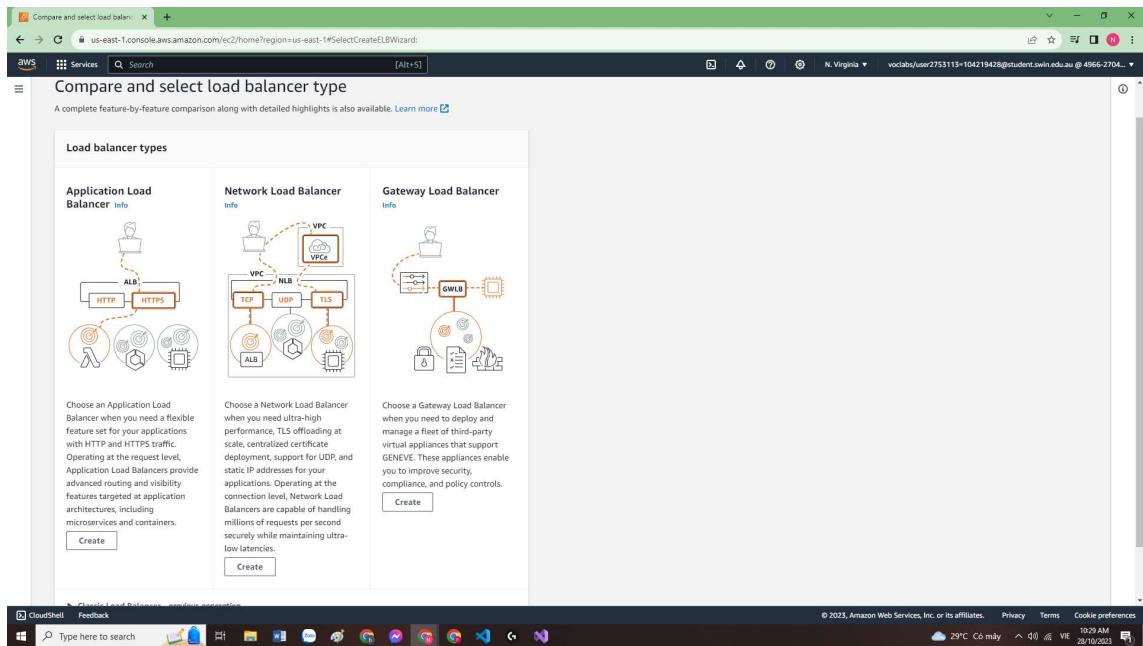
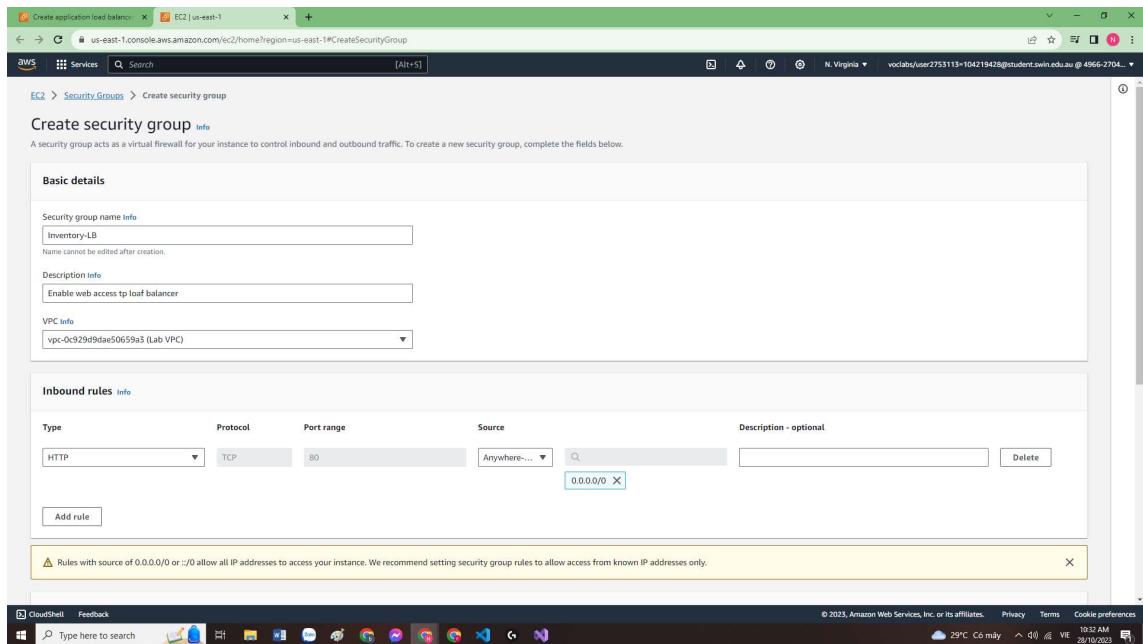


Nguyen Gia Binh – 104219428

## Step 1: Create load Balancer (choose application load balancer)



## 1.2: Create security group



# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudShell interface. A modal window is open, confirming the creation of a security group named 'sg-0c9a8c919a7aeef9e8' with the description 'Inventory-LB'. The 'Inbound rules' tab is selected, showing one rule: 'sg-0c61568b5e8776...' (IPv4) to port 80 (HTTP). The VPC ID is listed as 'vpc-0c929d9dae50659a3'.

## 1.3: Create target group

The screenshot shows the AWS CloudShell interface. A modal window is open, titled 'Step 1 Create target group | EC2 | us-east-1'. It's asking to 'Specify group details'. Under 'Basic configuration', the 'Instances' option is selected, which supports load balancing to instances within a specific VPC. The target group name is set to 'Inventory\_App'. The CloudShell interface at the bottom shows standard Windows icons.

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudFront Create Distribution wizard, Step 1: Set distribution settings. The page title is "Step 1: Set distribution settings". The distribution ID is "d1234567890123456789012345678901". The distribution name is "My First Distribution". The region is "US East (N. Virginia)". The origin is "My First Origin" (IP address 123.45.67.89). The protocol is "HTTP". The SSL certificate is "My First SSL Certificate". The comment is "This is my first distribution". The "Next Step" button is visible at the bottom.

**Step 1: Set distribution settings**

**Distribution ID:** d1234567890123456789012345678901

**Distribution Name:** My First Distribution

**Region:** US East (N. Virginia)

**Origin:** My First Origin (IP address 123.45.67.89)

**Protocol:** HTTP

**SSL Certificate:** My First SSL Certificate

**Comment:** This is my first distribution

**Next Step**

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudFront console with the following steps completed:

- Create application load balancer**: Step 2 Create target group | EC2 | us-east-1
- Step 1 Specify group details**: Target groups > Create target group
- Step 2 Register targets**: Register targets
- Review targets**: Targets (0)
- Target group details | EC2 | us-east-1**: Successfully created target group: Inventory-App.

**Available instances (1)**

Instance ID	Name	State	Security groups	Zone	Private IPv4 address
i-0d350c687870ffe0a	Web Server 1	Running	Inventory-App	us-east-1a	10.0.0.93

**Ports for the selected instances**  
Ports for routing traffic to the selected instances.  
80  
1-65535 (Separate multiple ports with commas)

**Review targets**

Targets (0)
No instances added yet

**Target group details | EC2 | us-east-1**: Successfully created target group: Inventory-App.

**Details**

Target type	Protocol : Port	Protocol version	VPC
Instance	HTTP: 80	HTTP1	vpc-0c929d9da50659a3
IP address type	Load balancer		
IPv4	None associated		

**Targets** | Monitoring | Health checks | Attributes | Tags

**Registered targets (0)**

Instance ID	Name	Port	Zone	Health status	Health status details
No registered targets					

**Actions**

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudShell interface with the following steps:

- Create Application Load Balancer**: The initial step where the user is prompted to enter the load balancer name.
- Basic configuration**: The user selects "IPv4" as the IP address type and "Dualstack" as the scheme.
- Network mapping**: The user maps the load balancer to two subnets: "us-east-1a (use1-az4)" and "us-east-1b (use1-az6)".
- Security groups**: The user selects up to 5 security groups.

The AWS CloudShell interface includes a toolbar with CloudShell, Feedback, and a search bar. The status bar at the bottom shows the date and time as 20/10/2023, 10:42 AM.

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudShell interface with the following steps completed:

- Security groups**: A security group named "Inventory-LB" is selected.
- Listeners and routing**: An HTTP listener on port 80 is configured to forward traffic to a target group named "Inventory-App".
- Add-on services - optional**: The "AWS Global Accelerator" option is checked.
- Summary**: The configuration details are summarized, including:
  - Basic configuration**: Load balancer name "Inventory-LB", facing "Internet-facing" and "IPv4".
  - Security groups**: Associated with VPC "vpc-0c9ac019a7ae99e8" and "Inventory-LB".
  - Network mapping**: Associated with subnets "us-east-1a" (subnet-02649118d45089c) and "us-east-1b" (subnet-068575884400823aa).
  - Listeners and routing**: An HTTP-80 listener is mapped to the "Inventory-App" target group.
  - Add-on services**: AWS Global Accelerator is selected.
  - Tags**: None.
- Attributes**: A note states that certain default attributes will be applied.

A prominent orange "Create load balancer" button is located at the bottom right of the summary screen.

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudShell interface with two tabs open:

- Load balancer created successfully!**: This tab displays a success message: "Successfully created load balancer: Inventory-LB". It includes a note: "Note: It might take a few minutes for your load balancer to be fully set up and ready to route traffic. Targets will also take a few minutes to complete the registration process and pass initial health checks." Below this, there's a "Suggested next steps" section with two items:
  - Review, customize, or configure attributes for your load balancer and listeners using the [Description](#) and [Listeners](#) tabs within [Inventory-LB](#).
  - Discover other services that you can integrate with your load balancer. Visit the [Integrated services](#) tab within [Inventory-LB](#).
- Target group details [EC2 | us-east-1]**: This tab shows the target group details for the newly created load balancer.

Below the tabs, the AWS navigation bar is visible with "Services" selected. The main content area shows the "Create Application Load Balancer" wizard, which has completed the "Create Application Load Balancer" step. A prominent orange "View load balancer" button is at the bottom right.

The screenshot shows the AWS CloudShell interface with the "Load balancers" page open. The left sidebar shows the EC2 navigation menu with "Load Balancers" selected. The main content area displays a table of existing load balancers:

Name	DNS name	State	VPC ID	Availability Zones	Type	Date created
Inventory-LB	Inventory-LB-258909412...	Provisioning...	vpc-0c929d9dae5065...	2 Availability Zones	application	October 28, 2023, 10:42 (UTC+07:00)

At the bottom of the page, a modal window titled "0 load balancers selected" is displayed with the message: "Select a load balancer above."

Nguyen Gia Binh – 104219428

## Step 2: Create Auto Scaling group

The screenshot shows the AWS EC2 Instances details page for instance i-0d330c687870ffe0a. The instance is a t2.micro instance running Amazon Linux (Inferred) with a public IPv4 address of 34.207.235.37. It has a Private IP DNS name of ip-10-0-0-93.ec2.internal and a VPC ID of vpc-0c929d9da50659a3. The instance is currently running and is part of the 'Web Server 1' Auto Scaling group.

**Create image** Info

An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance. You can create an image from the configuration of an existing instance.

**Instance ID:** i-0d330c687870ffe0a (Web Server 1)

**Image name:** Web Server AMI

**Image description (optional):** Lab AMI Web Server

**No reboot:**  Enable

**Instance volumes:**

Storage type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
EBS	/dev...	Create new snapshot fr...	8	EBS General Purpose S...	100		<input checked="" type="checkbox"/> Enable	<input type="checkbox"/> Enable

**Tags - optional:** A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Tag image and snapshots together  
Tag the image and the snapshots with the same tag.

Tag image and snapshots separately  
Tag the image and the snapshots with different tags.

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS EC2 Instances page. A message at the top states: "Currently creating AMI ami-081571629f8cd7845 from instance i-0d330c6b7870ffe0a. Check that the AMI status is Available before deleting the instance or carrying out other actions related to this AMI." Below this, there is a table titled "Instances (1) info" with one row. The row details are:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
Web Server 1	i-0d330c6b7870ffe0a	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a	ec2-34-207-235-37.co...	34.207.235.37	-

On the left sidebar, under the "Instances" section, there is a sub-section titled "Launch Templates" which includes "Launch Templates", "Spot Requests", "Savings Plans", "Reserved Instances", "Dedicated Hosts", and "Capacity Reservations".

## 2.2: Launch template

The screenshot shows the "Create launch template" wizard on the "Summary" step. The summary information is as follows:

- Software Image (AMI): Lab AMI Web Server ami-081571629f8cd7845
- Virtual server type (instance type): t2.micro
- Firewall (security group): Inventory-App
- Storage (volumes): 1 volume(s) - 8 GiB

On the left, there are several configuration sections:

- Launch template name and description:** Launch template name - required: Inventory-LT
- Template version description:** Lab Launch Template
- Auto Scaling guidance:** Select this if you intend to use this template with EC2 Auto Scaling (checkbox checked)
- Provide guidance to help me set up a template that I can use with EC2 Auto Scaling**
- Template tags:**
- Source template:**

At the bottom right of the summary section is a large orange "Create launch template" button.

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudShell interface with two windows open, both titled "Create launch template | EC2 | us-east-1".

**Top Window: Application and OS Images (Amazon Machine Image) - required**

- Software Image (AMI):** Lab AMI Web Server (ami-081571629fbcd7845)
- Virtual server type (instance type):** t2.micro
- Firewall (security group):** Inventory-App
- Storage (volumes):** 1 volume(s) - 8 GiB

**Bottom Window: Instance type**

- Instance type:** t2.micro (Family: t2, 1 vCPU, 1 GiB Memory, Current generation: true)
  - On-Demand Windows base pricing: 0.0116 USD per Hour
  - On-Demand SUSE base pricing: 0.0116 USD per Hour
  - On-Demand RHEL base pricing: 0.0116 USD per Hour
  - On-Demand Linux base pricing: 0.0116 USD per Hour
- Key pair (login):** vokey
- Network settings:**
  - Subnet Info: Don't include in launch template
  - Firewall (security groups): Select existing security group (Inventory-App sg-0636f5fde920183b7)

**Common UI Elements:**

- CloudShell Feedback
- Type here to search
- © 2023, Amazon Web Services, Inc. or its affiliates.
- Privacy Terms Cookie preferences
- 29°C Có mây
- 10:32 AM
- 28/10/2023

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudShell interface with the following details:

**Advanced details (Info)**

- IAM instance profile: [Inventory-App-Role](#) (arn:aws:iam::496617040295:instance-profile/Inventory-App-Role)
- Create new IAM profile
- Hostname type: [Don't include in launch template](#)
- DNS Hostname: [Info](#)
  - Enable resource-based IPv4 (A record) DNS requests
  - Enable resource-based IPv6 (AAAA record) DNS requests
- Instance auto-recovery: [Info](#)
  - Don't include in launch template
- Shutdown behavior: [Info](#)
  - Don't include in launch template
  - Not applicable for EC2 Auto Scaling
- Stop - Hibernate behavior: [Info](#)
  - Don't include in launch template
  - Not applicable for Amazon EC2 Auto Scaling
- Termination protection: [Info](#)
  - Don't include in launch template
- Stop protection: [Info](#)
  - Don't include in launch template
- Detailed CloudWatch monitoring: [Info](#)
  - Enable
- Additional charges apply

**Summary**

- Software Image (AMI): Lab AMI Web Server ami-081516299fd7d7845
- Virtual server type (instance type): t2.micro
- Firewall (security group): Inventory-App
- Storage (volumes): 1 volume(s) - 8 GiB

**Free tier: In your first year includes**

750 hours of t2.micro for t3.micro in the Regions in which t2.micro is unavailable instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the internet.

**Create launch template**

**CloudShell Feedback**

Type here to search

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 29°C Cố mây 10:32 AM 28/10/2023

The screenshot shows the AWS CloudShell interface with the following details:

**Metadata accessible: [Info](#)**

- Don't include in launch template

**Metadata transport: [Info](#)**

- Don't include in launch template

**Metadata version: [Info](#)**

- Don't include in launch template

**Metadata response hop limit: [Info](#)**

- Don't include in launch template

**Allow tags in metadata: [Info](#)**

- Don't include in launch template

**User data - optional: [Info](#)**  
Upload a file with your user data or enter it in the field.

```
#!/bin/bash
# Install Apache Web Server and PHP
yum install -y httpd mysql
amazon-linux-extras install -y php7.2
# Download Lab files
wget https://s3-us-west-2.amazonaws.com/ILT-TF-200-
2023-10-26/Lab/InstallApacheAndPHP-2023-10-26.zip
tar -xvf InstallApacheAndPHP-2023-10-26.zip
cd /var/www/html/
# Download and install the AWS SDK for PHP
wget https://github.com/aws/aws-sdk-php/releases/download/3.62.3/aws.zip
unzip aws-*.zip/www/html/
# Turn on web server
chkconfig httpd on
service httpd start
```

User data has already been base64 encoded

**Summary**

- Software Image (AMI): Lab AMI Web Server ami-081516299fd7d7845
- Virtual server type (instance type): t2.micro
- Firewall (security group): Inventory-App
- Storage (volumes): 1 volume(s) - 8 GiB

**Free tier: In your first year includes**

750 hours of t2.micro for t3.micro in the Regions in which t2.micro is unavailable instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the internet.

**Create launch template**

**CloudShell Feedback**

Type here to search

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 29°C Cố mây 10:32 AM 28/10/2023

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudShell interface with two tabs open: "Create launch template | EC2 | us-east-1" and "Images | EC2 | us-east-1". The main window displays a success message: "Successfully created Inventory-LT (lt-088447a8a266f60cc).". Below this, a "Next steps" section provides options: "Launch an instance", "Launch instance from this template", "Create an Auto Scaling group from your template", and "Create Auto Scaling Fleet". A "Create Spot Fleet" link is also present. At the bottom right is a "View launch templates" button.

**Success**  
Successfully created Inventory-LT (lt-088447a8a266f60cc).

**Next steps**

**Launch an instance**  
With On-Demand Instances, you pay for compute capacity by the second (for Linux, with a minimum of 60 seconds) or by the hour (for all other operating systems) with no long-term commitments or upfront payments. Launch an On-Demand Instance from your launch template.

**Create an Auto Scaling group from your template**  
Amazon EC2 Auto Scaling helps you maintain application availability and allows you to scale your Amazon EC2 capacity up or down automatically according to conditions you define. You can use Auto Scaling to help ensure that you are running your desired number of Amazon EC2 instances during demand spikes to maintain performance and decrease capacity during lulls to reduce costs.

**Create Auto Scaling Fleet**  
A Spot Instance is an unused EC2 instance that is available for less than the On-Demand price. Because Spot Instances enable you to request unused EC2 instances at steep discounts, you can lower your Amazon EC2 costs significantly. The hourly price for a Spot Instance (of each instance type in each Availability Zone) is set by Amazon EC2, and adjusted gradually based on the long-term supply of and demand for Spot Instances. Spot instances are well-suited for data-analysis, batch jobs, background processing, and optional tasks.

**Create Spot Fleet**

**View launch templates**

The screenshot shows the AWS CloudShell interface with the "EC2 | us-east-1" tab active. The main content area displays the "Inventory-LT (lt-088447a8a266f60cc)" launch template details. It includes sections for "Launch template details" (with fields like Launch template ID, Name, Version, and Owner), "Launch template version details" (with a table showing version 1 (Default)), and "Actions" buttons for launching, modifying, or deleting the template. The left sidebar lists various EC2 services and resources.

**Inventory-LT (lt-088447a8a266f60cc)**

**Launch template details**

Launch template ID lt-088447a8a266f60cc	Launch template name Inventory-LT	Default version 1	Owner arn:aws:sts::role/vocababs/edu.au
Actions ▲		Delete template	
Launch instance from template		Modify template (Create new version)	
Delete template version		Set default version	
Manage tags		Create Spot Fleet	
Create Auto Scaling group			

**Launch template version details**

Version 1 (Default)	Description Lab Launch Template	Date created 2023-10-28T03:53:12.000Z	Created by arn:aws:sts::role/vocababs/edu.au
Actions ▾		Delete template version	
Instance details		Advanced details	
AMI ID ami-081571629fbcd7845	Instance type t2.micro	Availability Zone -	Key pair name voockey
Security groups sg-0636f5fde920183b7			

**Actions**

**CloudShell Feedback**

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudShell interface with two windows open, both titled "Create Auto Scaling group | EC2".

**Top Window: Choose launch template**

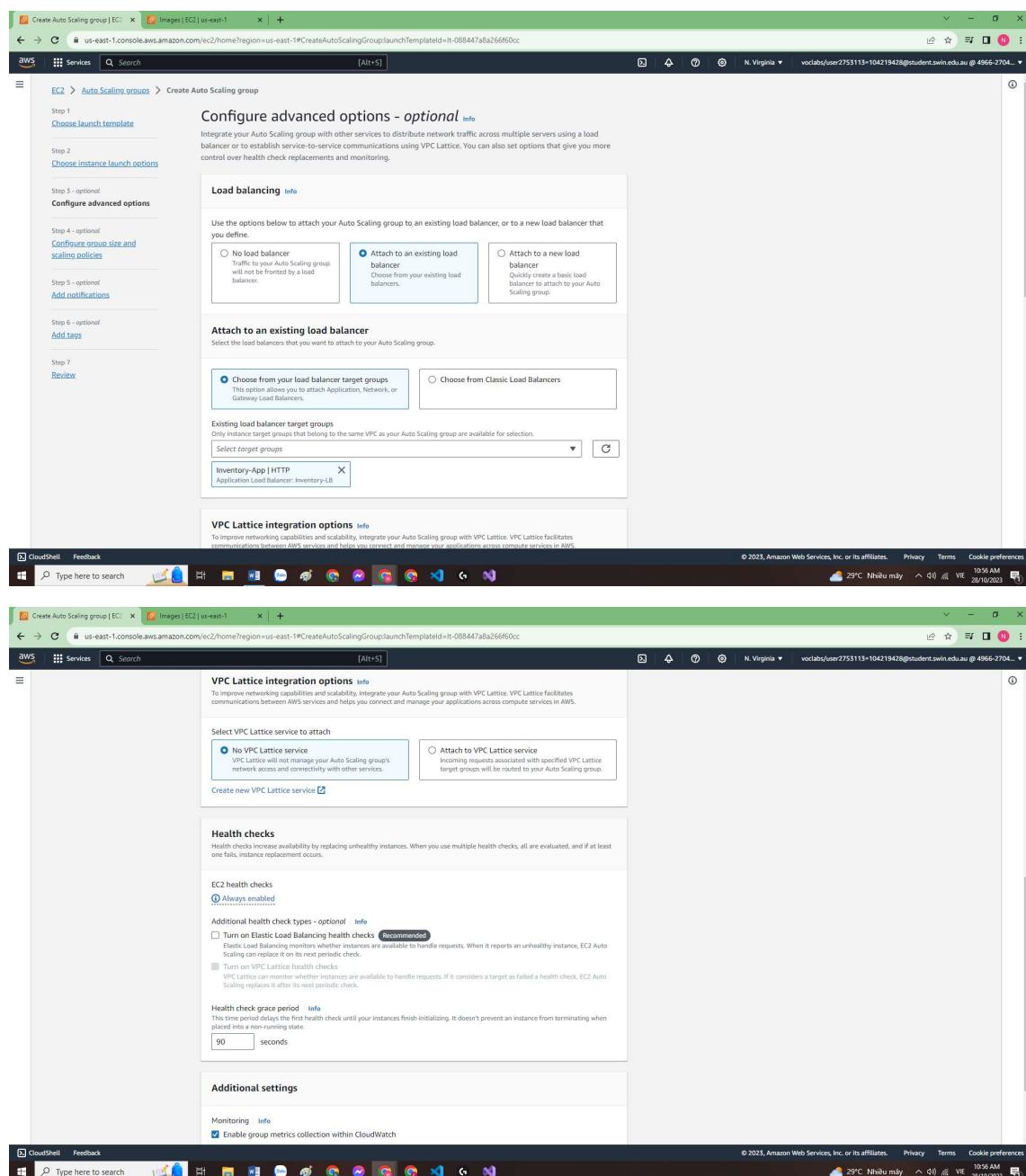
This window is Step 1 of the wizard. It asks to "Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group." A "Name" field contains "Inventory-ASG". Below it, a "Launch template" section shows "Inventory-LT" selected from a dropdown. The "Version" dropdown shows "Default (1)". The "Description" field is "Lab Launch Template". The "Launch template" table shows:

AMI ID	Security groups	Instance type
ami-0815716298cd7845	-	t2.micro

**Bottom Window: Choose instance launch options**

This window is Step 2 of the wizard. It asks to "Choose the VPC network environment that your instances are launched into, and customize the instance types and purchase options." The "Instance type requirements" section shows "Override launch template" is unchecked. The "Launch template" dropdown is set to "Inventory-LT". The "Instance type" dropdown is set to "t2.micro". The "Network" section shows the "VPC" dropdown set to "vpc-0929d9da50659a3 (Lab VPC)". The "Availability Zones and subnets" dropdown shows "us-east-1a | subnet-01f8fcbcc3814081f (Private Subnet 1) 10.0.0.9/23" and "us-east-1b | subnet-06659ea1bfaaf1cf4 (Private Subnet 2) 10.0.4.0/23".

Nguyen Gia Binh – 104219428



# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudShell interface with the following details:

**Create Auto Scaling group | EC2** - Step 4: Configure group size and scaling policies - optional

Set the desired, minimum, and maximum capacity of your Auto Scaling group. You can optionally add a scaling policy to dynamically scale the number of instances in the group.

**Group size - optional**

Specify the size of the Auto Scaling group by changing the desired capacity. You can also specify minimum and maximum capacity limits. Your desired capacity must be within the limit range.

Desired capacity: 2

Minimum capacity: 2

Maximum capacity: 4

**Scaling policies - optional**

Choose whether to use a scaling policy to dynamically resize your Auto Scaling group to meet changes in demand.

Target tracking scaling policy  
Choose a desired outcome and leave it to the scaling policy to add and remove capacity as needed to achieve that outcome.

None

**Instance scale-in protection - optional**

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

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

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

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudShell interface with two windows open:

- Top Window:** A modal dialog titled "Add tags - optional" (Info) is displayed. It contains a note about adding tags to instances and their attached EBS volumes. A table shows a single tag entry: "Name" (Key) and "Inventory-Aspl" (Value - optional). There is also a checkbox for "Tag new instances". Below the table, it says "49 remaining". At the bottom are "Cancel", "Previous", and "Next" buttons.
- Bottom Window:** The "Auto Scaling groups" page. The status bar at the top indicates "Inventory-ASG created successfully. Group metrics collection is enabled." The main table lists one Auto Scaling group:

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
Inventory-ASG	Inventory-LT   Version Default	0	Updating capacity...	2	2	2	us-east-1a, us-east-1b

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS CloudWatch Metrics console. A success message at the top states "Inventory-ASG created successfully. Group metrics collection is enabled." Below this, the "Auto Scaling groups (1) info" section displays a single entry for "Inventory-ASG". The table includes columns for Name, Launch template/configuration, Instances, Status, Desired capacity, Min, Max, and Availability Zones. The "Inventory-ASG" row shows "Inventory-UT" as the launch template, "Version Default" as the configuration, 2 instances, and a status of "-". The desired capacity is 2, with min and max values also set to 2. The availability zones listed are us-east-1a and us-east-1b.

## Step 3: Update security group

The screenshot shows the AWS Security Groups console. It is navigating to "Edit inbound rules" for a security group named "sg-0636f5fde920183b7 - Inventory-App". The "Edit inbound rules" page has a header "Inbound rules info" and a sub-header "Edit inbound rules". It lists one rule: "HTTP" on port 80 from "Traffic from load balancer" (source IP sg-0c9bc919a7ae9e8). There is a "Delete" button next to the source IP. At the bottom, there are "Add rule", "Cancel", "Preview changes", and "Save rules" buttons.

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS Cloud Console interface. The main title bar indicates the region is N. Virginia and the user is vocabs/user2753113-104219428@student.swin.edu.au @ 4966-2704... . The browser tab shows 'us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#SecurityGroups'. The left sidebar navigation includes EC2 Dashboard, Services, Search, and a list of AWS services like Instances, Images, Elastic Block Store, Network & Security, Load Balancing, and more. The main content area displays a success message: 'Inbound security group rules successfully modified on security group [sg-0636f5fde920183b7 | Inventory-App]'. Below this, the 'sg-0636f5fde920183b7 - Inventory-App' security group details are shown, including its name, ID, owner, and VPC ID. The 'Inbound rules' tab is selected, showing a table with columns: Name, Security group rule..., IP version, Type, Protocol, Port range, Source, and Description. A single rule is listed: 'sg-0636f5fde920183b7' with a description 'traffic from application servers'. At the bottom right of the main window are buttons for 'Cancel', 'Preview changes', and 'Save rules'.

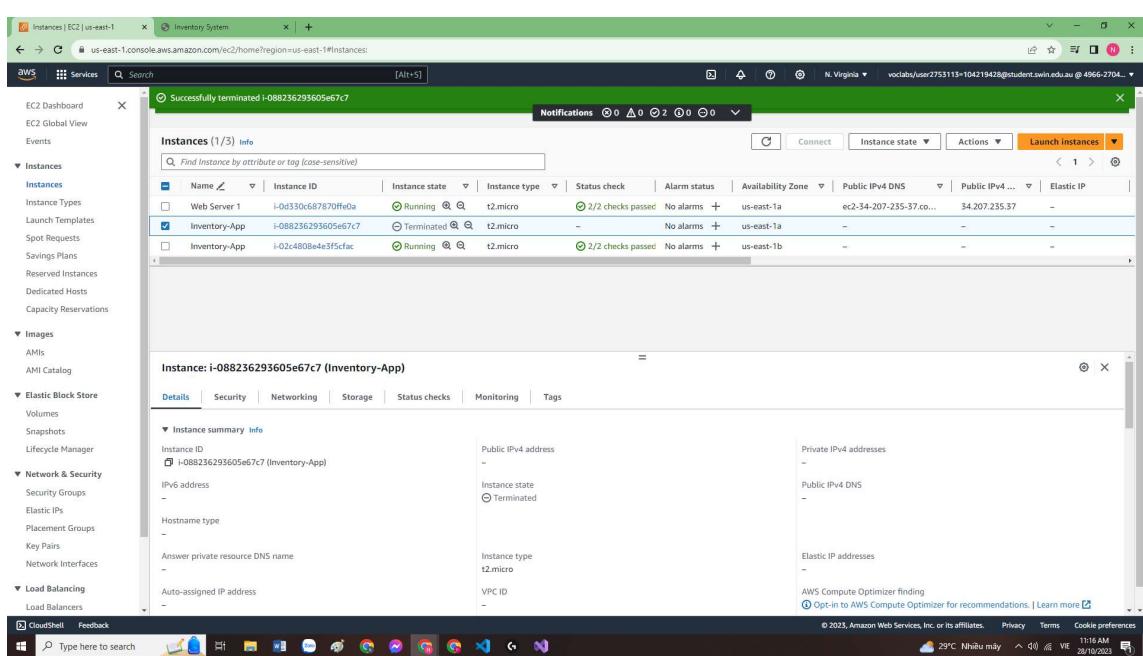
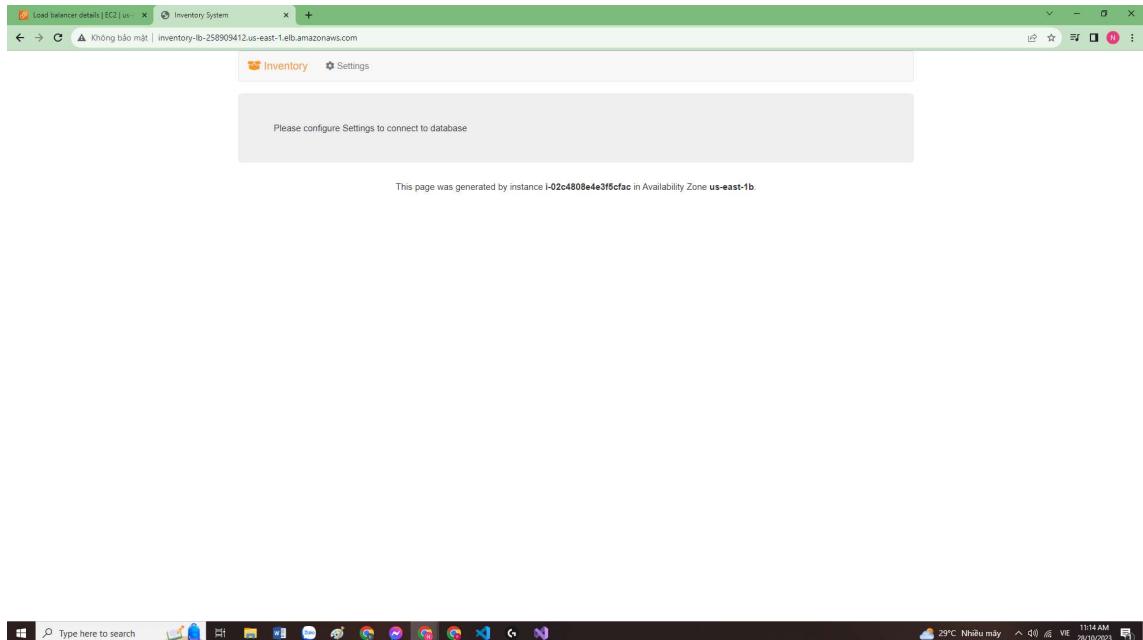
# Nguyen Gia Binh – 104219428

The screenshot shows the AWS EC2 Security Groups console. A success message at the top states: "Inbound security group rules successfully modified on security group [sg-0c1ba9a9cfac5f1b3] Inventory-DB". The main view displays the "Details" for the security group "Inventory-DB". It includes fields for Security group name (Inventory-DB), Security group ID (sg-0c1ba9a9cfac5f1b3), Description (Enable access to MySQL), Owner (496627040295), and VPC ID (vpc-0c929d9da50659a5). The Inbound rules section shows one rule: "sgr-0aa57ac8799621c4a" (Type: MySQL/Aurora, Protocol: TCP, Port range: 3306, Source: sg-0636f5fde920183b...), with a note "traffic from application". The browser status bar indicates the URL is us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#SecurityGroups&group\_id=sg-0c1ba9a9cfac5f1b3.

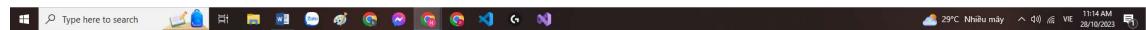
## Step 4: Testing

The screenshot shows the AWS EC2 Target Groups console. The target group "Inventory-App" is selected. The "Details" section shows the target type as "Instance", protocol as "HTTP: 80", and protocol version as "HTTP1". The VPC is "vpc-0c929d9da50659a5". The "Registered targets" table lists two healthy instances: "i-02c4808ede43f5cfac" and "i-088236293605e67c7", both with port 80 and zone us-east-1b. The browser status bar indicates the URL is us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#TargetGroups&target\_group\_arn=arn:aws:elasticloadbalancing:us-east-1:1496627040295:targetgroup/Inventory-App/b53c561c28b09db2.

Nguyen Gia Binh – 104219428



# Nguyen Gia Binh – 104219428



The screenshot shows the AWS EC2 Instances page for the "us-east-1" region. The left sidebar includes links for EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces, Load Balancing, Load Balancers, CloudShell, and Feedback. The main content area displays a table titled "Instances (4) Info" with the following data:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
Web Server 1	i-0d330x667870ffe0a	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1a	ec2-34-207-235-37.co...	34.207.235.37	-
Inventory-App	i-088236293605e67c7	Terminated	t2.micro	-	No alarms	+ us-east-1a	-	-	-
Inventory-App	i-0e56d9cb0f767a575	Pending	t2.micro	-	No alarms	+ us-east-1a	-	-	-
Inventory-App	i-024808e4e395fac	Running	t2.micro	2/2 checks passed	No alarms	+ us-east-1b	-	-	-

A modal window titled "Select an instance" is open, prompting the user to choose an instance from the list.

Nguyen Gia Binh – 104219428

## Step 5: making the database highly available

The image consists of three vertically stacked screenshots of the AWS RDS (Relational Database Service) console, specifically for modifying a DB instance named "inventory-db".

**Screenshot 1: Settings Tab**

This screenshot shows the "Settings" tab of the "Modify DB instance" configuration. It includes fields for:

- DB engine version:** Set to 8.0.33.
- DB instance identifier:** Set to "inventory-db".
- Manage master credentials in AWS Secrets Manager:** An unchecked checkbox with a note that some features won't be supported if managed this way.
- Auto generate a password:** An unchecked checkbox with a note that Amazon RDS can generate a password or you can specify your own.
- New master password:** A field containing "Inventory".
- Confirm master password:** A field containing "Inventory".

**Screenshot 2: Instance Configuration Tab**

This screenshot shows the "Instance configuration" tab. It includes sections for:

- DB instance class:** Set to "Burstable classes (includes t classes)" and "db.t5.small".
- Storage:** Set to "General Purpose SSD (gp2)".
- Allocated storage:** Set to 10 GiB.

**Note:** There are two informational notes in this section:

- Provisioning less than 100 GiB of General Purpose (SSD) storage for high throughput workloads could result in higher latencies upon exhaustion of the initial General Purpose (SSD) IO credit balance. [Learn more](#)
- After you modify the storage for a DB instance, the status of the DB instance will be in storage-optimization. Your instance will remain available as the storage-optimization operation completes. [Learn more](#)

**Screenshot 3: Final Review and Next Step**

This screenshot shows the final review step before saving the changes. It displays the modified settings and provides a "Save changes" button at the bottom.

# Nguyen Gia Binh – 104219428

The screenshot shows the 'Modify DB instance: inventory-db' page in the AWS RDS console. The 'Summary of modifications' section lists changes: 'DB instance class' from 'db.t3.micro' to 'db.t3.small', 'Multi-AZ deployment' from 'No' to 'Yes', and 'Allocated storage' from '5 GiB' to '10 GiB'. The 'Schedule modifications' section has 'Apply immediately' selected. A warning box notes potential performance impact during conversion to Multi-AZ. At the bottom are 'Cancel', 'Back', and 'Modify DB instance' buttons.

**Summary of modifications**

Attribute	Current value	New value
DB instance class	db.t3.micro	db.t3.small
Multi-AZ deployment	No	Yes
Allocated storage	5 GiB	10 GiB

**Schedule modifications**

When to apply modifications:

- Apply during the next scheduled maintenance window  
Current maintenance window: November 03, 2023 17:21 - 17:51 UTC+7
- Apply immediately  
Any modifications in this request and any pending modifications will be asynchronously applied as soon as possible, regardless of the maintenance window setting for this database instance.

**Potential performance impact when converting to Multi-AZ**

Your DB instance can experience a significant performance impact during and after converting to a Multi-AZ deployment. The impact is greater on DB instances with large amounts of storage and write-intensive workloads. We don't recommend this conversion on a production DB instance.

Cancel Back **Modify DB instance**

# Nguyen Gia Binh – 104219428

The screenshot shows the AWS RDS console with the 'Inventory System' database selected. The database is listed in the 'Databases' table with the following details:

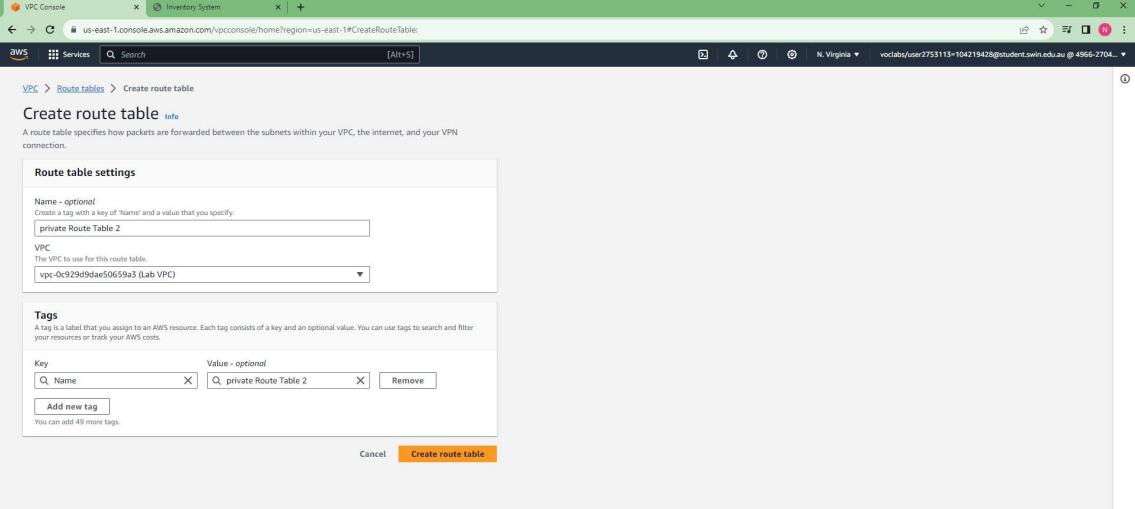
DB identifier	Status	Role	Engine	Region & AZ	Size	Actions	CPU	Current activity	Maintenance	VPC
inventory-db	Available	Instance	MySQL Community	us-east-1a	db.t3.micro	4 Actions	3.09%	0 Connections	none	vpc-0c929d9dae50655

## Step 6: Configure for a highly available NAT gateway

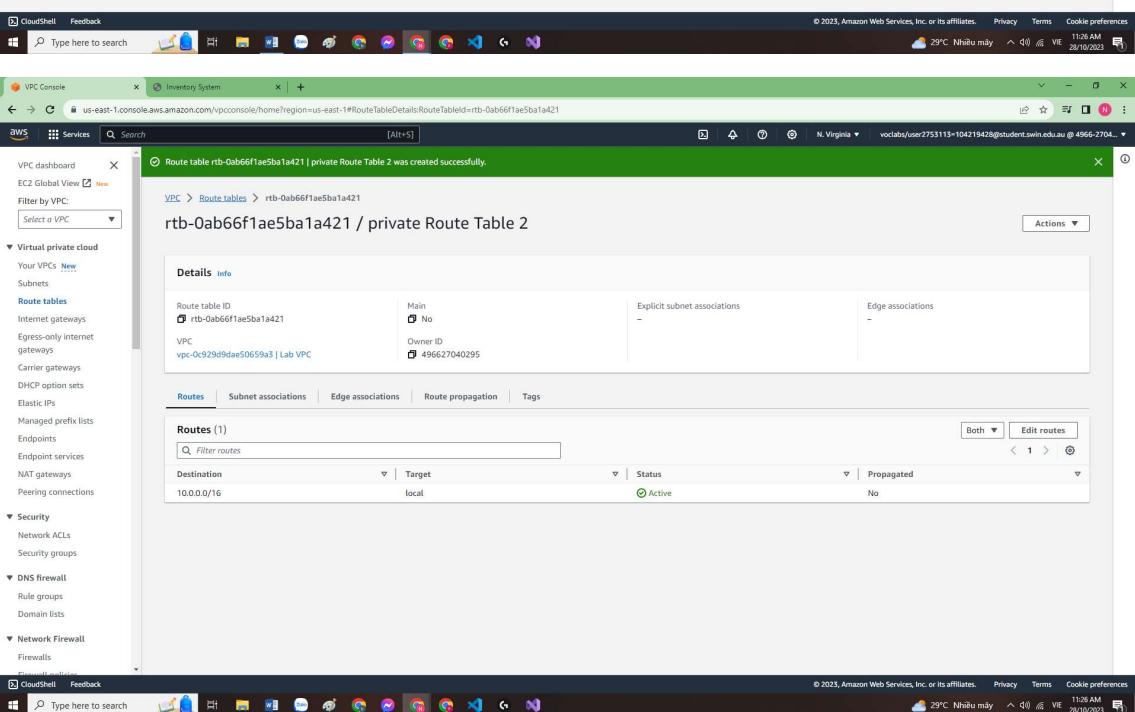
The screenshot shows the AWS VPC Console with the 'Create NAT gateway' wizard. The 'NAT gateway settings' step is displayed, with the following configuration:

- Name - optional:** my-nat-gateway-01
- Subnet:** subnet-068652588460823a6 [Public Subnet 2]
- Connectivity type:** Public (radio button selected)
- Elastic IP allocation ID:** epaalloc-0dfbcfa84d1e5e5984
- Tags:** No tags associated with the resource.

# Nguyen Gia Binh – 104219428



The screenshot shows the AWS VPC Console interface for creating a new Route Table. The 'Route table settings' section is active, showing a 'Name - optional' field containing 'private Route Table'. A 'VPC' dropdown is set to 'vpc-0c929d9dae50659a3 (Lab VPC)'. The 'Tags' section contains a single tag 'Name: private Route Table 2'. At the bottom right are 'Cancel' and 'Create route table' buttons.

The screenshot shows the AWS VPC Console after a Route Table has been successfully created. The message 'rtb-0ab66f1ae5ba1a421 | private Route Table 2 was created successfully.' is displayed. The 'Details' tab is selected, showing the Route Table ID 'rtb-0ab66f1ae5ba1a421', Main status 'No', and VPC 'vpc-0c929d9dae50659a3 | Lab VPC'. The 'Routes' tab shows one route entry: Destination '10.0.0.0/16' and Target 'local'. The status is 'Active' and 'Propagated' is 'No'.

# Nguyen Gia Binh – 104219428

Screenshot of the AWS VPC Console showing the 'Edit routes' interface for a specific route table.

The 'Edit routes' screen displays the following table:

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	NAT Gateway nat-0b6eed7ef61930186	-	No

Buttons at the bottom include 'Add route', 'Cancel', 'Preview', and 'Save changes'.

Below this, a success message is displayed: "Updated routes for rtb-0ab66f1ae5ba1a421 / private Route Table 2 successfully".

The 'Details' tab shows the following information:

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-0ab66f1ae5ba1a421	No	-	-
VPC	Owner ID 496627040295		

The 'Routes' tab lists two routes:

Destination	Target	Status	Propagated
0.0.0.0/0	nat-0b6eed7ef61930186	Active	No
10.0.0.0/16	local	Active	No

# Nguyen Gia Binh – 104219428

VPC Console > Inventory System > +

us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#editRouteTableSubnetAssociations?RouteTableId=rtb-0ab66f1ae5ba1a421

Updated routes for rtb-0ab66f1ae5ba1a421 / private Route Table 2 successfully

Details

VPC > Route table > rtb-0ab66f1ae5ba1a421 > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/4)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
Public Subnet 2	subnet-06852588450623aa6	10.0.1.0/24	-	rtb-03da5df2a7970d0a5 / Public Route Table
<input checked="" type="checkbox"/> Private Subnet 2	subnet-06659ea1bfaaf1cf4	10.0.4.0/23	-	rtb-041500ae7bed8100f / Private Route Table 1
Private Subnet 1	subnet-01fffbcc3814081f	10.0.2.0/23	-	rtb-041500ae7bed8100f / Private Route Table 1
Public Subnet 1	subnet-0a2849e11bd3069ac	10.0.0.0/24	-	rtb-03da5df2a7970d0a5 / Public Route Table

Selected subnets

subnet-06659ea1bfaaf1cf4 / Private Subnet 2
X

Cancel Save associations

CloudShell Feedback

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

29°C Nha Trang, Vietnam 11:29 AM 28/10/2023

VPC Console > Inventory System > +

us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#RouteTableDetails?RouteTableId=rtb-0ab66f1ae5ba1a421

Updated routes for rtb-0ab66f1ae5ba1a421 / private Route Table 2 successfully

Details

You have successfully updated subnet associations for rtb-0ab66f1ae5ba1a421 / private Route Table 2.

VPC dashboard > EC2 Global View > New

Filter by VPC: Select a VPC

Virtual private cloud

Your VPCs New

Subnets

Route tables

Internet gateways

Egress-only internet gateways

Carrier gateways

DHCP option sets

Elastic IPs

Managed prefix lists

Endpoints

Endpoint services

NAT gateways

Peering connections

Security

Network ACLs

Security groups

DNS firewall

Rule groups

Domain lists

Network Firewall

Firewalls

CloudShell Feedback

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

29°C Nha Trang, Vietnam 11:30 AM 28/10/2023