

High Availability Across Regions

We can achieve High Availability across multiple Availability Zones using Elastic Load Balancer.

However, if you want to achieve High Availability across different Regions then you can achieve this by using Route 53 service in AWS.

In the last lab, we've configured Highly Availability across different Availability Zones. In this lab, we are going to take this architecture to the next level by implementing High Availability across different AWS Regions.

Now even if entire AWS region gets destroyed due to any catastrophic event, your application will be still up and running 😊

Below is the list of tasks that we are going to perform in this lab.

- Task 1: Create a Web Application Firewall (WAF).
- Task 2: Create CloudFront Distribution with the Source as Elastic Load Balancer (ELB).
- Task 3: Copy the Amazon Machine Image (AMI) to another Region (Mumbai).
- Task 4: Launch a new Windows Server 2016-IIS Web Server in Mumbai region using AMI.
- Task 5: Create a Hosted Zone on Route 53.
- Task 6: Create Health Checks for the Windows Server 2016-IIS Web Server.
- Task 7: Create A records and define Failover routing policies in Route 53.
- Task 8: Test the Failover.

Task 1: Create a Web Application Firewall (WAF).

Web Application Firewall (WAF) is used to protect your application against Distributed Denial of Service (DDoS) attacks. We will make use of this Service to add additional layer of security to our IIS Web Server architecture.

Open the AWS Management Console and navigate to WAF & Shield Service.

The screenshot shows the AWS WAF console landing page. The left sidebar contains the 'WAF & Shield' menu with options like 'AWS WAF', 'Getting started', 'Web ACLs', 'IP sets', 'Regex pattern sets', 'Rule groups', 'AWS Marketplace', 'AWS Shield', and 'AWS Firewall Manager'. The main content area features the 'AWS WAF' header, a description of the service, a 'Get started with AWS WAF' box with a 'Create web ACL' button, and a 'Pricing (US)' section listing costs: '\$5 per web ACL per month (prorated hourly)', '\$1 per rule per month (prorated hourly)', and '\$0.6 per million request processed'. A 'What's new' section is also visible at the bottom.

On right hand side, under Web Application Firewall, click on Web ACLs.

The screenshot shows the 'Web ACLs' page in the AWS WAF console. A blue banner at the top introduces the new AWS WAF console and API experience, noting that the previous version is now named 'AWS WAF Classic'. The page shows the 'Web ACLs' section for the 'US East (N. Virginia)' region. A search bar is present, and a table lists web ACLs. Currently, no web ACLs are found, and a message states: 'You don't have any web ACLs in the US East (N. Virginia) Region created with this latest version of AWS WAF.' A 'Create web ACL' button is visible at the bottom of the table.

Now click on Create Web ACL. Give the Name as of your choice and select Resource Type as CloudFront Distribution and click Next.

Web ACL details

Name

The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and _ (underscore).

Description - optional

The description can have 1-256 characters.

CloudWatch metric name

The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and _ (underscore).

Resource type
Choose the type of resource to associate with this web ACL.

☒ CloudFront distributions

☐ Regional resources (Application Load Balancer and API Gateway)

Region
Choose the AWS region to create this web ACL in.

Global (CloudFront) ▼

Click on Add Rules and select Add Managed Rule Groups.

Add rules and rule groups [Info](#)

A rule defines attack patterns to look for in web requests and the action to take when a request matches the patterns. Rule groups are reusable collections of rules. You can use managed rule groups offered by AWS and AWS Marketplace sellers. You can also write your own rules and use your own rule groups.

Rules

If a request matches a rule, take the corresponding action. The rules are prioritized in order they appear.

EditDeleteAdd rules ▲

<input type="checkbox"/>	Name	Action
No rules.		
You don't have any rules added.		

Expand AWS managed rule groups, and select Windows Operating System rule at the bottom.

Windows operating system

Contains rules that block request patterns associated with exploiting vulnerabilities specific to Windows, (e.g., PowerShell commands). This can help prevent exploits that allow attacker to run unauthorized commands or execute malicious code.

200

- ☒ Add to web ACL
- ☒ Set rules action to count

This rule helps to protect our Windows based EC2 Instances against external Vulnerabilities.

Add rules and rule groups [Info](#)

A rule defines attack patterns to look for in web requests and the action to take when a request matches the patterns. Rule groups are reusable collections of rules. You can use managed rule groups offered by AWS and AWS Marketplace sellers. You can also write your own rules and use your own rule groups.

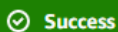
Rules

If a request matches a rule, take the corresponding action. The rules are prioritized in order they appear.

[Edit](#)[Delete](#)[Add rules ▼](#)

<input type="checkbox"/>	Name	Capacity	Action
<input type="checkbox"/>	AWS-AWSManagedRulesWindowsRuleSet	200	Count

Click on Add Rules. Keep default settings for next steps. Finally, review and Click on Create Web ACLs.

**Success**

You successfully created the web ACL: WAF_Web_ACLS

[AWS WAF](#) > [Web ACLs](#)

Web ACLs [Info](#)

Global (CloudFront) ▼

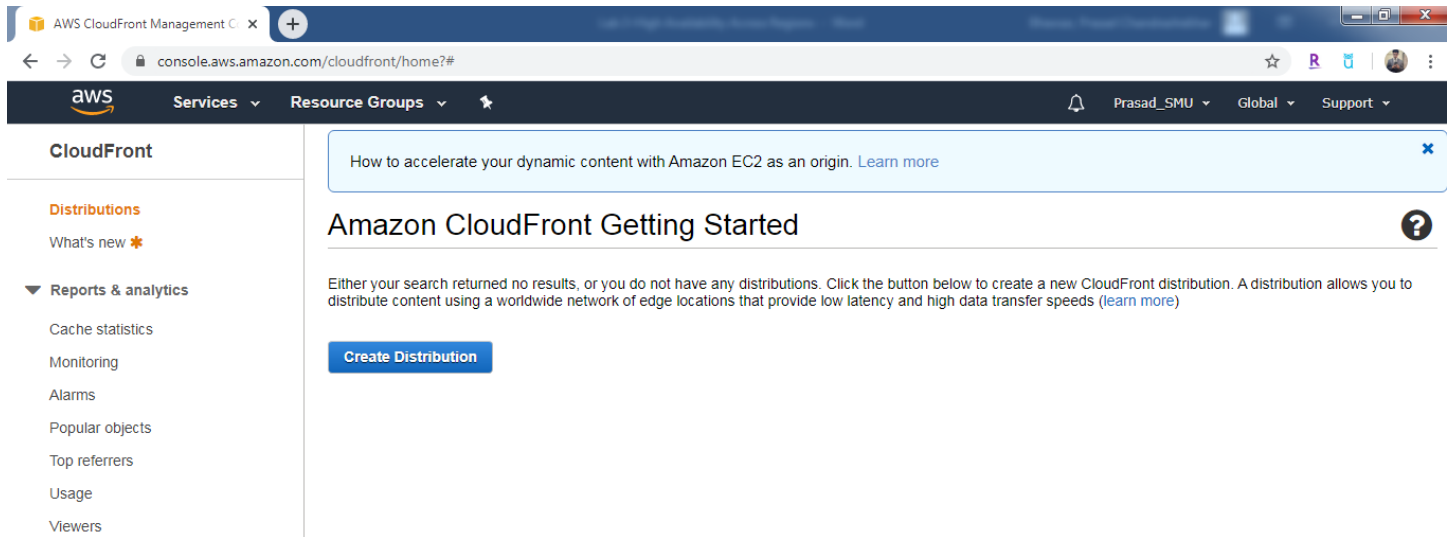
[Delete](#)[Create web ACL](#)

< 1 >

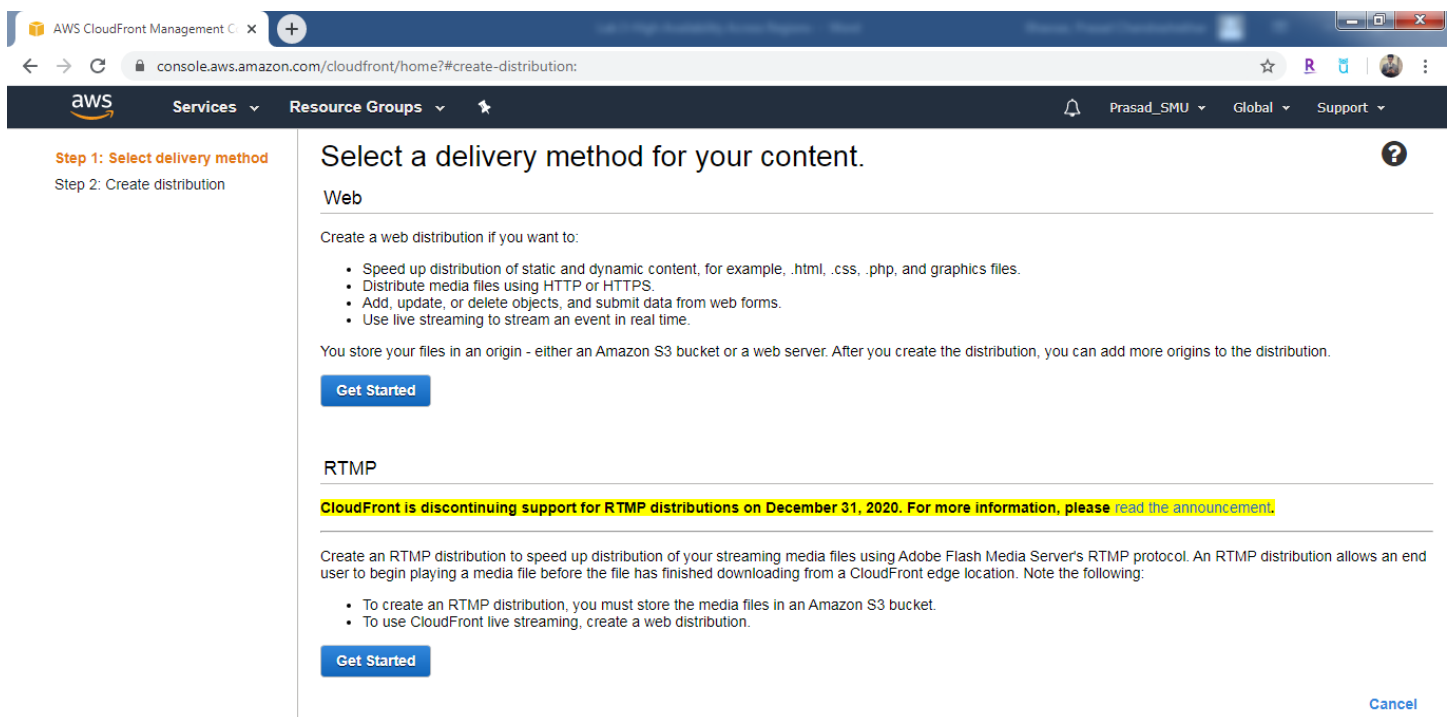
<input type="radio"/>	Name	Description	ID
<input type="radio"/>	WAF_Web_ACLS	-	3c770c78-ab70-4da2-b1cb-e5c4c5b88edd

Task 2: Create CloudFront Distribution.

Navigate to CloudFront Service.








Click on Create Distribution. Since we are going to distribute Web Application over HTTP and HTTPS, under Web select Get Started.



In the Origin Domain Name, select the Elastic Load Balancer that we've configured in the previous lab.

Origin Settings

Origin Domain Name	<input type="text" value="ELB-1731333003.us-east-1.elb.amazonaws.com"/>	
Origin Path	<input type="text"/>	
Origin ID	<input type="text" value="ELB-ELB-1731333003"/>	
Minimum Origin SSL Protocol	<div><input type="radio"/> TLSv1.2 <input type="radio"/> TLSv1.1 <input checked="" type="radio"/> TLSv1 <input type="radio"/> SSLv3</div>	
Origin Protocol Policy	<div><input checked="" type="radio"/> HTTP Only <input type="radio"/> HTTPS Only <input type="radio"/> Match Viewer</div>	


Since the content which is going to get served by CloudFront Edge Location is Static, we are keeping the Default TTL value as 86400 Seconds. It means for the specified TTL value, the content will be cached on the CloudFront Edge Locations.



Minimum TTL	<input type="text" value="0"/>
Maximum TTL	<input type="text" value="31536000"/>
Default TTL	<input type="text" value="86400"/>

In-case of Dynamic Content, we can specify the TTL value as 0 Seconds. It means any request to the CloudFront Distribution will be directly forwarded to specified Source location.

Under Distribution Settings, in AWS WAF Web ACL, specify the Web ACLs which we've configured in the previous task.

Distribution Settings

Price Class Use All Edge Locations (Best Performance) 

AWS WAF Web ACL WAF_Web_ACLs (wafv2)  

Once all the settings are reviewed, click on Create Distribution.

Creation of your distribution on all the CloudFront edge locations takes time. Have some patience and keep observing the status of CloudFront Distribution.

Viewing 1 to 1 of 1 Items					
Comment	Origin	CNAMEs	Status	State	Last Modified
-	ELB-1731333003.us-east-1.elb.amazonaws.com	-	Deployed	Enabled	2020-04-20 02:2

Once the CloudFront Distribution is created, click on Distribution ID. Under General Tab, verify the CloudFront Distribution Domain Name, WAF Web ACL etc.

General

Origins and Origin Groups

Behaviors

Error Pages

Restrictions

Invalidations

Tags

Edit

Distribution ID ETS6R1DIEBILV

ARN arn:aws:cloudfront::616399057974:distribution/ETS6R1DIEBILV

Log Prefix -


Delivery Method Web

Cookie Logging Off

Distribution Status Deployed

Comment -


Price Class Use All Edge Locations (Best Performance)

AWS WAF Web ACL WAF_Web_ACLs (wafv2) 

State Enabled

Alternate Domain Names (CNAMEs) -

SSL Certificate Default CloudFront Certificate (*.cloudfront.net)

Domain Name d3fa3qs2pqzc.cloudfront.net 

Custom SSL Client Support -

Now click on Origins and Origin Groups and verify the Source Location which is the Elastic Load Balancer (ELB).

General Origins and Origin Groups Behaviors Error Pages Restrictions Invalidations Tags

Origins

Create Origin Edit Delete

Origin Domain Name and Path	Origin ID	Origin Type	Origin Access Identity	Origin Protocol
<input type="checkbox"/> ELB-1731333003.us-east-1.elb.amazonaws.com	ELB-ELB-1731333003	Custom Origin	-	HTTP Only

Now Copy the CloudFront Distribution Name and put it in your Browser.

AWS CloudFront Management Console IIS Windows Server

Not secure | d3fa3qs2pqzc.cloudfront.net

Windows Server

Hello Prasad!
Congratulations, your application is running perfectly fine!!!!

Internet Information Services

Welcome Bienvenue Tervetuloa

ようこそ Benvenuto 歡迎

Bem-vindo Bienvenido Hoş geldiniz ברוכים הבאים Welkom

Καλώς ορίσαστε Vitejte Valkommen 환영합니다 Добро пожаловать Üdvözljük

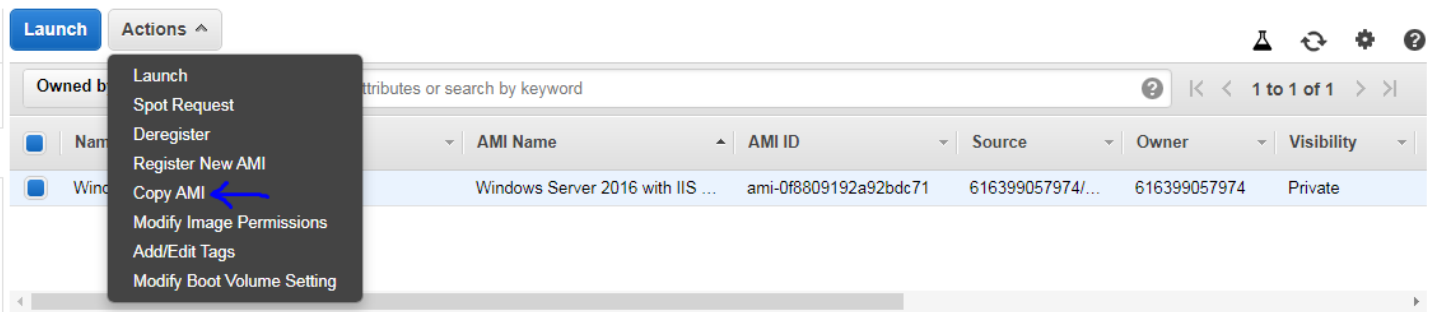
Microsoft Willkommen Velkommen Witamy

Website Opens, it means your application is now served by the CloudFront Distribution through Cache on Edge Locations. The resources utilization of base Infrastructure is now very minimal due to caching; hence we've now achieved Performance Efficiency successfully!!!!!!

Task 3: Copy the Amazon Machine Image (AMI) to another Region (Mumbai).

We have one Amazon Machine Image (AMI) ready for the deployment of Windows Server 2016 IIS Web Server in US East (N. Virginia) region. We are now going to copy the same AMI Image to a new region Asia Pacific (Mumbai) and then we will deploy identical Web Server in Asia Pacific (Mumbai) region.

Navigate to EC2 Service and click on AMIs. Select the existing AMIs, click on Actions and click on Copy.



Select the Destination Region as Asia Pacific (Mumbai).

Copy AMI

AMI ami-0f8809192a92bdc71 (Windows Server 2016 with IIS Image) will be copied to a new AMI. Set the new AMI settings below.

Destination region* Asia Pacific (Mumbai)

Name Windows Server 2016 with IIS Image

Description [Copied ami-0f8809192a92bdc71 from us-east-1] Windows Se

Encryption ☐ Encrypt target EBS snapshots ⓘ

Cancel Copy AMI

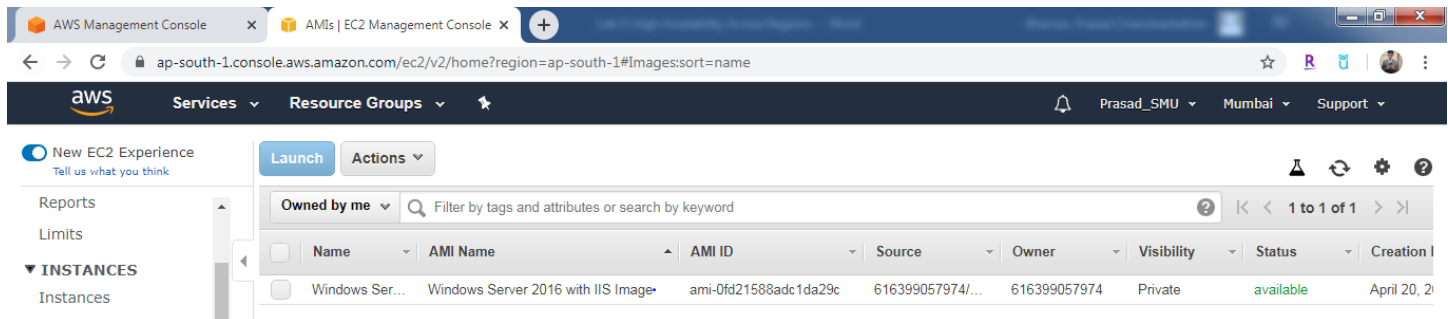
Click on Copy AMI. This will now deploy a new Amazon Machine Image (AMIs) in Asia Pacific (Mumbai) region.

Task 4: Launch a new Windows Server 2016-IIS Web Server in Mumbai region using AMI.

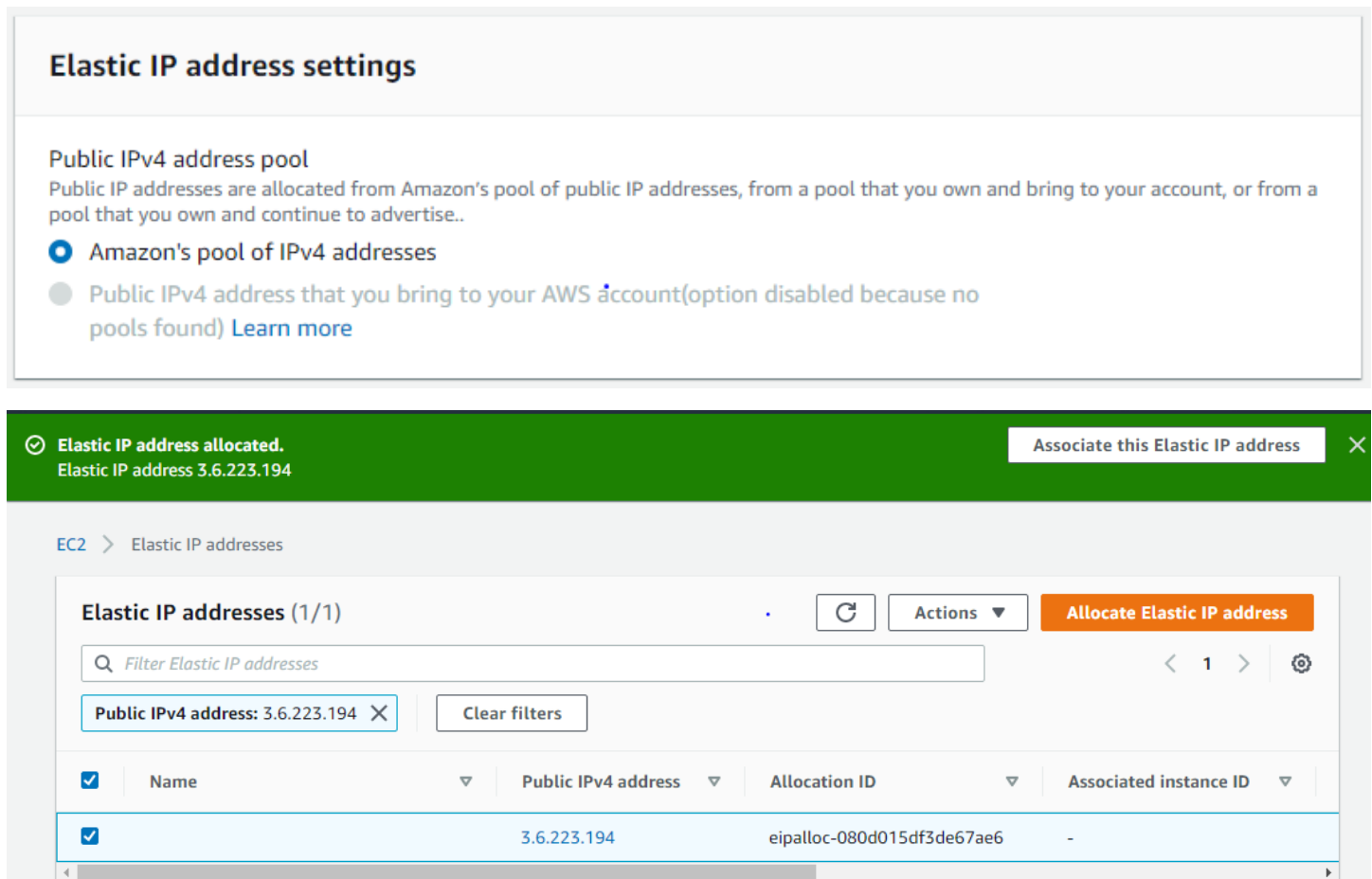
Now change the AWS region to Asia Pacific (Mumbai).

Navigate to EC2 Service and click on AMIs.

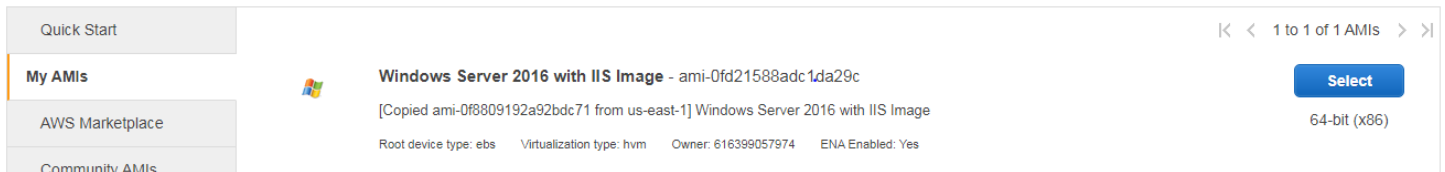
You'll notice that the AMI is now successfully copied to Asia Pacific (Mumbai) region from US East (N. Virginia) region.



Navigate to Elastic IPs and Allocate an Elastic IP for your EC2 Instance from Amazon's pool of IPv4 addresses.

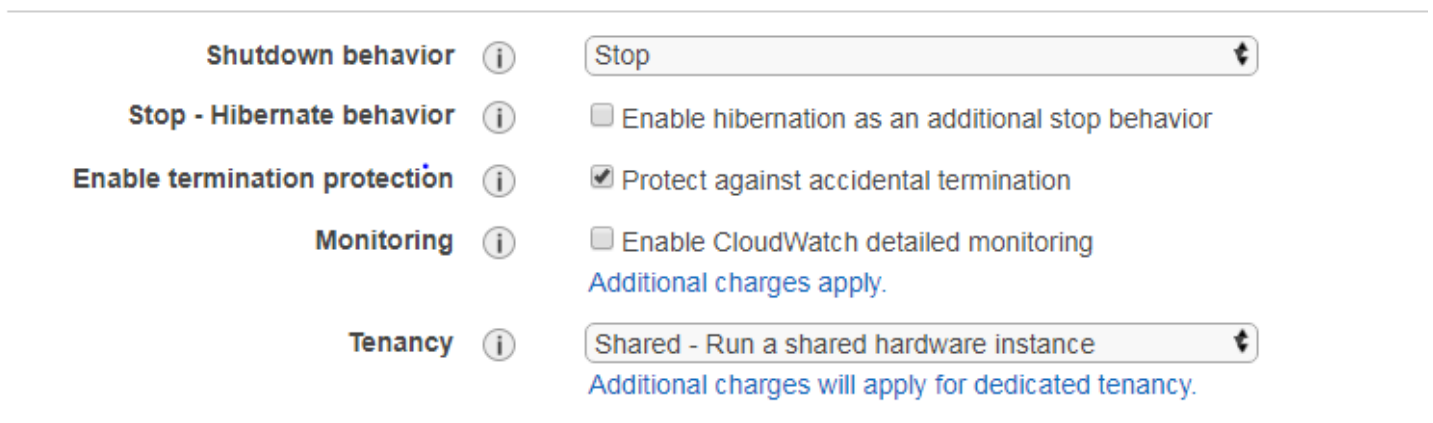


Navigate to Instances and click on Launch Instance. Click on My AMIs and select the existing AMI.



In Step 2: Choose Instance Type, keep the default Instance type as t2.micro and click Next.

In Step 3: Configure Instance Details, keep the Network as Default VPC and Subnet as of no preference. Make sure to select **enable protect against accidental termination** and **Shutdown behavior** as Stop and click Next.



In Step 4: Add Storage, keep the default EBS storage settings.

In Step 5: Add Tags, you can add Tags If you wish else click Next.

In Step 6: Configure Security Group, since we haven't configured any Security Group, you can either click a new Security Group for the Wen Server or click Next.

In Step 7: Review Instance Launch, review the configurations and click on Launch.

Here you can select proceed without a Key Pair since we are not going to take RDP of this Instance and click on Launch Instance.

Select an existing key pair or create a new key pair



A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Proceed without a key pair

☒ I acknowledge that I will not be able to connect to this instance unless I already know the password built into this AMI.

Cancel

Launch Instances

Wait till the EC2 Instance passes the 2/2 Status Check and Instance State as Running.

<input type="checkbox"/>	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status
<input type="checkbox"/>	Mumbai Server	i-066243cb3b8932c1f	t2.micro	ap-south-1a	running	2/2 checks ...	None

Once the Application Server is deployed, go back to Elastic IPs and click on the Allocated IP Address and click on Actions. Click on Associate Elastic IP Address.

Elastic IP addresses (1/1)

Public IPv4 address: 3.6.223.194

Clear filters

<input checked="" type="checkbox"/>	Name	Public IPv4 address	Allocation ID	Instance ID
<input checked="" type="checkbox"/>		3.6.223.194	eipalloc-080d015df3de67ae6	-

Actions

View details

Release Elastic IP addresses

Associate Elastic IP address

Disassociate Elastic IP address

Allocate Elastic IP address

1 > ⚙


Select the Instance that you've recently launched and click on Associate.

Elastic IP address: 3.6.223.194




Resource type
Choose the type of resource with which to associate the Elastic IP address.

☒ Instance


☐ Network interface

 If you associate an Elastic IP address to an instance that already has an Elastic IP address associated, this previously associated Elastic IP address will be disassociated but still allocated to your account. [Learn more.](#)

Instance



 i-066243cb3b8932c1f  

Private IP address
The private IP address with which to associate the Elastic IP address.






 Choose a private IP address

Reassociation
Specify whether the Elastic IP address can be reassociated with a different resource if it already associated with a resource.

☐ Allow this Elastic IP address to be reassociated

You can now verify that the Elastic IP Address has been assigned to the Web Server running in the Asia Pacific (Mumbai) region.

	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status
	Mumbai Server	i-066243cb3b8932c1f	t2.micro	ap-south-1a	 running	 2/2 checks ...	None 

Instance: **i-066243cb3b8932c1f (Mumbai Server)** Elastic IP: **3.6.223.194**

Description

Status Checks


Monitoring

Tags

Instance ID i-066243cb3b8932c1f

Instance state running

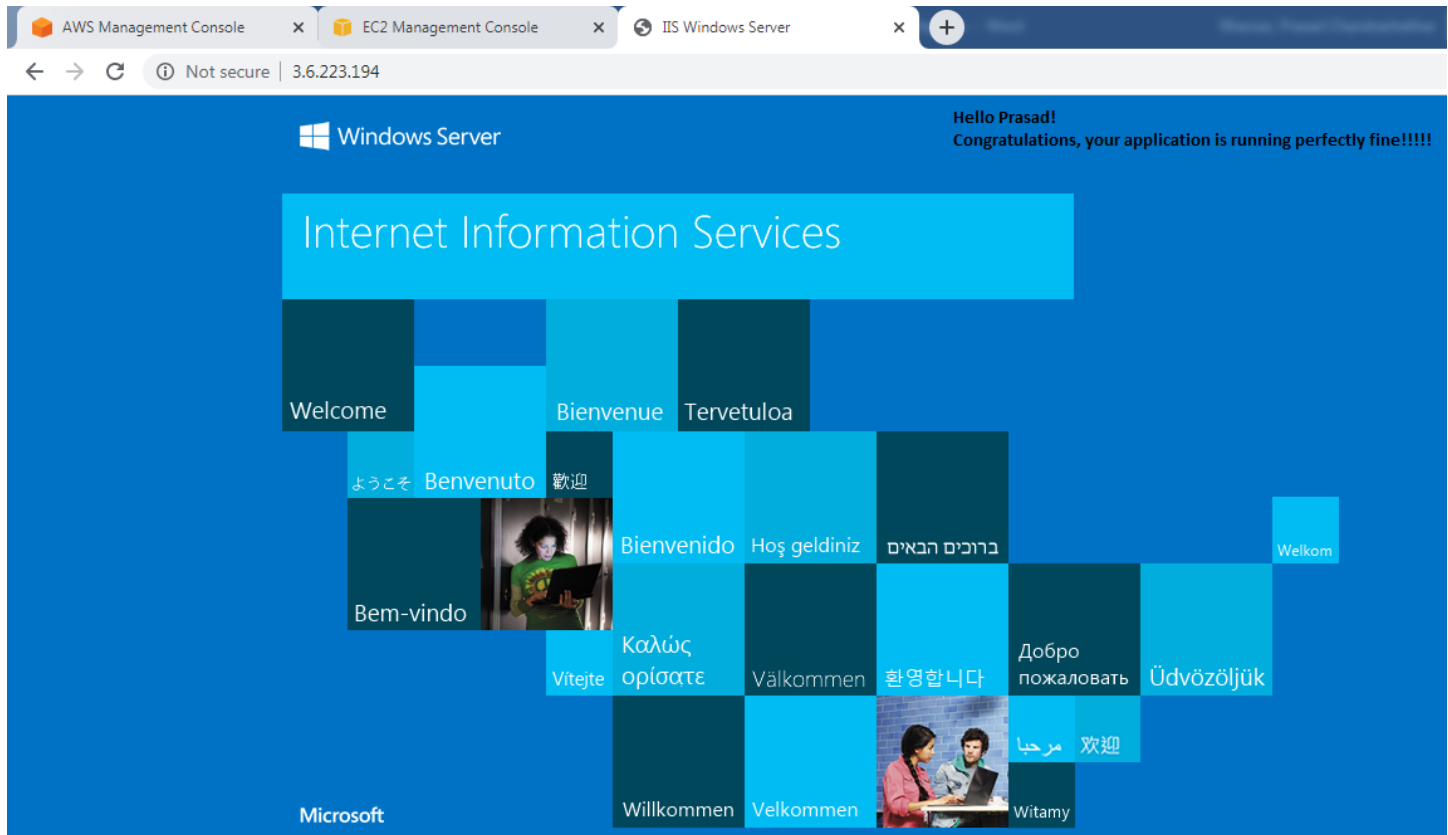
Public DNS (IPv4)

ec2-3-6-223-194.ap-south-1.compute.amazonaws.com 

IPv4 Public IP

3.6.223.194

Copy the EC2 Instance IP Address and put it in the browser, your application opens.



NOTE: In-case if the Web Page doesn't open then verify the Security Group associated with it. It means your application is now also running on different Region which is Asia Pacific (Mumbai). In the Summary, your application is running in two different regions as follows.

Asia Pacific (Mumbai): Application is hosted on a standalone EC2 Instance.

US East (N. Virginia): Application is running on Infrastructure wherein you've configured Elastic Load Balancer, Autoscaling etc.

Now in the next tasks, you will configure Asia Pacific (Mumbai) as the Primary and US East (N. Virginia) as the Secondary region to run the Application.

In-case if the entire Asia Pacific (Mumbai) region goes down, your application will be still up through the US East (N. Virginia). This is what we called as High Availability Across Multiple Regions.

Task 5: Create a Hosted Zone on Route 53.

Come back to the Original Region (N. Virginia).

Navigate to Route 53 Service and under DNS Management click on Get Started.

Amazon Route 53

You can use Amazon Route 53 to register new domains, transfer existing domains, route traffic for your domains to your AWS and external resources, and monitor the health of your resources.

DNS management

If you already have a domain name, such as example.com, Route 53 can tell the Domain Name System (DNS) where on the Internet to find web servers, mail servers, and other resources for your domain.
[Learn More](#)

[Get started now](#)

Traffic management

Route 53 traffic flow provides a visual tool that you can use to create and update sophisticated routing policies to route end users to multiple endpoints for your application.
[Learn More](#)

[Get started now](#)

Availability monitoring

Route 53 can monitor the health and performance of your application as well as your web servers and other resources. Route 53 can also redirect traffic to healthy resources.
[Learn More](#)

[Get started now](#)

Domain registration

If you need a domain name, you can find an available name and register it by using Route 53. You can also make Route 53 the registrar for existing domains that you registered with other registrars.
[Learn More](#)

[Get started now](#)

Click on Create Hosted Zone. Give Domain Name and keep type as Public Hosted Zone.

Create Hosted Zone

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.

Domain Name:

Comment:

Type:

A public hosted zone determines how traffic is routed on the Internet.

Click on Create. Your Domain is now configured.

Dashboard

Hosted zones

Health checks

Traffic flow

Traffic policies

Policy records

Domains

Registered domains

Back to Hosted Zones

Create Record Set

Import Zone File

Delete Record Set

Test Record Set

Record Set Name

Any Type

Aliases Only

Weighted Only

Displaying 1 to 2 out of 2 Record Sets

Name	Type	Value	Evaluate Target Health	Health
smu.edu	NS	ns-561.awsdns-06.net. ns-487.awsdns-60.com. ns-1736.awsdns-25.co.uk. ns-1243.awsdns-27.org.	-	-
smu.edu	SOA	ns-561.awsdns-06.net. awsdns-hostmaster.amazon.	-	-

To get started, click Create Record Set button or click an existing record set.

Task 6: Create Health Checks for the Windows Server 2016 IIS Web Server.

Now on left side, click on Health Checks. You'll now configure Health Check for the EC2 Instance in Mumbai region.

Click on Health Check. Give the Health Check Name, IP Address of your EC2 Instance running in Asia Pacific (Mumbai) region and click Next.

Name ⓘ

What to monitor ☒ Endpoint ⓘ

☐ Status of other health checks (calculated health check)

☐ State of CloudWatch alarm

Monitor an endpoint

Multiple Route 53 health checkers will try to establish a TCP connection with the following resource to determine whether it's healthy.
[Learn more](#)

Specify endpoint by ☒ IP address ☐ Domain name

Protocol ⓘ

IP address * ⓘ

Host name ⓘ

Port * ⓘ

Path / ⓘ

Advanced configuration

URL ⓘ

Health check type Basic - no additional options selected ([View Pricing](#))

Create a CloudWatch Alarm which sends the notification to SNS Topic Subscribers in-case of failure and click on Create Health Check.

Get notified when health check fails ?

If you want CloudWatch to send you an Amazon SNS notification, such as an email, when the status of the health check changes to unhealthy, create an alarm and specify where to send notifications.

Create alarm ☒ Yes ☐ No ?

CloudWatch sends you an Amazon SNS notification whenever the status of this health check is unhealthy for one minute.

Send notification to ☒ Existing SNS topic ☐ New SNS topic ?

Common_SNS_Topic

* Required

Cancel

Previous

Create health check

You'll observe that Health Check has been successfully configured which keeps on checking the Health of EC2 Instance launched in Asia Pacific (Mumbai) region.

Filter by keyword					1 to 1 of 1 health check	
	Name	Status	Description	Alarms		
<input type="checkbox"/>	Check 1	16 minutes ago now Healthy	http://3.6.223.194:80/	1 of 1 in OK		

If you navigate to CloudWatch service, you'll observe that the CloudWatch alarm has been successfully created for above Health Check.

Alarms (3) <input type="checkbox"/> Hide Auto Scaling alarms Clear selection Refresh Create composite alarm Actions Create alarm						
<input type="text" value="Search"/> OK Any type < 1 > Settings						
<input type="checkbox"/>	Name	State	Last state update	Conditions	Actions	
<input type="checkbox"/>	Check_1-awsroute53-f7cda20d-12a0-4cf7-af31-09ed83ce4367-Low-HealthCheckStatus	OK	2020-04-20 04:17:00	HealthCheckStatus < 1 for 1 datapoints within 1 minute		

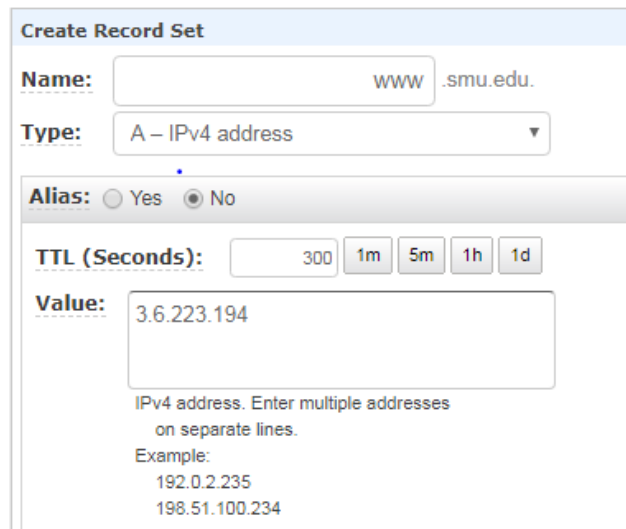
Task 7: Create A records and define Failover routing policies in Route 53.

Navigate to Route 53 Service, click on Hosted Zone that you've created and click on Create Record Set.

We'll now specify Asia Pacific (Mumbai) as the Primary and US East (N. Virginia) as the Secondary region for our IIS Web Application.

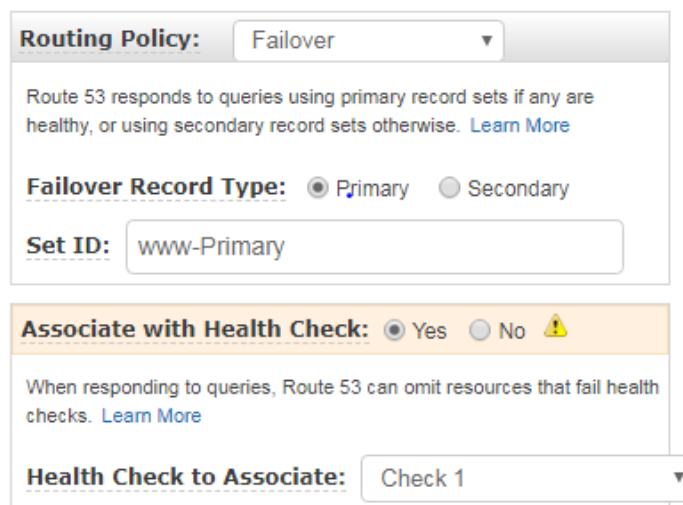
Create primary record set as follows.

Name will be "www"; hence the website becomes www.smu.edu. Value field contains the IP Address of the EC2 Instance which is running in the Asia Pacific (Mumbai) region.



The screenshot shows the 'Create Record Set' form in the AWS Route 53 console. The 'Name' field is filled with 'www.smu.edu.'. The 'Type' dropdown is set to 'A - IPv4 address'. The 'Alias' section has 'No' selected. The 'TTL (Seconds)' is set to '300'. The 'Value' field contains the IP address '3.6.223.194'. Below the value field, there is a note: 'IPv4 address. Enter multiple addresses on separate lines. Example: 192.0.2.235, 198.51.100.234'.

Select the Routing Policy as Failover. This Failover Record Type will be PRIMARY. Also associate Health Check that we've configured to this A Record and Click on Create Record.



The screenshot shows the 'Routing Policy' section in the AWS Route 53 console. The 'Routing Policy' dropdown is set to 'Failover'. Below it, a description states: 'Route 53 responds to queries using primary record sets if any are healthy, or using secondary record sets otherwise. [Learn More](#)'. The 'Failover Record Type' section has 'Primary' selected. The 'Set ID' field contains 'www-Primary'. The 'Associate with Health Check' section has 'Yes' selected, with a warning icon. Below this, a note states: 'When responding to queries, Route 53 can omit resources that fail health checks. [Learn More](#)'. The 'Health Check to Associate' dropdown is set to 'Check 1'.

Similarly, again click on Create Record Set and create A Record as follows.

Name will be “www”; hence the website becomes www.smu.edu. Select Alias as YES and give Alias Target as CloudFront Domain which is caching the Website from US East (N. Virginia) region.

Create Record Set

Name: .smu.edu.

Type:

Alias: ☒ Yes ☐ No

Alias Target:

Alias Hosted Zone ID: Z2FDTNDATAQYW2

You can also type the domain name for the resource. Examples:

- CloudFront distribution domain name: d111111abcdef8.cloudfront.net
- Elastic Beanstalk environment CNAME: example.elasticbeanstalk.com
- ELB load balancer DNS name: example-1.us-east-2.elb.amazonaws.com
- S3 website endpoint: s3-website.us-east-2.amazonaws.com
- Resource record set in this hosted zone: www.example.com
- VPC endpoint: example.us-east-2.vpce.amazonaws.com
- API Gateway custom regional API: d-abcde12345.execute-api.us-west-2.amazonaws.com
- Global Accelerator DNS name: a012345abc.awsglobalaccelerator.com

[Learn More](#)

Select the Failover Routing Policy and select Failover Record Type as Secondary. No need to specify Health Check here and click on Create.

Routing Policy:

Route 53 responds to queries using primary record sets if any are healthy, or using secondary record sets otherwise. [Learn More](#)

Failover Record Type: ☐ Primary ☒ Secondary

Set ID:

Evaluate Target Health: ☐ Yes ☒ No

Associate with Health Check: ☐ Yes ☒ No

Create

Your Records Sets should look like the below Screenshot.

<input type="checkbox"/>	Name	Type	Value	Evaluate Target Health
<input type="checkbox"/>	smu.edu.	NS	ns-561.awsdns-06.net. ns-487.awsdns-60.com. ns-1736.awsdns-25.co.uk. ns-1243.awsdns-27.org.	-
<input type="checkbox"/>	smu.edu.	SOA	ns-561.awsdns-06.net. awsdns-hostmaster.amazon.	-
<input type="checkbox"/>	www.smu.edu.	A	3.6.223.194	-
<input type="checkbox"/>	www.smu.edu.	A	ALIAS d3fa3qs2pqzc.cloudfront.net. (z2fdtndataqyw	No

Task 8: Test the Failover.

On the Route 53 Service, click on Test Record Set. Select Record Name as “www” and Record Type as “A” and click on Get Response.

Hosted zones
 Health checks
 Traffic flow
 Traffic policies
 Policy records
 Domains
 Registered domains
 Pending requests
 Resolver
 VPCs
 Inbound endpoints
 Outbound endpoints
 Rules

Check response from Route 53

This tool returns values based on the settings in a Route 53 hosted zone. You can use it to see how Route 53 works — what value is returned to a DNS request given the values that you specified in your resource record sets. For geolocation and latency resource record sets, you can also simulate requests from a particular DNS resolver and/or client IP address to find out what response a client with that resolver and/or IP address would receive from Route 53. [Learn More](#)

Hosted zone

Record name

Type*

Simulate sending DNS request from specific IP address (optional)

For geolocation and latency resource record sets, type the IP address of a DNS resolver. The **Response returned by Route 53** section displays the response that Route 53 returns to the specified IP address.

Resolver IP address

More options

*Required

Get response

Response returned by Route 53

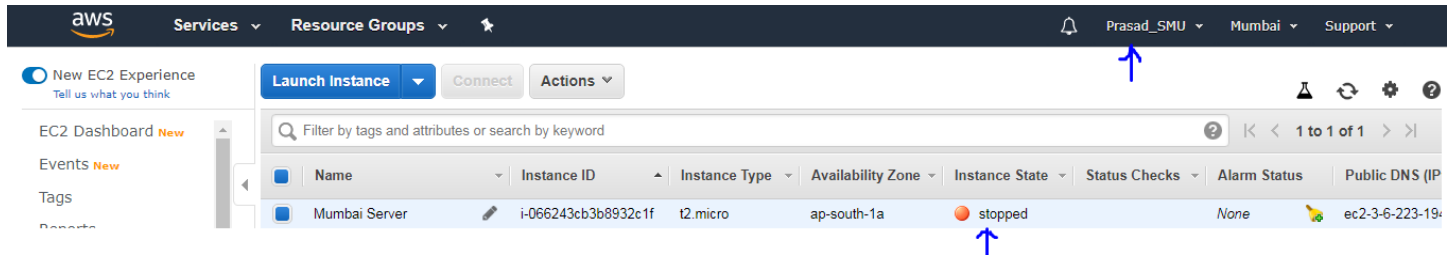
Response from Route 53 based on the following options.

DNS request sent to Route 53	www.smu.edu. IN A
EDNS0 client subnet IP	24
DNS response code	NOERROR
Protocol	UDP
Response returned by Route 53	3.6.223.194

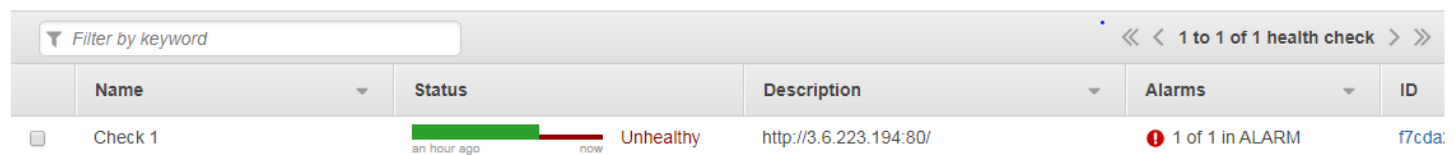
You'll notice that Response returned by Route 53 displays the IP Address of the EC2 Instance which is running in Asia Pacific (Mumbai) region. Which is absolutely Right 😊

Now let's STOP this server and test the Failover.

Go to Asia Pacific (Mumbai) region, navigate to EC2 Service and Stop the EC2 Instance.



In US EAST (N. Virginia) region, navigate to Route 53 and click on Health Check. You'll see the Health Check as UNHEALTHY. You now have received SNS notification on your Email



Click on Hosted Zones, select your Hosted Zone and click on Test Record Set. You'll now see the Response returned by Route 53 indicates that the Website is now serving by the CloudFront Edge Locations.

Check response from Route 53

This tool returns values based on the settings in a Route 53 hosted zone. You can use it to see how Route 53 works — what value is returned to a DNS request given the values that you specified in your resource record sets. For geolocation and latency resource record sets, you can also simulate requests from a particular DNS resolver and/or client IP address to find out what response a client with that resolver and/or IP address would receive from Route 53. [Learn More](#)

Hosted zone smu.edu.

Record name

Type*

Simulate sending DNS request from specific IP address (optional)

For geolocation and latency resource record sets, type the IP address of a DNS resolver. The **Response returned by Route 53** section displays the response that Route 53 returns to the specified IP address.

Resolver IP address

[More options](#)

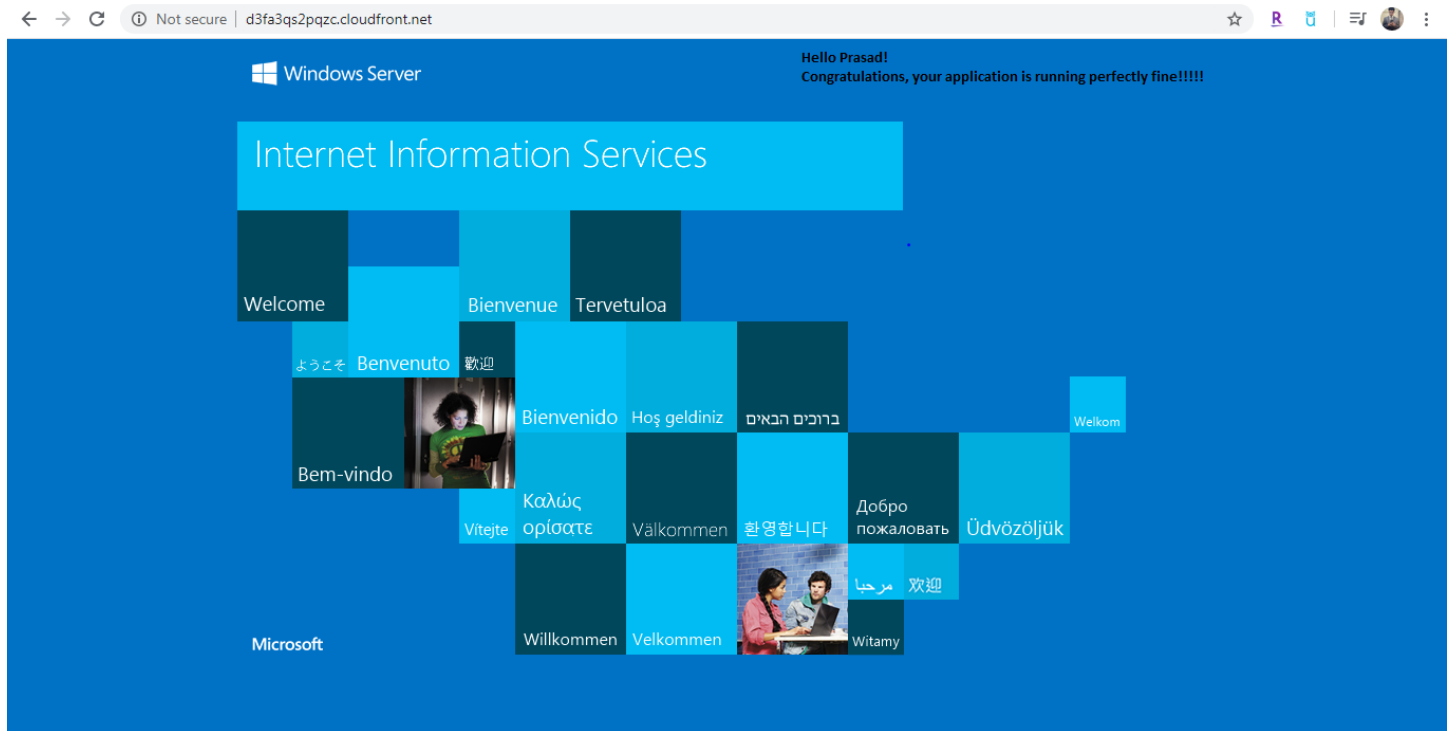
***Required** [Get response](#)

Response returned by Route 53

Response from Route 53 based on the following options.

DNS request sent to Route 53	www.smu.edu. IN A
EDNS0 client subnet IP	24
DNS response code	NOERROR
Protocol	UDP
Response returned by Route 53	52.85.145.32 52.85.145.222 52.85.145.28 52.85.145.6

Your website is running perfectly fine even though the entire primary region is down.



We have successfully achieved High Availability across the REGIONS.

You can now turn on the Web Server in Asia Pacific (Mumbai) region and Test the Record Set again. It should point out to the Primary Web Server which is hosted in Asia Pacific (Mumbai) region.

Filter by tags and attributes or search by keyword							
<input type="checkbox"/>	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status
<input type="checkbox"/>	Mumbai Server	i-066243cb3b8932c1f	t2.micro	ap-south-1a	running	2/2 checks ...	None

Health Check Status is now back to Healthy. You now have received SNS notification on your Email.

	Name	Status	Description	Alarms
<input type="checkbox"/>	Check 1	Healthy an hour ago	http://3.6.223.194:80/	1 of 1 in OK

Finally, DNS is now pointing back to the Primary EC2 Instance which is hosted in Asia Pacific (Mumbai) region.

Check response from Route 53

This tool returns values based on the settings in a Route 53 hosted zone. You can use it to see how Route 53 works — what value is returned to a DNS request given the values that you specified in your resource record sets. For geolocation and latency resource record sets, you can also simulate requests from a particular DNS resolver and/or client IP address to find out what response a client with that resolver and/or IP address would receive from Route 53. [Learn More](#)

Hosted zone smu.edu.

Record name

www

Type*

A

Response returned by Route 53

Response from Route 53 based on the following options.

DNS request sent to Route 53 www.smu.edu. IN A

EDNS0 client subnet IP 24

DNS response code NOERROR

Protocol UDP

Response returned by Route 53 3.6.223.194



Recommendations:

I real world scenario, if you have a registered Domain then you can host a Website on S3 Bucket and create Secondary Failover Policy for it in Route 53. Hence, even if the Primary region goes down, your website will be still up and running through the S3 Bucket.

This completes the lab on achieving High Availability across Regions.

If you have any questions, contact me on pbhavsar@smu.edu.