

# **Setting up/Tearing down an Elastic Load Balancer**

## **Creating our Elastic Load Balancer Environment**

In this documentation we will be learning what is an Elastic Load Balancer within the AWS cloud platform, creating the best environment for its setup, the process of its setup and its tear down when its no longer needed. Let us begin.

### ***What is an Elastic Load Balancer or ELB service in AWS?***

*An Elastic Load Balancer is a server that directs incoming traffic downstream to several targets within a virtual private network. These targets can be virtual machines like EC2's, packaged applications or containers, and other IP addressed devices in the network.*

*The ELB does health checks on the targeted devices to make sure they are functioning to receive any traffic. If the device fails a health check, the ELB automatically distributes the traffic to other targets, so the devices as well as the VPC, are not overwhelmed, damaged.*

So in short, the elastic load balancer divides incoming traffic to the targeted devices/services in a stable manner in the VPC. It's an elastic service because you can set it up and release it/tear it down whenever necessary.

To setup an ELB, it is best practice to setup other devices, or an environment that best serves the ELB. To do this, we will create a VPC, security groups, a target group and finally an ELB.

We detailed the process of setting up VPC, so we will continue to use the WuCloud VPC for the Elastic Load Balancer documentation.

Lets us define the devices/services of the ELB setup.

## ***What is a Security Group in AWS?***

*A Security Group is a network security device or **firewall** that monitors incoming traffic into a virtual machine based on defined security rules.*

Security groups secure the devices in the VPC. We will document the process of creating both security groups. One group for EC2's and another for our ELB.

## ***What is a Target Group in AWS?***

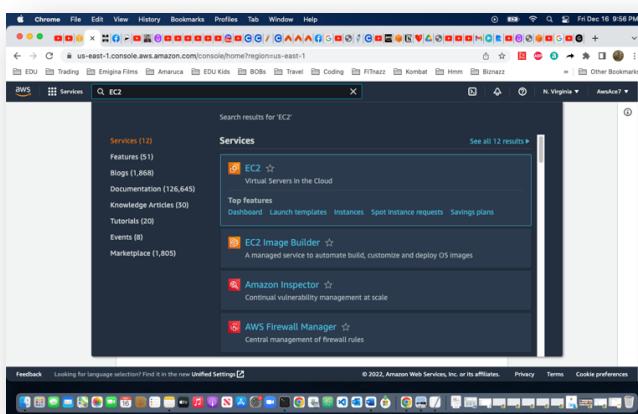
*A Target group is a device that tells the load balancer where to direct the traffic.*

We will set up a target group that will listen for specific traffic and inform the load balancer on where to send it.

Let's begin creating our Security Group.

## **Setting up our Security Group**

To create a Security Group, they are in the EC2 dashboard section of AWS services. Once we are signed in the AWS console area, we can go to the search bar at the top of the screen. Type “EC2” to bring up the EC2 option. We will select the EC2, to go to the EC2 dashboard page.



Here in the EC2 dashboard, on the left side menu, we will scroll down until we find the option “Security Groups”. We will select it.

The screenshot shows the AWS EC2 Dashboard in a web browser. The left sidebar has a collapsed "New EC2 Experience" section and expanded "Instances" and "Network & Security" sections. Under "Instances", options like Instances, Instance Types, Launch Templates, and Security Groups are listed. Under "Network & Security", options like Volumes, Snapshots, Lifecycle Manager, Security Groups, and Network Interfaces are listed. The main content area displays "Resources" for the US East (N. Virginia) Region, showing 0 instances (running), 3 elastic IPs, 7 key pairs, 0 placement groups, 0 snapshots, 0 dedicated hosts, 0 instances, 0 load balancers, 4 security groups, and 0 volumes. An "Account attributes" sidebar on the right lists supported platforms (VPC), default VPC (vpc-0df1cb3cae784a2d8), settings for EBS encryption, zones, EC2 Serial Console, default credit specification, and console experiments. A "Explore AWS" sidebar on the right suggests enabling best price-performance. A tooltip at the bottom left of the main content area provides information about Microsoft SQL Server Always On availability groups.

This screenshot is identical to the one above, but the "Security Groups" option under the "Network & Security" section in the left sidebar is now highlighted, indicating it has been selected.

In the Security group section, we will see at least one security group already made, which is the default.

**Note:** We **DO NOT TOUCH ANY DEFAULT DEVICES IN AWS, OR WE WILL BE TOUCHED IN A HORRIFIC CTHULHU (look it up) MANNER!**

Go to the right side of the screen and select the “Create Security Group” button.

The screenshot shows the AWS Management Console interface for the EC2 service. The left sidebar navigation includes 'Volumes', 'Snapshots', 'Lifecycle Manager', 'Network & Security' (with 'Security Groups' selected), 'Elastic IPs', 'Placement Groups', 'Key Pairs', and 'Network Interfaces'. The 'Load Balancing' and 'Auto Scaling' sections are also visible. The main content area displays a table titled 'Security Groups (4)'. The table has columns for Name, Security group ID, Security group name, VPC ID, and Description. The data is as follows:

Name	Security group ID	Security group name	VPC ID	Description
WakandaSecurity	sg-083c88f6311a14765	WakandaSecurity	vpc-0df1cb3cae784a2d8	We will not have
MySecureMoneyWo...	sg-092527783e1f3b856	MySecureMoneyWorld...	vpc-0df1cb3cae784a2d8	MySecureMone
-	sg-0c0c0054ca002aaaf2	default	vpc-08c2cc72ad9a850ea	default VPC sec
Default	sg-08de463c33ff76465	default	vpc-0df1cb3cae784a2d8	default VPC sec

At the bottom of the page, there are links for 'Feedback', 'Looking for language selection? Find it in the new Unified Settings', '© 2022, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

We begin filling out the input fields in the **Basic Details** section. We give the security group a name with no spaces. Next a description of what is the security group is for. Then for VPC, we click the “X” in the input field and select a *PREVIOUSLY* made VPC. In this case, the WuCloud VPC. In this case, the WuCloud VPC.

The screenshot shows the AWS Lambda console with the URL <https://us-east-1.console.aws.amazon.com/lambda/functions/CreateFunctionStep2?region=us-east-1>. The page title is "Create function". The "Runtime" dropdown is set to "Node.js 14.x". The "Add triggers" section is expanded, showing two triggers: "aws-lambda-powertools-test" and "aws-lambda-powertools-test-2". The "Code" section shows the "File" tab selected, with the file "index.js" open. The code contains a single line: `console.log("Hello world")`. The "Environment variables" section is collapsed. The "Advanced settings" section is also collapsed. The bottom of the screen shows the AWS Lambda navigation bar and a Mac OS X dock.

The screenshot shows the AWS Lambda console with the URL <https://us-east-1.console.aws.amazon.com/lambda/functions/CreateFunctionStep2?region=us-east-1>. The page title is "Create function". The "Runtime" dropdown is set to "Node.js 14.x". The "Add triggers" section is expanded, showing two triggers: "aws-lambda-powertools-test" and "aws-lambda-powertools-test-2". The "Code" section shows the "File" tab selected, with the file "index.js" open. The code contains a single line: `console.log("Hello world")`. The "Environment variables" section is collapsed. The bottom of the screen shows the AWS Lambda navigation bar and a Mac OS X dock.

For the **Inbound Rules** section, we will define the security rules, the security group will use to verify, accept incoming traffic data. We will set the rules for allowing HTTP, SSH and HTTPS web traffic. This is traffic coming for web ports or portals 80, 22 and 443.

We will also set a source rule as “anywhere” traffic using IPv4. Lastly for the descriptions, we will state what type of traffic it is.

The screenshot shows the AWS VPC Inbound Rules configuration page. The 'Type' dropdown is set to 'Custom TCP', 'Protocol' is 'TCP', and 'Port range' is '0'. The 'Source' dropdown is set to 'Custom' with the value 'vpc-08c2cc72ad9a850ea'. The 'Description - optional' field is empty. An 'Add rule' button is visible. Below this, the Outbound rules section is shown with 'All traffic' selected.

The screenshot shows the AWS VPC Inbound Rules configuration page. The 'Type' dropdown is set to 'HTTP', 'Protocol' is 'TCP', and 'Port range' is '80'. The 'Source' dropdown is set to 'Custom' with the value 'Anywhere-IPv4'. The 'Description - optional' field is empty. An 'Add rule' button is visible. Below this, the Outbound rules section is shown with 'All traffic' selected.

As you can see, we're setting up and finishing the inbound rules. We will NOT make any changes to **Outbound Rules**.

The screenshot shows the AWS VPC Inbound Rules configuration page. At the top, there is a search bar with the query "vpc-08c2cc72ad9a850ea". Below the search bar, the "Inbound rules" section is visible, showing one rule:

Type	Protocol	Port range	Source	Description - optional
HTTP	TCP	80	Anywhere	HTTP

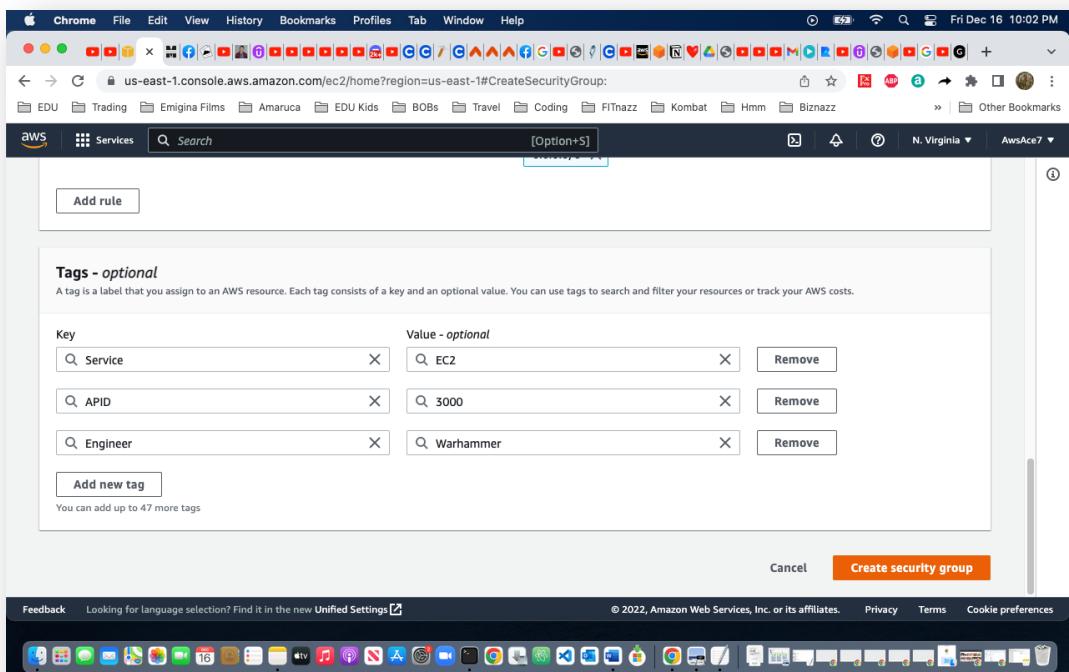
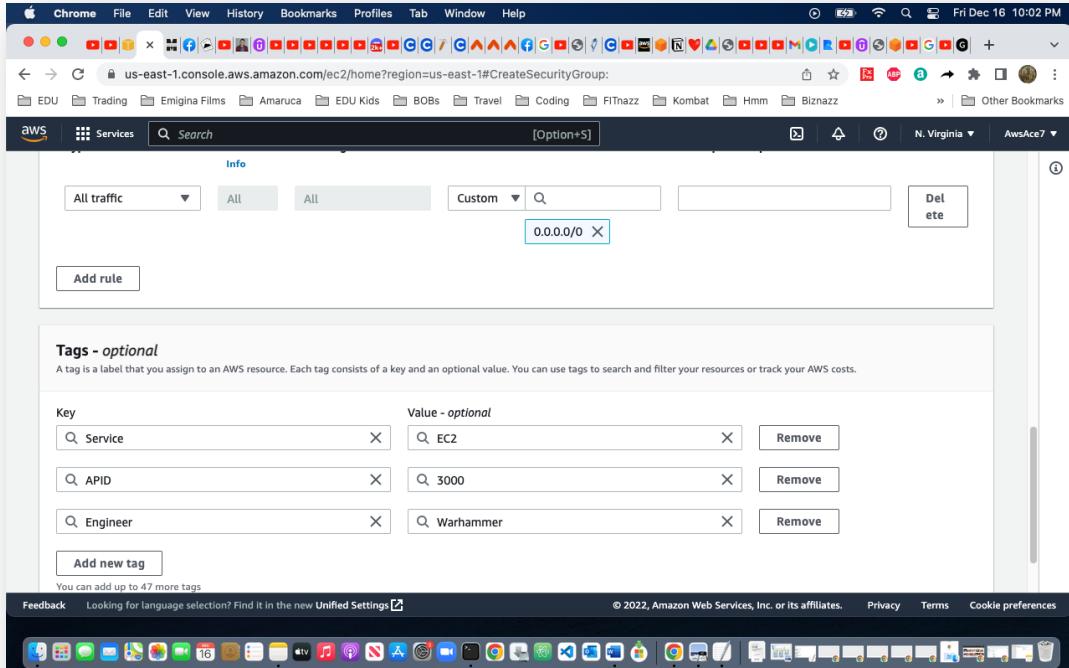
Below the Inbound rules, the "Outbound rules" section is shown, which is currently empty.

The screenshot shows the AWS VPC Inbound Rules configuration page after adding three new rules. The "Inbound rules" section now contains three entries:

Type	Protocol	Port range	Source	Description - optional
HTTP	TCP	80	Anywhere	HTTP
SSH	TCP	22	Anywhere	Secure Access
HTTPS	TCP	443	Anywhere	HTTPS

The "Outbound rules" section remains empty.

In the **Tags** section, we will use reference markers so we can quickly find this group when necessary. After we put in several tags, we will select the “create security group” button to complete setup.



Now we have our first Security Group created. We can click the hyperlink “Security Groups” to go to the security group dashboard.

Here we can select our new group and name it, so we do not confuse it with any other groups.

The screenshot shows the AWS EC2 Security Groups dashboard. A green success message at the top states: "Security group (sg-0093b683e88329a05 | MyWuSGEC2) was created successfully". Below this, the details for the new security group "sg-0093b683e88329a05 - MyWuSGEC2" are displayed. The "Details" section shows:

Security group name	MyWuSGEC2	Security group ID	sg-0093b683e88329a05	Description	Security Group for EC2	VPC ID	vpc-08c2cc72ad9a850ea
Owner	966320976686	Inbound rules count	3 Permission entries	Outbound rules count	1 Permission entry		

Below the details, there are tabs for "Inbound rules", "Outbound rules", and "Tags". The "Inbound rules" tab is selected. The bottom of the page includes a feedback link and standard copyright information.

The screenshot shows the AWS EC2 Security Groups dashboard. At the top, there is a search bar and a "Create security group" button. The main area displays a table of security groups:

Name	Security group ID	Security group name	VPC ID	Description
WakandaSecurity	sg-083c88f6311a14765	WakandaSecurity	vpc-0df1cb3cae784a2d8	We will not have
MySecureMoneyWo...	sg-092527783e1f3b856	MySecureMoneyWorld...	vpc-0df1cb3cae784a2d8	MySecureMo
WuSGEC2	sg-0093b683e88329a05	MyWuSGEC2	vpc-08c2cc72ad9a850ea	Security Gro
-	sg-0cc0054ca002aaaf2	default	vpc-08c2cc72ad9a850ea	default VPC
Default	sg-08de463c33ff76465	default	vpc-0df1cb3cae784a2d8	default VPC

Below the table, a specific security group "sg-0093b683e88329a05 - MyWuSGEC2" is selected, showing its details. The "Details" tab is selected, and a message says: "You can now check network connectivity with Reachability Analyzer". There is also a "Run Reachability Analyzer" button. The bottom of the page includes a feedback link and standard copyright information.

The following series of screenshots showcase the exact same process, but this time, a Security group for our Elastic Load Balancer.

A screenshot of a Chrome browser window on a Mac OS X desktop. The URL in the address bar is `us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateSecurityGroup`. The page title is "Create security group".  
**Basic details**  
Security group name: MyWuSGELB  
Description: Security Group for Elastic Load Balancer  
VPC: A dropdown menu shows two options: "vpc-0df1cb3cae784a2d8 (Default VPC)" and "vpc-08c2cc72ad9a850ea (TheWuCloud-vpc)". The second option is selected.  
Feedback: Looking for language selection? Find it in the new Unified Settings.  
Footer: © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

A screenshot of the same Chrome browser window, now showing the "Inbound rules" section of the security group creation page.  
**Inbound rules**  
Type: HTTP  
Protocol: TCP  
Port range: 80  
Source: Custom  
Description - optional: (empty)  
Delete  
Protocol: TCP  
Port range: All  
Destination: Custom  
Description - optional: (empty)  
Delete  
Feedback: Looking for language selection? Find it in the new Unified Settings.  
Footer: © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Chrome File Edit View History Bookmarks Profiles Tab Window Help Fri Dec 16 10:04 PM

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateSecurityGroup:

EDU Trading Emigina Films Amarula EDU Kids BOBs Travel Coding FITnazz Kombat Hmm Biznazz Other Bookmarks N. Virginia AwsAce7

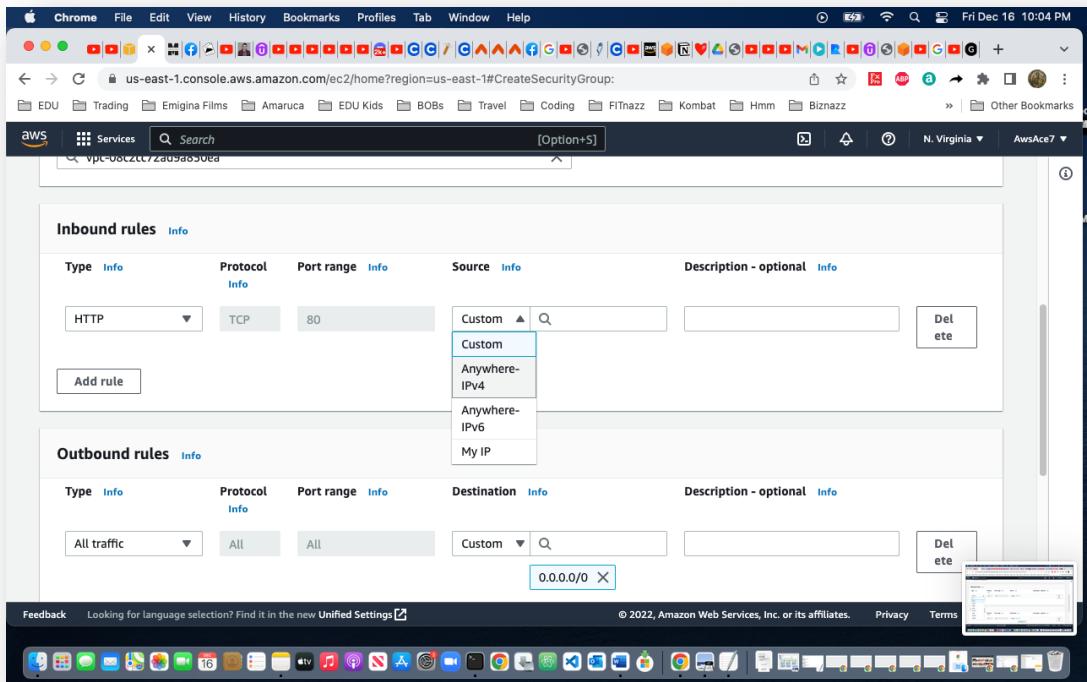
Inbound rules Info

Type	Protocol	Port range	Source	Description - optional
HTTP	TCP	80	Custom	
Add rule				

Outbound rules Info

Type	Protocol	Port range	Destination	Description - optional
All traffic	All	All	Custom	0.0.0.0/0
Delete				

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms



Chrome File Edit View History Bookmarks Profiles Tab Window Help Fri Dec 16 10:04 PM

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateSecurityGroup:

EDU Trading Emigina Films Amarula EDU Kids BOBs Travel Coding FITnazz Kombat Hmm Biznazz Other Bookmarks N. Virginia AwsAce7

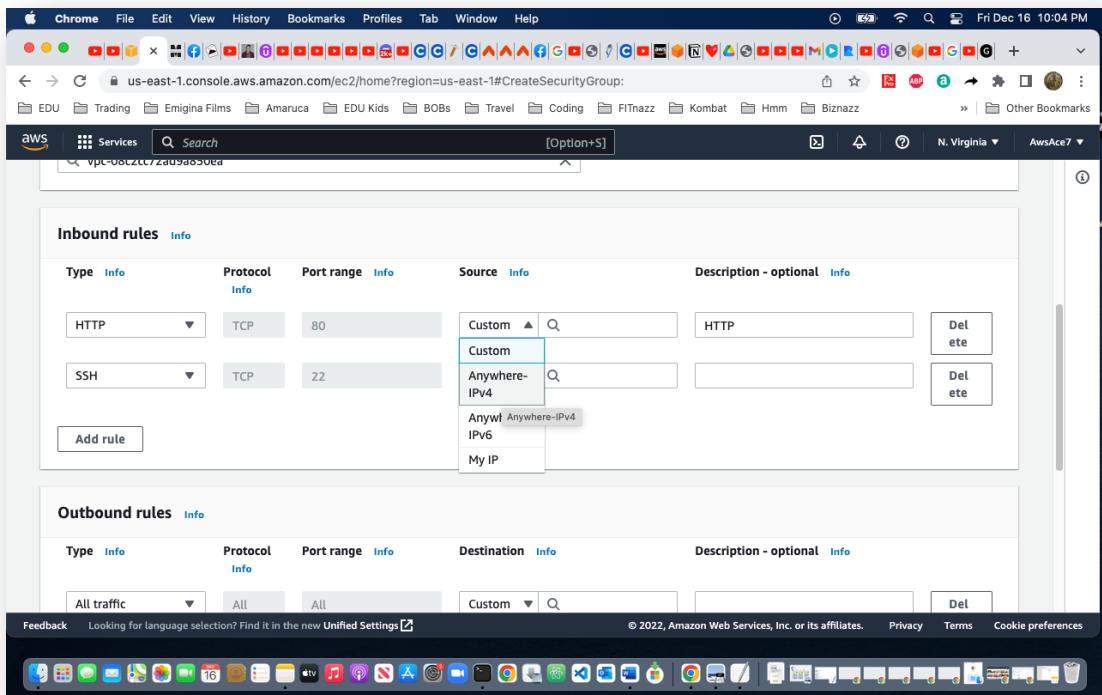
Inbound rules Info

Type	Protocol	Port range	Source	Description - optional
HTTP	TCP	80	Custom	HTTP
SSH	TCP	22	Custom	Anywhere-IPv4
Add rule				

Outbound rules Info

Type	Protocol	Port range	Destination	Description - optional
All traffic	All	All	Custom	
Delete				

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



Chrome File Edit View History Bookmarks Profiles Tab Window Help Fri Dec 16 10:05 PM

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateSecurityGroup:

EDU Trading Emigina Films Amarula EDU Kids BOBs Travel Coding FITnazz Kombat Hmm Biznazz Other Bookmarks N. Virginia AwsAce7

AWS Services Search [Option+S]

Inbound rules [Info](#)

Type <a href="#">Info</a>	Protocol <a href="#">Info</a>	Port range <a href="#">Info</a>	Source <a href="#">Info</a>	Description - optional <a href="#">Info</a>	Actions
HTTP	TCP	80	Anywh... <a href="#">Search</a>	HTTP 0.0.0.0/0 <a href="#">X</a>	Delete
SSH	TCP	22	Anywh... <a href="#">Search</a>	Secure Access 0.0.0.0/0 <a href="#">X</a>	Delete
HTTPS	TCP	443	Anywh... <a href="#">Search</a>	HTTPS 0.0.0.0/0 <a href="#">X</a>	Delete

Add rule

Outbound rules [Info](#)

Feedback Looking for language selection? Find it in the new Unified Settings [?](#)

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Chrome File Edit View History Bookmarks Profiles Tab Window Help Fri Dec 16 10:05 PM

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateSecurityGroup:

EDU Trading Emigina Films Amarula EDU Kids BOBs Travel Coding FITnazz Kombat Hmm Biznazz Other Bookmarks N. Virginia AwsAce7

AWS Services Search [Option+S]

Type [Info](#) Protocol [Info](#) Port range [Info](#) Destination [Info](#) Description - optional [Info](#)

All traffic	All	All	Custom <a href="#">Search</a>	0.0.0.0/0 <a href="#">X</a>	Delete
-------------	-----	-----	-------------------------------	-----------------------------	--------

Add rule

**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.

Key  Value - optional  Remove

Add new tag

You can add up to 49 more tags

Cancel Create security group

Feedback Looking for language selection? Find it in the new Unified Settings [?](#)

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the AWS CloudWatch Metrics console showing the creation of a new security group rule.

The URL is <https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateSecurityGroup>.

The page shows a table for defining a new rule. A single row is selected with the IP range `0.0.0.0/0`.

**Add rule**

All traffic	All	All	Custom	0.0.0.0/0	X	Delete
<b>Add rule</b>						

**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.

Key	Value - optional	Remove
Service	Load Balancer	X Remove
APID	3001	X Remove
Engineer	Halo Warhammer	X Remove

**Add new tag**  
You can add up to 47 more tags

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the AWS CloudWatch Metrics console showing the successful creation of a new security group.

The URL is <https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#SecurityGroup:groupId=sg-00a124a8943f68478>.

A green success message indicates the security group was created successfully.

**sg-00a124a8943f68478 - MyWuSGELB**

**Details**

Security group name MyWuSGELB	Security group ID sg-00a124a8943f68478	Description Security Group for Elastic Load Balancer	VPC ID vpc-08c2cc72ad9a850ea
Owner 966520976686	Inbound rules count 3 Permission entries	Outbound rules count 1 Permission entry	

**Inbound rules** | Outbound rules | Tags

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Chrome File Edit View History Bookmarks Profiles Tab Window Help Fri Dec 16 10:06 PM

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#SecurityGroups:

EDU Trading Emigina Films Amaruga EDU Kids BOBs Travel Coding FITnazz Kombat Hmm Biznazz Other Bookmarks N. Virginia AwsAce7

New EC2 Experience Tell us what you think

EC2 Dashboard EC2 Global View Events Tags Limits Instances Instances Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances Dedicated Hosts Scheduled Instances Capacity Reservations

Security Groups (1/6) Info Actions Export security groups to CSV Create security group

Filter security groups

Name	Security group ID	Security group name	VPC ID	Description
WuSGELB	sg-00a124a8943f68478	MyWuSGELB	vpc-08c2cc72ad9a850ea	Security Group
WakandaSecurity	sg-083c88f6311a14765	WakandaSecurity	vpc-0df1cb3cae784a2d8	We will not h
MySecureMoneyWo...	sg-092527783e1f3b856	MySecureMoneyWorld...	vpc-0df1cb3cae784a2d8	MySecureMo
WuSGEC2	sg-0093b683e88329a05	MyWuSGEC2	vpc-08c2cc72ad9a850ea	Security Group

sg-00a124a8943f68478 - MyWuSGELB

Details Inbound rules Outbound rules Tags

You can now check network connectivity with Reachability Analyzer Run Reachability Analyzer

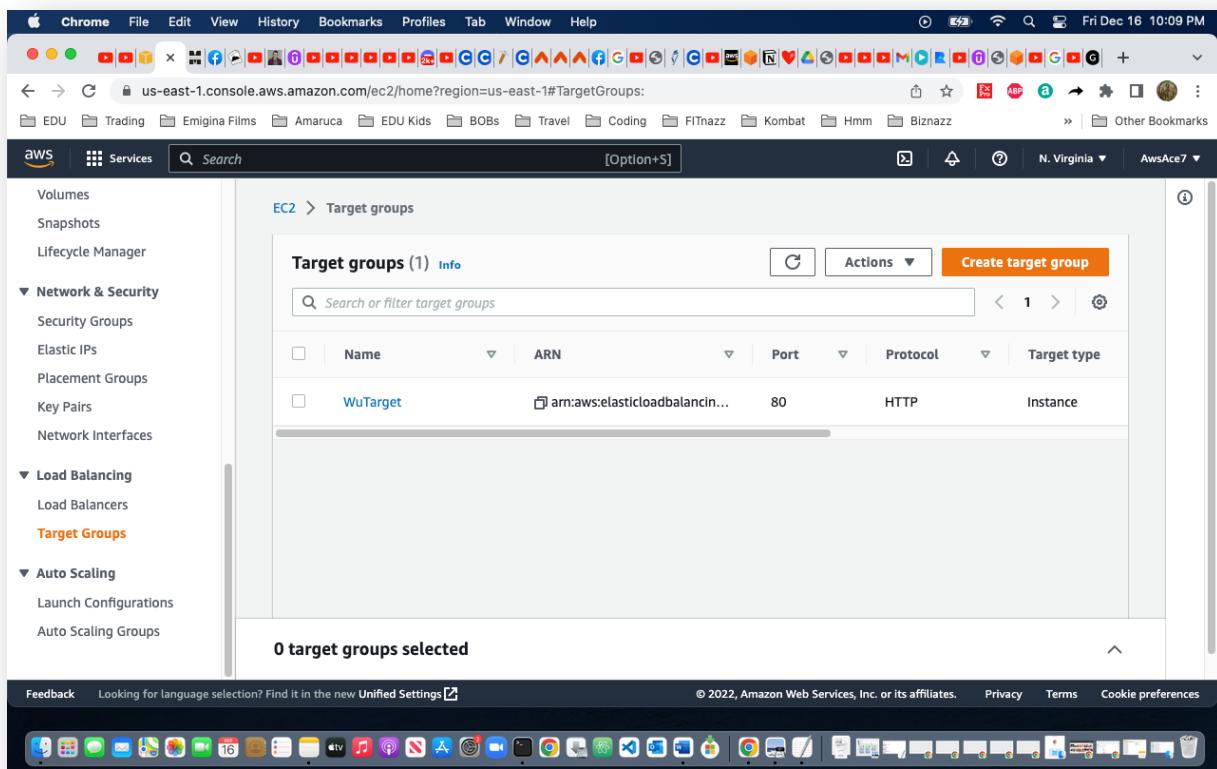
Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## Setting up a Target Group

Now that we have set up the firewalls for both our EC2's and our Load Balancer, we will set up the Target Group to look for or “listen for” web traffic.

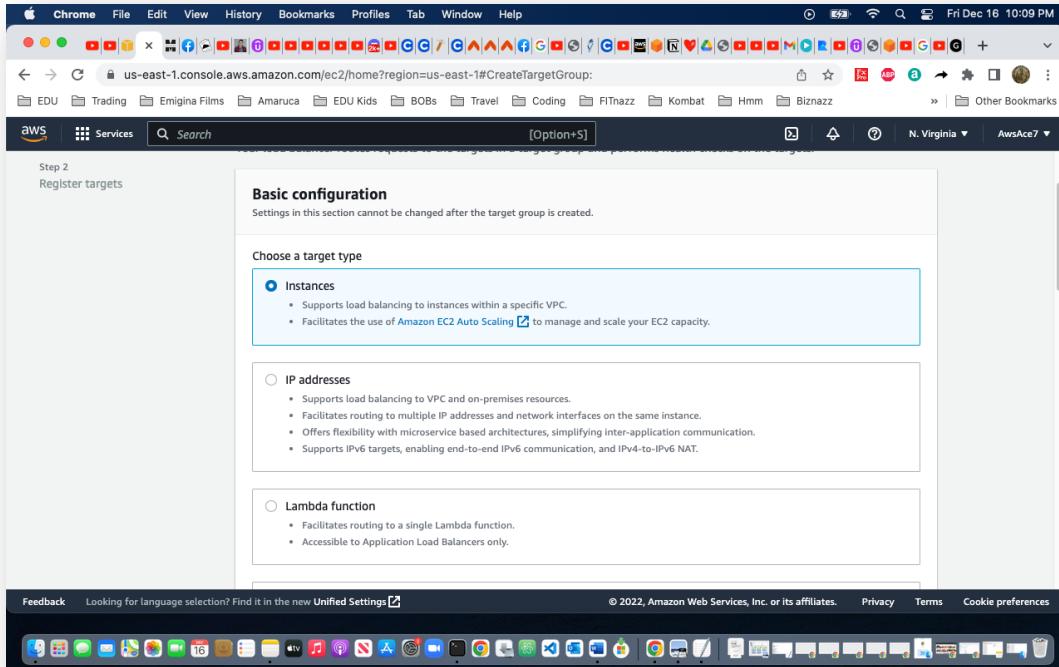
Go to the left side menu and scroll down until you find the “Target Groups” option. Select it.

Once on the Target groups page, go to the right side of the screen and select the “create target group”, button.



The screenshot shows the AWS Lambda console interface. The left sidebar has a tree view with 'Network & Security' expanded, showing 'Target Groups' which is selected and highlighted in orange. The main content area is titled 'Target groups (1) Info'. It displays a single target group named 'WuTarget' with the ARN 'arn:aws:elasticloadbalancing:us-east-1:123456789012:targetgroup/WuTarget/555555555555'. Below the table, a message says '0 target groups selected'. At the bottom of the page, there are links for 'Feedback', 'Language selection', 'Privacy', 'Terms', and 'Cookie preferences'.

For the **Basic Configuration** section, we will select “target type” what we want for our Target group. We select “instances” because it allows up to target our future EC2’s and set up an **Auto Scaling Group**; *a prime feature, that some engineers say is the heart of the AWS cloud platform. We will learn about Auto Scaling Groups later.*



Next, we will give our Target group a name, with no spaces. Leave Protocol as HTTP, port 80 (*we leave it this way because of the TYPE of load balancer we will set up later*).

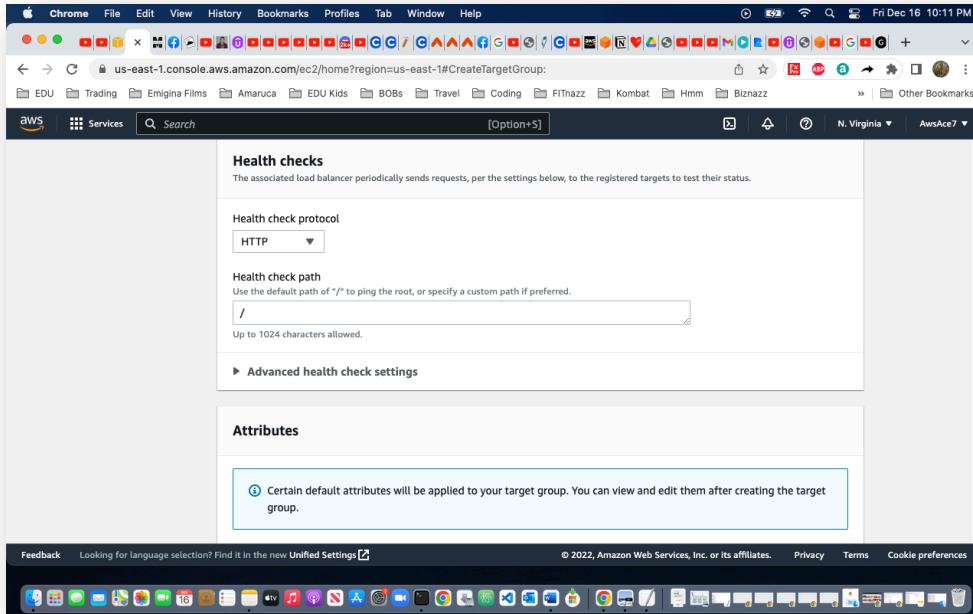
For VPC, we make sure we will our previously made VPC, which is the WuCloud VPC.

The screenshot shows a Chrome browser window on a Mac OS X desktop. The URL is `us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateTargetGroup`. The page is titled "Create target group". The "Target group name" field contains "WuTGT1". The "Protocol" dropdown is set to "HTTP" and the "Port" input is "80". Under "VPC", the "Default VPC" dropdown shows "Default VPC" selected, with the value "vpc-0df1cb3cae78a2d8" and "IPv4: 172.31.0.0/16". The "Protocol version" section has three options: "HTTP1" (selected), "HTTP2", and "gRPC". At the bottom, there are "Feedback", "Privacy", "Terms", and "Cookie preferences" links, along with a Mac OS X dock at the bottom.

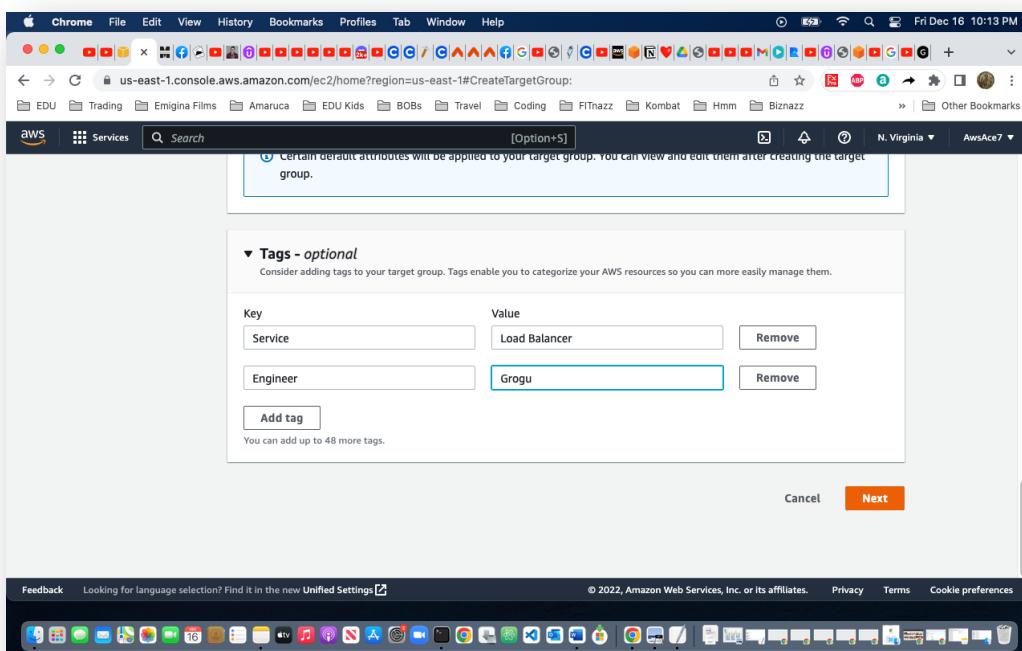
This screenshot is identical to the one above, but the "Default VPC" dropdown now shows "TheWuCloud-vpc" selected, with the value "vpc-08c2cc72ad9a850ea" and "IPv4: 10.136.0.0/16". The rest of the interface and the Mac OS X dock are the same.

For Health Checks, we leave the Health check protocol, “HTTP” as well as the health check path as is. We scroll down to the **Tags** section and again place some reference markers to be able find this device when necessary.

Then select the “Next” button for further configurations.



The screenshot shows the 'Health checks' configuration page. The 'Health check protocol' is set to 'HTTP' and the 'Health check path' is set to '/'. A note indicates that certain default attributes will be applied to the target group. The 'Attributes' section is collapsed.



The screenshot shows the 'Tags - optional' configuration page. Two tags are defined: 'Service' with value 'Load Balancer' and 'Engineer' with value 'Grogu'. An 'Add tag' button is available for additional tags. The 'Cancel' and 'Next' buttons are at the bottom.

For the “Register Targets” page under the **Available instances** section, we leave as is, since we haven’t created any instances also known as EC2’s beforehand.

We go to the **Review Targets**, to check for everything we’ve set up is accurate. Lastly we select the “create target group” button.

This is an optional step to create a target group. However, to ensure that your load balancer routes traffic to this target group you must register your targets.

**Available instances (0)**

Instance ID	Name	State	Security groups	Zone	Subnet ID
No Available Instances					

**0 selected**

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

[Include as pending below](#)

**Review targets**

**Targets (0)**

All	Instance ID	Name	Port	State	Security groups	Zone	Subnet ID
No Instances added yet							

Specify instances above, or leave the group empty if you prefer to add targets later.

0 pending      Cancel      Previous      **Create target group**

As you can see, we've successfully created our Target group *WuTG1*, with our groups details on the following screenshot.

The screenshot shows the AWS EC2 Target Groups page. A green success message at the top says "Successfully created target group: WuTG1". The main table lists two target groups:

Name	ARN	Port	Protocol	Target type
WuTG1	arn:aws:elasticloadbalancing:us-east-1:966320976686:targetgroup/WuTG1/f66bb62b1464ea29	80	HTTP	Instance
WuTarget	arn:aws:elasticloadbalancing:us-east-1:966320976686:targetgroup/WuTarget/1f66bb62b1464ea29	80	HTTP	Instance

Below the table, it says "0 target groups selected".

The screenshot shows the details page for the target group WuTG1. It displays the following information:

Target type	Protocol : Port	Protocol version	VPC
Instance	HTTP: 80	HTTP1	vpc-08c2cc72ad9a850ea

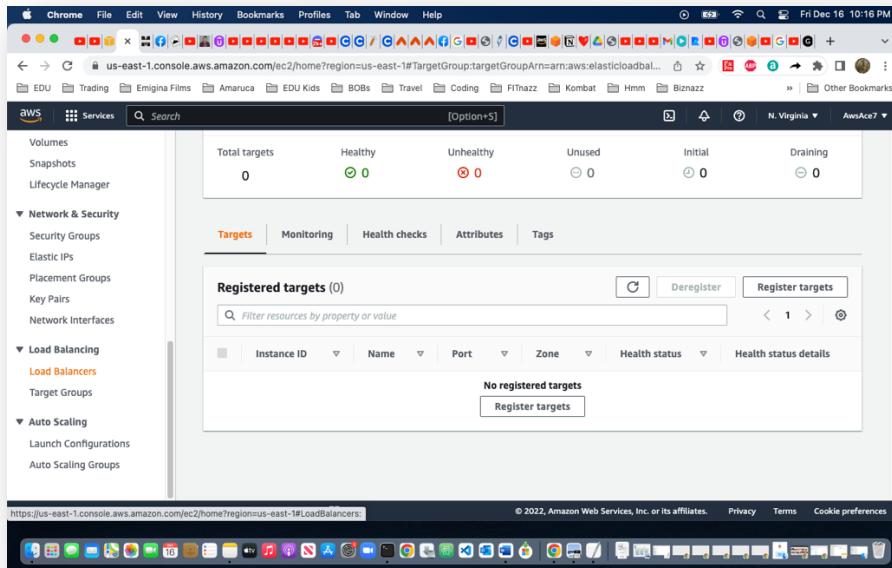
Under "Targets", there are columns for Total targets, Healthy, Unhealthy, Unused, Initial, and Draining. The values are 0, 0, 0, 0, 0, and 0 respectively.

At the bottom, there are tabs for Targets, Monitoring, Health checks, Attributes, and Tags. Below these tabs, there is a section for Registered targets (0) with buttons for Deregister and Register targets.

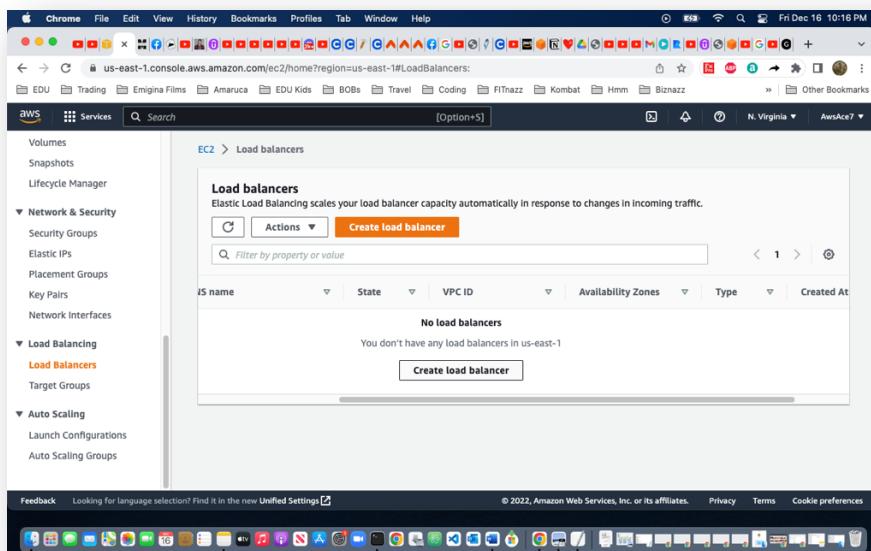
## Setting Up an Elastic Load Balancer

Now with our environment set up, we will begin provisioning our Load Balancer.

Go to the EC2 menu on the left side and scroll until you find the “Load Balancer” option. Select it to go to the Load Balancer page. Once on that page, select the “create load balancer” button.



This screenshot shows the AWS Lambda console. The left sidebar includes options like Volumes, Snapshots, Lifecycle Manager, Network & Security, Load Balancing (with Load Balancers selected), and Auto Scaling. The main content area displays a table for 'Registered targets (0)' with columns for Instance ID, Name, Port, Zone, Health status, and Health status details. A large orange 'Register targets' button is at the bottom. The top navigation bar shows the URL as https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LoadBalancers: and the date Fri Dec 16 10:16 PM.



This screenshot shows the AWS EC2 Load Balancers page. The left sidebar has the same structure as the previous screenshot. The main content area features a heading 'Load balancers' with a sub-note about automatic scaling. It includes a search bar, a 'Create load balancer' button, and a table for managing load balancers. The table has columns for LS name, State, VPC ID, Availability Zones, Type, and Created At. A note at the bottom says 'No load balancers' and 'You don't have any load balancers in us-east-1'. The URL in the address bar is https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LoadBalancers: and the date is Fri Dec 16 10:16 PM.

On the “Select Load balancer type” page, we see several load balancer distinctions. Here I will describe the **3 main** load balancer types:

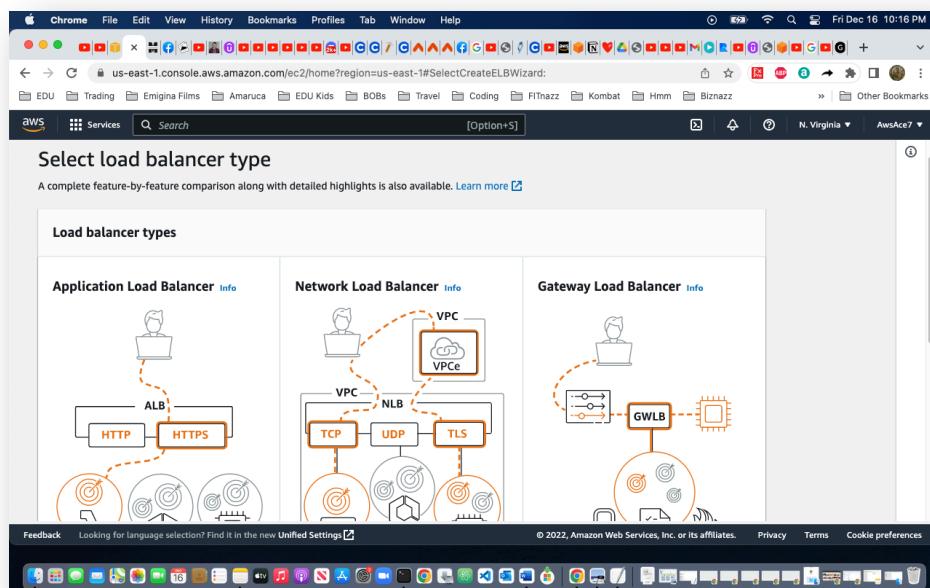
**Application Load Balancer:** this load balancer is designed to receive, send HTTP, and HTTPS web traffic downstream to applications within the VPC. This is the **Application layer or Layer 7** of a computer network.

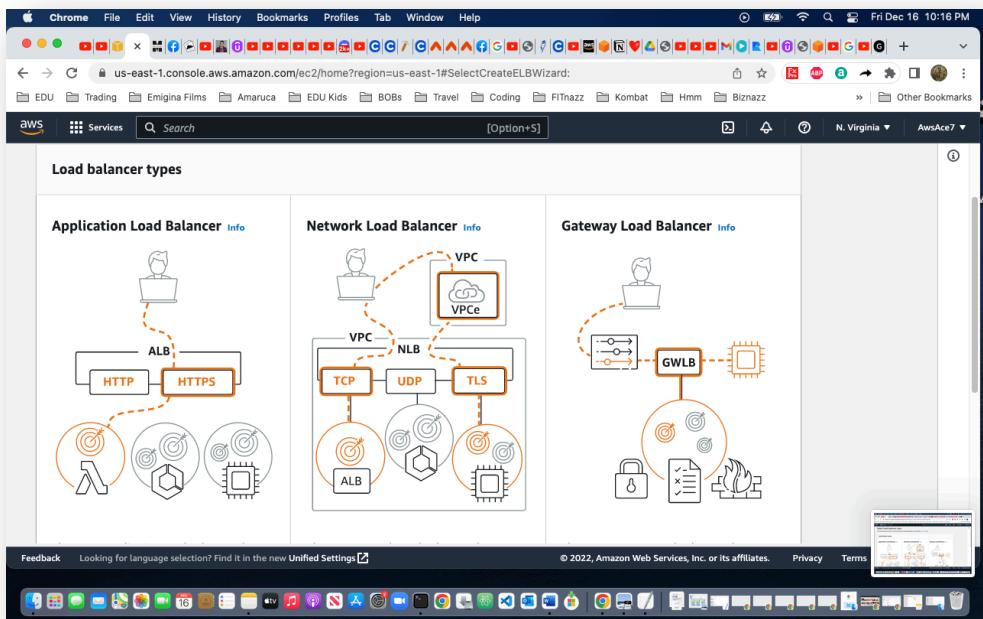
**Note:** This is why in the Target group, we specified the “Protocol”, “Protocol version” and “Health checks” fields as HTTP, because that is the **TYPE** of traffic our load balancer receives and directs to our application. This is the load balancer we will create.

**Network Load Balancer:** this load balancer is designed for **high performance networks** that can handle up to **millions of requests per second**. All of the requests TCP, UDP, TLS traffic can be processed with minimal response time/delay otherwise known as **low latency**. This is the **Transport layer or Layer 4** of a computer network.

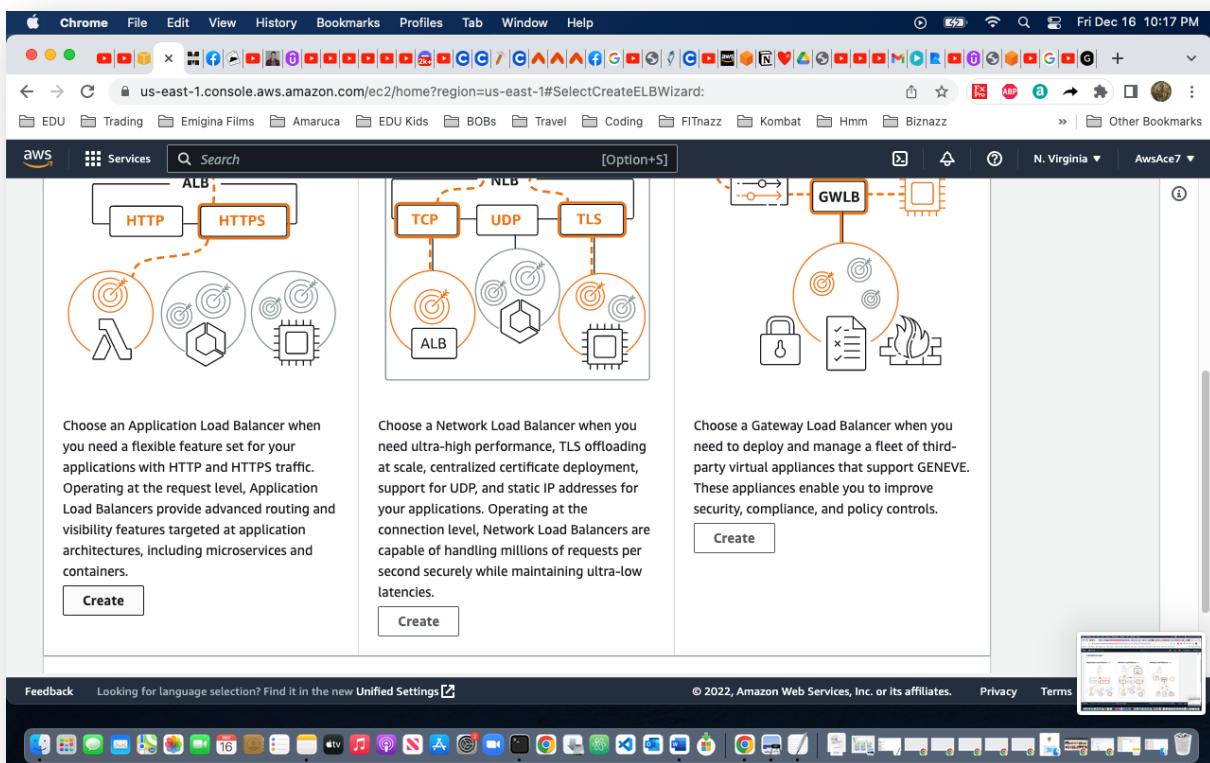
**Gateway Load Balancer:** this load balancer directs incoming traffic to **3<sup>rd</sup> party firewall virtual appliance** that screens/verifies the traffic, **BEFORE** allowing the traffic to reach any application in an network. This is the **Network layer or Layer 3** of a computer network.

The following series of screenshots illustrate the different types of load balancers.

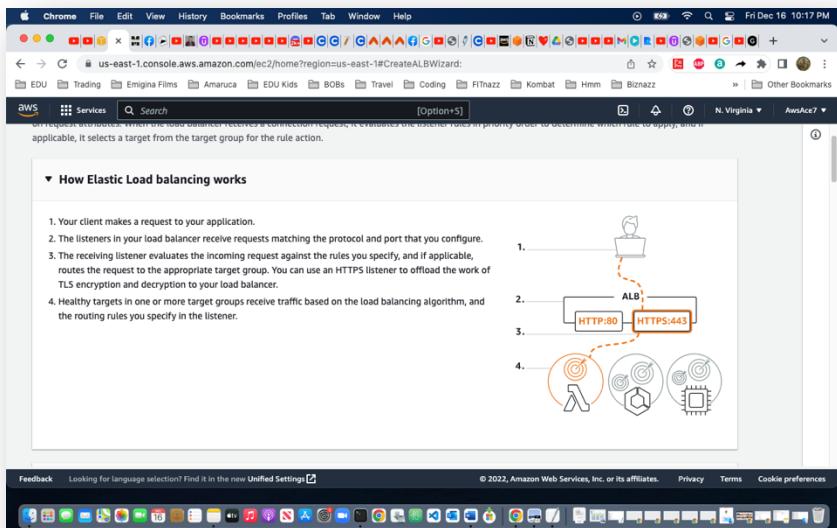




For the application load balancer box, we select the “create” button below.



In the screenshot below, it describes how load balancing works in general.



In the **Basic Configuration** section, we give our ELB a name, “WuELB” as you can see. We select the “internet facing” and “IPv4” choices under the *Scheme* subsection.

The screenshot shows a web browser displaying the 'Create ALB Wizard' on the 'Basic configuration' step. The form includes fields for the load balancer name, scheme, and IP address type.

**Basic configuration**

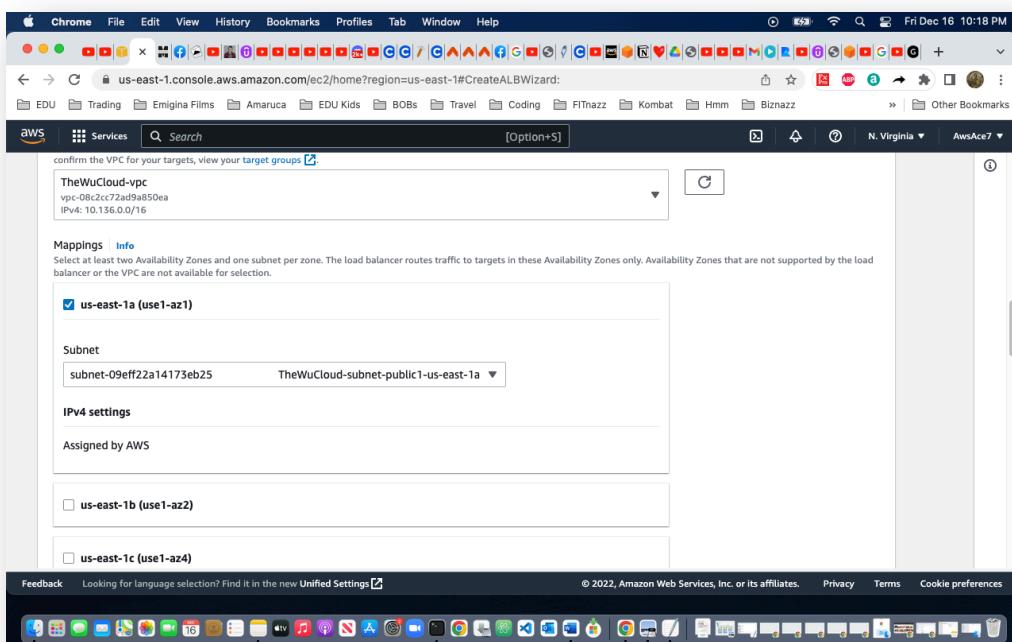
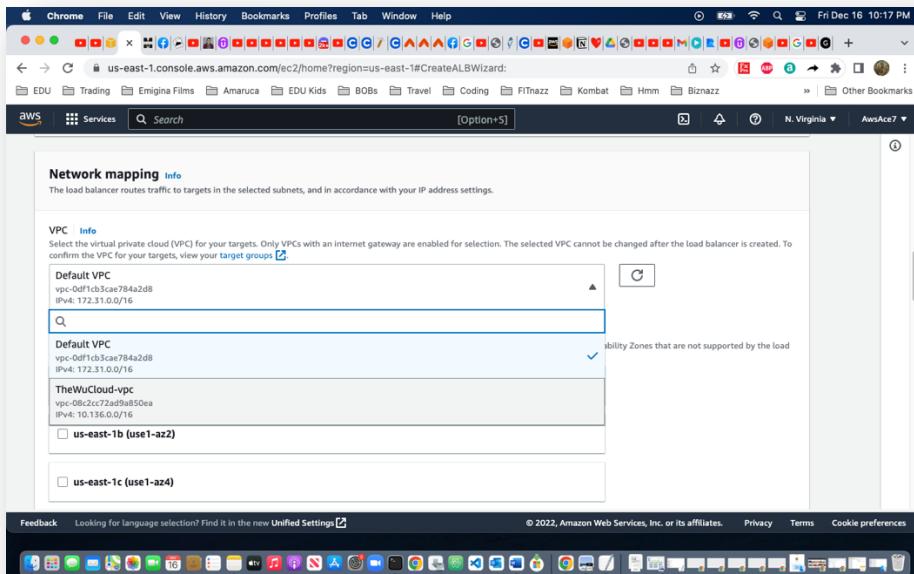
**Load balancer name**  
Name must be unique within your AWS account and cannot be changed after the load balancer is created.

**Scheme** [Info](#)  
Scheme cannot be changed after the load balancer is created.  
 **Internet-facing**  
An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)  
 **Internal**  
An internal load balancer routes requests from clients to targets using private IP addresses.

**IP address type** [Info](#)  
Select the type of IP addresses that your subnets use.  
 **IPv4**  
Recommended for internal load balancers.  
 **Dualstack**  
Includes IPv4 and IPv6 addresses.

For the **Network Mapping** section, we choose our VPC, the WuCloud under the *VPC* subsection.

For the *Mapping*; in each availability zone (us-east-1a, us-east-1b, us-east-1c) we choose the **public subnets**. *The public subnets communicate directly with the internet gateway which allows the HTTP, HTTPS traffic our load balancer will receive and direct to them.*



Chrome File Edit View History Bookmarks Profiles Tab Window Help Fri Dec 16 10:18 PM

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateALBWizard:

EDU Trading Emina Films Amaruka EDU Kids BOBs Travel Coding Fitnazz Kombat Hmm Biznazz Other Bookmarks N. Virginia AwsAce7

aws Services Search [Option+S]

subnet-09eff22a14173eb25 TheWuCloud-subnet-public1-us-east-1a ▾

IPv4 settings  
Assigned by AWS

us-east-1b (use1-az2)

Subnet subnet-070f8501ee1db50a9 TheWuCloud-subnet-public2-us-east-1b ▾

IPv4 settings  
Assigned by AWS

us-east-1c (use1-az4)

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

This screenshot shows the AWS Create ALB Wizard interface. It displays two public subnets: 'us-east-1a' and 'us-east-1b'. Both subnets are assigned by AWS and have their IPv4 settings checked. The 'us-east-1a' subnet is currently selected. The interface includes a feedback bar at the bottom.

Chrome File Edit View History Bookmarks Profiles Tab Window Help Fri Dec 16 10:18 PM

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateALBWizard:

EDU Trading Emina Films Amaruka EDU Kids BOBs Travel Coding Fitnazz Kombat Hmm Biznazz Other Bookmarks N. Virginia AwsAce7

aws Services Search [Option+S]

Subnet subnet-070f8501ee1db50a9 TheWuCloud-subnet-public2-us-east-1b ▾

IPv4 settings  
Assigned by AWS

us-east-1c (use1-az4)

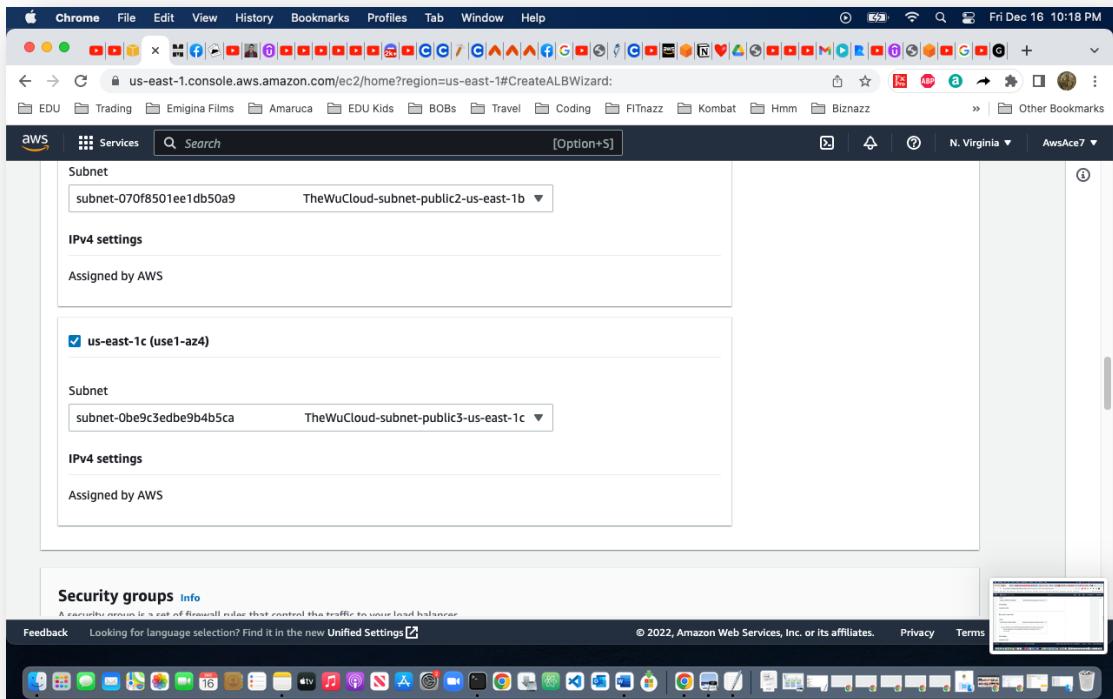
Subnet subnet-0dec91266ec4a0b09 TheWuCloud-subnet-private6-us-east-1c ▾

⚠ The subnet for your internet-facing load balancer must have a route to an internet gateway. You can update the subnet's route table in the VPC Console.

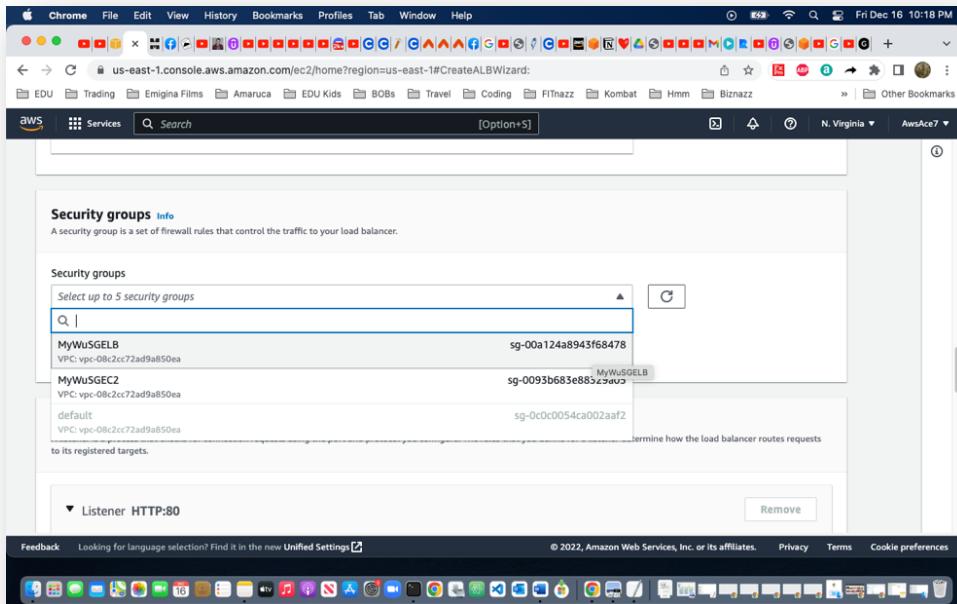
IPv4 settings  
Assigned by AWS

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

This screenshot shows the AWS Create ALB Wizard interface. It displays two subnets: 'us-east-1b' and 'us-east-1c'. The 'us-east-1b' subnet is assigned by AWS and has its IPv4 settings checked. The 'us-east-1c' subnet is also assigned by AWS but has its IPv4 settings unchecked. A warning message is displayed, stating that the subnet for the internet-facing load balancer must have a route to an internet gateway and providing a link to the VPC console. The interface includes a feedback bar at the bottom.



For the **Security Group** section, we select the security group “WuELB” we created earlier, to protect our new load balancer.



The screenshot shows the AWS Lambda console interface. At the top, there's a navigation bar with links like 'File', 'Edit', 'View', 'History', 'Bookmarks', 'Profiles', 'Tab', 'Window', and 'Help'. Below the navigation bar, the URL is `us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#CreateALBWizard`. The main content area has a title 'Security groups' with a 'Info' link. It says, 'A security group is a set of firewall rules that control the traffic to your load balancer.' Below this, there's a section for 'Security groups' with a dropdown menu 'Select up to 5 security groups' and a 'Create new security group' button. A security group named 'MyWuSGELB sg-00a124a8943f68478' is listed, along with its VPC information: 'VPC: vpc-08c2cc72ad9a850ea' and a 'MyWuSGELB' button. Below this, there's a 'Listeners and routing' section with a 'Info' link. It says, '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.' A 'Listener HTTP:80' is listed with a 'Remove' button. At the bottom of the page, there are links for 'Feedback', 'Looking for language selection? Find it in the new Unified Settings', '© 2022, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

Here in the **Listeners and Routing** section, we will instruct our load balancer to listen to the previously made *Target group*, that will send the *HTTP/HTTPS traffic*.

For “Protocol” we leave it *HTTP* and “Port” we leave it 80. For “Default action”, we forward the listening to our target group, which is *WuTG1*.

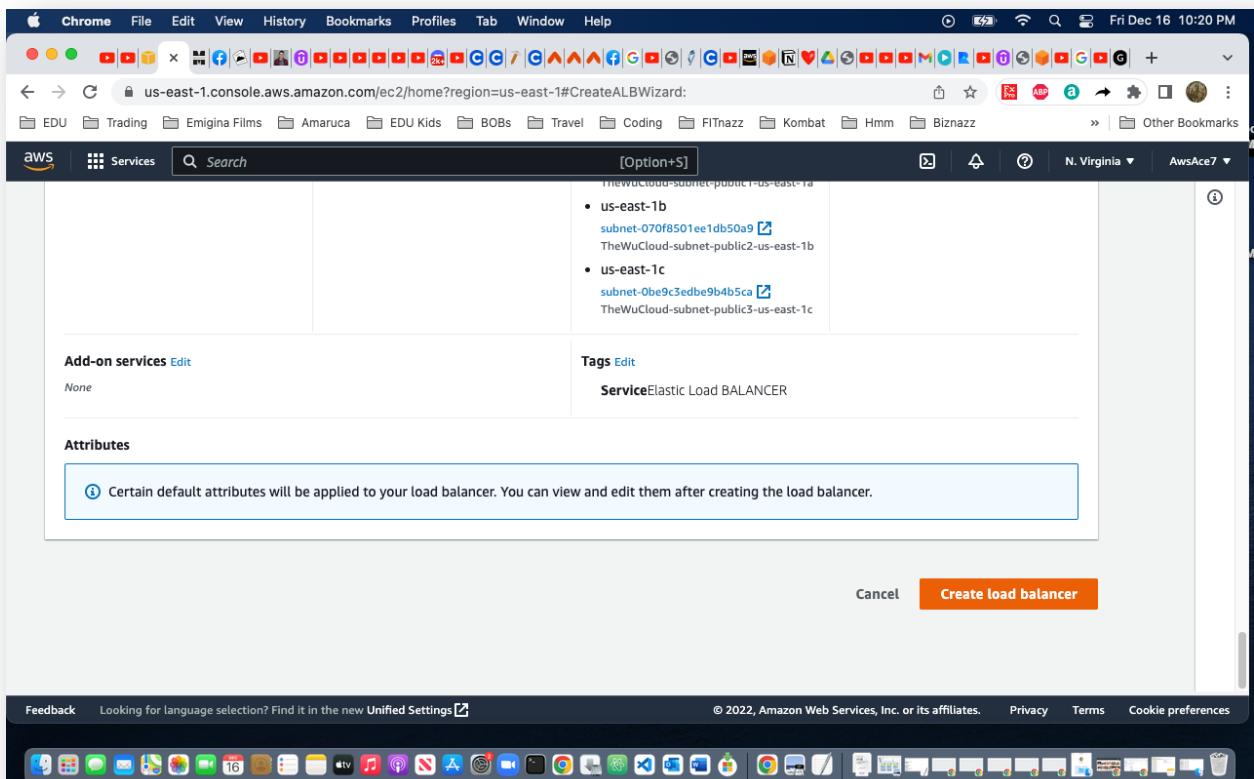
The screenshot shows the AWS Lambda console interface. At the top, there's a navigation bar with links like 'File', 'Edit', 'View', 'History', 'Bookmarks', 'Profiles', 'Tab', 'Window', and 'Help'. Below the navigation bar, the URL is `us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#CreateALBWizard`. The main content area has a title 'Listeners and routing' with a 'Info' link. It says, '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.' A 'Listener HTTP:80' is listed with a 'Remove' button. Below this, there's a table for 'Protocol' and 'Port'. The 'Protocol' dropdown is set to 'HTTP' and the 'Port' dropdown is set to '80'. To the right of the table, there's a 'Default action' section with a 'Info' link. It says, 'Forward to WuTG1 Target type: Instance, IPv4'. A dropdown menu shows 'HTTP' and a 'Create target group' button. Below this, there's a section for 'Listener tags - optional' with a 'Add listener tag' button and a note that you can add up to 50 more tags. At the bottom of the page, there are links for 'Feedback', 'Looking for language selection? Find it in the new Unified Settings', '© 2022, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

For the **Tags** section, we create Key/Value reference markers to be able to find our ELB later, when necessary.

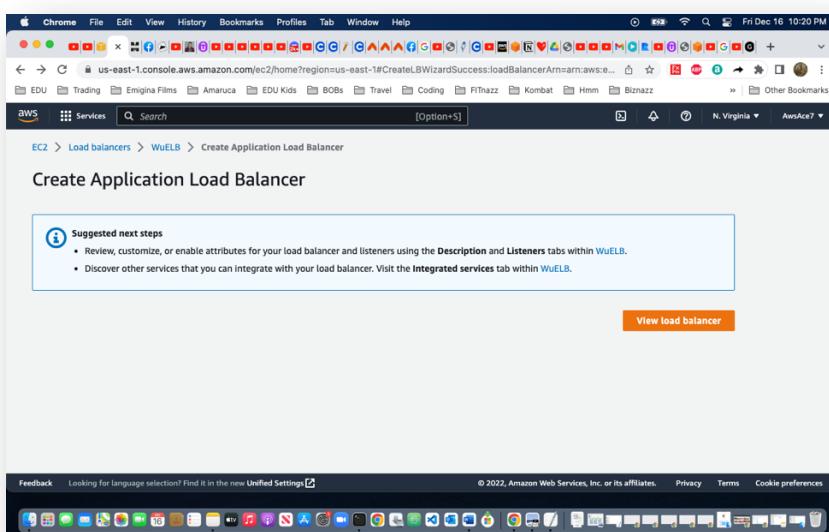
The screenshot shows the 'Tags - optional' section of the AWS CloudFront Create ALB Wizard. It includes a note about adding tags to categorize resources, a key-value pair ('Service' and 'Elastic Load BALANCER'), and a summary section with tabs for Basic configuration, Security groups, Network mapping, and Listeners and routing.

Here in the **Summary**, we make sure all the details of our ELB is accurate before launch. If something is wrong, we can choose to *edit* wherever the error may be. If everything is correct, we select the “create load balancer” button to finish setup.

The screenshot shows the 'Summary' step of the AWS CloudFront Create ALB Wizard. It displays the configuration details: a WUELB (Internet-facing, IPv4), a MyWUESELB security group, a VPC (sg-00a12a8943f68478), and a single listener for port 80. The 'Tags' section shows 'Service' as the value for the 'Elastic Load BALANCER' key. The 'Attributes' section is currently empty.



After a successful load balancer provision, we can select the “view load balancer” button. Once the page we will see our new load balancer “WuELB”. We can then select it to see its details.



The screenshot shows the AWS EC2 Load Balancers page. On the left, there's a sidebar with various EC2-related links like Dashboard, Global View, and Instances. The main area is titled "Load balancers (1)" and contains a table with one row for "WuELB". The table columns include Name, DNS name, State, VPC ID, and Availability Zone. The DNS name is "WuELB-1962638764.us-east-1.elb.amazonaws.com", State is "Provisioning", VPC ID is "vpc-08c2cc72ad9a850ea", and Availability Zone is "3 Availability Zones". A search bar at the top is set to "search: WuELB".

In the **Details** section, we can see all the elements of the load balancer. Especially for the *Availability Zones*, we can see the **public subnets** that will receive the traffic, from the load balancer, once it “listens” to the target group.

This screenshot shows the "Details" page for the load balancer "WuELB". It displays the configuration of the application load balancer. Key details shown include:

- Load balancer type:** Application
- DNS name:** WuELB-1962638764.us-east-1.elb.amazonaws.com (A Record)
- Status:** Provisioning
- VPC:** vpc-08c2cc72ad9a850ea
- IP address type:** IPv4
- Scheme:** Internet-facing
- Availability Zones:** subnets-9eff22a14173eb25 (us-east-1a (use1-az1)), subnets-070f8501ee1db50a9 (us-east-1b (use1-az2)), subnets-0be9c3edbe9b4b5ca (us-east-1c (use1-az3))
- Hosted Zone:** Z355XDOTRQ7X7K

You can also select “Network Mapping” tab under the **details** section to again see the destinations of incoming traffic, the 3 public subnets within the VPC network.

The screenshot shows the AWS Network Mapping page for a load balancer. On the left, there's a sidebar with links like EC2 Dashboard, EC2 Global View, Events, Tags, Limits, Instances, and Capacity Reservations. The main content area has tabs for 'Edit IP address type' and 'Edit subnets'. Under 'VPC', it lists 'vpc-08c2cc72ad9a850ea' with 'IPv4: 10.136.0.0/16' and 'IPv6: -'. To the right, 'IP address type' is set to 'IPv4'. Below this is a 'Mappings' table:

Zone	Subnet	IPv4 address	Private IPv4 address
us-east-1a (use1-az1)	subnet-09eff22a14175eb25	Assigned by AWS	Assigned from CIDR 10.136.0.0/16
us-east-1b (use1-az2)	subnet-070f8501ee1db50a9	Assigned by AWS	Assigned from CIDR 10.136.0.0/16
us-east-1c (use1-az4)	subnet-0be9c3edbe9b4b5ca	Assigned by AWS	Assigned from CIDR 10.136.0.0/16

Lastly, by selecting the “Security” tab you can see the security group our load balancer belongs to.

The screenshot shows the AWS Security Groups page for the same load balancer. The sidebar is identical to the previous screenshot. The main content area has tabs for 'Listeners', 'Network mapping', 'Security' (which is selected), 'Monitoring', 'Integrations', 'Attributes', and 'Tags'. Under 'Security groups (1)', it shows a single entry: 'sg-00a124a8943f68478' with the name 'MyWuSGELB' and the description 'Security Group for Elastic Load Balancer'. A tooltip for the subnet '0be9c3edbe9b4b5ca' is visible above the table.

## **Tearing Down an Elastic Load Balancer**

Now we will tear down or terminate our application load balancer.

When terminating devices/services within the AWS cloud platform it best practice to terminate them in a particular sequence to minimize mistakes, unexpected costs and damages to our account. As of now, we have a VPC network, 2 security groups, target group and a load balancer.

It is best to terminate the load balancer first. The security groups and target group are globally free services, that will NOT be billed to your AWS account, so they can remain.

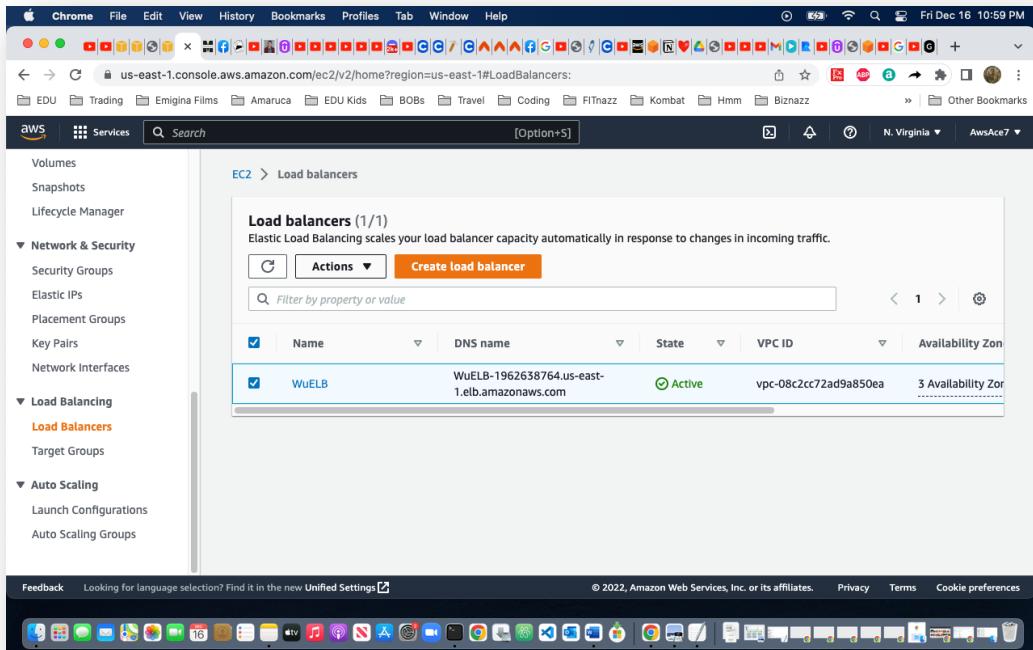
Then the VPC network should be terminated in pieces; the NAT gateways, then the VPC itself, with the final release of the elastic IP's. But the VPC setup and tear down was documented in previous work, so only the load balancer will be deleted here.

**Note:** we will be setting up and terminating **auto scaling groups** soon. This service allows you to designate what type and number of virtual machines you want automatically assembled and disassembled based on the demand of I.T. resources needed at any given time.

*If we had created an auto scaling group, that would be deleted before the load balancer. This is because any virtual machine deleted in the network would be automatically restored by the auto scaling group. To prevent non stop creation/deletion of instances, the auto scaling group would have to be torn down first.*

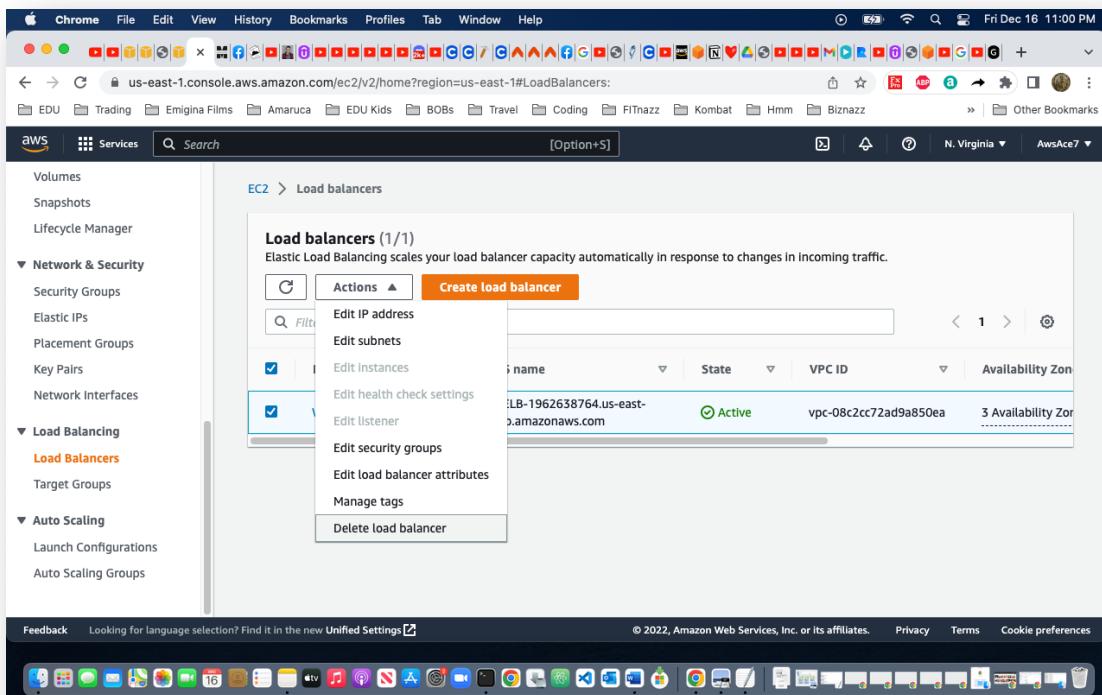
Let us begin the load balancer termination process.

Go to EC2 dashboard menu on the left and scroll to select the “Load Balancers”, option. Check mark the box next to the load balancer you wish to delete. Next, click the drop-down arrow on the “Actions” button.



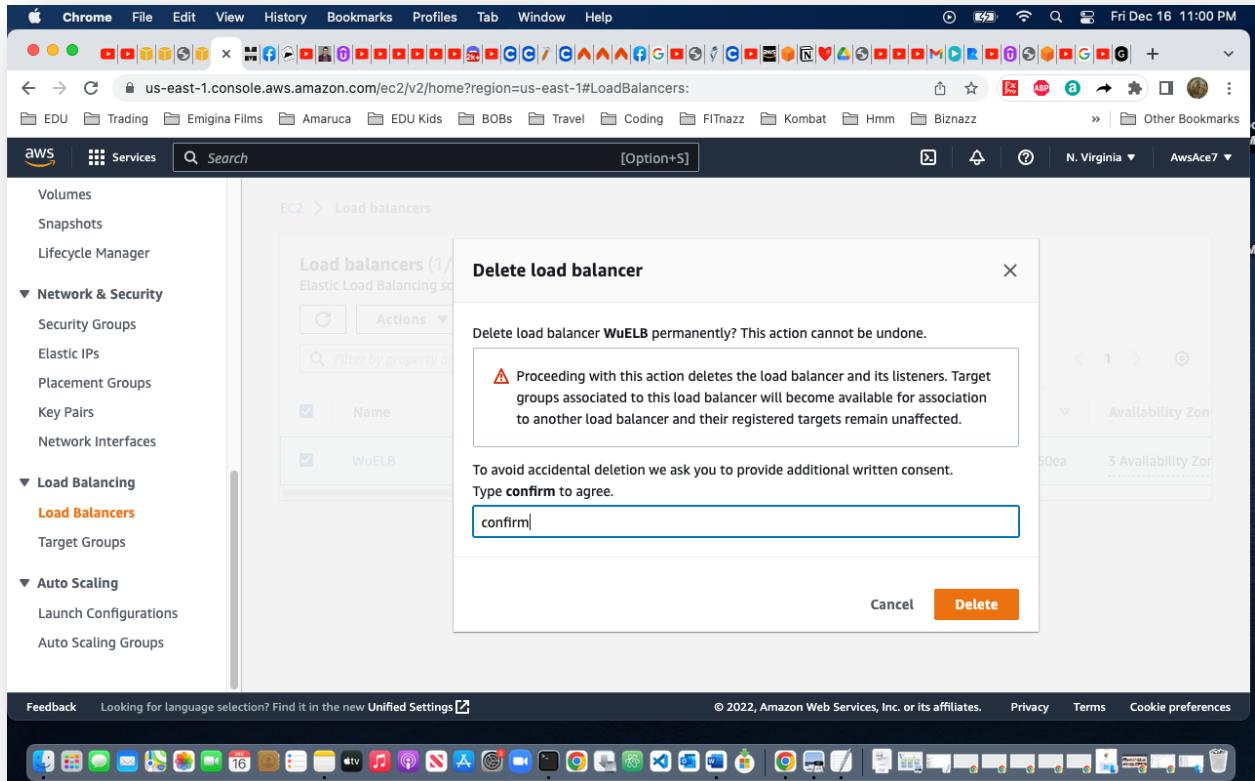
The screenshot shows the AWS EC2 Load Balancers page. On the left sidebar, under the 'Load Balancing' section, the 'Load Balancers' option is selected. In the main content area, there is a table titled 'Load balancers (1/1)'. The table lists one item: 'WuELB' with the details: 'DNS name: WuELB-1962638764.us-east-1.elb.amazonaws.com', 'State: Active', 'VPC ID: vpc-08c2cc72ad9a850ea', and '3 Availability Zones'. The 'Actions' button is visible above the table, and a search bar is present at the top of the table.

Once the menu appears, scroll down to select the “delete load balancer”, option.



The screenshot shows the same AWS EC2 Load Balancers page as before, but with the 'Actions' dropdown menu open. The 'Delete load balancer' option is highlighted with a blue border. The rest of the page remains the same, showing the single load balancer 'WuELB' in the main content area.

A “**Delete load balancer**” dialog box will appear. Type “delete” in the confirmation field. Then select the “delete” button.



The load balancer should be fully terminated in less than a minute. No further actions are needed in this process.

This concludes the setup and tear down of an elastic load balancer on the AWS cloud platform.