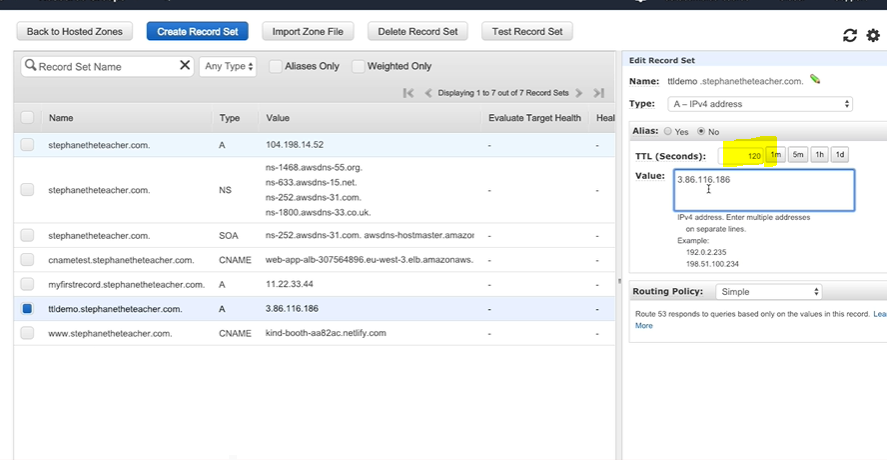
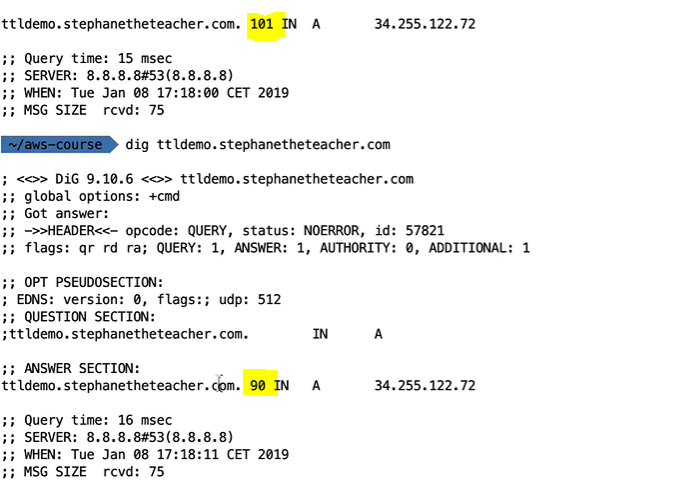
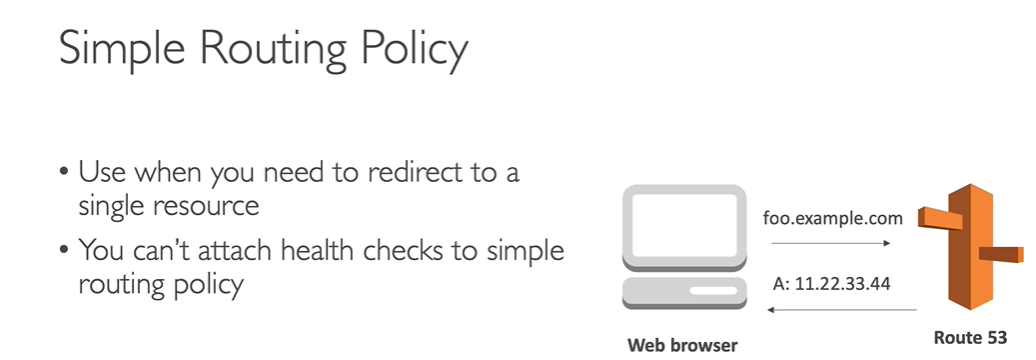


Web browser store the information in cache for 300sec, so when next DNS request will not go to Route53 instead it will pick from cache if the request is sent with in 300sec.



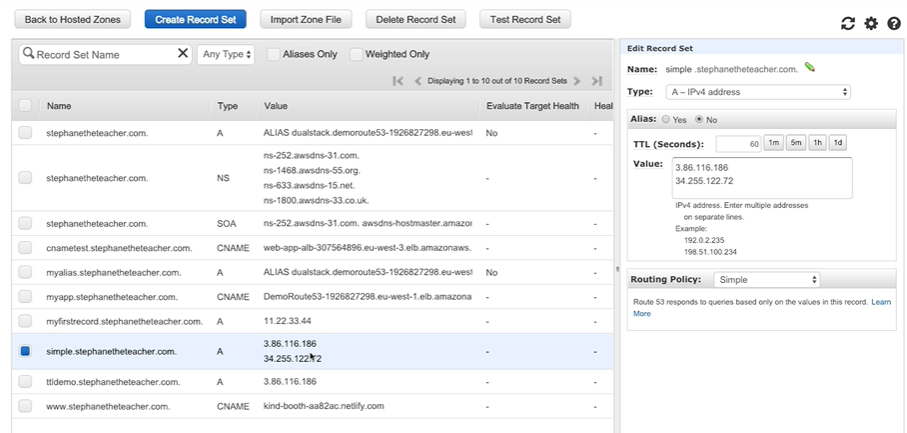
Dig is command to check the TTL time (dig <dns name>)

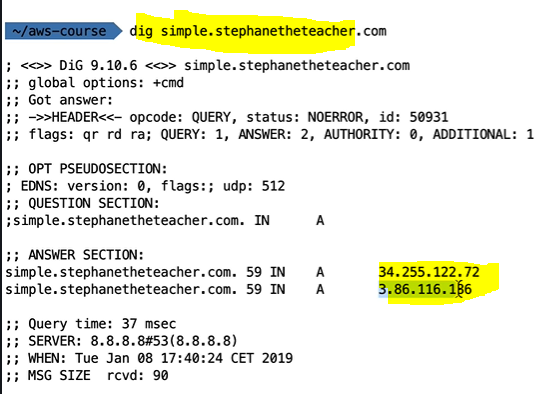


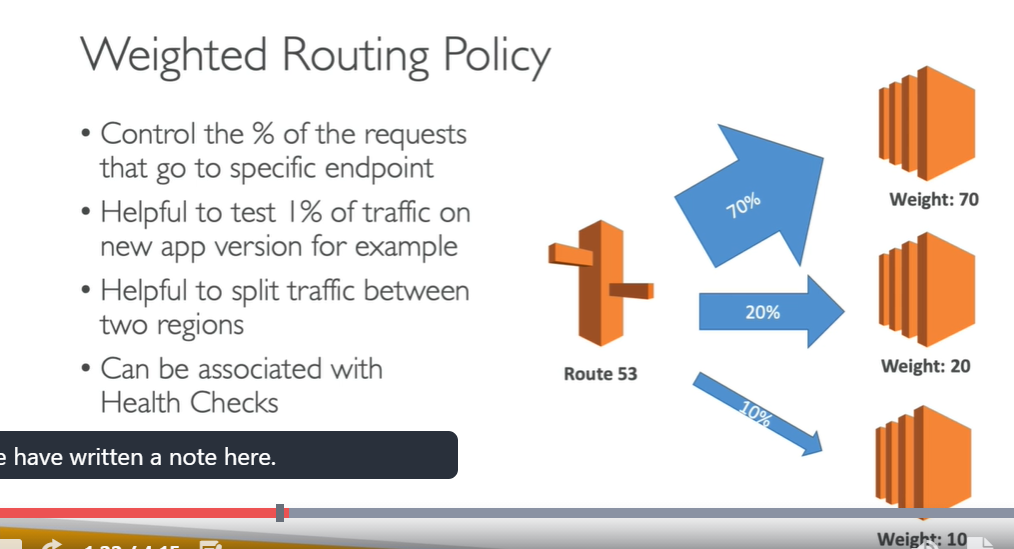


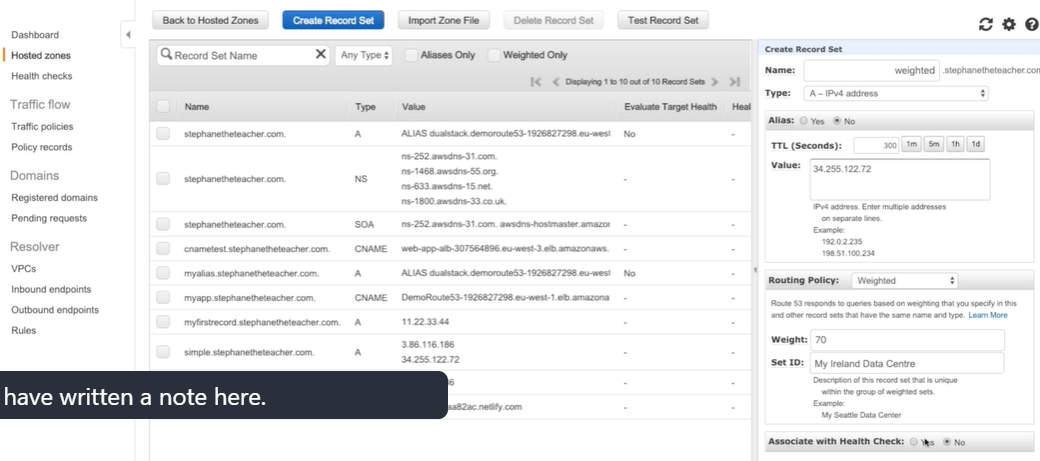
Simple routing: value can be single ip or multiple ips’, if it is single ip request will go to the same server, if the value has multiple ip’s then request will go to one of the server randomly

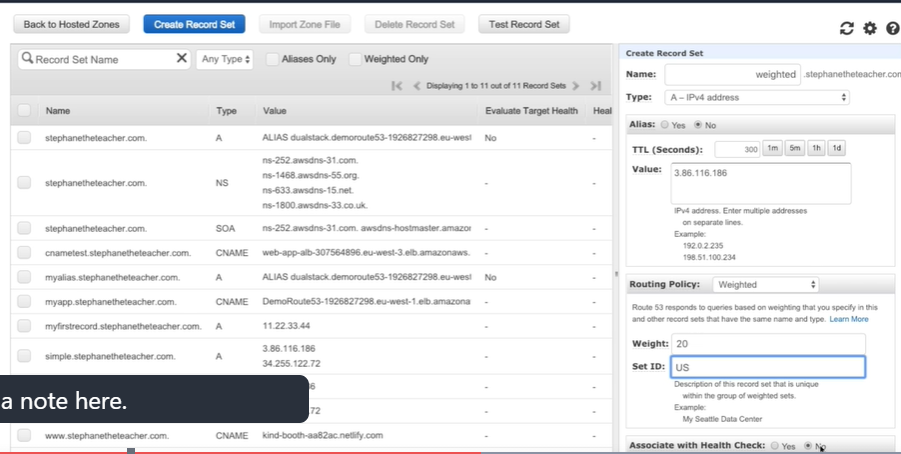
Note: Session refresh will after TTL time(60 sec as per the below) to observe the request routings.

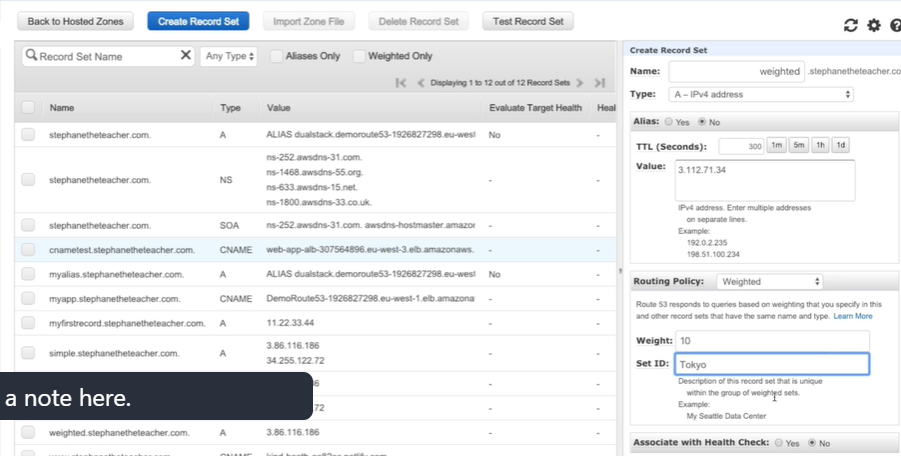




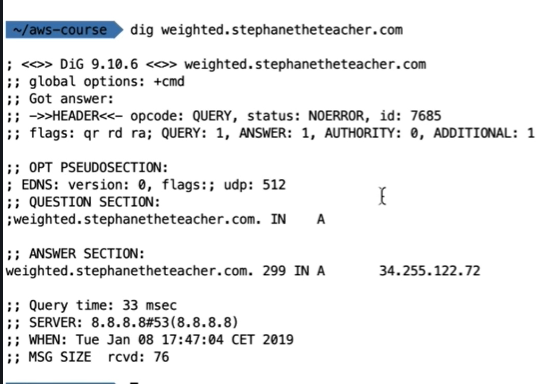


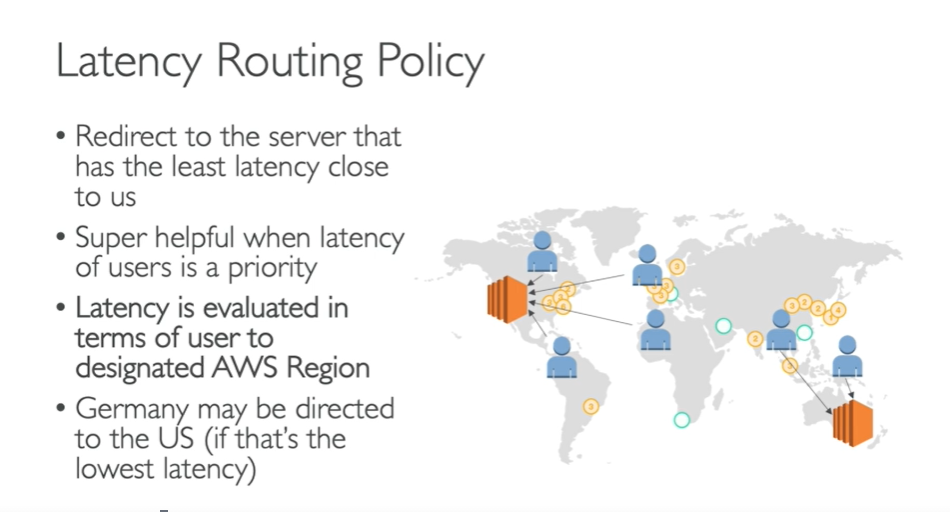




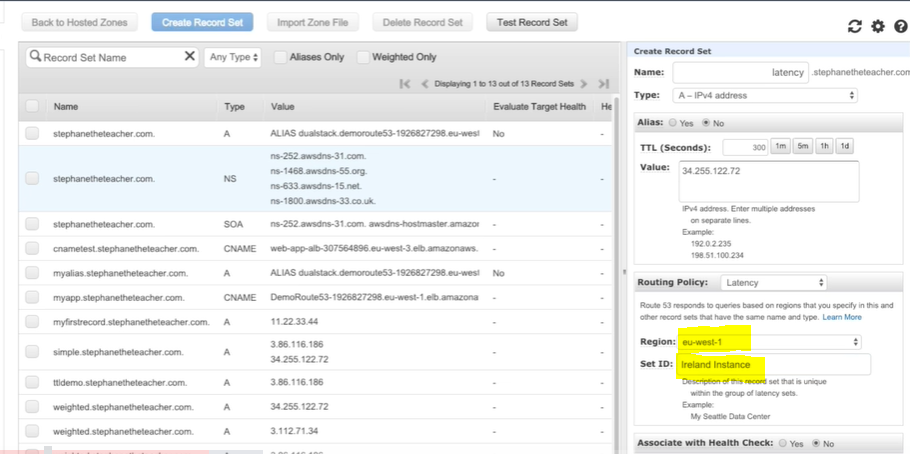


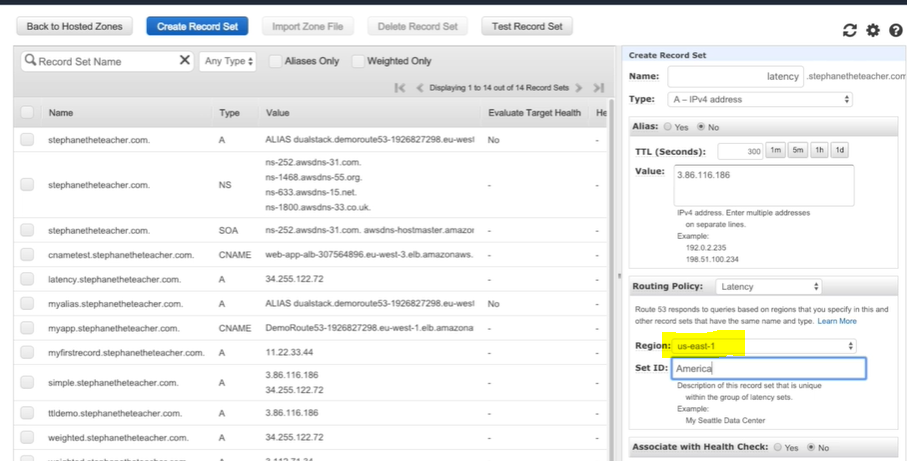
IF we dig the DNS , we will get one ip not all the three , it will work on the weight based





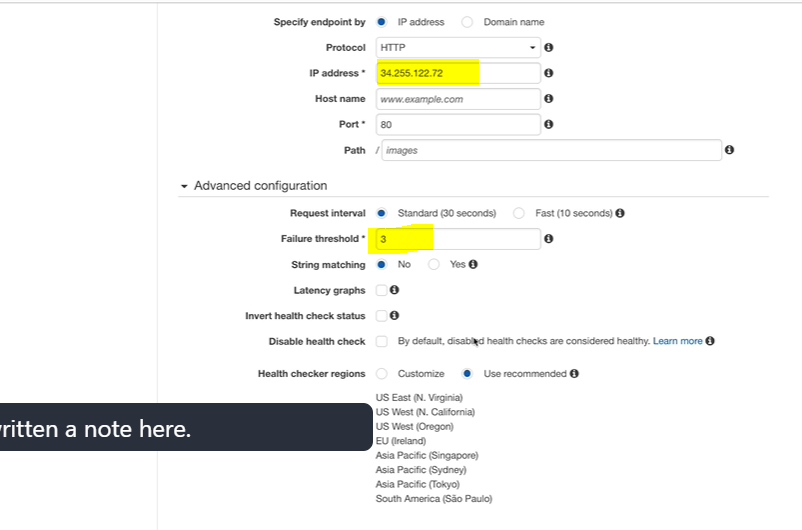
Create record sets based on region

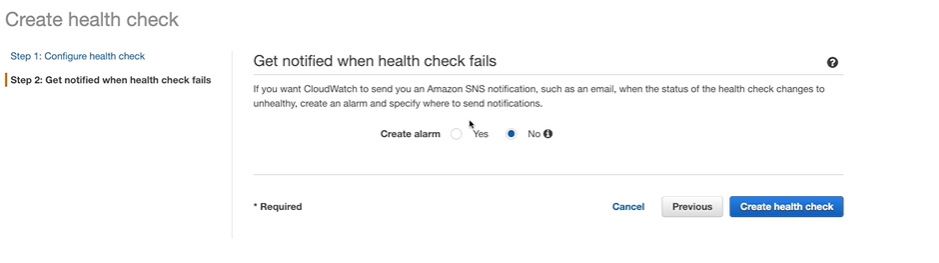




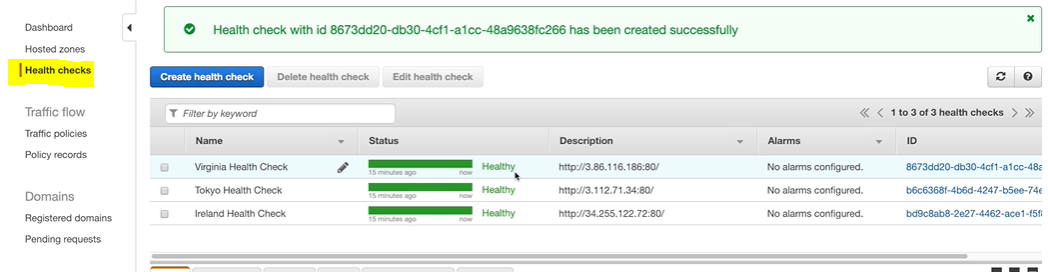
Route53 will check the request from which region they are coming based on that it will route to the corresponding servers, so that low latency will be maintained

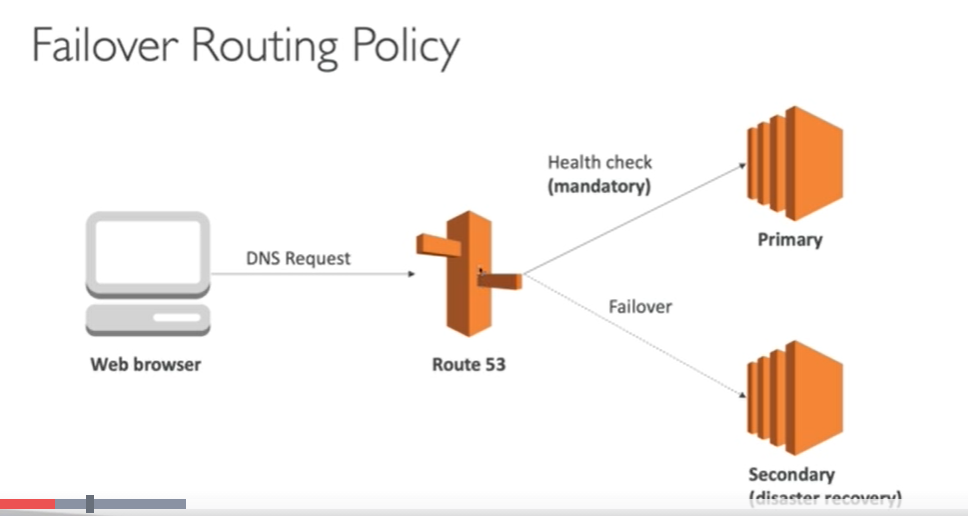
Configuring Health Checks:

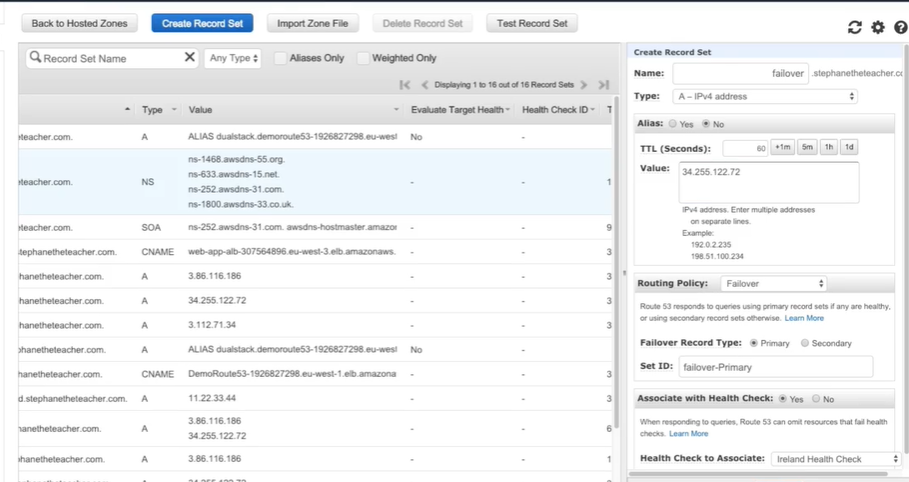


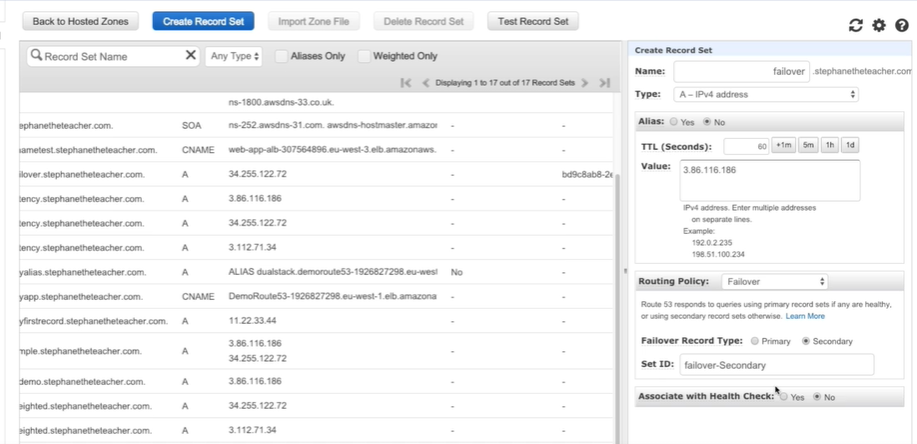


Create health checks for the endpoints(for each server)



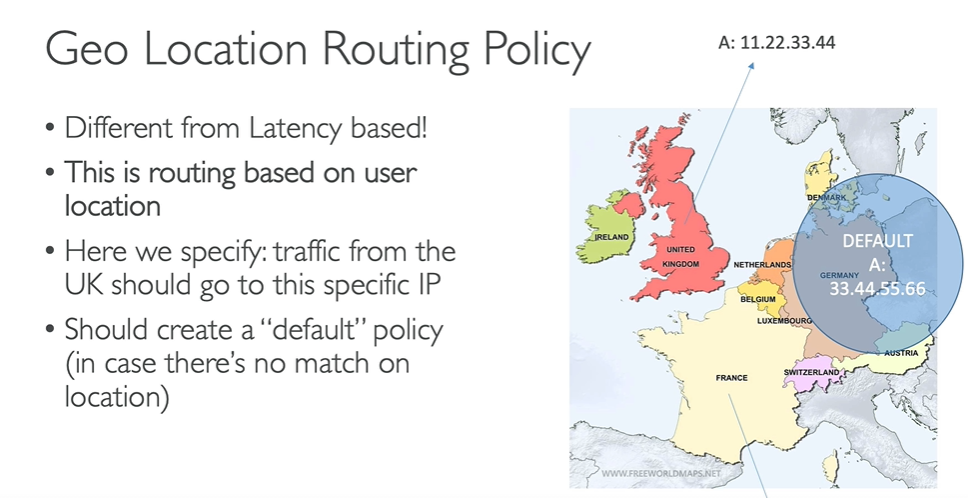


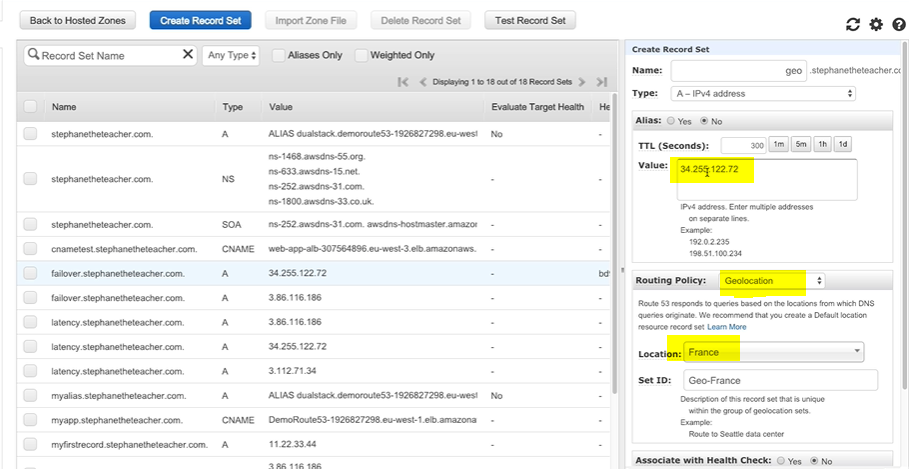


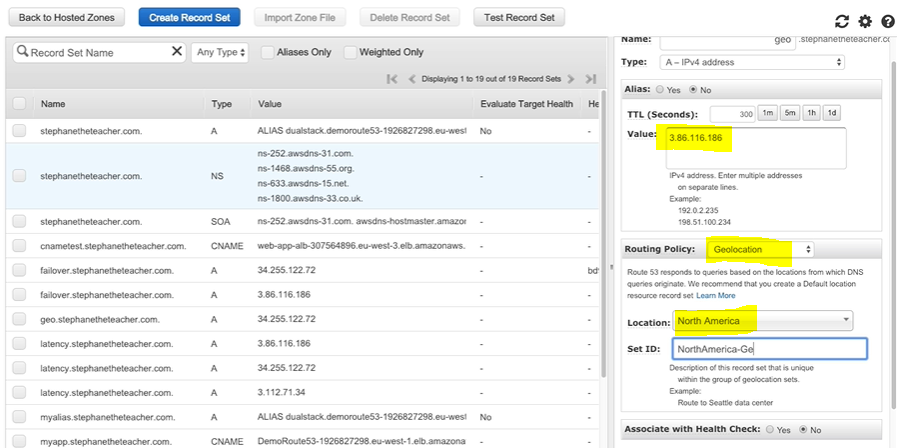


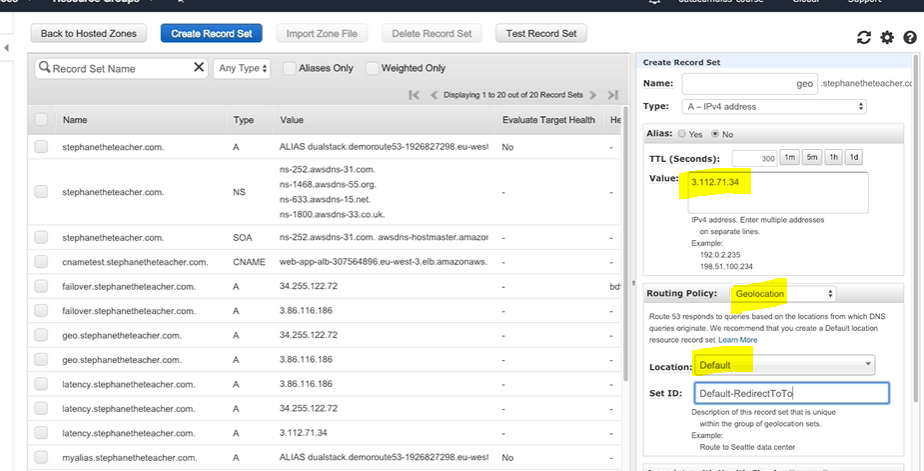
For secondary not required to select health check , it will pick by default.

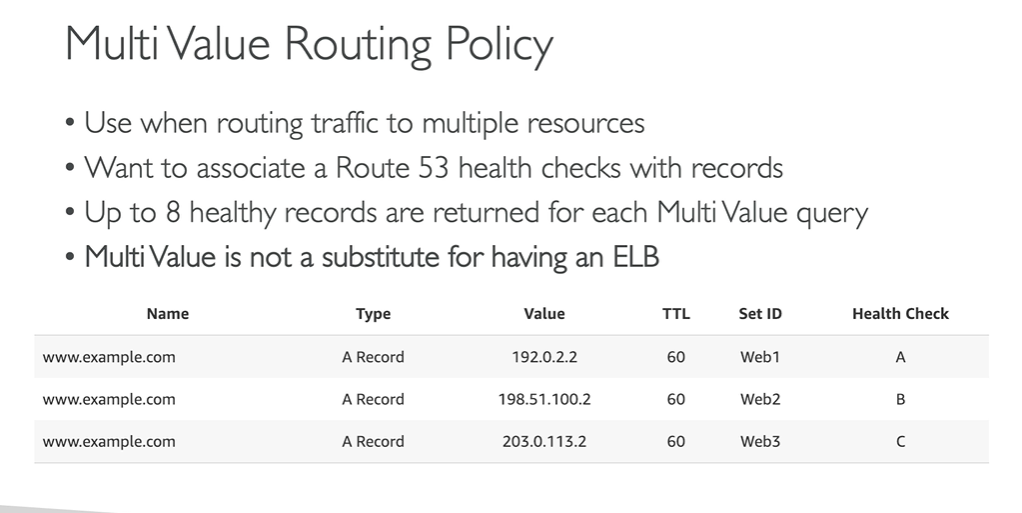
To validate : Stop the primary server , health check will be failed , traffic will be routed to secondary server.

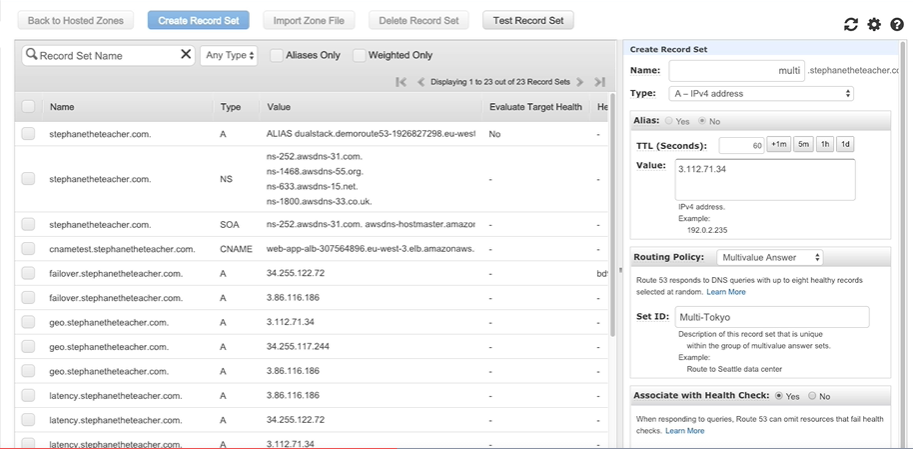


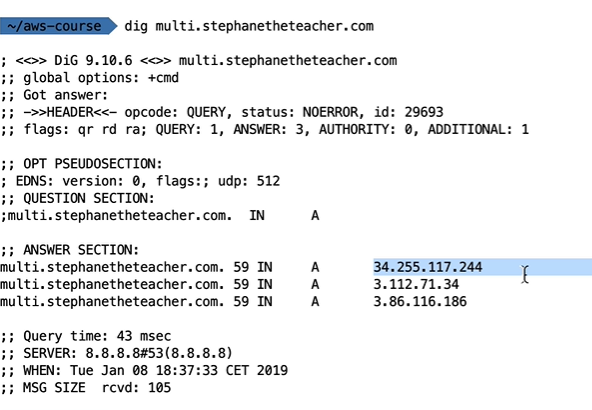






create multiple records with health checks, like below





Multi value works like a load balancer , can have multiple records in multi value by attaching health checks.

If any of the server fails , it will send the traffic to other available servers in the multi value group.