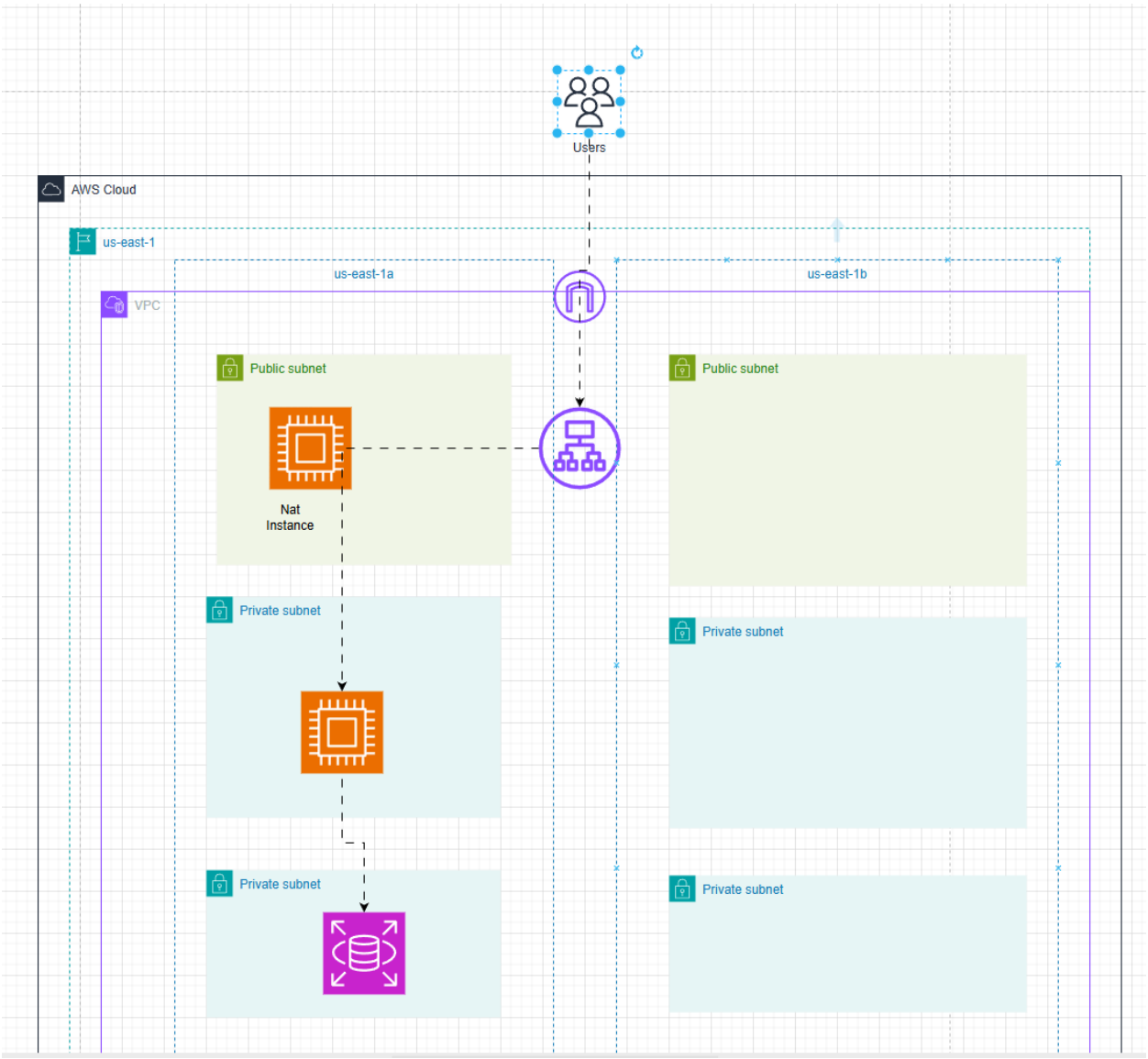
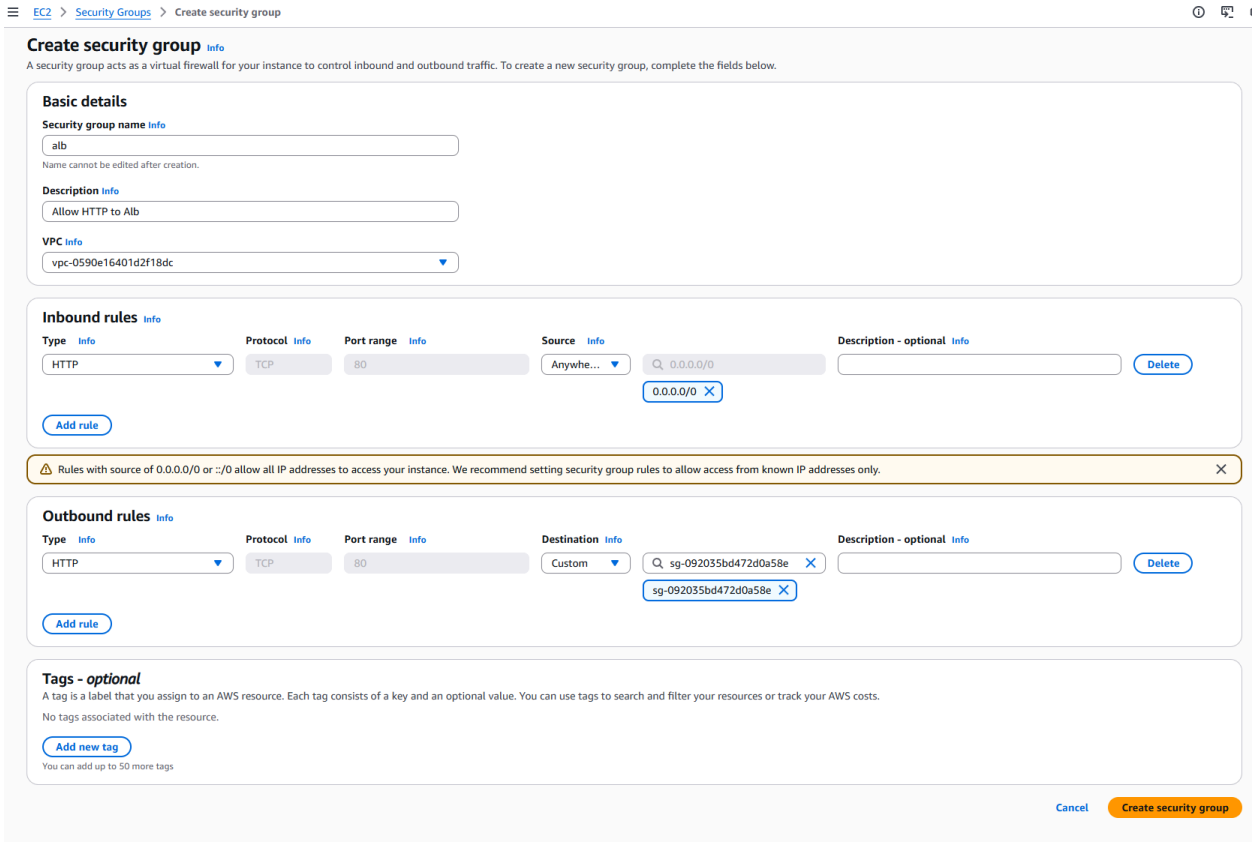
1. 

2. Create the application load balancer security group
    a. Allow HTTP inbound from every where
    b. Allow HTTP outbound to the php server security group



3. Create target group called php-servers
    a. Add the php-server to it

Create target group

A target group can be made up of one or more targets. Your load balancer routes requests to the targets in a target group and performs health checks on the targets.

Settings - *immutable*

Choose a target type and the load balancer and listener will route traffic to your target. These settings can't be modified after target group creation.",

Target type

Indicate what resource type you want to target. Only the selected resource type can be registered to this target group.

- (●) **Instances**
  Supports load balancing to instances in a VPC. Integrate with Auto Scaling Groups or ECS services for automatic management.
  Suitable for: ALB NLB GWLB
- ( ) **IP addresses**
  Supports load balancing to VPC and on-premises resources. Facilitates routing to IP addresses and network interfaces on the same instance. Supports IPv6 targets.
  Suitable for: ALB NLB GWLB
- ( ) **Lambda function**
  Supports load balancing to a single Lambda function. ALB required as traffic source.
  Suitable for: ALB
- ( ) **Application Load Balancer**
  Allows use of static IP addresses and PrivateLink with an Application Load Balancer. NLB required as traffic source.
  Suitable for: NLB

Target group name

Name must be unique per Region per AWS account.

php-servers

Accepts: a-z, A-Z, 0-9, and hyphen (-). Can't begin or end with hyphen. 1-32 total characters; Count: 11/32

Protocol

Protocol for communication between the load balancer and targets.

HTTP

Port

Port number where targets receive traffic. Can be overridden for individual targets during registration.

80

1-65535

IP address type

Only targets with the indicated IP address type can be registered to this target group.

- (●) **IPv4**
  Each instance has a default network interface (eth0) that is assigned the primary private IPv4 address. The instance's primary private IPv4 address is the one that will be applied to the target.
- ( ) **IPv6**
  Each instance you register must have an assigned primary IPv6 address. This is configured on the instance's default network interface (eth0). Learn more

VPC

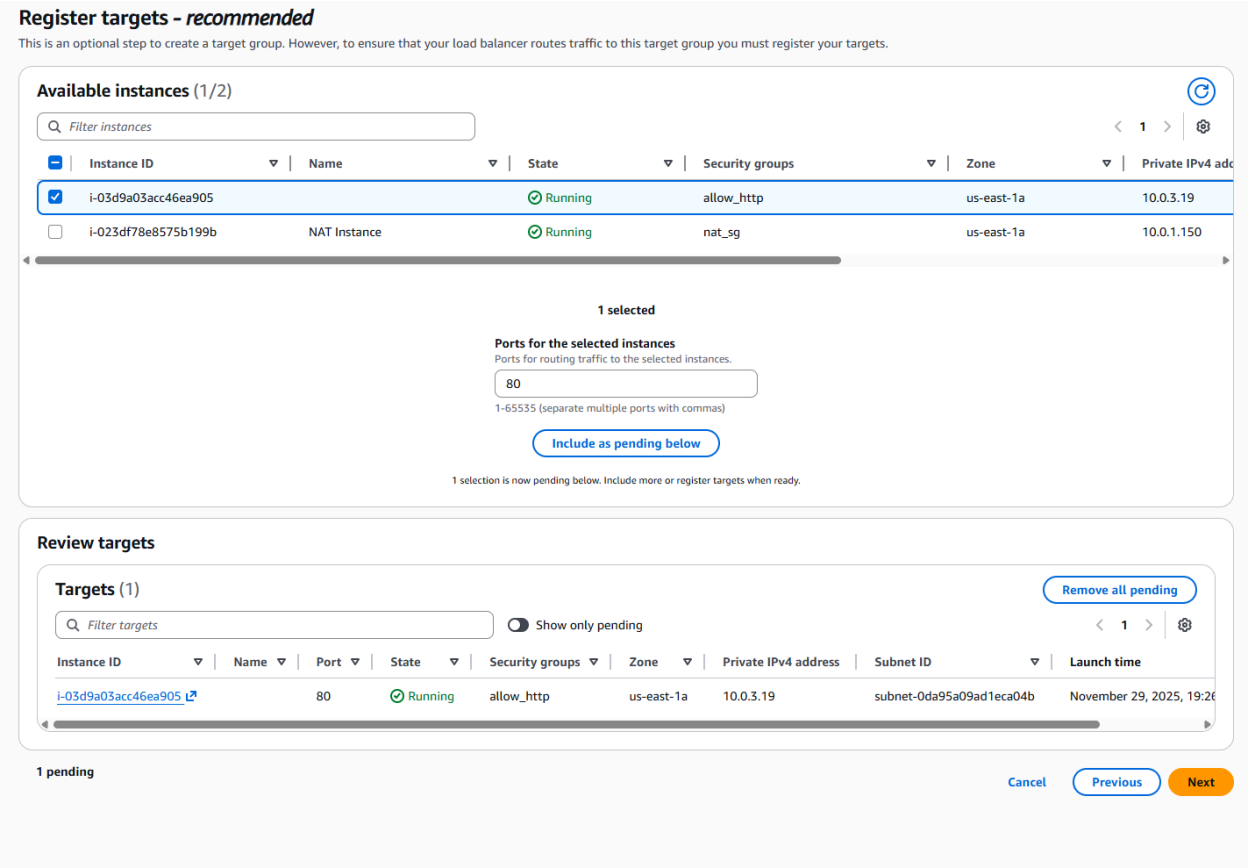
Select the VPC with the instances that you want to include in the target group. Only VPCs that support the IP address type selected above are available in this list.

vpc-0590e16401d2f18dc
10.0.0.0/16

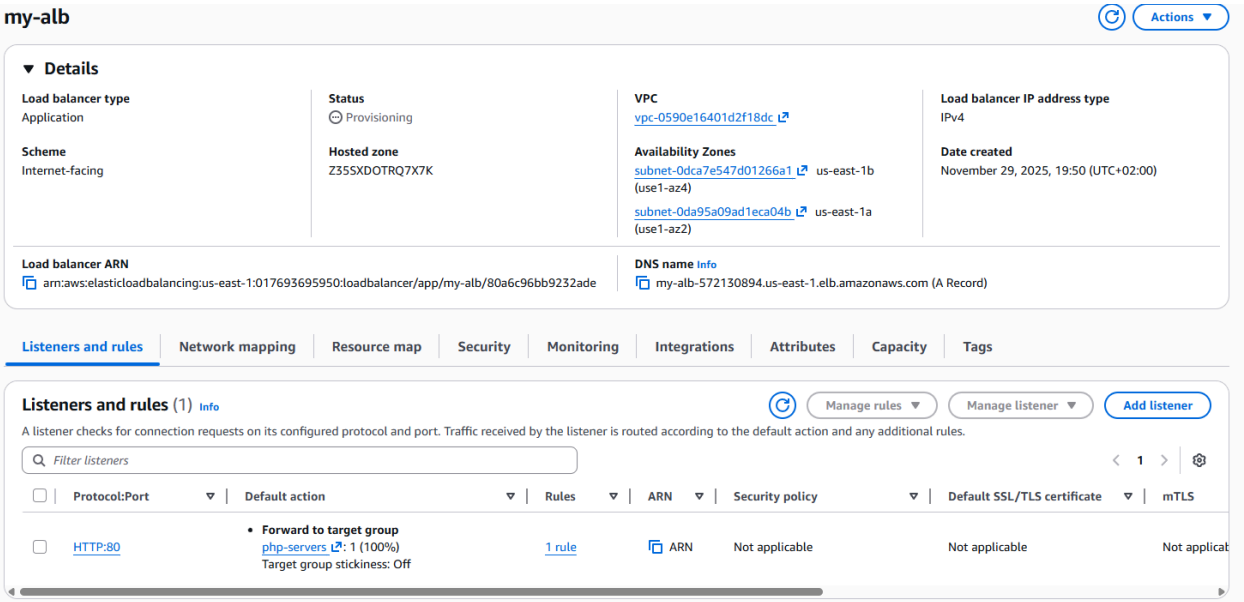
Create VPC

Protocol version

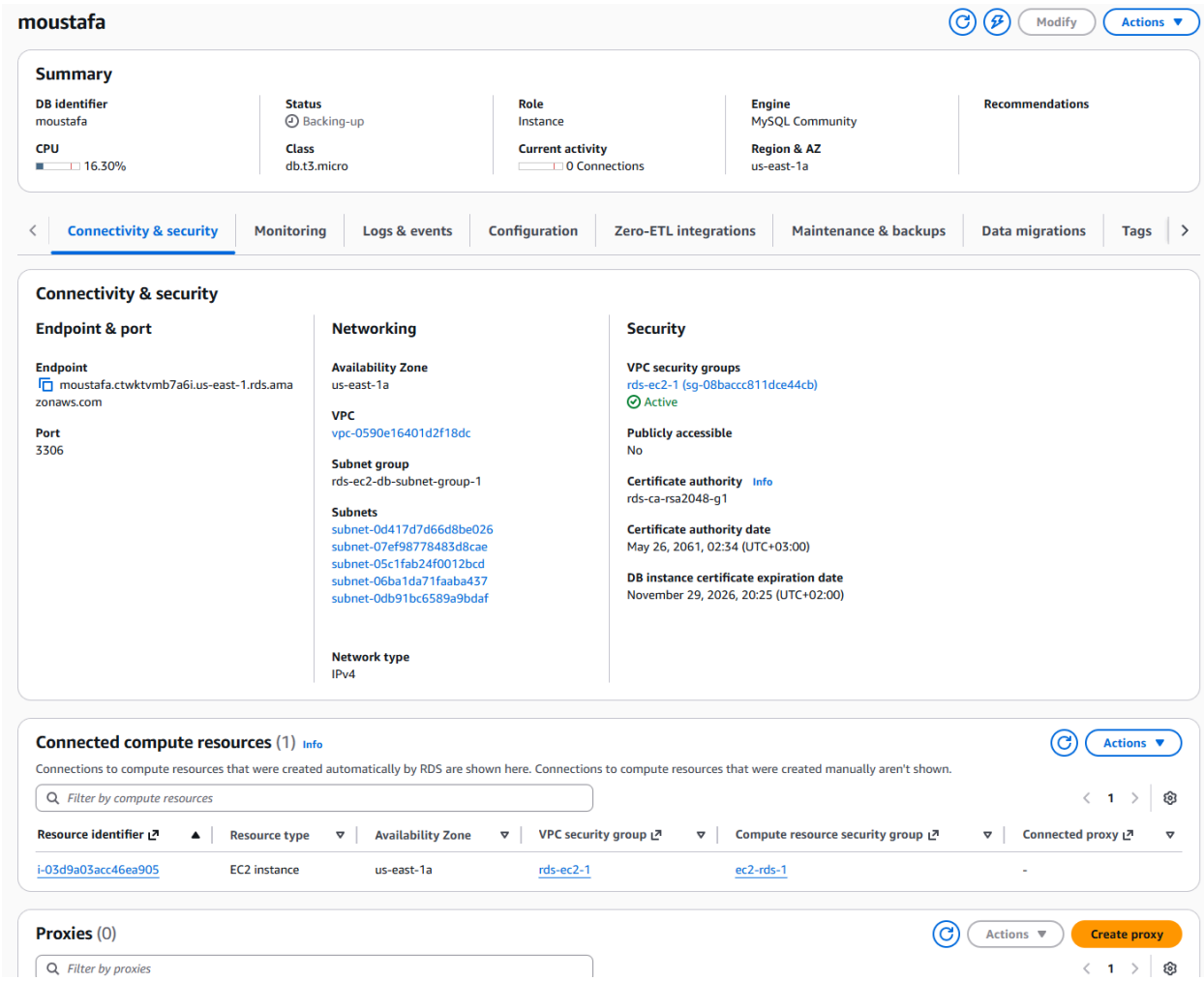
- (●) **HTTP1**
  Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.
- ( ) **HTTP2**
  Send requests to targets using HTTP/2. Supported when the request protocol is HTTP/2 or gRPC, but gRPC-specific features are not available.



4. Create internet facing alb in the correct vpc and correct **public** subnets and correct sg and select all public subnets, create a listener on port HTTP And add the php-server target group as a target for it
    a. Wait until it's state is active



5. Create a subnet group with the third tier subnets (10.0.5.0/24, 10.0.6.0/24)
6. Create RDS database
    a. Engine type: MYSQL
    b. Template free tier (single AZ DB)
    c. DB identifier with your choice and master username “admin”
    d. Choose a master password
    e. Db.t3.micro db instance class
    f. Connect to the php-server
    g. Disable the public access



7. Connect to the php server
    a. Switch to the root user (sudo -i)
    b. Move to /var/www/html
    c. Update the php.index file with the content below
    d. Update the password and the hostname

```php
<?php
error_reporting(E_ALL);
ini_set('display_errors', 1);

$dbname = 'mysql';  // don't update this line
$dbuser = 'admin';
$dbpass = '';
$dbhost = '';

$link = mysqli_connect($dbhost, $dbuser, $dbpass) or die("Unable to Connect to '$dbhost'");
mysqli_select_db($link, $dbname) or die("Could not open the db '$dbname'");

$test_query = "SHOW TABLES FROM $dbname";
$result = mysqli_query($link, $test_query);

$tblCnt = 0;
while($tbl = mysqli_fetch_array($result)) {
  $tblCnt++;
}

if (!$tblCnt) {
  echo "Connected successfully<br />\n";
} else {
  echo "Connected successfully<br />\n";
}
?>
```