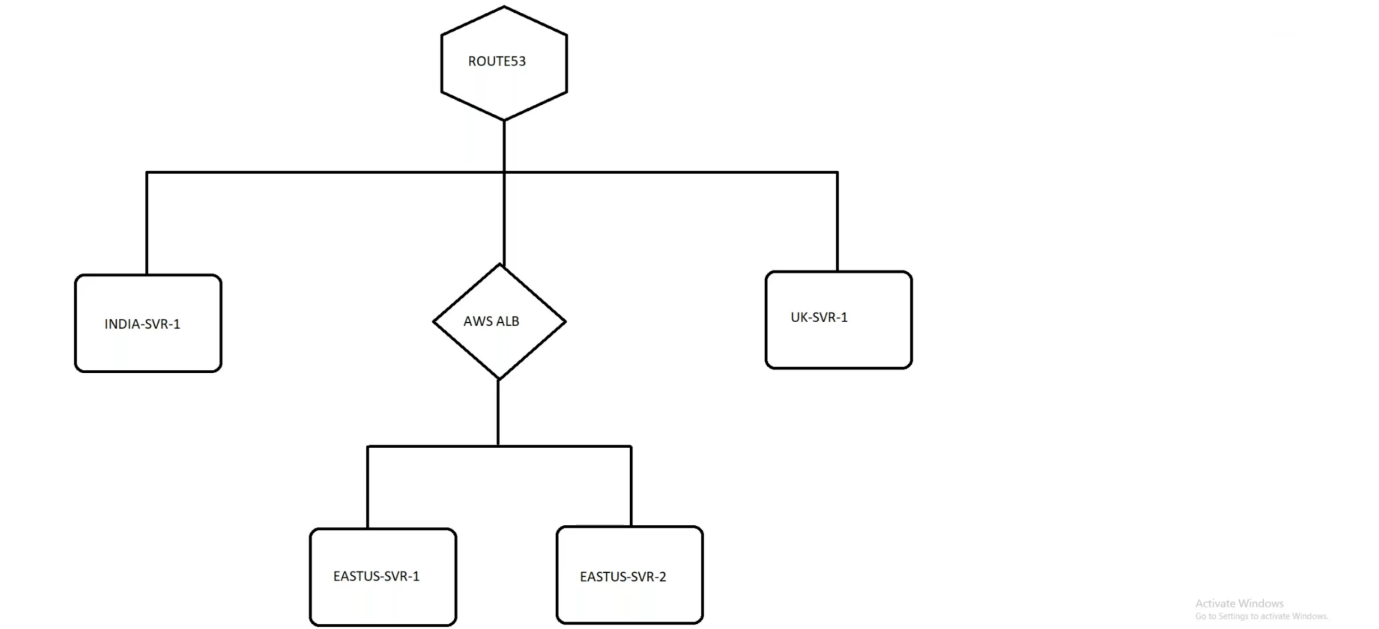
**22.AWS-B30-Route53-RoutingPolicies**

--- in this session, we will discuss about global load balancer. In aws, we have 2 global load balancers.

1. Route53 and Routing policies
2. Global accelerator

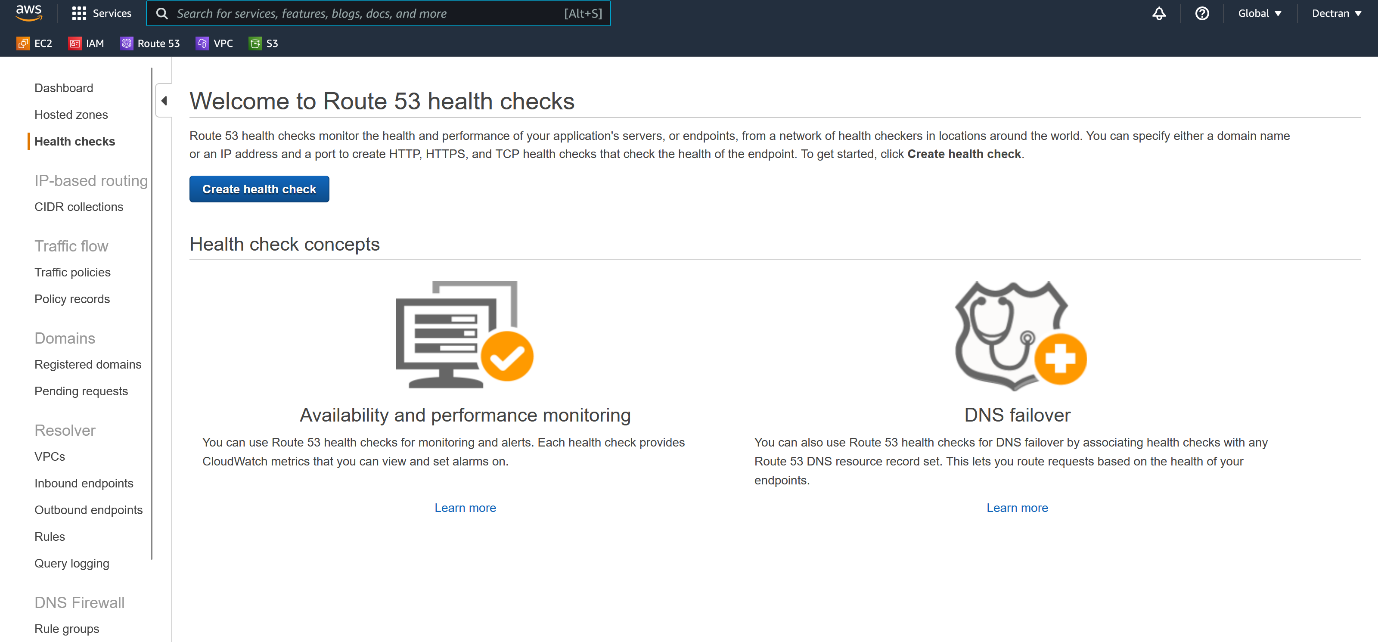
**Route53 and Routing policies**



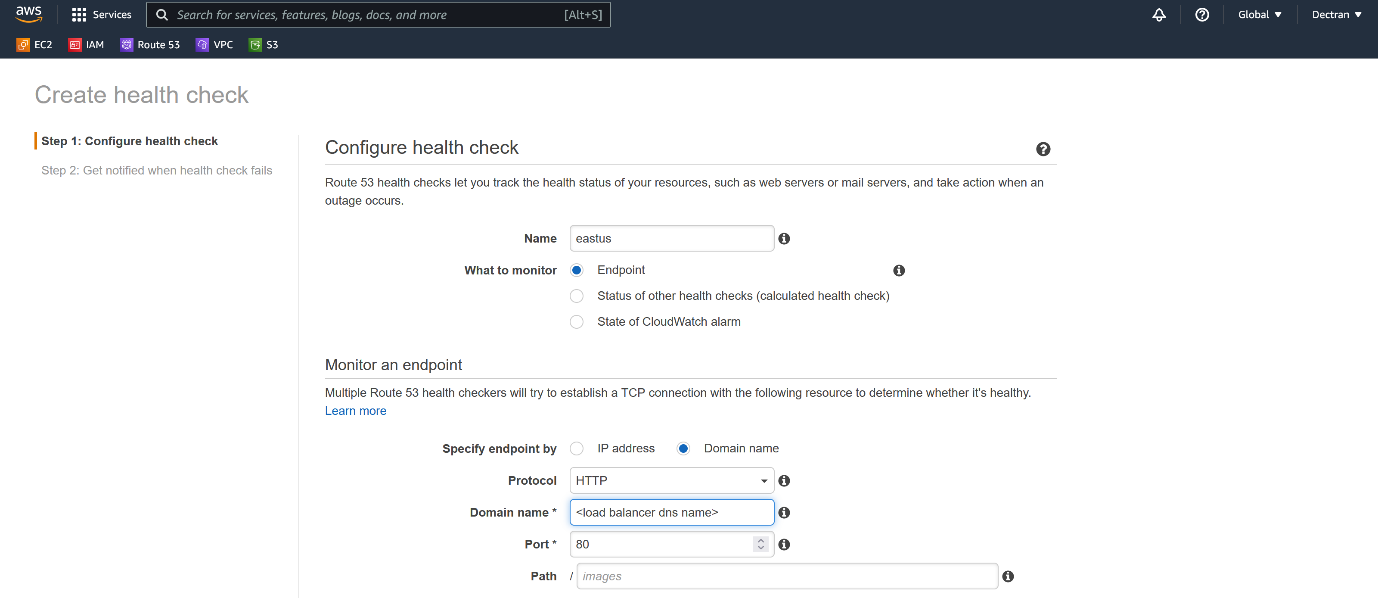
--- **scenario** – how can we use route53 when we deployed our application in multi region. In present use case 2 server attached to application load balancer and that load balancer is attached with route53. These 2 servers are in eastus-svr-1 and eastus-svr-2. I have deployed 2 other servers in 2 different regions and these 2 servers are also attached to route53.

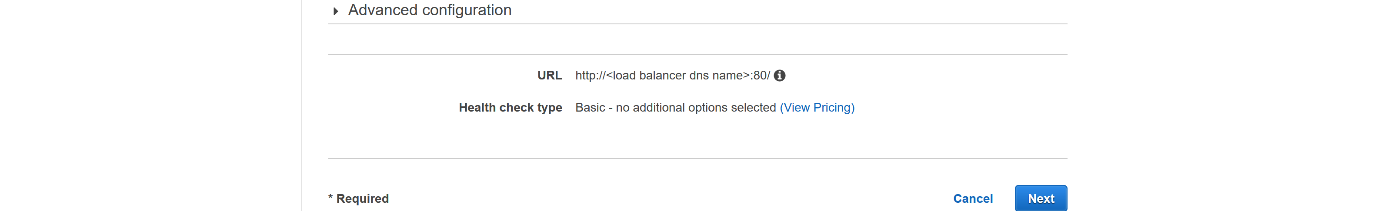
**Create health check for east-us**

--- **create a health check based on domain name.**



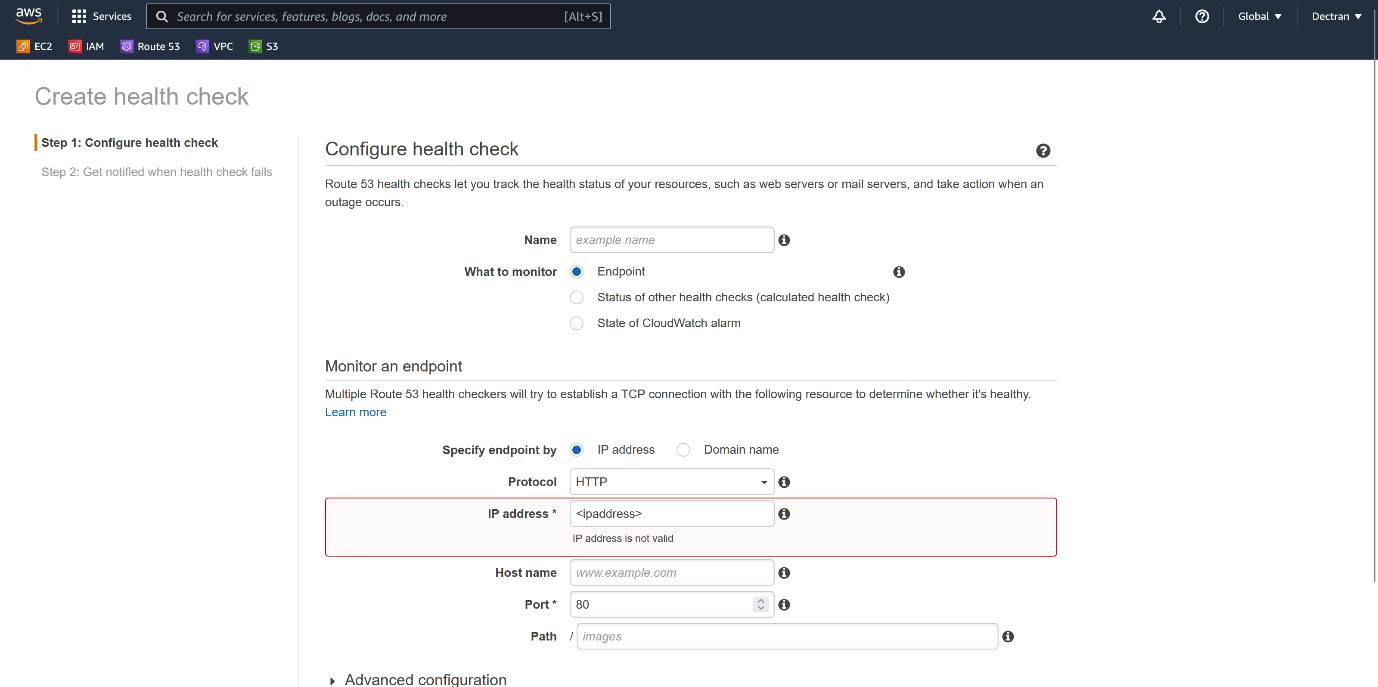
--- click on create health check





--- click on next to create a health check.

--- **create a health check based on ip address.**



--- **note** – click on next to create health check based on ipaddress.

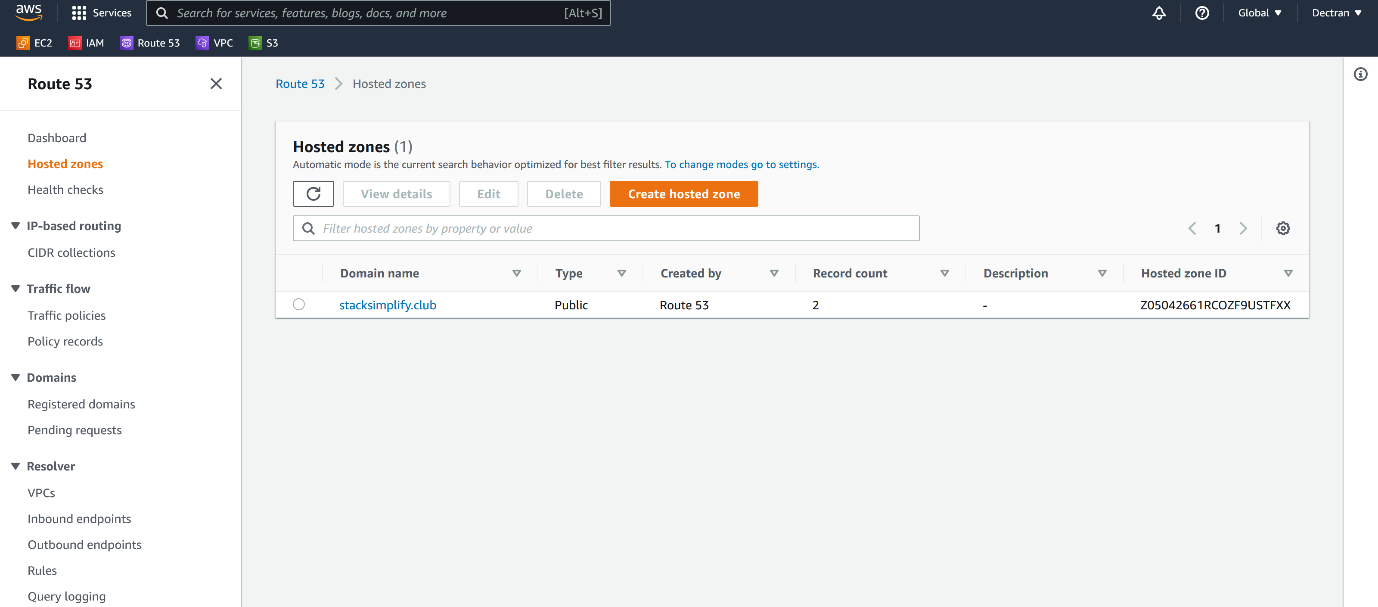
**Routing policies**

--- **Reference** - https://d ocs.aws.amazon.com/Route53/latest/DeveloperGuide/routing-policy.html

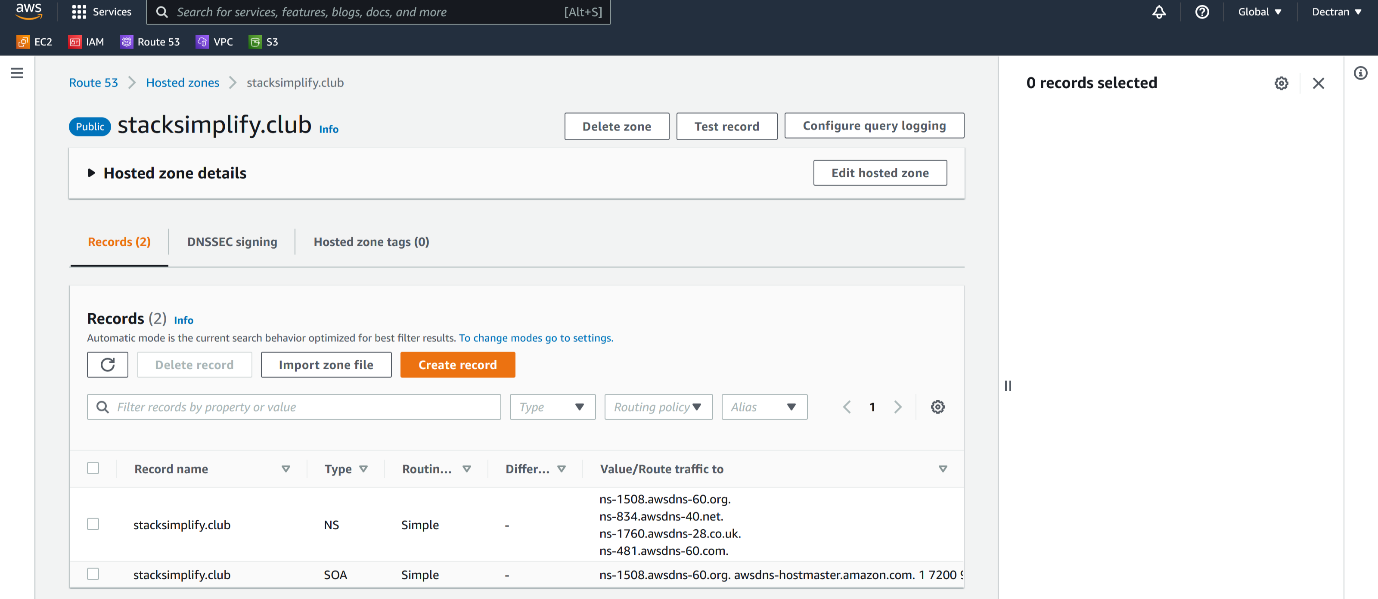
**Failover routing policy**

--- Use when you want to configure active-passive failover. You can use failover routing to create records in a private hosted zone. If the primary server is down then the traffic will divert to secondary server.

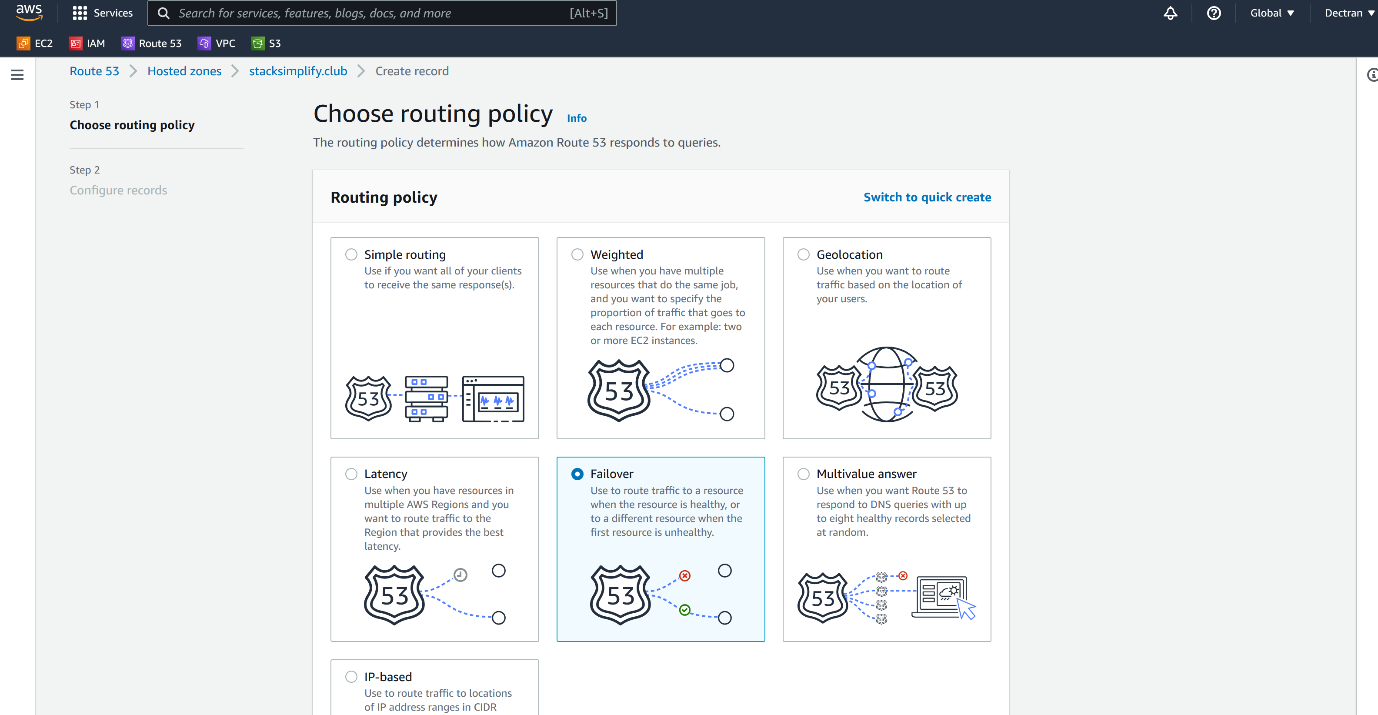
--- go to the aws route53.



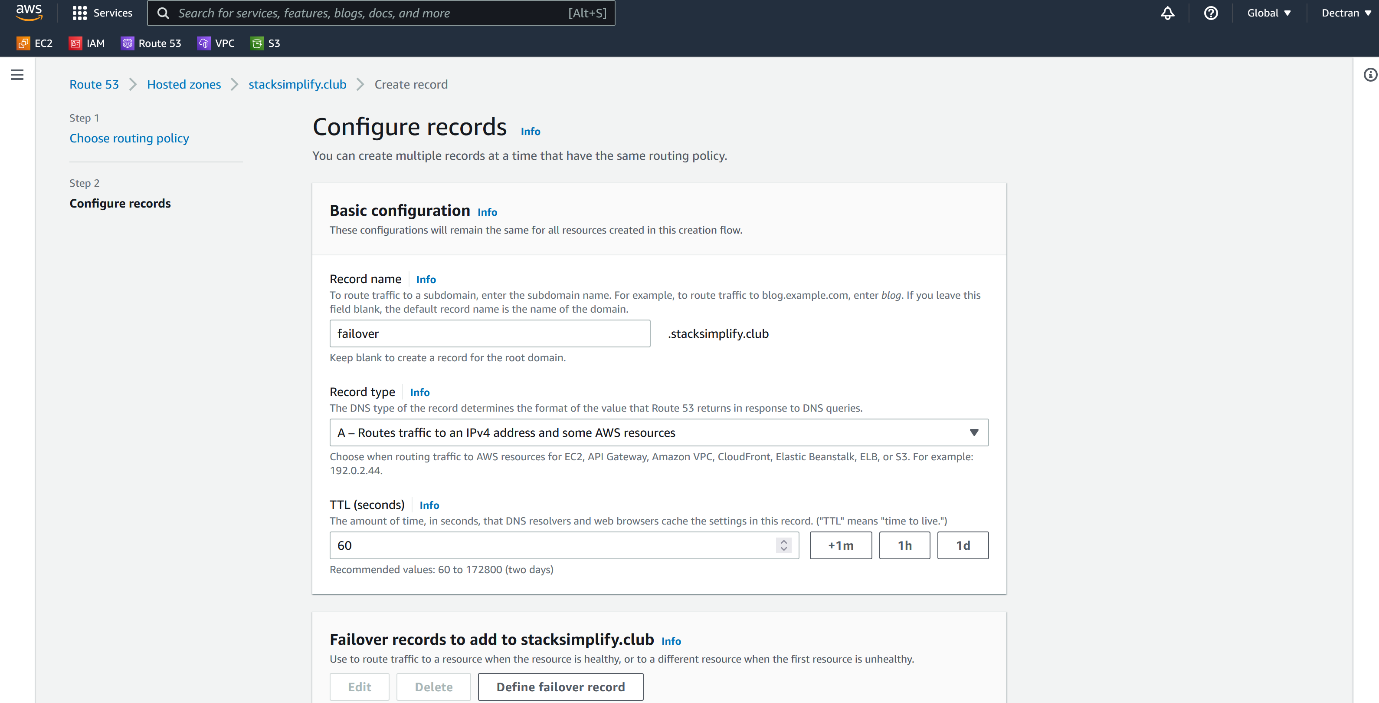
--- click on hosted zone.



--- click on create a record.

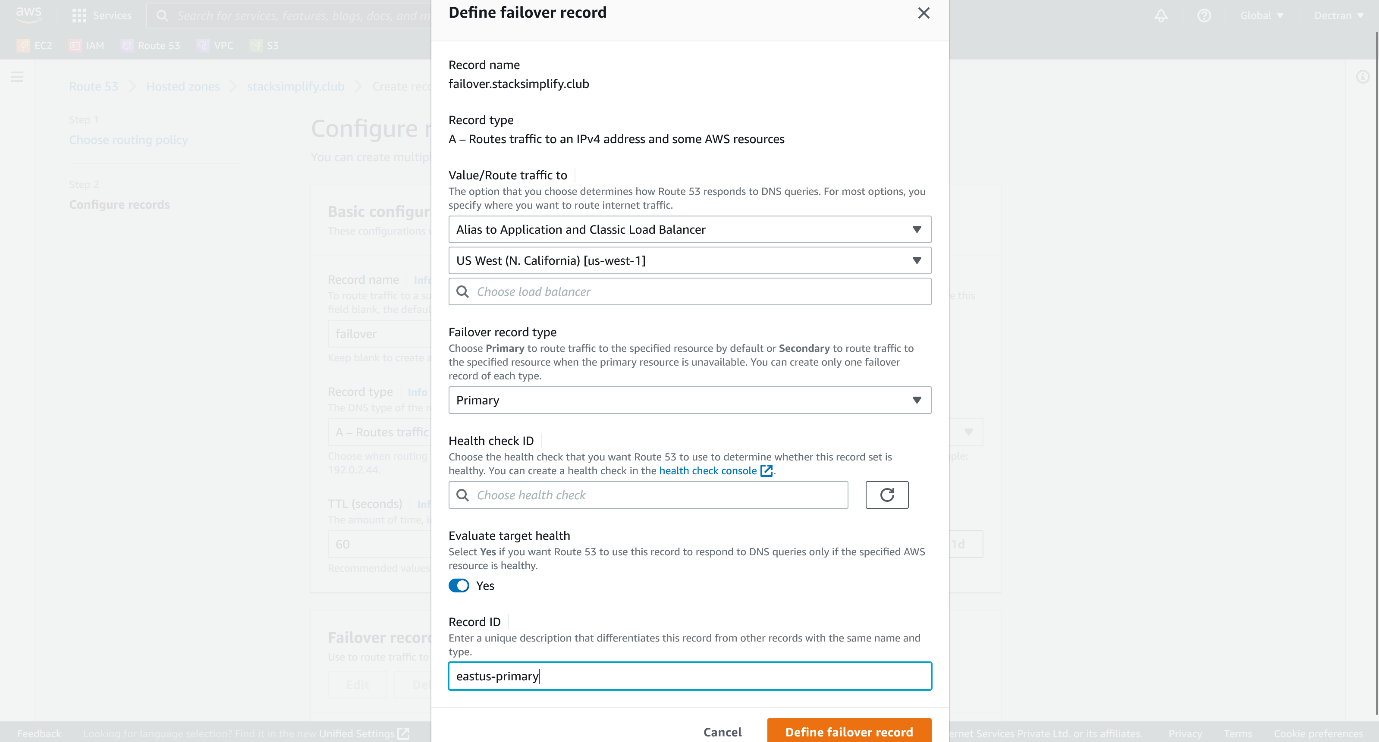


--- select failover and click on next.



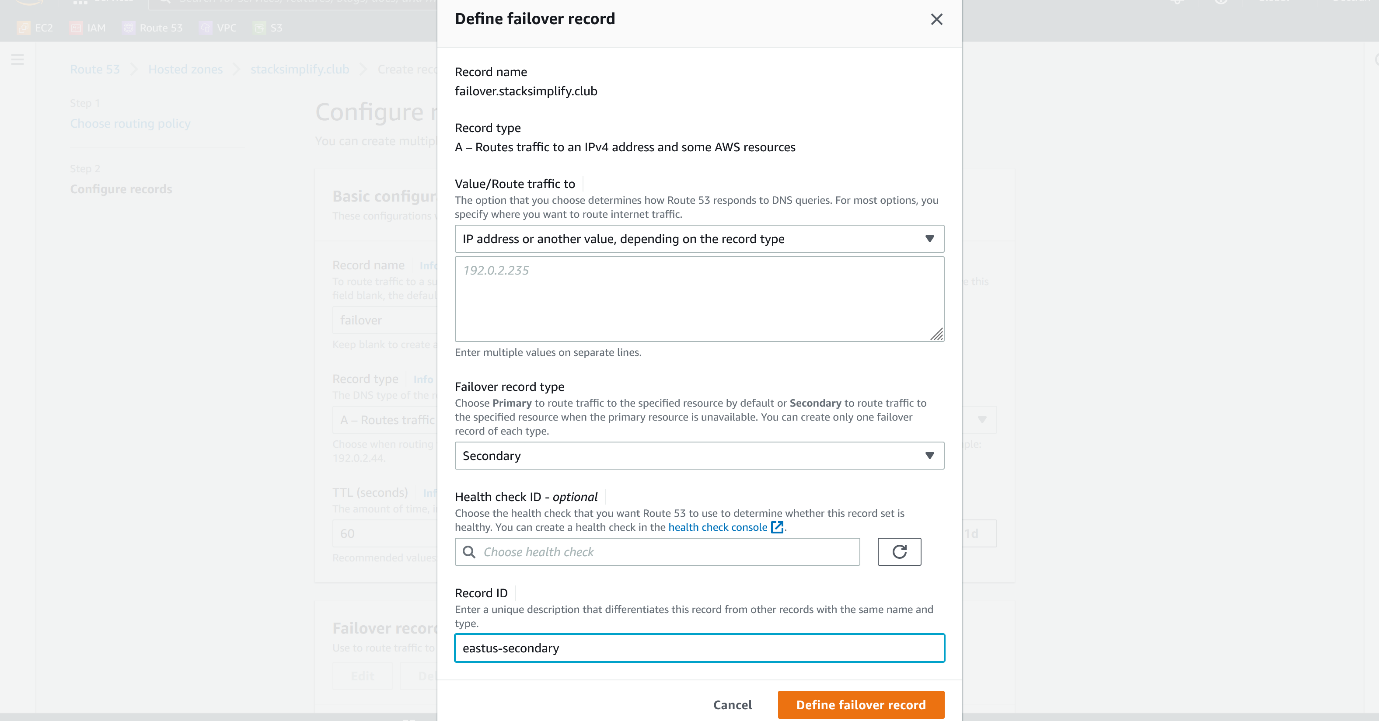
--- **TTL = 60** – it will refresh for every 1 minute.

--- click on Define failover record to add primary and secondary record.



--- click on define failover record.

--- **note** – I will add secondary target as ipaddress



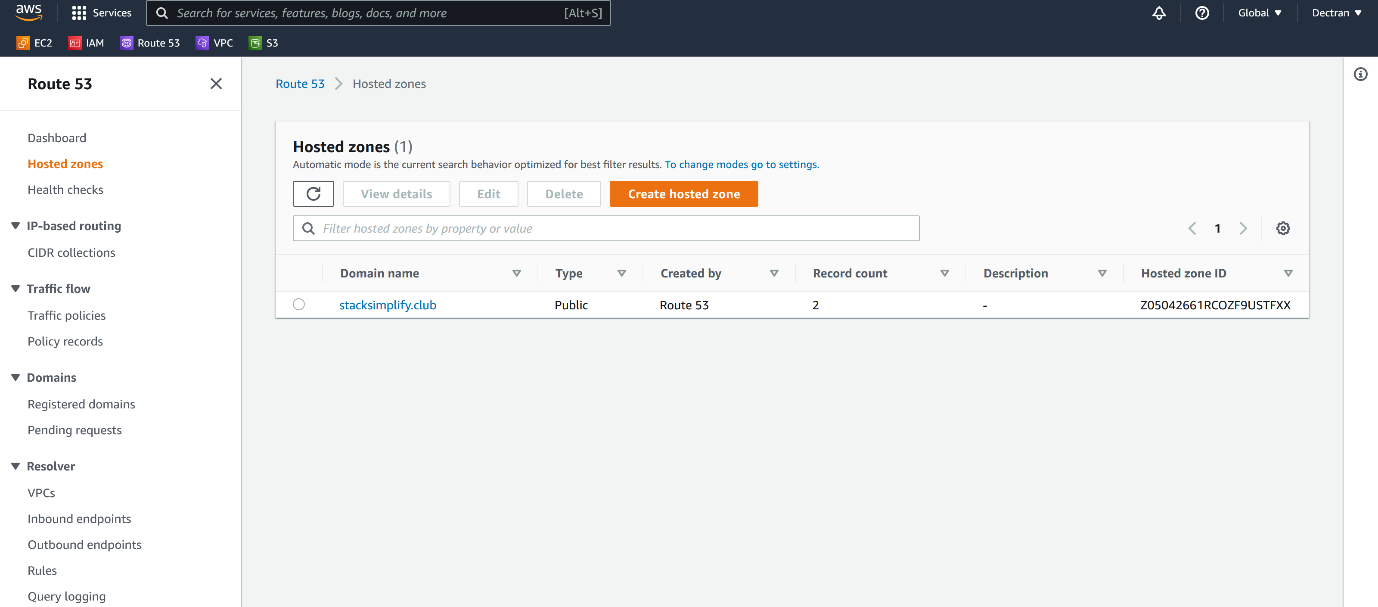
--- click on define failover record.

**Latency routing policy**

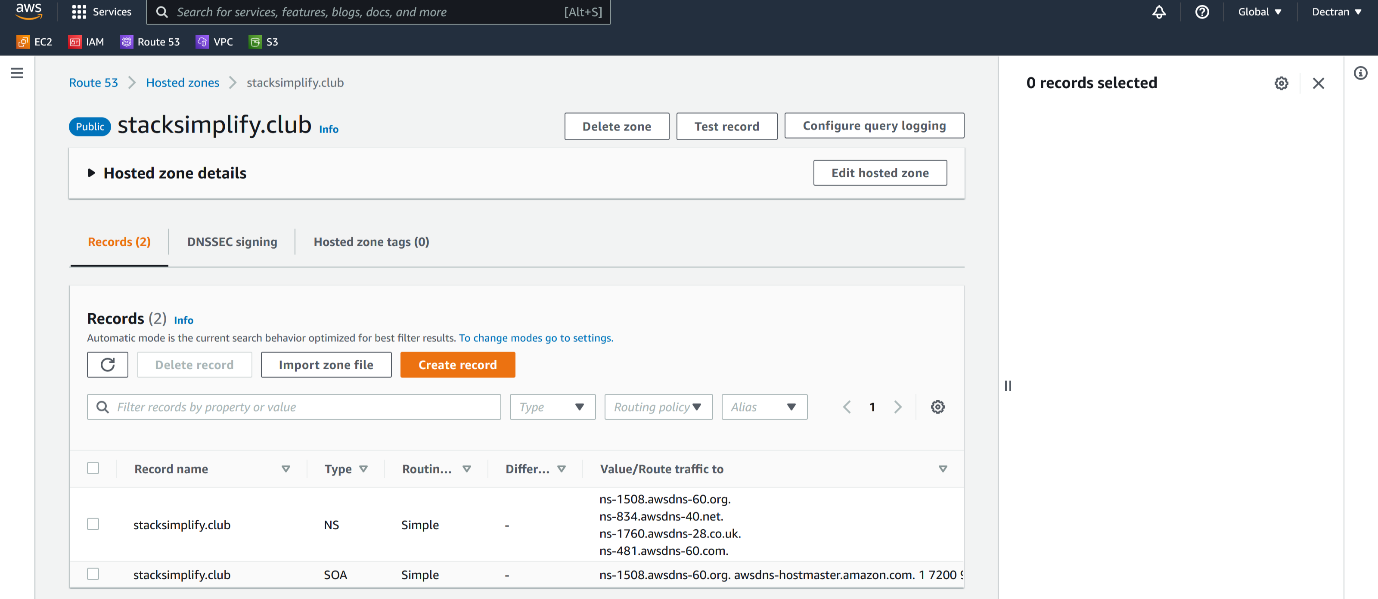
--- Use when you have resources in multiple AWS Regions and you want to route traffic to the region that provides the best latency. You can use latency routing to create records in a private hosted zone.

--- you want to connect your customer traffic to low latency servers then use latency routing policies.

--- go to the aws route53.



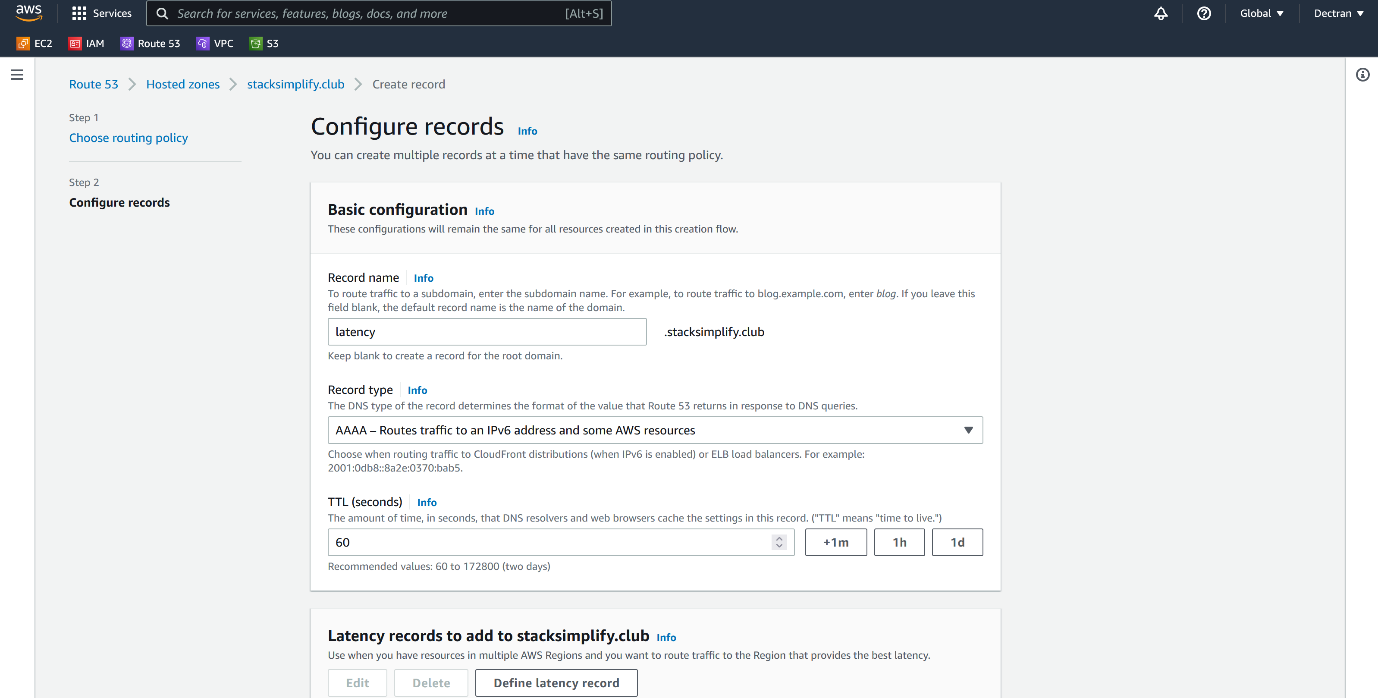
--- click on hosted zone.



--- click on create a record.

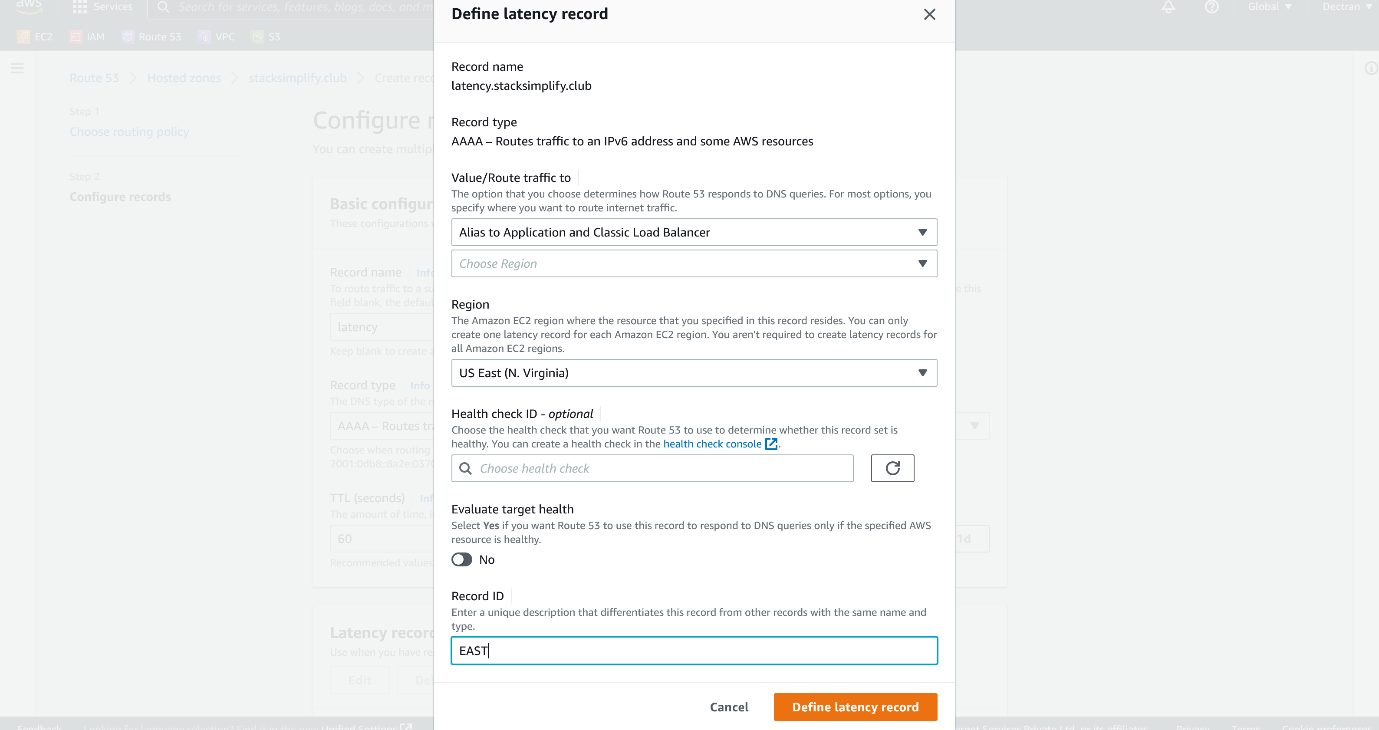


--- select latency and click on next.



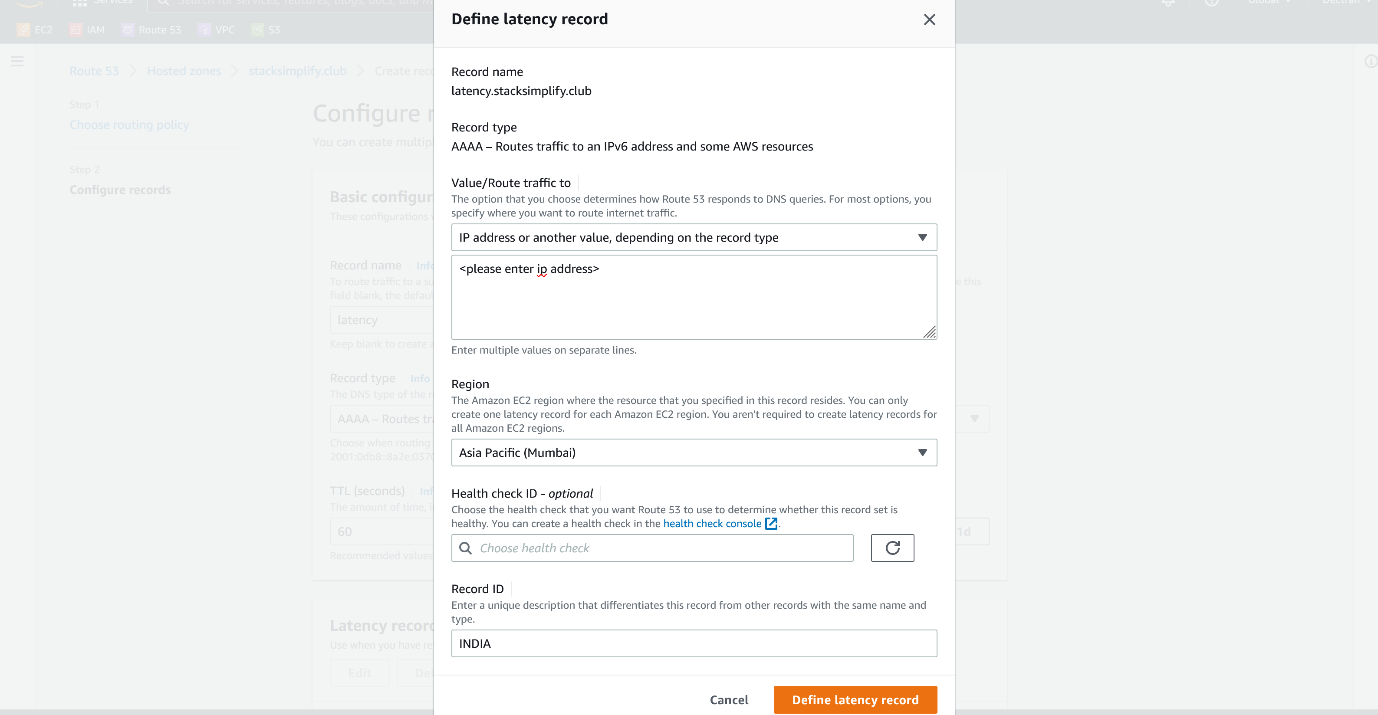
--- TTL = 60 – it will refresh for every 60 seconds.

--- click on define latency record.



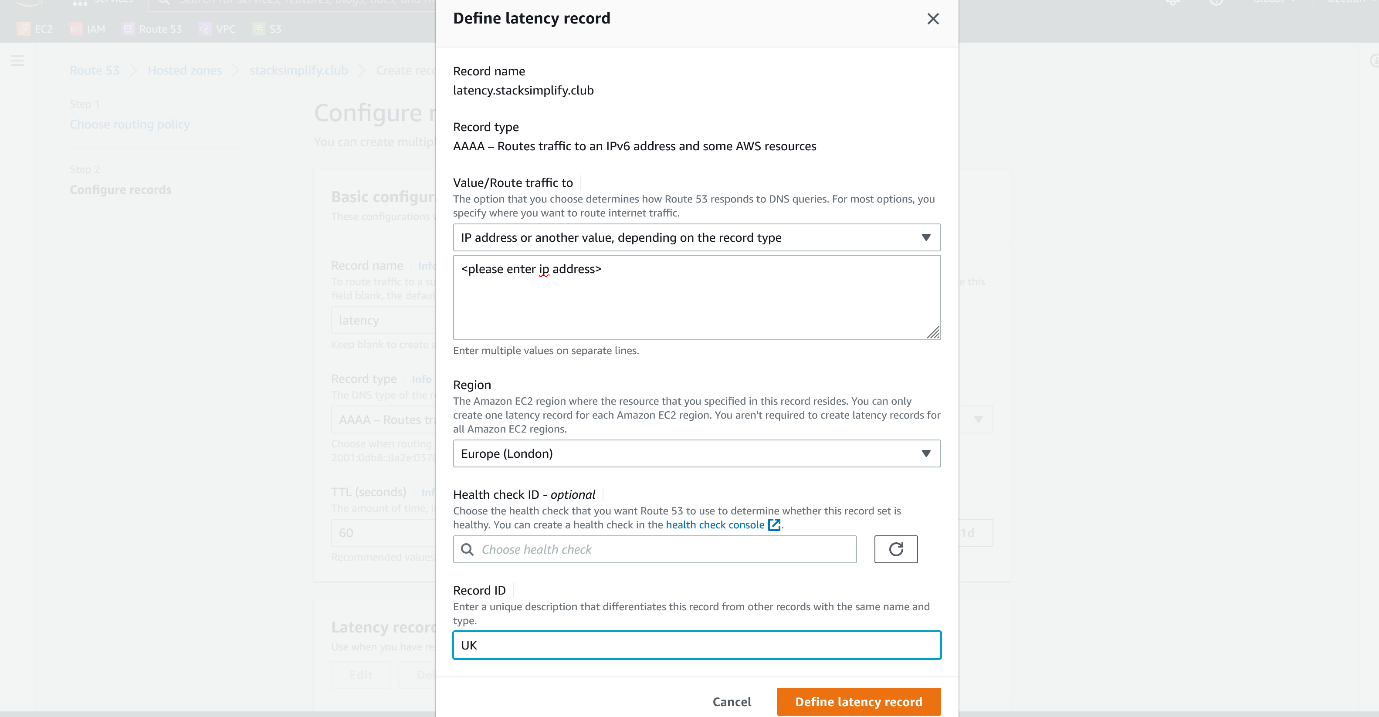
--- no need to worry about health check id and evaluate target health. Click on define latency record.

--- **note** - add another India server to the latency record by click on define latency record.

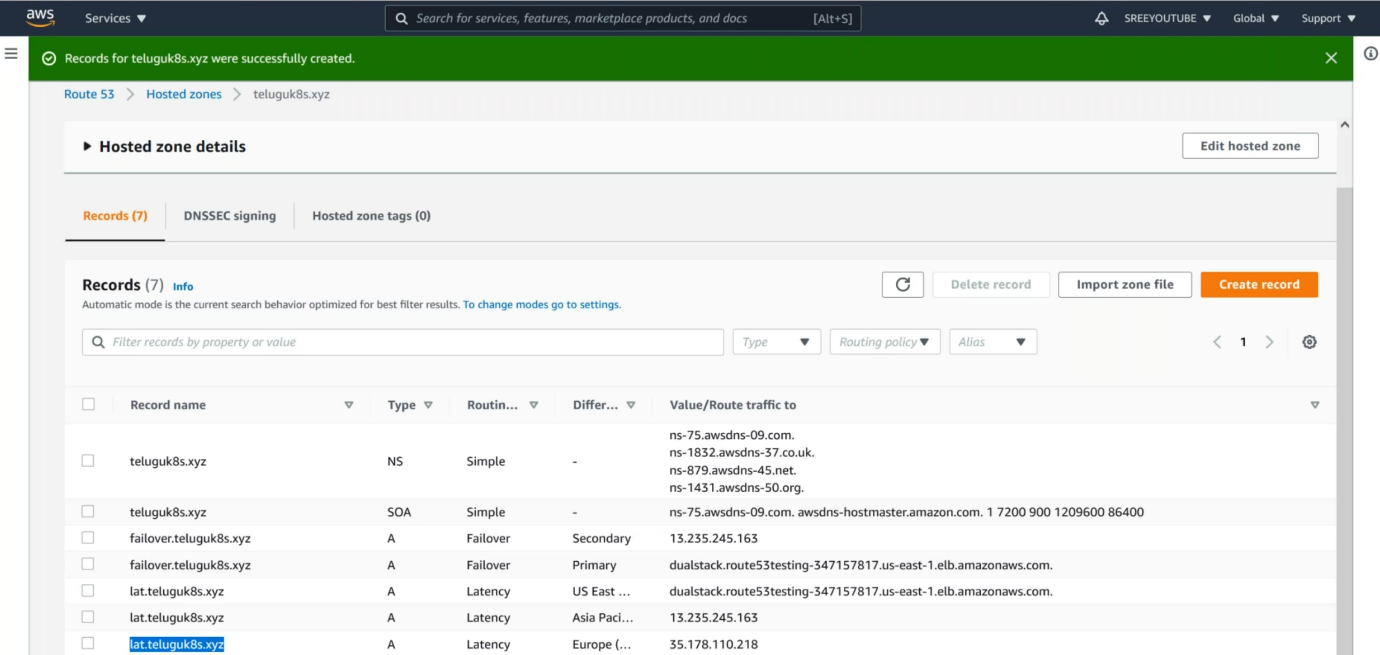


--- no need to worry about health check id and evaluate target health. Click on define latency record.

--- **note** - add another UK server to the latency record by click on define latency record.



--- NOTE – after these steps, click on create.



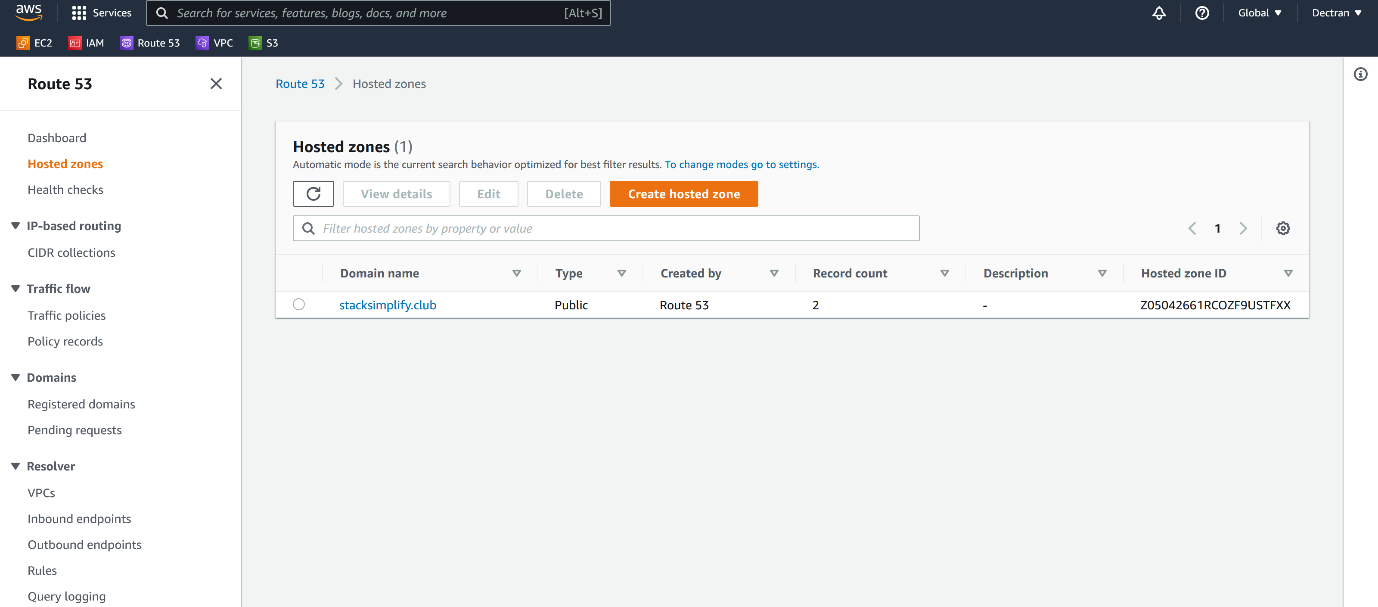
--- **note** – if your customers access your application from India then that request will go the Indian servers. Same goes to the other servers as well.

**Geolocation routing policy**

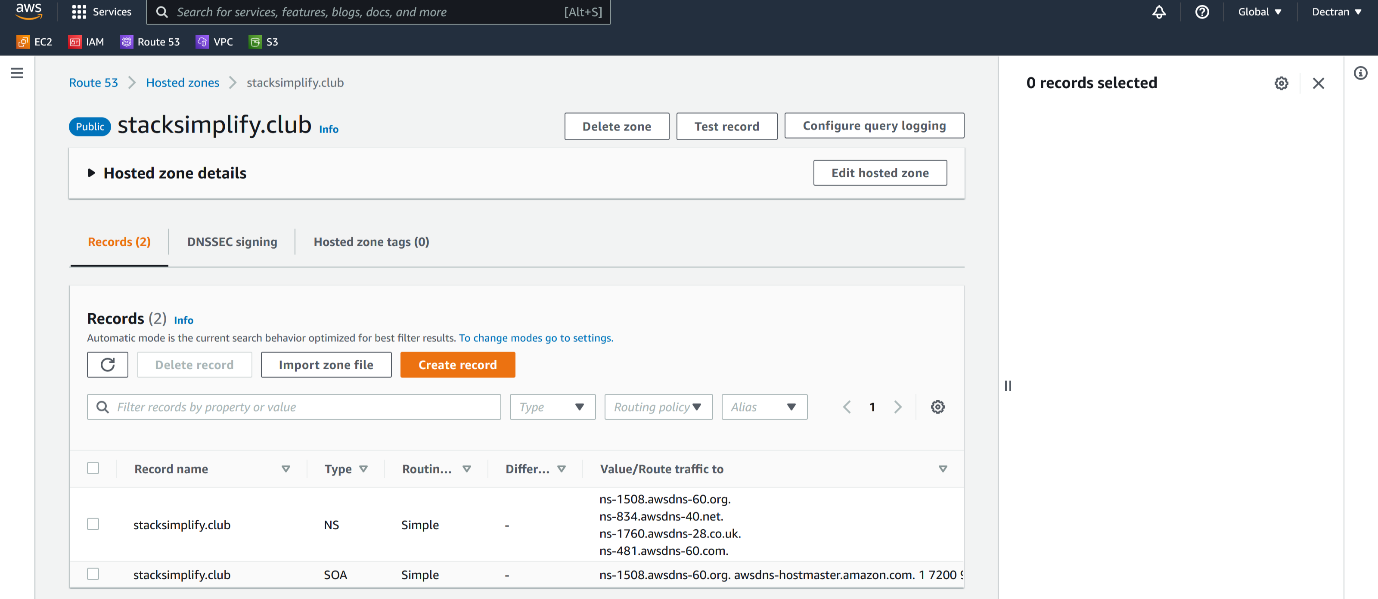
--- Use when you want to route traffic based on the location of your users. You can use geolocation routing to create records in a private hosted zone

--- latency and geolocation are some ways similar.

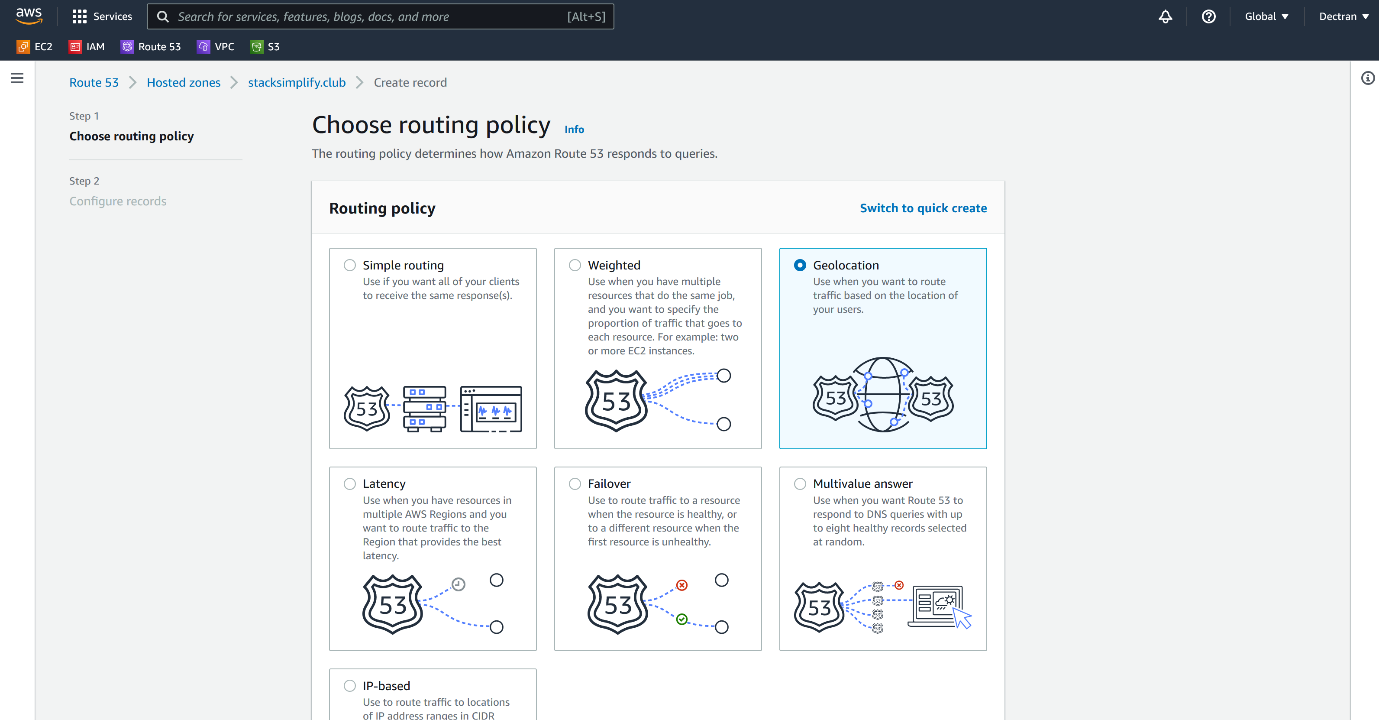
--- go to the aws route53.



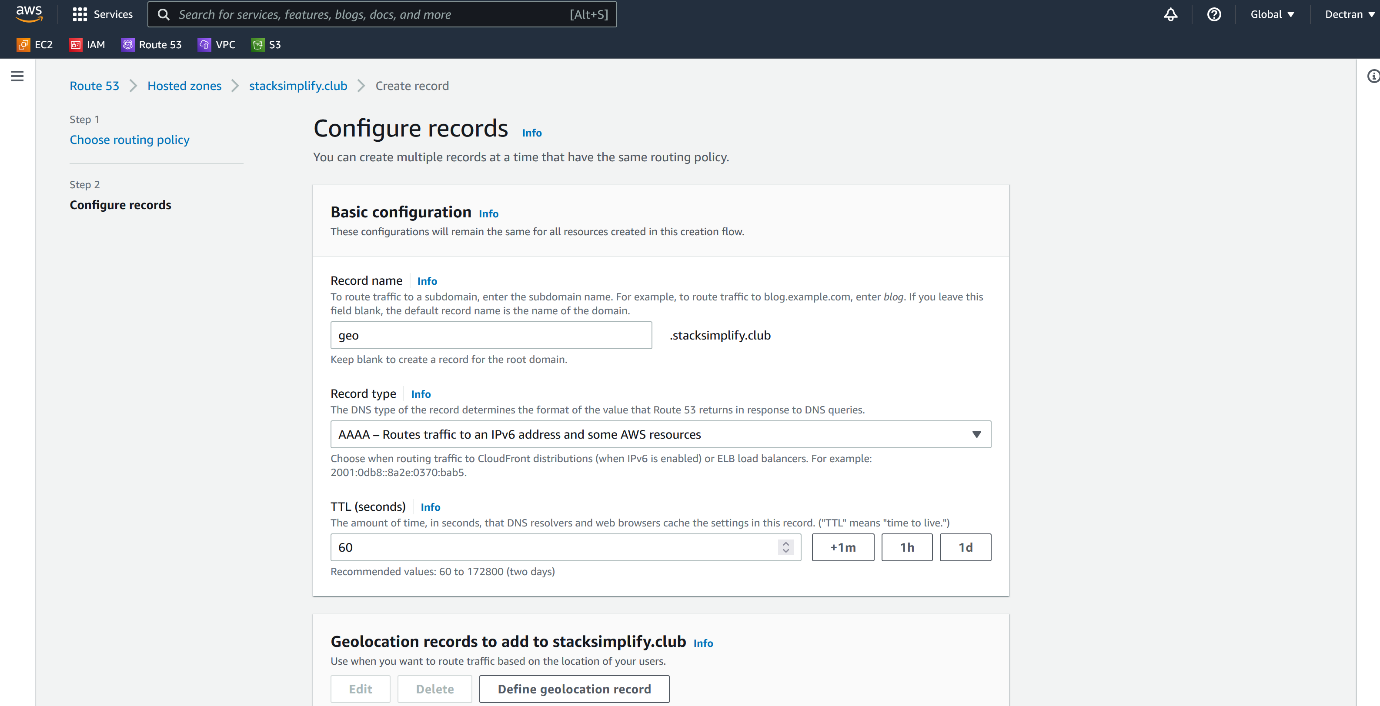
--- click on hosted zone.



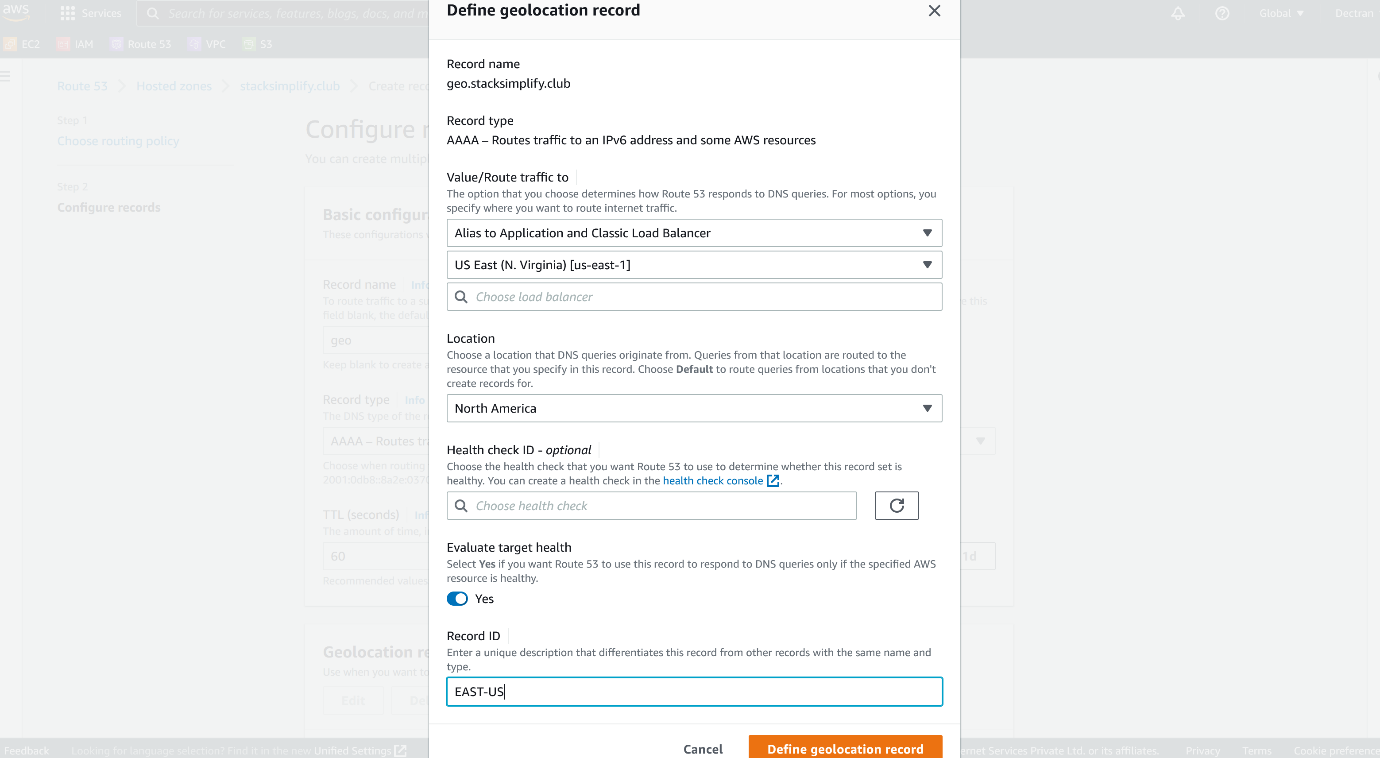
--- click on create a record.



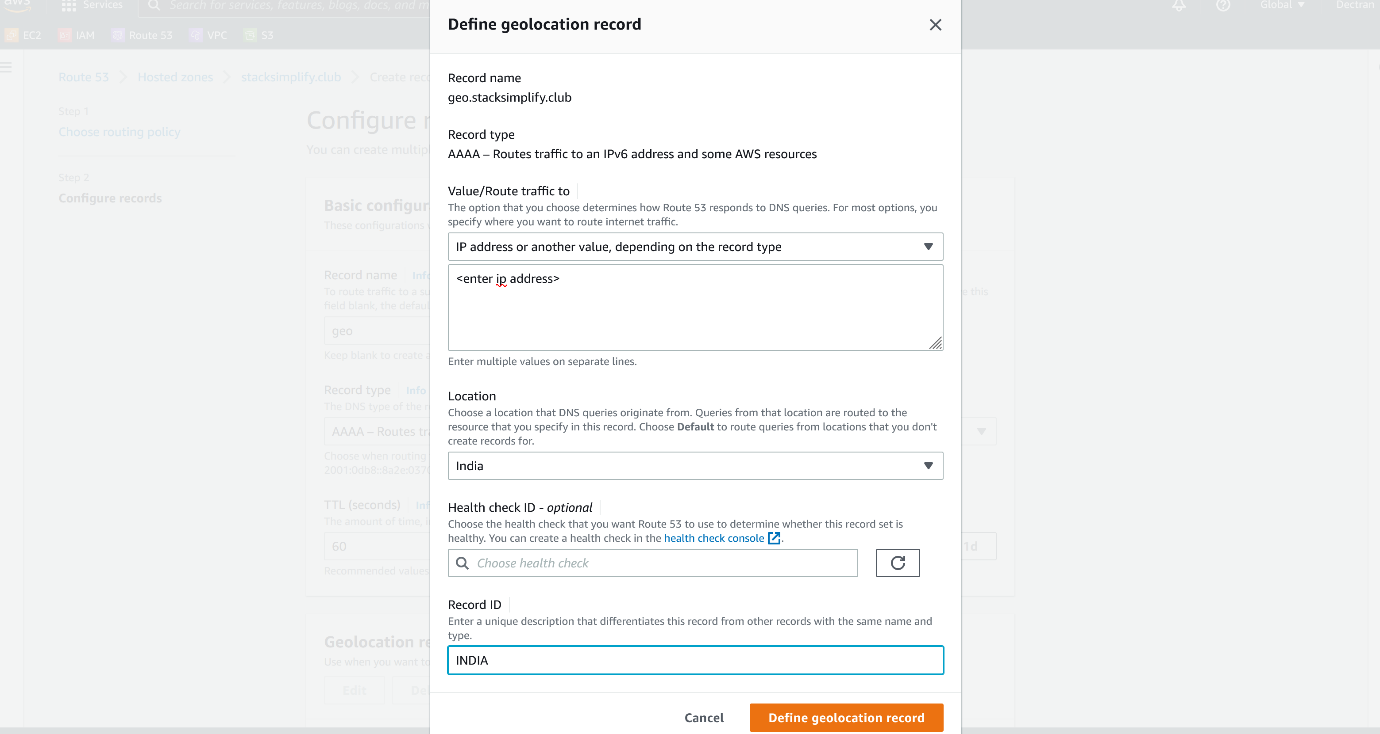
--- click on geolocation.



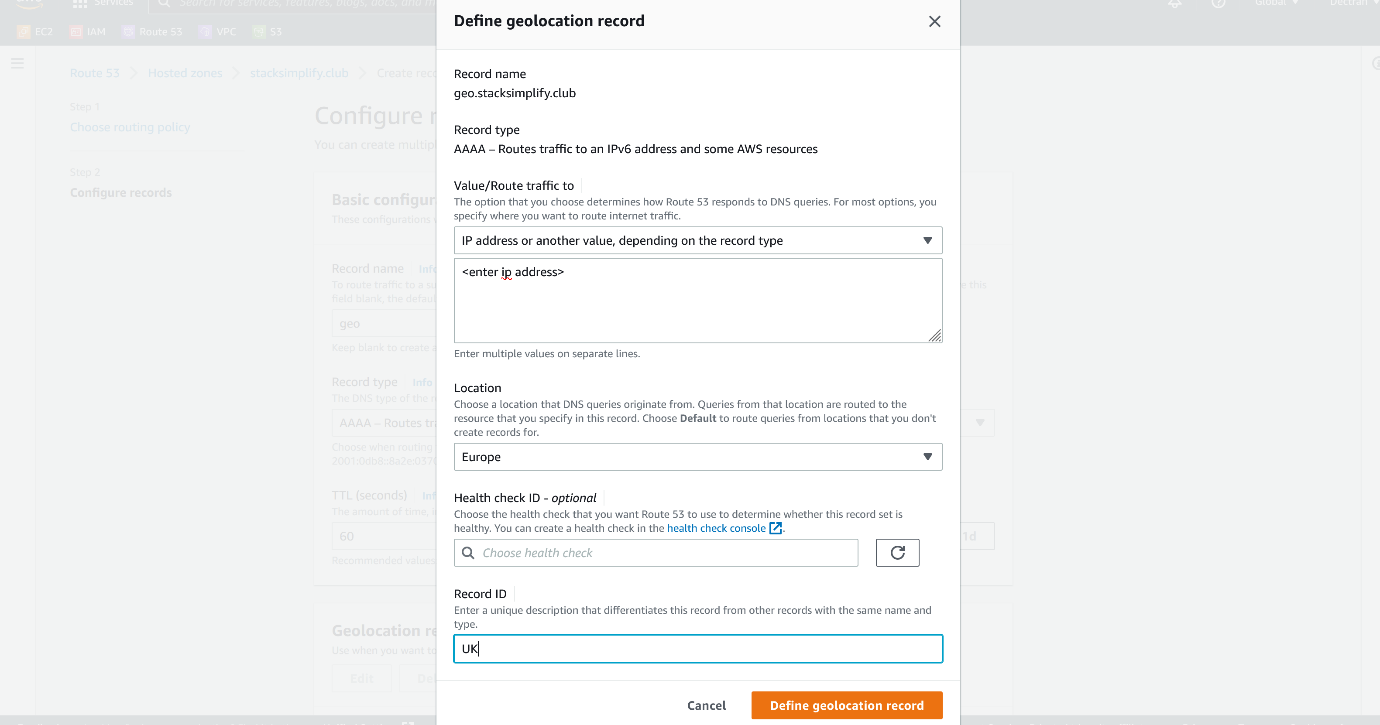
--- click define geolocation record.



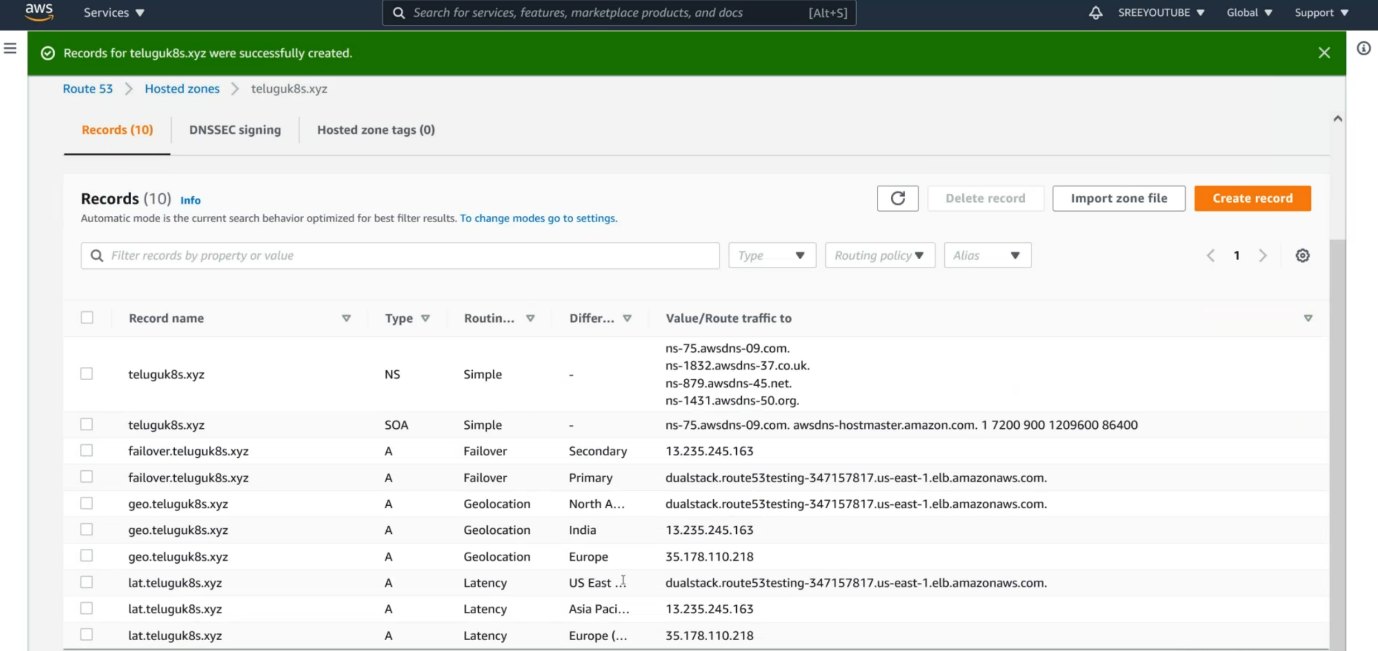
--- **note** – add another record.



--- Added India record. People from India will connect to this server.



--- now server for Europe is added, people from UK will connect to this server.



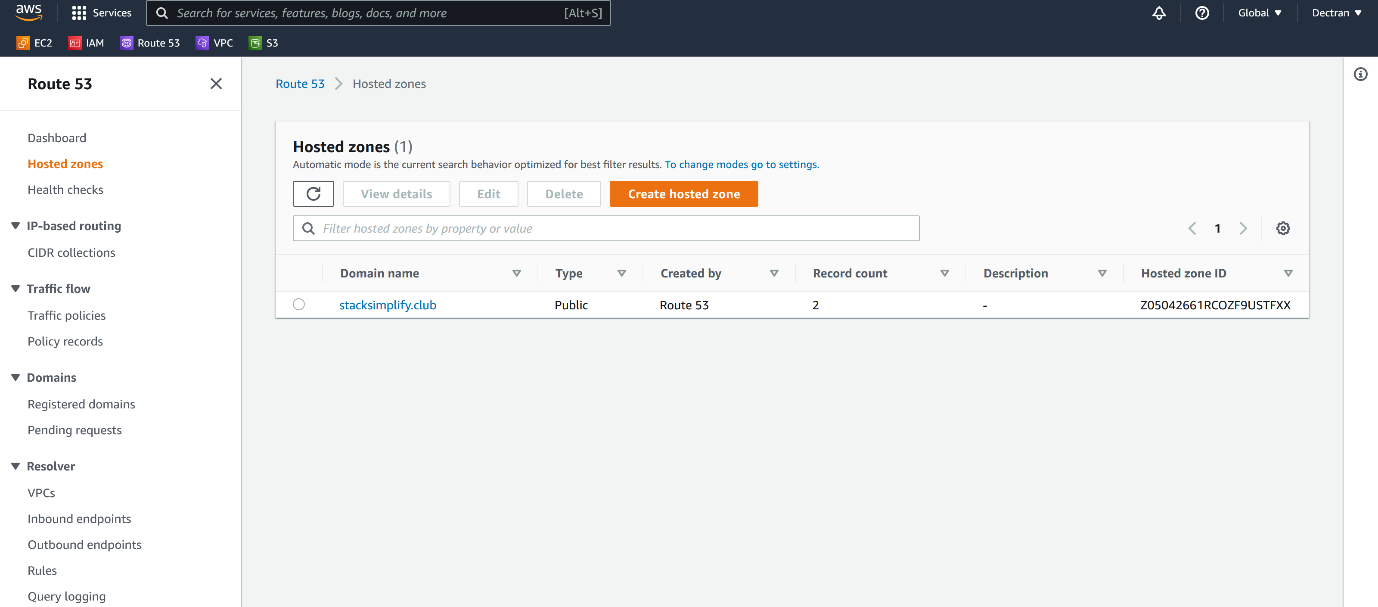
--- geolocation added.

**Weighted routing policy**

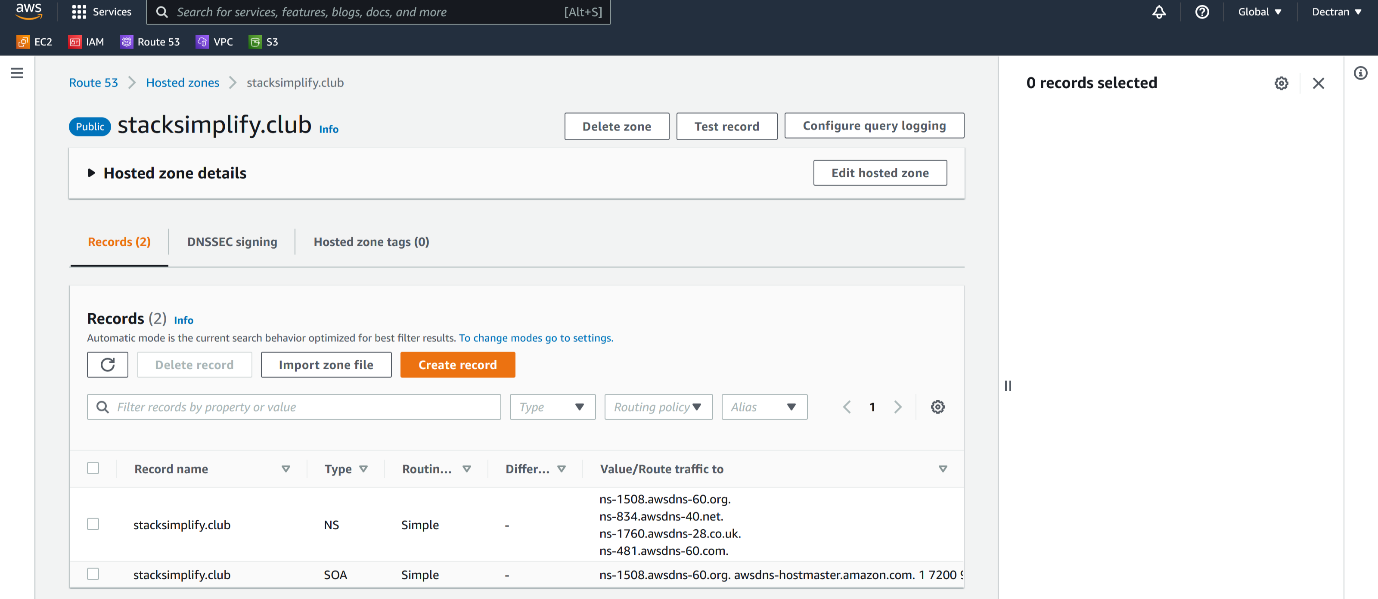
--- Use to route traffic to multiple resources in proportions that you specify. You can use weighted routing to create records in a private hosted zone.

--- **scenario** – I want to send more traffic to east-us and India, want to send less traffic to UK then use this weighted routing policy.

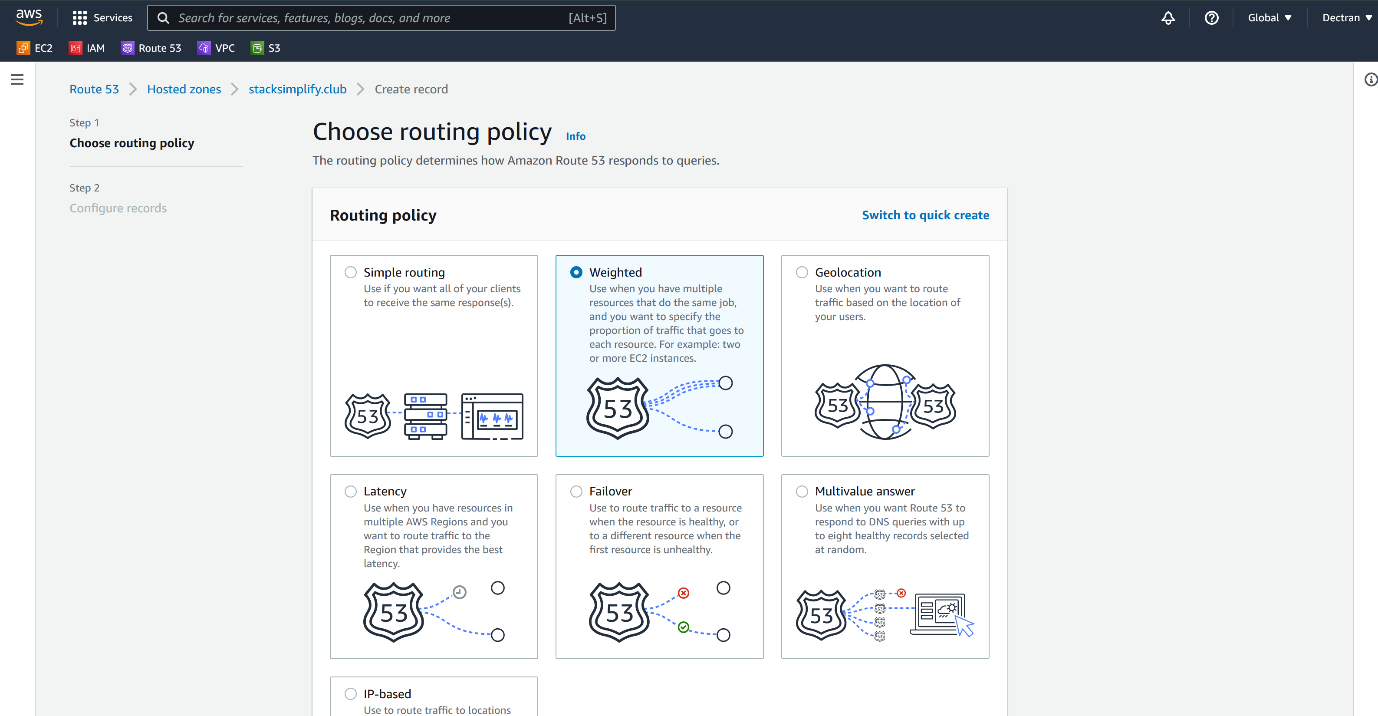
--- go to the aws route53.



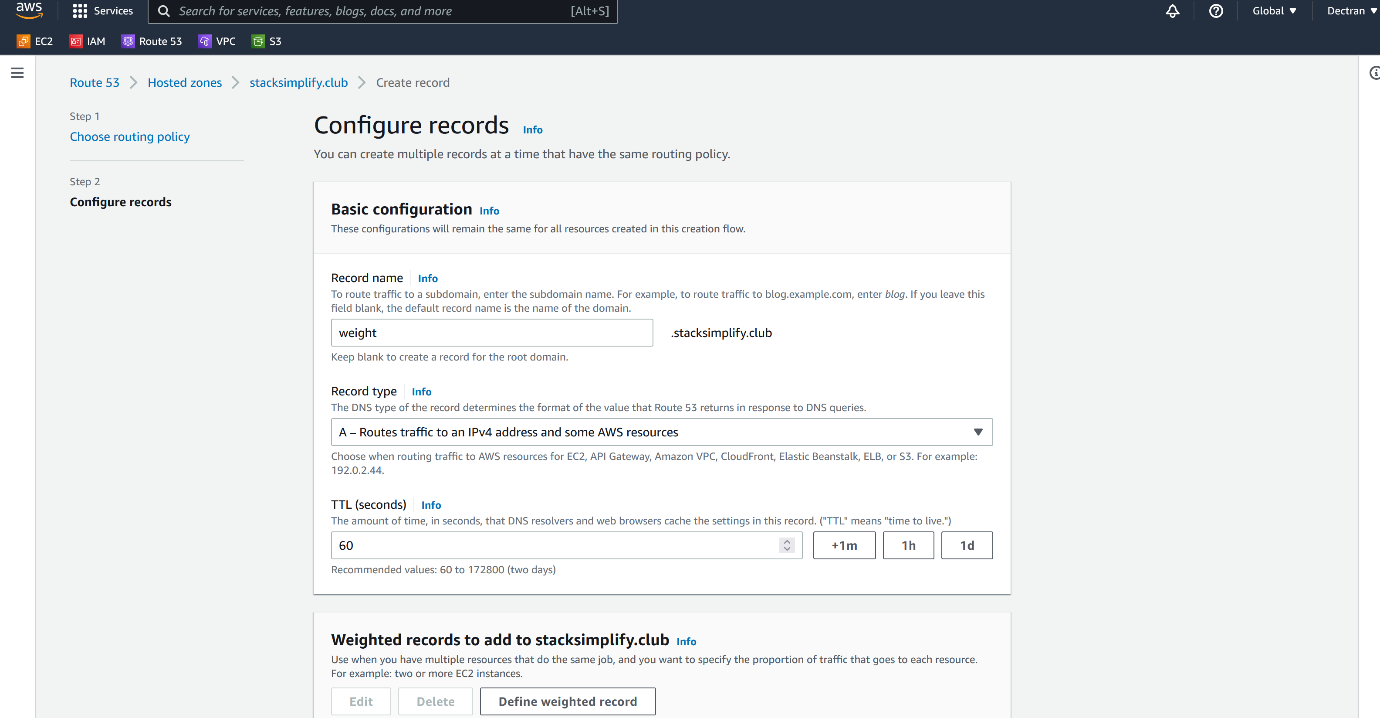
--- click on hosted zone.



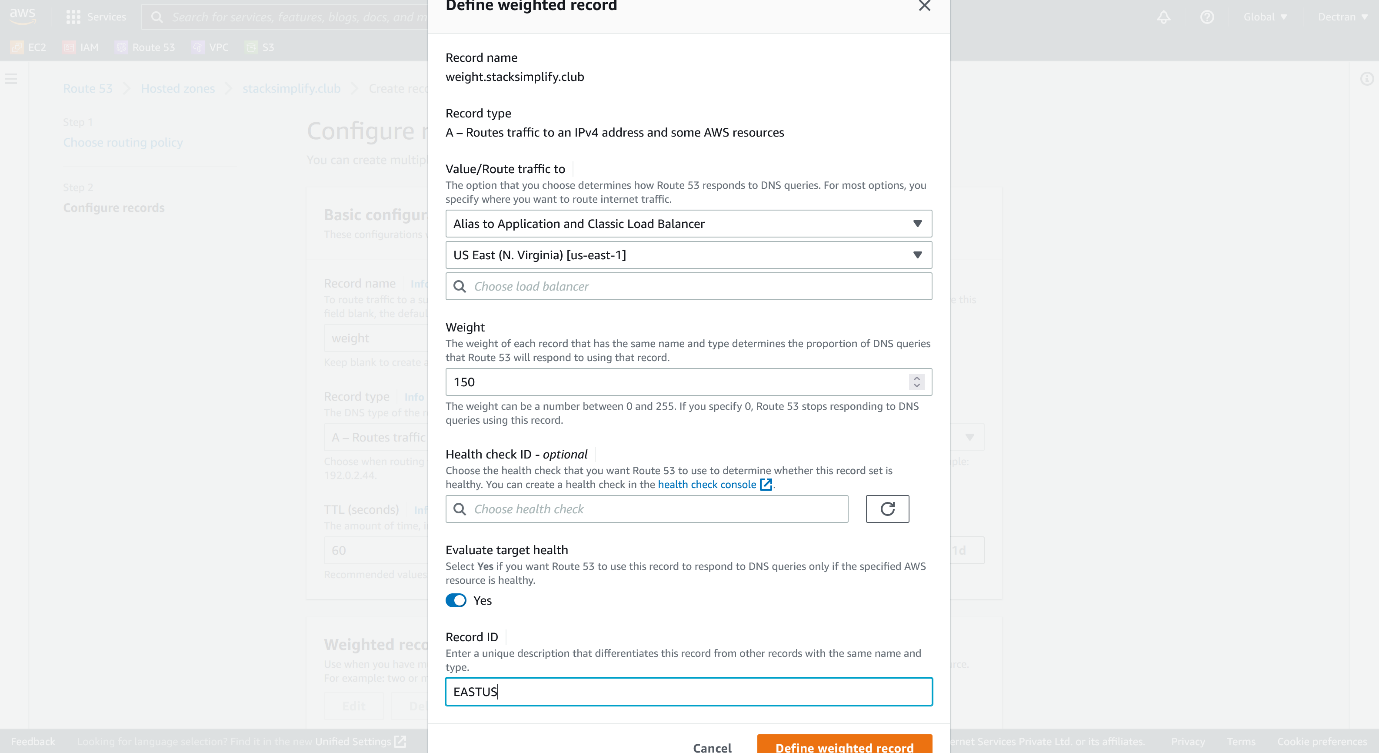
--- click on create a record.



--- select weighted and click on next

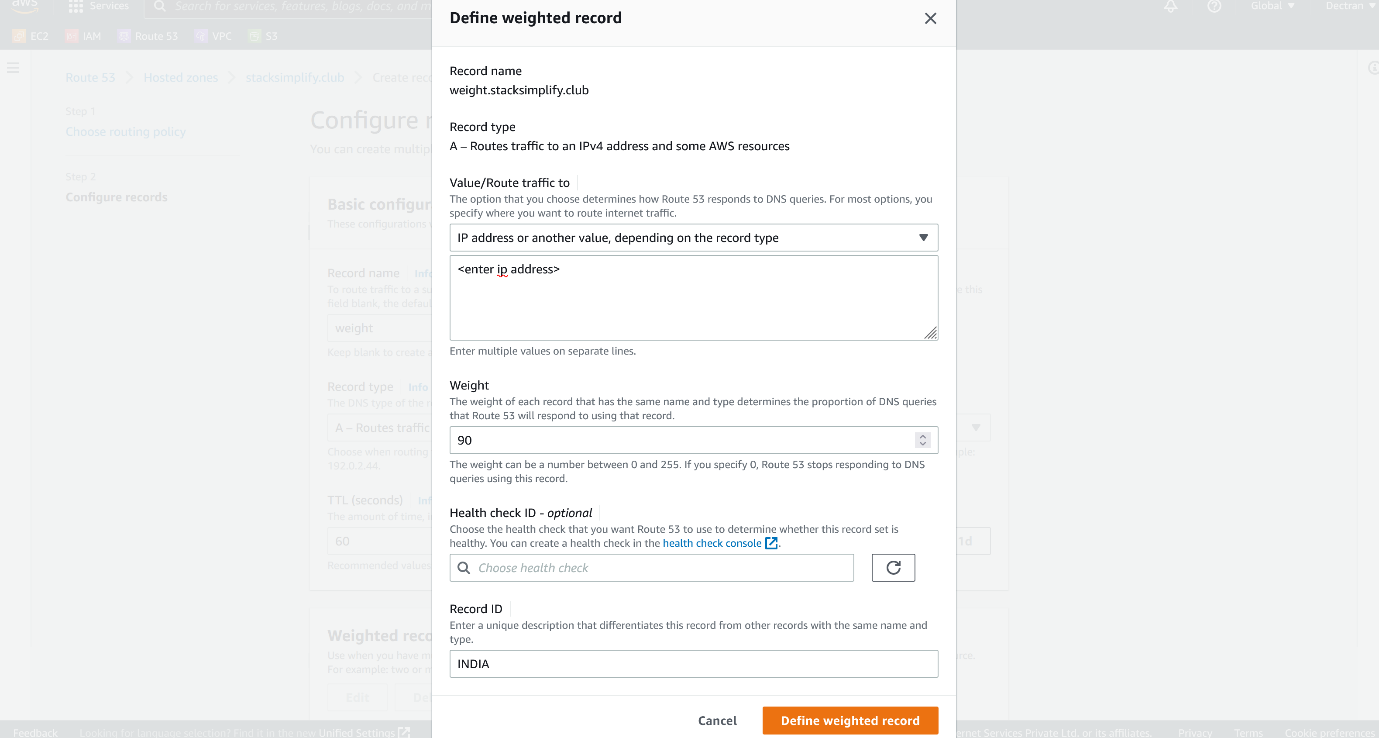


--- click on define weighted record.

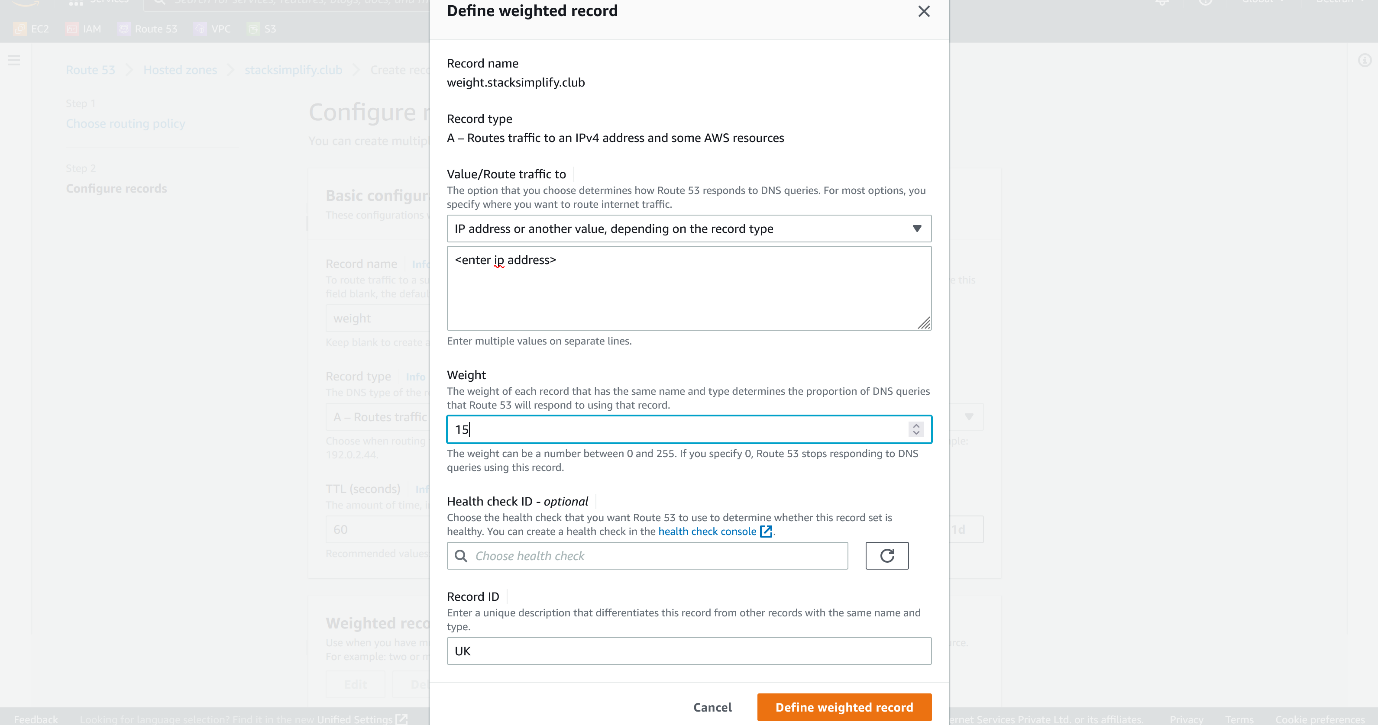


--- **NOTE** – scale of 0 – 255, I am giving 150 to EASTUS SERVERS, this much of traffic will be routed to EASTUS servers.

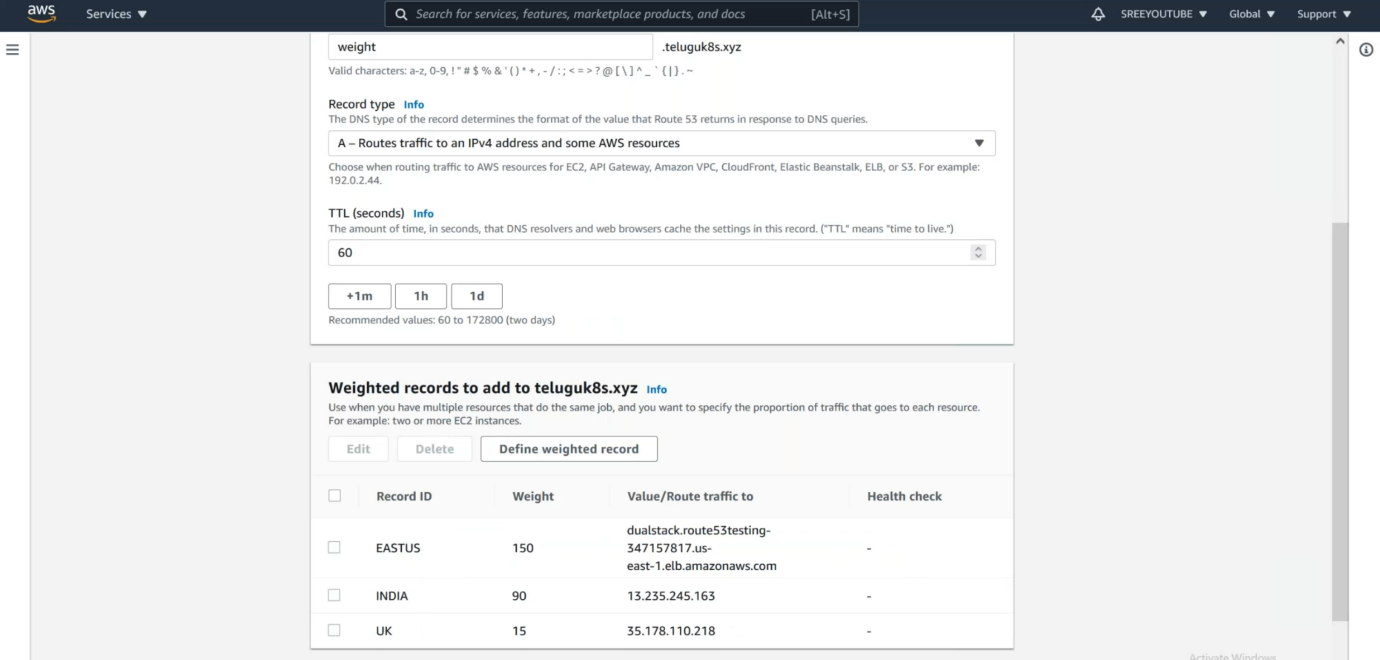
--- add another record for Indian servers.



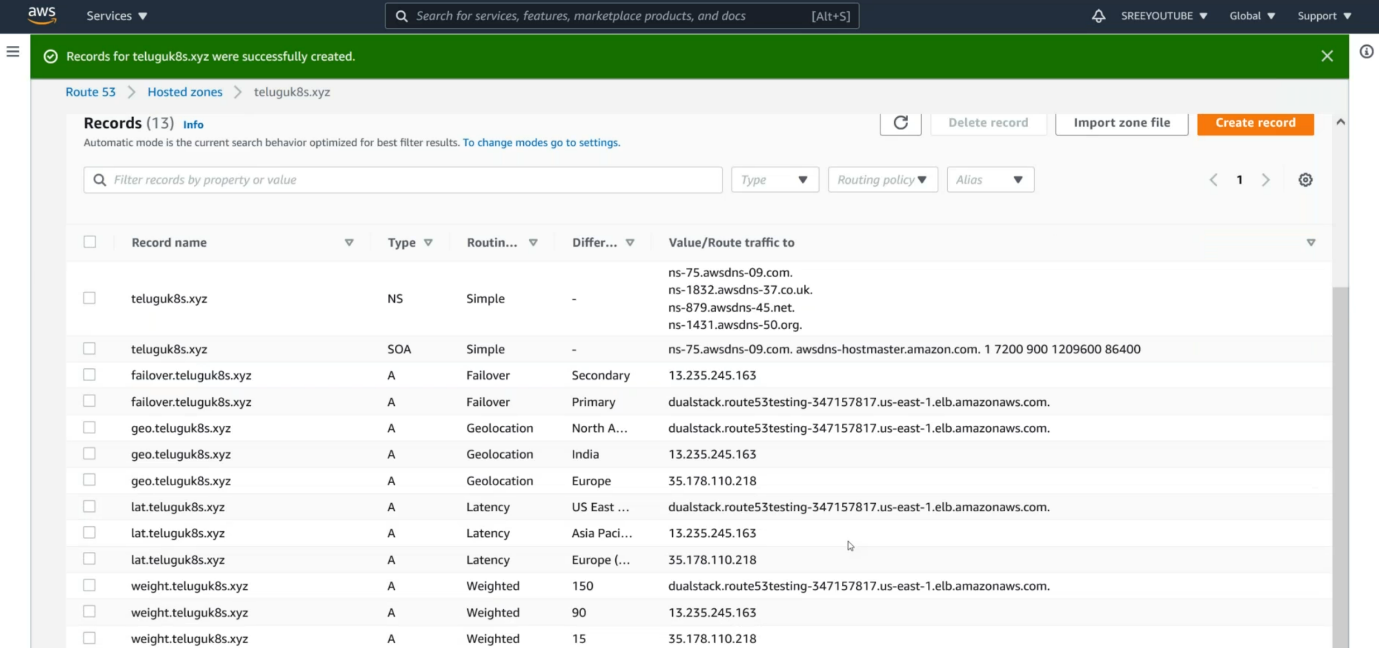
--- **NOTE** – here the weight for the Indian servers is 90, previously I have given 150 for EASTUS and now I am given 90 for Indian server. The total is 240. The remaining 15 weight will be assigned to UK servers.



--- click on define weight record.



--- the weight records are added.



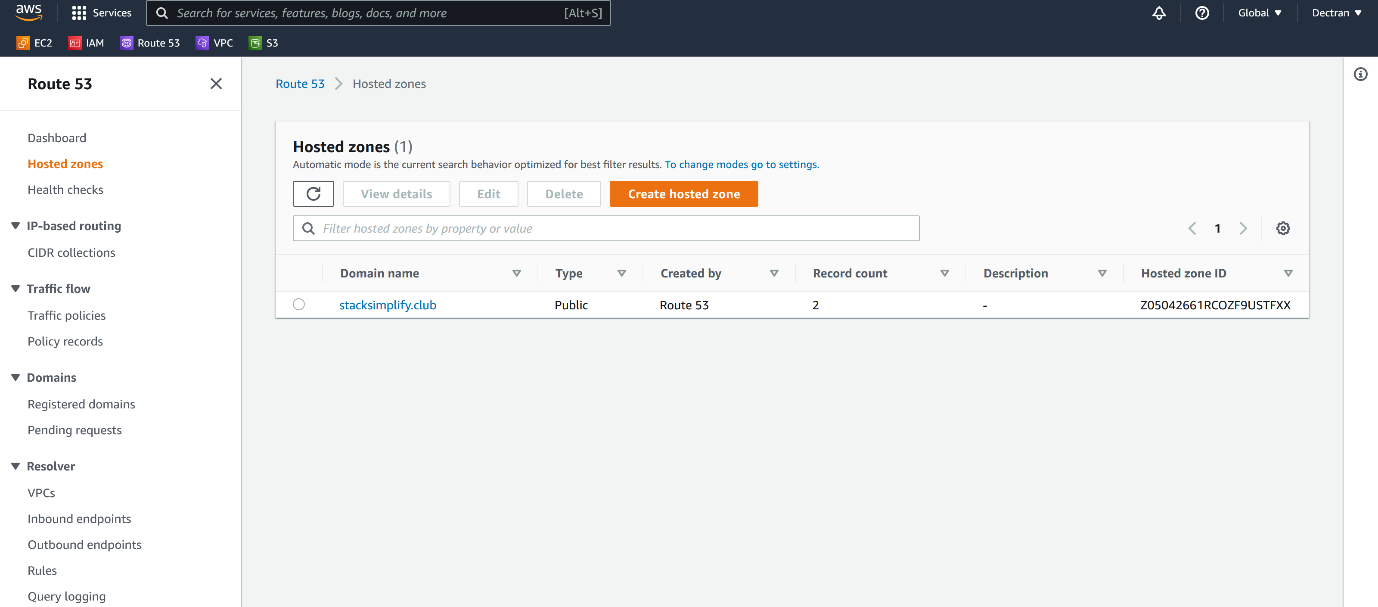
--- this reference purposes.

**Multivalue answer routing policy**

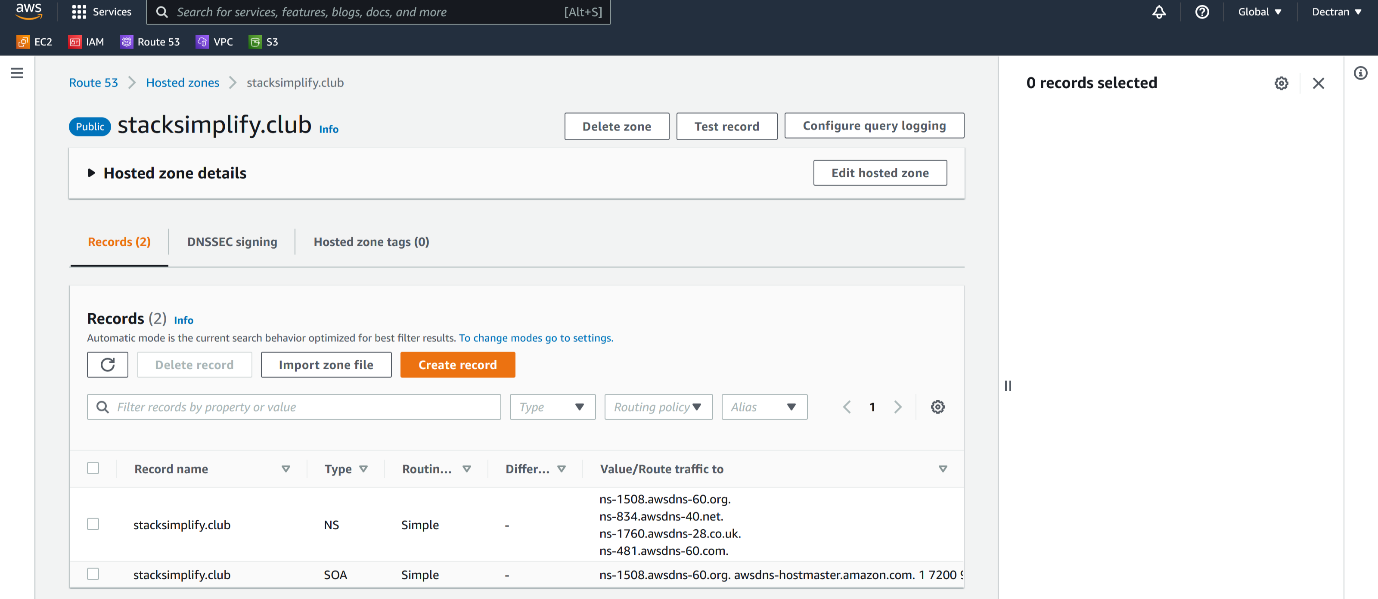
– Use when you want Route 53 to respond to DNS queries with up to eight healthy records selected at random. You can use multivalue answer routing to create records in a private hosted zone.

--- scenario – you do not want route53 to load balance, I have an external application that will act as a load balancer. In this scenario we will use multivalue answer routing.

--- go to the aws route53.



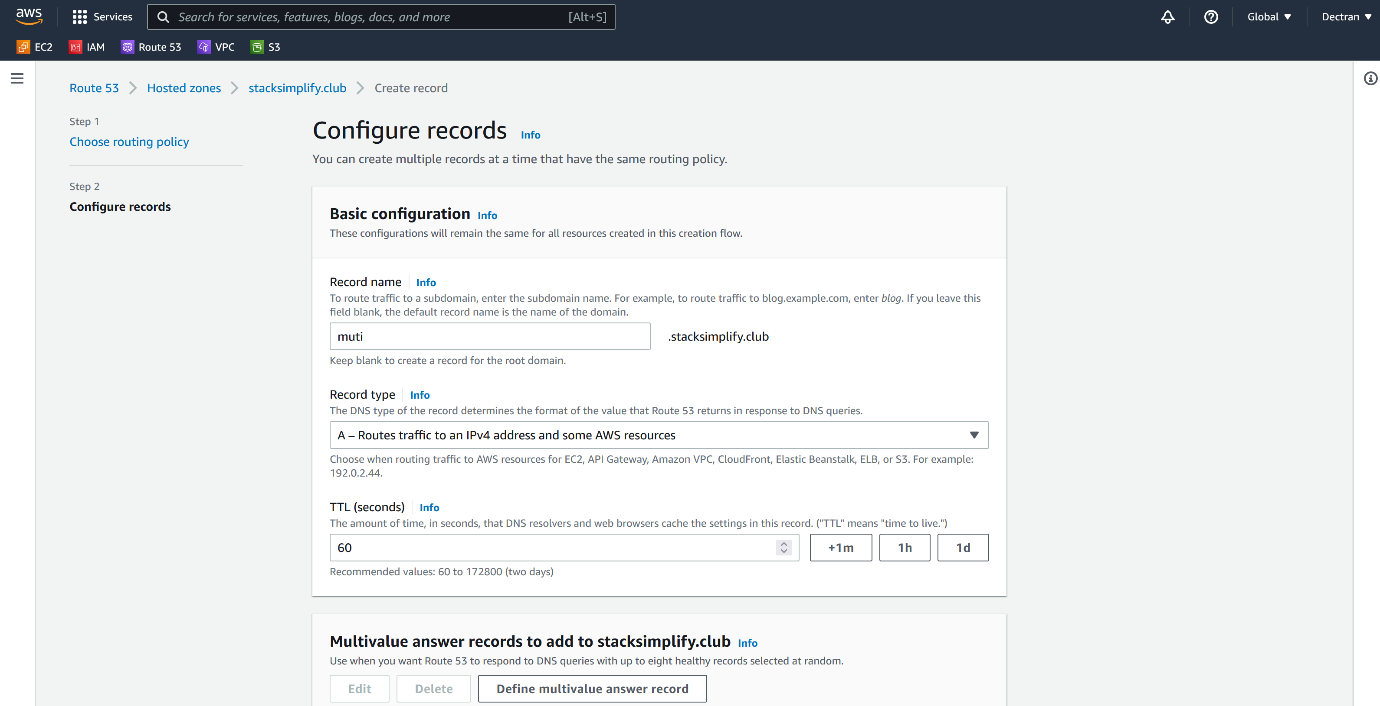
--- click on hosted zone.



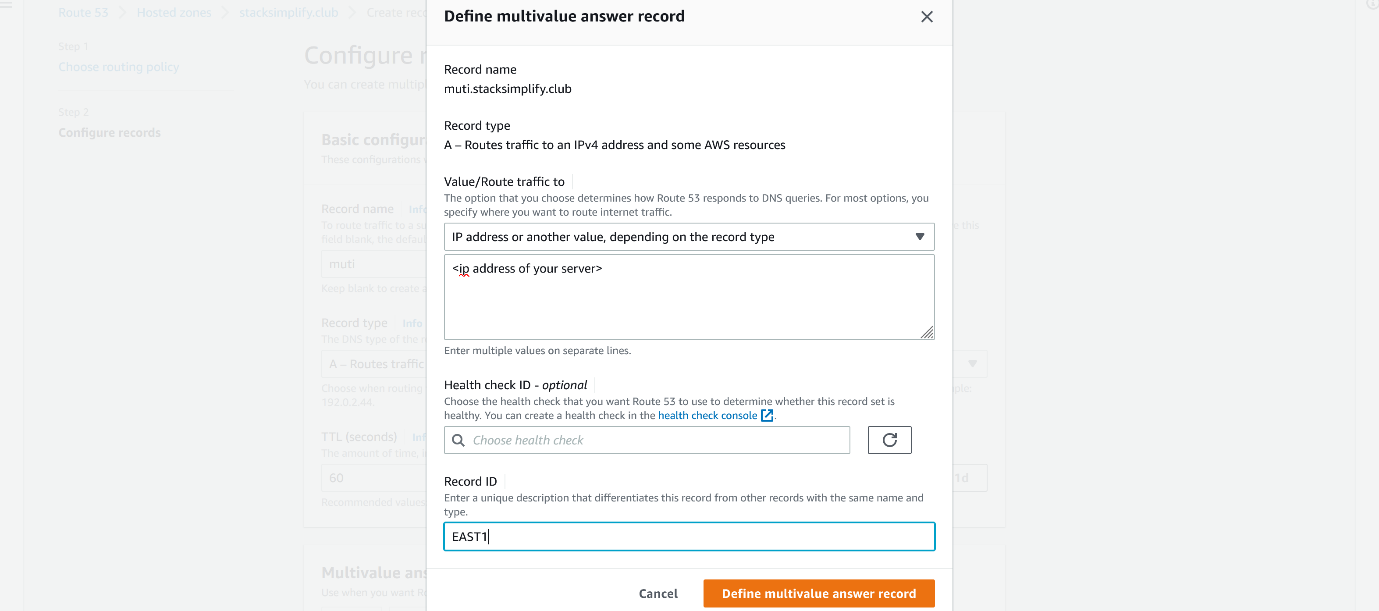
--- click on create a record.



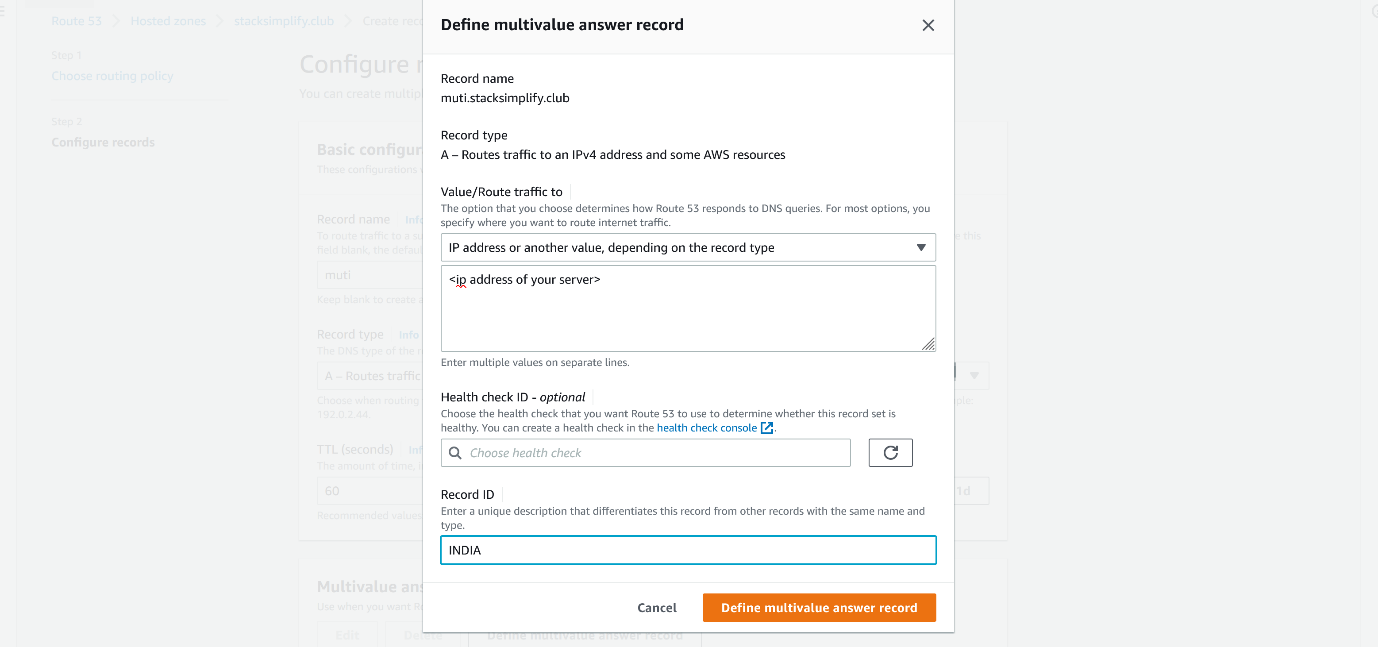
--- click on multivalue answer.



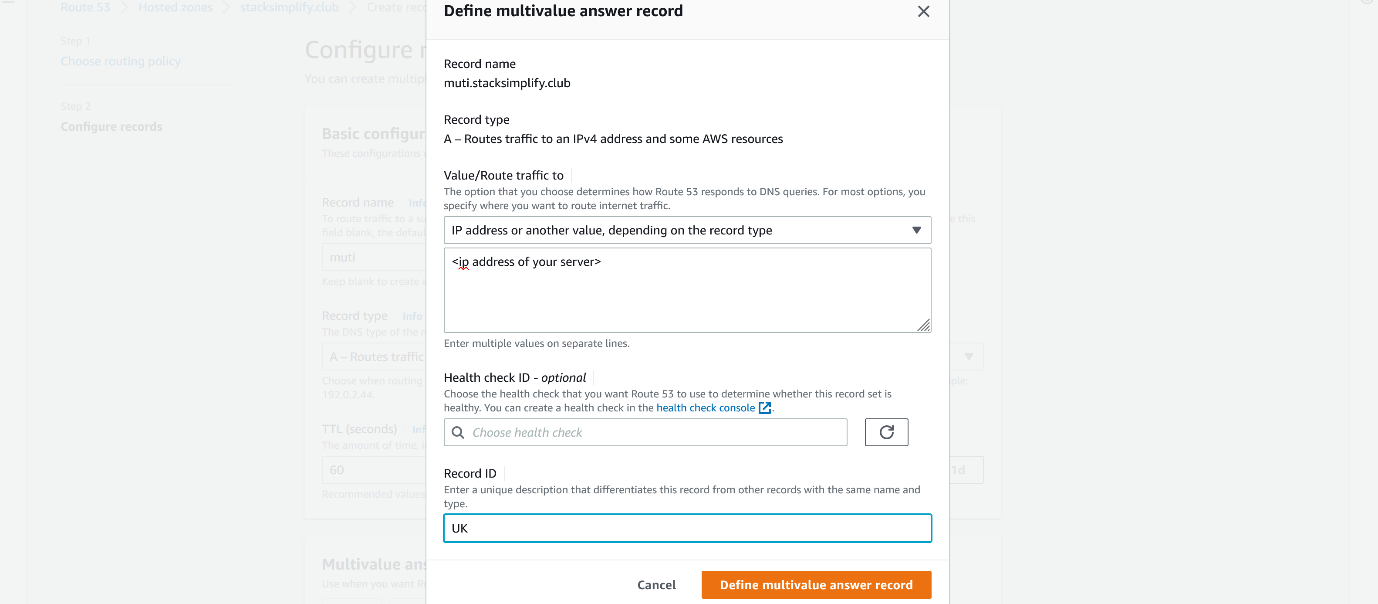
--- click on define multivalue answer record



--- define another record.

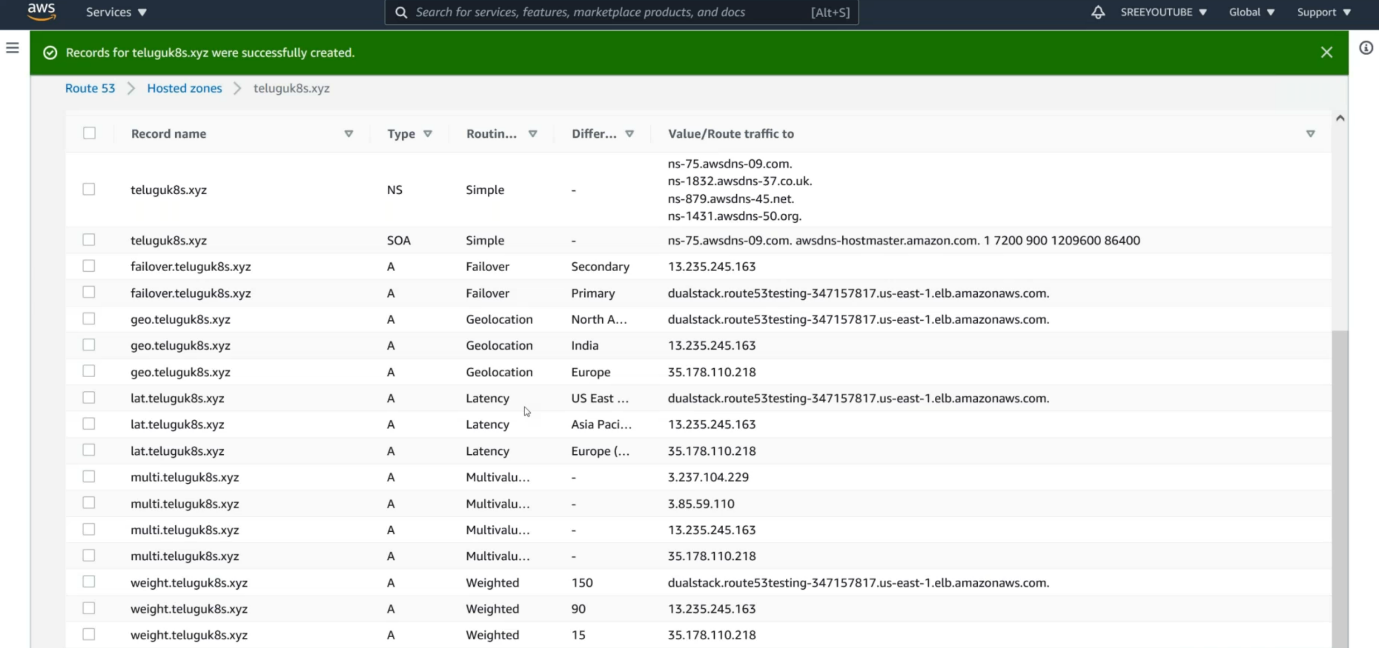


--- define another record.



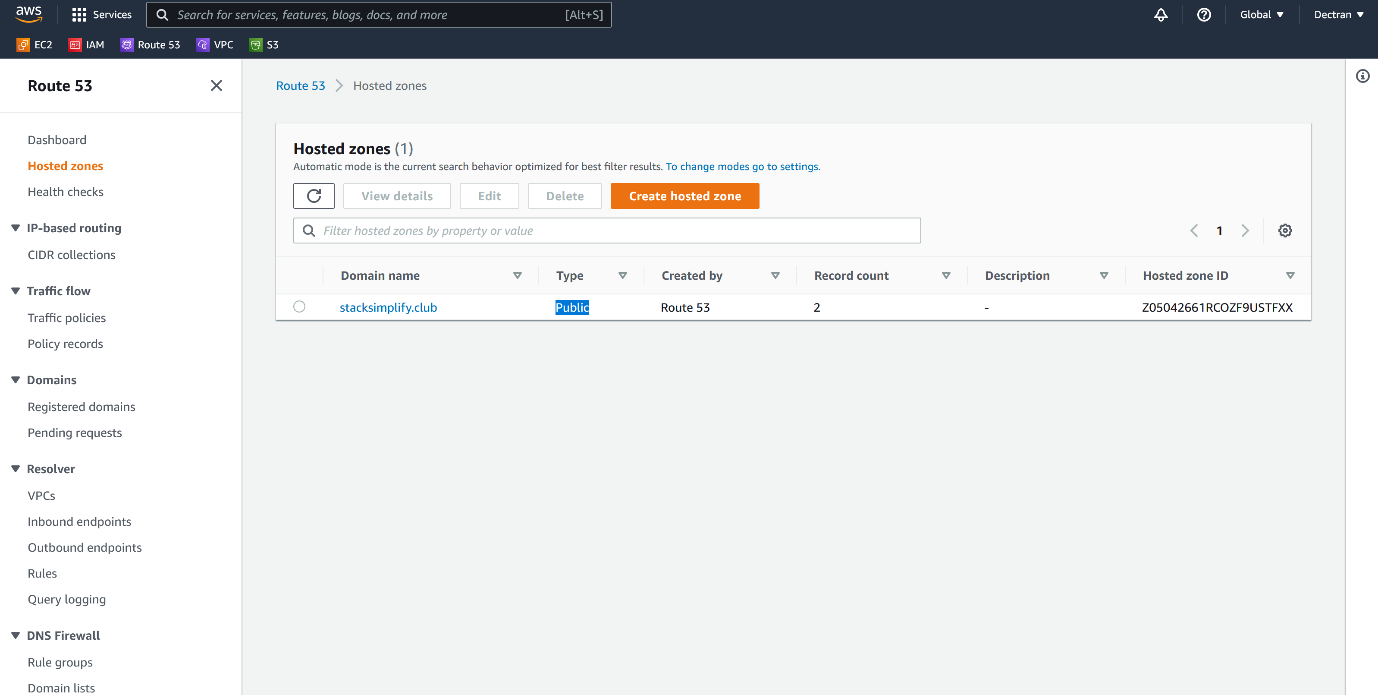
--- note – so far, I have 3 records.

--- nslookup weight.stacksimplify.club



--- my reference only.

**Public hosted zones**



--- **note** – these are called public hosted zones. Anyone from the internet can connect.

--- **nslookup stacksimplify.club** – public hosted zone can be resolved using nslookup.

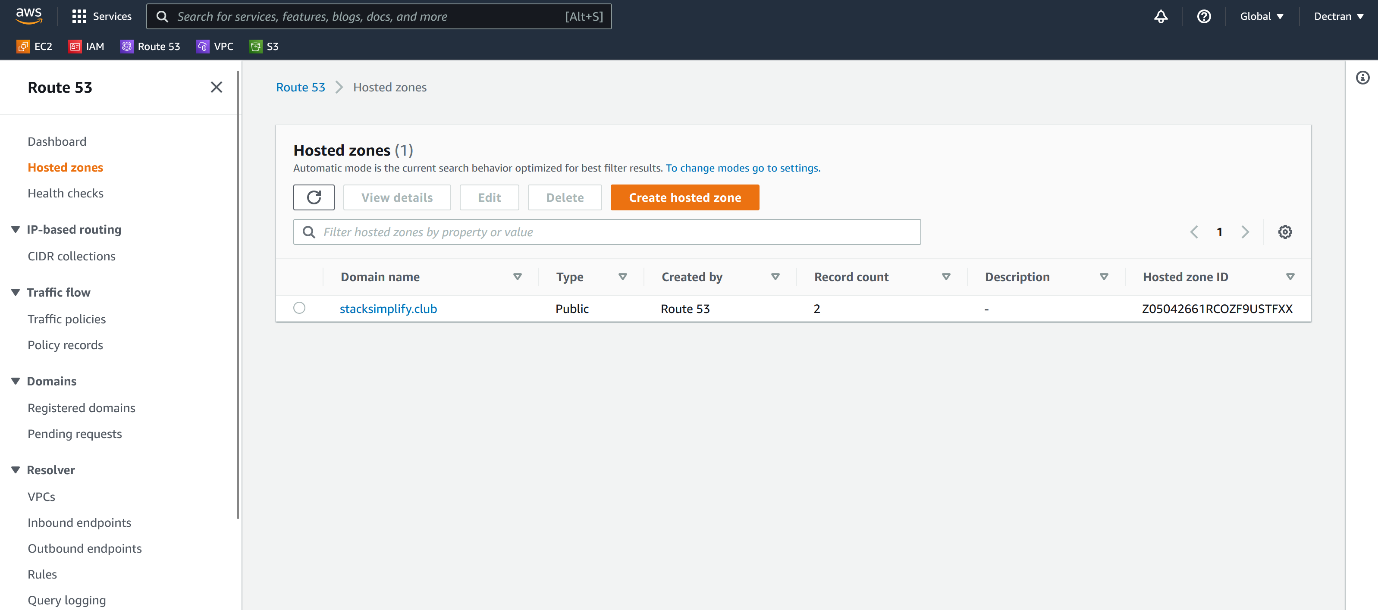
--- **note** – route53 also supports private hosted zone.

**Private hosted zone**

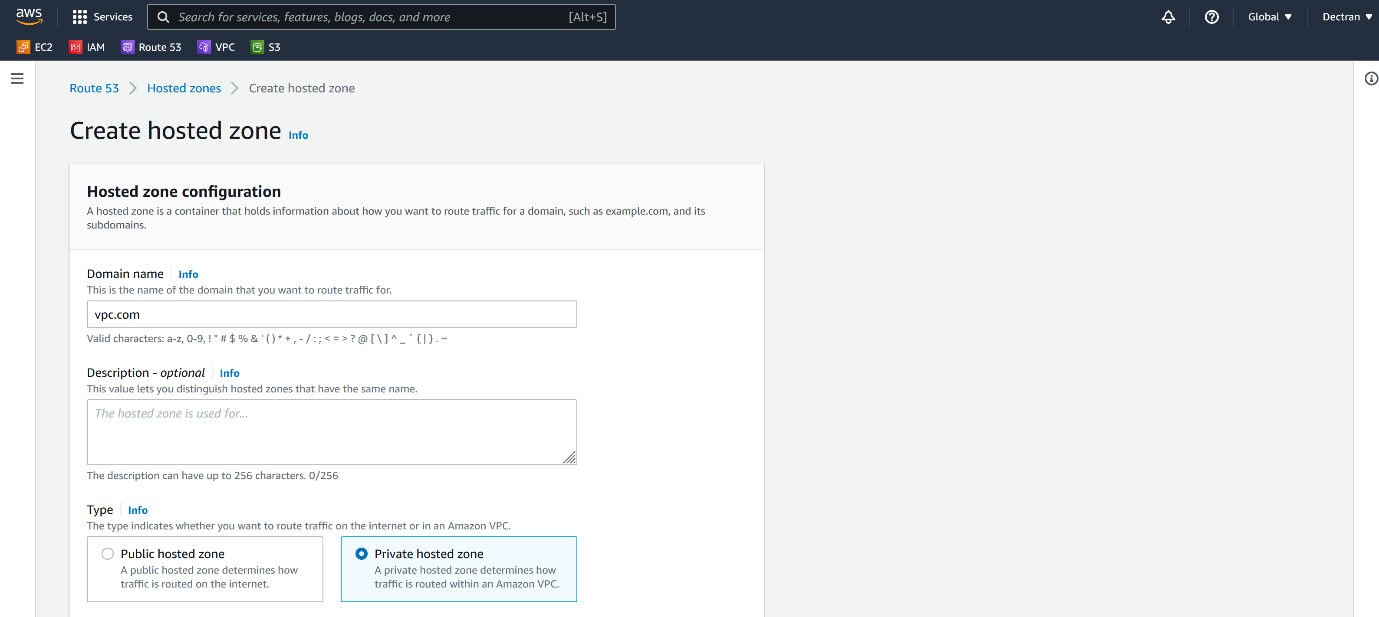
--- **note** – the private hosted zones are less used in real time.

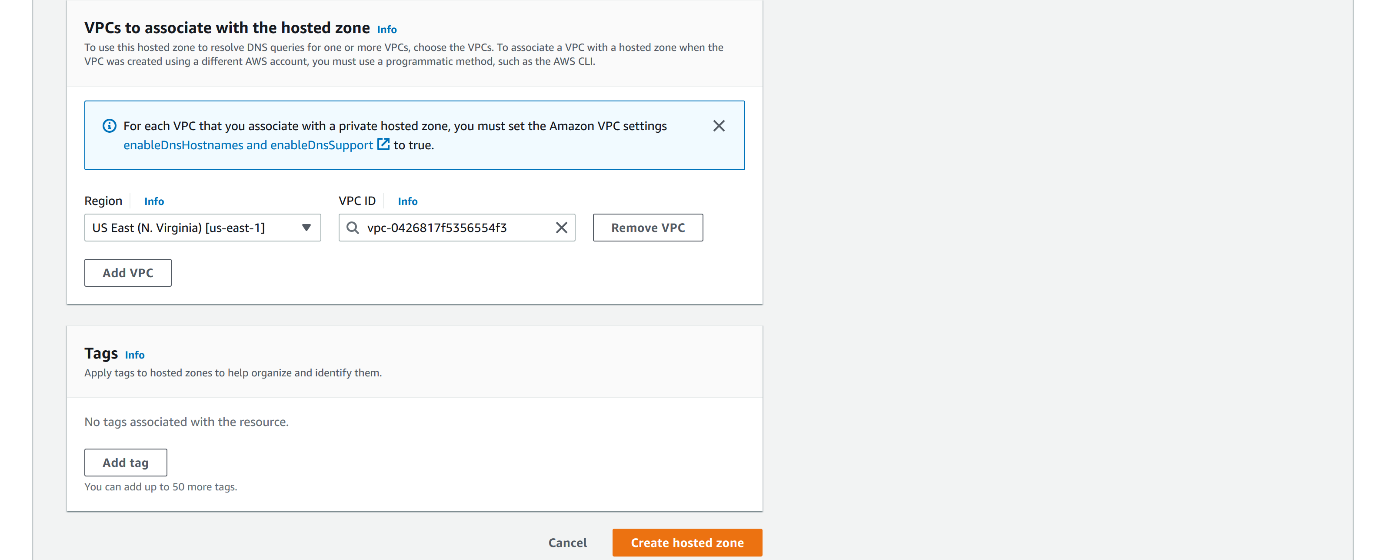
--- note – the domain name of private hosted zone is valid internally, traffic is limited within the vpc.

--- go to the aws route53.



--- click on crate hosted zone.





--- click on create hosted zone.