

Case study – ELB,ASG AND ROUTE 53

Problem Statement: You work for XYZ Corporation that uses on premise solutions and a limited number of systems. With the increase in requests in their application, the load also increases. So, to handle the load the corporation has to buy more systems almost on a regular basis. Realizing the need to cut down the expenses on systems, they decided to move their infrastructure to AWS

Tasks To Be Performed:

- 1. Manage the scaling requirements of the company by:**
 - a. Deploying multiple compute resources on the cloud as soon as the load increases and the CPU utilization exceeds 80%**
 - b. Removing the resources when the CPU utilization goes under 60%**
- 2. Create a load balancer to distribute the load between compute resources.**
- 3. Route the traffic to the company**

Tasks To Be Performed:

- 1. Manage the scaling requirements of the company by:**
 - a. Deploying multiple compute resources on the cloud as soon as the load increases and the CPU utilization exceeds 80%**

Create 2 EC2 Instance

The screenshot shows the AWS Management Console with the EC2 service selected. The left sidebar is collapsed, and the main area displays the 'Instances' table. The table has columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. Two instances are listed:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
my-case-study-1	i-0e7a33a34cde52b56	Running	t2.micro	Initializing	View alarms	us-east-1c
My-case-study-2	i-004aa3537c74f922a	Running	t2.micro	-	View alarms	us-east-1c

A modal window titled 'Select an instance' is open at the bottom, listing the same two instances.

- In first instance I Installed nginx

By using below command

Sudo su sudo apt-get update

Sudo apt install ngnix

Sudo systemctl start nginx

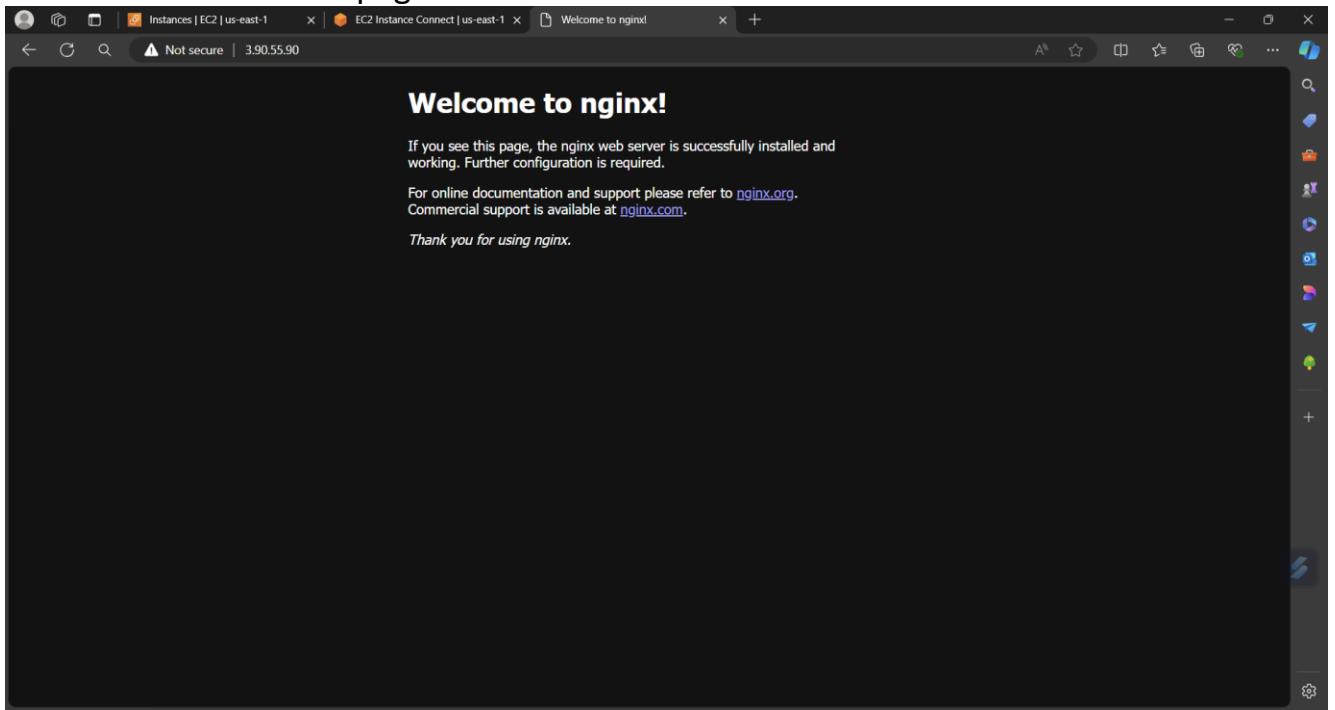
Sudo systemctl status nginx

```
Instances | EC2 | us-east-1 x EC2 Instance Connect | us-east-1 x +  
https://us-east-1.console.aws.amazon.com/ec2-instance-connect/sh?connType=standard&instanceId=i-0e7a33a34cd52b56&osUser=ubuntu&region=us-east-1  
AWS Services Search [Alt+S] Lambda CloudFront DynamoDB  
EFS IAM EC2 CloudWatch Route 53 VPC S3 RDS CloudFormation  
Running kernel seems to be up-to-date.  
No services need to be restarted.  
No containers need to be restarted.  
No user sessions are running outdated binaries.  
No VM guests are running outdated hypervisor (qemu) binaries on this host.  
root@ip-172-31-35-8:~# home/ubuntu# sudo systemctl start nginx  
root@ip-172-31-35-8:~# home/ubuntu# sudo systemctl status nginx  
● nginx.service - A high performance web server and a reverse proxy server  
    Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)  
    Active: active (running) since Fri 2024-07-26 09:45:49 UTC; 52s ago  
      Docs: man:nginx(8)  
   Process: 2032 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master process on; (code=exited, status=0/SUCCESS)  
   Process: 2034 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)  
 Main PID: 2035 (nginx)  
    Tasks: 2 (limit: 1130)  
   Memory: 1.7M (peak: 1.9M)  
     CPU: 13ms  
    CGroup: /system.slice/nginx.service  
           ├─2035 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"  
           └─2036 "nginx: worker process"  
  
Jul 26 09:45:49 ip-172-31-35-8 systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...  
Jul 26 09:45:49 ip-172-31-35-8 systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.  
root@ip-172-31-35-8:~#
```

i-0e7a33a34cde52b56 (my-case-study-1)

Public IPs: 3.90.55.90 Private IPs: 172.31.35.8

- You will find this page



In the same way create 2nd instance

Sudo apt-get update

Sudo apt install apache2

Sudo systemctl start apache2



- Now Create target group

The screenshot shows the AWS Management Console with the EC2 service selected. In the left navigation pane, under the 'Instances' section, 'Target groups' is listed. The main content area displays the 'Target groups' page with a heading 'Target groups info'. A search bar at the top says 'Filter target groups'. Below it is a table header with columns: Name, ARN, Port, Protocol, Target type, and Load balancer. A message 'No target groups' is displayed, followed by 'You don't have any target groups in us-east-1'. At the bottom, there is a large orange button labeled 'Create target group'.

The screenshot shows the 'Step 1 Create target group' wizard. The title bar says 'Step 1 Create target group | EC2'. The main content area has a heading 'Target group name' with an input field containing 'case-study'. Below it, a note says 'A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.' Under 'Protocol : Port', 'HTTP' is selected in a dropdown, and '80' is entered in the port field. A note below says 'Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation.' Under 'IP address type', 'IPv4' is selected with a note explaining it. Under 'VPC', a dropdown menu lists 'vpc-0f35eb05bab1970fd' with the note 'Only VPCs that support the IP address type selected above are available in this list.' At the bottom, there are 'Next Step' and 'Cancel' buttons.

The screenshot shows the AWS CloudWatch Metrics Insights interface. A search bar at the top contains the query: `CloudWatch Metrics Insights / @hour`. The results table below has one row with the following details:

CloudWatch Metrics Insights	CloudWatch Metrics Insights	CloudWatch Metrics Insights	CloudWatch Metrics Insights
CloudWatch Metrics Insights	CloudWatch Metrics Insights	CloudWatch Metrics Insights	CloudWatch Metrics Insights

Below the table, there is a section titled "Metrics for the selected CloudWatch Metrics Insights". It includes a dropdown menu for "Metric Name" set to "CloudWatch Metrics Insights", a dropdown for "Time range" set to "Last hour", and a "Next" button.

- Now launch template

The screenshot shows the AWS EC2 Launch Templates landing page. The top navigation bar includes tabs for Launch templates, EC2 Instance Connect, EC2 Instance Connect, Welcome to nginx!, and Apache2 Ubuntu Default Page. The left sidebar has sections for EC2 Dashboard, EC2 Global View, Events, Console-to-Code (Preview), Instances (selected), Instance Types, Launch Templates (selected), Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images (AMIs, AMI Catalog), and Elastic Block Store (Volumes, Snapshots, Lifecycle Manager). The main content area features a large heading "EC2 launch templates" and sub-headings "Streamline, simplify and standardize instance launches". It explains how launch templates automate instance launches, simplify permission policies, and enforce best practices across an organization. A call-to-action button "Create launch template" is visible. Below this, there's a section titled "Benefits and features" with two items: "Streamline provisioning" and "Simplify permissions". The "Streamline provisioning" section says "Minimize steps to provision instances. With EC2 Auto Scaling, updates to a launch template can be automatically". The "Simplify permissions" section says "Create shorter, easier to manage IAM policies. Learn more". To the right, there's a "Documentation" section with links to "Documentation" and "API reference". The bottom of the page includes footer links for CloudShell, Feedback, Privacy, Terms, and Cookie preferences, along with a copyright notice for 2024, Amazon Web Services, Inc. or its affiliates.

The screenshot shows the 'Create launch template' wizard in the AWS Management Console. The current step is 'Launch template name and description'. The 'Launch template name - required' field contains 'my-case-study-temp'. Below it, a note says 'Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '*', '@'.' The 'Template version description' field contains '234'. Under 'Auto Scaling guidance', there is an info icon and a checkbox for 'Provide guidance to help me set up a template that I can use with EC2 Auto Scaling', which is checked. On the right, the 'Summary' pane shows the selected software image (Canonical, Ubuntu, 24.04 LTS), instance type (t2.micro), security group, and storage (1 volume(s) - 8 GiB). A tooltip for the free tier is displayed, stating: 'Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month; 30 GiB of EBS storage. 2 million I/Os. 1 GB of'. At the bottom are 'Cancel' and 'Create launch template' buttons.

- Now launch Auto scaling

The screenshot shows the 'Create Auto Scaling group' wizard in the AWS Management Console. The current step is 'Choose launch template'. On the left, a sidebar lists steps: Step 1 (Choose launch template), Step 2 (Choose instance launch options), Step 3 - optional (Configure advanced options), Step 4 - optional (Configure group size and scaling), Step 5 - optional (Add notifications), Step 6 - optional (Add tags), and Step 7 (Review). The main area shows a 'Name' section with an 'Auto Scaling group name' field containing 'case-study-ASg'. Below it is a 'Launch template' section with a note: 'For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.' A dropdown menu shows 'my-case-study-temp' as the selected launch template. At the bottom are 'Launch template' and 'Next Step' buttons.

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

EC2 Instance Connect | Instances | EC2 | us-east-1 | EC2 Instance Connect | Welcome to nginx | Apache2 Ubuntu Default | N. Virginia | shreyas bhinge

Services Search [Alt+S]

EFS IAM EC2 CloudWatch Route 53 VPC S3 RDS CloudFormation Lambda CloudFront DynamoDB

Network Info

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC
Choose the VPC that defines the virtual network for your Auto Scaling group.
vpc-0f35eb05bab1970fd 172.31.0.0/16 Default Create a VPC

Availability Zones and subnets
Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.
Select Availability Zones and subnets

us-east-1a | subnet-04729737e84d3d8e5 172.31.80.0/20 Default
us-east-1b | subnet-0989a45b9963ba159 172.31.16.0/20 Default
us-east-1c | subnet-0d2482df5fd86a62 172.31.32.0/20 Default
us-east-1d | subnet-0bc7f43f2b6c80d0f 172.31.0.0/20 Default

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

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

EC2 Instance Connect | Instances | EC2 | us-east-1 | EC2 Instance Connect | Welcome to nginx | Apache2 Ubuntu Default | N. Virginia | shreyas bhinge

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Attach to a new load balancer

Attach to an existing load balancer

No load balancer

Choose from Classic Load Balancers

Choose from Application Load Balancers

HTTP

Only instances within the same VPC as your Auto Scaling group are available for selection.

Select instance groups

On or off-priority load balancing is not yet associated with your load balancer. In order for routing and scaling to occur, you will need to attach the target group to a load balancer. This can be done later in the [Load Balancing console](#).

[Create Auto Scaling group](https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1>CreateAutoScalingGroup) | [EC2 Instance Connect](#) | [Instances | EC2 | us-east-1](#) | [EC2 Instance Connect](#) | [Welcome to nginx!](#) | [Apache2 Ubuntu Default](#)

N. Virginia shreyas bhinge

aws Services Search [Alt+S]

EFS IAM EC2 CloudWatch Route 53 VPC S3 RDS CloudFormation Lambda CloudFront DynamoDB

Health checks

Health checks increase availability by replacing unhealthy instances. When you use multiple health checks, all are evaluated, and if at least one fails, instance replacement occurs.

EC2 health checks

Always enabled

Additional health check types - optional Info

Turn on Elastic Load Balancing health checks Recommended
Elastic Load Balancing monitors whether instances are available to handle requests. When it reports an unhealthy instance, EC2 Auto Scaling can replace it on its next periodic check.

Turn on VPC Lattice health checks
VPC Lattice can monitor whether instances are available to handle requests. If it considers a target as failed a health check, EC2 Auto Scaling replaces it after its next periodic check.

Health check grace period Info

This time period delays the first health check until your instances finish initializing. It doesn't prevent an instance from terminating when placed into a non-running state.

60 seconds

Additional settings

Monitoring Info

Enable group metrics collection within CloudWatch

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

[Create Auto Scaling group](https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1>CreateAutoScalingGroup) | [EC2 Instance Connect](#) | [Instances | EC2 | us-east-1](#) | [EC2 Instance Connect](#) | [Welcome to nginx!](#) | [Apache2 Ubuntu Default](#)

N. Virginia shreyas bhinge

aws Services Search [Alt+S]

EFS IAM EC2 CloudWatch Route 53 VPC S3 RDS CloudFormation Lambda CloudFront DynamoDB

Step 3 - optional Configure advanced options

Step 4 - optional Configure group size and scaling

Step 5 - optional Add notifications

Step 6 - optional Add tags

Step 7 Review

Desired capacity type
Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances)

Desired capacity
Specify your group size.

3

Scaling Info
You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits
Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity	Max desired capacity
1	3

Equal or less than desired capacity Equal or greater than desired capacity

Automatic scaling - optional
Choose whether to use a target tracking policy Info
You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Create Auto Scaling group

EC2 Instance Connect | Instances | EC2 | us-east-1 | EC2 Instance Connect | Welcome to nginx! | Apache2 Ubuntu Default

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

aws Services Search [Alt+S]

EFS IAM EC2 CloudWatch Route 53 VPC S3 RDS CloudFormation Lambda CloudFront DynamoDB

Instance scale-in protection

Instance scale-in protection

Enable instance protection from scale in

Step 5: Add notifications

Notifications

No notifications

Step 6: Add tags

Tags (0)

Key Value Tag new instances

No tags

Cancel Previous Create Auto Scaling group

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Auto Scaling groups | EC2 | EC2 Instance Connect | Instances | EC2 | us-east-1 | EC2 Instance Connect | Welcome to nginx! | Apache2 Ubuntu Default

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

aws Services Search [Alt+S]

EFS IAM EC2 CloudWatch Route 53 VPC S3 RDS CloudFormation Lambda CloudFront DynamoDB

EC2 > Auto Scaling groups

Auto Scaling groups (1) Info

Launch configurations Launch templates Actions Create Auto Scaling group

Search your Auto Scaling groups

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	A...
case-study-ASg	my-case-study-temp Version Default	3	-	3	1	3	us...

0 Auto Scaling groups selected

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS Auto Scaling Groups page for the group 'case-study-ASg'. The 'Automatic scaling' tab is selected. A callout box highlights the note: 'Scaling policies resize your Auto Scaling group to meet changes in demand. With reactive dynamic scaling policies, you can track specific CloudWatch metrics and take action when the CloudWatch alarm threshold is met. Use predictive scaling policies along with dynamic scaling policies in the following situations: when your application demand changes quickly, but with a recurring pattern, or when your EC2 instances require more time to initialize.' Below this, the 'Dynamic scaling policies (0)' section shows a 'Create dynamic scaling policy' button. Further down, the 'Predictive scaling policies (0)' section also has a 'Create predictive scaling policy' button.

The screenshot shows the 'Specify metric and conditions' step of the 'Create alarm' wizard in CloudWatch Metrics. It is Step 1 of 4. The 'Metric' section contains a 'Graph' preview and a 'Select metric' button. Other steps are visible on the left: Step 2 (Configure actions), Step 3 (Add name and description), and Step 4 (Preview and create). At the bottom right are 'Cancel' and 'Next' buttons.

Screenshot of the AWS CloudWatch Metrics console showing the "Select metric" step of creating a new alarm.

The left sidebar shows the navigation path: CloudWatch > Alarms > Create alarm.

The main area is titled "Select metric". It displays a timeline from 07:30 to 10:15 with a highlighted range from 6.53 to 6.76. A checkbox labeled "CPUUtilization" is selected.

The search results table lists metrics from various sources:

Source	Metric Name	Status
awseb-e-55jirny2ah-stack-AWSEBAutoScalingGroup-4mQDolVfLVD3	StatusCheckFailed	No alarms
awseb-e-55jirny2ah-stack-AWSEBAutoScalingGroup-4mQDolVfLVD3	StatusCheckFailed_Instance	No alarms
case-study-ASg	CPUUtilization	No alarms
case-study-ASg	EBSReadOps	No alarms
case-study-ASg	EBSReadBytes	No alarms
case-study-ASg	EBSWriteBytes	No alarms

Buttons at the bottom right include "Cancel" and "Select metric".

Screenshot of the AWS CloudWatch Metrics console showing the "Configure actions" step of creating a new alarm.

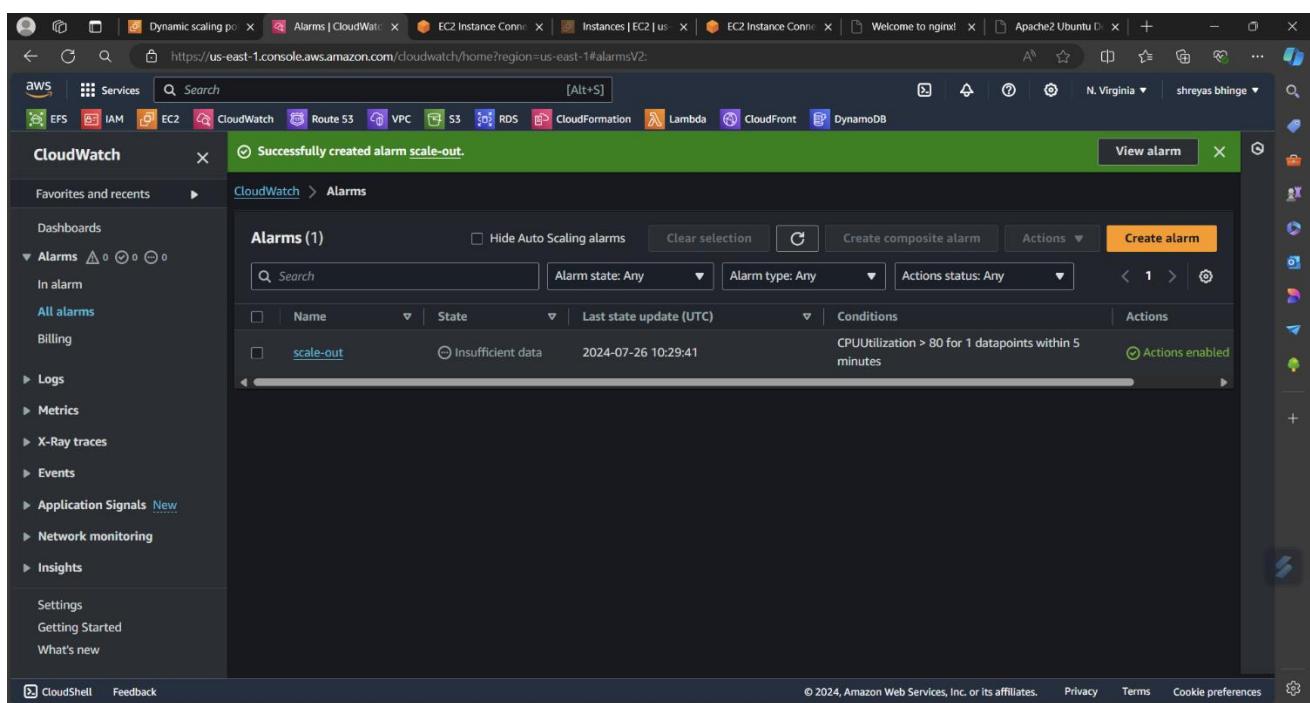
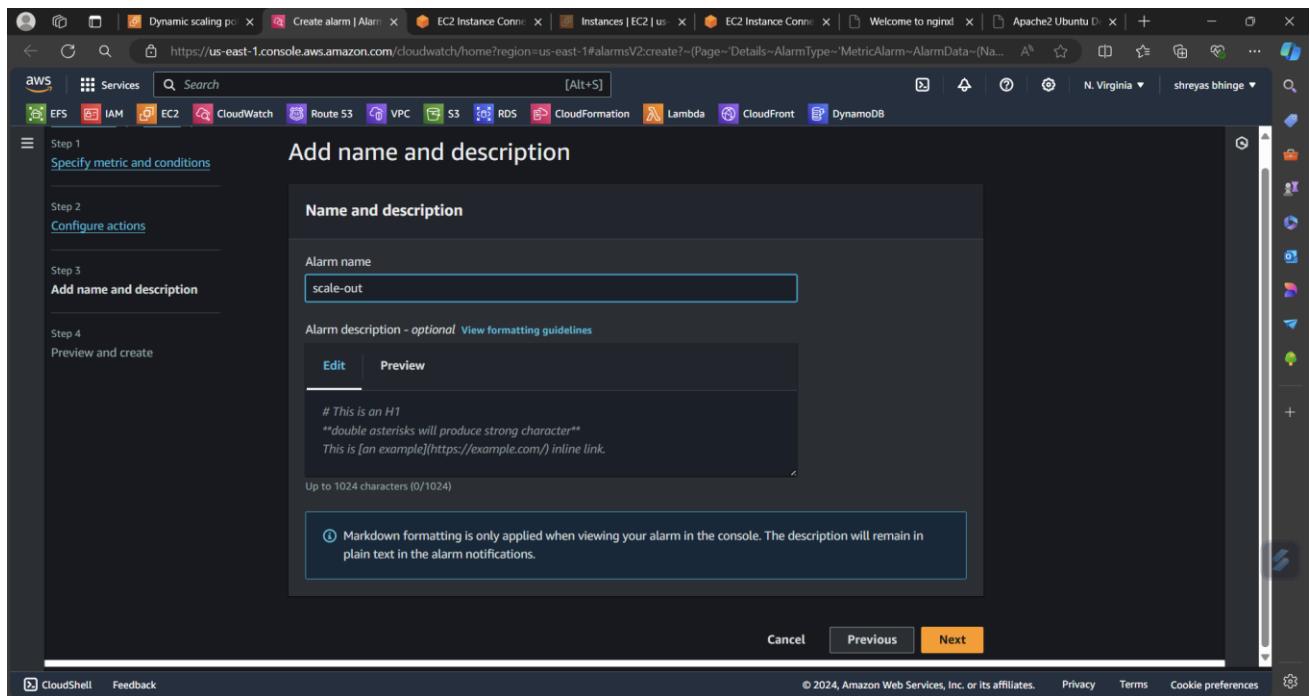
The left sidebar shows the navigation path: CloudWatch > Alarms > Create alarm.

The main area is titled "Notification".

Alarm state trigger: "In alarm" is selected. Other options are "OK" (metric within threshold) and "Insufficient data" (alarm just started or no data).

Send a notification to...: A search bar shows "Default(CloudWatch_Alarms_Topic)". Below it, a note says "Only topics belonging to this account are listed here. All persons and applications subscribed to the selected topic will receive notifications." A "Topic has no endpoints" link and an "Add notification" button are also present.

Lambda action: This section is currently empty.



b. Removing the resources when the CPU utilization goes under 60%

Follow the same steps from dynamic scaling for scale in

The screenshot shows the AWS EC2 Dashboard with a success message: "Dynamic scaling policy created or edited successfully." It lists two scaling policies: "Scale-in" and "dynamic-scaling".

- Scale-in:** Step scaling, Enabled, No alarm selected. Add 0 capacity units when $+\infty \leq \text{Metric name} < 0$. 300 seconds to warm up after each step.
- dynamic-scaling:** Step scaling, Enabled. breaches the alarm threshold: $\text{CPUUtilization} > 80$ for 1 consecutive periods of 300 seconds for the metric dimensions: $\text{AutoScalingGroupName} = \text{case-study-ASG}$. Add 0 capacity units when $80 \leq \text{CPUUtilization} < +\infty$. 300 seconds to warm up after each step.

2. Create a load balancer to distribute the load between the compute resource

- Create a load balancer (Application L.B)

The screenshot shows the "Compare and select load balancer type" page. It compares three types:

- Application Load Balancer**: Handles HTTP and HTTPS traffic, supporting Lambda functions.
- Network Load Balancer**: Handles TCP, UDP, and TLS traffic, supporting VPC and ALB.
- Gateway Load Balancer**: Handles traffic for VPC endpoints, AWS Lambda, and AWS API Gateway.

How Application Load Balancers work

Basic configuration

Load balancer name

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

My-Load-balancer-alb

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme [Info](#)

Scheme can't 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. Compatible with the IPv4 and Dualstack IP address types.

Load balancer IP address type [Info](#)

Select the type of IP addresses that your subnets use. Public IPv4 addresses have an additional cost.

IPv4

Includes only IPv4 addresses.

Dualstack

Includes IPv4 and IPv6 addresses.

Dualstack without public IPv4

Includes a public IPv6 address, and private IPv4 and IPv6 addresses. Compatible with [internet-facing](#) load balancers only.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Security groups [Info](#)

A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can create a new security group.

Security groups

Select up to 5 security groups

default sg-0fe899c4168d8970d VPC: vpc-0f55eb05bab1970fd

Listeners and routing [Info](#)

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

▼ Listener HTTP:80

Protocol Port Default action [Info](#)

HTTP	: 80	Forward to case-study	HTTP
Target type: Instance, IPv4			

Create target group

Listener tags - optional

Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS EC2 Load Balancers console. The left sidebar is collapsed, and the main area displays a table titled 'Load balancers (1)'. The table has columns for Name, DNS name, State, VPC ID, Availability Zones, and Type. One row is visible: 'My-Load-balancer-alb' (DNS name: My-Load-balancer-alb-179..., State: Active, VPC ID: vpc-0f35eb05bab1970fd, 6 Availability Zones, Type: application). Below the table, a message says '0 load balancers selected' and 'Select a load balancer above.'

3. Route traffic to the company

Now go to route 53

Select hosted zone

The screenshot shows the 'Create hosted zone' step in the AWS Route 53 wizard. The 'Hosted zone configuration' section is open. It includes fields for 'Domain name' (myhostedzone.com), 'Description - optional' (bcbscbscbscbsb), and 'Type' (Public hosted zone). A note states: 'A hosted zone is a container that holds information about how you want to route traffic for a domain, such as example.com, and its subdomains.'

Quick create record

Record 1

Record name	.myhostedzoneis.com	Record type	A – Routes traffic to an IPv4 address and some AWS resources
Keep blank to create a record for the root domain.			
Route traffic to	Alias	Alias to Application and Classic Load Balancer	US East (N. Virginia)
<input type="text" value="dualstack.My-Load-balancer-alb-1798026423.us-east-1.elb.amazonaws.com"/>			
Routing policy	Simple routing	Evaluate target health	<input checked="" type="radio"/> Yes

Add another record

Create records

Route 53

myhostedzoneis.com

Hosted zone details

Records (3)

Record ...	Type	Routin...	Differ...	Alias	Value/Route traffic to	TTL (s...)	Health ..
myhosted...	NS	Simple	-	No	ns-1337.awsdns-39.org. ns-1539.awsdns-00.co.uk. ns-515.awsdns-00.net. ns-242.awsdns-30.com.	172800	-
myhosted...	SOA	Simple	-	No	ns-1337.awsdns-39.org. aw...	900	-
www.myh...	A	Simple	-	Yes	dualstack.my-load-balancer-...	-	-

- Hit the Domain after few hours

