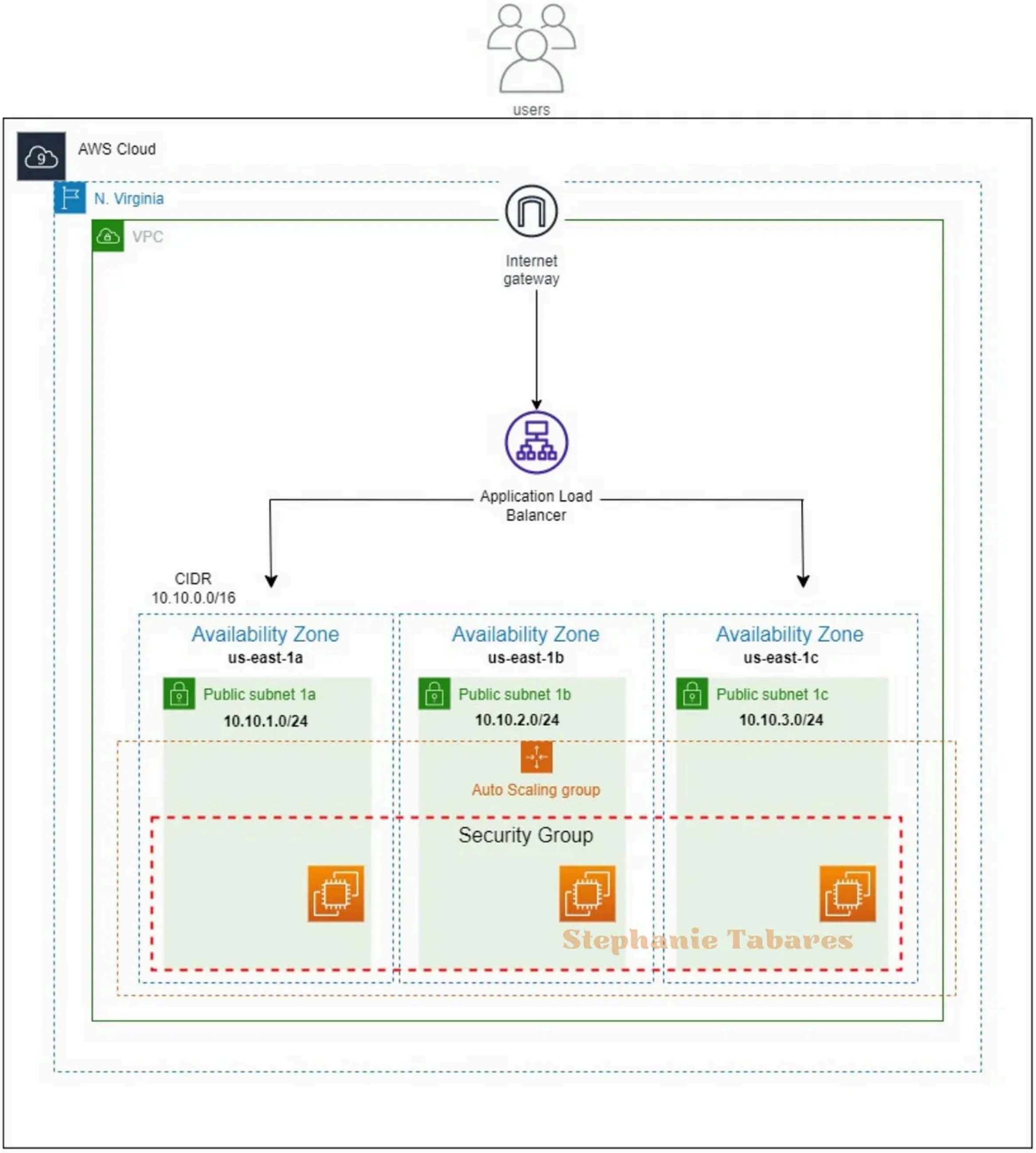
Creating a High-Availability Web Application Infrastructure on AWS with VPC, Auto Scaling, and Application Load Balancer

Stephanie Tabares · Follow

6 min read · Feb 21, 2023

5



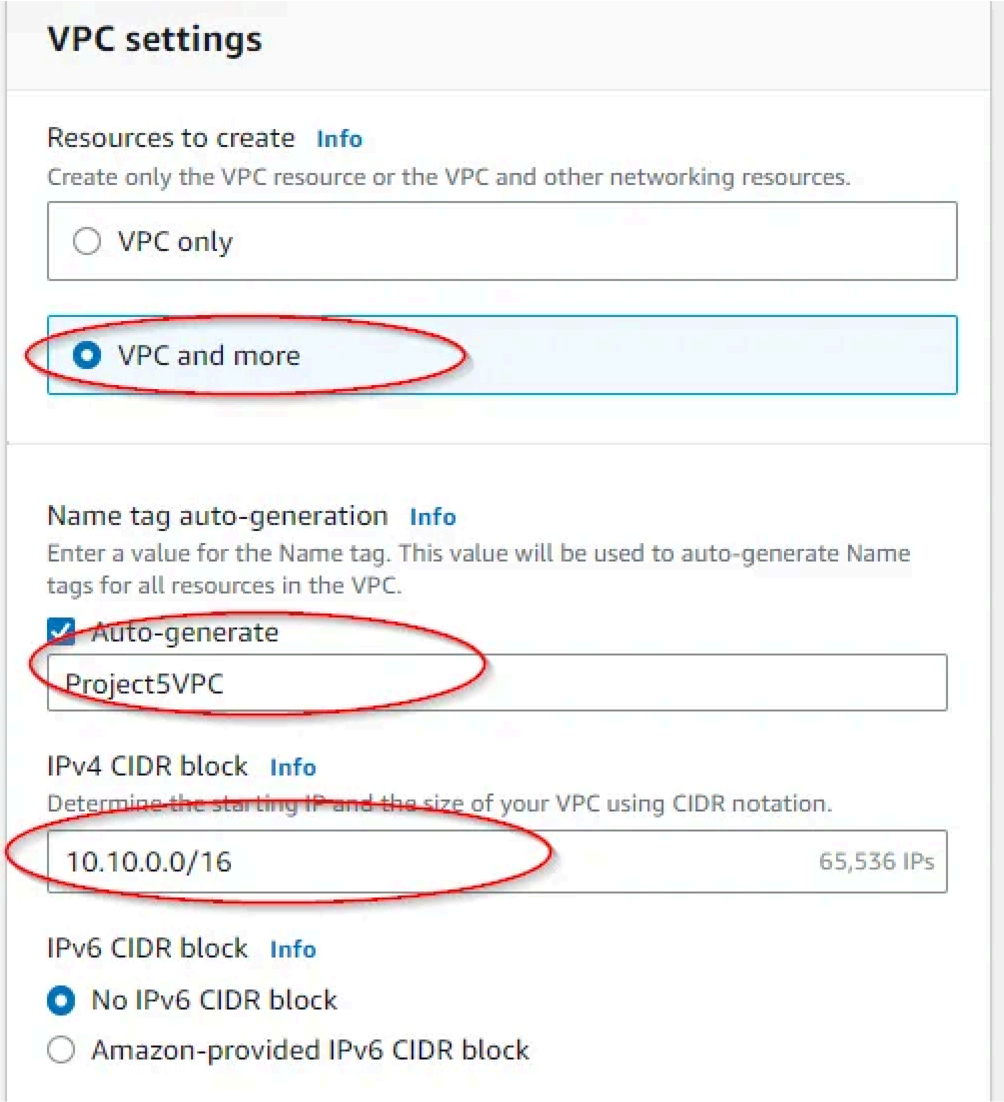
Welcome to project #5! In today's digital age, having a reliable and scalable web application infrastructure is crucial for any business. In this project, we will be creating a Virtual Private Cloud (VPC) on Amazon Web Services (AWS) with three public subnets and an autoscaling group to ensure high availability and scalability of our web application. We'll also be configuring an Application Load Balancer to distribute traffic to the autoscaling group, and implementing security groups to ensure the safety and security of our infrastructure. So, let's get started and build an infrastructure that can handle any amount of traffic with ease!

FOUNDATIONAL

1. For this project you MUST create a diagram of your AWS architecture and use it as your Preview Image for your Medium.
2. Create a VPC with cidr 10.10.0.0/16
3. Create three public subnets with 10.10.1.0/24 & 10.10.2.0/24 & 10.10.3.0/24
4. Create an autoscaling group using t2.micro instances. All instances should have apache installed on each instance with the ability to check any random IP address and be able to produce a test page. Ensure the autoscaling group is using the public subnets from #2.
5. The autoscaling min and max should be 2 and 5.
6. Create an Application Load Balancer to distribute traffic to the autoscaling group.
7. Create web server security group that allows inbound traffic from HTTP from your Application Load Balancer.
8. Create a load balancer security group that allows inbound traffic from HTTP from 0.0.0.0/0.

Creating a VPC

1. In the search bar type VPC -> Create VPC -> VPC and more -> Name your VPC -> Add CIDR block 10.10.0.0/16



VPC settings

Resources to create Info

Create only the VPC resource or the VPC and other networking resources.

○ VPC only

● VPC and more

Name tag auto-generation Info

Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

☑ Auto-generate

Project5VPC

IPv4 CIDR block Info

Determine the starting IP and the size of your VPC using CIDR notation.

10.10.0.0/16 65,536 IPs

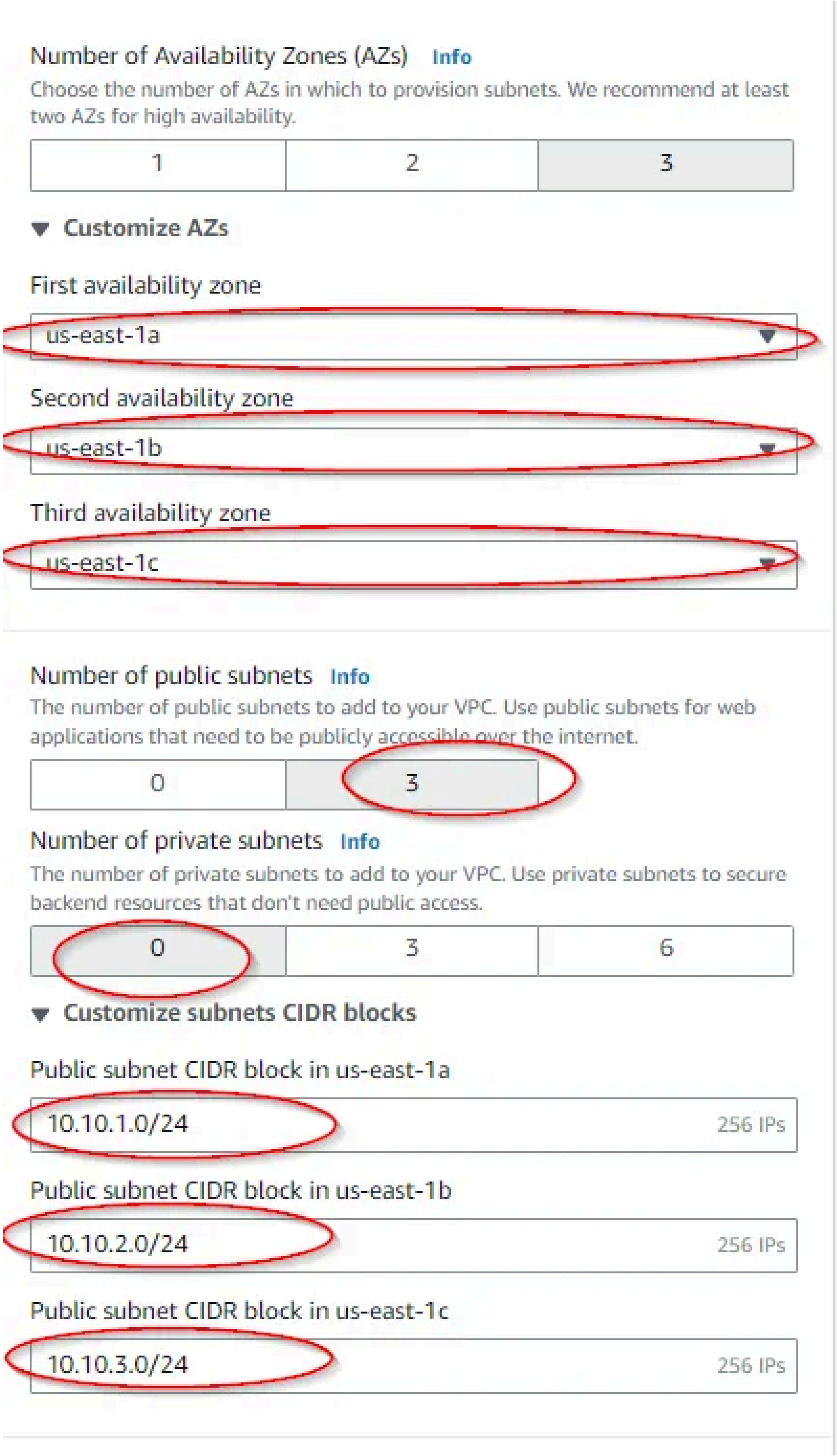
IPv6 CIDR block Info

● No IPv6 CIDR block

○ Amazon-provided IPv6 CIDR block

2. Deploying subnets in different availability zones provides high availability and fault tolerance for your applications. This is because if one availability zone becomes unavailable, the other two can continue to handle traffic and requests, preventing any disruption to your application or service.

-Create three public subnets with 10.10.1.0/24 & 10.10.2.0/24 & 10.10.3.0/24



Number of Availability Zones (AZs) Info

Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

1 2 [✓ 3]

▼ Customize AZs

First availability zone

us-east-1a ▼

Second availability zone

us-east-1b ▼

Third availability zone

us-east-1c ▼

Number of public subnets Info

The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.

0 [✓ 3]

Number of private subnets Info

The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.

[✓ 0] 3 6

▼ Customize subnets CIDR blocks

Public subnet CIDR block in us-east-1a

10.10.1.0/24 256 IPs

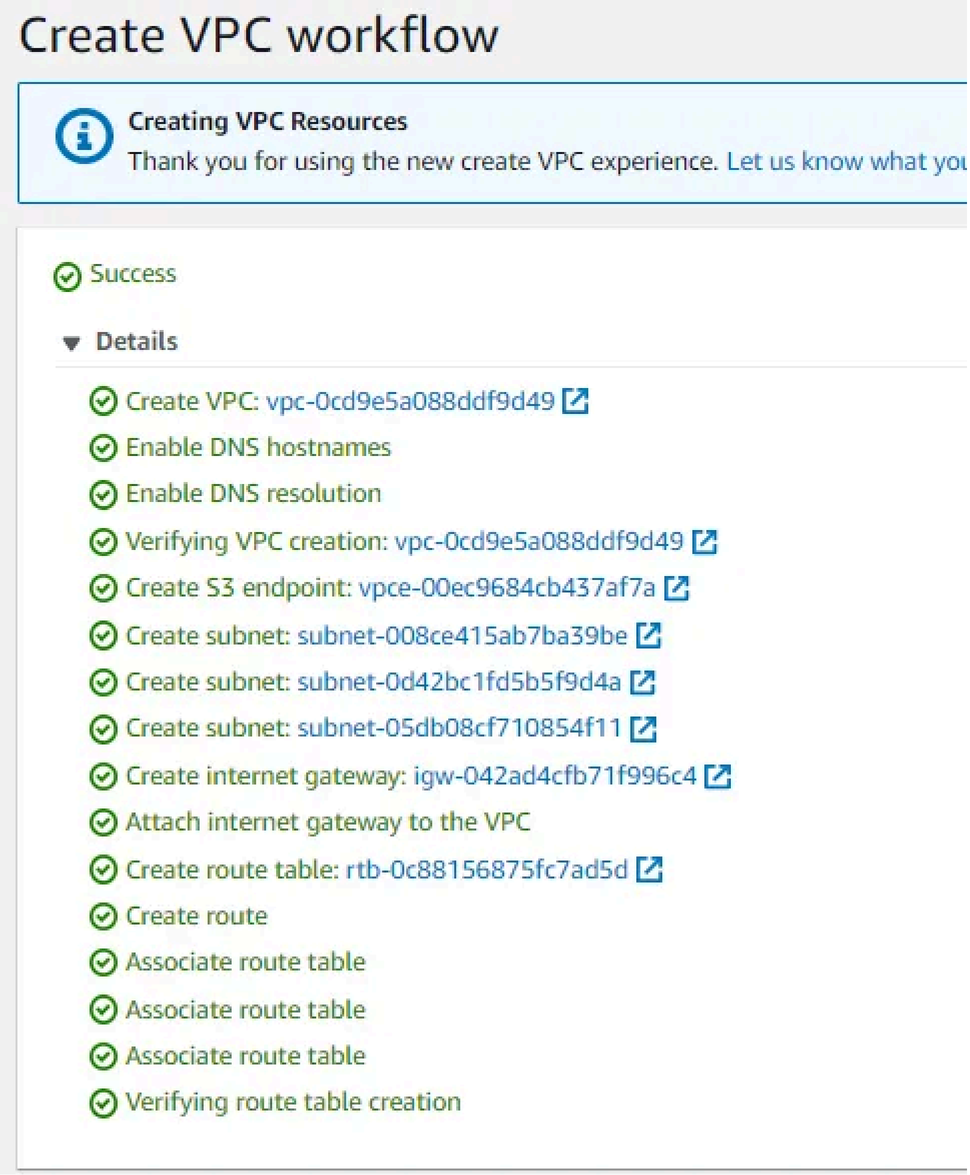
Public subnet CIDR block in us-east-1b

10.10.2.0/24 256 IPs

Public subnet CIDR block in us-east-1c

10.10.3.0/24 256 IPs

3. Create VPC



Create VPC workflow

Creating VPC Resources
Thank you for using the new create VPC experience. Let us know what you

✓ Success

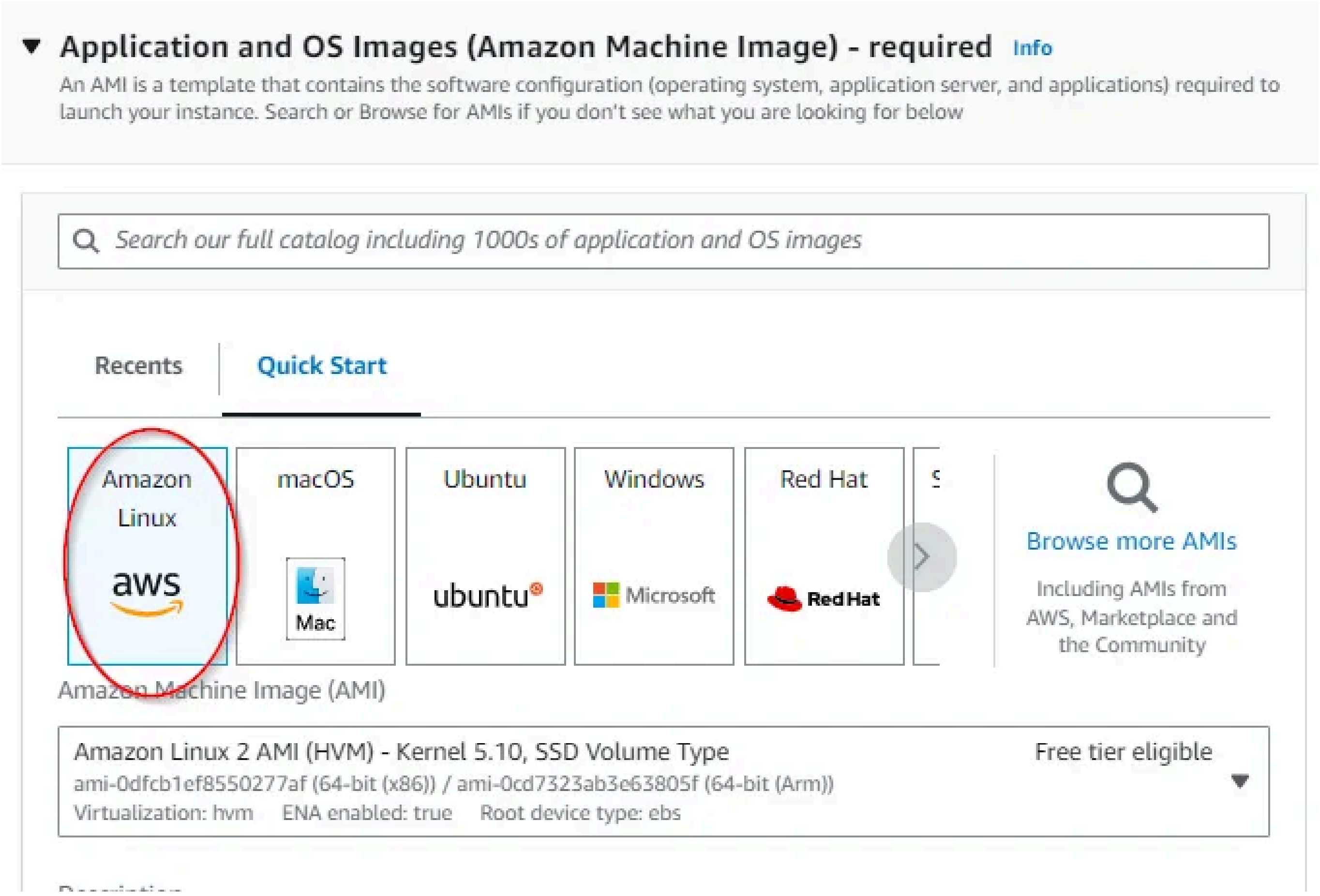
▼ Details

✓ Create VPC: vpc-0cd9e5a088ddf9d49
✓ Enable DNS hostnames
✓ Enable DNS resolution
✓ Verifying VPC creation: vpc-0cd9e5a088ddf9d49
✓ Create S3 endpoint: vpce-00ec9684cb437af7a
✓ Create subnet: subnet-008ce415ab7ba39be
✓ Create subnet: subnet-0d42bc1fd5b5f9d4a
✓ Create subnet: subnet-05db08cf710854f11
✓ Create internet gateway: igw-042ad4cfb71f996c4
✓ Attach internet gateway to the VPC
✓ Create route table: rtb-0c88156875fc7ad5d
✓ Create route
✓ Associate route table
✓ Associate route table
✓ Associate route table
✓ Verifying route table creation

Congrats! Now let’s move on to the next step!

Create a launch template

1. In the search bar type EC2 -> Scroll down to Instances -> Launch templates -> Select Create launch template
2. Name launch template. For AMI select Amazon Linux. For instance type select t2.micro. Select your keypair name.



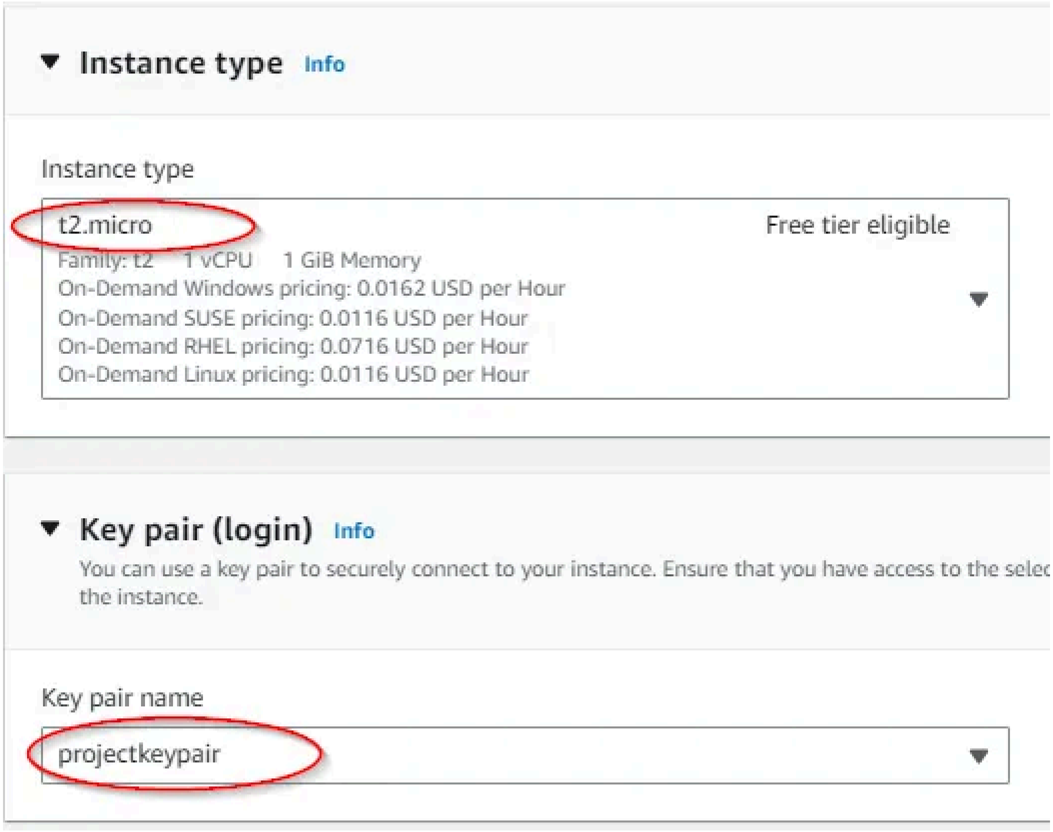
▼ Application and OS Images (Amazon Machine Image) - required Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images
Recents
Quick Start
Amazon Linux
aws
macOS
Mac
Ubuntu
ubuntu
Windows
Microsoft
Red Hat
Red Hat
S
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type — Free tier eligible
ami-0dfcb1ef8550277af (64-bit (x86)) / ami-0cd7323ab3e63805f (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs



▼ Instance type Info

Instance type

t2.micro — Free tier eligible
Family: t2 1 vCPU 1 GiB Memory
On-Demand Windows pricing: 0.0162 USD per Hour
On-Demand SUSE pricing: 0.0116 USD per Hour
On-Demand RHEL pricing: 0.0716 USD per Hour
On-Demand Linux pricing: 0.0116 USD per Hour

▼ Key pair (login) Info

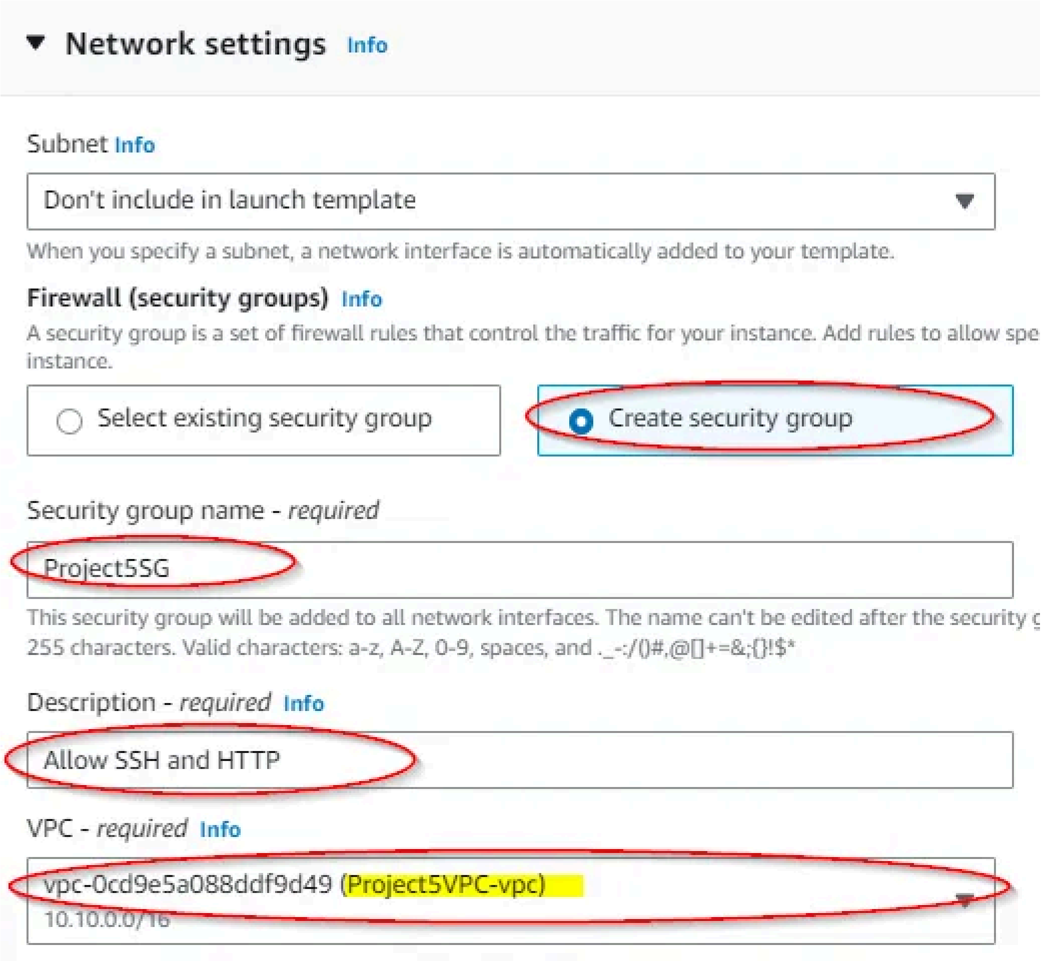
You can use a key pair to securely connect to your instance. Ensure that you have access to the selec the instance.

Key pair name

projectkeypair

3. Under Network settings select Create security group -> Name your security group -> Allow SSH and HTTP -> Select your VPC (it’s

automatically in default so you have to change it)



▼ Network settings Info

Subnet Info

Don't include in launch template ▼

When you specify a subnet, a network interface is automatically added to your template.

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow spe
instance.

○ Select existing security group ● Create security group

Security group name - *required*

Project5SG

This security group will be added to all network interfaces. The name can't be edited after the security g
255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=&;{}!$*

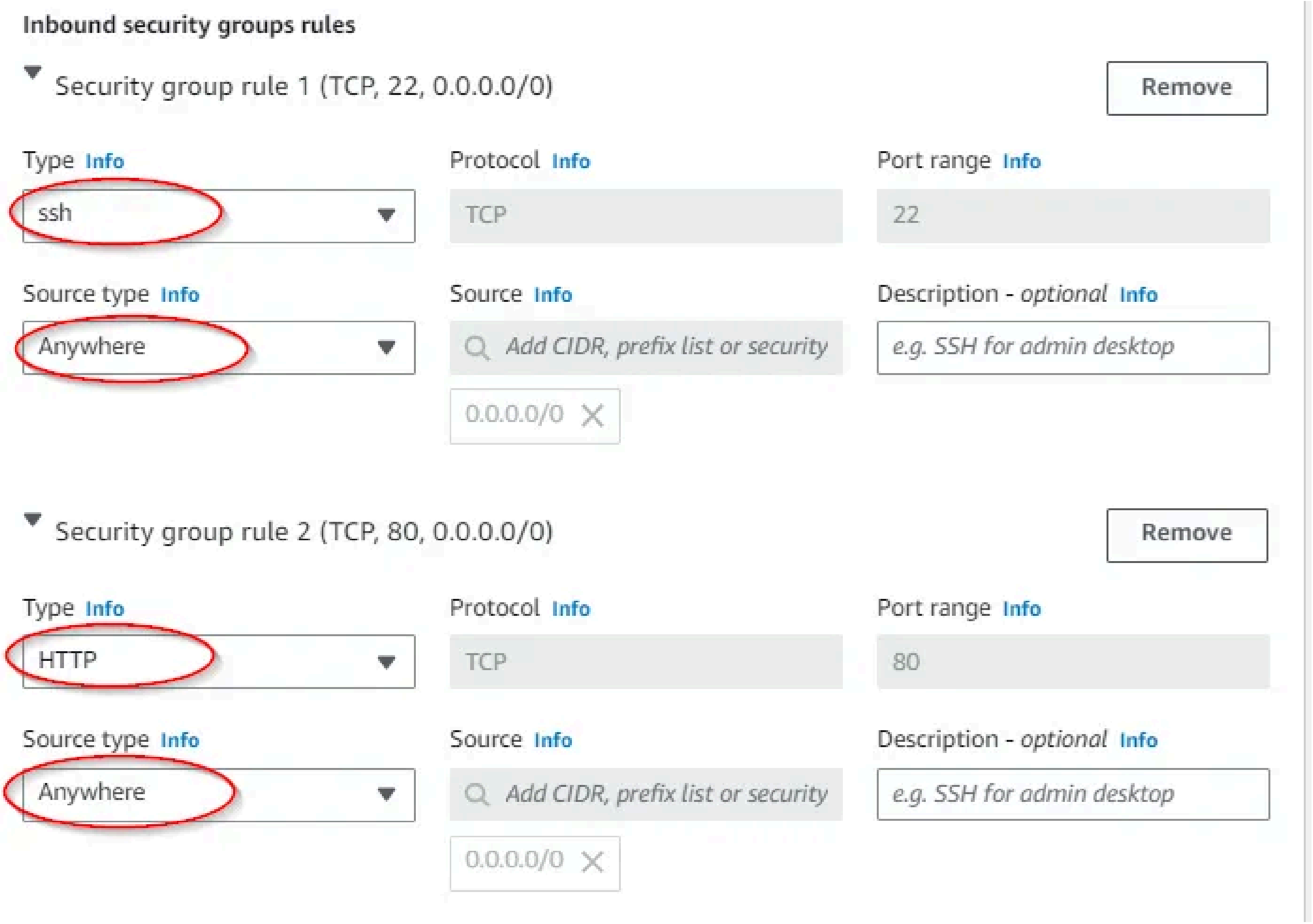
Description - *required* Info

Allow SSH and HTTP

VPC - *required* Info

vpc-0cd9e5a088ddf9d49 (Project5VPC-vpc)
10.10.0.0/16

4. For inbound security rules add SSH and HTTP



Inbound security groups rules

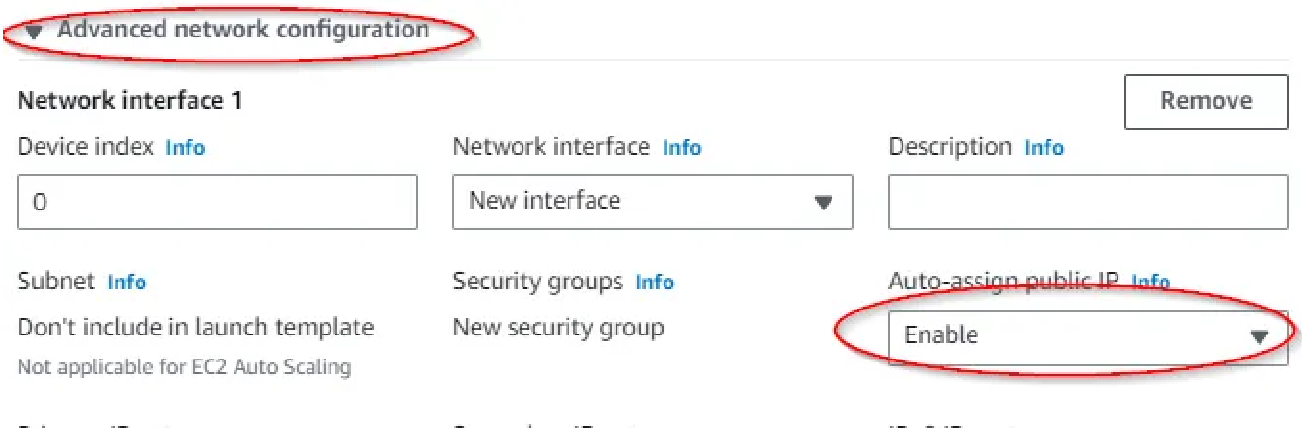
Security group rule 1 (TCP, 22, 0.0.0.0/0)

Type: ssh
Protocol: TCP
Port range: 22
Source type: Anywhere
Source: Add CIDR, prefix list or security
0.0.0.0/0
Description - optional: e.g. SSH for admin desktop

Security group rule 2 (TCP, 80, 0.0.0.0/0)

Type: HTTP
Protocol: TCP
Port range: 80
Source type: Anywhere
Source: Add CIDR, prefix list or security
0.0.0.0/0
Description - optional: e.g. SSH for admin desktop

5. For Advanced network configuration select Enable for Auto-assign public IP. A public IP address is necessary for instances in a public subnet to communicate with the internet, receive incoming traffic, and respond to requests from external clients or users.



Advanced network configuration

Network interface 1

Device index: 0
Network interface: New interface
Description:
Subnet: Don't include in launch template
Not applicable for EC2 Auto Scaling
Security groups: New security group
Auto-assign public IP: Enable

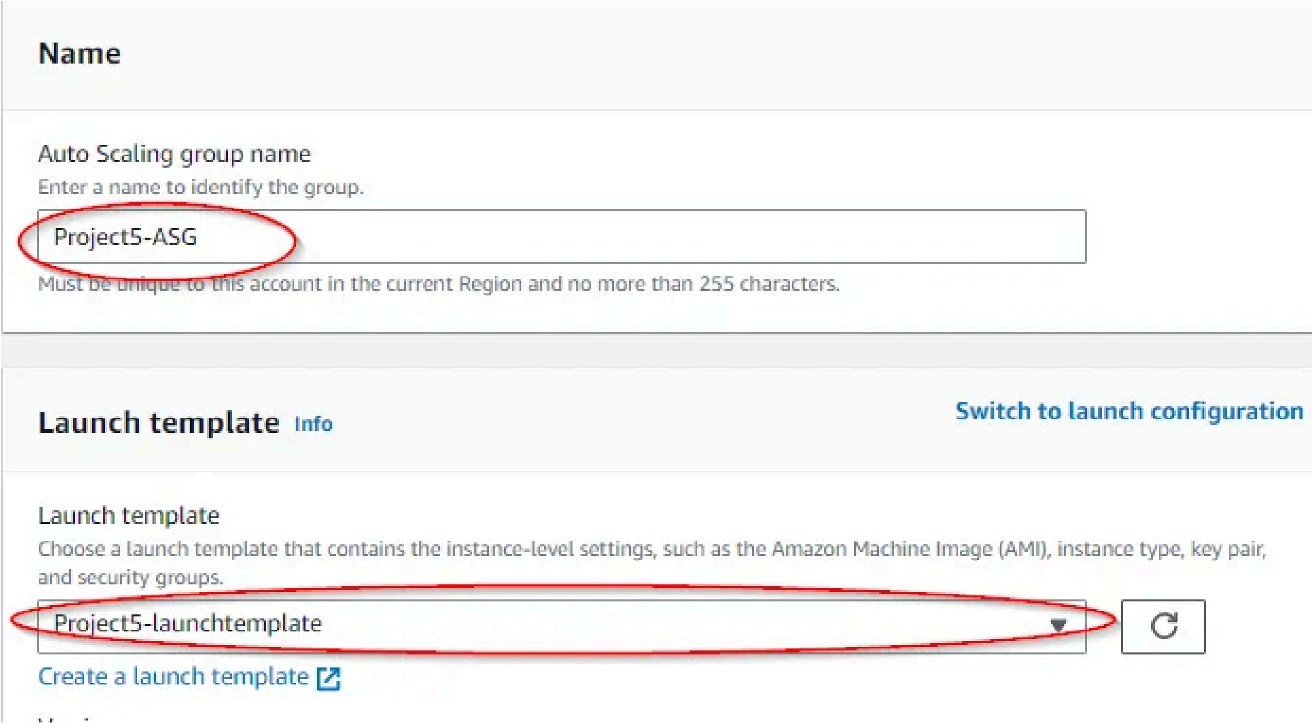
6. In Advanced details scroll to the bottom until you see User data and paste following the command

```
#!/bin/bash
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd
EC2AZ=$(curl -s http://169.254.169.254/latest/meta-data/placement/availabilit
echo '<center><h1>This Amazon EC2 instance is located in Availability Zone: A
sed "s/AZID/$EC2AZ/" /var/www/html/index.txt >
/var/www/html/index.html
```

7. Create Launch template

Create an Auto Scaling Group

1. Type EC2 in the search bar -> On the left-hand side locate Auto Scaling -> Auto Scaling Groups -> Select Create Auto Scaling Group -> Name group -> Select launch template we just created



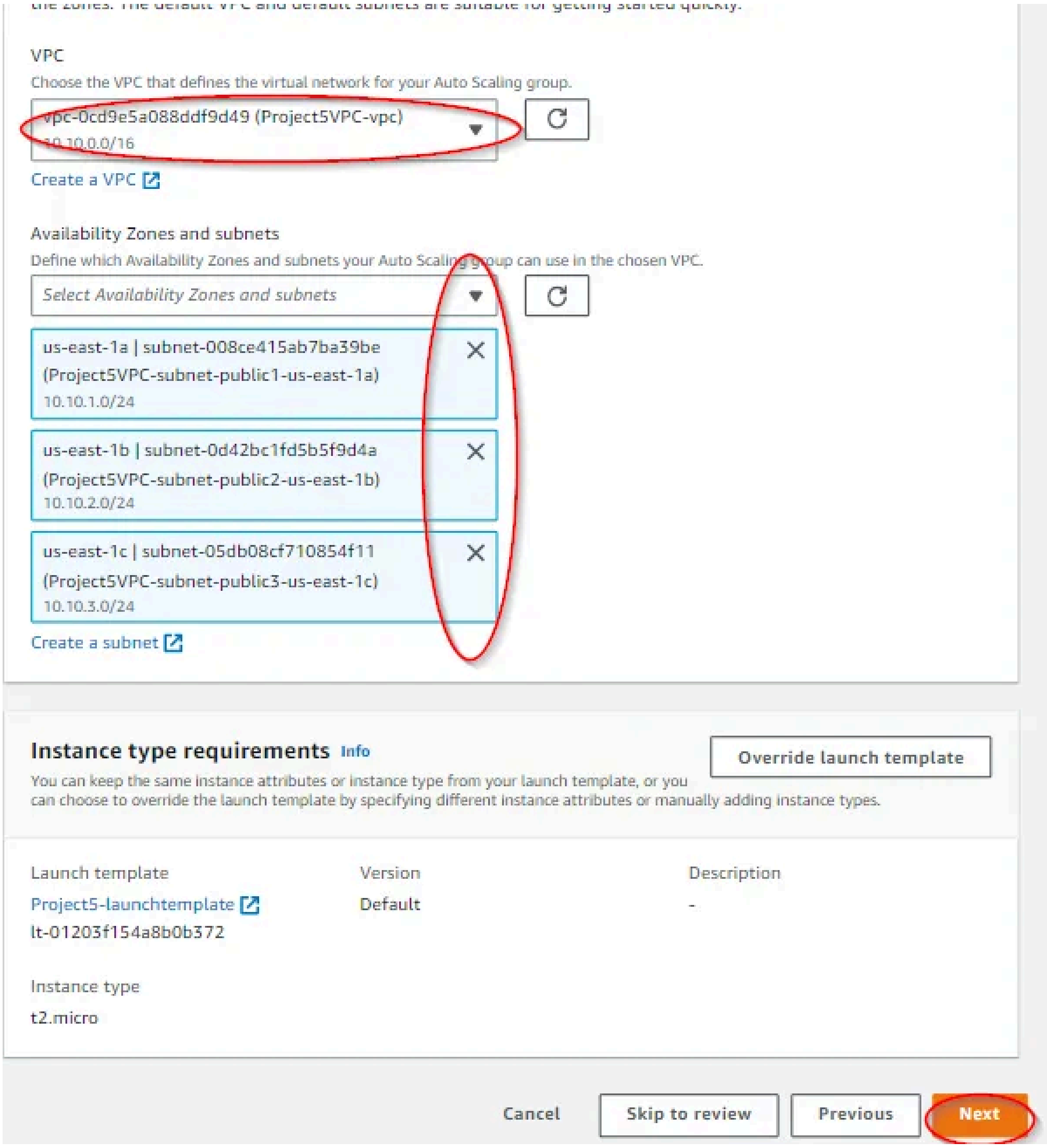
Name

Auto Scaling group name
Enter a name to identify the group.
Project5-ASG
Must be unique to this account in the current Region and no more than 255 characters.

Launch template Info

Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.
Project5-launchtemplate

2. Select the VPC we created earlier -> Select all Availability Zones and subnets -> Select next



the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC

Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-0cd9e5a088ddf9d49 (Project5VPC-vpc)
10.10.0.0/16

Availability Zones and subnets

Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

us-east-1a | subnet-008ce415ab7ba39be (Project5VPC-subnet-public1-us-east-1a) 10.10.1.0/24

us-east-1b | subnet-0d42bc1fd5b5f9d4a (Project5VPC-subnet-public2-us-east-1b) 10.10.2.0/24

us-east-1c | subnet-05db08cf710854f11 (Project5VPC-subnet-public3-us-east-1c) 10.10.3.0/24

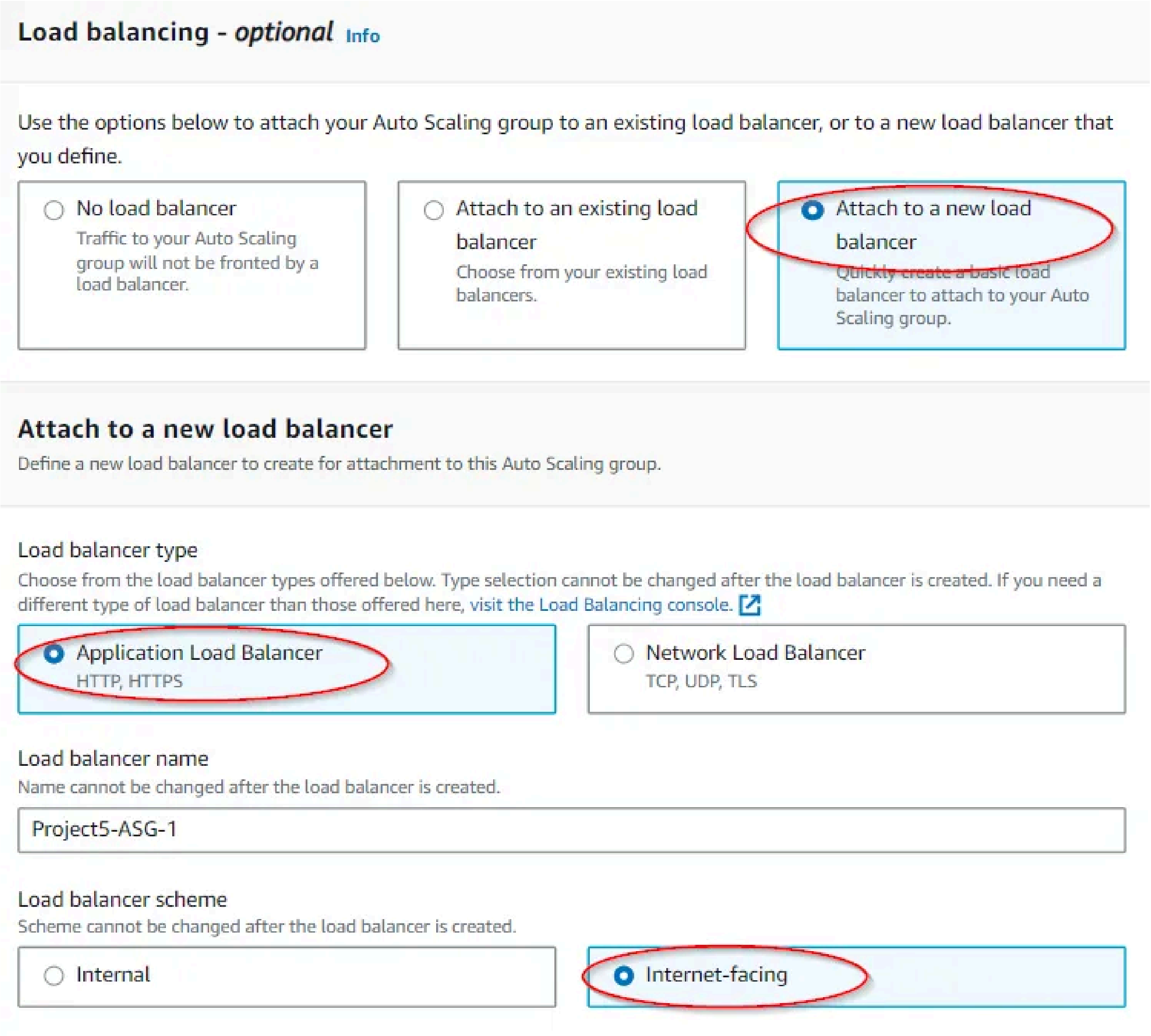
Instance type requirements Info

You can keep the same instance attributes or instance type from your launch template, or you can choose to override the launch template by specifying different instance attributes or manually adding instance types.

Launch template	Version	Description
Project5-launchtemplate lt-01203f154a8b0b372	Default	-

Instance type
t2.micro

3. We attach a new load balancer in order to distribute incoming traffic evenly across all the instances in the group, ensuring that no single instance becomes overwhelmed or overloaded with requests. An internet-facing load balancer has a public IP address, which clients on the internet can use to connect to your application.



Load balancing - *optional* Info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

○ No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

○ Attach to an existing load balancer
Choose from your existing load balancers.

● Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

Attach to a new load balancer

Define a new load balancer to create for attachment to this Auto Scaling group.

Load balancer type
Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the Load Balancing console.

● Application Load Balancer
HTTP, HTTPS

○ Network Load Balancer
TCP, UDP, TLS

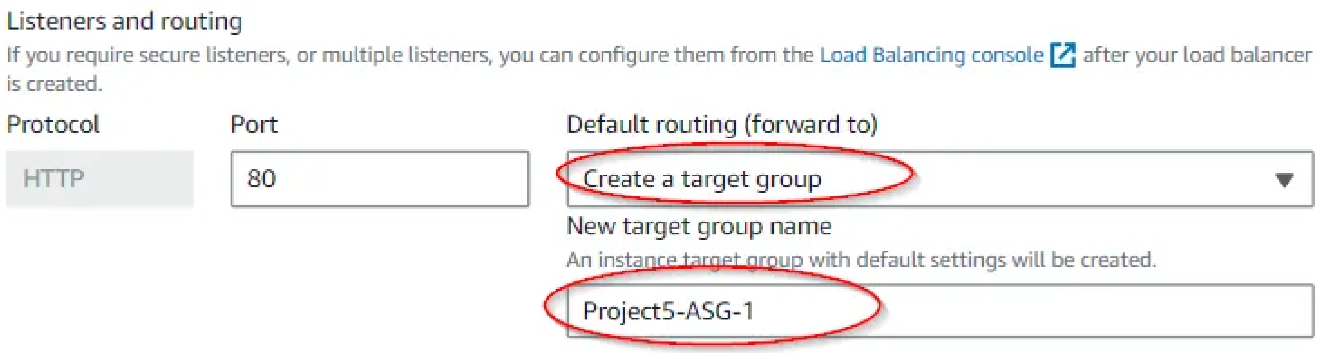
Load balancer name
Name cannot be changed after the load balancer is created.
Project5-ASG-1

Load balancer scheme
Scheme cannot be changed after the load balancer is created.

○ Internal

● Internet-facing

4. When you create a listener and routing rule on your load balancer, you specify which target group(s) should receive the incoming traffic.

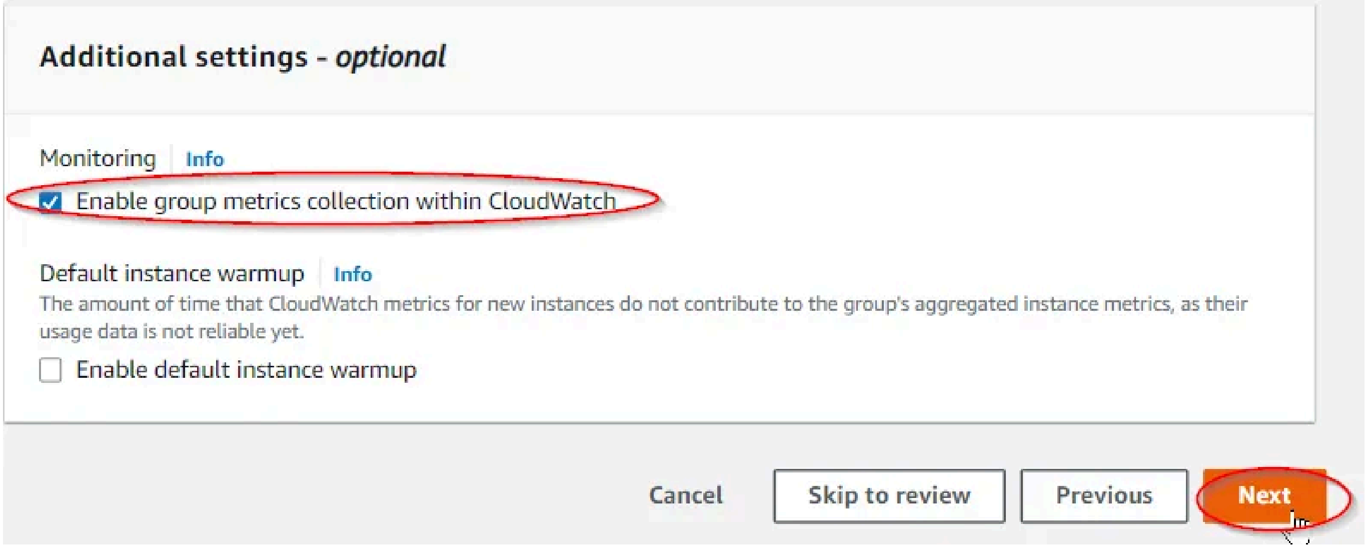


Listeners and routing
If you require secure listeners, or multiple listeners, you can configure them from the Load Balancing console after your load balancer is created.

Protocol	Port	Default routing (forward to)
HTTP	80	Create a target group ▼

New target group name
An instance target group with default settings will be created.
Project5-ASG-1

5. Enabling group metrics collection with CloudWatch for your load balancer allows you to monitor the performance of your load balancer and its associated resources.



Additional settings - *optional*

Monitoring Info

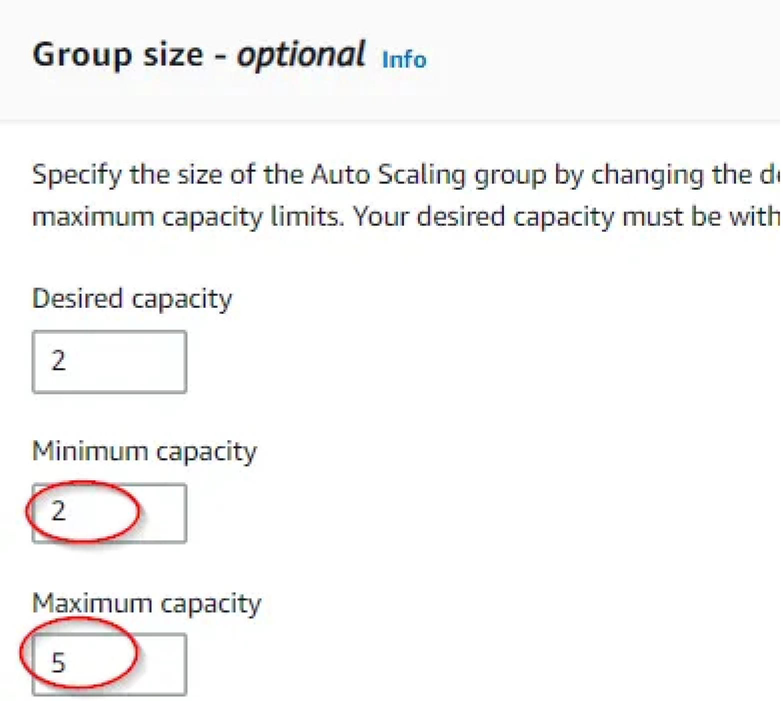
☑ Enable group metrics collection within CloudWatch

Default instance warmup Info

The amount of time that CloudWatch metrics for new instances do not contribute to the group's aggregated instance metrics, as their usage data is not reliable yet.

□ Enable default instance warmup

6. The Auto Scaling minimum should be 2 and the maximum 5.



Group size - *optional* Info

Specify the size of the Auto Scaling group by changing the d
maximum capacity limits. Your desired capacity must be with

Desired capacity

2

Minimum capacity

2

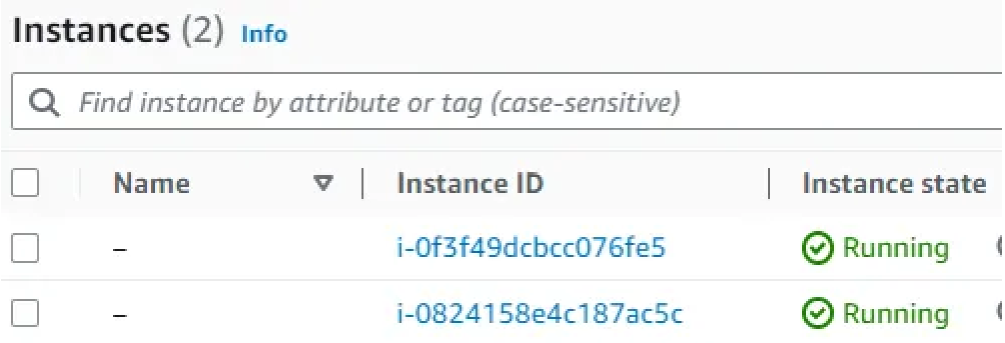
Maximum capacity

5

7. Click Next until you locate Create Auto Scaling Group

Let’s see if our Instances are up and running!

1. In the search bar type EC2 -> Instances

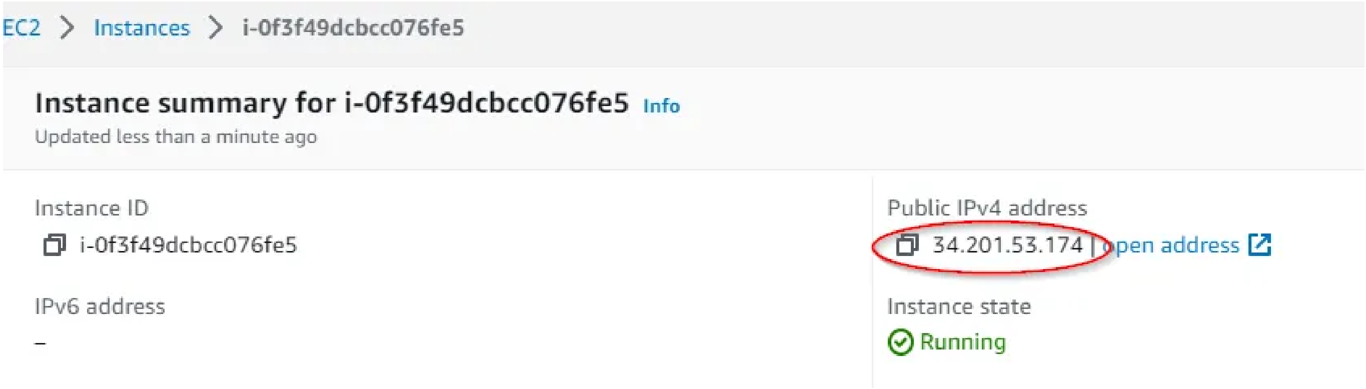


Instances (2) Info

Find instance by attribute or tag (case-sensitive)

□	Name	Instance ID	Instance state
□	–	i-0f3f49dcbcc076fe5	✓ Running
□	–	i-0824158e4c187ac5c	✓ Running

2. Locate the Public IPv4 address -> Open browser -> http://34.201.53.174



EC2 > Instances > i-0f3f49dcbcc076fe5

Instance summary for i-0f3f49dcbcc076fe5 Info

Updated less than a minute ago

Instance ID

i-0f3f49dcbcc076fe5

IPv6 address

–

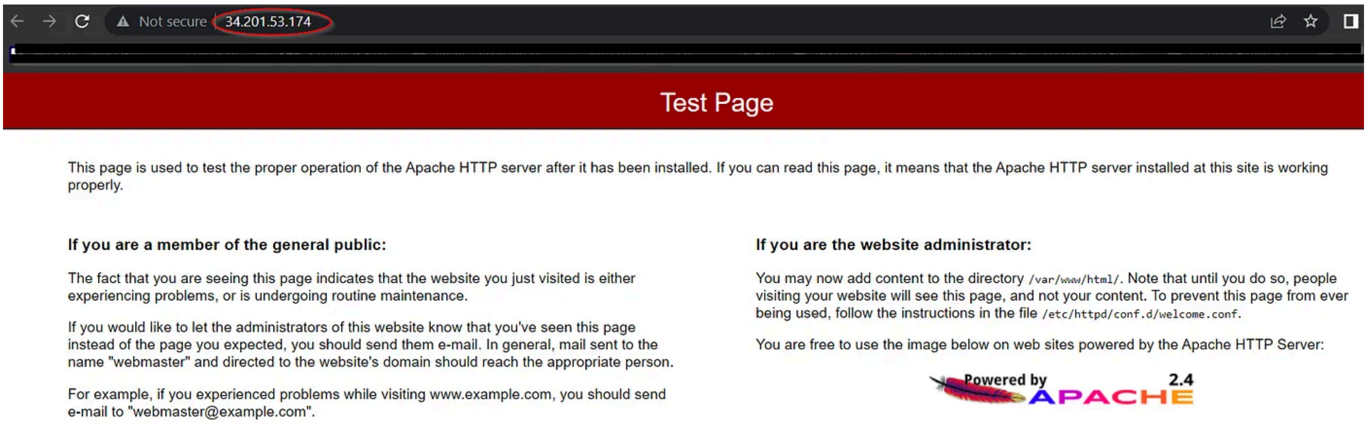
Public IPv4 address

34.201.53.174 | open address

Instance state

✓ Running

3. Everything looks good to go!



Not secure | 34.201.53.174

[redacted]

Test Page

This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.

If you are a member of the general public:

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting www.example.com, you should send e-mail to "webmaster@example.com".

If you are the website administrator:

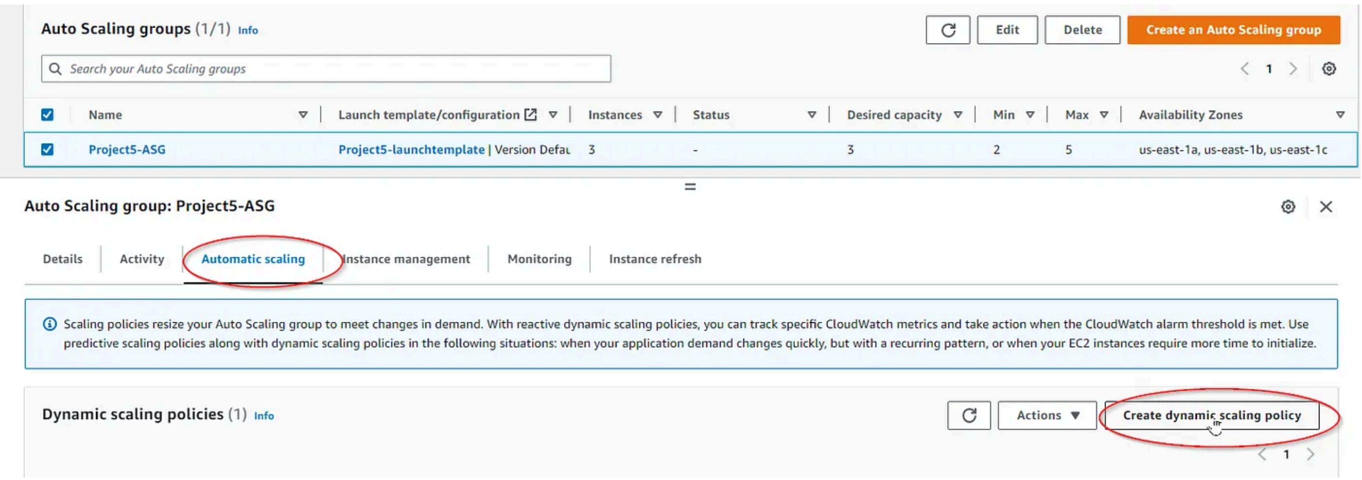
You may now add content to the directory `/var/www/html/`. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file `/etc/httpd/conf.d/welcome.conf`.

You are free to use the image below on web sites powered by the Apache HTTP Server:

Advanced:

Add a target policy for the ASG to scale after cpu utilization is above 50%. After the autoscaling group has been created, find a stress tool to be able to stress an instance above 50% to see if your scaling policy works! After the autoscaling group has been created, find a stress tool to be able to stress an instance above 50% to see if your scaling policy works!

1. In the search bar type EC2-> Scroll down to Auto Scaling groups > Select group > Go to Automatic scaling > Create dynamic scaling policy


Auto Scaling groups (1/1) Info
Edit
Delete
Create an Auto Scaling group
Search your Auto Scaling groups
1

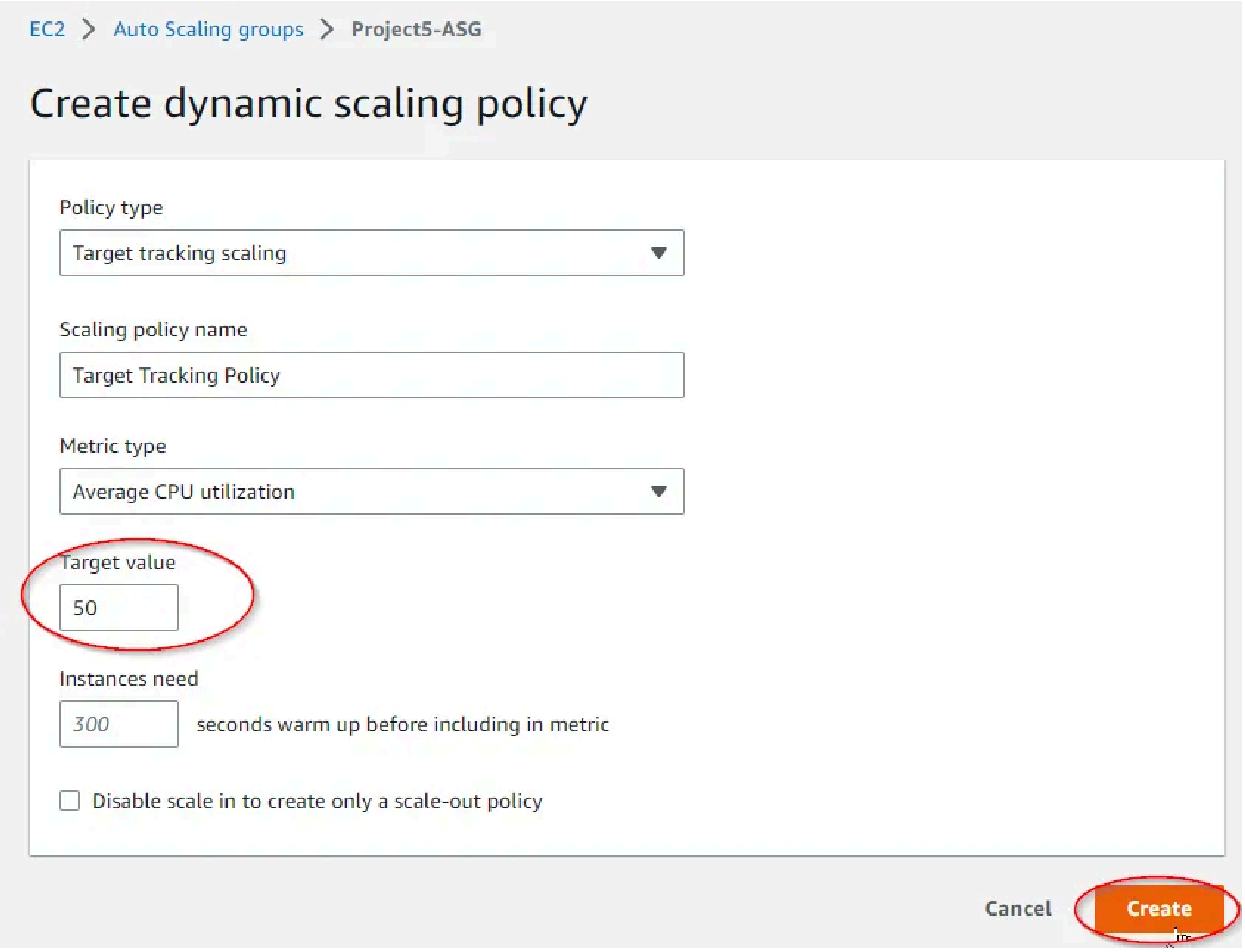
☑	Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
☑	Project5-ASG	Project5-launchtemplate | Version Defau	3	-	3	2	5	us-east-1a, us-east-1b, us-east-1c

Auto Scaling group: Project5-ASG
Details
Activity
Automatic scaling
Instance management
Monitoring
Instance refresh

Scaling policies resize your Auto Scaling group to meet changes in demand. With reactive dynamic scaling policies, you can track specific CloudWatch metrics and take action when the CloudWatch alarm threshold is met. Use predictive scaling policies along with dynamic scaling policies in the following situations: when your application demand changes quickly, but with a recurring pattern, or when your EC2 instances require more time to initialize.

Dynamic scaling policies (1) Info
Actions ▼
Create dynamic scaling policy
1

3. Enter 50 for the target value -> Create



EC2 > Auto Scaling groups > Project5-ASG

Create dynamic scaling policy

Policy type: Target tracking scaling

Scaling policy name: Target Tracking Policy

Metric type: Average CPU utilization

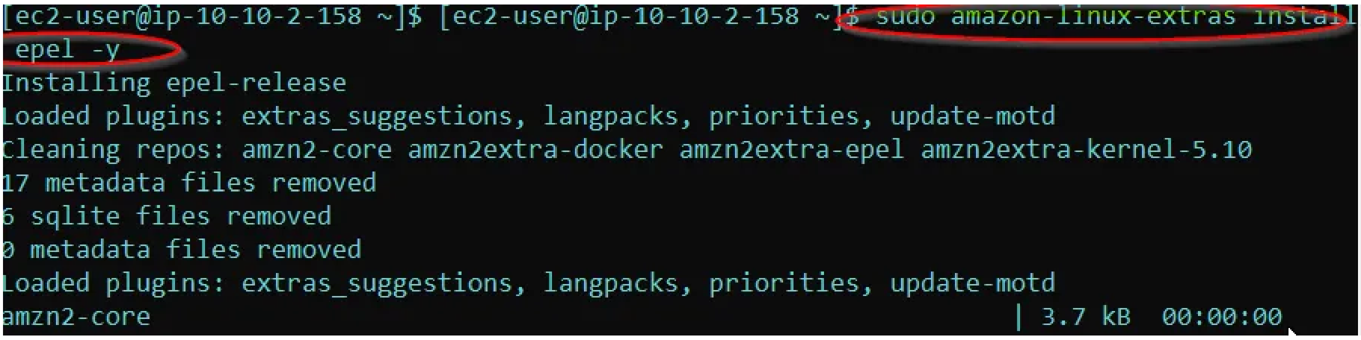
Target value: 50

Instances need 300 seconds warm up before including in metric

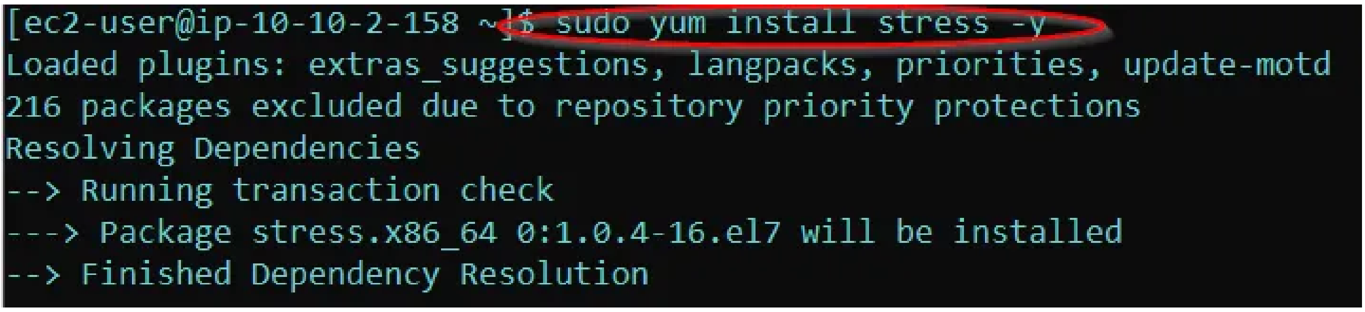
☐ Disable scale in to create only a scale-out policy

4. SSH into one of your instances and run the following commands to install a stress utility

```
sudo amazon-linux-extras install epel -y
sudo yum install stress -y
```



```
[ec2-user@ip-10-10-2-158 ~]$ [ec2-user@ip-10-10-2-158 ~]$ sudo amazon-linux-extras install
 epel -y
Installing epel-release
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-docker amzn2extra-epel amzn2extra-kernel-5.10
17 metadata files removed
6 sqlite files removed
0 metadata files removed
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                   | 3.7 kB  00:00:00
```

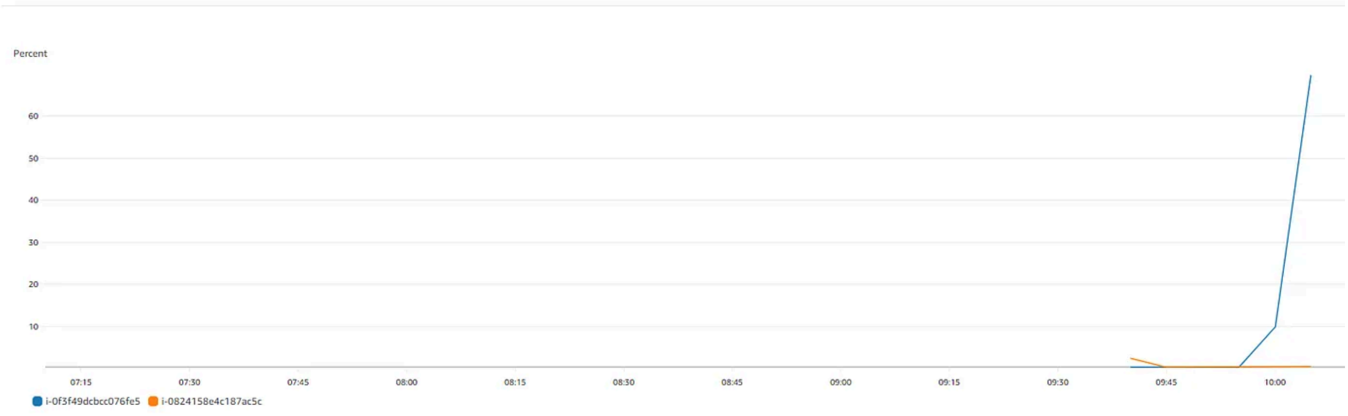


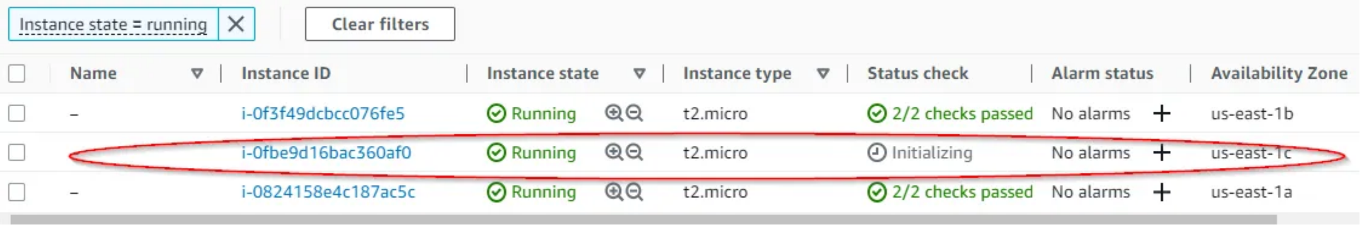
```
[ec2-user@ip-10-10-2-158 ~]$ sudo yum install stress -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
216 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
---> Package stress.x86_64 0:1.0.4-16.el7 will be installed
--> Finished Dependency Resolution
```

5. Once installed, CPU load can be generated using Stress by running:

```
stress --cpu 1 --timeout 300
```

6. Instance surpassed 60% and generated a new instance





Instance state = running ✕ | Clear filters

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
□	–	i-0f3f49dcbcc076fe5	Running	t2.micro	2/2 checks passed	No alarms +	us-east-1b
□		i-0fbe9d16bac360af0	Running	t2.micro	Initializing	No alarms +	us-east-1c
□	–	i-0824158e4c187ac5c	Running	t2.micro	2/2 checks passed	No alarms +	us-east-1a

Clean up

1. In the search bar type EC2 -> Detach Load balancer -> Delete Auto Scaling Group -> Delete Launch Template -> Terminate Instance

2. In the search bar type VPC -> Delete VPC

In conclusion, by following these steps to create a VPC with subnets, an autoscaling group, and a load balancer, you have set up a highly available web application infrastructure on AWS. This infrastructure will allow for automatic scaling and distribution of traffic to provide a reliable and responsive user experience. Additionally, the security groups you have set up will ensure that only authorized traffic is allowed to access your resources, providing a secure environment for your web application.