

# Lab 07

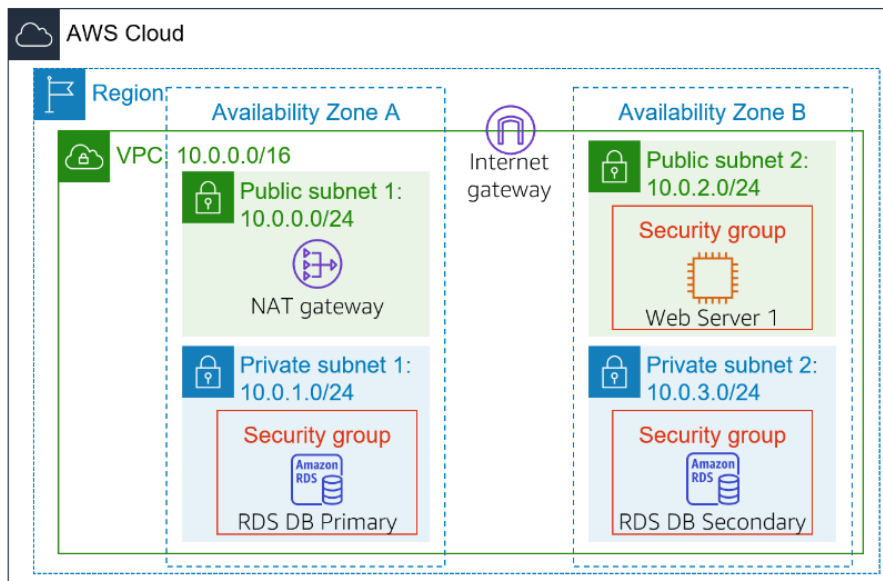
## COS20019

### CLOUD COMPUTING ARCHITECTURE

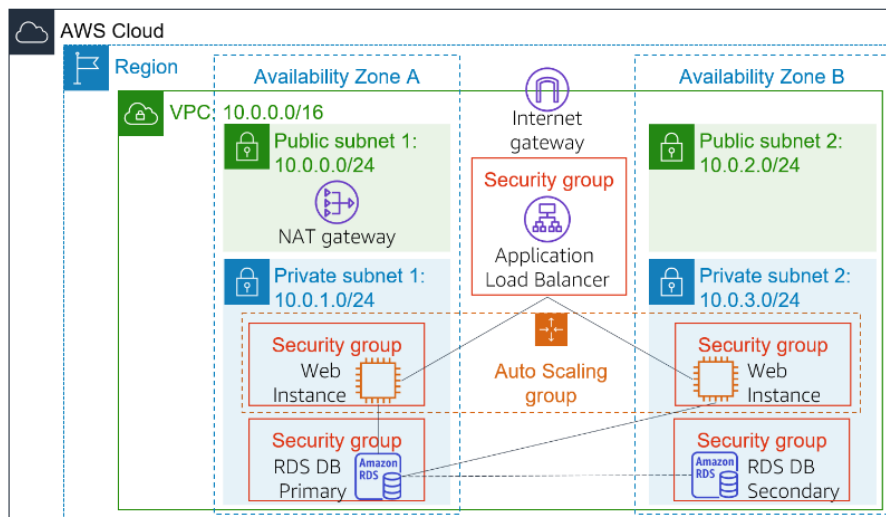
Tran Thanh Minh

#### Scenario

You start with the following infrastructure:



The final state of the infrastructure is:



## Task 1: Create an AMI for auto scaling

### 1. Creating AMI

The screenshot shows the 'Create image' page in the AWS Management Console. The breadcrumb trail is 'EC2 > Instances > i-0ae01d6bc7b5f768 > Create image'. The page title is 'Create image' with an 'info' icon. A sub-header states: '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.' The form includes: 'Instance ID' (i-0ae01d6bc7b5f768 (Web Server 1)), 'Image name' (Web Server 1), 'Image description - optional' (Lab AMI for Web Server), 'No reboot' (checked), 'Enable' (unchecked), and 'Instance volumes'.

## Taks 2: Create a Load Balancer

### 1. Configure target groups base on the instruction from the lab

The screenshot shows the 'Specify group details' page for creating a target group. The breadcrumb trail is 'EC2 > Target groups > Create target group'. The page title is 'Specify group details' with a sub-header: 'Your load balancer routes requests to the targets in a target group and performs health checks on the targets.' The page is divided into two steps: 'Step 1: Specify group details' and 'Step 2: Register targets'. Under 'Basic configuration', 'Choose a target type' has three options: 'Instances' (selected), 'IP addresses', and 'Lambda function'. Below this, 'Target group name' is 'LabGroup'. 'Protocol' is 'HTTP' and 'Port' is '80'. 'VPC' is 'Lab VPC'. A note states: 'A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.'

### 2. Configure network mapping for the Application Load Balancer with the selection of both public subnets

The screenshot shows the 'Network mapping' page for the target group. The breadcrumb trail is 'EC2 > Target groups > LabGroup > Network mapping'. The page title is 'Network mapping' with a sub-header: 'The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.' The 'VPC' is 'Lab VPC'. Under 'Mappings', there are two subnets selected: 'us-east-1a (use 1-az4)' and 'us-east-1b (use 1-az6)'. Each subnet has an 'IPv4 address' assigned by AWS.

### 3. Configure security groups and listeners and routing section for the load balancer

The screenshot shows the AWS Management Console interface for configuring a Load Balancer. The 'Security groups' section is expanded, showing a dropdown menu with 'Web Security Group' selected. Below it, the 'Listeners and routing' section is also expanded, showing a 'Listener HTTP:80' configuration. The 'Protocol' is set to 'HTTP', the 'Port' is '80', and the 'Default action' is 'Forward to LabGroup'. The 'Forward to' dropdown is set to 'LabGroup', and the 'Target type' is 'Instance, IP'. The 'HTTP' dropdown is set to 'HTTP'. The 'Add listener tag' button is visible, along with a note that you can add up to 50 more tags.

## Task 3: Create a Launch Template and Auto scaling group

### 1. Create launch template

The screenshot shows the 'Create launch template' wizard in the AWS Management Console. The 'Launch template name and description' section is filled out with 'LabConfig' as the name and 'A prod webserver for HP-App' as the description. The 'Auto Scaling guidance' section has 'Provide guidance to help me set up a template that I can use with EC2 Auto Scaling' selected. The 'Template tags' and 'Source template' sections are empty. The 'Summary' section on the right shows the configuration: 'Software Image (AMI)' is 'Lab AMI for Web Server', 'Virtual server type (instance type)' is 't2.micro', 'Firewall (security group)' is 'Web Security Group', and 'Storage (volumes)' is '1 volume(s) - 8 GB'. A 'Free tier' notification is displayed, stating that the first year includes 750 hours of t2.micro (or t3.micro in the Region in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GB of EBS storage, 1 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. The 'Create launch template' button is highlighted in orange.

### 2. Configure Application and OS images and instance type sections

The screenshot shows the 'Launch template contents' page in the AWS Management Console. The 'Application and OS Images (Amazon Machine Image) - required' section is expanded, showing a search bar and a list of AMIs. The 'Web Server 1' AMI is selected. The 'Instance type' section is also expanded, showing a list of instance types. The 't2.micro' instance type is selected. The 'Summary' section on the right shows the configuration: 'Software Image (AMI)' is 'Lab AMI for Web Server', 'Virtual server type (instance type)' is 't2.micro', 'Firewall (security group)' is 'Web Security Group', and 'Storage (volumes)' is '1 volume(s) - 8 GB'. A 'Free tier' notification is displayed, stating that the first year includes 750 hours of t2.micro (or t3.micro in the Region in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GB of EBS storage, 1 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet. The 'Create launch template' button is highlighted in orange.

### 3. Configure network setting for the template

The screenshot shows the 'Create new EC2 instance' page in the AWS console, specifically the 'Network settings' tab. The 'Subnet info' section has 'Don't include in launch template' selected. The 'Firewall (security group) info' section has 'Select existing security group' selected, with 'Web Security Group' (sg-020174dbec3f0dc7f) chosen. The 'Summary' panel on the right lists the configuration: Software Image (AMI) 'Lab AMI for Web Server', Virtual server type (instance type) 't2.micro', Firewall (security group) 'Web Security Group', and Storage (volumes) '1 volume(s) - 8 GiB'. A blue callout box indicates that the 'Free tier' includes 750 hours of t2.micro instances per month.

### 4. Enable detailed cloud watch monitoring for allowing auto scaling to react quickly to changing utilization.

The screenshot shows the 'Create new EC2 instance' page in the AWS console, specifically the 'Termination protection' and 'Detailed CloudWatch monitoring' tabs. Both 'Don't include in launch template' and 'Enable' are selected. The 'Summary' panel on the right shows the same configuration as the previous step.

### 5. Create auto scaling group

The screenshot shows the 'Create Auto Scaling group' page in the AWS console, Step 1: 'Choose launch template or configuration'. The 'Name' field is 'Auto Scaling group name' with the value 'Lab Auto Scaling Group'. The 'Launch template' section shows 'LabConfig' selected. The 'Additional details' section shows 'Storage (volumes)' as '1' and 'Date created' as 'Mon Jun 26 2023 09:35:47 GMT+0700 (Già Đông Dương)'. The 'Summary' panel on the right shows the configuration: Software Image (AMI) 'Lab AMI for Web Server', Virtual server type (instance type) 't2.micro', Firewall (security group) 'Web Security Group', and Storage (volumes) '1 volume(s) - 8 GiB'.

### 6. Configure the network for the auto scaling group

The screenshot shows the 'Create Auto Scaling group' page in the AWS console, Step 2: 'Choose instance launch options'. The 'Network' section shows 'VPC' as 'vpc-0f8d044d0473aa3b3 (Lab VPC)' and 'Availability Zones and subnets' as 'us-east-1a | subnet-0abc77cde055804c1 (Private Subnet 1)' and 'us-east-1b | subnet-083203662f301066 (Private Subnet 2)'. The 'Summary' panel on the right shows the configuration: Software Image (AMI) 'Lab AMI for Web Server', Virtual server type (instance type) 't2.micro', Firewall (security group) 'Web Security Group', and Storage (volumes) '1 volume(s) - 8 GiB'.

7. Attach it to the load balancer have been created before , and enable the group metrics collection within cloudwatch.

The screenshot shows the 'Configure advanced options' page for an Auto Scaling group in the AWS Management Console. The page is titled 'Configure advanced options - optional' and includes a brief introduction. It is divided into three main sections: 'Load balancing', 'Attach to an existing load balancer', and 'VPC Lattice integration options'. In the 'Load balancing' section, the 'Attach to an existing load balancer' option is selected. In the 'Attach to an existing load balancer' section, the 'Choose from your load balancer target groups' option is selected, and the 'LabGroup [HTTP Application Load Balancer: LabELB]' target group is chosen. In the 'VPC Lattice integration options' section, the 'No VPC Lattice service' option is selected.

**Load balancing** [info](#)

Integrate your Auto Scaling group with other services to distribute network traffic across multiple servers using a load balancer or to establish service-to-service communications using VPC Lattice. You can also set options that give you more control over health check replacements and monitoring.

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 an existing load balancer**

Select the load balancers that you want to attach to your Auto Scaling group.

☒ Choose from your load balancer target groups  
This option allows you to attach Application, Network, or Gateway Load Balancers.

☐ Choose from Classic Load Balancers

Existing load balancer target groups  
Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

Select target groups

LabGroup [HTTP Application Load Balancer: LabELB]

**VPC Lattice integration options** [info](#)

To improve networking capabilities and scalability, integrate your Auto Scaling group with VPC Lattice. VPC Lattice facilitates communications between AWS services and helps you connect and manage your applications across compute services in AWS.

Select VPC Lattice service to attach

☒ No VPC Lattice service  
VPC Lattice will not manage your Auto Scaling group's network access and connectivity with other services.

☐ Attach to VPC Lattice service  
Incoming requests associated with specified VPC Lattice target groups will be routed to your Auto Scaling group.

[Create new VPC Lattice service](#)

8. Configure group size and scaling policies to ensure auto scaling automatically add/remove instance and always keep between 2 and 6 instances running.

The screenshot shows the 'Group size' and 'Scaling policies' pages for an Auto Scaling group in the AWS Management Console. The 'Group size' section allows specifying the size of the Auto Scaling group by changing the desired capacity, minimum capacity, and maximum capacity. The 'Scaling policies' section allows choosing whether to use a scaling policy to dynamically resize the Auto Scaling group to meet changes in demand. The 'Target tracking scaling policy' is selected, and the scaling policy name is 'LabScalingPolicy'. The metric type is 'Average CPU utilization', the target value is 60, and the instances need 300 seconds warm up before including in metric.

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

6

**Scaling policies - optional**

Choose whether to use a scaling policy to dynamically resize your Auto Scaling group to meet changes in demand. [info](#)

☒ 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

Scaling policy name

LabScalingPolicy

Metric type

Average CPU utilization

Target value

60

Instances need

300 seconds warm up before including in metric

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

Task 4: Verify that load balancing is working

## 1. Both instances has passed the load balancer's health check

The screenshot shows the AWS Management Console interface for Target groups. The top section, 'Target groups (1/1)', displays a table with one entry: 'LabGroup' with an ARN of 'arn:aws:elasticloadbalancing:us-east-1:6281-7364-9391:targetgroup/LabGroup/1424a22d5019222f'. Below this, the 'Target group: LabGroup' details are shown, including tabs for Details, Targets, Monitoring, Health checks, Attributes, and Tags. The 'Registered targets (2)' section shows two instances, both with a 'healthy' status.

Instance ID	Name	Port	Zone	Health status
i-07e8fa0b8d9b59229	Lab Instance	80	us-east-1b	healthy
i-07906001cd8cc843b	Lab Instance	80	us-east-1a	healthy

## 2. Get the DNS name of the load balancer

The screenshot shows the AWS Management Console interface for Load balancers. The top section, 'Load balancers (1/1)', displays a table with one entry: 'LabELB' with a DNS name of 'LabELB-1931556504.us-east-1.elb.amazonaws.com'. Below this, the 'Load balancer: LabELB' details are shown, including tabs for Details, Listeners, Network mapping, Security, Monitoring, Integrations, Attributes, and Tags. The 'Details' section shows the load balancer type as 'Application', status as 'Active', and DNS name as 'LabELB-1931556504.us-east-1.elb.amazonaws.com (A Record)'.

Meta-Data	Value
InstanceID	i-07e8fa0b8d9b59229
Availability Zone	us-east-1b

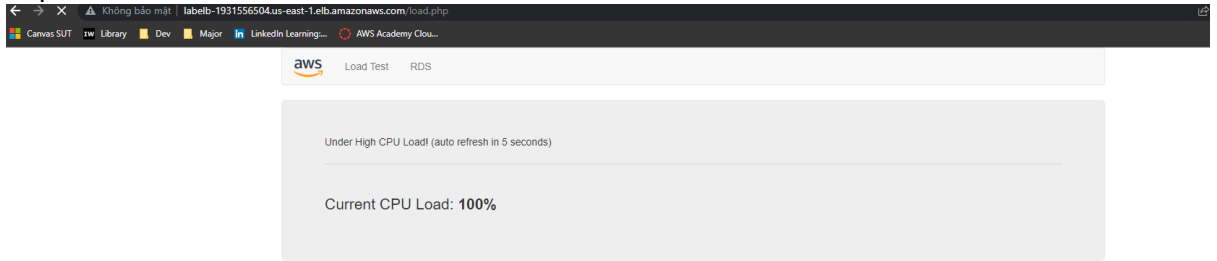
Current CPU Load: 1%

## 3. The result of going to the DNS page

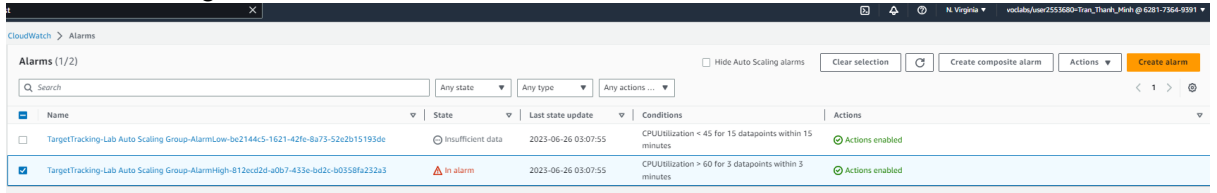
The screenshot shows a web browser window with the address bar displaying 'labelb-1931556504.us-east-1.elb.amazonaws.com'. The browser tabs include 'Canvas SUT', 'Library', 'Dev', 'Major', 'LinkedIn Learning...', and 'AWS Academy Clo...'. The main content area shows the AWS logo and 'Load Test' and 'RDS' links. Below this, a table displays meta-data for the load balancer, including InstanceID and Availability Zone. At the bottom, it shows the current CPU load as 1%.

## Task 5: Test auto scaling

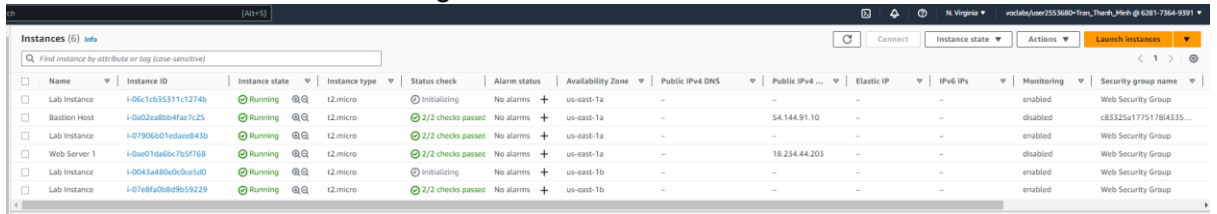
## 1. Implement load test



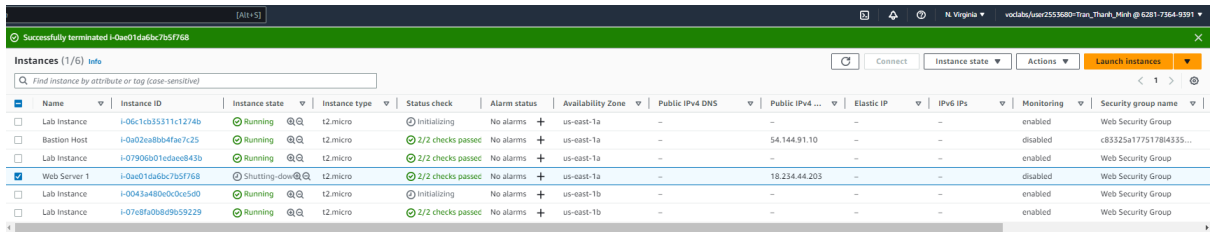
## 2. The state change



## 3. More than 2 instances are running



## Task 6: terminate web server 1



## 1.