

# Assignment 2 – Developing a highly available Photo Album website

Ta Quang Tung – 104222196

Tutorial date: Saturday (3 P.M. – 5 P.M.)

Link to photo album website: <http://webserverbalancer-1710659052.us-east-1.elb.amazonaws.com/photoalbum/album.php>

## I - CREATE THE VPC AND NAT INSTANCE

This section lists the steps needed to create and configure the VPC and NAT instance, which fulfils requirements 2.1 and 2.3 of the instructions.

### A - CREATE THE VPC

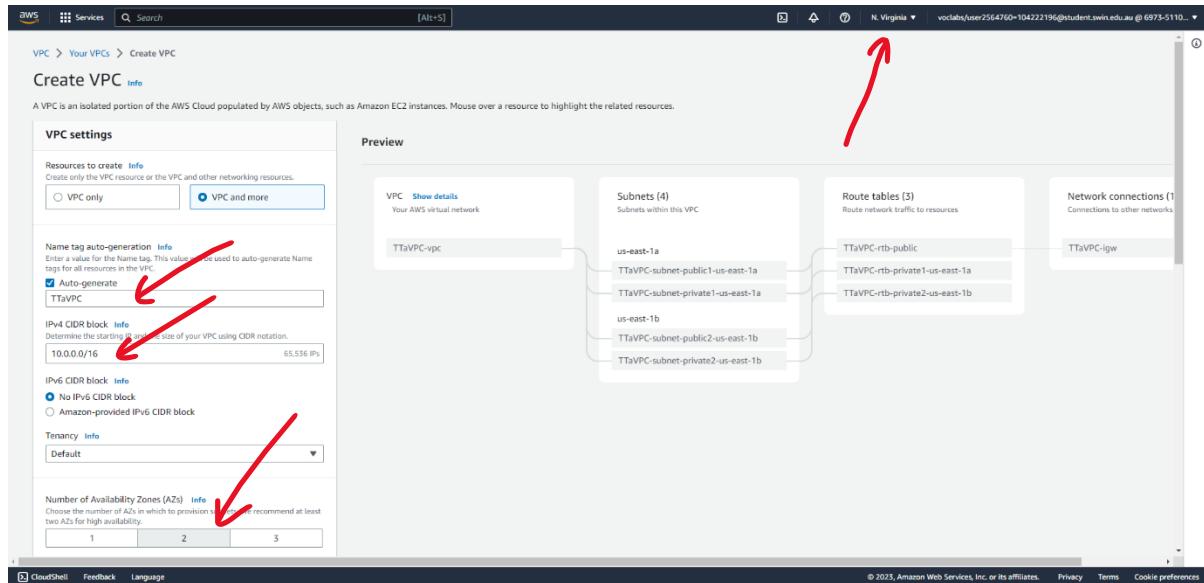


Fig. 1: Create the VPC (part 1/2), specifying its name, CIDR block, and number of AZs.

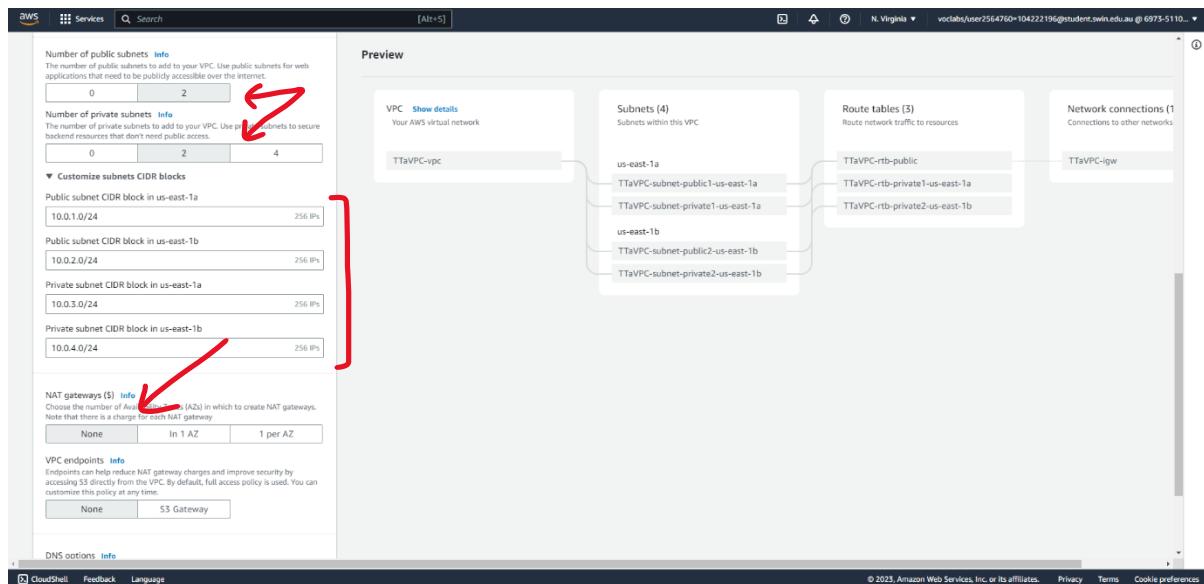


Fig. 2: Create the VPC (part 2/2), specifying the number of private/public subnets per AZ, subnets' CIDR blocks. The NAT gateway option is unselected because a NAT instance will be created instead.

## B – CREATE THE NAT INSTANCE

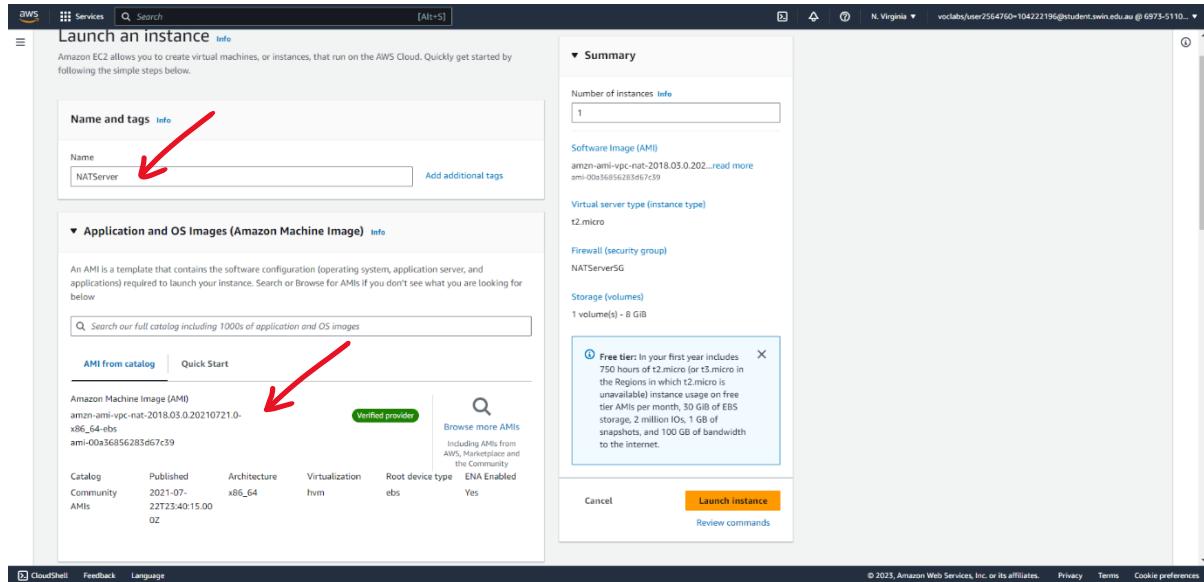


Fig. 3: Create the NAT instance (part 1/3), specifying the name and correct AMI.

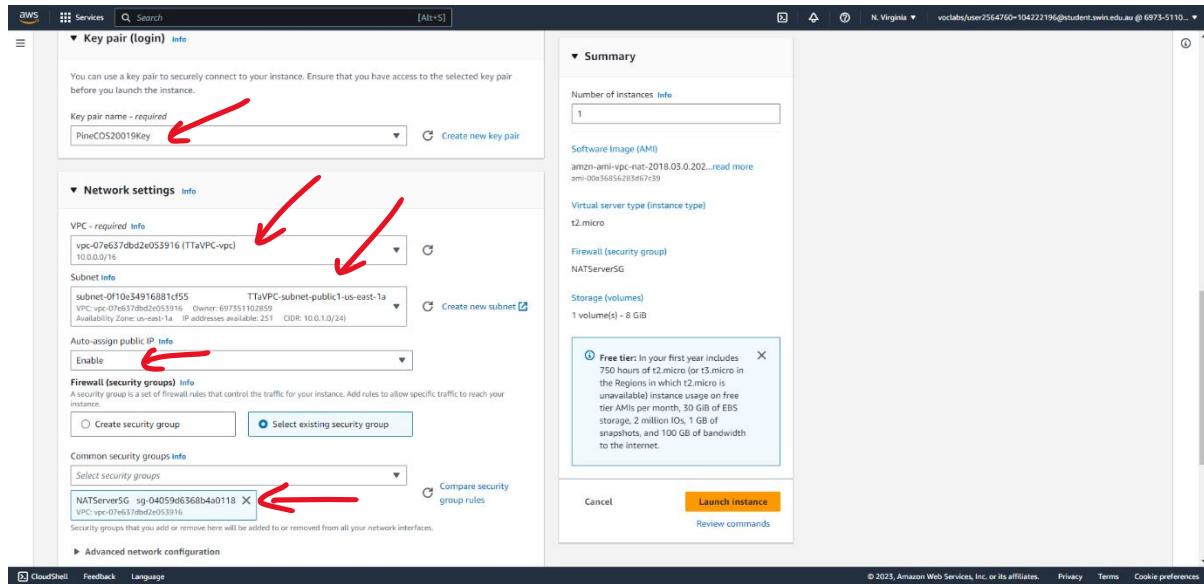


Fig. 4: Create the NAT instance (part 2/3), specifying the key pair and correct network settings.

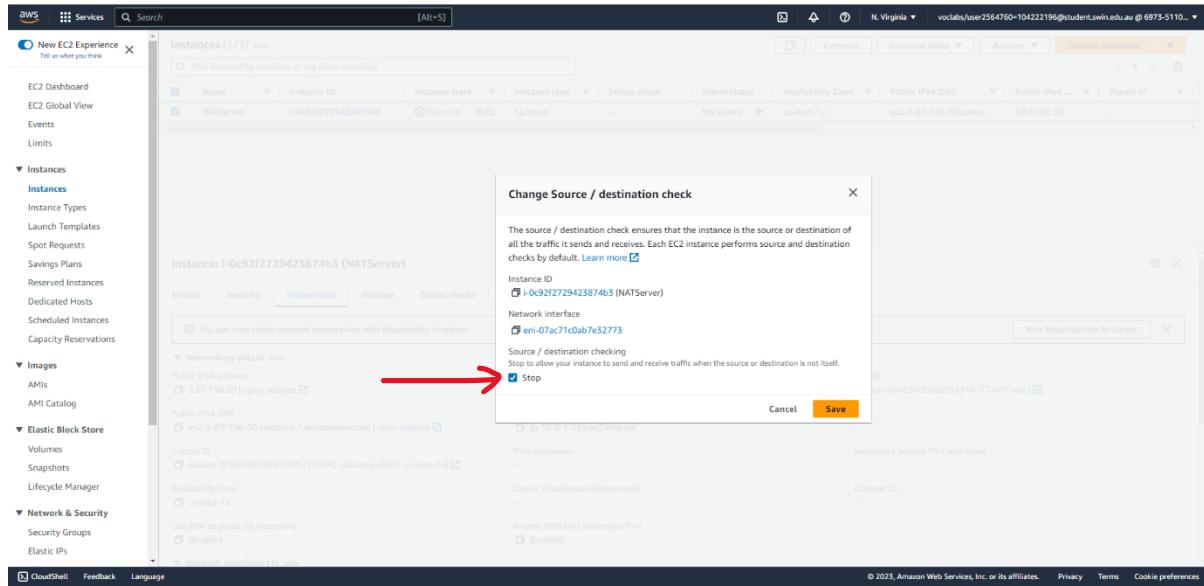


Fig. 5: Create the NAT instance (part 3/3), disabling source/destination check so that the instance can send/receive traffic from other sources/destinations.

Route table ID	Subnet associations	Main	VPC	Owner ID
rtb-00a47717c69a488e	-	Yes	vpc-07e637dbd2e053916   Tta...	697351102859
rtb-0e10c2bd3ff2bc4c	subnet-0cd46f5e06d706...	No	vpc-07e637dbd2e053916   Tta...	697351102859
rtb-0f838656b59e9c71	subnet-098a257a77395...	No	vpc-07e637dbd2e053916   Tta...	697351102859
rtb-0613985b211defbbc	2 subnets	No	vpc-07e637dbd2e053916   Tta...	697351102859
rtb-0c82c9938ad8f69c5	-	Yes	vpc-0da4efcd8e406d2a5	697351102859

Destination	Target	Status	Propagated
0.0.0.0/0	eni-07ac71c0ab7e52773	Active	No
10.0.0.0/16	local	Active	No

Fig. 6: Create a route on each private route table that directs internet-bound traffic to the NAT instance (1/2).

Route table ID	Explicit subnet associations	Main	VPC	Owner ID
rtb-00a47717c69a488e	-	Yes	vpc-07e637dbd2e053916   Tta...	697351102859
rtb-0e10c2bd3ff2bc4c	subnet-0cd46f5e06d706...	No	vpc-07e637dbd2e053916   Tta...	697351102859
rtb-0f838656b59e9c71	subnet-098a257a77395...	No	vpc-07e637dbd2e053916   Tta...	697351102859
rtb-0613985b211defbbc	2 subnets	No	vpc-07e637dbd2e053916   Tta...	697351102859
rtb-0c82c9938ad8f69c5	-	Yes	vpc-0da4efcd8e406d2a5	697351102859

Destination	Target	Status	Propagated
0.0.0.0/0	eni-07ac71c0ab7e52773	Active	No
10.0.0.0/16	local	Active	No

Fig. 7: Create a route on each private route table that directs internet-bound traffic to the NAT instance (2/2).

## II - SET UP THE RDS INSTANCE

This section lists the steps needed to create the RDS instance, which fulfils requirement 2.7 in the instructions.

**Subnet group details**

- VPC ID: vpc-07e637dbd2e053916
- ARN: arn:aws:rds:us-east-1:697351102859:subgrp:dbsubnetgroup
- Supported network types: IPv4
- Description: Subnet group for RDS.

**Subnets (2)**

Availability zone	Subnet ID	CIDR block
us-east-1b	subnet-098a257a7739561d	10.0.4.0/24
us-east-1a	subnet-0cd46f5e06e7068ad	10.0.3.0/24

**Tags (0)**

Manage tags

CloudShell Feedback Language

Fig. 8: Create a subnet group for the database instance, specifying the two private subnets of the VPC.

**Compute resource**

Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

Don't connect to an EC2 compute resource  
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

Connect to an EC2 compute resource  
Set up a connection to an EC2 compute resource for this database.

**Network type info**

To use dual-stack mode, make sure that you associate an IPv6 CIDR block with a subnet in the VPC you specify.

IPv4  
Your resources can communicate only over the IPv4 addressing protocol.

Dual-stack mode  
Your resources can communicate over IPv4, IPv6, or both.

**Virtual private cloud (VPC) info**

Choose the VPC. The VPC defines the virtual networking for this DB instance.

TTaVPC-vpc (vpc-07e637dbd2e053916)  
4 Subnets, 2 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change its VPC.

**DB subnet group info**

Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB instance can use in the VPC that you selected.

dbsubnetgroup  
2 Subnets, 2 Availability Zones

**Public access info**

Yes  
RDS assigns a public IP address to the database. Amazon EC2 instances and other resources outside of the VPC can connect to your database. Resources inside the VPC can also connect to the database. Choose one or more VPC security groups that specify which resources can connect to the database.

No  
RDS doesn't assign a public IP address to the database. Only Amazon EC2 instances and other resources inside the VPC can connect to your database. Choose one or more VPC security groups that specify which resources can connect to the database.

**MySQL**

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

CloudShell Feedback Language

Fig. 9: Create a MySQL database instance similar to Assignment 1B, this time specifying the new VPC and subnet group.

The screenshot shows the AWS RDS console with the 'photodb' database instance selected. The 'Summary' tab is active, displaying basic metrics like CPU usage (2.88%), status (Available), and engine (MySQL Community). The 'Connectivity & security' tab is also visible. A red arrow points from the 'Endpoint' field ('photodb.c6w0siafkfbt.us-east-1.rds.amazonaws.com') to the 'Security' section, which lists the 'DBServerSG' security group as active.

Fig. 10: The database instance has been successfully created. It is using the DBServerSG security group, which will be configured later. The database endpoint will be used to set up the webpage on the development server.

### III - SET UP THE LAMBDA FUNCTION

This section lists the steps needed to set up the Lambda function, which fulfills requirement 2.6 in the instructions.

The screenshot shows the 'Create new function' wizard. In the 'Basic information' step, a red arrow points to the 'Function name' field containing 'CreateThumbnail'. Another red arrow points to the 'Runtime' dropdown, which is set to 'Python 3.7'. A third red arrow points to the 'Existing role' dropdown, which is set to 'LabRole'.

Fig. 11: Create a new Lambda function, specifying its name, runtime language, and IAM role of LabRole.

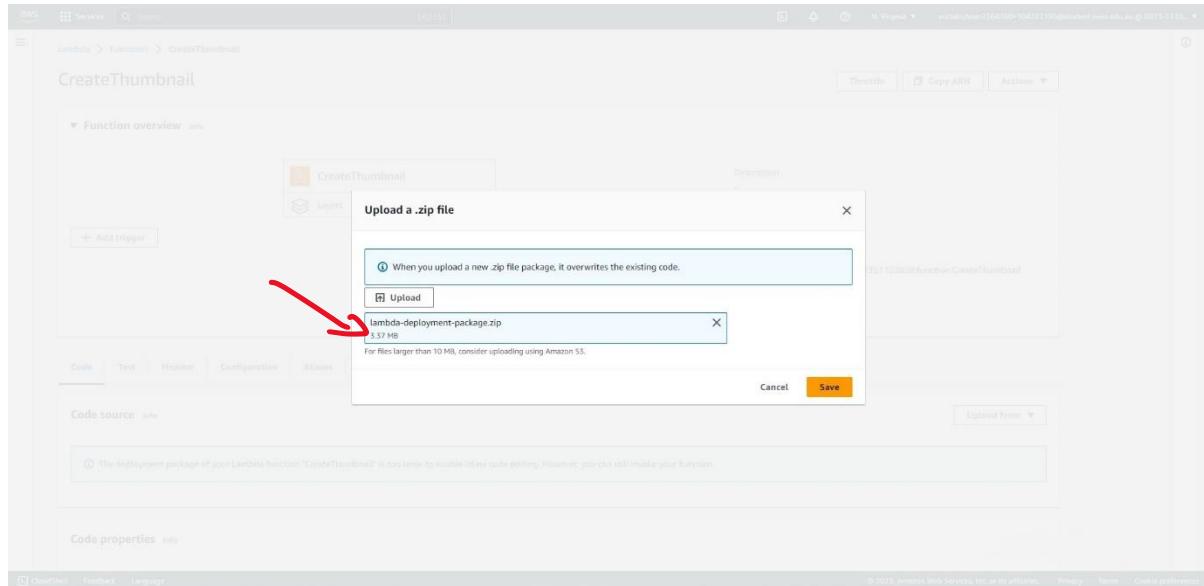


Fig. 12: Upload the .zip file containing the code for the Lambda function.

#### IV - SET UP THE S3 BUCKET

This section lists the steps needed to set up the S3 bucket for photo storage, which fulfils requirement 2.2 in the instructions.

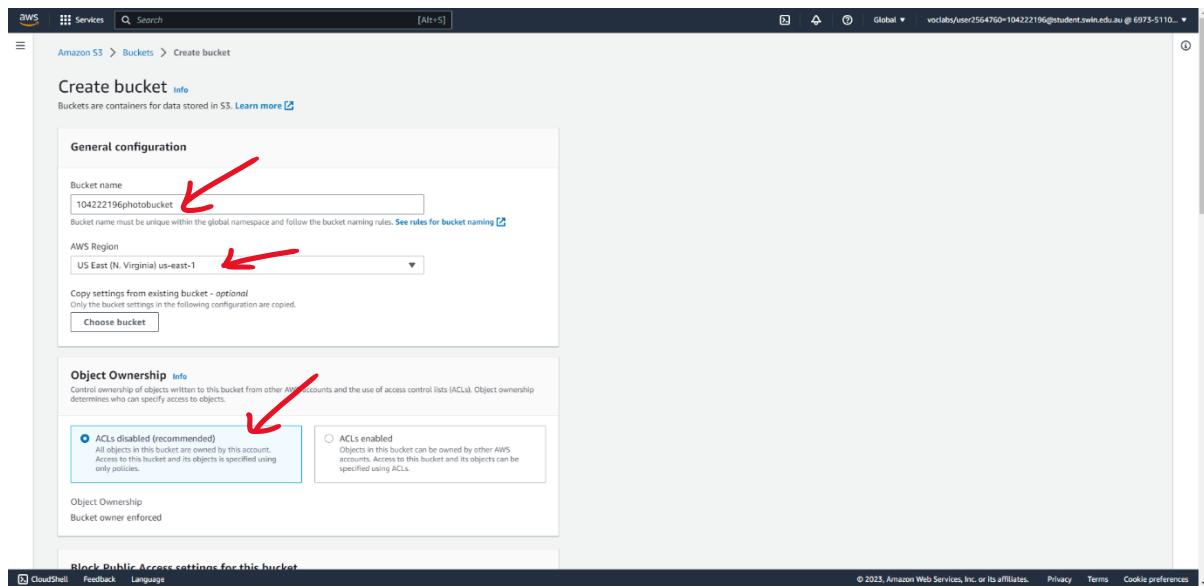


Fig. 13: Create a new S3 bucket (part 1/2), specifying its name, region, and object ownership settings.

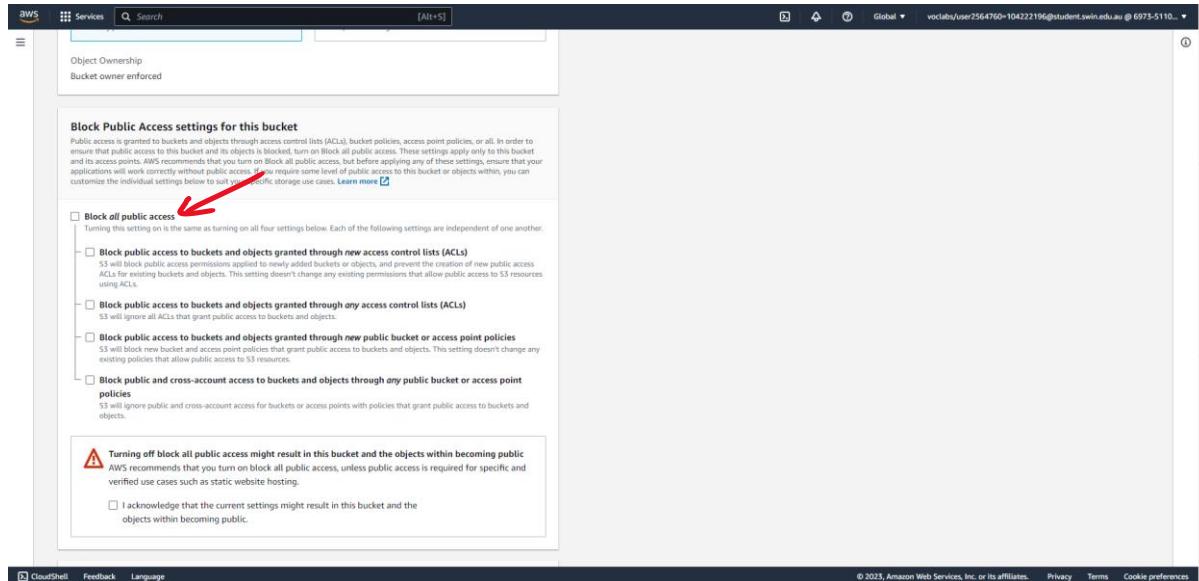


Fig. 14: Create a new S3 bucket (part 2/2), opening all public access to the bucket.

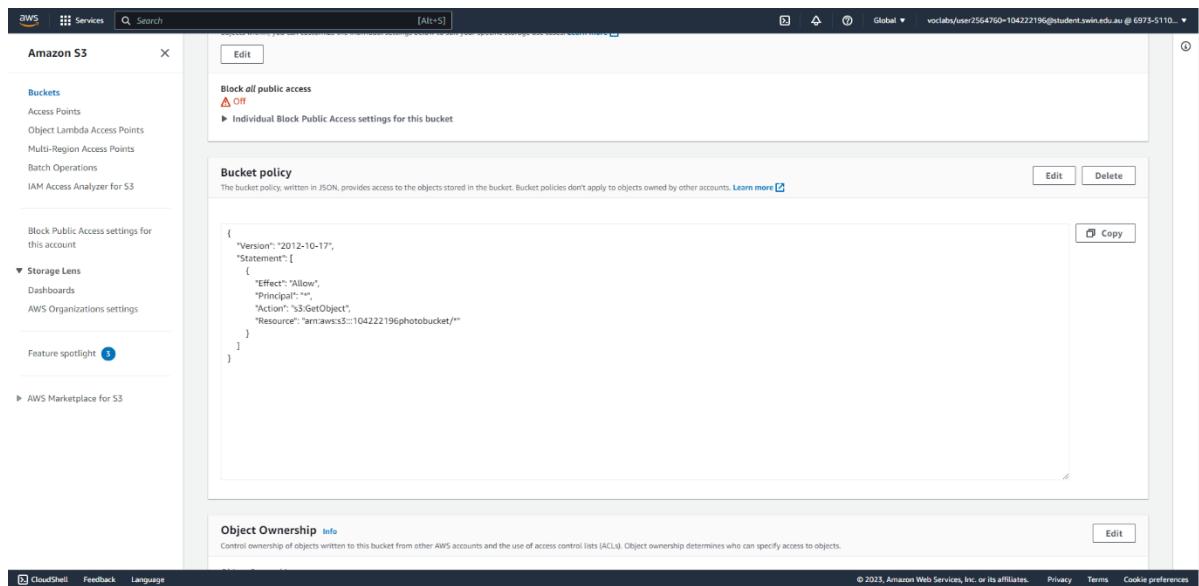


Fig. 15: Specify an access policy for the bucket, which at the moment allows all objects to be publicly retrieved. This policy will be tightened later.

## V - SET UP THE DEV SERVER

This section lists the steps needed to configure the Development Server instance, which will be used to create the AMI from which new Web Server instances will be launched. This step fulfils requirement 2.5 in the instructions.

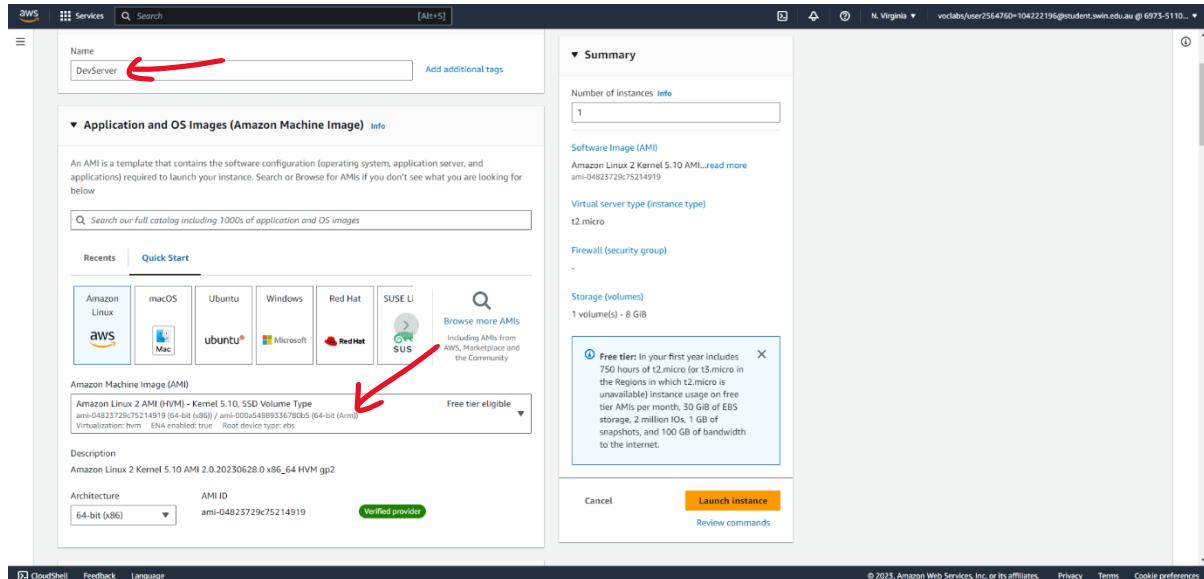


Fig. 16: Create a new EC2 instance for the Dev Server (part 1/3), specifying the name and correct AMI.

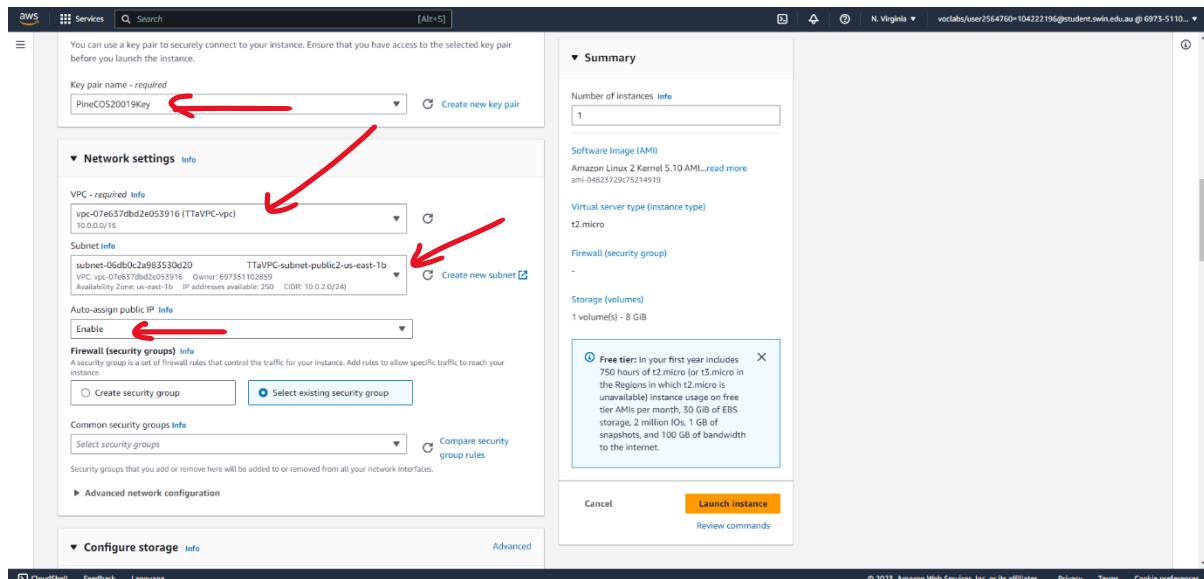


Fig. 17: Create a new EC2 instance for the Dev Server (part 2/3), specifying the key pair, VPC, subnet, public IP, and security group (which is currently empty)

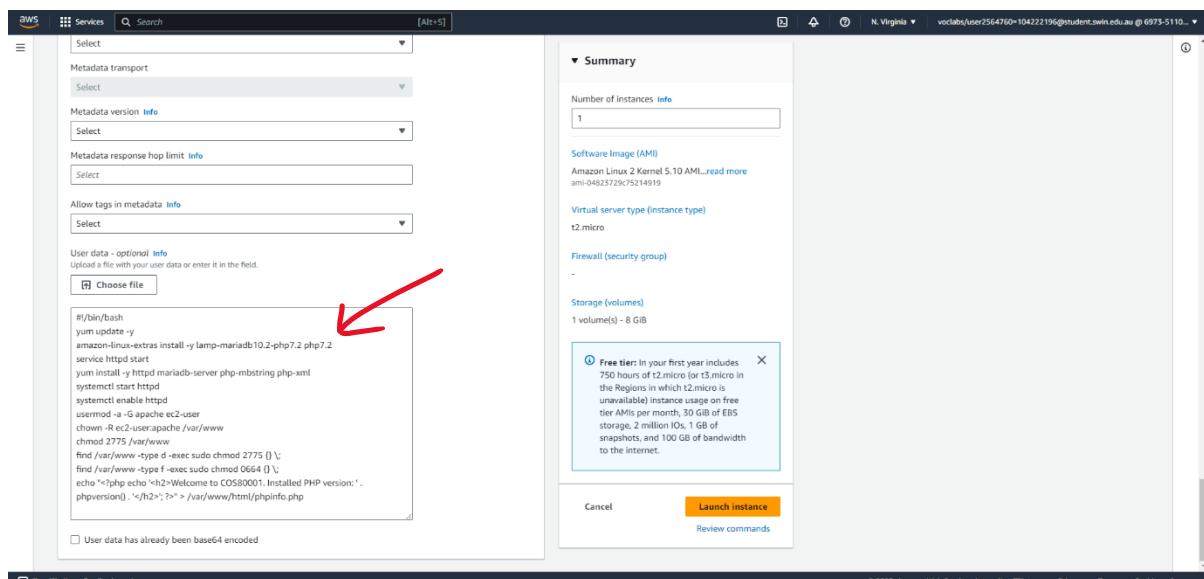


Fig. 18: Create a new EC2 instance for the Dev Server (part 3/3), specifying the user data script to install Apache server.

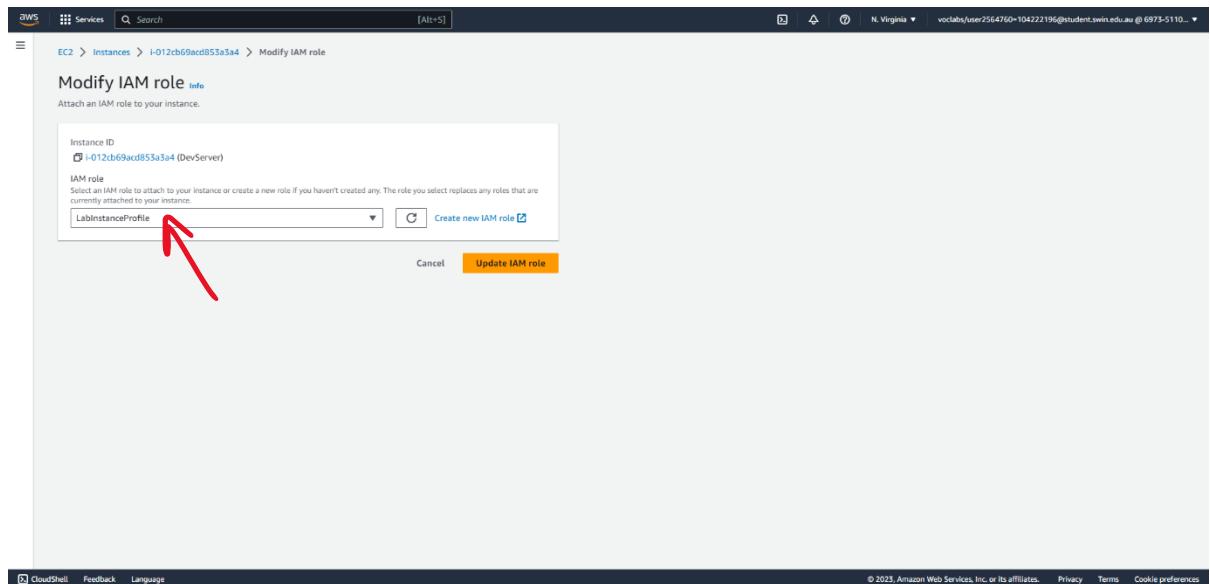


Fig. 19: Assign the correct IAM role to the Dev Server instance so that it can call AWS resources.

```
[ec2-user@ip-10-0-2-8:~]$ ls
aws aws.zip photoalbum phpinfo.php phpmyadmin phpMyAdmin-4.8.2-english.zip
[ec2-user@ip-10-0-2-8:~]$
```

A screenshot of an SSH terminal window titled "ec2-user@ip-10-0-2-8:~". The command "ls" is run, showing several files: "aws", "aws.zip", "photoalbum", "phpinfo.php", "phpmyadmin", and "phpMyAdmin-4.8.2-english.zip". Two red arrows point upwards from the bottom of the terminal window towards the command line area, indicating where user input was entered.

Fig. 20: SSH into the Dev Server and install the PHPMyAdmin package to configure the database instance as well as the AWS SDK to interact with AWS services through PHP code.

The screenshot shows the phpMyAdmin interface for a database named 'assignment2'. In the left sidebar, under the 'New' category, there is a 'photos' entry. A red arrow points from this entry to the 'photos' table in the main content area. Another red arrow points from the 'photos' entry to the 'Indexes' section below the table structure. The main content area displays the 'Table structure' for the 'photos' table, which has six columns: id, title, description, date\_created, keywords, and reference. The 'id' column is defined as an int type with AUTO\_INCREMENT, while the other five are varchar(255) types. Below the table structure, there is a section for creating an index, followed by 'Information' and 'Space usage' sections.

Fig. 21: After configuring PHPMyAdmin similar to Assignment 1B, access the database instance and create a table to store the necessary data.

The screenshot shows the WinSCP interface displaying the contents of the '/var/www/html/photoalbum/' directory. The address bar at the top shows the full path. A red arrow points from the 'constants.php' file in the list view to the address bar. The list view shows various files including '..', 'uploads', 'album.php', 'constants.php', 'defaultstyle.css', 'mydb.php', 'photo.php', 'photouploader.php', 'photouploadtemplat...', and 'utils.php'. Each file's name, size, last modified date, permissions, and owner are listed.

Fig. 22: After the database table, S3 bucket, and Lambda function have been set up, modify the constants.php file in the source code and upload everything to the correct directory on Dev Server through WinSCP.

## VI - CREATE THE TARGET GROUP, ELB, AND AUTO-SCALING GROUP

This section lists the steps needed to create the load balancer and auto-scaling group (ASG), which fulfills requirements 2.3 and 2.4 in the instructions.

### A – CREATE THE TARGET GROUP

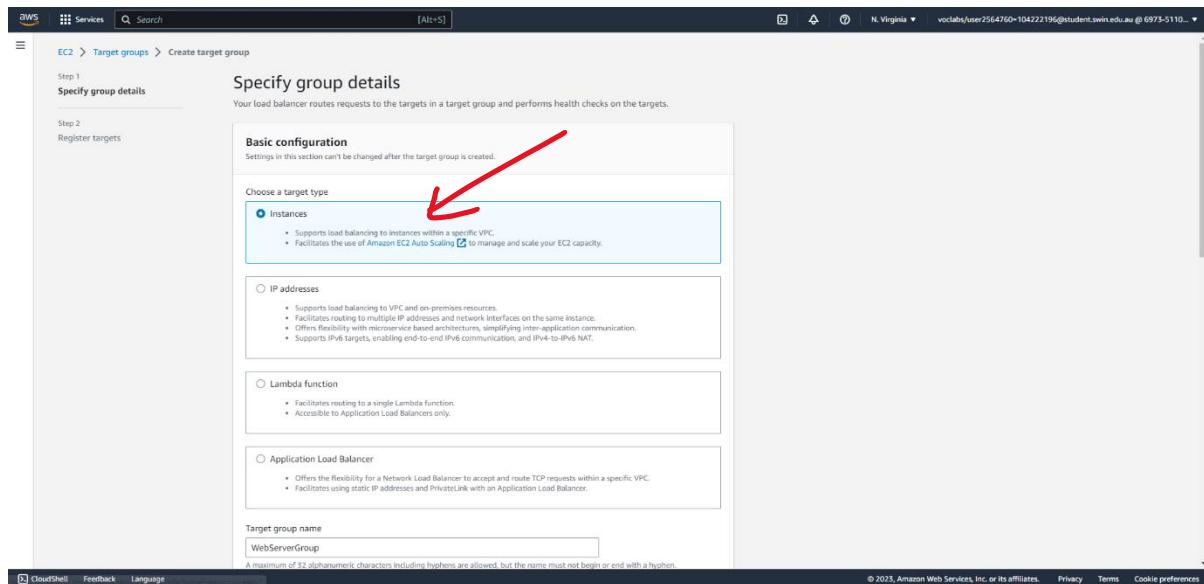


Fig. 23: Create a new target group (part 1/3), setting the type of target as Instances.

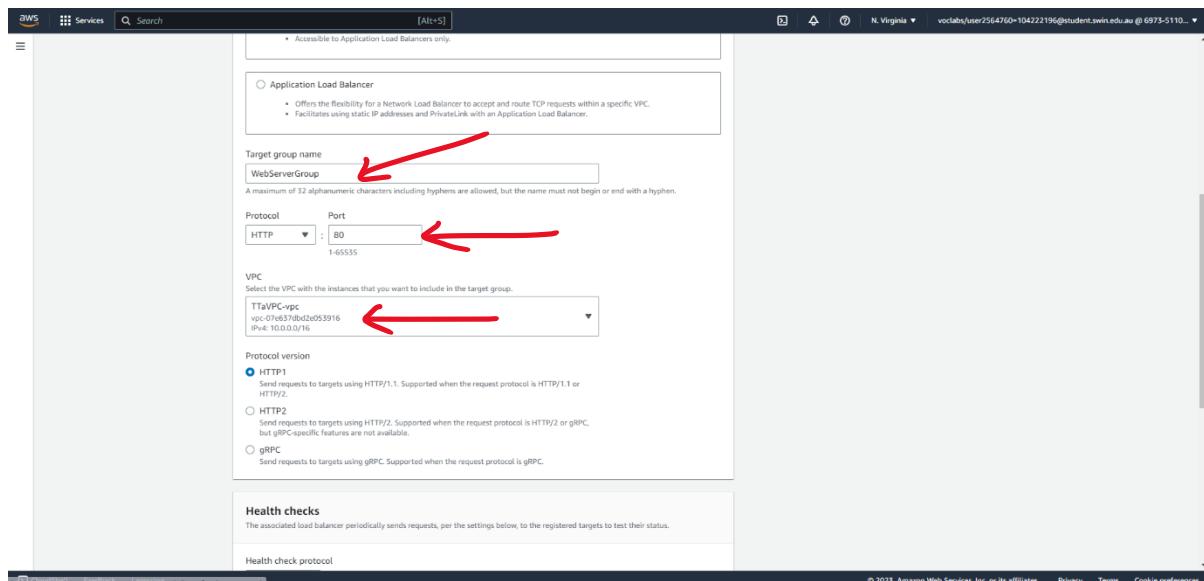


Fig. 24: Create a new target group (part 2/3), specifying the name, protocol, and VPC.

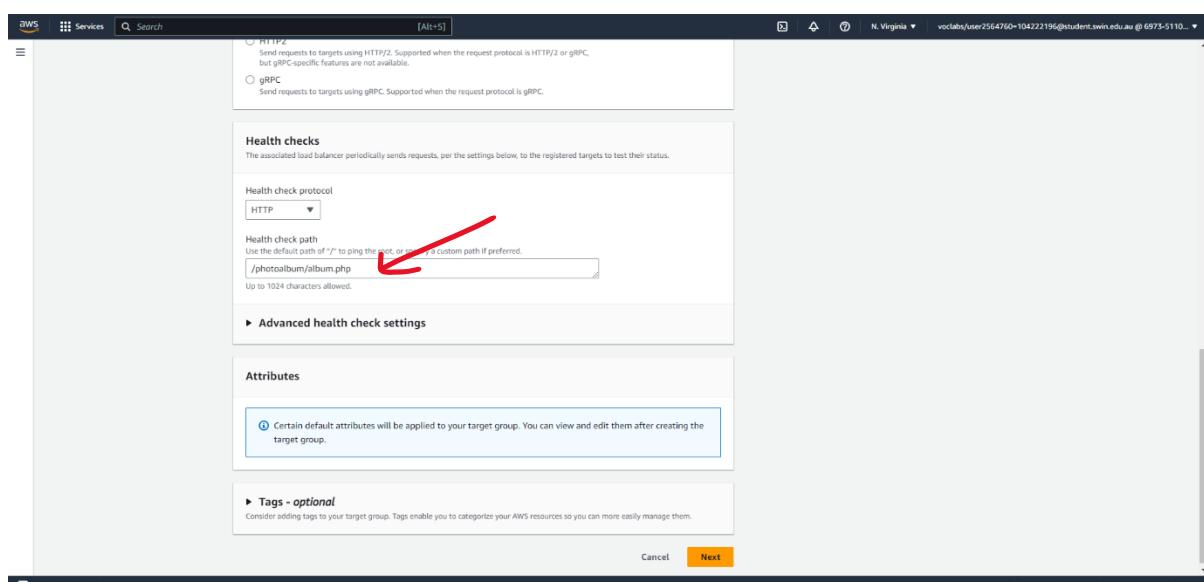


Fig. 25: Create a new target group (part 3/3), specifying the health check settings.

## B – CREATE THE APPLICATION LOAD BALANCER

**Create Application Load Balancer**

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and containers, based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule to apply, and if applicable, it selects a target from the target group for the rule action.

**Basic configuration**

Load balancer name: Name must be unique within your AWS account and can't be changed after the load balancer is created. **WebServerBalancer**

Scheme: **Internet-facing**

IP address type: **IPv4**

**Network mapping**

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

Fig. 26: Create an application load balancer (part 1/3), specifying the name and scheme.

**Network mapping**

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC: **TTaVPC-vpc**

Mappings:

- us-east-1a (use1-az6)**: Subnet: subnet-0f10e34916881cf55, Target: TTaVPC-subnet-public1-us-east-1a
- us-east-1b (use1-az1)**: Subnet: subnet-06db0c2a983530d20, Target: TTaVPC-subnet-public2-us-east-1b

Fig. 27: Create an application load balancer (part 2/3), specifying the VPC and public subnets.

**Listeners and routing**

A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

**Listener HTTP:80**

Protocol: <b>HTTP</b>	Port: <b>80</b>	Default action: <b>Info</b>
Forward to: <b>WebServerGroup</b> (target type: Instance, IPv4)		

**Add listener tag**

**Add-on services - optional**

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback Language

Fig. 28: Create an application load balancer (part 3/3), specifying the security group (which currently allows all inbound traffic) and listener.

### C – CREATE THE AUTO-SCALING GROUP

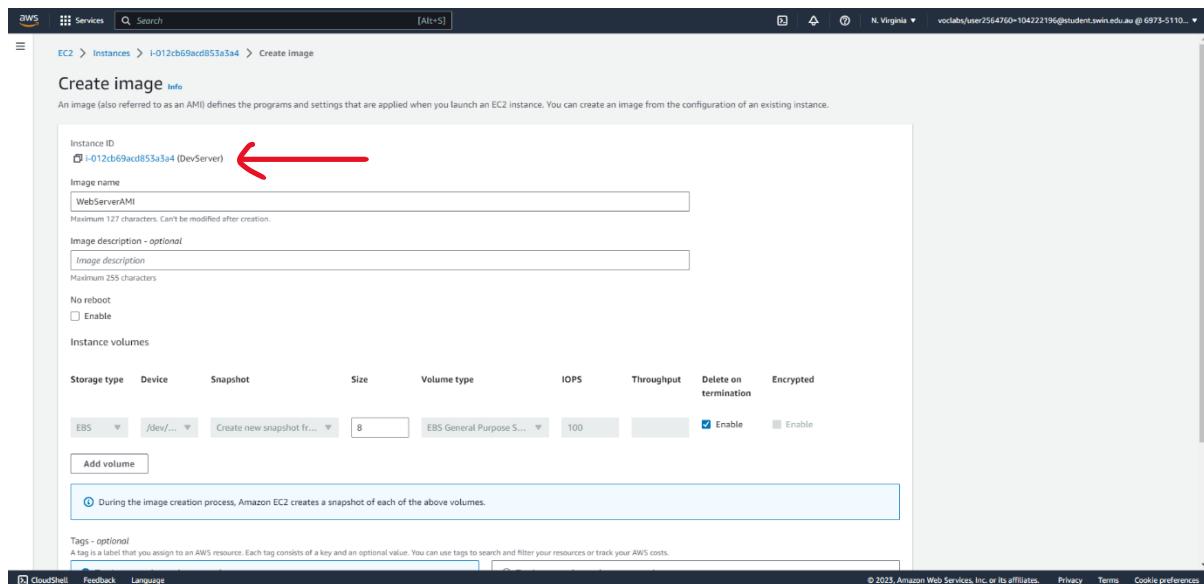


Fig. 29: Create an AMI from the Dev Server instance.

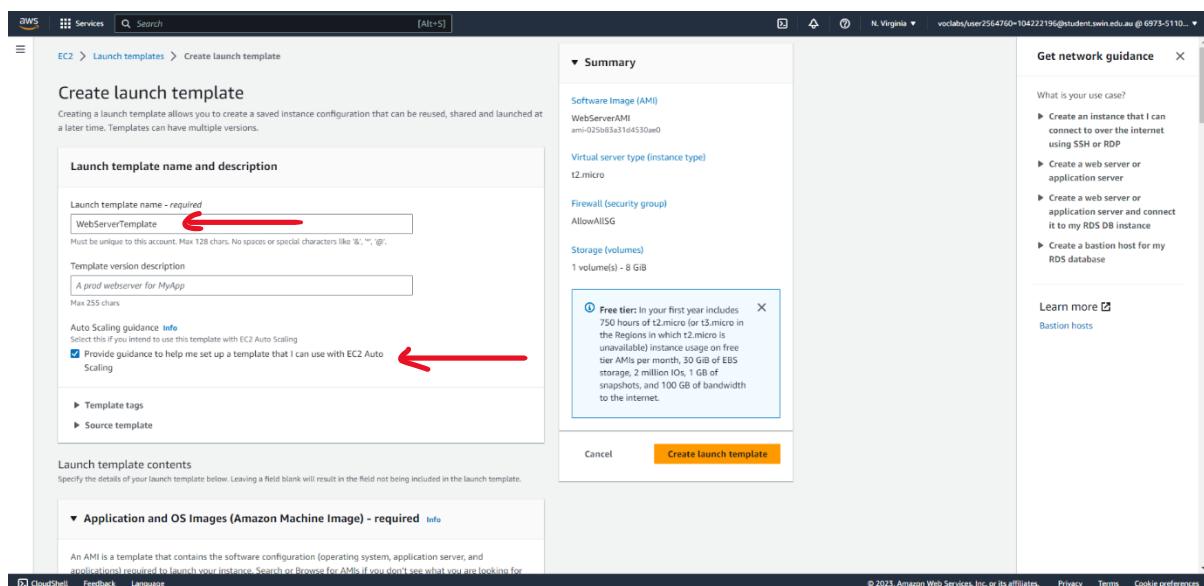


Fig. 30: Create a launch template using the Dev Server AMI (part 1/3), specifying the name and selecting “provide guidance for auto-scaling”.

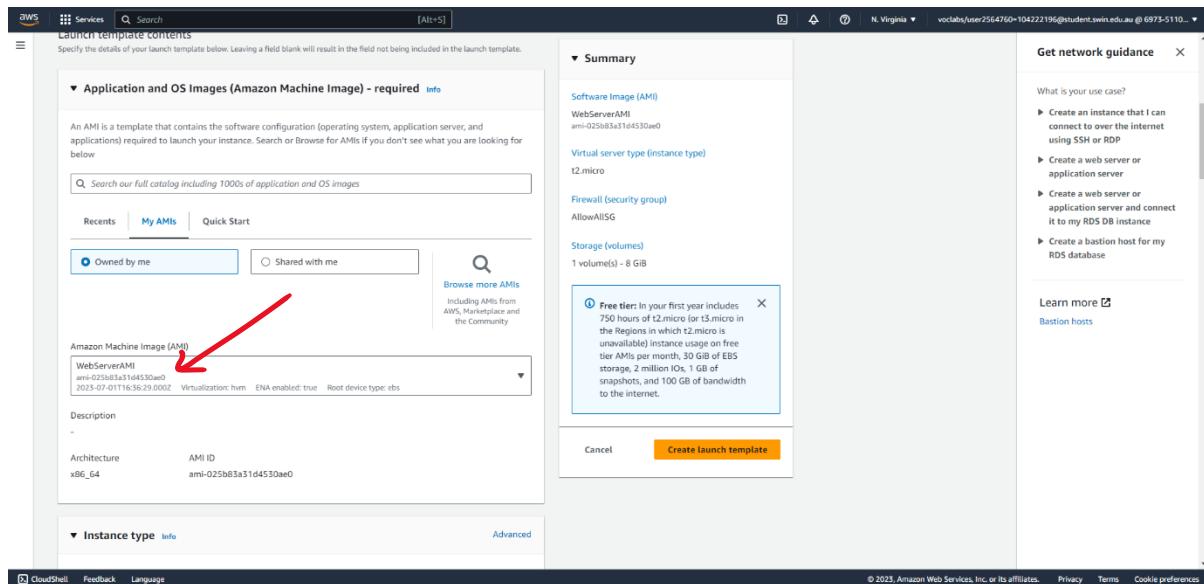


Fig. 31: Create a launch template using the Dev Server AMI (part 2/3), selecting the correct AMI.

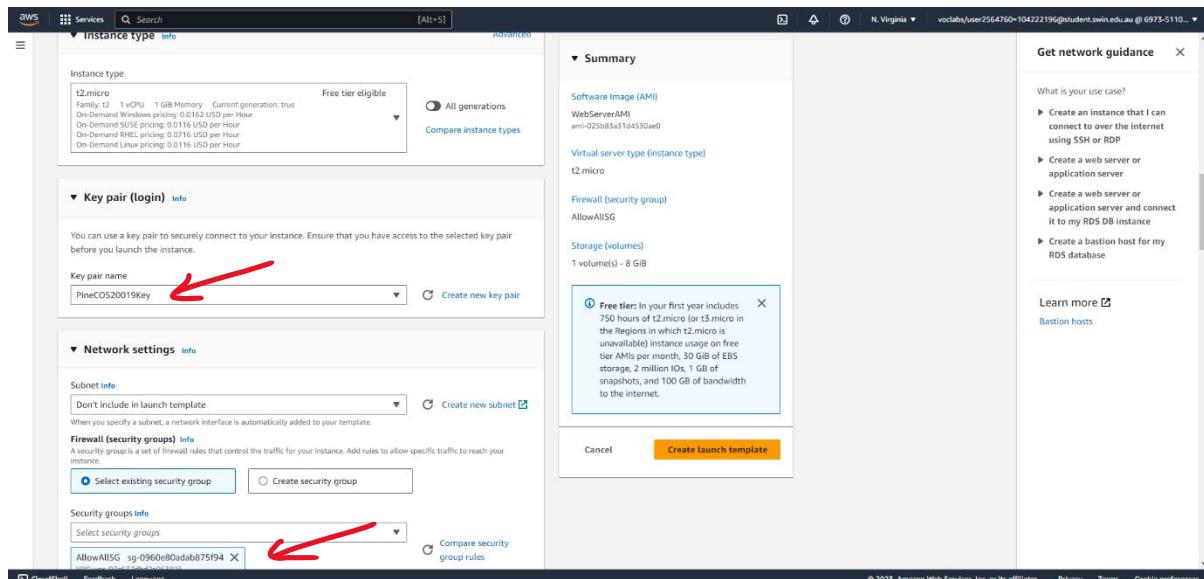


Fig. 32: Create a launch template using the Dev Server AMI (part 3/3), selecting the key pair and security group (which currently allows all traffic in and out; however, the correct security group will be assigned later, resulting in a new version of the launch template.)

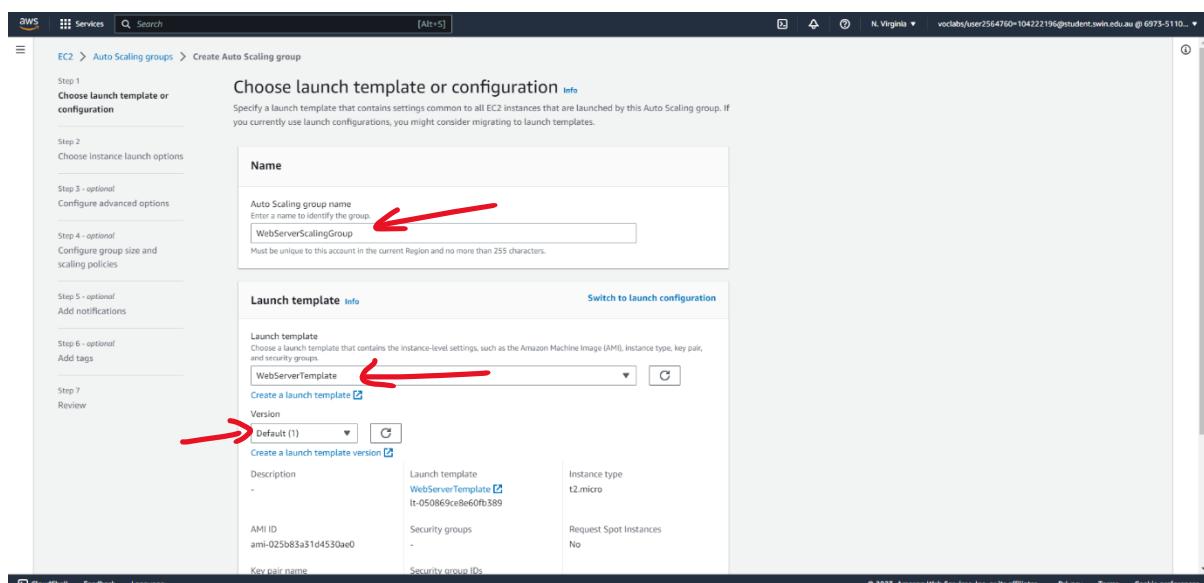


Fig. 32: Create an auto scaling group from the launch template (part 1/5): giving a name, selecting the correct launch template (Currently the ASG uses the version of the launch template without the correct security group. This will be corrected later.)

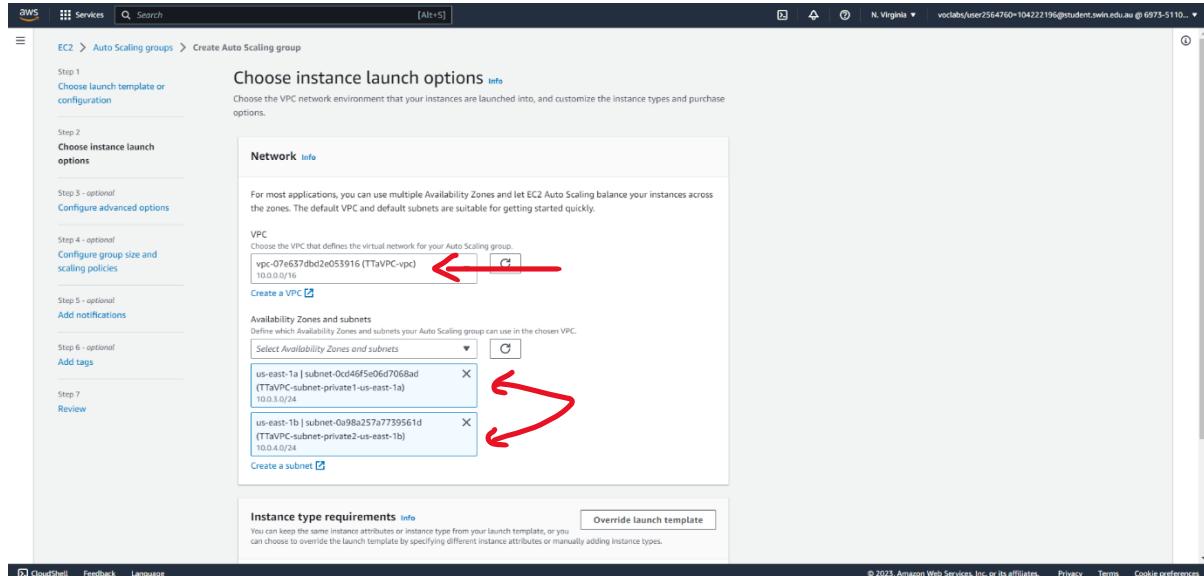


Fig. 33: Create an auto scaling group from the launch template (part 2/5): selecting the correct VPC and private subnets.

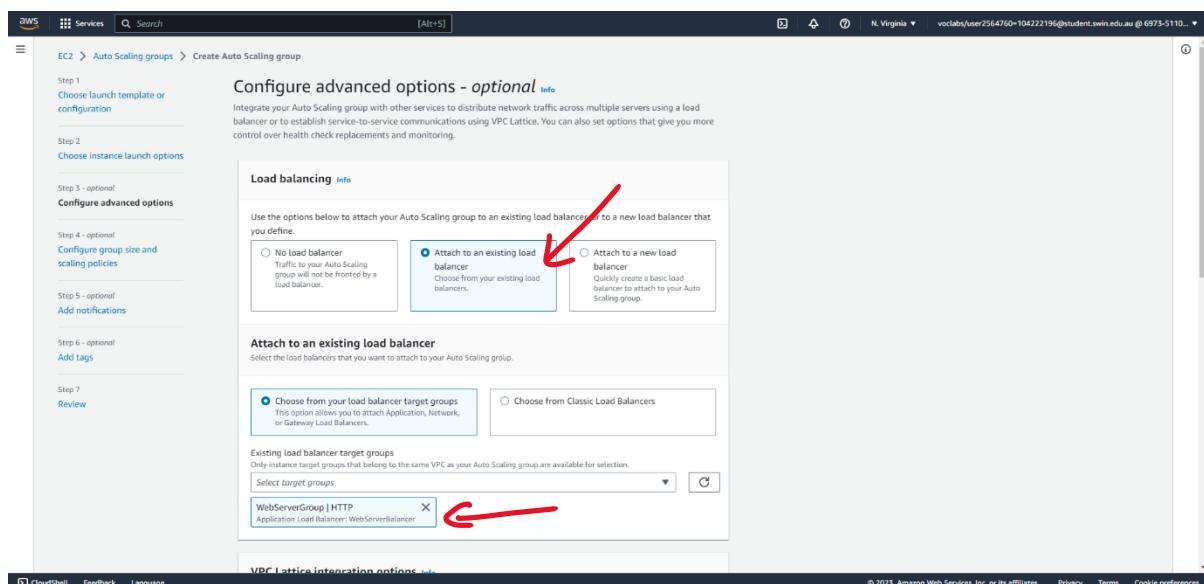


Fig. 34: Create an auto scaling group from the launch template (part 3/5): attaching it to the Web Server target group created earlier.

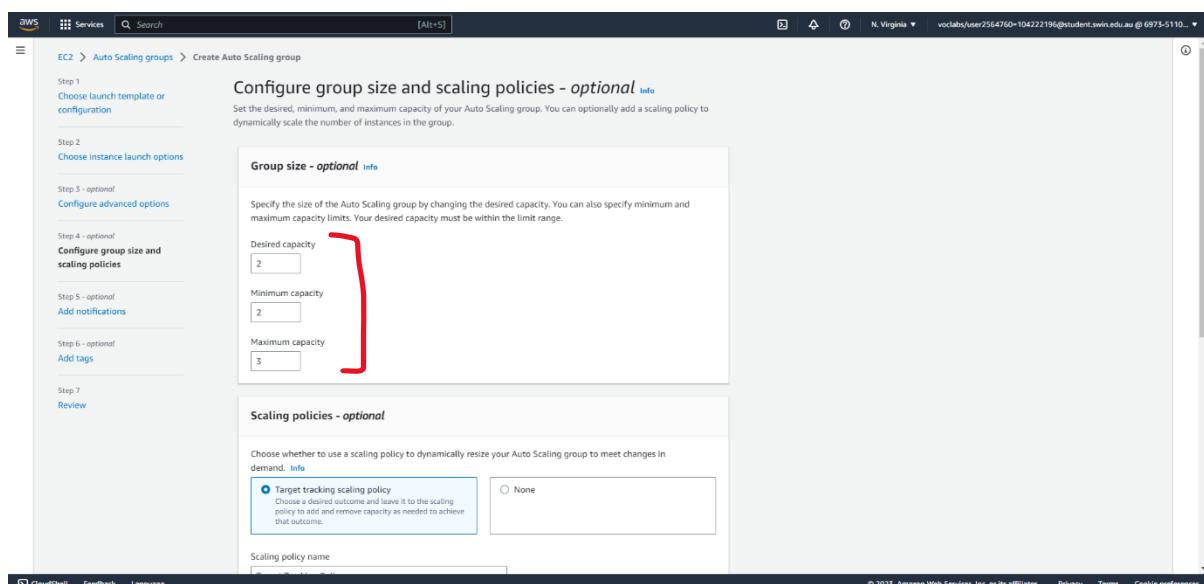


Fig. 35: Create an auto-scaling group from the launch template (part 4/5): setting the group size.

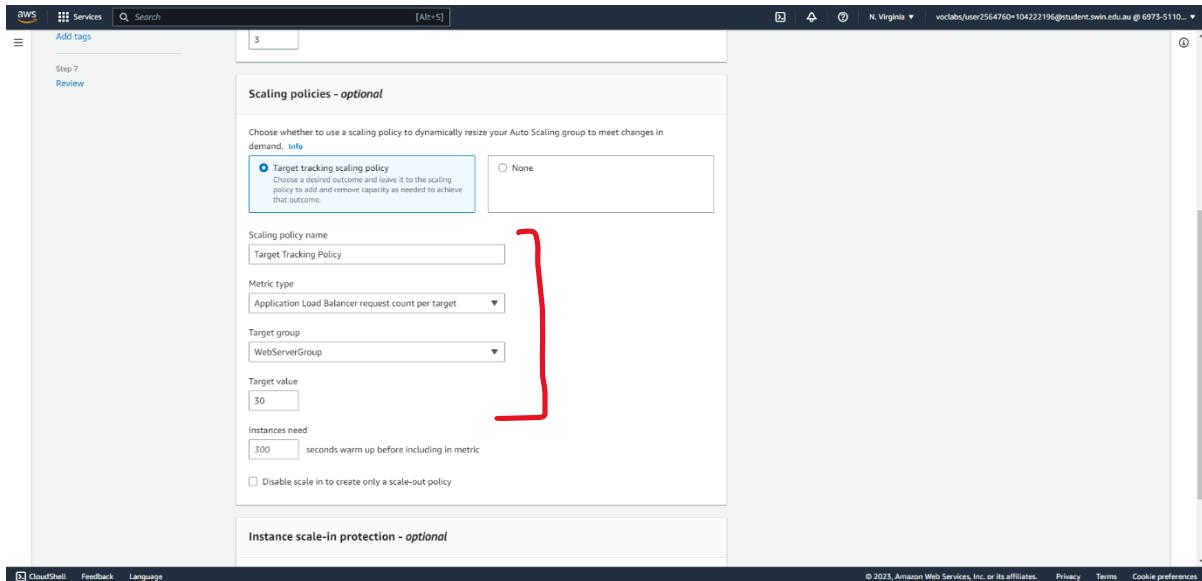


Fig. 36: Create an auto scaling group from the launch template (part 5/5): defining a scaling policy to keep the request count per target at 30.

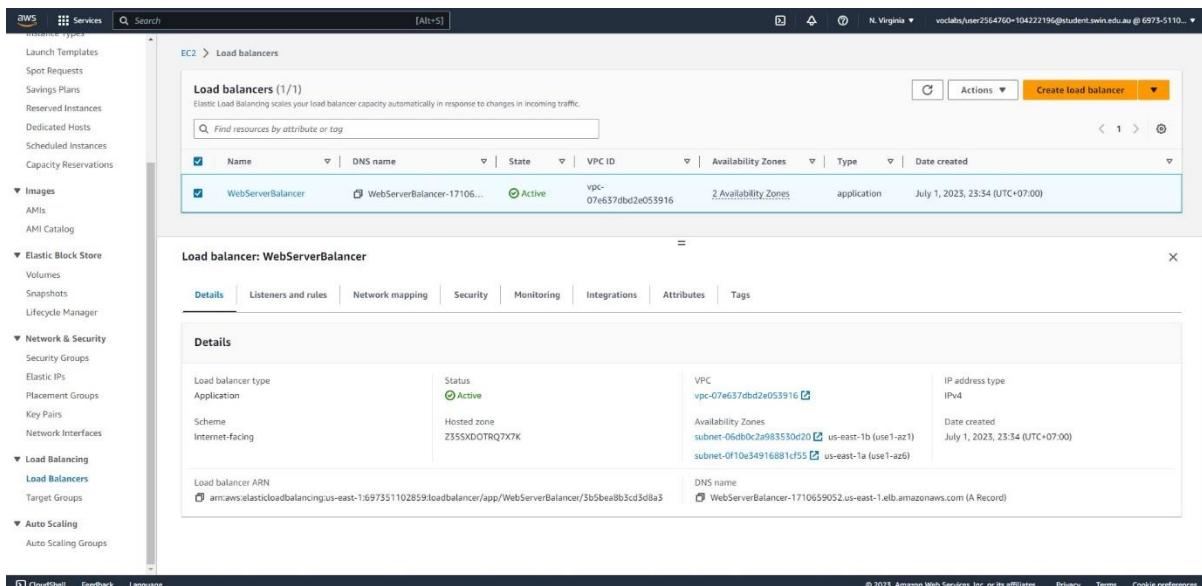


Fig. 37: The load balancer successfully created.

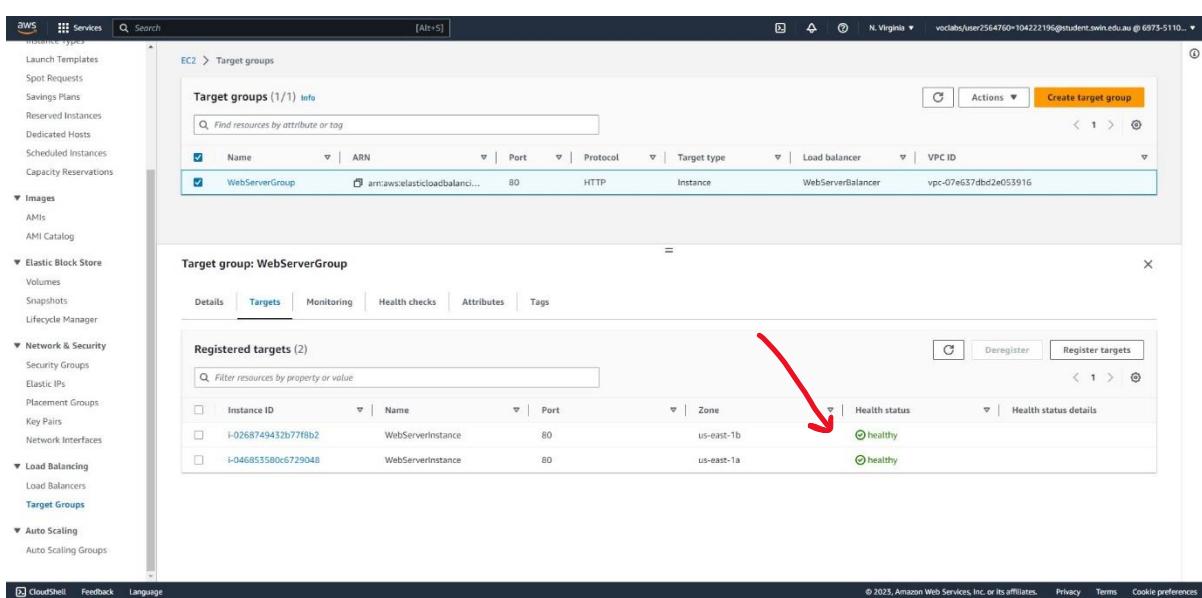


Fig. 38: The target group successfully created with two healthy instances launched by the ASG.

The screenshot shows the AWS Management Console with the URL [https://console.aws.amazon.com/ec2/v2/home?#AutoScalingGroups:group/WebServerScalingGroup](#). The left sidebar is collapsed. The main content area displays the 'Auto Scaling groups' page with one item listed: 'WebServerScalingGroup'. A red arrow points from the text 'two healthy instances' in the caption to the 'Health status' column in the 'Instances' table, which shows two entries both marked as 'Healthy'.

Fig. 39: The auto scaling group successfully created with two healthy instances.

## VII - CREATE SECURITY GROUPS, NACLs, AND SPECIFY BUCKET POLICIES

This section lists the steps needed to create the security groups and NACLs, which fulfils requirements 2.8 and 2.9 in the instructions. This section also completes the bucket policy to block access to photos outside the website.

### A – CREATE SECURITY GROUPS

The screenshot shows the 'Create security group' page under 'EC2 > Security Groups'. The 'Basic details' section has 'Security group name' set to 'ELBSG' and 'Description' set to 'Security group for Elastic Load Balancer.'. The 'VPC' dropdown shows 'vpc-07e637dbd2e053916'. The 'Inbound rules' section contains two rules: one for 'HTTP' (Protocol TCP, Port range 80, Source Anywhere..., Description optional) and one for 'HTTPS' (Protocol TCP, Port range 443, Source Anywhere..., Description optional). Both rules have a 'Delete' button next to them. A red arrow points from the text 'allowing inbound HTTP(S) traffic from the Internet.' in the caption to the 'Source' dropdowns, which show 'Anywhere...'.

Fig. 40: Create the security group for the ELB allowing inbound HTTP(S) traffic from the Internet.

EC2 > Load balancers

**Load balancers (1/1)**

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Name	DNS name	State	VPC ID	Availability Zones	Type	Date created
WebServerBalancer	WebServerBalancer-1710...	Active	vpc-07e637dbd2e053916	2 Availability Zones	application	July 1, 2023, 23:34 (UTC+07:00)

**Load balancer: WebServerBalancer**

- Details
- Listeners and rules
- Network mapping
- Security**
- Monitoring
- Integrations
- Attributes
- Tags

**Security groups (1)**

A security group is a set of firewall rules that control the traffic to your load balancer.

Security Group ID	Name	Description
sg-0078a532945a56106	ELBSG	Security group for Elastic Load Balancer.

Activate Windows  
Go to Settings to activate Windows.

CloudShell Feedback Language

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Fig. 41: Associate the security group with the ELB.

EC2 > Security Groups > Create security group

**Create security group**

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

**Basic details**

Security group name [Info](#)  
WebServerSG

Name cannot be edited after creation.

Description [Info](#)  
Security group for web servers.

VPC [Info](#)  
vpc-07e637dbd2e053916

**Inbound rules**

Type	Protocol	Port range	Source	Description - optional
HTTP	TCP	80	Custom	Allows inbound HTTP traffic from the ELB.
HTTPS	TCP	443	Custom	Allows inbound HTTPS traffic from the ELB.

Add rule

CloudShell Feedback Language

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Fig. 42: Create a security group for the web server instances allowing all HTTP(S) traffic from the ELB.

N. Virg. vocabs/user2564760+10422196@student.swin.edu.au @ 6973-511...

**Auto Scaling group: WebServerScalingGroup**

**Launch template**

Launch template: It-050869c08e60ff389  
WebServerTemplate

AMI ID: ami-025b83a31d4530ae0

Instance type: t2.micro

Version: Latest

Security group IDs: sg-0f10708ca80ac6e55

Description:

**Security Groups (1/3) Info**

search: sg-0f10708ca80ac6e55

Name	Security group ID	Security group name	VPC ID	Description
sg-004df56f6fd48e047	NATServerSG	vpc-07e637dbd2e053916 ...	Security group for NAT.	
sg-0c4d11051ea24e1	DBServerSG	vpc-07e637dbd2e053916 ...	Security group for RDS.	
<b>sg-0f10708ca80ac6e55</b>	<b>WebServerSG</b>	vpc-07e637dbd2e053916 ...	Security group for web.	

sg-0f10708ca80ac6e55 - WebServerSG

**Details**

Security group name: WebServerSG

Owner: 697351102859

Inbound rules count: 2 Permission entries

Outbound rules count: 1 Permission entry

Run Reachability Analyzer

CloudShell Feedback Language

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Fig. 43: A new version of the launch template has been created to include the new Web Server security group. The ASG has been reconfigured to use the latest version of the launch template, which contains the correct security group. All old instances of the ASG have been terminated and recreated to include this change.

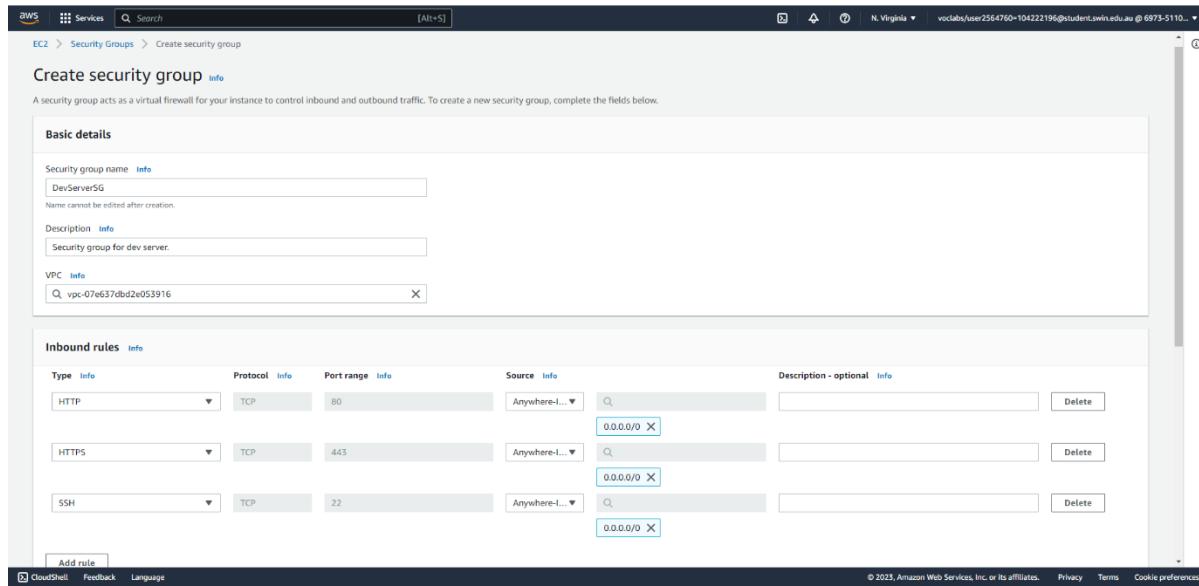


Fig. 44: Create a new security group for the Dev Server allowing all inbound HTTP(S) and SSH traffic from the Internet.

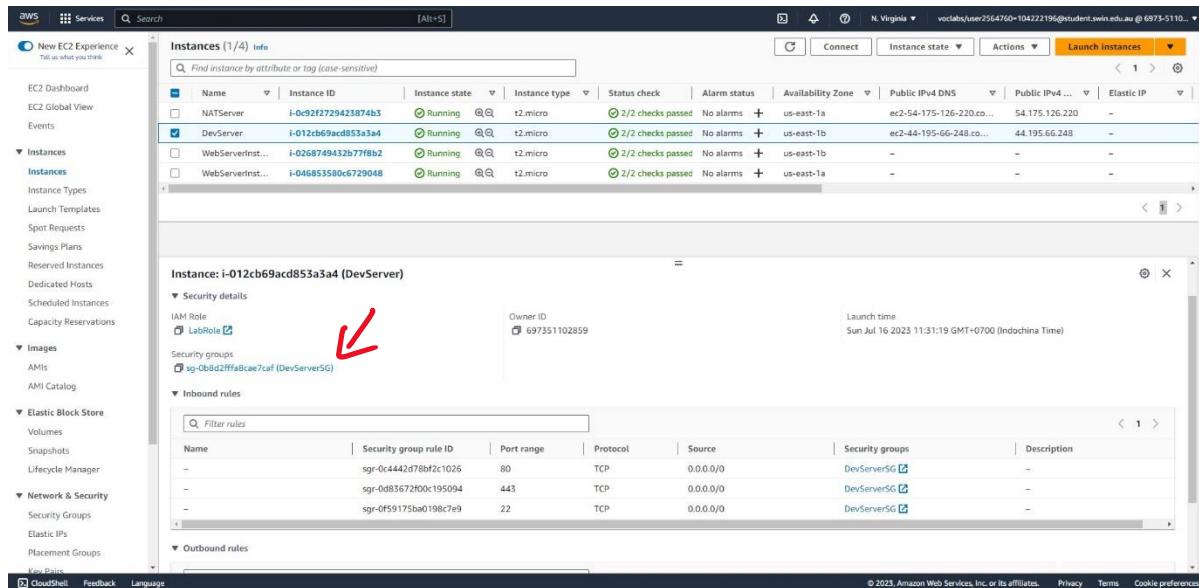


Fig. 45: Assign the security group to the Dev Server.

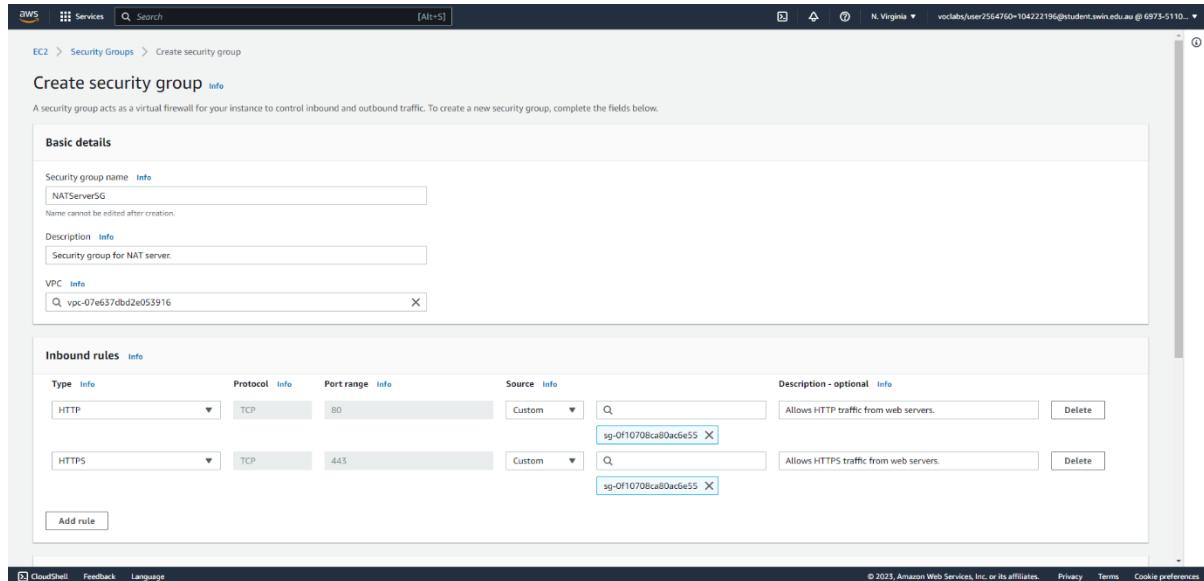


Fig. 46: Create a new security group for the NAT server allowing all inbound HTTP(S) traffic from the web server instances.

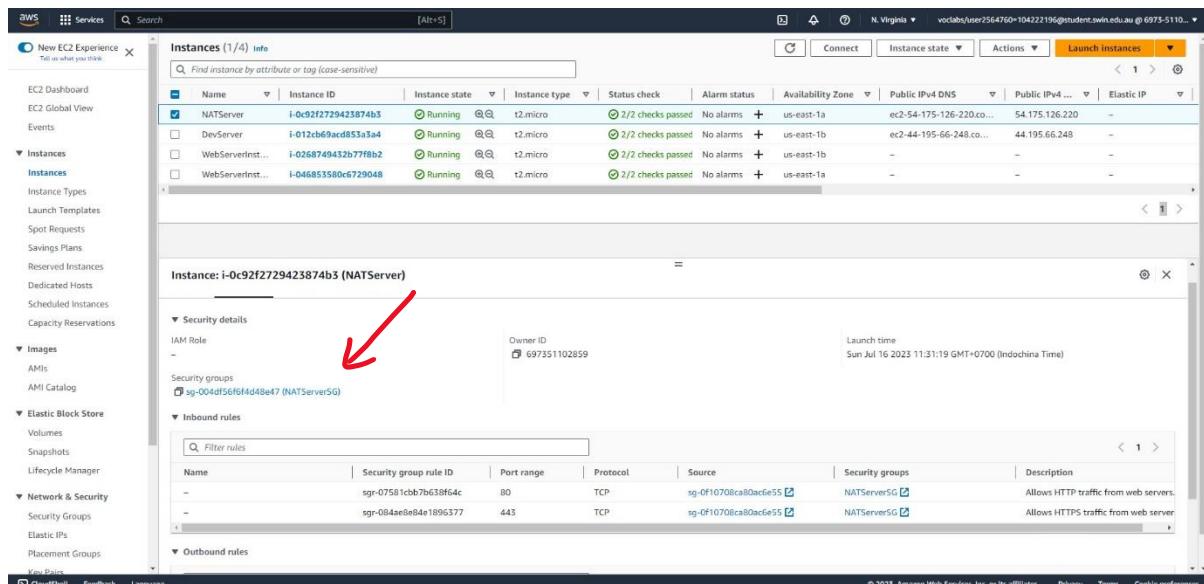


Fig. 47: Assign the security group to the NAT server.

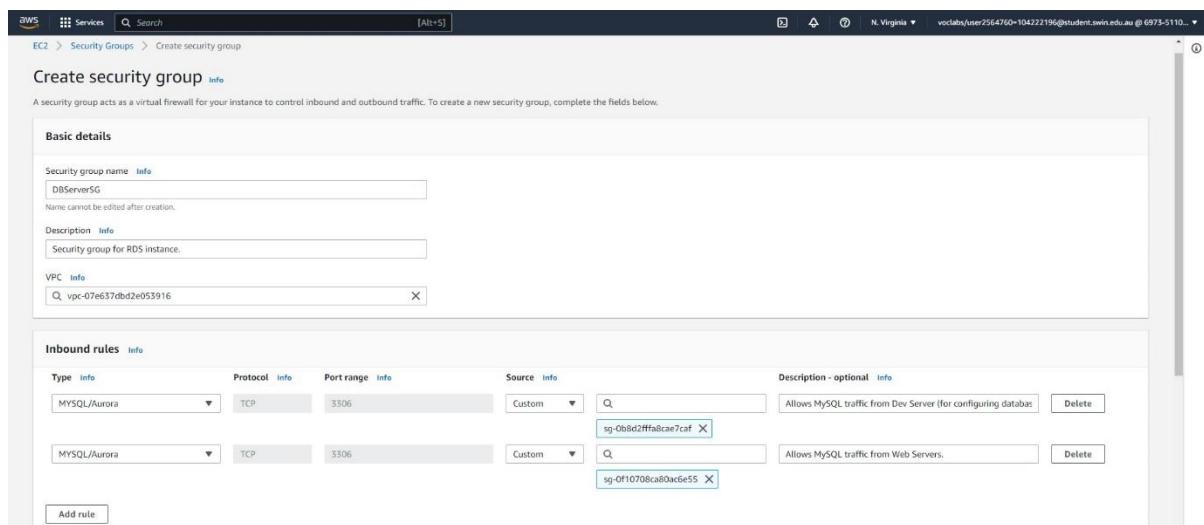


Fig. 48: Create a security group for the database instance allowing inbound MySQL traffic from the Dev and Web server instances.

Fig. 49: Assign the security group to the DB instance.

## B – CREATE THE NACL

Fig. 50: Create a network ACL for the two private subnets.

Fig. 51: Specify the inbound rules for the NACL such that ICMP traffic from 10.0.2.8/32 (the private IP address of the Dev server) is denied.

Rule number	Type	Protocol	Port range	Destination	Allow/Deny
1	All ICMP - IPv4	ICMP (1)	All	10.0.2.8/32	Deny
2	All traffic	All	All	0.0.0.0/0	Allow
	All traffic	All	All	0.0.0.0/0	Deny

Fig. 52: Specify the outbound rules for the NACL such that ICMP traffic from 10.0.2.8/32 (the private IP address of the Dev server) is denied.

Rule number	Type	Protocol	Port range	Destination	Allow/Deny
1	All ICMP - IPv4	ICMP (1)	All	10.0.2.8/32	Deny
2	All traffic	All	All	0.0.0.0/0	Allow
	All traffic	All	All	0.0.0.0/0	Deny

Fig. 53: Associate both private subnets with the NACL.

## C – COMPLETE THE BUCKET POLICY

The screenshot shows the AWS S3 Bucket Policy configuration page. The left sidebar includes options like Buckets, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Block Public Access settings, Storage Lens, Dashboards, AWS Organizations settings, Feature spotlight, and AWS Marketplace for S3. The main content area is titled 'Bucket policy' and contains the following JSON code:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion"
      ],
      "Resource": "arn:aws:s3:::104222196photobucket/*",
      "Condition": {
        "StringLike": {
          "aws:Referer": "http://webserverbalancer-1710659052.us-east-1.elb.amazonaws.com/*"
        }
      }
    }
  ]
}
```

Below the policy is the 'Object Ownership' section, which states that bucket owner enforced and ACLs are disabled. The 'Access control list (ACL)' section is also present. The bottom of the page includes standard AWS footer links and copyright information.

Fig. 54: Configure the S3 bucket policy so that images are only accessible through the photo album website (which is available through the load balancer.)

## VIII – FINAL RESULT

This section shows the final webpage after everything has been correctly configured.

The screenshot shows the final webpage at the URL <http://webserverbalancer-1710659052.us-east-1.elb.amazonaws.com/photoalbum/album.php>. The page displays the following information:

- Student name:** Ta Quang Tung
- Student ID:** 104222196
- Tutorial session:** Saturday 3:00P.M.
- Uploaded photos:**

Under the 'Uploaded photos:' heading, there is a link to [Upload more photos](#).

Photo	Name	Description	Creation date	Keywords
	Golden Hour Album Cover	Golden Hour by Kacey Musgraves	2023-07-16	music, golden hour, kacey musgraves
	Melodrama Album Cover	Melodrama by Lorde	2023-07-16	music, melodrama, lorde

Fig. 55: After uploading two photos, they are correctly shown on the web page.

The screenshot shows the phpMyAdmin interface connected to a MySQL database named 'assignment2'. The 'photos' table is selected, displaying two rows of data:

	id	title	description	date_created	keywords	reference
<input type="checkbox"/>	15	Golden Hour Album Cover	Golden Hour by Kacey Musgraves	2023-07-16	music, golden hour, kacey musgraves	<a href="https://104222196photobucket.s3.amazonaws.com/gold...">https://104222196photobucket.s3.amazonaws.com/gold...</a>
<input type="checkbox"/>	16	Melodrama Album Cover	Melodrama by Lorde	2023-07-16	music, melodrama, lorde	<a href="https://104222196photobucket.s3.amazonaws.com/melo...">https://104222196photobucket.s3.amazonaws.com/melo...</a>

Fig. 56: The photo metadata is correctly stored in the database.

The screenshot shows the AWS S3 console with the bucket '104222196photobucket' selected. The bucket contains four objects:

Name	Type	Last modified	Size	Storage class
golden-hour.png	png	July 16, 2023, 22:31:49 (UTC+07:00)	421.0 KB	Standard
melodrama.png	png	July 16, 2023, 22:32:33 (UTC+07:00)	546.3 KB	Standard
resized-golden-hour.png	png	July 16, 2023, 22:31:53 (UTC+07:00)	125.7 KB	Standard
resized-melodrama.png	png	July 16, 2023, 22:32:35 (UTC+07:00)	151.4 KB	Standard

Fig. 57: The photos and their resized versions are correctly added to the S3 bucket.