Elastik Yük Dengeleme (ELB)

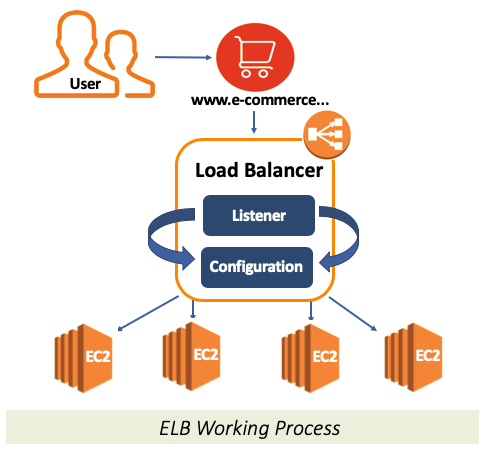
ELB (Elastic Load Balancing), AWS'deki mevcut kaynaklarınız (bulut sunucuları) için yük dağıtımı yapmanıza olanak tanıyan bir hizmettir.

Gelen uygulama trafiğini Amazon EC2 bulut sunucuları, container'lar gibi birden çok hedefe otomatik olarak dağıtır.

1 adet sunucu kurduk ve web sitemizi hizmete açtık. Bunun bir e-ticaret sitesi olduğunu ve 7/24 çalışması gerektiğini varsayalım. Ancak, sanal makinemiz herhangi bir nedenle arızalanırsa, kurulum, ayarlama vb. Nedeniyle tekrar çalışmasını sağlamak muhtemelen 30 dakika sürecektir. Bu nedenle, müşterileri ve parayı kaybedersiniz.

Altyapıyı birden fazla sunucu üzerinden inşa etmeniz gerekiyor. Ve bu sefer bu sunucular arasındaki trafiği yönlendirmeniz gerekiyor. Buna, bu işi sizin için halleden **Load Balancer** adını verdiğimiz şey budur.

### Load Balancer Çalışma Süreci



ELB (Elastic Load Balancing) temel olarak gelen trafiği doğrudan sanal makineler yerine Load Balancer'a yönlendirebileceğimiz bir ağ hizmetidir.

Bu sayede trafiği belirlediğimiz kurallara göre düzenleyebiliyor ve trafiği hedef kuruluma yönlendirebiliyoruz.

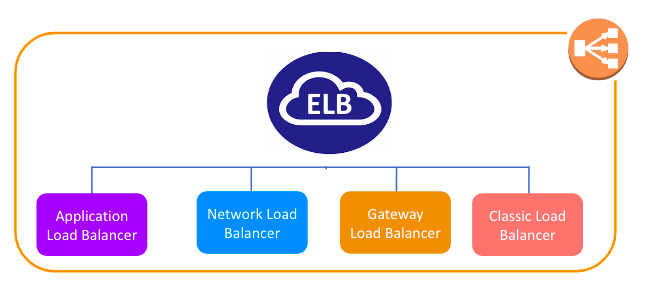
Load Balancer temel olarak 2 bileşenden oluşur.

* **Dinleyici**
* **Konfigürasyon**

**Dinleyici,** adından da anlaşılacağı gibi, bileşen ayarlamalarına göre belirli bir bağlantı noktası üzerinden gelen trafiği dinler.

Yapılandırma, Dinleyicinin analizini dikkate alan ve trafiği yönlendirmeyi sağlayan bir kurallar kümesidir.

### ELB Türleri:



### Application Load Balancer

OSI Modelinin 7. Katmanında çalıştırılır

Gelen trafiği tanımlar ve doğru kaynaklara iletir. **Paketin içeriğini** okuyarak buradaki bilgilere göre isteği yönlendirebilirsiniz.

Yani trafiği paketin içeriğine göre yönlendirir. Örneğin, bir URL /API uzantısına sahipse, uygun uygulama kaynaklarına yönlendirilir.

HTTP, HTTPS tabanlı etkileşimli web sayfaları, mobil uygulamalar ve kapsayıcılar gibi görevlerimiz varsa Application Load Balancer'ı kullanırız. **HTTP ve HTTPS** trafiğinin yük dengelemesi için en uygun olanıdır.

Network Load Balancer gibi, AWS'nin Auto Scaling işleviyle uyumlu Application **Load Balancer'ı** da kullanabilirsiniz.

### Network Load Balancer

OSI modelinin Katman 4'ünde çalıştırılır.

Aktarım katmanında (TCP/SSL) yönlendirme yapar ve saniyede milyonlarca isteği işleyebilir. **Yüksek performans gerektiğinde** **TCP trafiğinin** yükünü dengelemek için en uygun seçenektir.

Bir yük dengeleyici bir bağlantı aldığında, akış karması yönlendirme algoritması kullanarak hedef gruptan bir hedef seçer. Paketiniçeriğine bakmıyorum, temel yönlendirmeyi üzerindeki kurallara göre yapıyor.

Temel bir TCP tabanlı bağlantı noktası üzerinden **basit ve hızlı** bir yük dengeleyiciye ihtiyacımız varsa, bu Network Load Balancer'dır.

Gateway Load Balancer

OSI modelinin Katman 3'ünde (Ağ Katmanı) çalıştırılır.

Gateway Load Balancer'lar, güvenlik duvarları, izinsiz giriş algılama ve önleme sistemleri ve derin paket inceleme sistemleri gibi sanal gereçleri dağıtmanıza, ölçeklendirmenize ve yönetmenize olanak tanır.

Tüm bağlantı noktalarındaki tüm IP paketlerini dinler ve trafiği dinleyici kuralında belirtilen hedef gruba iletir.

### Classic Load Balancer

**OSI** modelinin Katman 4'ünde ve en eski ELB tipinde hizmet vermektedir.

Classic Load Balancer, birden çok Amazon EC2 bulut sunucusunda temel yük dengeleme sağlar ve hem istek düzeyinde hem de bağlantı düzeyinde çalışır.

Classic Load Balancer, EC2-Classic ağı içinde oluşturulan uygulamalar için tasarlanmıştır.

HTTP veya HTTPs trafiğinin yükünü dengelemek ve katman 7'ye özgü özellikleri kullanmak için de kullanılabilir. AWS artık Classic Load Balancer'ı kullanmanızı önermemektedir.

## Otomatik Ölçeklendirme:

## Otomatik ölçeklendirme, talep zaman içinde değiştiğinde mevcut sanal makinelerimin artmasını veya azalmasını otomatikleştiriyor.

## Uygulamanızın yükünü işlemek için doğru sayıda Amazon EC2 bulut sunucusuna sahip olduğunuzdan emin olmanıza yardımcı olur.

## Gerektiğinde yeni bir sanal makine ekleyebilir veya kaldırabiliriz. Böylece, aldığınız isteğin boyutuna göre bu talebe yanıt vermek için sisteminizi genişletebilir ve talep azaldığında otomatik olarak daraltabilirsiniz.

### Otomatik Ölçeklendirmenin Avantajları



Ölçeklendirmeyi Hızlı Bir Şekilde Ayarlayın:

AWS Auto Scaling, tek ve sezgisel bir arabirimde birden çok kaynak için hedef kullanım düzeyleri belirlemenize olanak tanır. Diğer konsollara gitmek zorunda kalmadan ölçeklenebilir tüm kaynaklarınızın ortalama kullanımını hızlı bir şekilde görebilirsiniz.

Yalnızca İhtiyacınız Olan Kadar Ödeyin:

AWS Auto Scaling, AWS hizmetlerini kullanırken kullanımınızı ve maliyet verimliliğinizi optimize etmenize yardımcı olabilir, böylece yalnızca gerçekten ihtiyacınız olan kaynaklar için ödeme yaparsınız. Talep düştüğünde, AWS Auto Scaling fazla kaynak kapasitesini otomatik olarak kaldırarak fazla harcama yapmanızı önler. AWS **Auto Scaling'in kullanımı ücretsizdir ve AWS ortamınızın maliyetlerini optimize etmenize olanak tanır.**

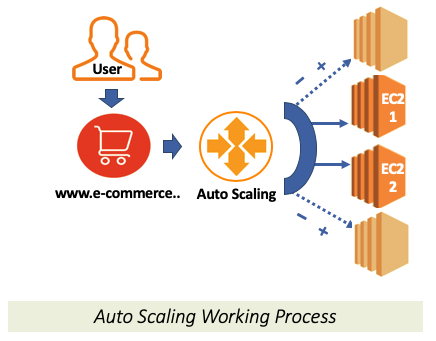
Performansı Otomatik Olarak Koruyun:

AWS Auto Scaling'i kullanarak iş yükleri periyodik, öngörülemeyen veya sürekli değişen durumlarda bile optimum uygulama performansını ve erişilebilirliğini korursunuz. AWS Auto Scaling, istediğiniz performans düzeylerinde çalıştıklarından emin olmak için uygulamalarınızı sürekli olarak izler.

Akıllı ölçeklendirme kararları alın:

AWS Auto Scaling yerine ölçeklendirme yapmaya çalışırsanız her zaman en uygun çözümü bulamayabilirsiniz. AWS Auto Scaling, tüm ölçeklendirme politikalarını otomatik olarak oluşturur ve tercihinize göre sizin için hedefler belirler. AWS Auto Scaling, uygulamanızı izler ve talepler değiştikçe kaynak gruplarınıza gerçek zamanlı olarak otomatik olarak kapasite ekler veya kaynak gruplarınızdan kapasite kaldırır.

### Otomatik Ölçeklendirme Çalışma Süreci



### Stres Aracı'nı Yükleme

* İlk önce Terminal üzerinden ec2 örneğine bağlanın
* Kök ayrıcalıkları elde edin

~ sudo su

* Ardından stres uygulamasını aşağıdaki komutlarla yükleyip çalıştırın ve CPU kullanımını artırın.

~ yum yükleme https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm

~ yum yükleme stresi

~ stres -- cpu 80 --timeout 2000

(veya)

~ sudo amazon-linux-extras yüklemek epel -y

~ sudo yum install -y stres

~ stres -- cpu 80 --timeout 20000

* Stres aracı, bulut sunucusunun CPU kullanımını artırmaya başlayacaktır.
* Gerekirse çalışan diğer örnekler için de aynı işlemi yapın. CPU kullanımı kademeli olarak artacaktır.
* Sonuçları görmek için bir süre bekleyin.