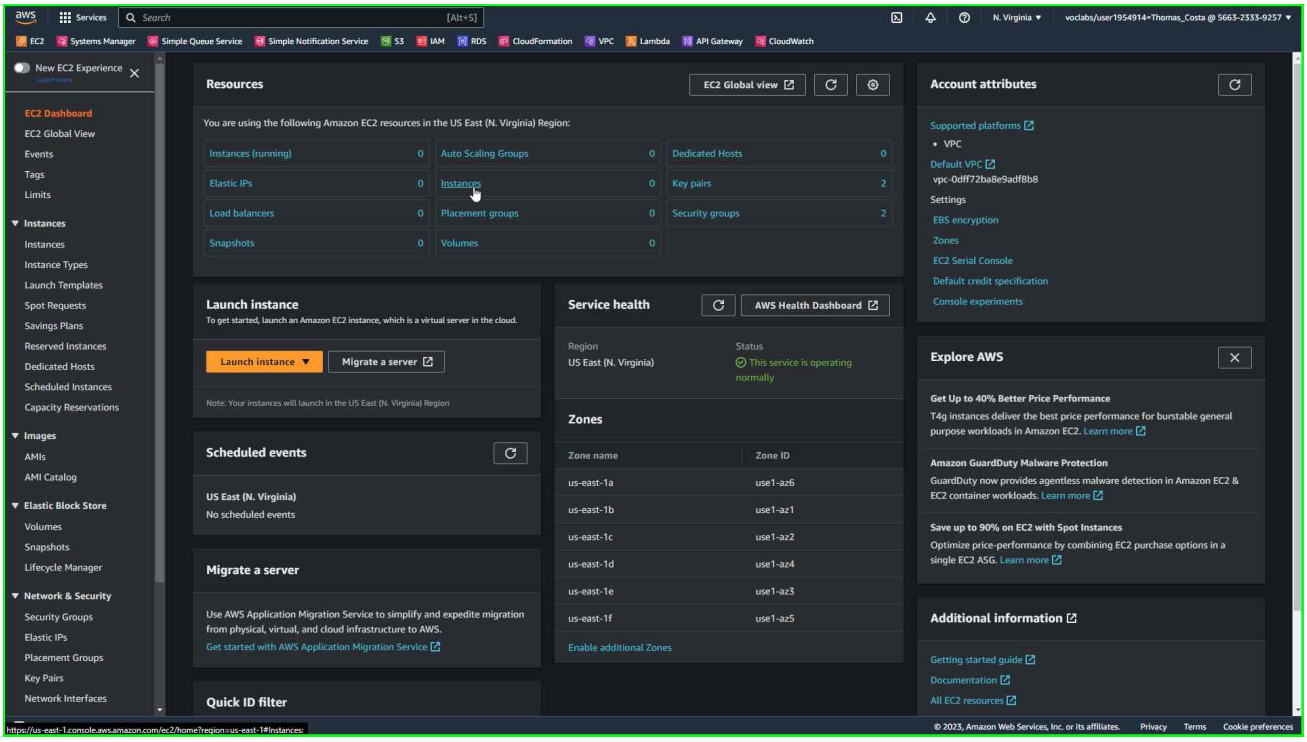


Em **Activity History** podemos ver que a criação dos EC2 foi disparada:

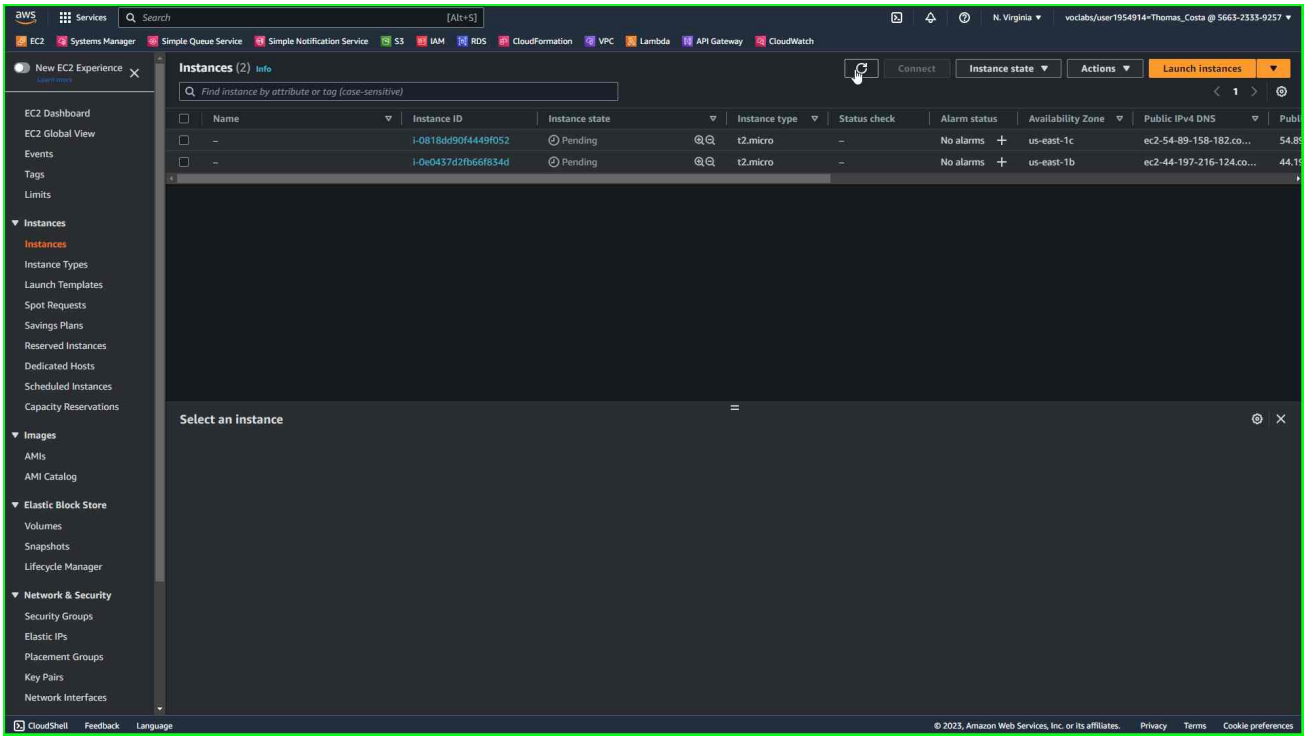


Implantando um Application Load Balancer (ALB) na AWS com EC2

Voltando para a tela principal do EC2 selecionando a opção **Instances**:

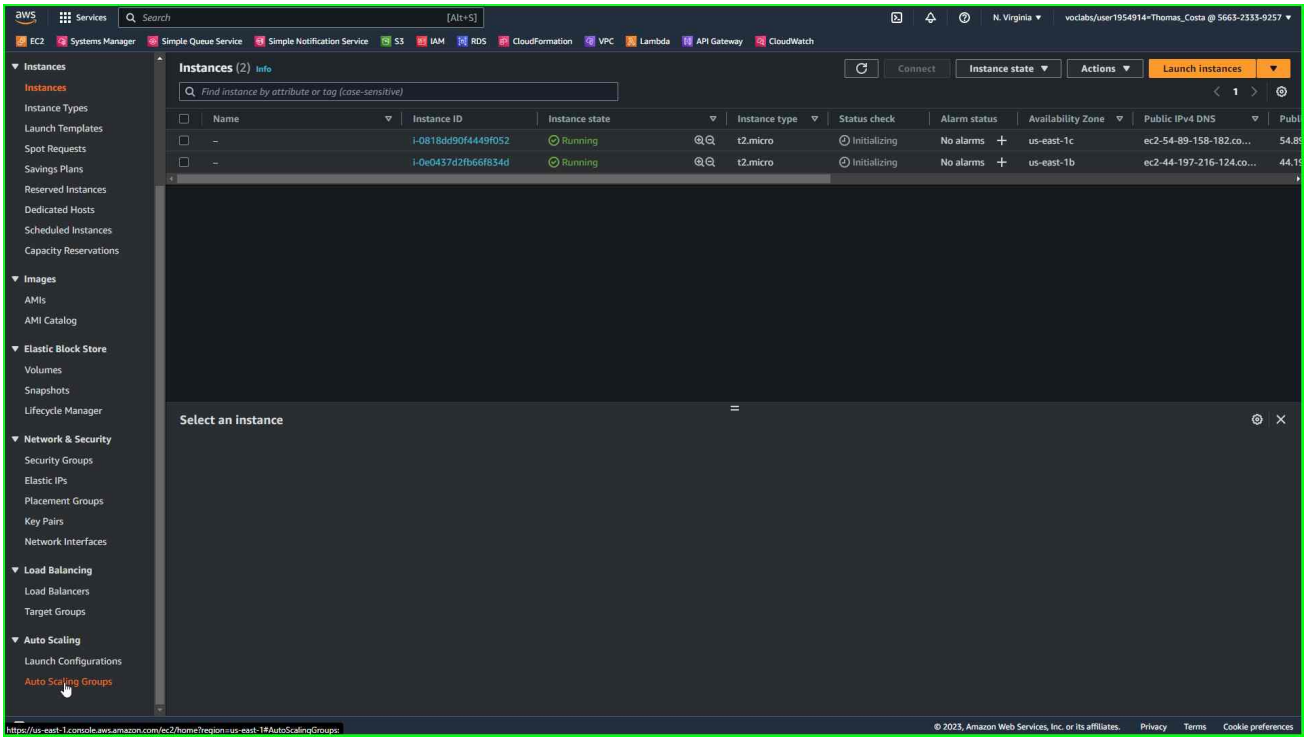


Duas instâncias estão sendo criadas de acordo com os parâmetros do ASG:

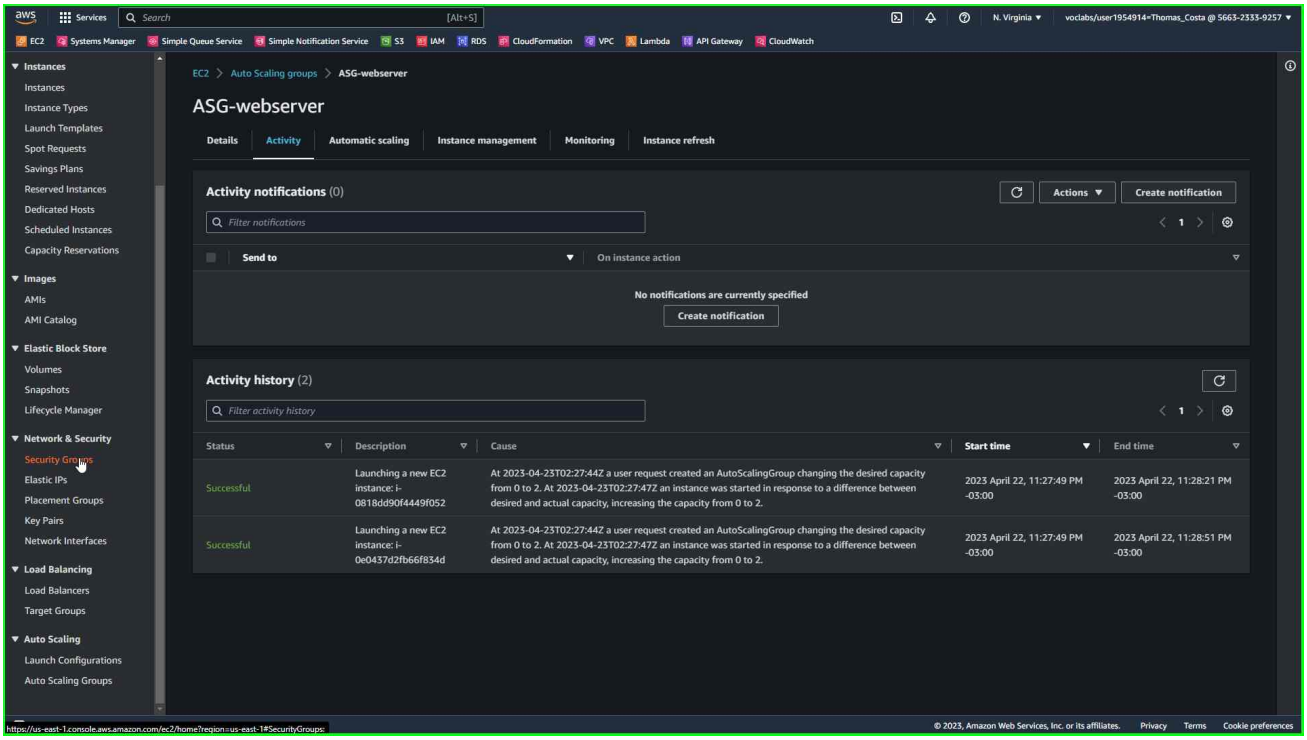


Implantando um Application Load Balancer (ALB) na AWS com EC2

Instâncias criadas com sucesso:

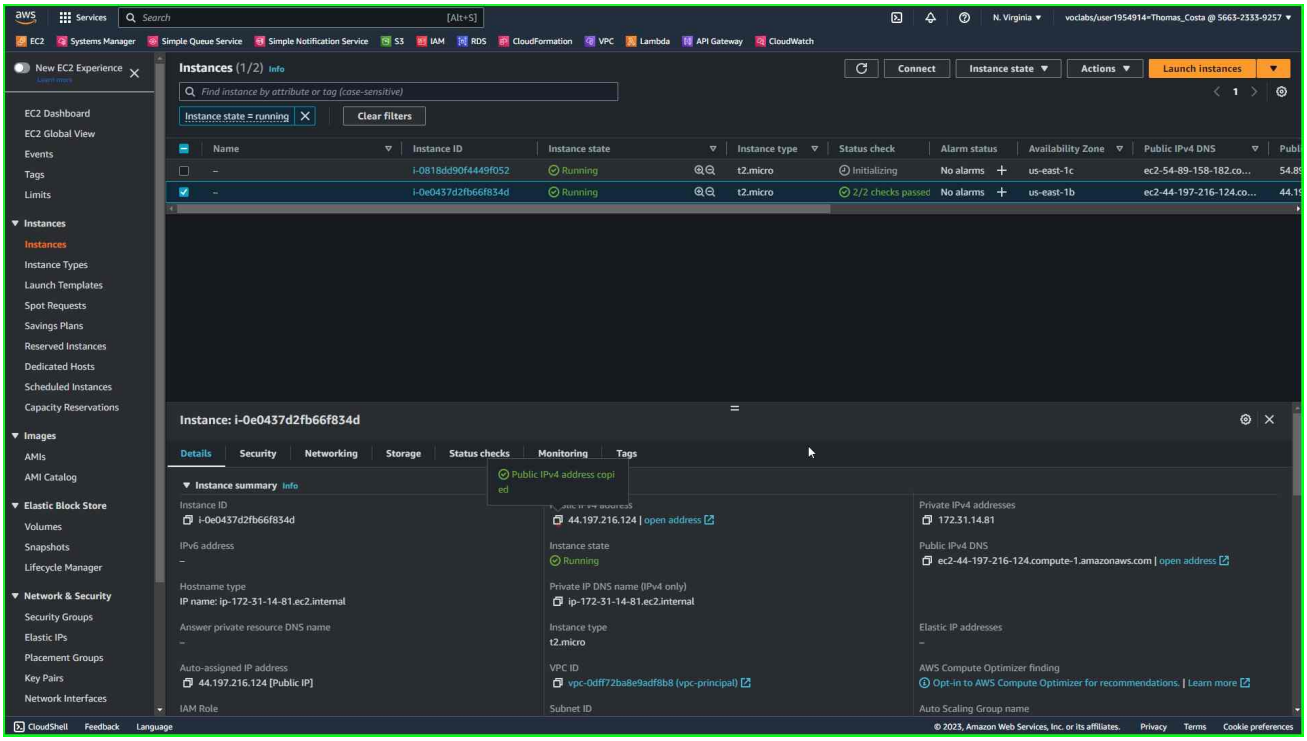


Em **Activity History** do ASG foi notificado que as duas instâncias foram criadas:



Implantando um Application Load Balancer (ALB) na AWS com EC2

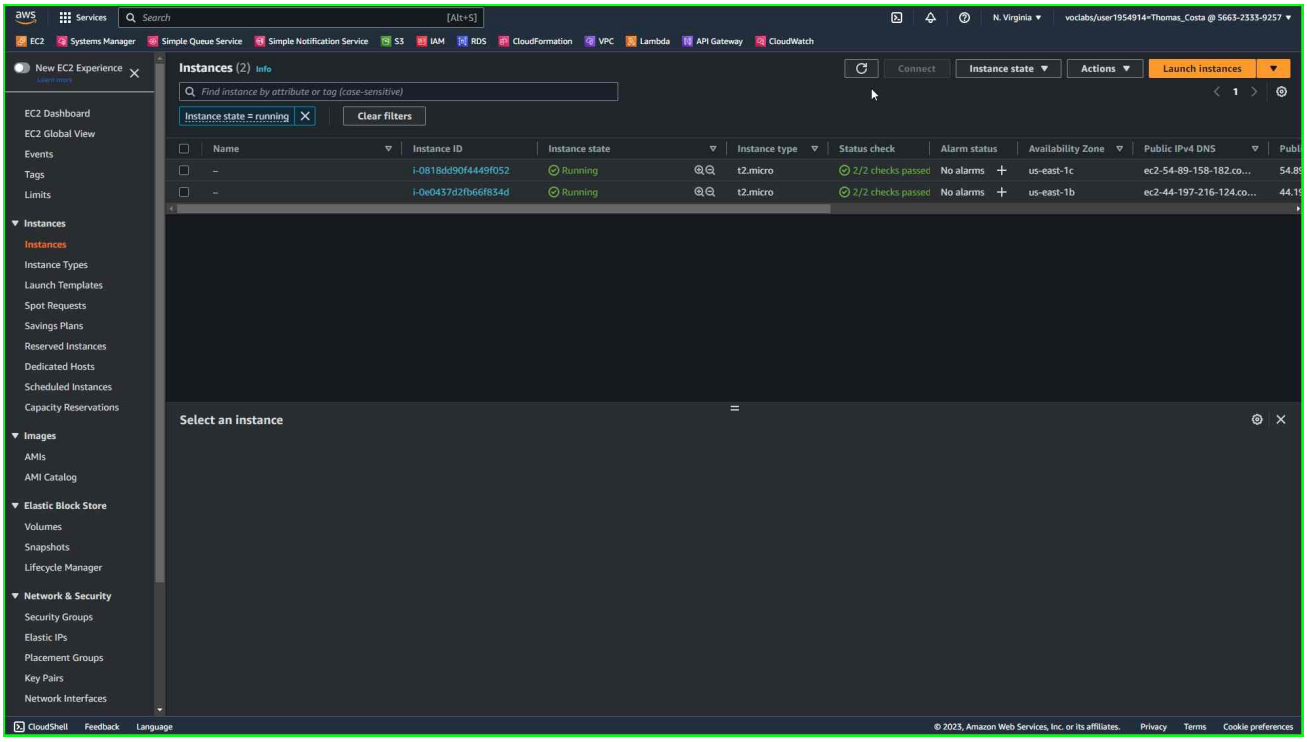
Selecione o endereço público de uma das instâncias para acessar o servidor Apache HTTP:



Resposta do servidor Apache HTTP no navegador:

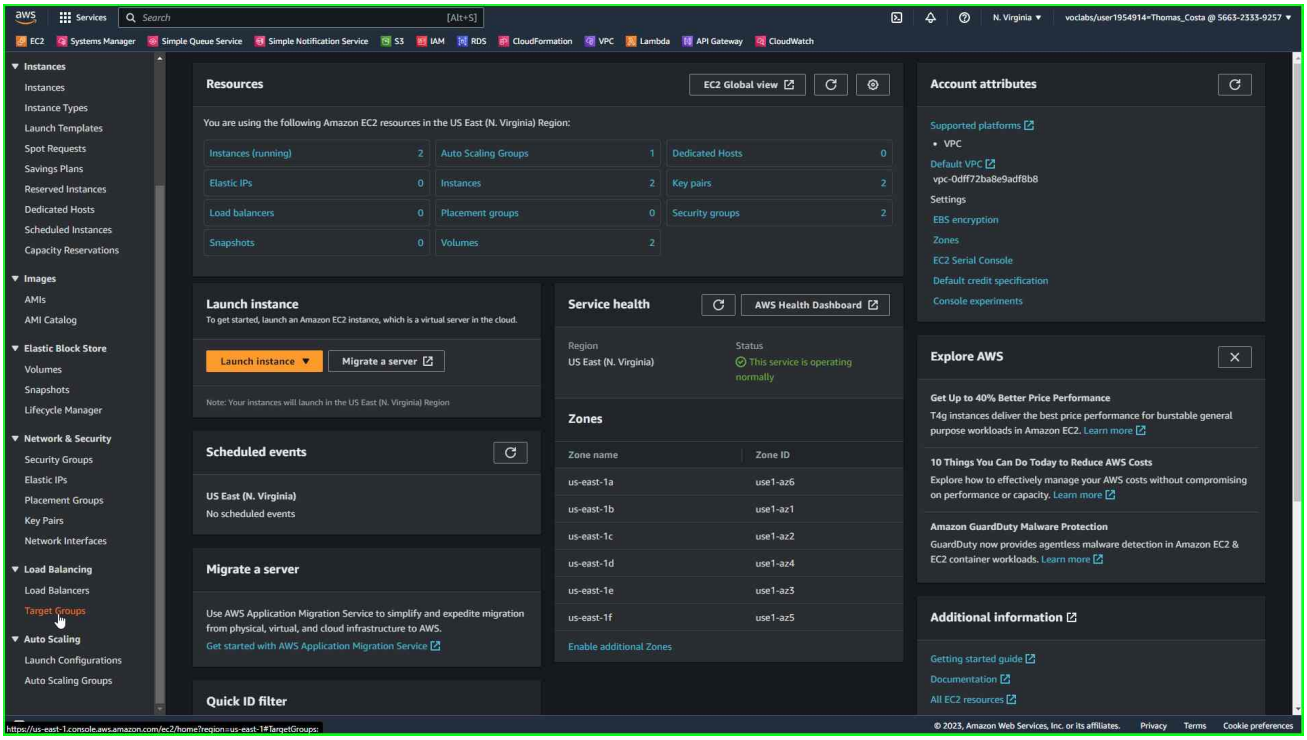
Implantando um Application Load Balancer (ALB) na AWS com EC2

Instâncias criadas com sucesso e **Status check** em ordem:

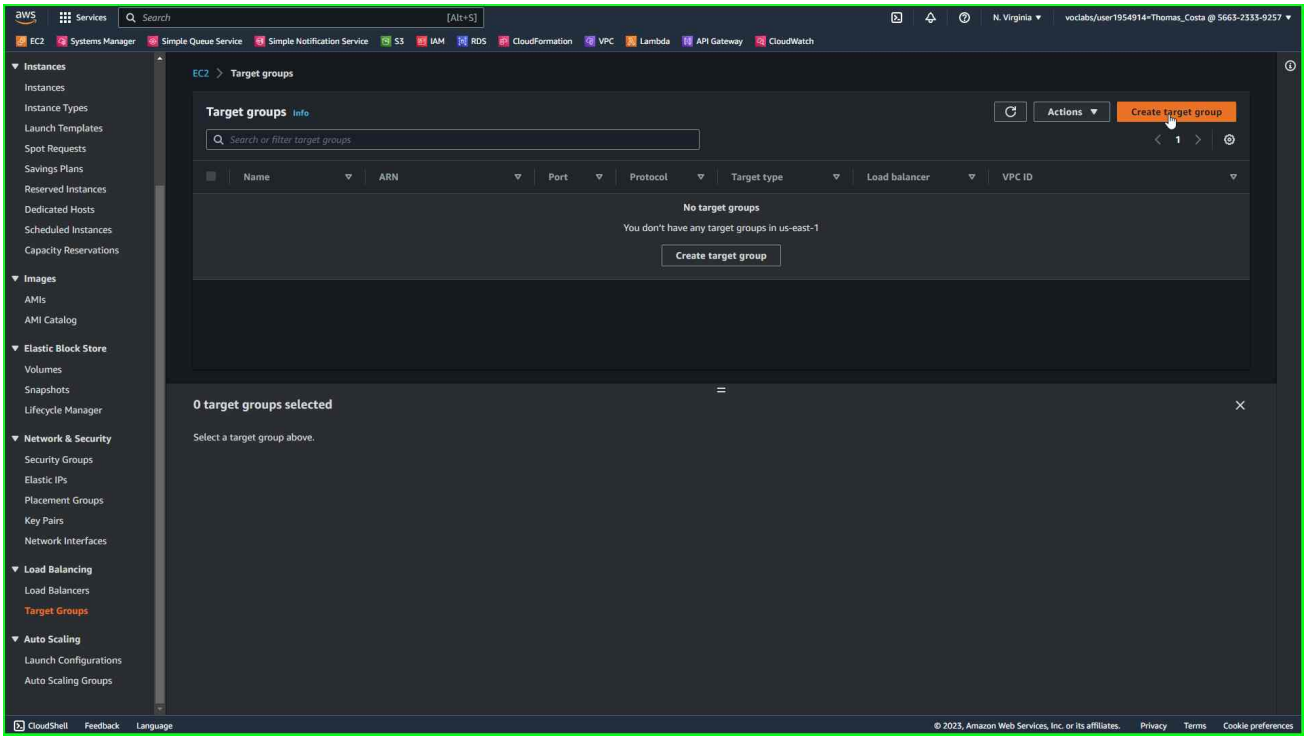


Parte 3 – Criando o Target Group

Selecionar a opção **Target Groups**:

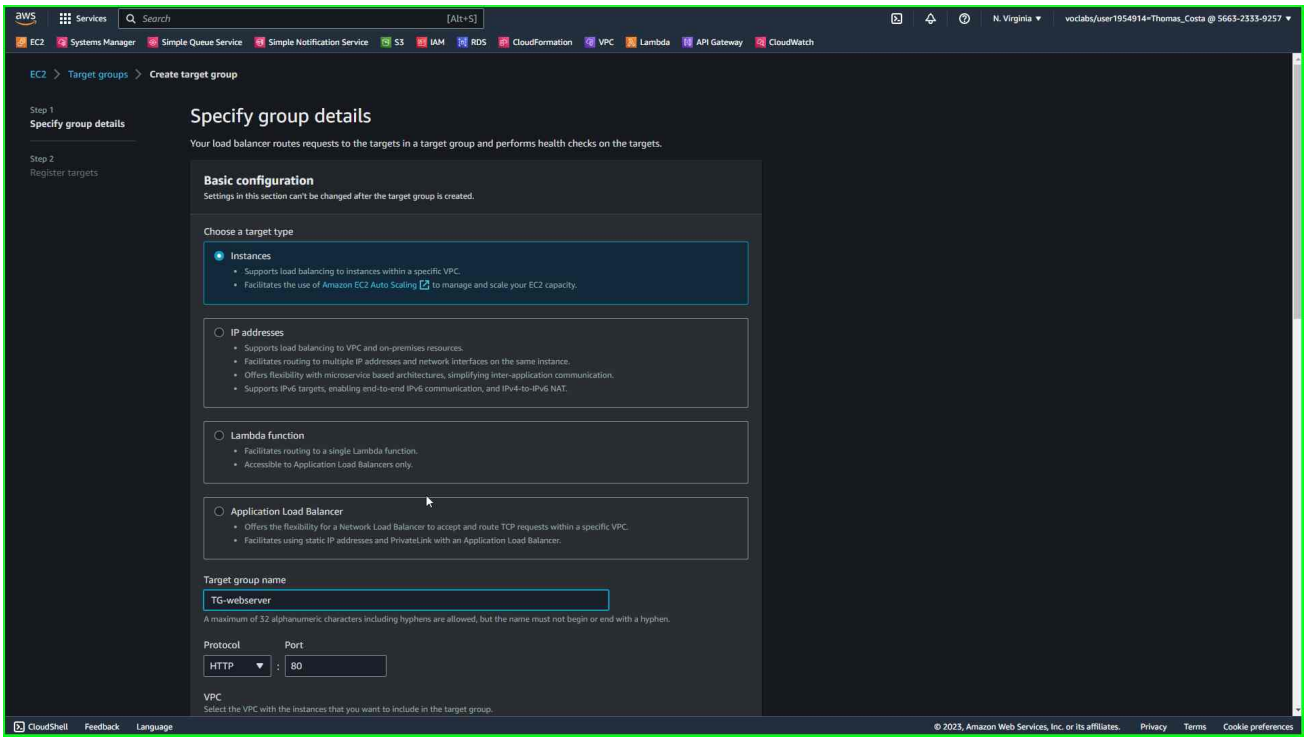


Na tela principal, clicar no botão **Create target group**:

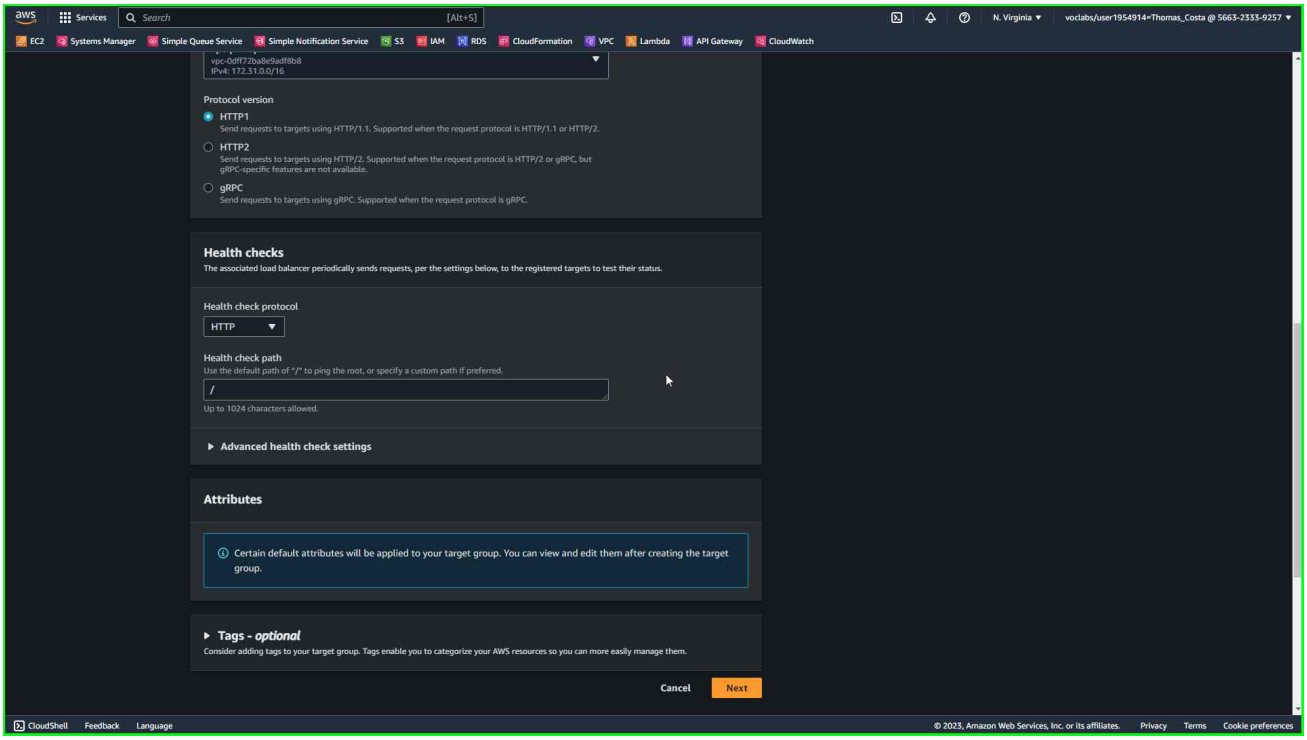


Implantando um Application Load Balancer (ALB) na AWS com EC2

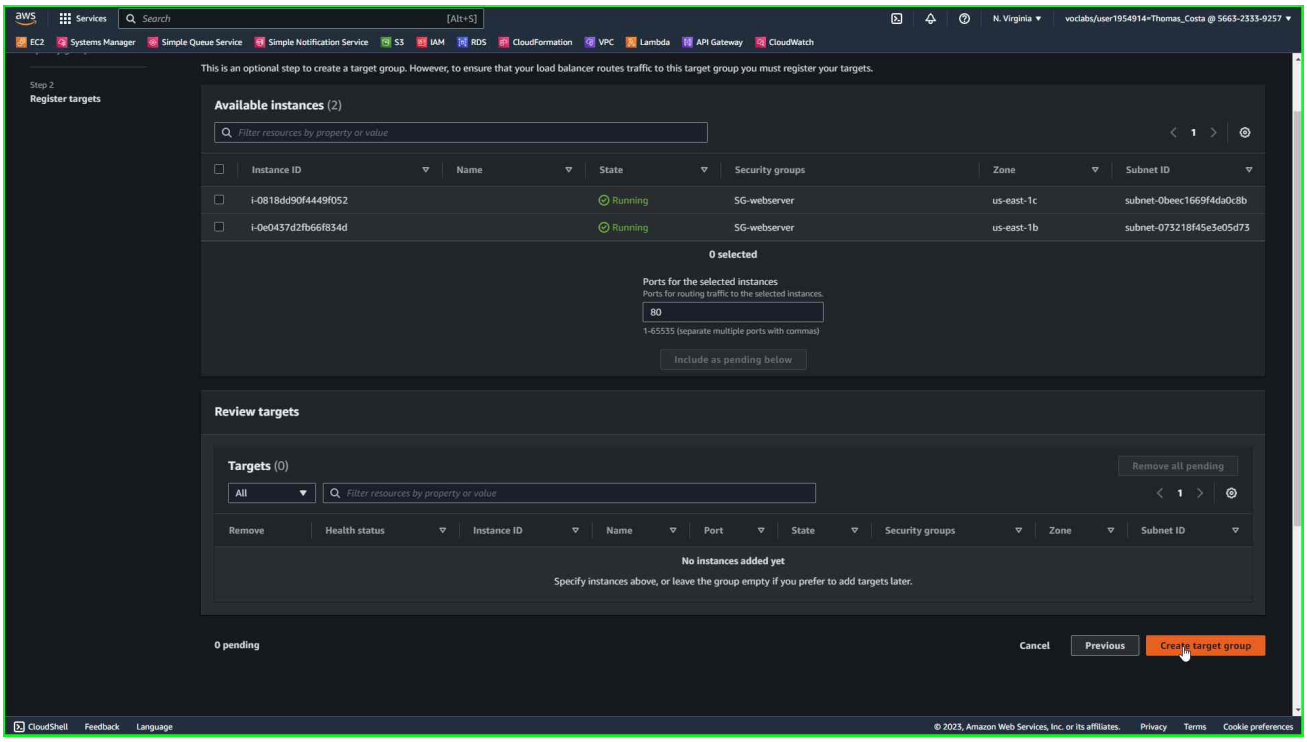
Colocar o nome do Target group de **TG-webserver**:



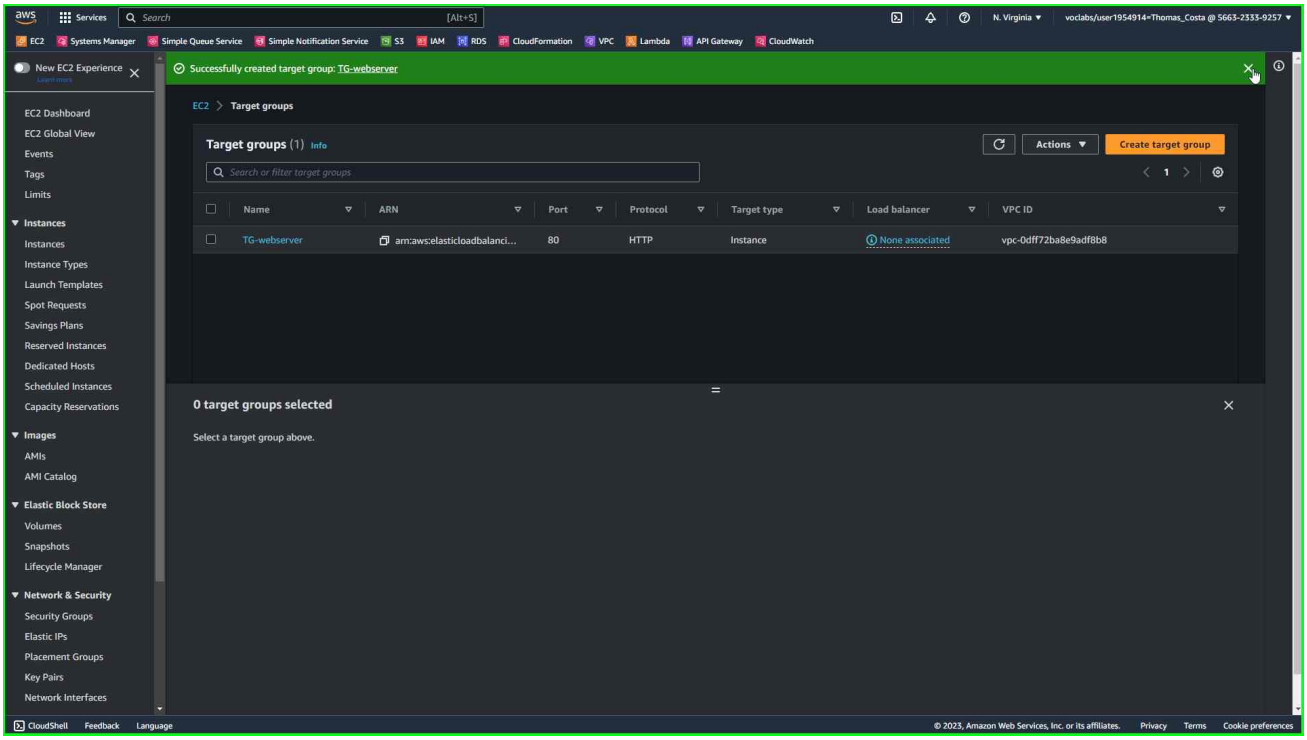
Selecionar as opções conforme imagem abaixo:



Nesta tela clique somente em **Create target group**:

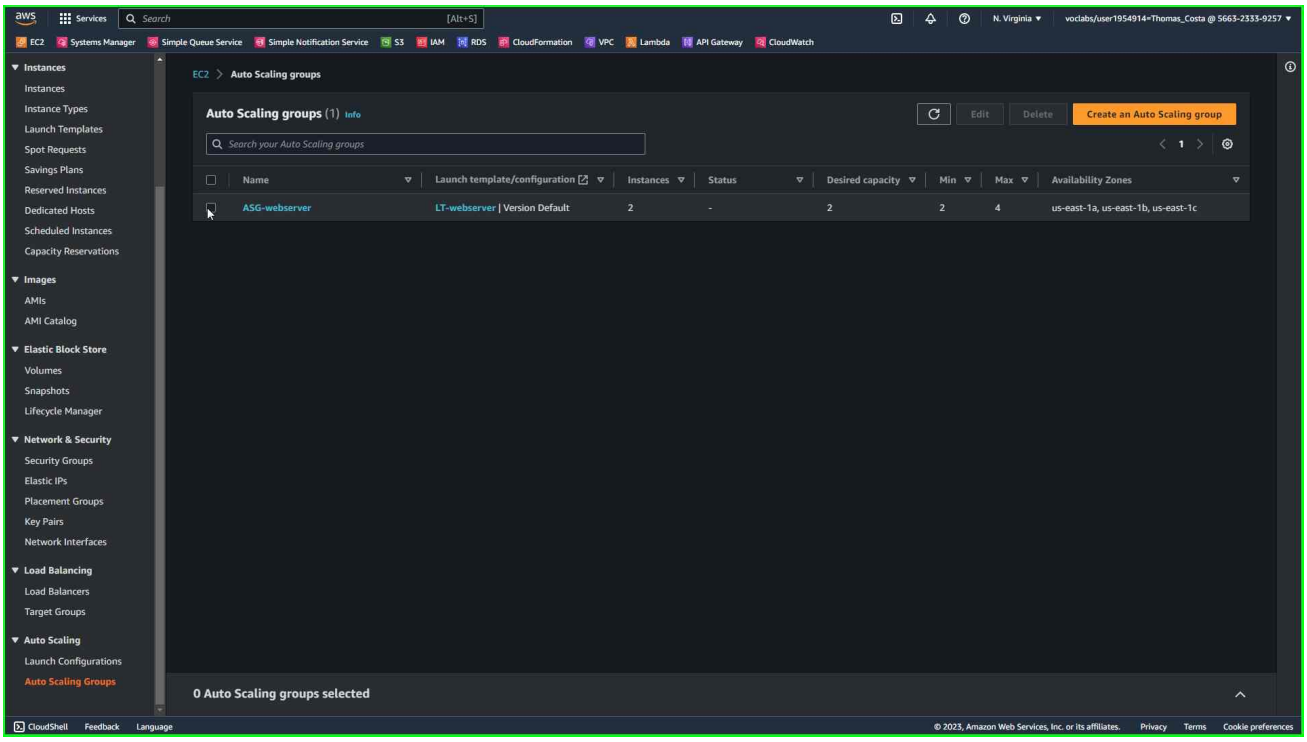


Target group foi criado com sucesso:



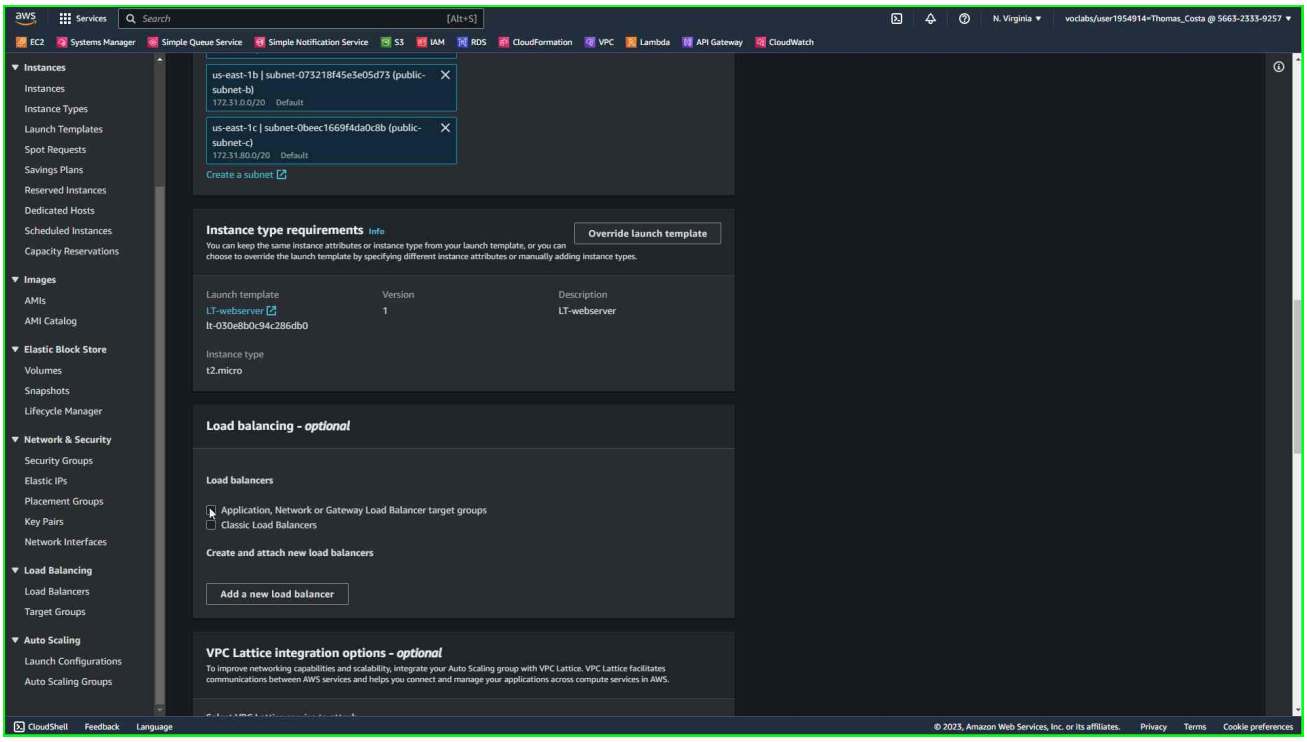
Implantando um Application Load Balancer (ALB) na AWS com EC2

Vamos voltar para a tela do **Auto Scaling groups** selecionar o **ASG-webserver**:

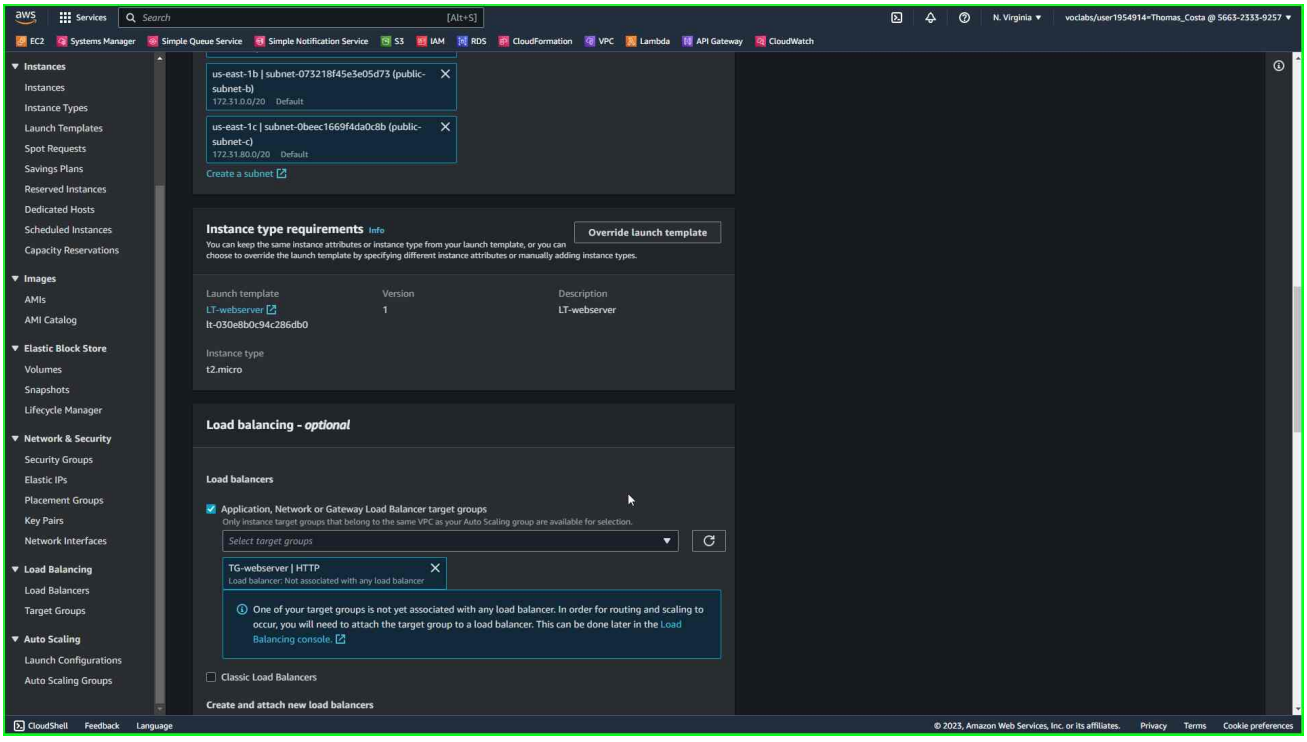


Implantando um Application Load Balancer (ALB) na AWS com EC2

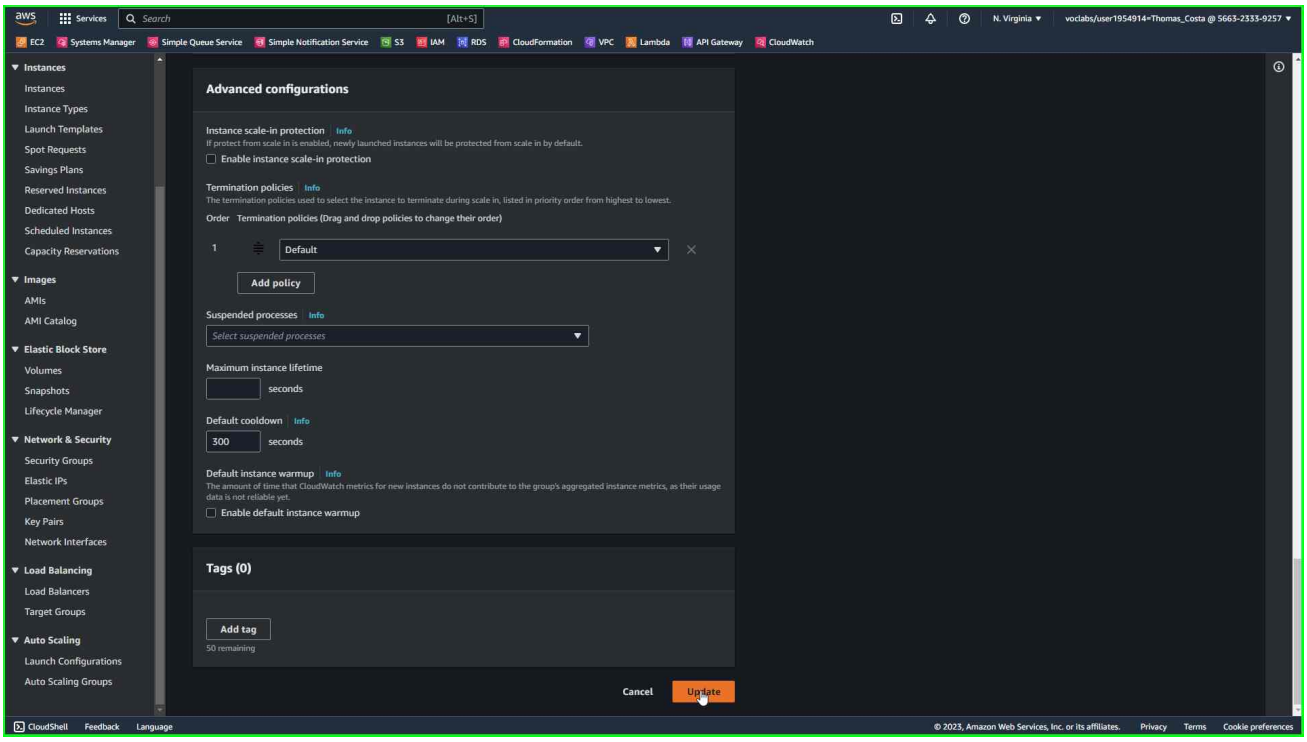
Selecionar a opção **Application, Network or Gateway Load Balancer target groups**:



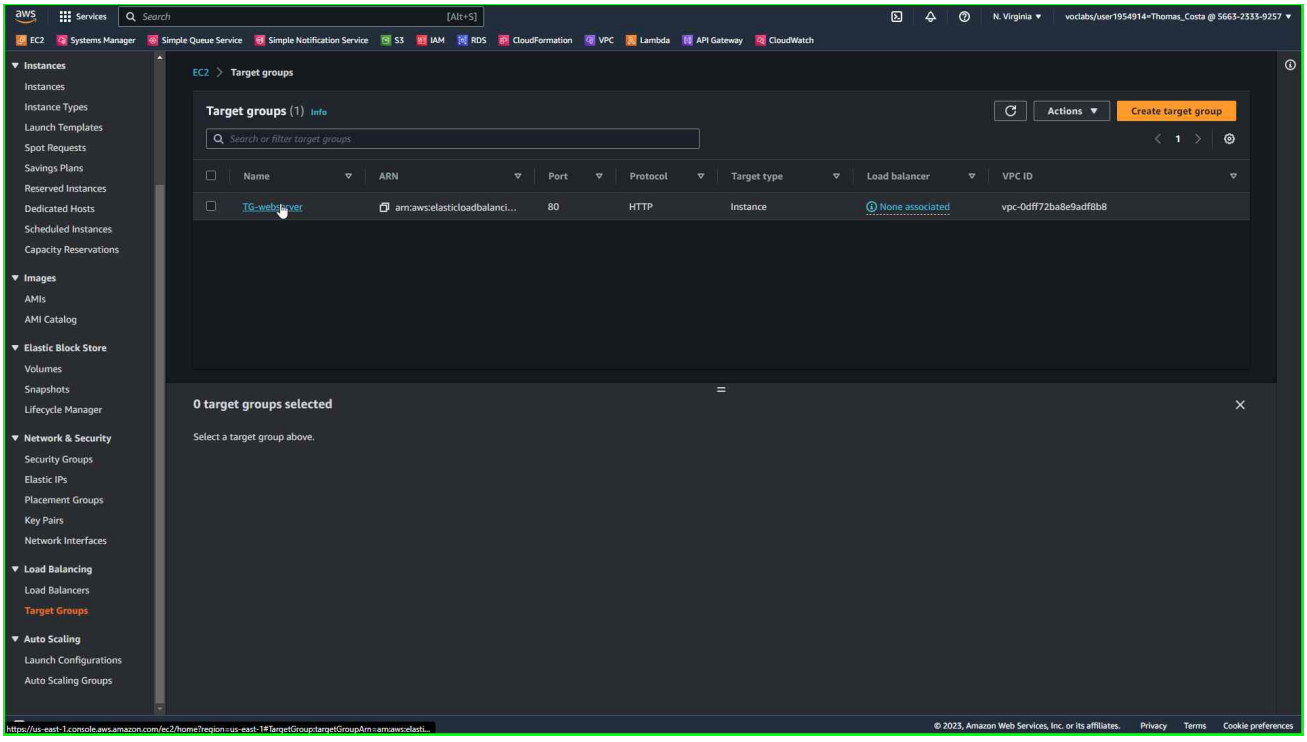
Selecionar o Target group **TG-webserver**:



Clique em **Update**:

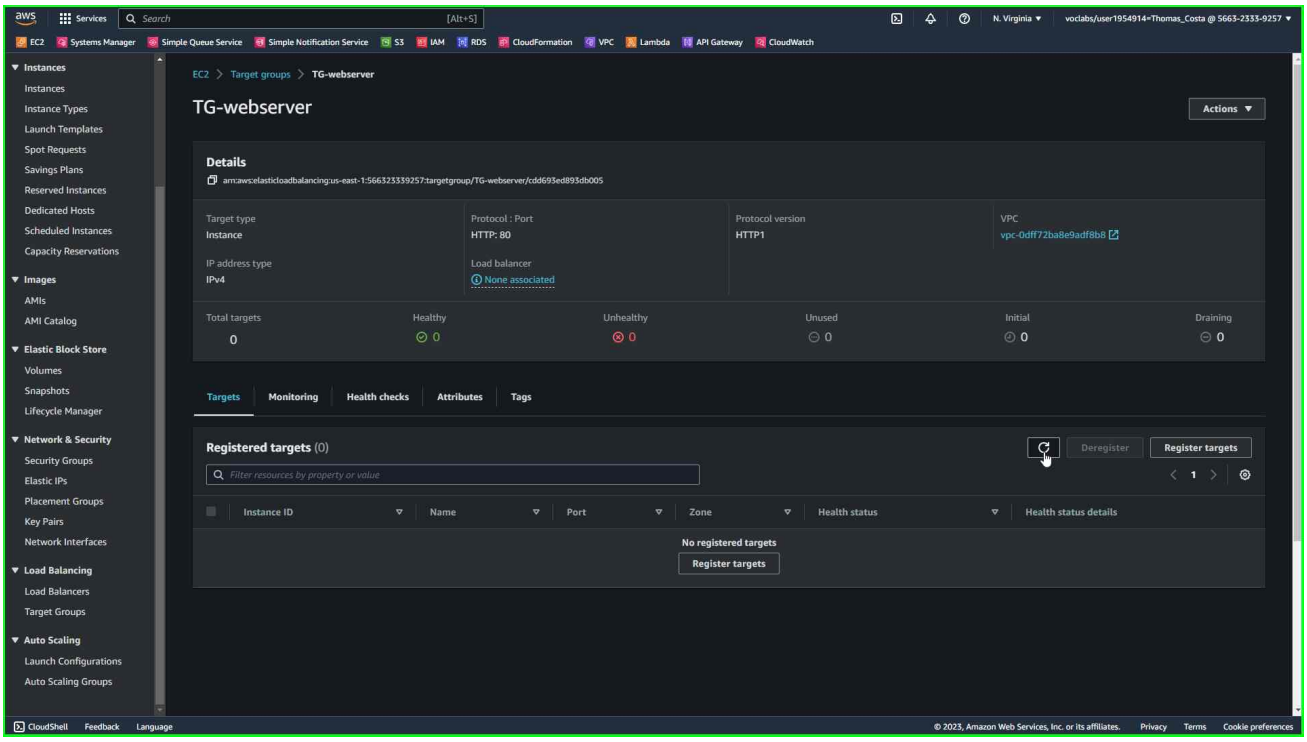


Volte a tela do **Target groups** e selecione o **TG-webserver**:

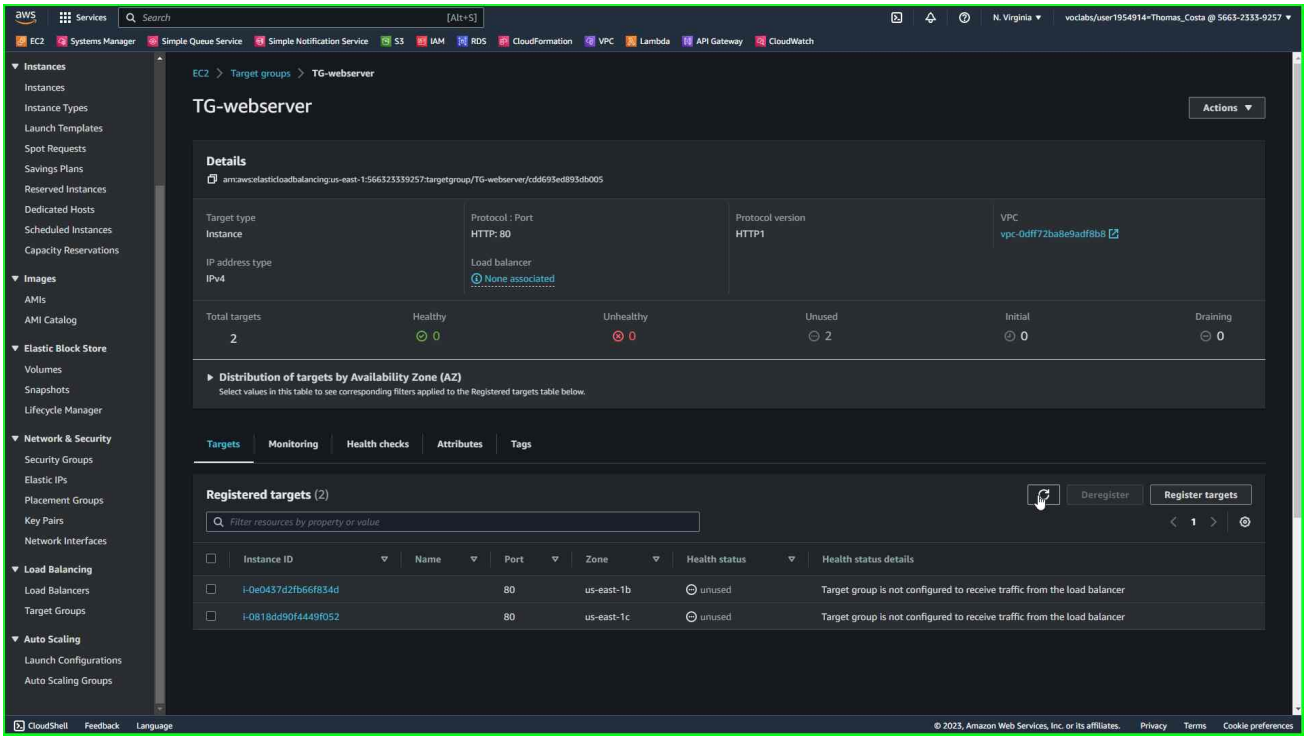


Implantando um Application Load Balancer (ALB) na AWS com EC2

Clique no botão mostrado na imagem abaixo:

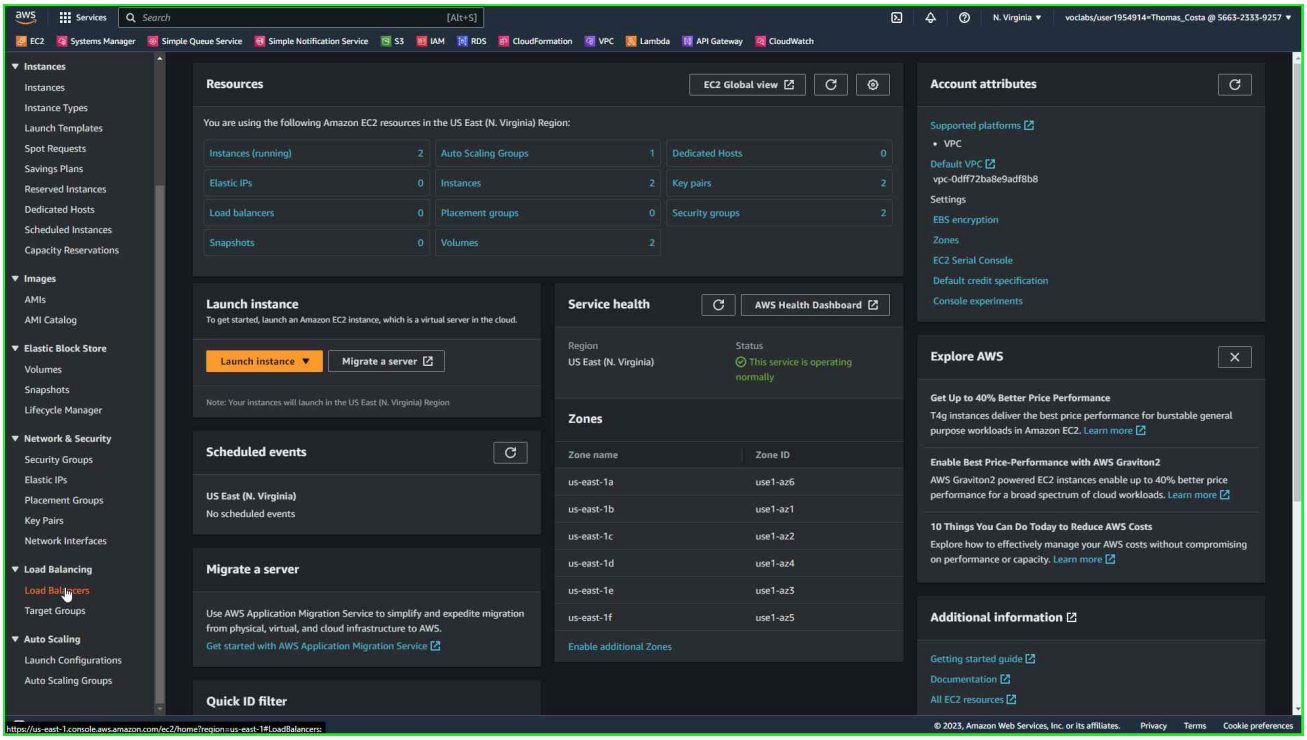


As instâncias EC2 estão atribuídas corretamente para o Target group mas com **Health Status** em **unused**:

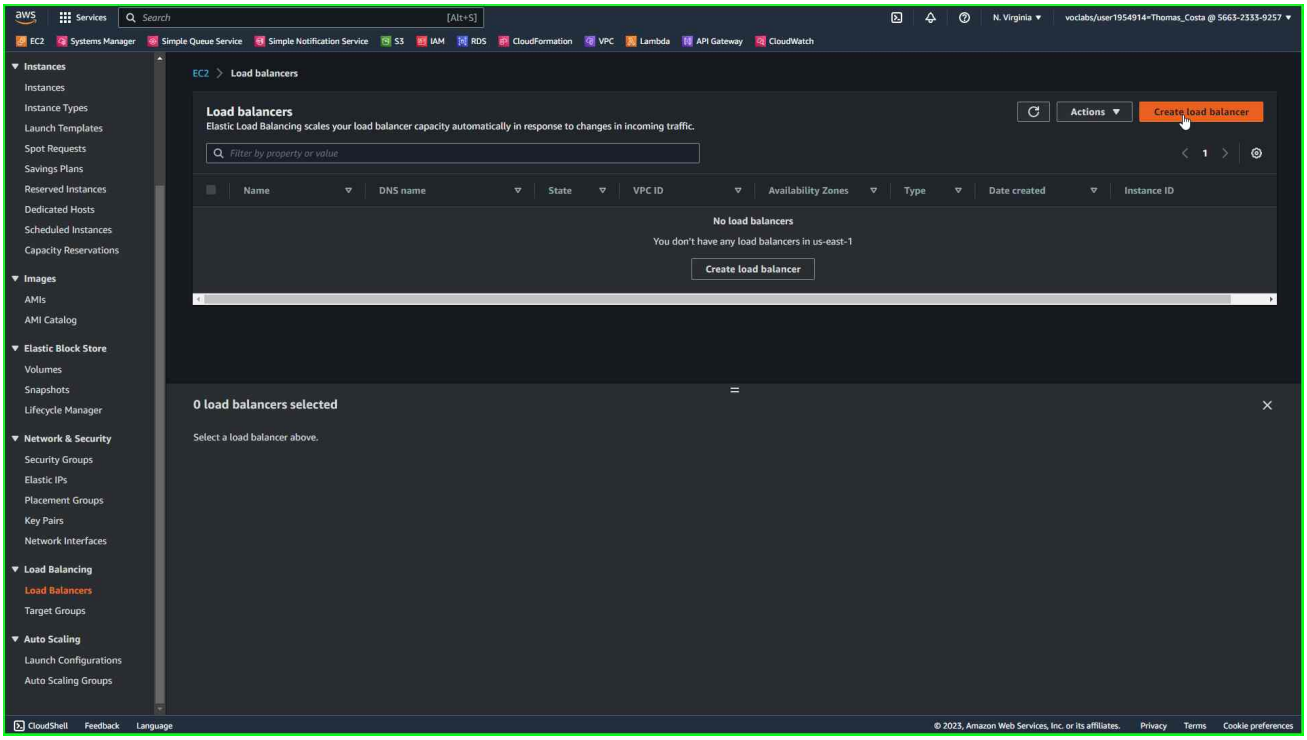


Parte 4 – Criando o Application Load Balancer (ALB)

Na tela principal do EC2, selecionar a opção **Load Balancers**:

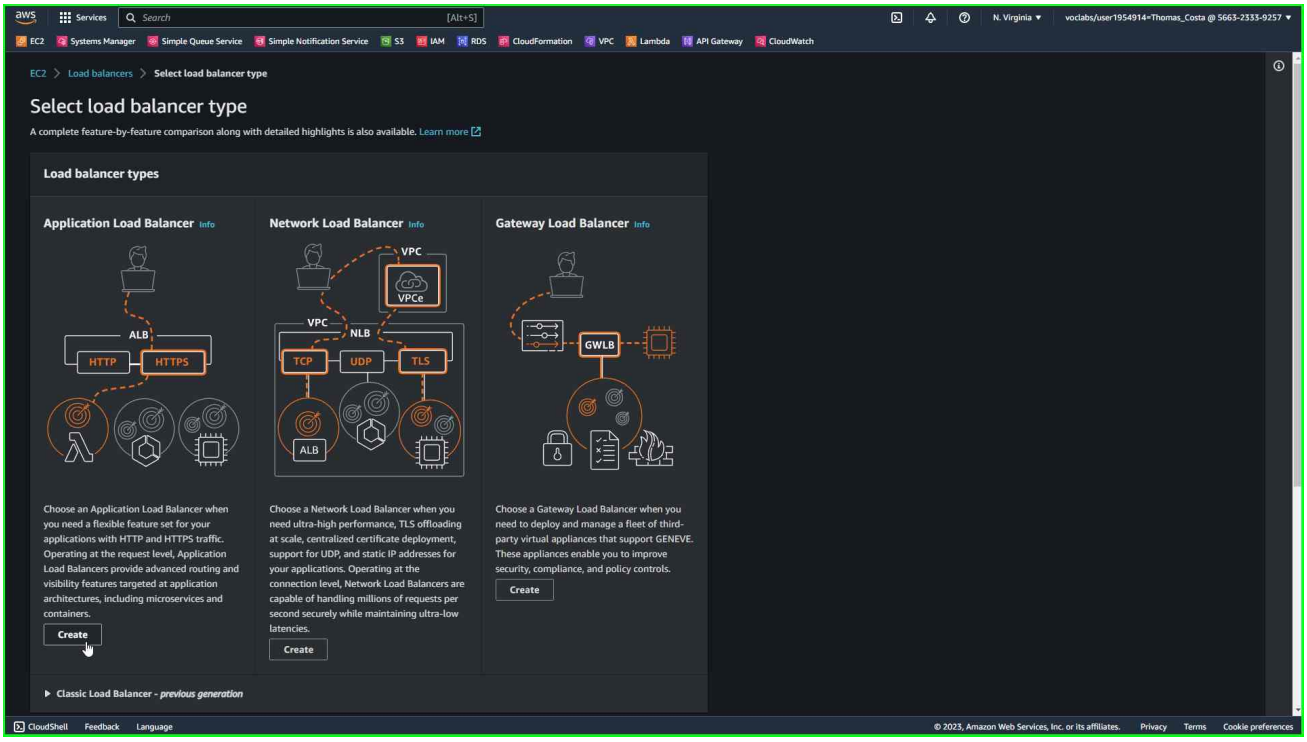


Clique no botão **Create load balancer**:

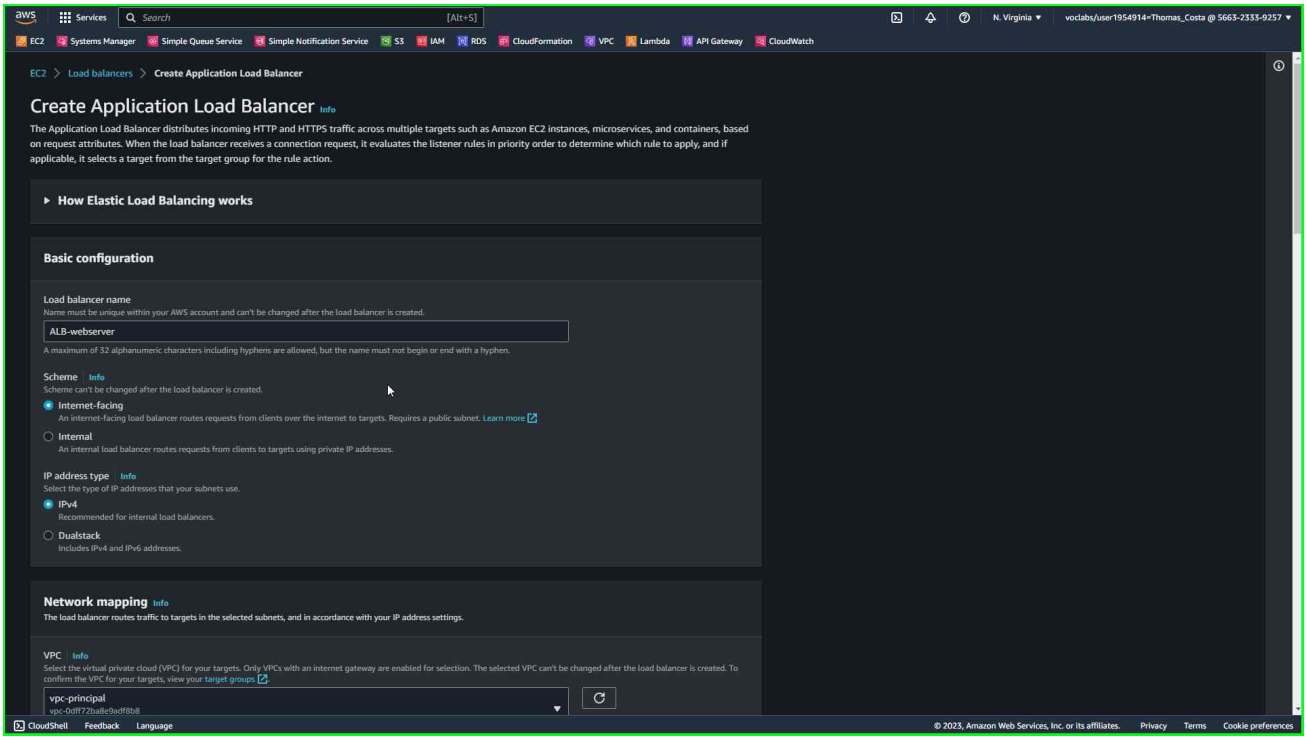


Implantando um Application Load Balancer (ALB) na AWS com EC2

Clique no botão **Create** do item **Application Load Balancer**:

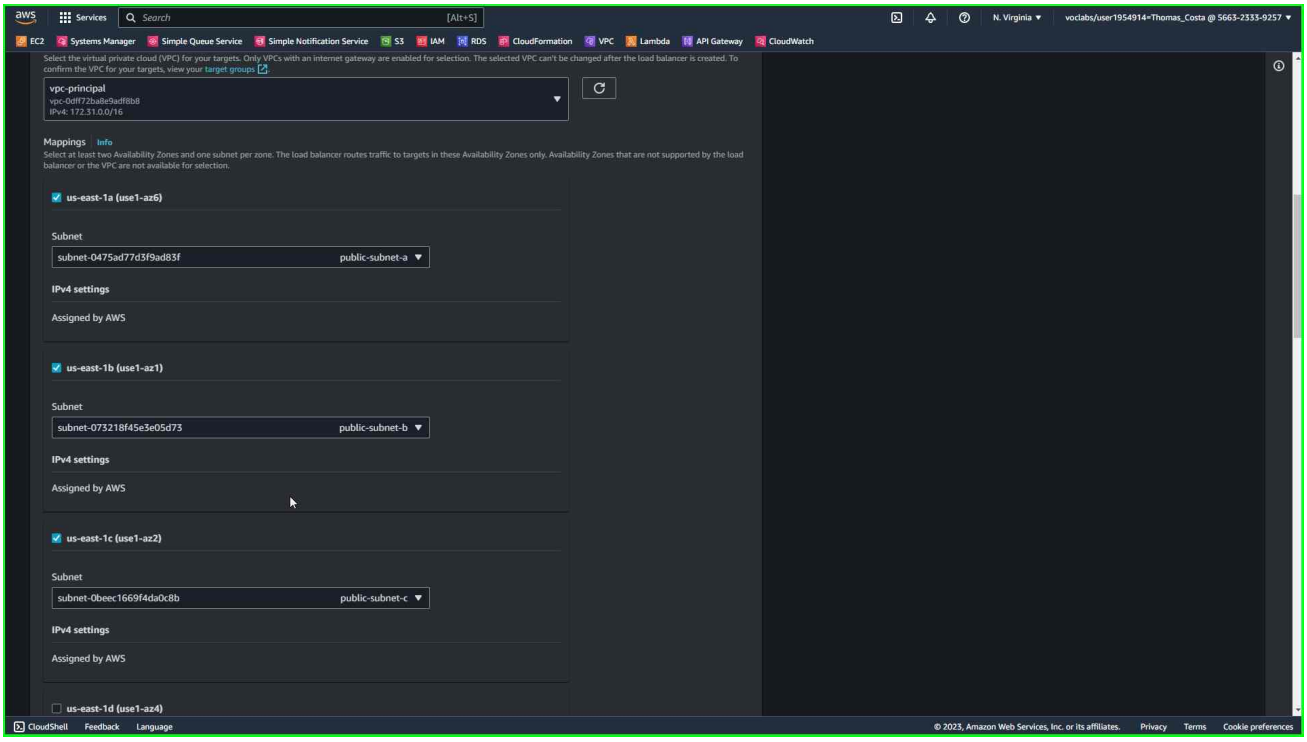


Coloque o nome do ALB de **ALB-webserver**:

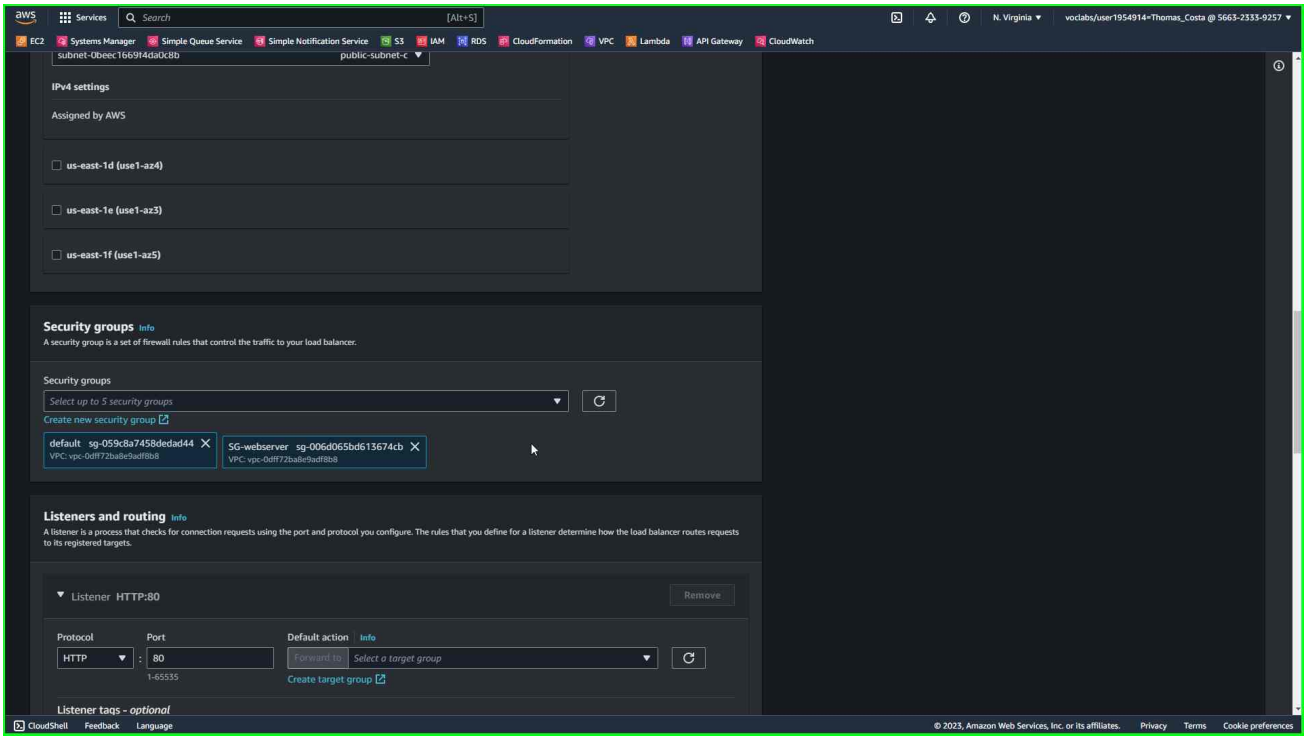


Implantando um Application Load Balancer (ALB) na AWS com EC2

Selecione 3 AZ conforme imagem abaixo:

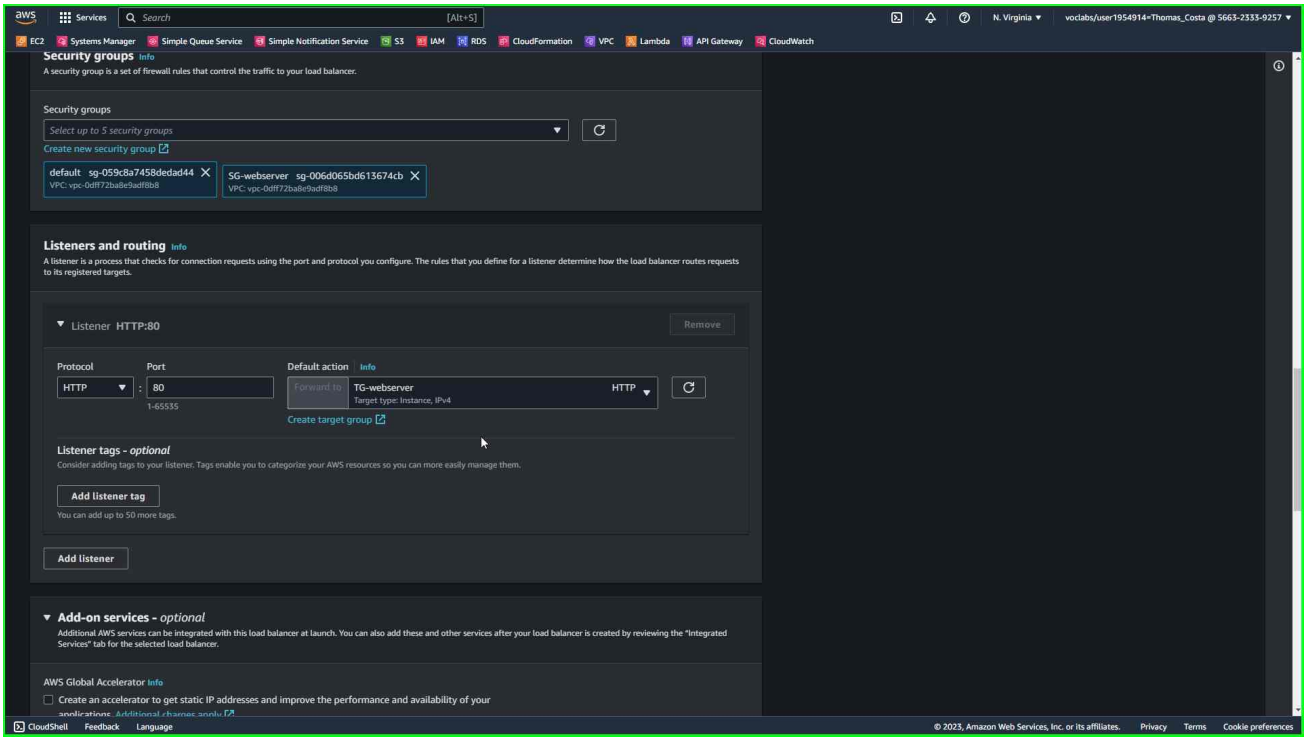


Selecione o Security Group **SG-webserver**:

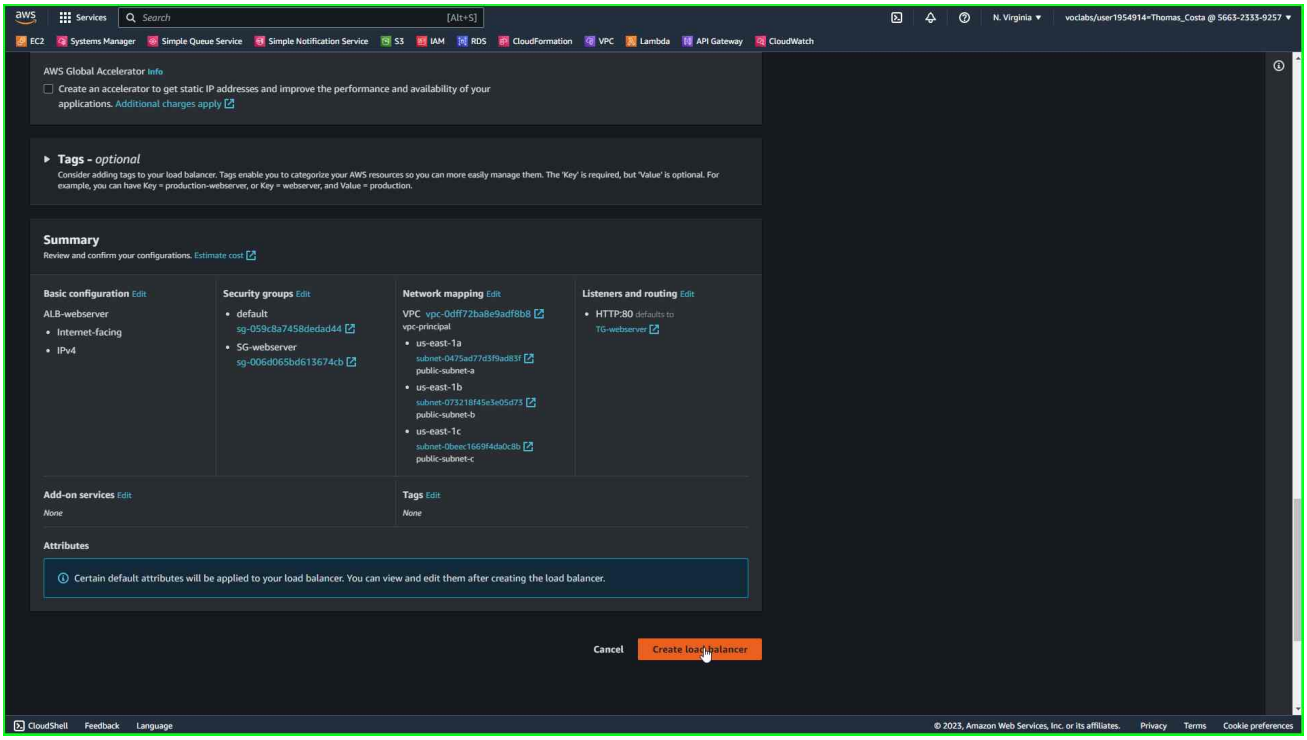


Implantando um Application Load Balancer (ALB) na AWS com EC2

Selecione o Target group **TG-webserver**:

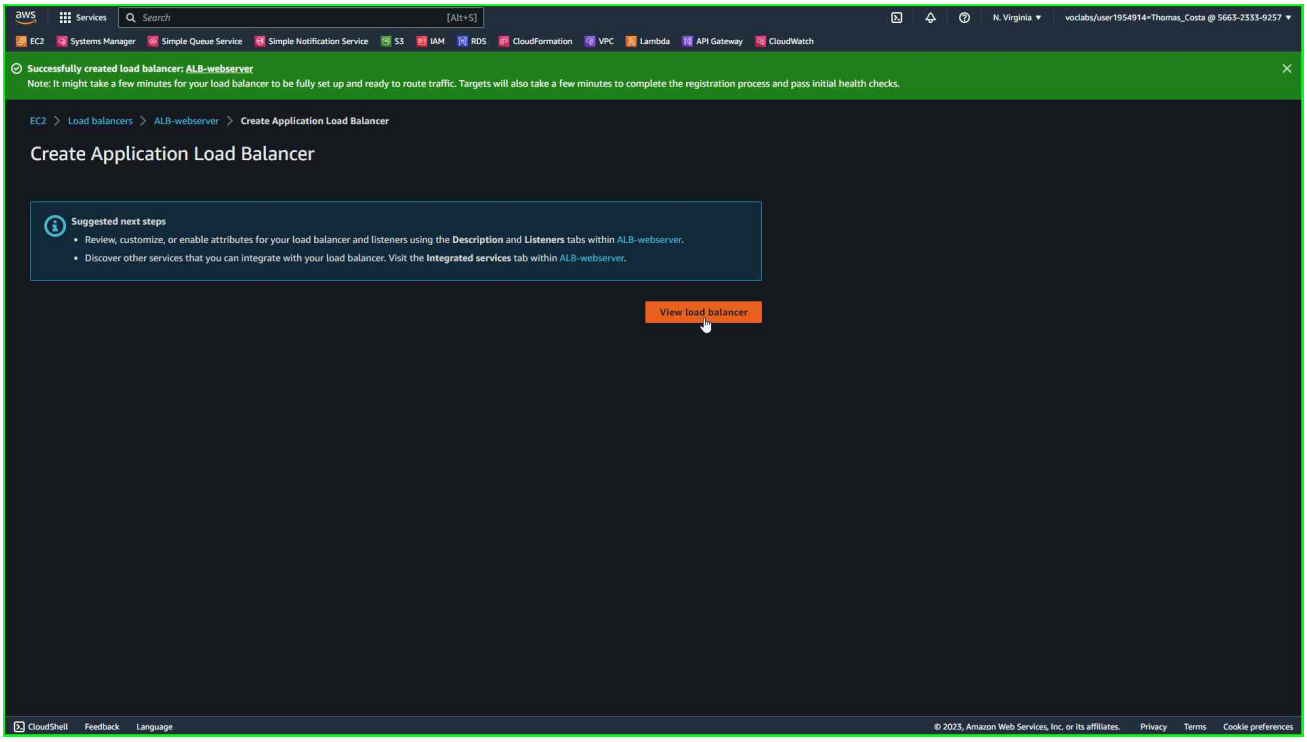


Clique no botão **Create load balancer**:

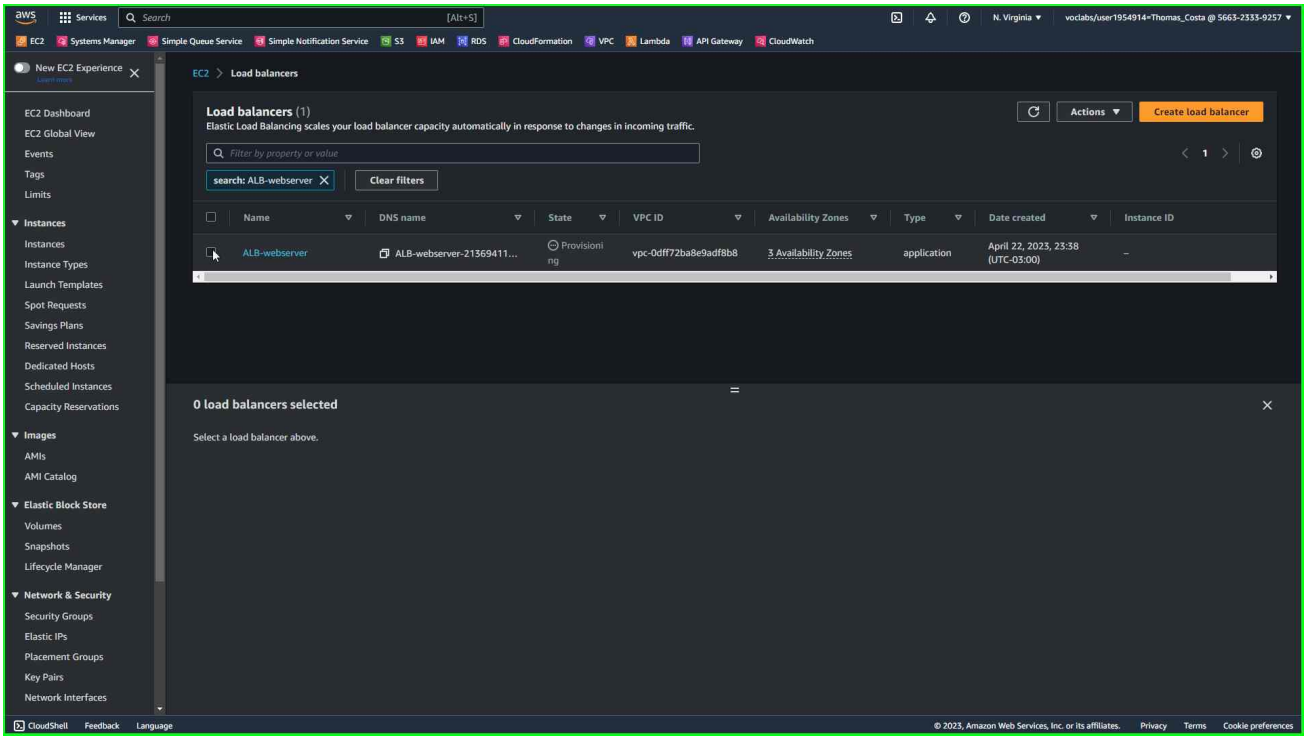


Implantando um Application Load Balancer (ALB) na AWS com EC2

ALB criado com sucesso:

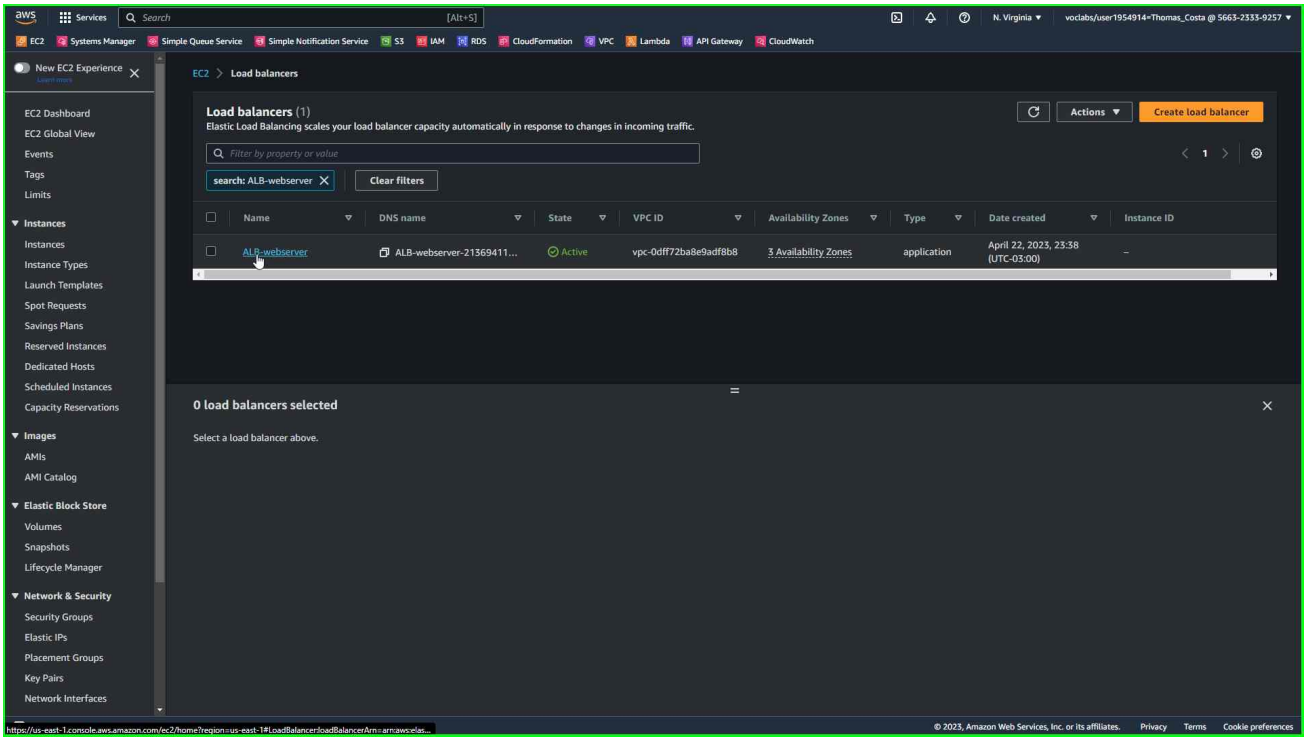


Aguardar o balanceador sair do status **Provisioning**:

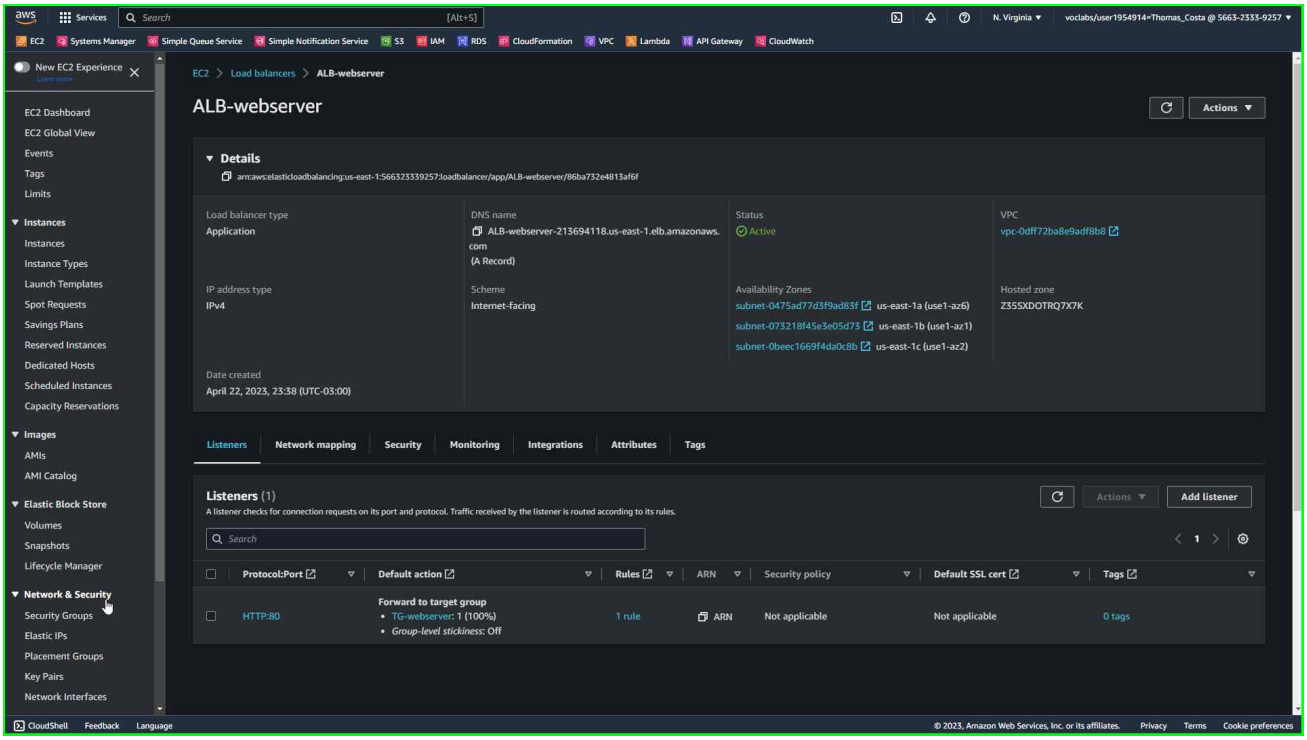


Implantando um Application Load Balancer (ALB) na AWS com EC2

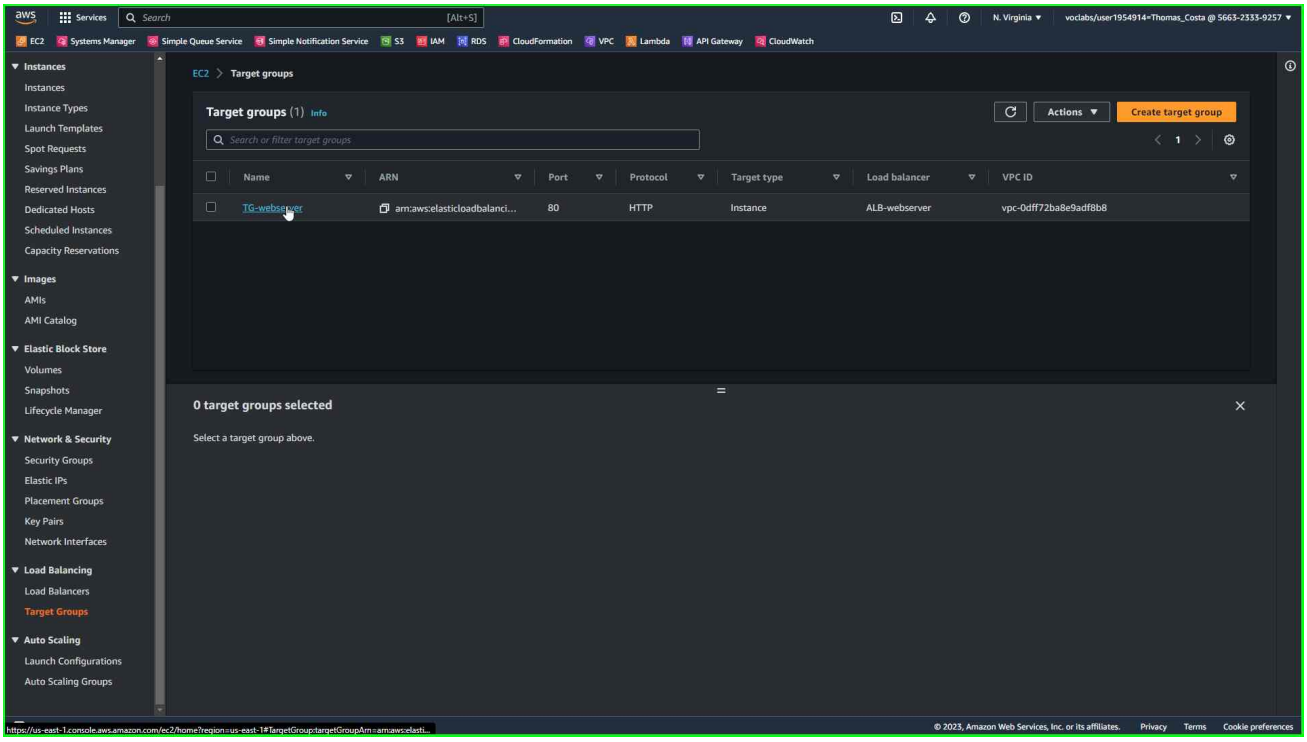
Status do ALB em **Active,** pronto para ser utilizado:



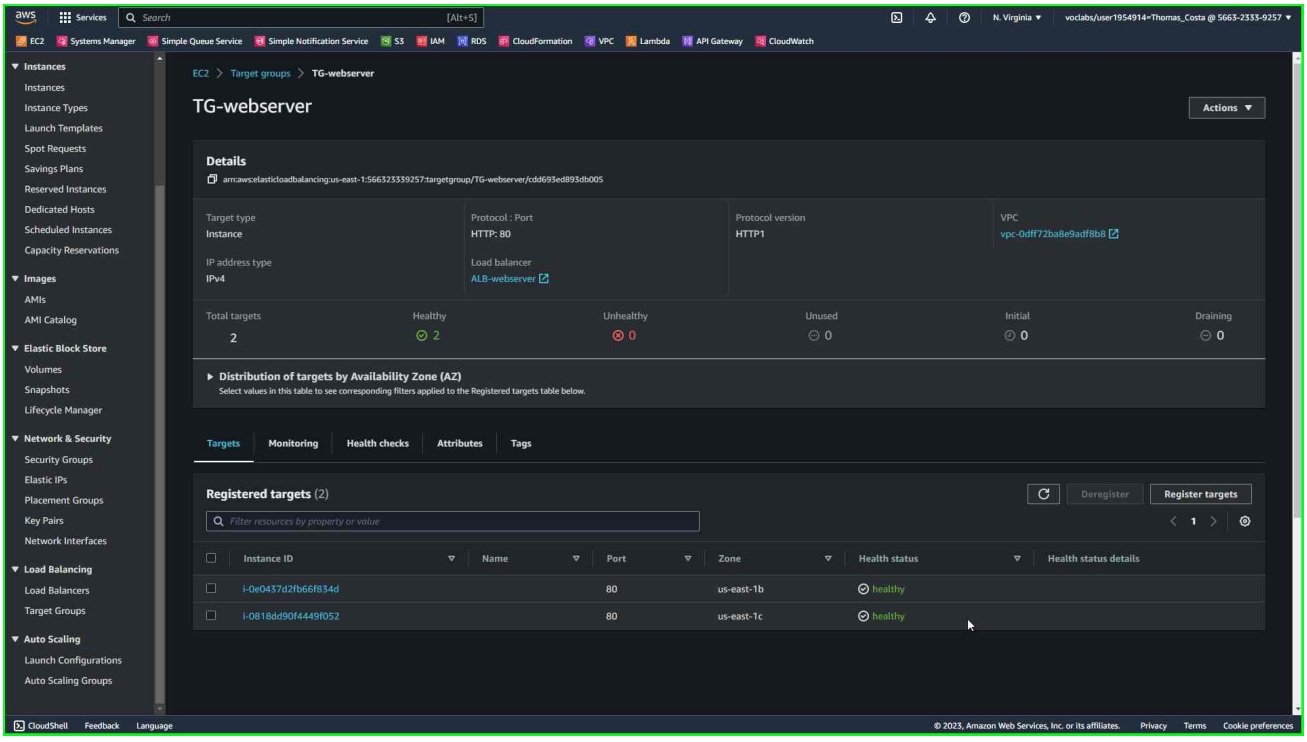
Selecione o ALB para ver as suas propriedades:



Voltar a tela com a lista de Target groups e selecionar o que criamos nos itens anteriores:

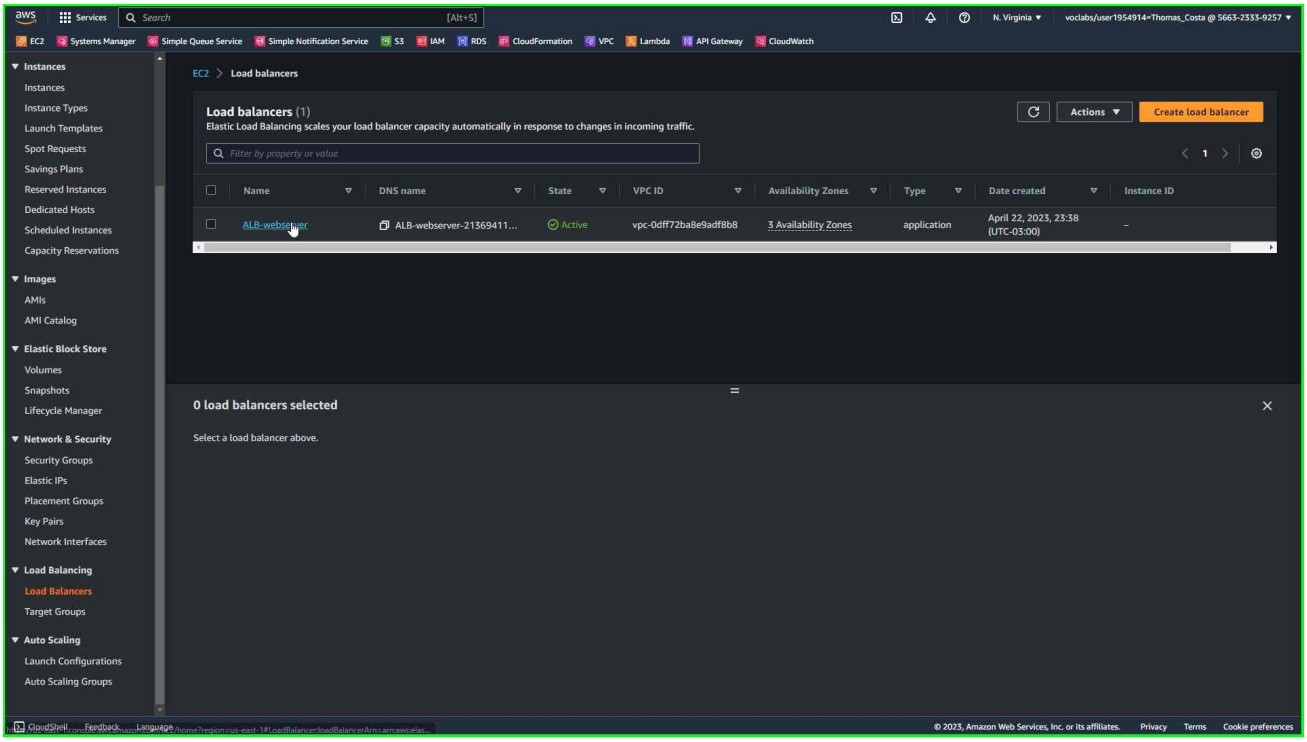


Balanceador reconhece as instâncias através do Target group conforme podemos ver o **Health Status** em **healthy**:

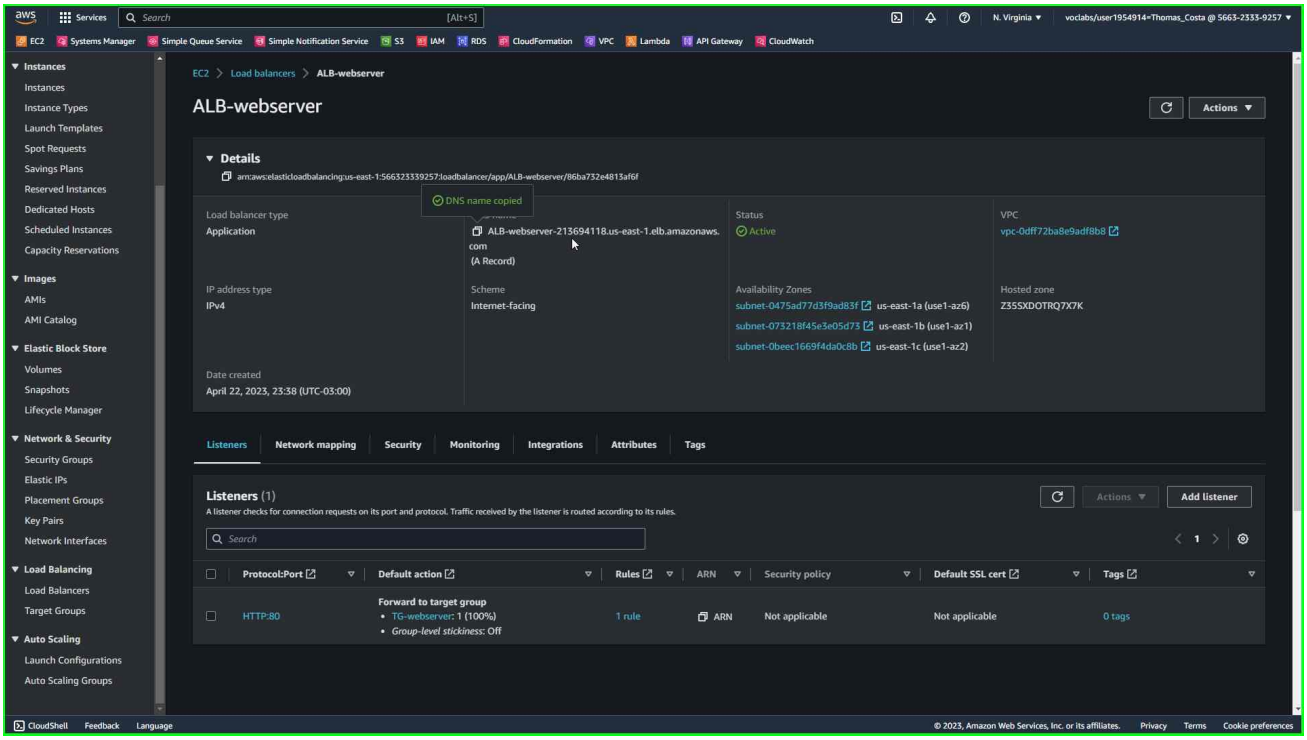


Implantando um Application Load Balancer (ALB) na AWS com EC2

Selecione novamente o balanceador:



Copie o endereço público do balanceador para testar no navegador:



Balanceador funcionando e acessando os EC2. Podemos verificar que a cada requisição a solicitação é enviada para uma instância:

Scripts do User Data

```
#!/bin/bash

sudo yum update -y

sudo yum install -y httpd

sudo systemctl start httpd

sudo systemctl enable httpd

EC2AZ=$(TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-

ttl-seconds: 21600"` && curl -H "X-aws-ec2-metadata-token: $TOKEN" -v

http://169.254.169.254/latest/meta-data/placement/availability-zone)

EC2IPPUBLIC=$(TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-

token-ttl-seconds: 21600"` && curl -H "X-aws-ec2-metadata-token: $TOKEN" -v

http://169.254.169.254/latest/meta-data/public-ipv4)

EC2IPPRIVATE=$(TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-

token-ttl-seconds: 21600"` && curl -H "X-aws-ec2-metadata-token: $TOKEN" -v

http://169.254.169.254/latest/meta-data/local-ipv4)

echo '<html><head><title>Minha Página EC2</title></head><body><h1>Zona de Disponibilidade e

IPs</h1><p><b>Zona de Disponibilidade:</b> AZID</p><p><b>IP Público:</b> IPPUBLIC</p><p><b>IP

Privado:</b> IPPRIVATE</p></body></html>' > /var/www/html/index.txt

sudo sed "s/AZID/$EC2AZ/" /var/www/html/index.txt > /var/www/html/index1.txt

sudo sed "s/IPPUBLIC/$EC2IPPUBLIC/" /var/www/html/index1.txt > /var/www/html/index2.txt

sudo sed "s/IPPRIVATE/$EC2IPPRIVATE/" /var/www/html/index2.txt > /var/www/html/index.html
```