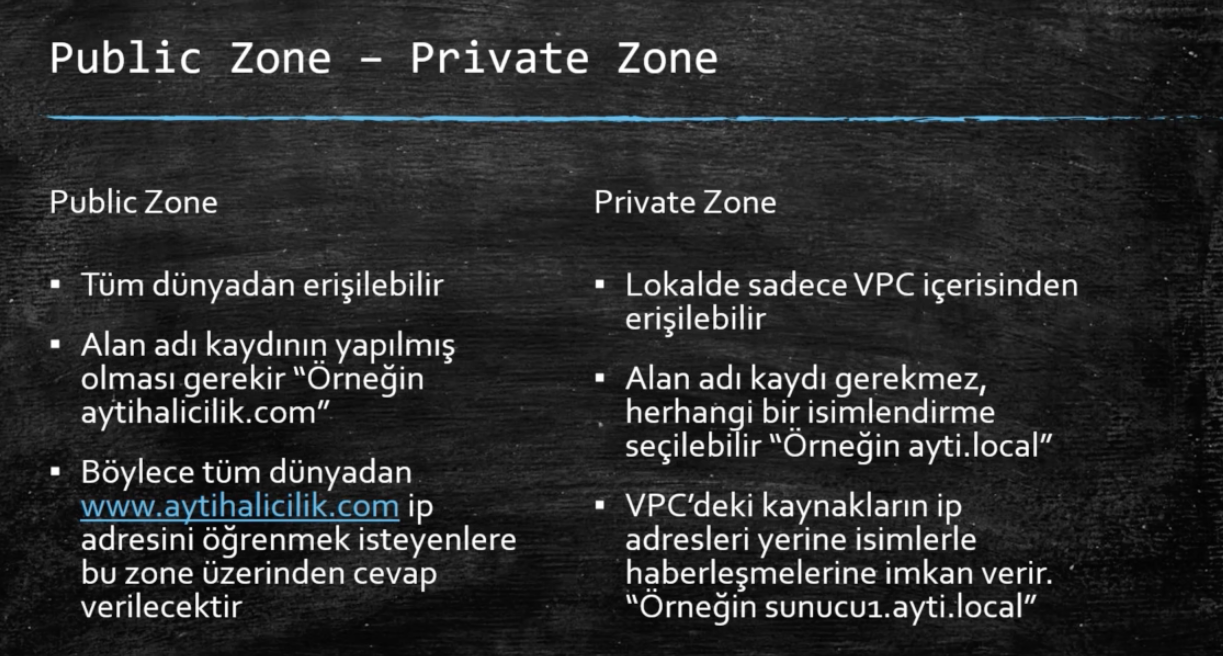
Route 53

Amazon Route 53, yüksek kullaılabilirlik ve ölçeklenebilirlik sunan bir bulut Alan Adı Sitemi (DNS) servisidir. Geliştiricilere ve işletmelere, son kullanıcıların internet uygulamalarına yönlendirilmeleri için son derece güvenilir ve uygun maliyetli bir yol sunmak amacıyla tasarlanmıştır; [www.example.com](http://www.example.com) gibi isimleri 192.0.2.1 gibi bilgisayarların birbirleriyle bağlantı kurmak için kullandığı sayısal IP adreslerine çevirir. Amazon Route 53, IPV4 ve IPv6 ile tamamen uyumludur. Diğer DNS servis sağlayıcılardan ayrışan en önemli yanı ise API desteğine sahip olması. Bu sayede uygulamalarının DNS ihtiyacını ve güncellemeleri otomatize etme imkanı sağlıyor. Trafiğimizi çeşitli tanımlamalar ile dağıtma imkanı sağlıyor. Bir nevi load başancing özelliği var. Coğrafi olarak da dağıtabiliyor. Ayrıca health check yaparak failover senaryoları uygulama imkanı tanıyor.

Yeni alan adları satın almamıza da imkan veriyor.



Bir zone yarattığımız zaman otomatik olarak iki kaydımız kendiliğinden yaratılır

1. tüm DNS alanlarının varsayılan kaydı olan SOA yani start odf authoroty

Bu kayıt bu alanla ilgili yönetimsel kayıtları barındırır. Özellikle bu alan adını başka bir yere transfer edeceksek buradaki bilgiler önemlidir. Ayrıca aynı zonu üzerinden barındıran birden fazla DNS sunucununda aralarında ne sıklıkla bilgi paylaşacağını SOA kaydı belirler.

Her zone için en temel kayıttır. tüm yönetimsel bilgileri içerir. Bu zone hangi DNS sunucuları tarafından tutuluyor, bunun yetkili dns suncuları nedir. bu iki dns sunucu ne şekilde birbirleri ile haberleşmeleri ve ne sıklıkla birbirleri ile haberleşmeleri gerekiyor gibi ayarlar SOA kaydı ile belirlenir. Yani example.com zone unun bütün temel kayıtları bu SOA kaydının altında tutulur.

2. NS name server kaydı: bu alan adını barındırmaya yetkili isim sunucularının listesini içerir. Bu iki kayıt dışında yaratılabilecek bir çok kayıt vardır. Bu zone un üstünde barındırmaya yetkili DNS sunucularının listesini verir. example. lokali buradaki dns sunucuları tarafından tutulur.

a. A Record : En basit DNS kaydıdır. www gibi isimleri 1.2.3.4 gibi IP adreslerini eşleştirdiğimiz DNS kaydırır.

b.CNAME kaydı: Takma isim kaydıdır. xxx gibi isimleri diğer isimlerle eşleştirmeye yarar.

[www.example.com](http://www.example.com) ----🡪 mobile.example.com adresini www.example .com ile eşleştirir.

Böylece IP adresi güncellemek gerekirse her seferinde tüm kayıtları güncellemez tek kayıtla işi çözeriz

c. MX kaydı. Bu alan adı ile ilgili mail hizmeti sağlayan sunucuların IP adreslerini belirtmemize yarar.

d. TXT kaydı: Adından da anlaşılacağı üzere txt değerleri tutmamızı sağlar. Özellikle doğrulama işlerinde bu ratz kayıtlar tutulur.

SRV k:IP adresleri yanında port bilgisi gibi ek bilgiler göndermek istediğimizde kullandığımız servis kaydı

SPF: mail gönderme ile alakalı spam ve spoofin i engellemek için oluşturulan kayıtlarıdr.

Alias: sadece Route 53 e has bir kayıttır. CNAME ile aynıdır hemen hemen. IP adresi ile isim eşleştirirsin. Bununla yaratılan kayıtlar ELB ya da cloudfront gibi servislere bağlama şansı elde edilir. CNAME den een temel farkı bunun tamamen otomatik olarak AWS tarafından güncelleniyor olması ve tamamen ücretsiz olmasıdır.

NOT: Alias kaydını başka bir DNS sunucu yazılımında göremezsiniz. Bu sadece AWS e özeldir. Aslında CNAME in birebir aynısı. Ama CNAME bizim konsolun sol kısmında yaratılan kayıtların bağlanmasını sağlıyorken Alias kaydı ise DNS sunucu üzerinde bir isim yaratıp bunu AWS in kendi kaynaklarının S3, ELB, Elastik Beanstalk ları vs ları baülayabiliyoruz. CNAME i biz yaratıyoruz ve bizim güncellememiz gerekiyor. Alias ı ise bizim güncellemememiz gerekmiyor. Bu bağlananın IP adresi her güncellendiğinde bu kayıt da güncelleniyor ve kesintisiz işlem sağlanıyor. Ama CNAME de bu IP updatelerini kendimiz update etmemiz gerekiyor. Bir diğer farkı ise CNAME ücretli ama Alias ücretsiz.

Public Hosted Zone

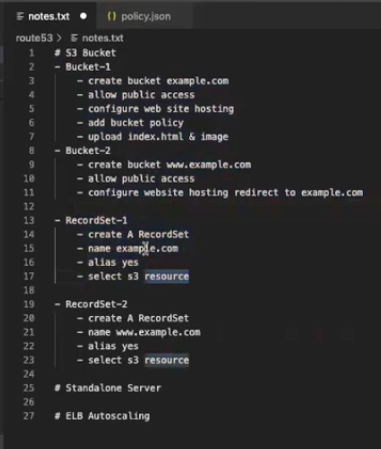
Dış dünyadan da erişilip dış dünyadan sorgu atılıp sorguya cevap verilecek alan. exmple.com ile ilgili tüm işlemler bu public hosted zone un altında yaratılır

Private Hosted Zone

Sadece VPC içerisideki kaynakların sorgulama yaparken ulaşabildikleri ve tamamen VPC içerisindeki kendi özel DNS çözümlememize biz ptivate zone diyoruz.

TTL: DNS ile ilgili en önemli bilinmesi gereken şeylerin başındadır bu TTL . her DNS sunucu her zone içinde barındıran varsayılan hem de her kayıtta ayrı ayrı belirtilen bir TTL time tol ive yaşam süresine sahip. Bunu şöyle düşünelim: [www.google.com](http://www.google.com) u öğrenmek için DNS sunucusuna bir sorgu gönderdim ve DNS bunun cevabını bana gönderdi. Tekrar bu adrese yapmak istediğimde DNS e tekrar sordum tekrar cevap gelir. Bunun binlerce defa yaptığımı düşünürseniz DNS sunucunun ne kadar yüklenildiği anlaşılır. Bunun olmaması için TTL özelliği DNS sunucuya eklenmiştir. Ben DNS sunucuya bir sorgu attığımda bu cevap belirtilen TTL süresince kendi üzerimde barındırır ve o sürece tekrar sorgu attığımda o saklanan süre boyunca saklanan o adres kullanılır. Yani IP adresinin değişmediği statik bir ortamda bunun yüksek olması sorun değil ama eğer IP adresleri sık sık değişiyor ise ya da migration gibi bir işlem yapılıyor ise bu TTL değerlerinin düşük olmasında fayda var.

Chris S3 ve Route 53



İki adet S3 bucket oluştur ve full public permission ver.

serkangumus.me dediğine index.html yükle ve static web hosting’i ayarla. Buna root bucket denir.

diğerini de [www.serkangumus.me](http://www.serkangumus.me) adını ver ve ilkine redirect yap ve bucket policy i de make public yapmayı unutma.

Sonra Route 53 ü aç

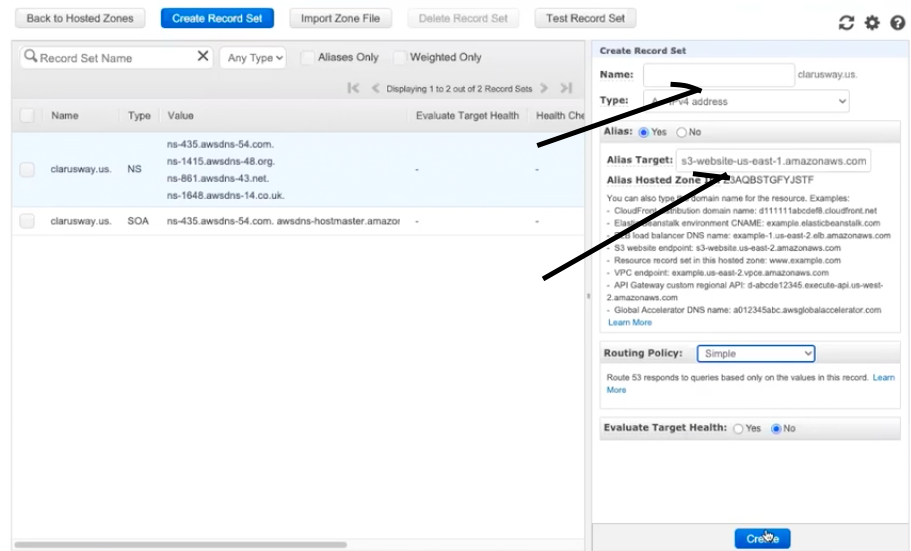
create record set’i tıkla

Name e hiçbirşey yazma ki bu senin ayarını root bucket’ına yönlendirilmesini sağlar.

İlk kısmı boş bırak ki root bucket ın ile aynı isim olsun ‘serkangumus.me’, Type A-IPv4 address yap ve Alias’a tes işaretle

Alias target otomatikman yukarıdaki girdiğin isim ile aynı olan bucket ı gösterecektir ve onu seç.

başka bir ayar yapma create e bas



sonrasında da ikinci bucket i ayarlamak için bu sefer yine create denir.

www koy başına ki [www.serkangumus.me](http://www.serkangumus.me) olur

Type A-IPv4 address seç

bunu seçince otomatikman Alias Hosted zone da aynı adlı S3 bucket çıkacaktır. Onu seç

sonrasınd da create a bas

Uygulama 2 (Chris)

Şimdi de cloudformation ile oluşturulan stack i burada yönlendirelim. Bunun için Chris in verdiği lab4 uzantılı yaml dosyası kullanılarak bir stack ayağa kaldıralım. Burada ikinci sayfada en alttaki kutucuğa işaret koymayı unutma. Ayrıca EC2 full Access ve cloud formation fullaacces olan bir ec2-user-2 rolü oluşturmayı da unutma. Bu rolün ismi yaml file daki isim ile uyşmalı. Bazen bu rolü eklediğimizde kabul etmiyor. Sonra rolü silip devam ettiğimizde kabul ediyor. Belki admin yetkisi olmasından kaynaklanıyordur.

Tüm stack oluştuktan sonra endpointte oluşan IP adresi alınır.

Route 5e e gelinir.

demo.serkangumus.me yazılır

Type:A-IPv4 address

Alias : No seçilir.

Value bölümüne endpoint den alınan IP adresi yapıştırılır.

sonrasında demo1.serkangumus.me adresinden aynı adrese ulaşılabilir.

Uygulama 3 (Chris)

Bu örnek için lab5.yaml file kullanılır. cloudformationda bu stack ayaya kaldırılır. stack içerisinde ec2 instance, ELB ve Autoscaling gruplar yer almaktadır. en sonda output bölümünde karşımıza çıkan adres alınır. Route 53 açılır ve create record set tıklanır

Name: elb.serkangumus.me

Type: A-IPv4 addres seçilir

Alias : Yes

Alias Target: ELB classic load balancer dolu gelir. o seçilir

create denir

Health Check nasıl yaratılır?

Load balancer üzerinde görülenin aynıdır. Bizim AWS kaynağımız gibip health check kaynağına sunucuya bir sorgu atıyor ve cevap alabilirse bu durumda sağlıklı diyor, alamazsa unhealthy olarak kabul ediyor.

Create Health Check:

Name: İsim ver

What a monitör. Endpoint monitorize edilecek

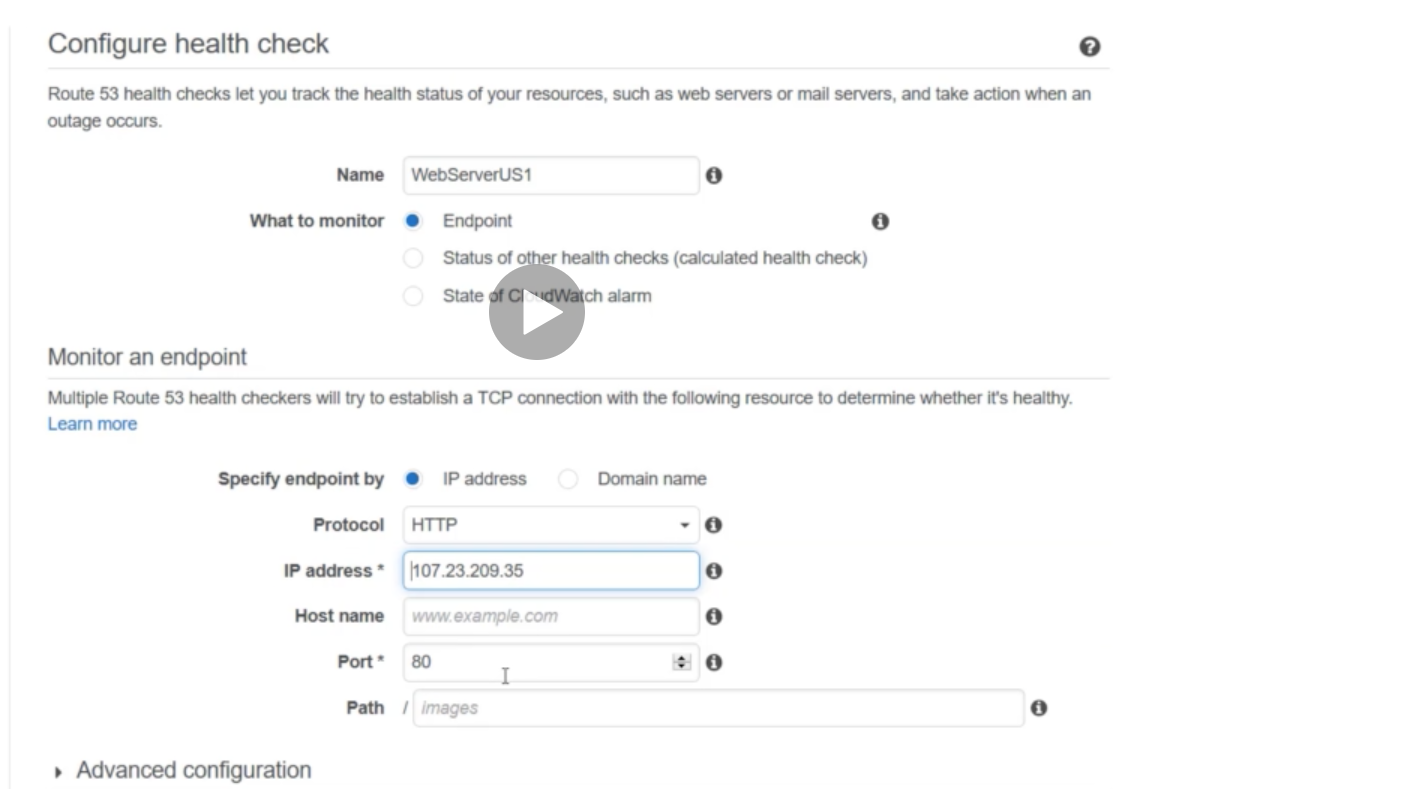
Specify endpoint: IP adres üzerinden özelleştirilebilir ya da domain name de verilebilir burada. Bu durumda health check domain name üzerinden yapılacaktır

Protocol: http üzerinden yapılacak (websitesi kontrolü)

IP adress : Hangi IP adresi kontrol edilecek o belirtilir.

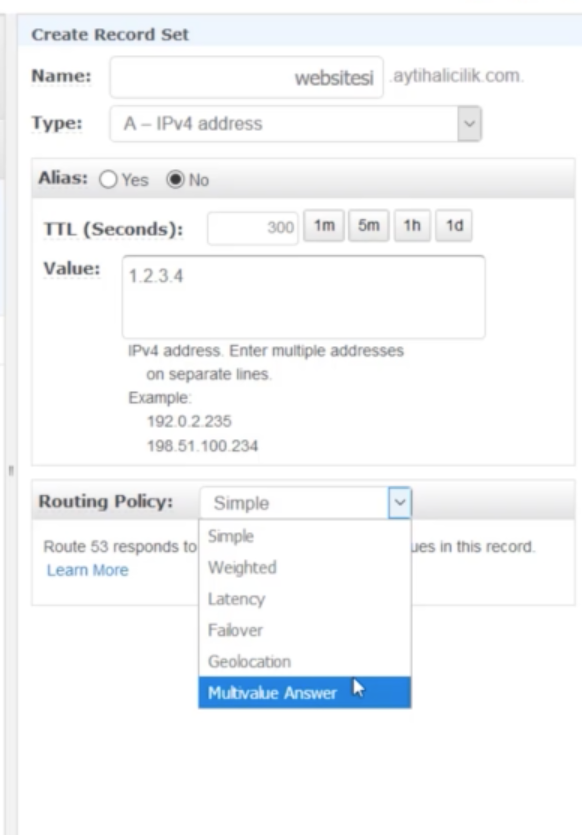
Port :80

Path verilir.

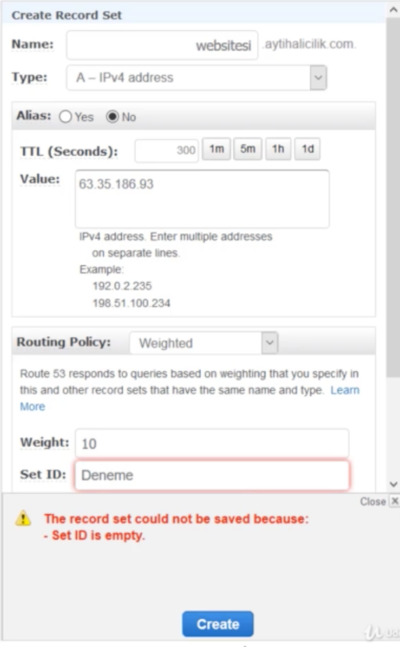


example.com private ile example.com public arasındaki en temel fark, pirvate olan sadece VPC den erişilebilir. public olan ise dünyanın her yerinden erişilebilir. Public olduğunu da NS kaydından anlaşılır.

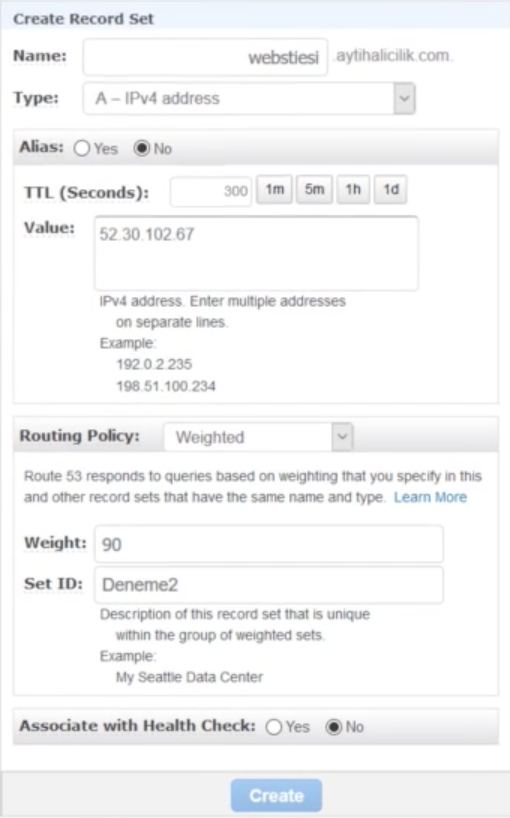
Aşağıdaki resimde de görüleceği üzere, AWS in Route 53 hizmeti basit bir DNS hizmetinden çok fazlasını barındırır. Burada aşağıdaki seçenekler seçilerek load balancing işlemi uygulanabilr.



Weighted: AWS burada belirlenen ağırlağa göre dağıtacak. Heme aynı kayıttan iki sunucu oluşturup ilk sunucu IP adresini girip 10 diyelim,



Sonra bir tane daha oluşturalım.



Bu oluşturulana da 90 girelim. Artık bu DNS sunucuya bir sorgu atıldığında bunların 9’u bir sunucuya, 1 ini diğer sunucuya yönlemdirecek. Sadece DNS kullanılarak bir load balancing oluşturulmuş olur.

O IP adresine ping atılarak hangi IP adresi geri dönülecek bakılabilir. Ve bu IP adresi TTL süresi süresince de aynı kalacaktır.

Latency nasıl kurulur:

Route 53 DNS sorgusu atan kullanıcının hangi lokasyona daha kısa sürede eriştiğine karar verip ona göre uygun sunucuya yönlendirdiği Routing şekline DNS üzerinde latency based routing dniyor.

İki farklı sunucuda aynı kaydı yaratıp neler olduğuna bakalım.

Value’ya bir avrupadan bir de Amerika’dan sunucu girelim.



Birinci kaydı oluştururken yukarıda görüldüğü gibi AVRUPADA BİR SUNUCU ATANIR.



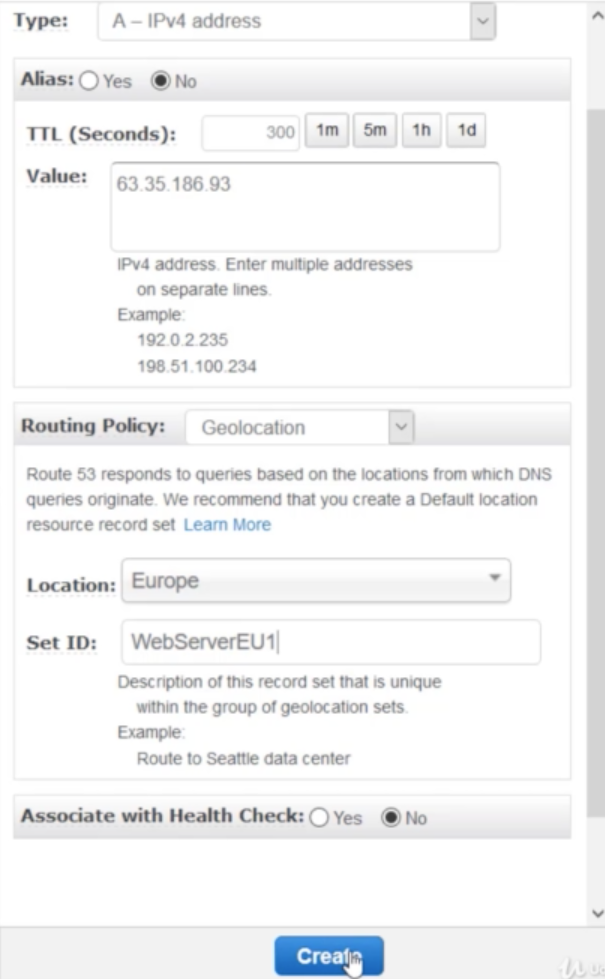
Bu da Amerika açılan kayıt

Şimdi AWS sorgu yapanın lokasyonuna bakacak ve onun yerini latency değerlerine göre değerlendirecek geçmişi kayıtlarından daha sonra en düşük latency e göre bir yönlendirmede bulunacak.

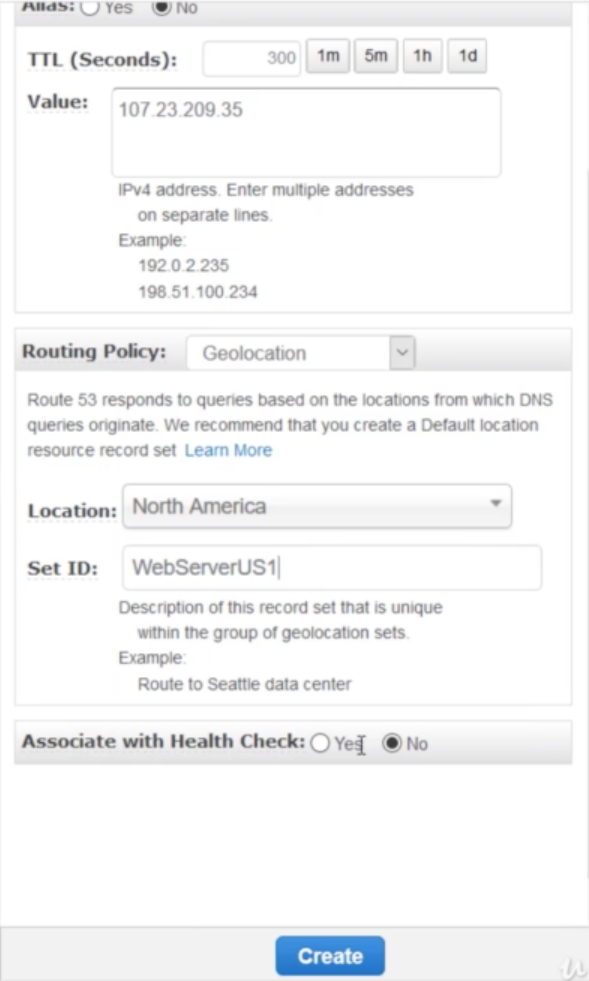
Geolocation nasıl yapılır

Latency e benzer ama ondan biraz farklıdır. latency de latency sürelerine göre bir önceliklendirme yapar. Fakat biz kullanıcının nerde olup nereye gideceğini söylemedik. Sadece latency üzerinden AWS in daha önceden ölçtüğü latency değerleri üzerinden önceliklendirme yaptığını gördük.

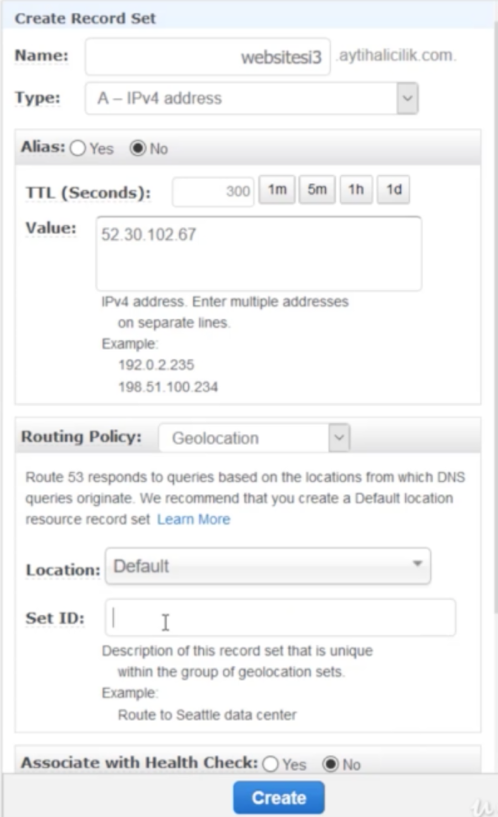
Fakat ben kullanıcının nereye gideceğini belirlersem önceden bunu geolocation ile yaparım. Yani burada erişilecek sunucuyu ben belirlerim.



Dikkat edilirse Eurppe dan gelenleri özellikle bir sunucuya yönlendirmiş oldu bu şekilde. BU sefer bir tane daha oluşturayım ve buna da lokasyon olarak North America seçersem. Bu durumda oradan ulaşanların erişeceği server ı da belirlemiş olurum



Hatta kıta yerine bir kayıt daha oluşturalım ve burada default

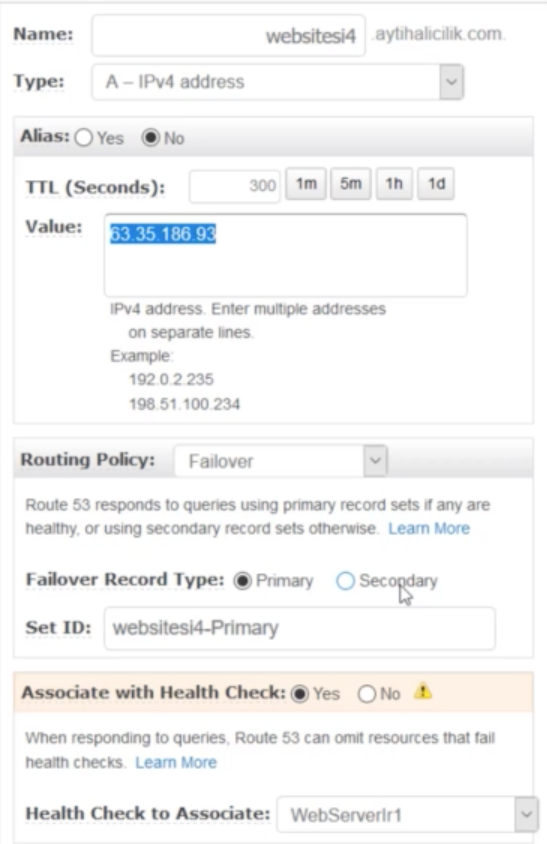


Default un anlamı az önceki belirlediğimiz yerlerden sorgu atılırsa bu durumda yönlendirilen yerlere dön. Geri kalan nereden sorgu atılırsa atılsın bu durumda belirlediğim default IP yi dön.

Failover Routing Policy:

Birden fazla endpoint girerek eğer birine erişilemez ise diğerine yönlendir diyoruz. Burada healthcheck ler işimize yarıyor. Bu kayıtlara bak ve ona göre yönlendir diyebiliyoruz

Kayıt yapılırken primary mi yoksa secondery mı diye soruyor. Ben sorgu attığımda önce healthcheck e bakacak ve eğer sağlıklı ise primary i döndirir fakat sağlıklı değilse demek ki bu sunucuya erişilemiyor diyecek bu durumda secondery i döndürecek.





Trafic flow kısmından tüm bu yapılanlar görselleştirilebiliyor,

