

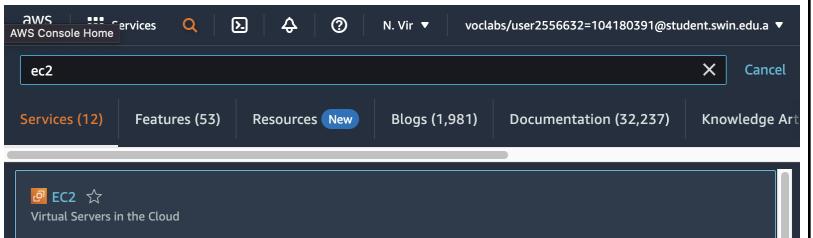
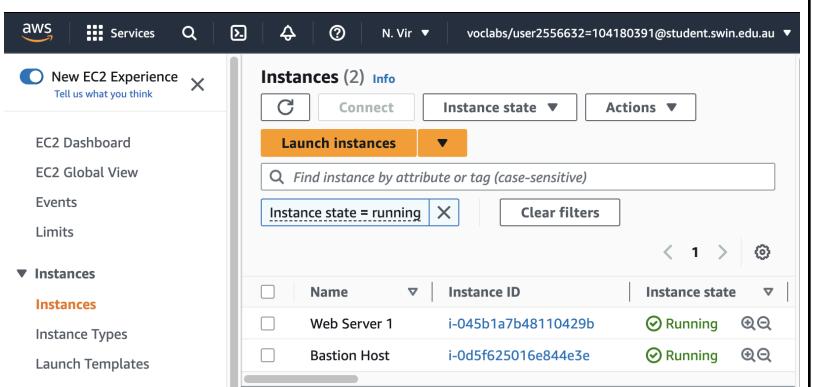
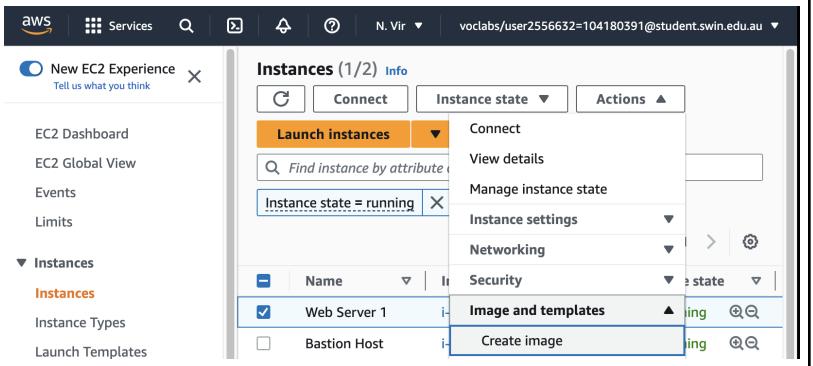
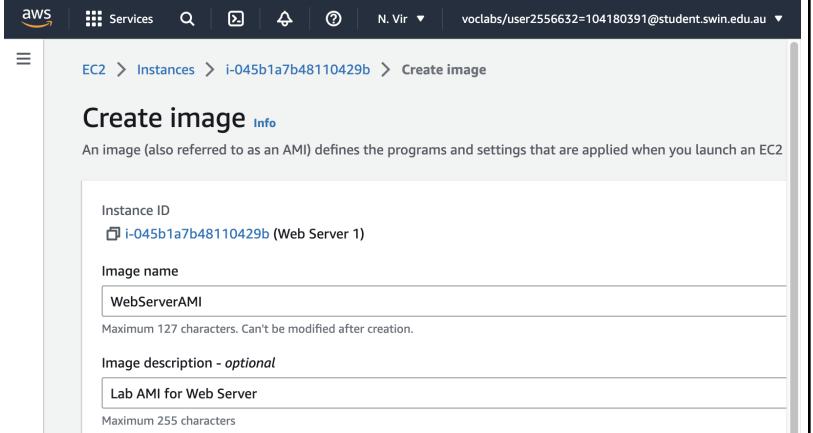


ACF Lab 6: Scaling and Load Balance your architecture

June 20, 2023

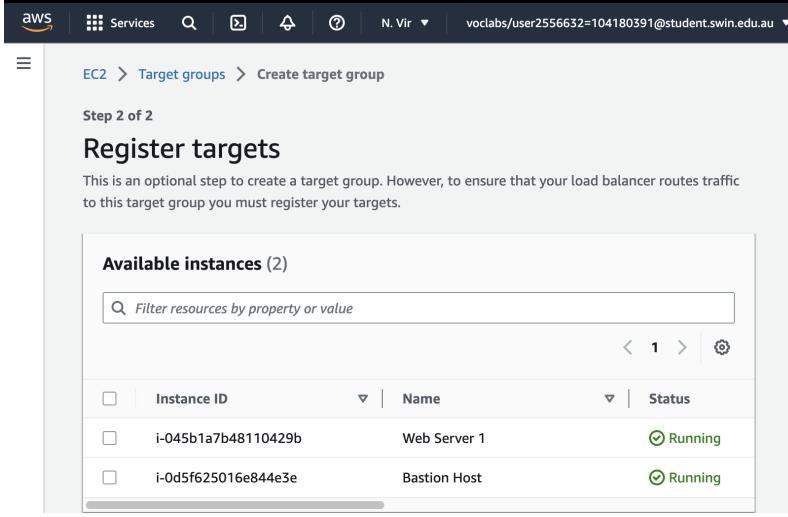
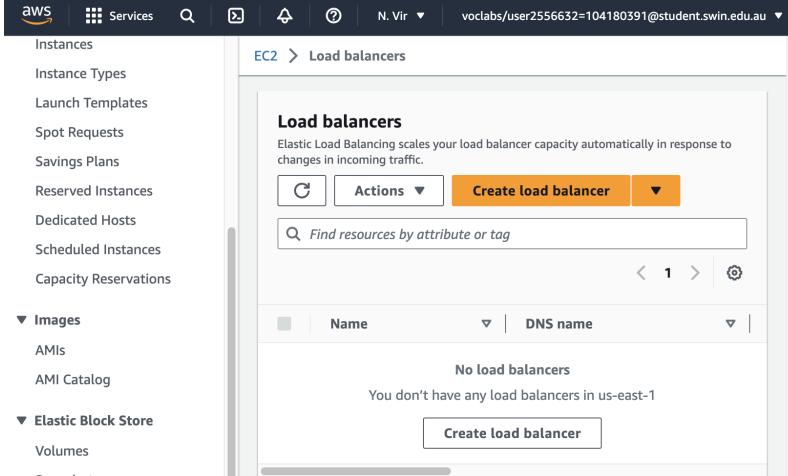
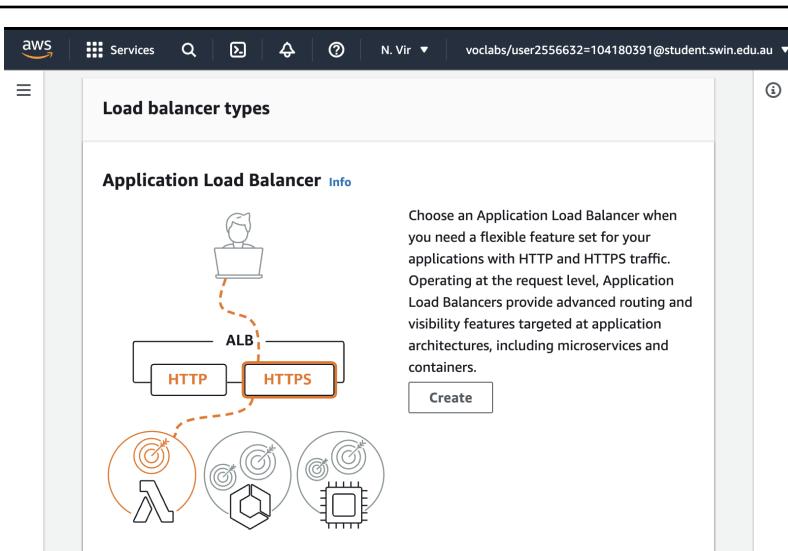
Luu Tuan Hoang
Student ID: 104180391

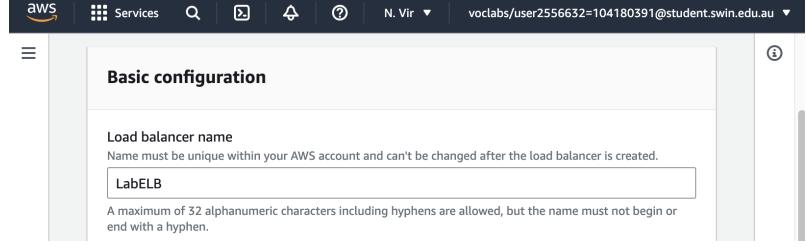
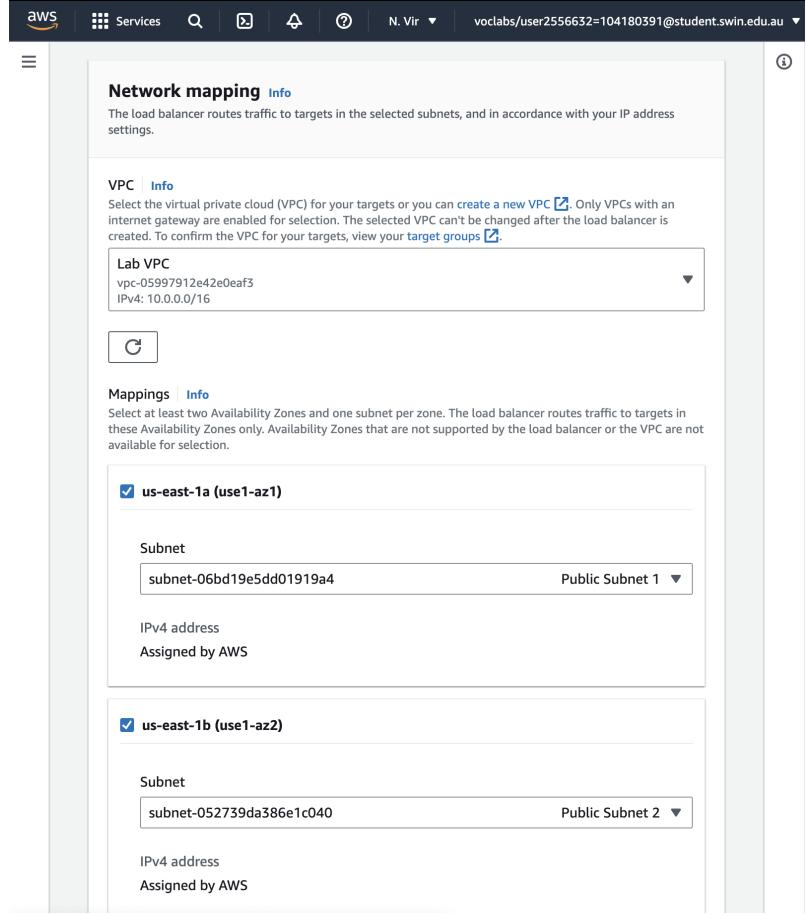
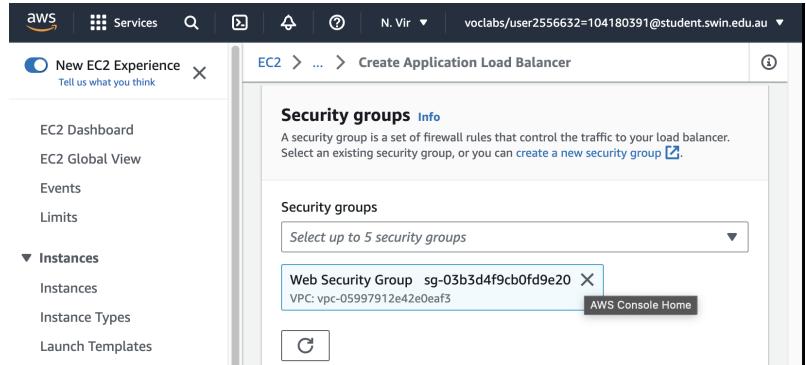
Task 1: Create an AMI for Auto Scaling

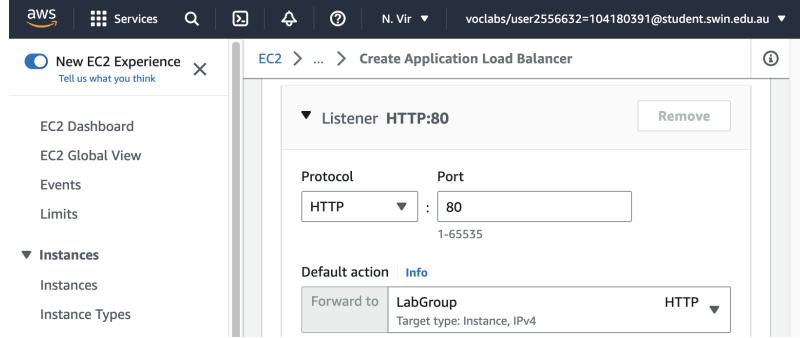
Step	Description	Screenshot
1	In the AWS Management Console, in the search box next to Services , search for and select EC2 .	
2	In the left navigation pane, choose Instances . First, confirm that the instance is running.	
3	Select Web Server 1 . In the Actions menu, choose Image and templates > Create image .	
4	Configuration: <ul style="list-style-type: none">- Image name: WebServerAMI- Image description: Lab AMI for Web Server Choose Create image .	

5	A confirmation banner displays the AMI ID for the new AMI.	
---	---	--

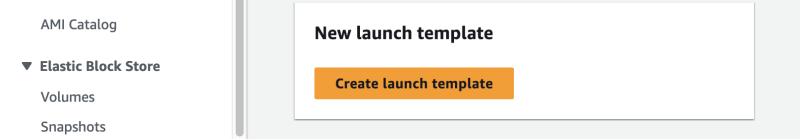
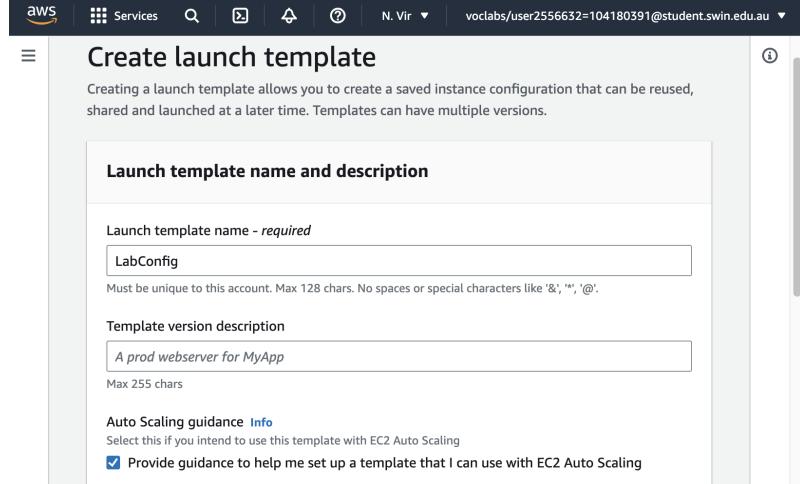
Task 2: Create a Load Balancer		
Step	Description	Screenshot
1	In the left navigation pane, choose Target Groups . Choose Create target group .	
2	Choose a target type: Instances	
3	Target group name , enter: LabGroup Select Lab VPC from the VPC drop-down menu.	

4	<p>Choose Next. The Register targets screen appears.</p> <p>Review the settings and choose Create target group.</p>	
5	<p>In the left navigation pane, choose Load Balancers.</p> <p>At the top of the screen, choose Create load balancer.</p>	
6	<p>Use Application Load Balancer that operates at the request level (layer 7), routing traffic to targets — EC2 instances, containers, IP addresses and Lambda functions — based on the content of the request.</p> <p>Under Application Load Balancer, choose Create.</p>	

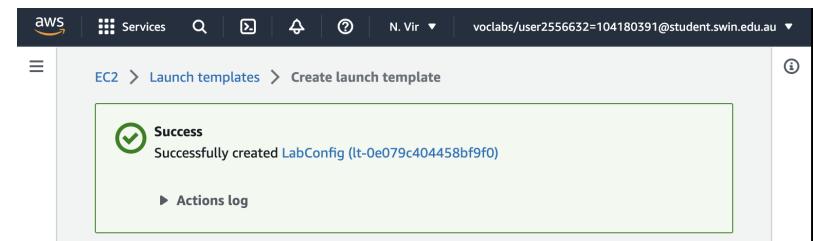
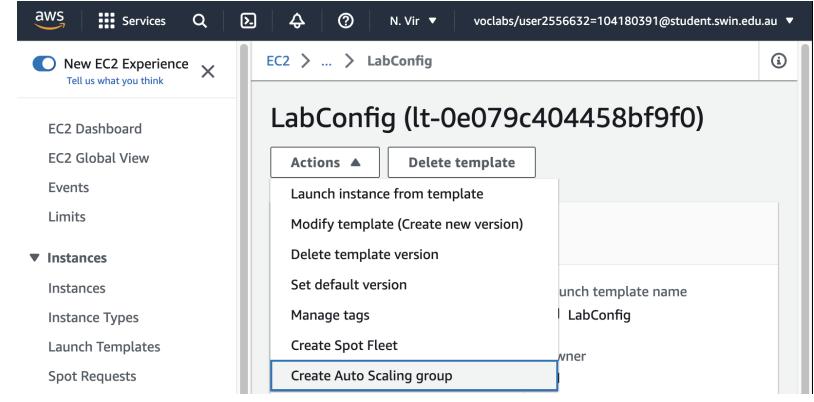
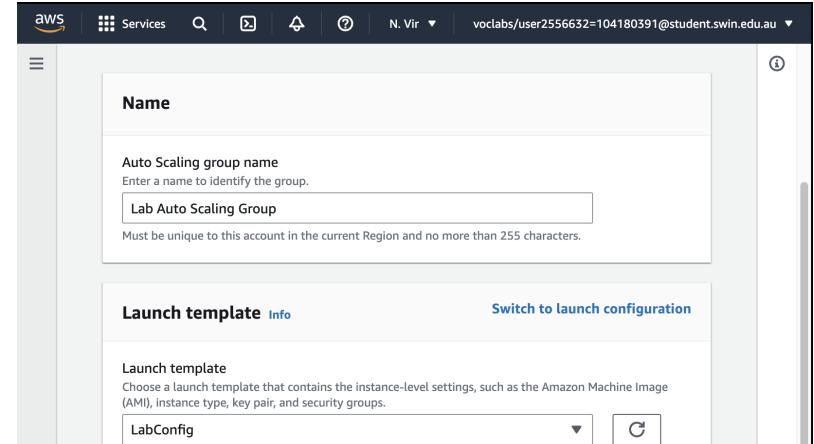
7	<p>Under Load balancer name, enter: LabELB.</p>	
8	<p>Scroll down to the Network mapping section, configuration:</p> <ul style="list-style-type: none"> - For VPC, choose Lab VPC - Choose the first displayed Availability Zone, then select Public Subnet 1 from the Subnet drop down menu that displays beneath it. - Choose the second displayed Availability Zone, then select Public Subnet 2 from the Subnet drop down menu that displays beneath it. - Two subnets should be selected: Public Subnet 1 and Public Subnet 2. 	
9	<p>In the Security groups section:</p> <ul style="list-style-type: none"> - Choose the Security groups drop down menu and select Web Security Group - Below the drop down menu, choose the X next to the default security group to remove it. - The Web Security Group security group should now be the only one that appears. 	

10	<p>For the Listener HTTP:80 row, set the Default action to forward to LabGroup.</p> <p>Scroll to the bottom and choose Create load balancer.</p>	
----	--	--

Task 3: Create a Launch Template and an Auto Scaling Group

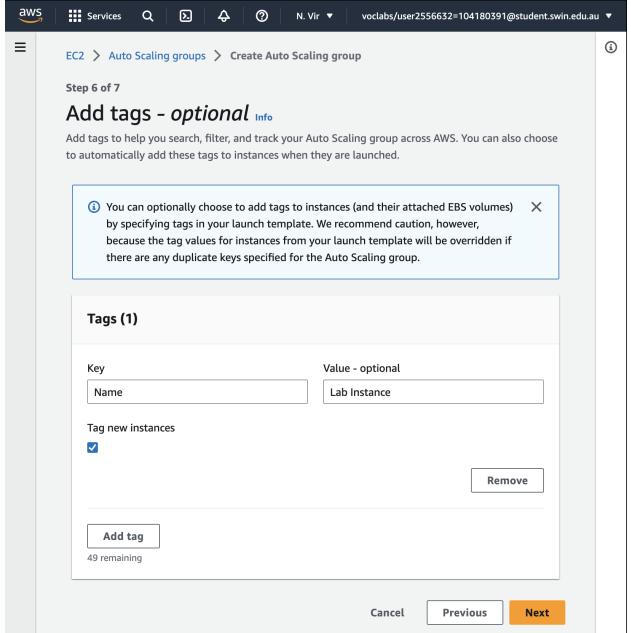
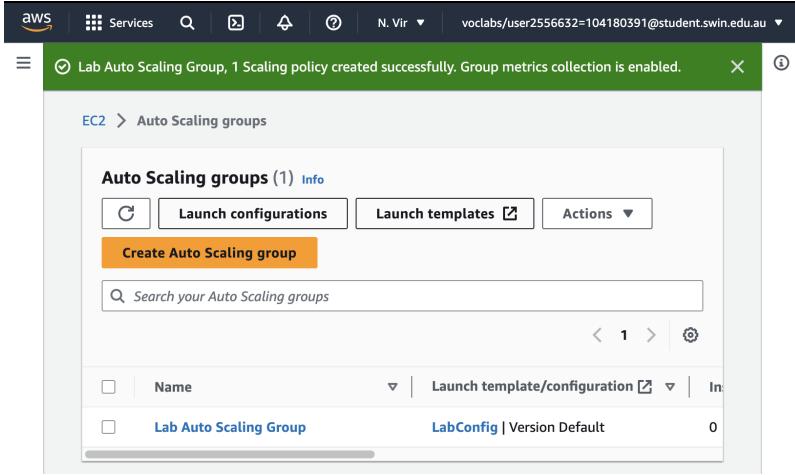
Step	Description	Screenshot
1	Choose Create launch template .	
2	<p>Configure the launch template settings and create it:</p> <ul style="list-style-type: none"> - Launch template name: LabConfig - Under Auto Scaling guidance, select Provide guidance to help me set up a template that I can use with EC2 Auto Scaling 	

3	<p>In the Application and OS Images (Amazon Machine Image) area, choose My AMIs.</p> <p>Amazon Machine Image (AMI): choose Web Server AMI</p>	
4	<p>Instance type: choose t2.micro</p> <p>Key pair name: choose vockey</p>	
5	<p>Firewall (security groups): choose Select existing security group</p> <p>Security groups: choose Web Security Group</p>	
6	<p>Scroll down to the Advanced details area and expand it.</p> <p>Scroll down to the Detailed CloudWatch monitoring setting. Select Enable</p>	

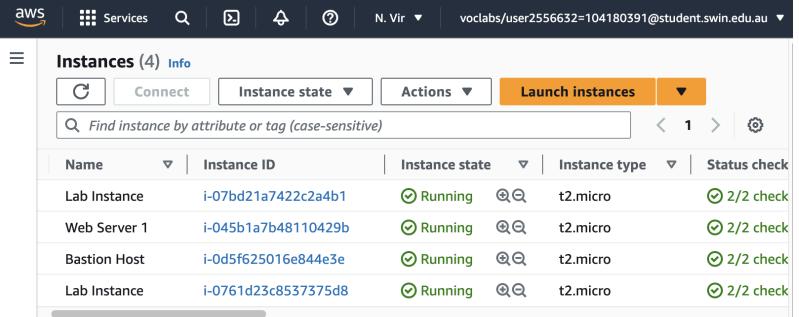
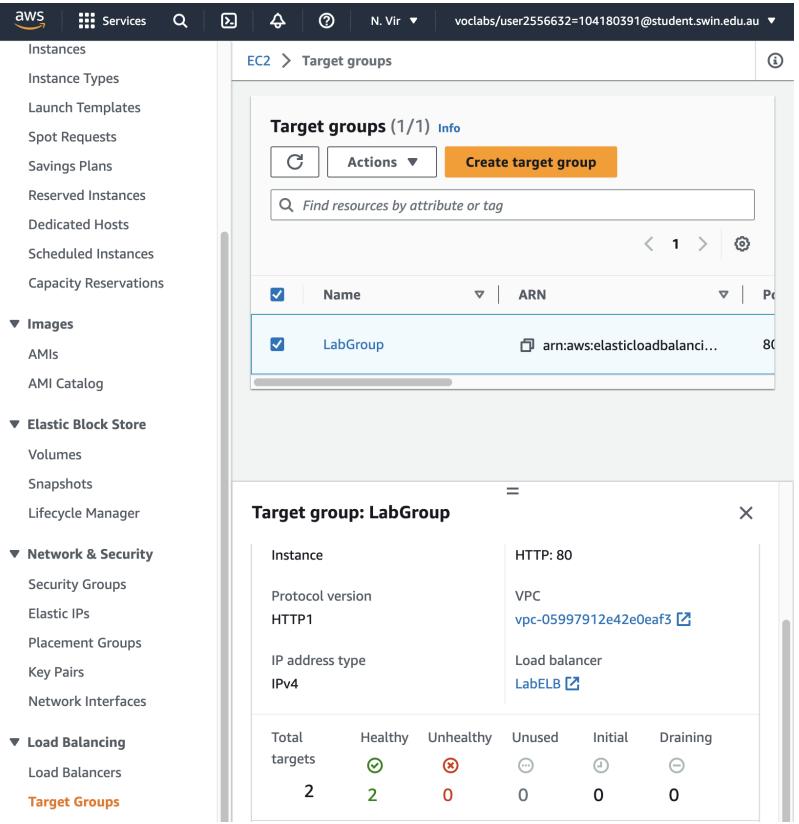
7	<p>Choose Create launch template.</p> <p>Choose the LabConfig launch template.</p>	
8	<p>From the Actions menu, choose Create Auto Scaling group</p>	
9	<p>Configure the details in Step 1 (Choose launch template or configuration):</p> <ul style="list-style-type: none"> - Auto Scaling group name: Lab Auto Scaling Group - Launch template: confirm that the LabConfig template you just created is selected. - Choose Next 	

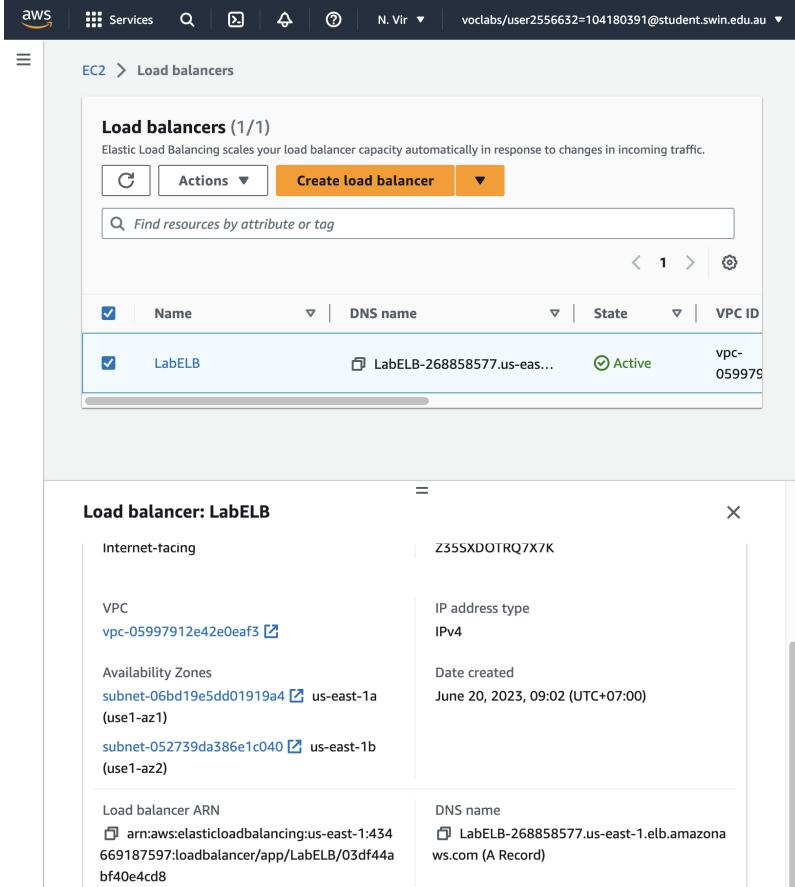
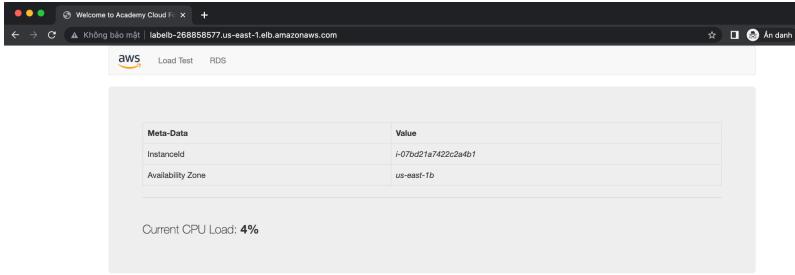
10	<p>Configure the details in Step 2 (Choose instance launch options):</p> <ul style="list-style-type: none"> - VPC: choose Lab VPC - Availability Zones and subnets: Choose Private Subnet 1 and then choose Private Subnet 2. - Choose Next 	
11	<p>Choose Attach to an existing load balancer</p> <p>Existing load balancer target groups: select LabGroup.</p>	

12	<p>In the Additional settings pane: Select Enable group metrics collection within CloudWatch</p>	
13	<p>Configure the details in Step 4 (Configure group size and scaling policies - optional):</p> <p>Under Group size, configure:</p> <ul style="list-style-type: none"> - Desired capacity: 2 - Minimum capacity: 2 - Maximum capacity: 6 <p>Under Scaling policies, choose Target tracking scaling policy and configure:</p> <ul style="list-style-type: none"> - Scaling policy name: LabScalingPolicy - Metric type: Average CPU Utilization - Target value: 60 <p>Choose Next</p>	
14	<p>Configure the details in Step 5 (Add notifications - optional):</p> <p>Auto Scaling can send a notification when a scaling event takes place. You will use the default settings.</p> <p>Choose Next</p>	

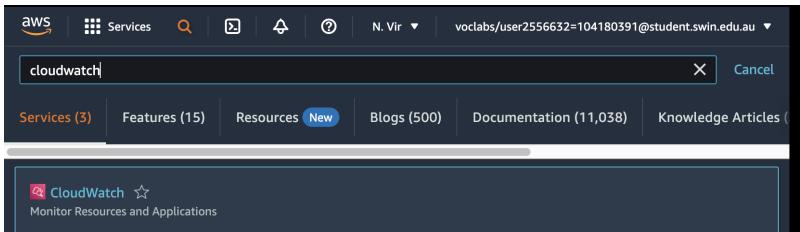
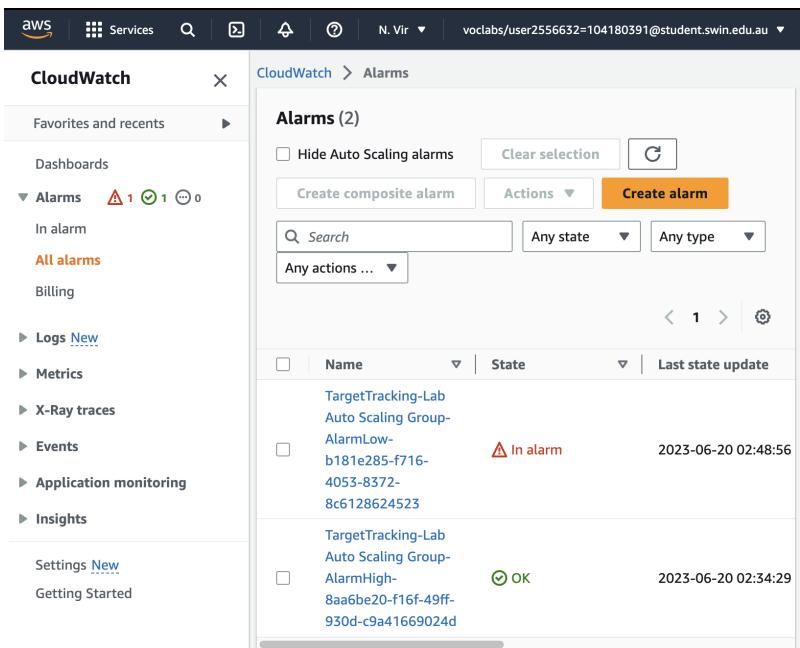
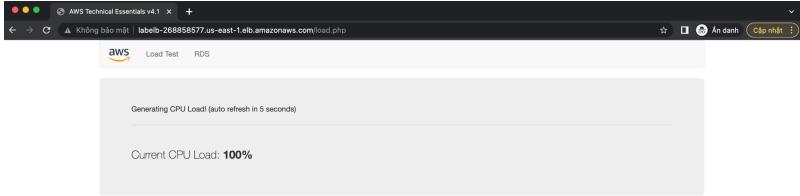
15	<p>Review the details of the Auto Scaling group</p> <p>Choose Create Auto Scaling group</p>	
16	Auto scaling group successfully created.	

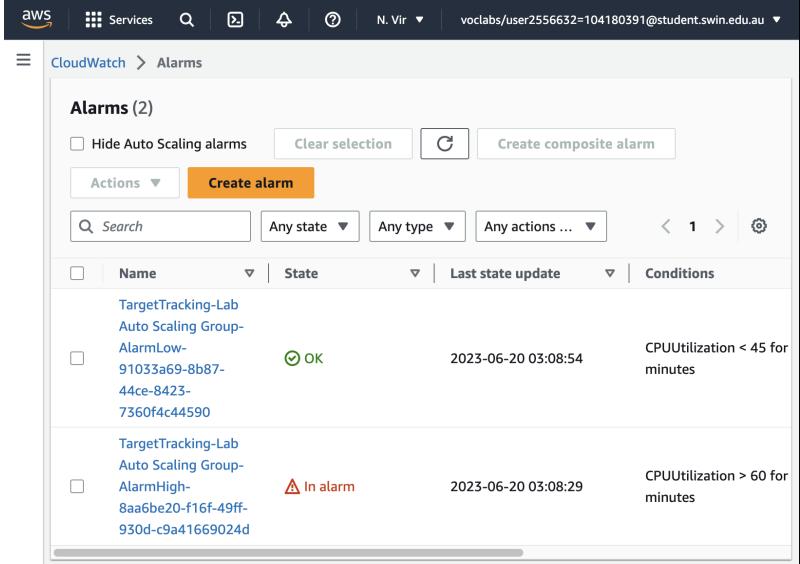
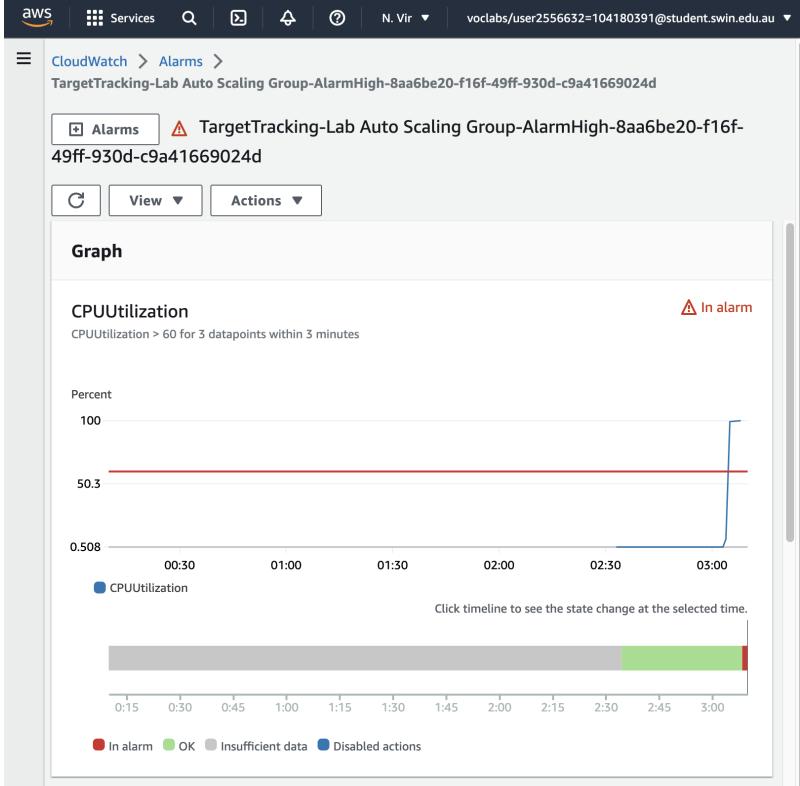
Task 4: Verify that Load Balancing is Working

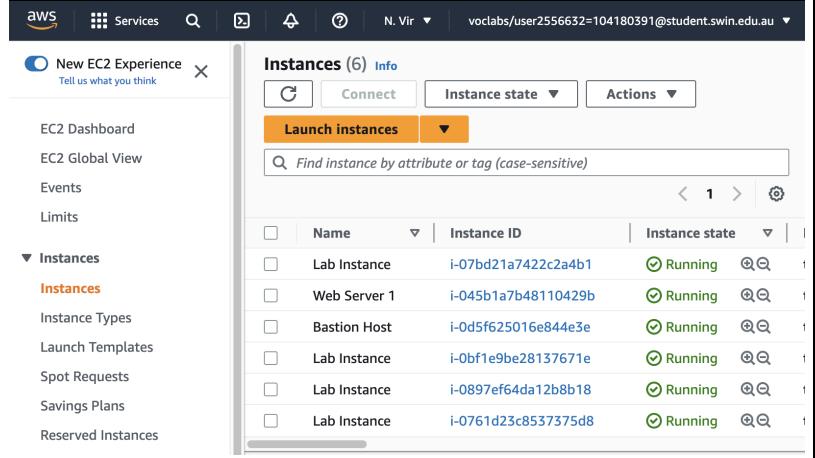
Step	Description	Screenshot																									
1	<p>There are two new instances named Lab Instance. These were launched by Auto Scaling.</p>	 <p>The screenshot shows the AWS EC2 Instances page with four instances listed:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Instance ID</th> <th>Instance state</th> <th>Instance type</th> <th>Status check</th> </tr> </thead> <tbody> <tr> <td>Lab Instance</td> <td>i-07bd21a7422c2a4b1</td> <td>Running</td> <td>t2.micro</td> <td>2/2 check</td> </tr> <tr> <td>Web Server 1</td> <td>i-045b1a7b48110429b</td> <td>Running</td> <td>t2.micro</td> <td>2/2 check</td> </tr> <tr> <td>Bastion Host</td> <td>i-0d5f625016e844e3e</td> <td>Running</td> <td>t2.micro</td> <td>2/2 check</td> </tr> <tr> <td>Lab Instance</td> <td>i-0761d23c8537375d8</td> <td>Running</td> <td>t2.micro</td> <td>2/2 check</td> </tr> </tbody> </table>	Name	Instance ID	Instance state	Instance type	Status check	Lab Instance	i-07bd21a7422c2a4b1	Running	t2.micro	2/2 check	Web Server 1	i-045b1a7b48110429b	Running	t2.micro	2/2 check	Bastion Host	i-0d5f625016e844e3e	Running	t2.micro	2/2 check	Lab Instance	i-0761d23c8537375d8	Running	t2.micro	2/2 check
Name	Instance ID	Instance state	Instance type	Status check																							
Lab Instance	i-07bd21a7422c2a4b1	Running	t2.micro	2/2 check																							
Web Server 1	i-045b1a7b48110429b	Running	t2.micro	2/2 check																							
Bastion Host	i-0d5f625016e844e3e	Running	t2.micro	2/2 check																							
Lab Instance	i-0761d23c8537375d8	Running	t2.micro	2/2 check																							
2	<p>In the left navigation pane, choose Target Groups.</p> <p>Select LabGroup</p> <p>Choose the Targets tab.</p> <p>Two target instances named Lab Instance should be listed in the target group.</p> <p>Wait until the Status of both instances transitions to healthy.</p>	 <p>The screenshot shows the AWS EC2 Target Groups page with the LabGroup target group selected. The Targets tab is active, showing two healthy targets:</p> <table border="1"> <thead> <tr> <th>Instance</th> <th>HTTP: 80</th> </tr> </thead> <tbody> <tr> <td>Protocol version</td> <td>VPC</td> </tr> <tr> <td>Placement Groups</td> <td>vpc-05997912e42e0af3</td> </tr> <tr> <td>IP address type</td> <td>Load balancer</td> </tr> <tr> <td>IPv4</td> <td>LabELB</td> </tr> </tbody> </table> <p>Summary table:</p> <table border="1"> <thead> <tr> <th>Total targets</th> <th>Healthy</th> <th>Unhealthy</th> <th>Unused</th> <th>Initial</th> <th>Draining</th> </tr> </thead> <tbody> <tr> <td>2</td> <td>2</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> </tbody> </table>	Instance	HTTP: 80	Protocol version	VPC	Placement Groups	vpc-05997912e42e0af3	IP address type	Load balancer	IPv4	LabELB	Total targets	Healthy	Unhealthy	Unused	Initial	Draining	2	2	0	0	0	0			
Instance	HTTP: 80																										
Protocol version	VPC																										
Placement Groups	vpc-05997912e42e0af3																										
IP address type	Load balancer																										
IPv4	LabELB																										
Total targets	Healthy	Unhealthy	Unused	Initial	Draining																						
2	2	0	0	0	0																						

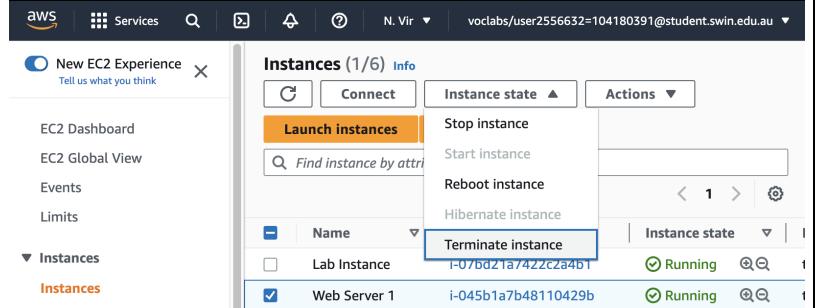
3	<p>In the left navigation pane, choose Load Balancers.</p> <p>Select the LabELB load balancer.</p> <p>In the Details pane, copy the DNS name of the load balancer, making sure to omit "(A Record)".</p> <p>Open a new web browser tab, paste the DNS Name, and press Enter.</p>	 <p>The screenshot shows the AWS EC2 Load Balancers console. At the top, there's a header with the AWS logo, services menu, search bar, and user information. Below it, the main title is "Load balancers (1/1)". A sub-header says "Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic." There are buttons for "Actions" and "Create load balancer". A search bar is present. The main table lists one load balancer:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>DNS name</th> <th>State</th> <th>VPC ID</th> </tr> </thead> <tbody> <tr> <td>LabELB</td> <td>LabELB-268858577.us-eas...</td> <td>Active</td> <td>vpc-05997912e42e0ef3</td> </tr> </tbody> </table> <p>A modal window titled "Load balancer: LabELB" is open, displaying detailed information:</p> <ul style="list-style-type: none"> Internet-facing: Z35SXDOTRQ7X7K VPC: vpc-05997912e42e0ef3 IP address type: IPv4 Availability Zones: <ul style="list-style-type: none"> subnet-06bd19e5dd01919a4 us-east-1a (use1-az1) subnet-052739da386e1c040 us-east-1b (use1-az2) Date created: June 20, 2023, 09:02 (UTC+07:00) Load balancer ARN: arn:aws:elasticloadbalancing:us-east-1:434669187597:loadbalancer/app/LabELB/03df44abf40e4cd8 DNS name: LabELB-268858577.us-east-1.elb.amazonaws.com (A Record) 	Name	DNS name	State	VPC ID	LabELB	LabELB-268858577.us-eas...	Active	vpc-05997912e42e0ef3
Name	DNS name	State	VPC ID							
LabELB	LabELB-268858577.us-eas...	Active	vpc-05997912e42e0ef3							
4	This indicates that the Load Balancer received the request, sent it to one of the EC2 instances, then passed back the result.	 <p>The screenshot shows a web browser window with the URL "labelb-268858577.us-east-1.elb.amazonaws.com". The page displays meta-data information:</p> <table border="1"> <thead> <tr> <th>Meta-Data</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>InstanceId</td> <td>i-07bd21a7422c2a4b1</td> </tr> <tr> <td>Availability Zone</td> <td>us-east-1b</td> </tr> </tbody> </table> <p>At the bottom, it says "Current CPU Load: 4%".</p>	Meta-Data	Value	InstanceId	i-07bd21a7422c2a4b1	Availability Zone	us-east-1b		
Meta-Data	Value									
InstanceId	i-07bd21a7422c2a4b1									
Availability Zone	us-east-1b									

Task 5: Test Auto Scaling

Step	Description	Screenshot
1	In the search box next to Services , search for and select CloudWatch .	
2	<p>In the left navigation pane, choose All alarms.</p> <p>Two alarms will be displayed. These were created automatically by the Auto Scaling group. They will automatically keep the average CPU load close to 60% while also staying within the limitation of having two to six instances.</p>	
3	<p>Return to the browser tab with the web application.</p> <p>Choose Load Test beside the AWS logo.</p> <p>This will cause the application to generate high loads. The browser page will automatically refresh so that all instances in the Auto Scaling group will generate load.</p>	

4	<p>In less than 5 minutes, the AlarmLow alarm should change to OK and the AlarmHigh alarm status should change to In alarm.</p>	 <p>The screenshot shows the AWS CloudWatch Alarms page with two entries:</p> <ul style="list-style-type: none"> TargetTracking-Lab Auto Scaling Group-AlarmLow- 91033a69-8b87-44ce-8423-7360f4c44590: State is OK, last updated 2023-06-20 03:08:54. Condition: CPUUtilization < 45 for minutes. TargetTracking-Lab Auto Scaling Group-AlarmHigh- 8aa6be20-f16f-49ff-930d-c9a41669024d: State is In alarm, last updated 2023-06-20 03:08:29. Condition: CPUUtilization > 60 for minutes.
5	<p>The AlarmHigh chart indicates an increasing CPU percentage. Once it crosses the 60% line for more than 3 minutes, it will trigger Auto Scaling to add additional instances.</p>	 <p>The screenshot shows the AWS CloudWatch Metrics Graph for CPUUtilization. The graph displays the following data:</p> <ul style="list-style-type: none"> Series: CPUUtilization (blue line). Y-axis: Percent (0.508, 50.3, 100). X-axis: Time (0:15, 0:30, 0:45, 1:00, 1:15, 1:30, 1:45, 2:00, 2:15, 2:30, 2:45, 3:00). Annotations: A red horizontal line at 50.3 and a blue line at 0.508. A red triangle icon labeled "In alarm" is positioned at the top right of the graph area. Legend: Shows status indicators: In alarm (red), OK (green), Insufficient data (grey), and Disabled actions (blue). <p>A note at the bottom of the graph states: "Click timeline to see the state change at the selected time."</p>

6	<p>More than two instances labeled Lab Instance should now be running. The new instance(s) were created by Auto Scaling in response to the CloudWatch alarm.</p>	 <table border="1"> <thead> <tr> <th>Name</th> <th>Instance ID</th> <th>Instance State</th> </tr> </thead> <tbody> <tr> <td>Lab Instance</td> <td>i-07bd21a7422c2a4b1</td> <td>Running</td> </tr> <tr> <td>Web Server 1</td> <td>i-045b1a7b48110429b</td> <td>Running</td> </tr> <tr> <td>Bastion Host</td> <td>i-0d5f625016e844e3e</td> <td>Running</td> </tr> <tr> <td>Lab Instance</td> <td>i-0bf1e9be28137671e</td> <td>Running</td> </tr> <tr> <td>Lab Instance</td> <td>i-0897ef64da12b8b18</td> <td>Running</td> </tr> <tr> <td>Lab Instance</td> <td>i-0761d23c8537375d8</td> <td>Running</td> </tr> </tbody> </table>	Name	Instance ID	Instance State	Lab Instance	i-07bd21a7422c2a4b1	Running	Web Server 1	i-045b1a7b48110429b	Running	Bastion Host	i-0d5f625016e844e3e	Running	Lab Instance	i-0bf1e9be28137671e	Running	Lab Instance	i-0897ef64da12b8b18	Running	Lab Instance	i-0761d23c8537375d8	Running
Name	Instance ID	Instance State																					
Lab Instance	i-07bd21a7422c2a4b1	Running																					
Web Server 1	i-045b1a7b48110429b	Running																					
Bastion Host	i-0d5f625016e844e3e	Running																					
Lab Instance	i-0bf1e9be28137671e	Running																					
Lab Instance	i-0897ef64da12b8b18	Running																					
Lab Instance	i-0761d23c8537375d8	Running																					

Task 6: Terminate Web Server 1		
Step	Description	Screenshot
1	<p>Select Web Server 1 (and ensure it is the only instance selected).</p> <p>In the Instance state menu, choose Instance State > Terminate Instance.</p>	
2	Instance terminated	