# AWS Route-53 (DNS Service)

**What is Route 53 ?**

- Highly available and scalable, reliable and cost-effective Domain Name System (DNS) service.
- It also offers with Domain Name Registration [Create New or Transfer Existing]
- It translates end users requests from names to IP and vice versa. For an example [Name= gurujise.com and IP=35.15.1.24]
- Supported with health checks to route traffic to the healthy endpoints
- It has different types of routing policy which help you do design your solution based on your requirement, [Round Robin, Weighted, latency based, Geo, Geoproximity]
- Supported with public and private domains
- Pricing
    - $0.50 pr hosted zone /month for first 25 hosted zones
    - $0.10 per hosted zone / month for additional hosted zones

**Functions of Route 53 :**

- DNS Management
- Traffic Management
- Availability Monitoring
- Domain Registration

**Features of Route 53 :**

- GUI based management (management Console)
- Domain Registration
- Health check and monitoring
- DNS failover
- Multiple routing policies

**Commonly used DNS records supported by Route53 :**

- A Record : Name/URL to IPv4
- AAAA Record : Name/URL to IPv6
- CNAME : Name/URL to Name/URL
- Alias Record : Name/URL to AWS Resources

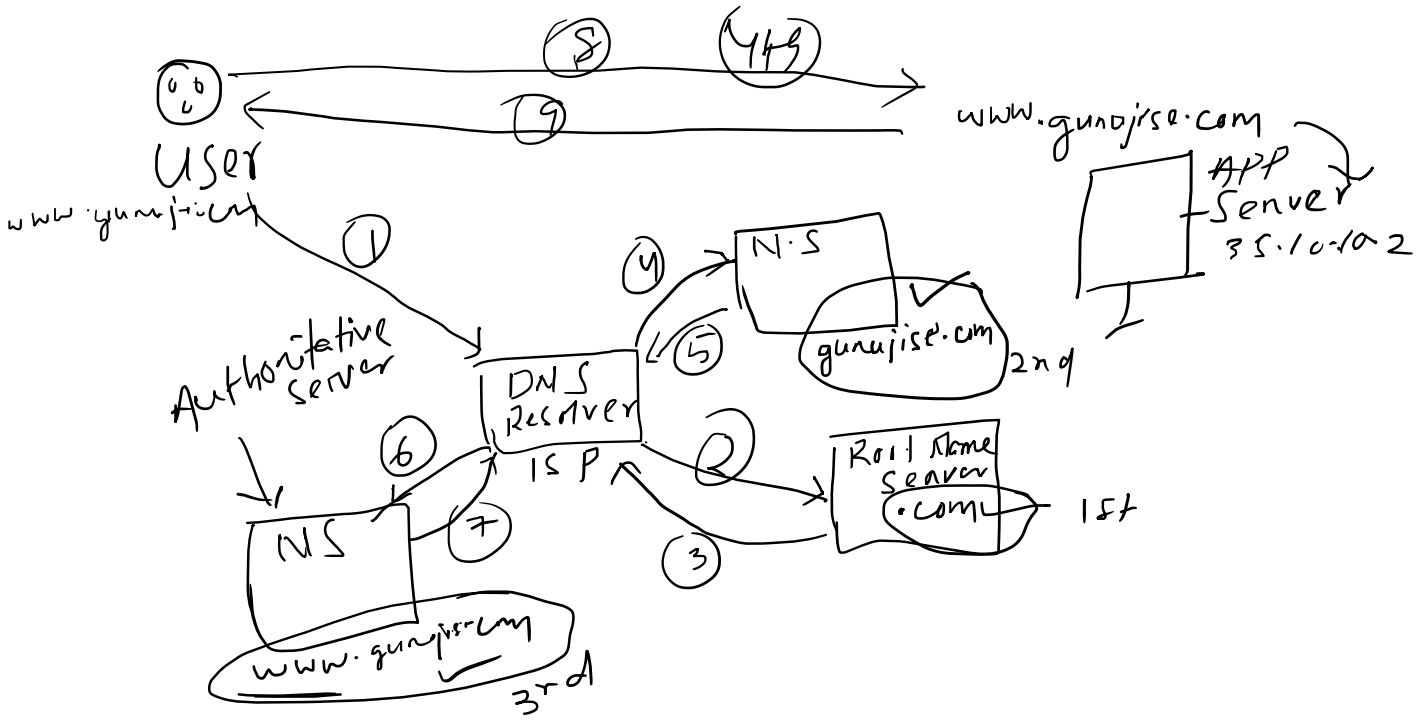
**Other Important Terms :**

- TTL (Time To Live)
    - For an example if an user hits an URL and get the response back and if your TTL is set to 100 seconds then then the browser will cached the response/IP for next 100 second and after 100 second if the user tray again then the request will again send to the Route53
    - Min TTL = 60 second
    - Max TTL = 24 hours
- CNAME Record
    - It can redirect DNS queries to any DNS records

        - From one URL to another, but only for non-root domains, for example blogs.gurujise.com
    - You can't create CNAME record for root domain, for example gurujise.com
    - You can't create CNAME record that has the same name as the hosted zone (zone apex)
    - Redirect DNS queries for a record name regardless of record type, such as A or AAAA
    - Amazon charges for CNAME queries
- ALIS Record
    - It can only redirect quires to selected aws resources, such as
        - S3
        - CloudFront
        - Another record in Route53 hosted zone that you created
    - It works for both root and non-root domains
        - Alias abc.com -> grujise.com
        - Alias abc.com -> bogs.gurujise.com
    - Alis records can be created at the top node of DNS namespaces, which is also known as zone apex
        - If the DNS is gurujise.com then zone apex can be gurujise.com
    - You can create alias record for gurujise.com that can route traffic to www.gurujise.com
    - Amazon doesn't charge for alias queries to AWS resources

**Understanding the concept Domain/URL :**

- Example : **https://www.gurujise.com**
    - **https** : protocol
    - **www** : subdomain
    - **gurujise** : Domain Name
    - **com** : Top-level Domain
    - **gurujise.com** : Root Domain/Zone Apex




- Generic level Domain : .com .org .net
- Geographic Level Domain : .in .cn .pk

**Hosted Zone :**

- If you purchased your domain from AWS then it get created automatically and if you purchased from outside then you have to create it manually
- Hosted zone is collection if record sets for a domain
- Once you created a hosted zone you can continue to create record sets for the hosted zone
- When you create hosted zone it creates Name Servers (NS) records and Start Of Authority (SOA)

**Record sets :**

- Record sets are the subdomains created for hosted zone or root domain

**Route 53 Routing Policies :**

- Simple routing
  - Default
  - One to one mapping
  - Point a domain to a single resource
- Failover routing
  - HA, failover
  - With active passive solution
  - Worked based on health check

    - It required two records (primary and secondary)
    - Uses a simple policy unless it's unhealthy
    - It works only with public hosted zone
- Weighted routing
    - You can balance traffic based on the percentage to route traffic to two different servers
    - When you are running your application on a same domain with multiple servers you can manage distributing traffic
    - During new feature/content deployment, or during maintenance
    - Weights can be assign any number from 1 to 255 and it calculates based on the total number of weight values assigned on each server
    - Calculation example (total server=3 , values given = server-1: 2, server-2: 3 and server-3: 5, sum of weights = 2+3+5=10, each server= [2/10*100=20%], [3/10*100=30%], [5/10*100=50%])
- Latency based routing
    - Based on the lower latency it sends queries
    - It doesn't depends on the location rather depends on fasted response
- Geo-location routing
    - Country wise, language wise, restriction
    - Content can be localized
    - Works based on geographic location from where DNS queries originated from
    - Geo location can be specified by continent, by country, by state in US (e.g. NA and Canada)
    - In case of overlapping geographic location, priority goes to smallest geographic region
    - It works based on IP mapping
- Geo-proximity routing
    - Route traffic based on the physical distance between the region and your users
    - If traffic on one location/region is getting higher you can shrink the region and share the traffic with another region
- Multi value answer routing
    - Respond to DNS queries with up to eight healthy records selected at random
    - Maximum 8 Ips/Values can be defined, based on latency or shortest path queries can be sent

**Creating Active-Passive Failover Routing :**

**Step-01 :**

- Create a Route 53 health check for primary resource before creating the failover routing policy. No need to create health check for secondary site
- Go to "Health Checks" and create on "Create health check"
- Then follow the steps as mentioned below and then click on "Next"

## Configure health check

Route 53 health checks let you track the health status of your resources, such as web servers or mail servers, and take action when an outage occurs.

**Name** gurujise-primary-hc

**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** HTTP

**Domain name *** gurujise.com

**Port *** 80

**Path** / *images*

### Advanced configuration

**Request interval** ( ) Standard (30 seconds) (•) Fast (10 seconds)

**Failure threshold *** 1

**String matching** (•) No ( ) Yes

**Latency graphs** [ ]

**Invert health check status** [ ]

**Disable health check** [ ] By default, disabled health checks are considered healthy. Learn more

**Health checker regions** ( ) Customize (•) Use recommended

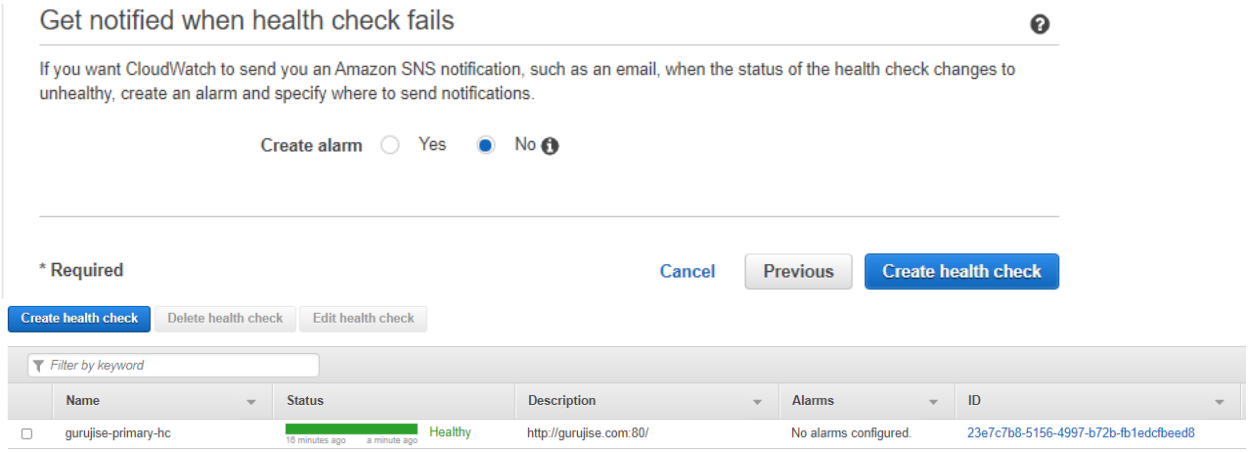
US East (N. Virginia)
US West (N. California)
US West (Oregon)
EU (Ireland)
Asia Pacific (Singapore)
Asia Pacific (Sydney)
Asia Pacific (Tokyo)
South America (São Paulo)

**URL** http://gurujise.com:80/

**Health check type** Basic + additional options: Fast Interval (View Pricing)

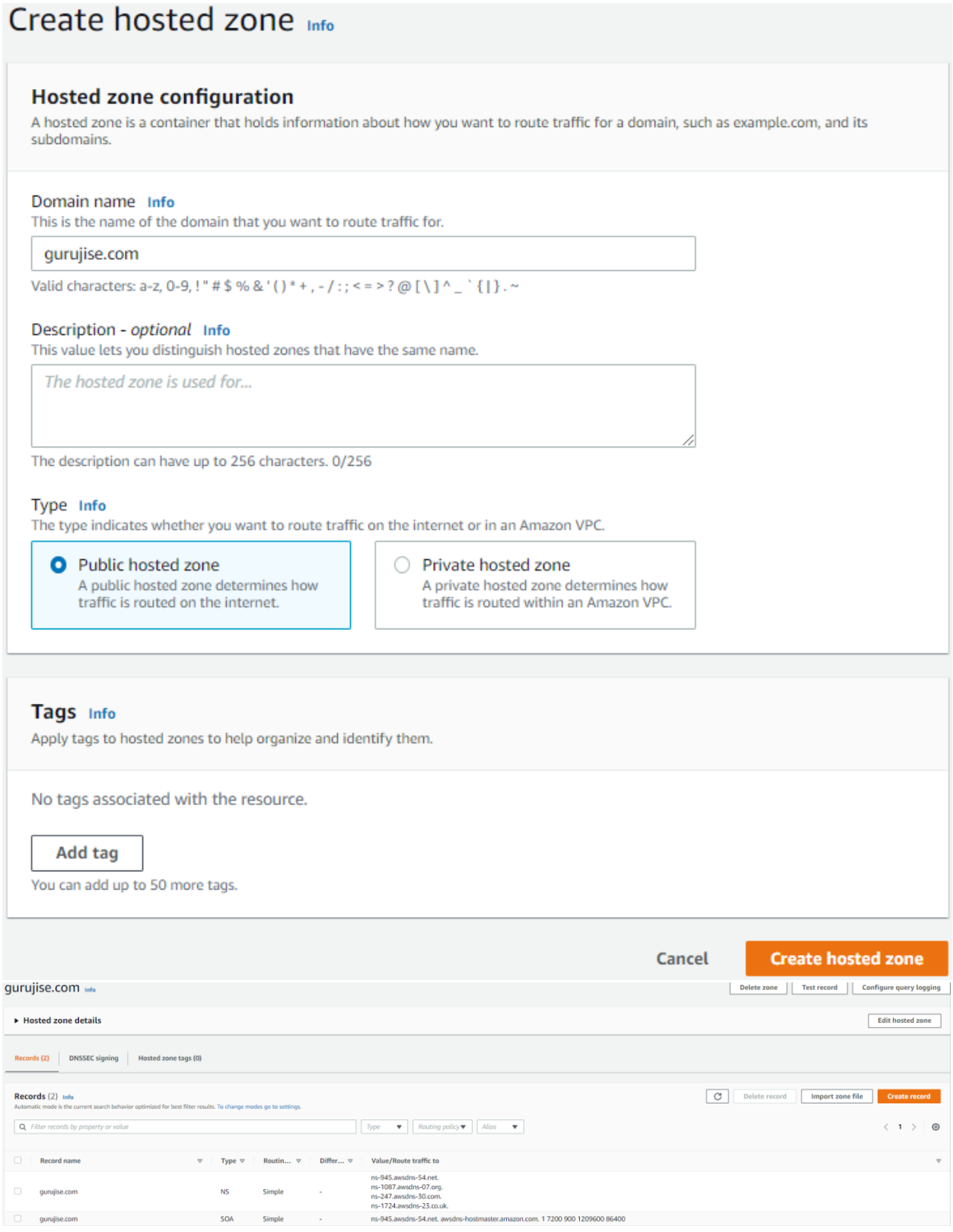
* Required

Cancel Next



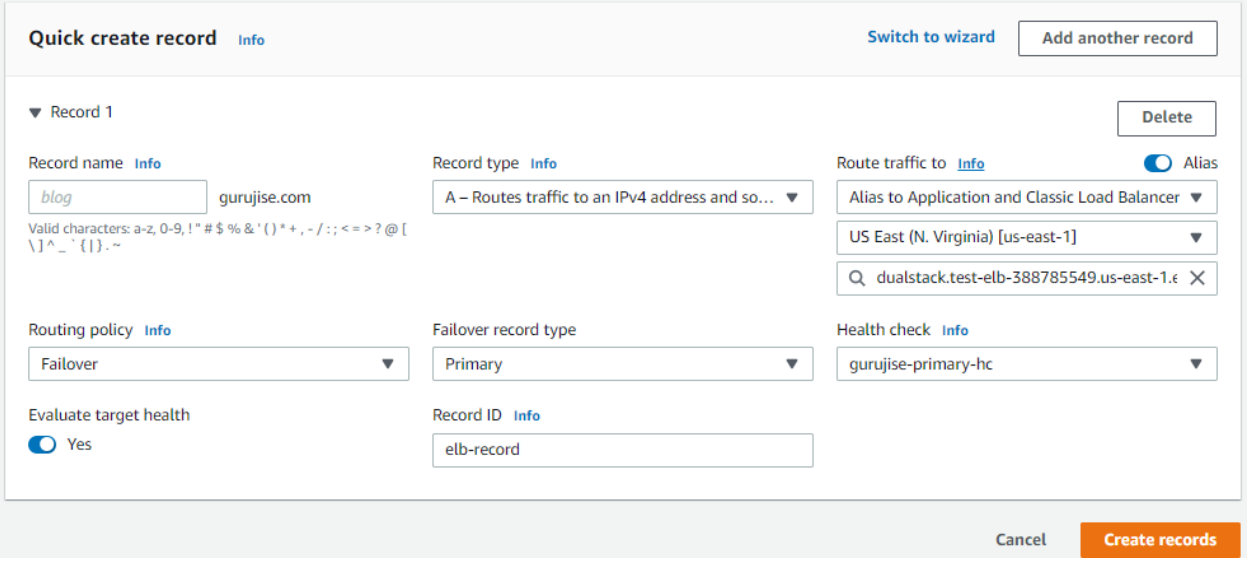

**Step-02 :**

- Before performing the below steps I assume you already created a “Hosted Zone” in Route 53 for your domain

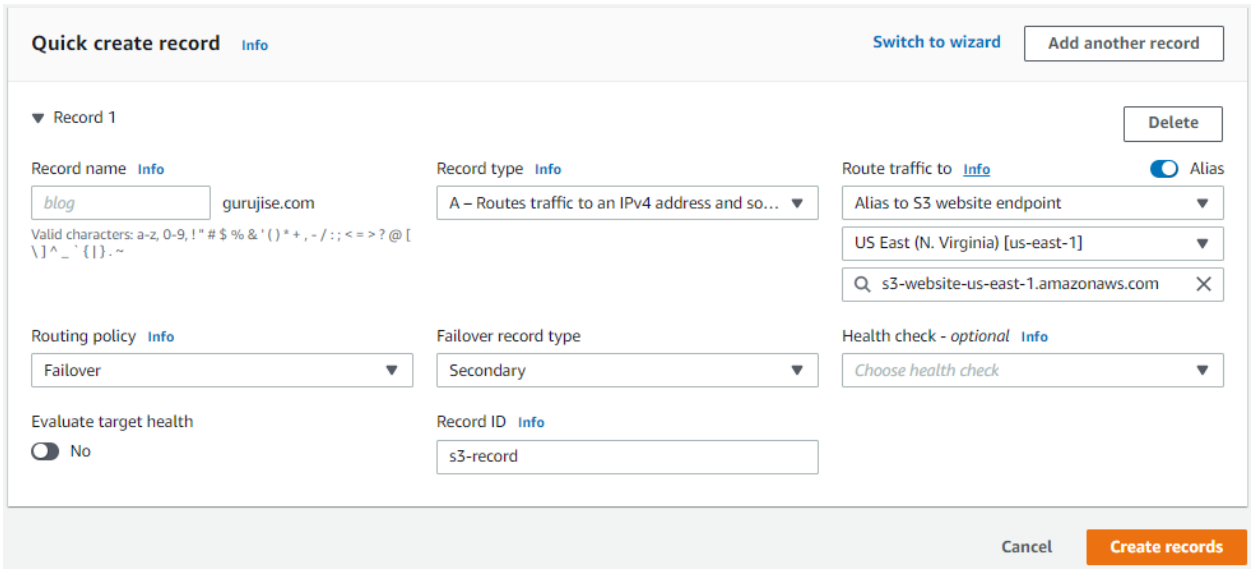



- Create two records for primary and secondary site followed by the below instruction (you can create your secondary site S3 static website as well for testing instead of creating two ELBs)
    - Enter a Name for your resource (such as resource.example.com).

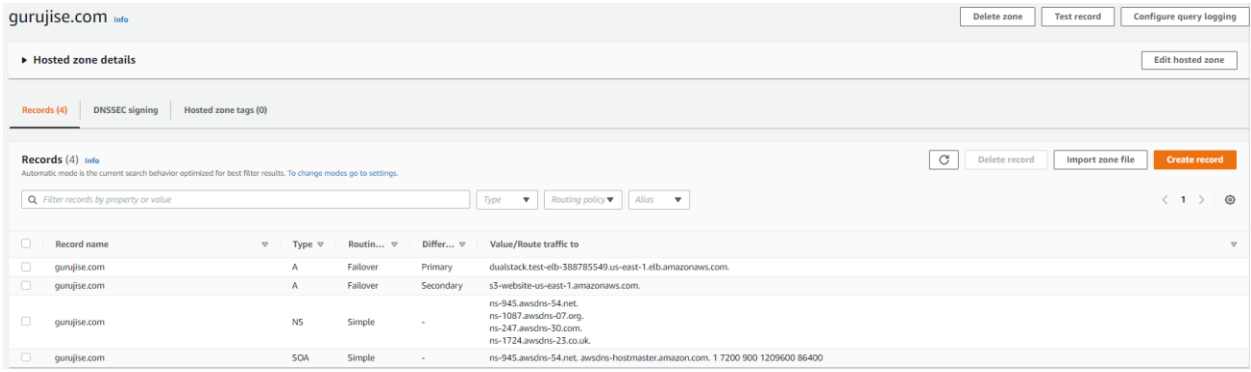
- Note: Use the same name for both resources.
- For Type, choose A – IPv4 address.
- For Alias, choose Yes.
- For Alias Target, enter the DNS name of your primary resource. The Alias Hosted Zone ID then appears.
- For Routing Policy, choose Failover.
- For your primary record, choose Primary for Failover Record Type. Enter a unique name (such as elb-Primary) for Set ID.
- For your secondary record, choose Secondary for Failover Record Type. Enter a unique name (such as elb-Secondary) for Set ID.
- For Evaluate Target Health, choose Yes for your primary record. Choose No for your secondary record.
- For your primary record, choose Yes for Associate with Health Check. For Health Check to Associate, choose the health check that you created for your primary resource.
- Choose Create to create your record.
- Primary resource record with ELB and EC2




- Secondary resource record with S3 static website




- Finally it should look like this




- Now you can go and stop your primary site (EC2 instance) to make the site unhealthy so that the failover will happen and route the traffic to the secondary site (S3 statis website)

- If you are successfully done with above lab, not time comes for deleting all resources to stop your billing.