



TÜBİTAK

BULUT KULLANIMI

ARAŞTIRMA SERİSİ - SAYI 7



BİLGEM

YAZILIM TEKNOLOJİLERİ ARAŞTIRMA ENSTİTÜSÜ

Simge ve Kısaltmalar

Kısaltmalar	Açıklama
TÜBİTAK	Türkiye Bilimsel ve Teknolojik Araştırma Kurumu
BİLGEM	Bilişim ve Bilgi Güvenliği İleri Teknolojiler Araştırma Merkezi
YTE	Yazılım Teknolojileri Araştırma Enstitüsü
RAM	Random Access Memory (Rastgele Erişim Belleği)
AWS	Amazon Web Services

Yazar

Ahmet SAY

Yayın Koordinatörü

Elif ŞENYİĞİT

Editörler

Serkan KOÇ
Sevinç KARAKAŞ
Tuğçe YILMAZ

Tasarım

Şeyma KOÇER

©2023 - Tüm hakları saklıdır.

İletişim: 0(312) 289 92 22 - yte.bilgi@tubitak.gov.tr

yte.bilgem.tubitak.gov.tr

Yayınlanan yazıların sorumluluğu yazarına aittir, TÜBİTAK BİLGEM sorumlu tutulamaz.

İçindekiler

Önsöz	4
Giriş	5
Bulut Sağlayıcı Nedir?	6
Bulut Kullanılmayan Veri Merkezlerinde Oluşan Problemler	6
1. İletişim ve İnsan Faktörü	6
2. Güvenlik ve Yetkilendirme	7
3. Veri Kaybı ve Felaket Kurtarma	7
4. Performans ve Kapasite	8
5. Maliyet	8
6. Sunulan Hizmetler	9
7. Sunucu Bakımı	9
Bulut Sağlayıcıya Geçiş Yöntemi ve Aşamaları	10
1. Bulut Sağlayıcı Seçilmeli	10
2. Ekibin Seçilen Bulut Sağlayıcı Üzerinde Tecrübe Kazanması Sağlanmalı	10
3. Veri Merkezinden Bulut Sağlayıcıya Aktarım Yapılmalı	11
4. Kaynaklar Bulut Üzerinde Oluşturulmalı	11
5. Kaynaklar Optimize Edilmeli	12
Bulut Kullanımının Değerlendirilmesi	12
1. İletişim ve İnsan Faktörü	12
2. Güvenlik ve Yetkilendirme	13
3. Veri Kaybı ve Felaket Kurtarma	13
4. Performans ve Kapasite	13
5. Maliyet	14
6. Sunulan Hizmetler	15
7. Sunucu Bakımı	16
Sonuç ve Öneriler	17
Kaynakça	18

Önsöz

TÜBİTAK BİLGEM Yazılım Teknolojileri Araştırma Enstitüsü, 2012 yılından bu yana yazılım teknolojilerinde AR-GE faaliyetleri yürüten bir araştırma kuruluşudur. Araştırma faaliyetlerinde elde ettiği birikimini stratejik, hassas ve kritik projeler yürüterek kamu adına hayata geçirmekte; kurumlarımıza dijital dönüşüm, yazılım geliştirme teknolojileri ve kalite süreçleri konusunda danışmanlık vermektedir.

Araştırma Serisi ile TÜBİTAK BİLGEM YTE kurum içi çalışmaların yaygınlaştırılması ve sektörün erişimine açılması amaçlanmaktadır. Araştırma Serisi'nde yayınlanan çalışmalar TÜBİTAK BİLGEM YTE çalışanlarının projelerde elde ettiği bilgi birikimini paylaşmak adına derlenmiştir. Bu çalışmalar ile ülkemizin yazılım sektörüne katkı sağlanması hedeflenmektedir.

Giriş

Yazılım teknolojilerinin öneminin ve bu teknolojilerle geliştirilen uygulamaların kullanımının gün geçtikçe artması, bu uygulamalar için gereken bilişim altyapısı ihtiyaçlarını da artırmıştır. Bu ihtiyaçlar geliştirilen uygulamanın türüne ve kullanımına bağlı olarak maliyet, kapasite, güvenlik gibi farklı alanlarda karşımıza çıkmaktadır. Yazılım kuruluşları, bilişim altyapısını kendileri yönetmekteyken, bu ihtiyaçları giderememe problemi ile karşı karşıya gelerek alternatif yollar aramaya başlamışlardır. Sonuç olarak bilişim altyapısını hazır sunan bulut bilişim sistemleri oluşturulmuştur. Günümüzde altyapısı tamamen bulut üzerine kurulan, altyapısının bir kısmı kendi veri merkezinde, diğer bir kısmı bulut üzerinde olan ve altyapısını tamamen kendi veri merkezinden yöneten kuruluşlar mevcuttur. Bu çalışmada öncelikle bulut kullanmayan kuruluşların karşılaştığı problemler kategorilere ayrılarak ve örneklendirilerek incelenmiştir. Sonrasında bu problemleri bir bulut sağlayıcı kullanarak çözmek için uygulanması gereken adımlar belirtilmiştir. Son olarak, belirtilen problemlerin nasıl giderilmiş olacağı ve karşılaşılabilecek problemler anlatılmıştır.



Bulut Sağlayıcı Nedir?

Bulut sağlayıcılar yazılım kuruluşlarının veri merkezlerini kendilerinin yönetmelerine alternatif olarak kurulmuş sistemlerdir. Kuruluş, ihtiyacı olan bilişim altyapısını bir bulut sağlayıcı kullanarak giderir. Bulut sağlayıcı kullanmanın ana motivasyonu maliyetleri ve iş yükünü azaltmaktır.

Bulut Kullanılmayan Veri Merkezlerinde Oluşan Problemler

Yazılım geliştirme projesi yürüten kuruluşların bir veri merkezine ihtiyacı vardır. Bu veri merkezinde geliştiricilerin yazdığı kodların ve proje ile ilgili dokümanların yer aldığı, sürekli entegrasyon araçlarının çalıştırıldığı, statik kod analizinin yapıldığı, uygulamanın arayüz üzerinden test edildiği, kod gözden geçirmesinin yapıldığı sunucular yer almaktadır. Bu kapsamda geliştiricilerin yeni bir sunucuya ihtiyacı olabilmekte ve veri merkezini yöneten bilişim sistemleri ekibinin bu sunucu ihtiyacını karşılaması gerekmektedir. Söz konusu projenin canlı ortamda da yer aldığı düşünülürse uygulamanın çalışması gereken sunucuların da veri merkezinde olması gerekmektedir. Bu veri merkezinin kuruluşun kendi çatısı altında bulunması ilk başta ne kadar mantıklı gözükse de zamanla çeşitli problemleri barındırdığı ortaya çıkmıştır. Bu problemler 7 ana başlıkta incelenebilir.

1. İletişim ve İnsan Faktörü

Geliştiriciler ve bilişim sistemleri ayrı ekipler olarak çalıştığı için arada bir iletişim kanalı olmak zorundadır. Örnek olarak, geliştiricinin yeni bir sunucuya ihtiyacı olduğu durumda, geliştirici bu isteği bilişim sistemlerine iletir ve istek sıraya alınır. Ancak geliştirici, sunucunun kendisine ne zaman iletileceğini bilmemektedir. Bu işlem saatler ve ya günler sürebilir. Talep oluşturan ve talebi karşılayanlar farklı kişiler olduğu için taleplerin eksiksiz yerine getirilmesi konusunda da sıkıntılar yaşanabilmektedir. Örneğin geliştirici, projede kullanılan kod deposu için okuma yetkisi olan bir kullanıcı talep etmektedir. Ancak kendisine verilen kullanıcıda okuma ve yazma yetkisi vardır. Bunu düzeltmek için tekrar iletişime geçmek ve tekrar beklemek gerekir.

Bunun yanı sıra bilişim sistemleri ekibi eğer yoğunsa geliştiricilerin taleplerini arka plana da alabilir. Böyle durumlarda geliştirici, talebinin tamamlanması için bir başkasını beklediğinde, diğer bir deyişle insan faktörü devreye girdiğinde zaman kaybı yaşar, işi sekteye uğrar ve motivasyonu düşer.



Şekil 1. Bulut Sağlayıcı Kullanmayan Veri Merkezinde Yaşanan Sunucu Teslim Süreci

Tüm bunlara ek olarak, eğer yazılım geliştiricinin kuruluşu ile bilişim sistemlerinin kuruluşu farklı ise iletişim daha da zorlaşır ve bu süreç daha da sıkıntılı hale gelir. Çünkü bu senaryoda geliştiricinin istekleri, bilişim sistemleri için daha düşük öncelikli hale gelmektedir.

2. Güvenlik ve Yetkilendirme

Geliştiricilerin, kendilerine verilen sunucularda işlerini gerçekleştirebilmesi için, sunucuların bazı işlemlere yetkilerinin olması gerekmektedir. İnternet erişimi en çok ihtiyaç duyulan yetkilerden birisidir. Örneğin projede kullanılacak bir sunucuya kurulum yapılması gerektiğinde, bilişim sistemleri sunucuya internet erişimi vermezse kurulumlar gerçekleştirilemez. Bazı veri merkezlerinin güvenlik politikaları, internet erişimi vermeyecek kadar katı olabilir. Böyle durumlarda kurulum yapmak çok zor hatta imkânsız hale gelebilir.

Yetkilendirme konusunda bir diğer problem, güvenlik ayarlarının yanlış yapılandırılması olarak karşımıza çıkmaktadır. Bir sunucuya veya kullanıcıya verilen yetkiler doğru olmayabilir. Bir güvenlik kuralı başka bir kural ile çakışabilir. Bu durum da geliştirme sürecini olumsuz etkilemektedir.

3. Veri Kaybı ve Felaket Kurtarma

Herhangi bir veri kaybı yaşanmaması için projeye ait sunucuların düzenli olarak yedeklenmesi gerekmektedir. Bu işlem bilişim sistemleri tarafından yapılmaktadır. Ancak olası bir güç kesintisi tüm veri merkezini savunmasız bırakmaktadır. Tüm sürecin tek bir veri merkezi tarafından yönetildiği durumda felaket kurtarma sağlanamamaktadır.

4. Performans ve Kapasite

Uygulamanın kullandığı sunucuların hangi özelliklere sahip olacağını yazılım geliştirme ekibi belirler. Ancak belirlenen her özellik bilişim sistemleri tarafından onaylanmayabilir. Bazı durumlarda istenen kaynakların çok fazla olduğu, daha azıyla yetinilmesi gerektiği söylenmektedir. Sonuç olarak uygulama ihtiyacı olan performansa ulaşmayabilir.

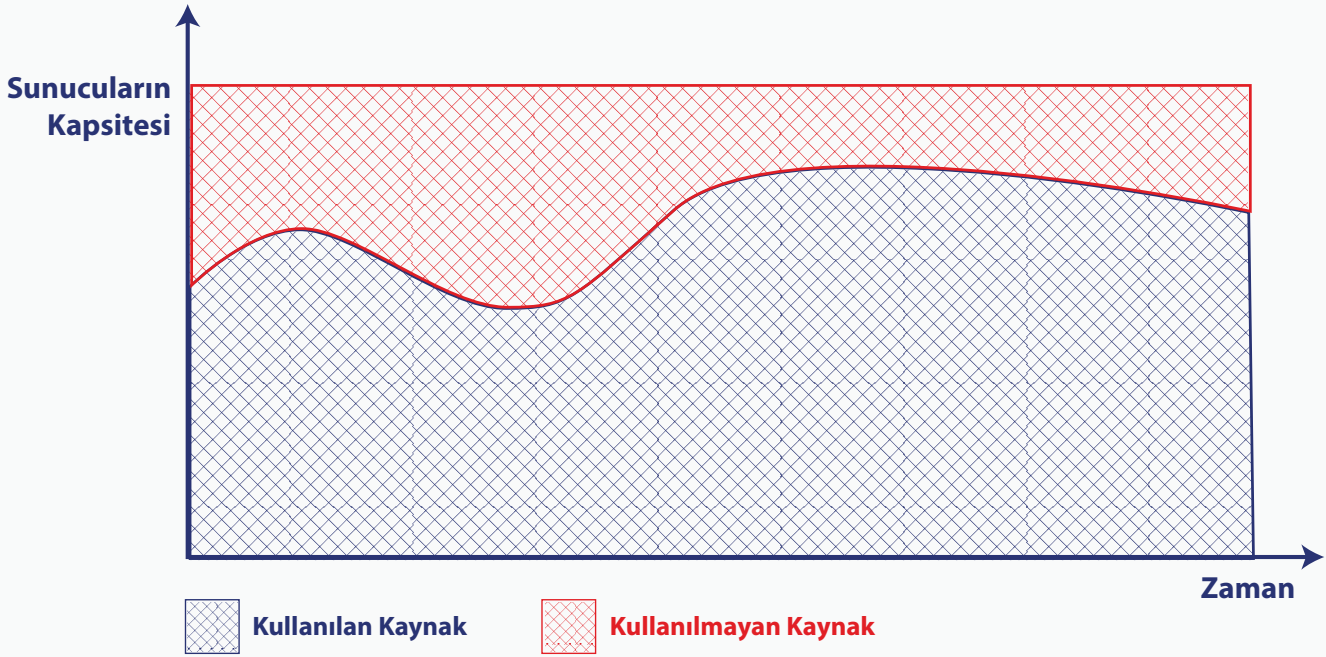
Eğer olaya bilişim sistemlerinin gözünden bakmak gerekirse, veri merkezi her ne kadar geliştiricinin tüm ihtiyaçlarını karşılamak için planlanmış olsa da veri merkezindeki donanımın bir sınırı vardır. Geliştiriciler denemek istediği her senaryoyu veri merkezini kullanarak deneyemez. Örneğin geliştiriciler canlı ortamda oluşabilecek yoğun bir ağ trafiğini denemek için yüksek donanımlara sahip 100 adet sunucu talep ederse, böyle bir istek veri merkezinin kapasitesini aşabilir. Aşmasa bile böyle bir istek çok büyük bir iş gerektirecektir.

5. Maliyet

Yazılım projeleri tarafından kullanılan sunucular çoğu zaman sanal sunucudur. Bu sanal sunucular, veri merkezinde bulunan gerçek sunucular üzerinde kurulu olarak çalışmaktadır. Eğer projenin daha çok sunucuya ihtiyacı varsa ve veri merkezinin kapasitesi yetmezse, kuruluş bilişim sistemlerinin kullanabilmesi için daha çok donanım satın almak zorundadır. Ancak satın alınan donanım zamanla eskimekte ve bir süre sonra kullanılamaz hale gelmektedir. Bu durumda kuruluşun var olan donanımlarını da periyodik olarak yenilemesi gerekmektedir. Bu süreç bakım masrafı olarak ele alınabilir.

Bunun yanı sıra alınan donanıma ise her durumda ihtiyaç olmamaktadır. Canlı ortamda çalışan bir uygulama düşünüldüğünde, uygulama günün her saatinde aynı yoğunlukta çalışmamaktadır. Örneğin, mesai saatleri arasındaki yoğun trafiği kaldırabilmek adına 10 tane sunucunun çalışıyor olması gerekirken, gece yarısında 1 tane sunucu yeterli olmaktadır. Ancak geri kalan 9 sunucuyu kapatma imkânı olmadığı için bu sunucular boşuna çalışmakta ve boşuna kaynak harcamaktadır. Yazılım ekibi ve bilişim sistemleri arasında bir entegrasyon sağlanması, geliştirme ve bilişim sistemleri ekipleri birbirinden bağımsız olduğu için bu işlemin yönetimi oldukça zor olacaktır. Bu sunucular ihtiyaç duyulmadığı zaman kapatılsa bile, gerekli donanımı satın almak için harcanan para geri döndürülemez. Kuruluş donanımları kendisi üzerine sahiplendiği için, bu donanımların hangi yoğunlukta kullanılacağı fark etmeksizin, 7/24 kullanılacakmış gibi para ödenerek satın alınması gerekir. İhtiyaç olan donanımın az olması ise birim fiyatın daha pahalı olmasına neden olacaktır. Çünkü kuruluş, donanım sağlayıcıların büyük satın alımlarda yapacağı indirimlerden yararlanmamış olur.

Aynı problem geliştirme sürecinde de gerçekleşebilir. Örneğin sürekli entegrasyon adımlarını çalıştırmakla sorumlu sunucuların her biri tek seferde bir adım çalıştırabilir. Sunucuların hepsi meşgul olduğunda bir kuyruk oluşur. Bu kuyruğu azaltmak için yeni sunucu talep edilebilir, ancak adımların çalışmadığı durumda bu sunucular yine boşuna kaynak harcıyor olacaktır.



Şekil 2. Bulut Sağlayıcı Kullanmayan Veri Merkezindeki Sunucuların Kaynak Kullanımı

Yukarıda belirtilen sebeplerden ötürü kuruluşun hem ne kadar donanıma ihtiyaç duyduğunu hesaplaması, hem de sahip olduğu veri merkezinin bakım masraflarını karşılaması gerekmektedir.

6. Sunulan Hizmetler

Bir kuruluşteki bilişim sistemlerinin geliştiricilere sunduğu hizmet genellikle sanal sunuculardır. Buna hizmet olarak altyapı adı verilir. Geliştiricinin ihtiyacı ne kadar basit olursa olsun, ihtiyacının karşılanması için bir sunucu gerekir. Buradaki problem, basit bir işlem için bile bir sunucu ihtiyacı olduğudur. Sunucunun bilişim sistemleri tarafından ayarlanması ve geliştirici tarafından da uygun araçların yüklenmesi gerekir. Sunucuları doğrudan geliştiricilerin yönetmediği ve kodların geliştiriciler için daha ön planda olduğu hizmet olarak platform (Platform as a Service), veya sunucularla hiçbir etkileşimin olmadığı, geliştiricinin sadece kodunu yazıp ek bir çaba harcamadan çalıştırabildiği hizmet olarak fonksiyon (Function as a Service) gibi hizmetler geleneksel veri merkezlerinde bulunmamaktadır.

7. Sunucu Bakımı

Veri merkezindeki donanımların bakımının yapılması gerektiği gibi, projede kullanılan sunucuların da kendi içerisinde bakımlarının yapılması gerekir. İşletim sisteminin güncellenmesi en yaygın bakım ihtiyaçlarından birisidir. Bu ihtiyacı geliştiriciler veya bilişim sistemleri üstlenebilir. Ancak asıl problem bu işlem için zaman harcanmasıdır. Çok fazla sunucu olduğu durumda ekiplerin bu tarz güncellemeleri yapması daha da zor hale gelmektedir.

Bulut Sağlayıcıya Geçiş Yöntemi ve Aşamaları

Kuruluşların bulut sağlayıcı kullanmadığı durumda karşılaştığı problemlerin bir bulut sağlayıcı kullanılarak nasıl çözüleceği ve hangi adımların izlenmesi gerektiği aşağıda sıralanmıştır.

1. Bulut Sağlayıcı Seçilmeli

Proje ekibinin ihtiyaçlarını karşılayacak bir bulut sağlayıcı seçilmelidir. Bu aşamada ilk önce verilerin hangi coğrafi bölgede/bölgelerde tutulacağını belirlemek, bu bölgelerde veri merkezleri bulunan bulut sağlayıcıların seçilmesi gerekmektedir. Seçilen bölge ile uygulamanın son kullanıcılarının arasındaki gecikme göz önünde bulundurulmalıdır. Büyük bulut sağlayıcıların hepsi dünyanın birçok bölgesinde hizmet vermektedir. Bu yüzden gecikme büyük bir problem oluşturmamaktadır.

İncelenen bulut sağlayıcının, projenin kullandığı araçları kullanabilecek hizmetlerinin olduğu kontrol edilmelidir. Eğer bir bulut sağlayıcı tüm ihtiyaçları karşılamıyorsa birden fazla bulut sağlayıcı da kullanılabilir. Ancak bu çok uç bir senaryodur. Sektörde önde gelen bir bulut sağlayıcı çoğu durumda tüm ihtiyaçları karşılamaktadır.

İhtiyaçların yanı sıra bu ihtiyaçları karşılayan hizmetlerin ücretleri de kontrol edilmelidir. Bu ücretler diğer bulut sağlayıcılarla karşılaştırılmalı ve uygunluğu onaylanmalıdır.

2. Ekibin Seçilen Bulut Sağlayıcı Üzerinde Tecrübe Kazanması Sağlanmalı

Seçilen bulut sağlayıcının doğru şekilde kullanılabilmesi için ekibin bilgi birikiminin olması gerekmektedir. Kaynak olarak resmi dokümantasyonlardan ve internetteki diğer bağımsız kaynaklardan yararlanılabilir. Her bir ekip üyesinin buraya çaba harcaması ve eşit bilgi birikimine ulaşması mümkün olmayabilir, ancak birden fazla kişinin öğrenmesi projenin devamlılığı açısından faydalı olacaktır.

Öğrenme aşamasında güvenlik faktörüne özellikle dikkat edilmelidir. Çünkü ekip artık projenin güvenliğinden de sorumludur. Örneğin sunucuların internet erişimi her zaman olmalı mı, uygulama içinde kullanılan hassas bilgiler nerede depolanacak gibi konulara ekstra özen gösterilmeli ve detaylı şekilde araştırılmalıdır. Diğer konularda olduğu gibi güvenlik konusunda da iyi pratikler ve uygulama yöntemleri resmi dokümantasyonlarda ve bağımsız kaynaklarda mevcuttur.

3. Veri Merkezinden Bulut Sağlayıcıya Aktarım Yapılmalı

Bulut sağlayıcı kullanılmaya başlanmadan önce, kuruluşun veri merkezinde proje ile ilgili gerekli verilerinin aktarılması gerekmektedir. Projenin kod deposu, veri tabanı, dosyalar bunlara örnek gösterilebilir. Aktarım şekli verinin büyüklüğüne ve internet hızına göre değişiklik gösterebilir. Eğer veri büyüklüğü veya internet hızı nedeniyle yükleme işlemi aylar süreceyse, internet üzerinden yükleme yapmak yerine fiziksel cihazlar talep edilebilir. Bu yöntem ile bulut sağlayıcı tarafından veri merkezine fiziksel cihazlar getirilecek, kablo ile hızlı bir şekilde bu cihazlara yükleme yapılacak ve cihazlar bulut sağlayıcıya geri gönderilecektir.

Veri merkezindeki her bir verinin aktarılması şart değildir. Eğer güvenlik sebebiyle bazı verilerin kuruluştaki kalması daha uygunsa bu veriler kuruluştaki kalabilir. Hem bulut sağlayıcının, hem de şirket içi sunucuların kullanıldığı yapıya hibrit bulut adı verilmektedir.

Bunun yanı sıra verilerin şirket içi sunucular üzerinden erişilmesi gerekiyorsa, ancak verinin büyüklüğü sebebiyle depolamanın bulut üstünde yapılması planlanıyorsa aradaki eşleşme bir ağ geçidi ile sağlanabilir.



Şekil 3. Ağ Geçidi ile Sağlanan Hibrit Depolama Yöntemi

4. Kaynaklar Bulut Üzerinde Oluşturulmalı

Veri aktarımı tamamlandıktan sonra projenin ihtiyaç duyduğu kaynaklar bulut sağlayıcı üzerinde oluşturulmalıdır. Bu aşamada eski mimari ile yenisinin birebir aynı olmasına gerek yoktur. Örneğin uygulama Kubernetes üzerinde çalışıyorsa sunucuların manuel olarak tek tek kurulmasına gerek yoktur. Bulut sağlayıcının sağladığı Kubernetes hizmeti ile çok daha kolay ve hızlı bir kurulum gerçekleştirilebilir veya ihtiyaç duyulan işlem basit bir kod parçasından oluşuyor ise, bunun için sıfırdan yeni bir sunucu oluşturmak yerine hizmet olarak fonksiyon olan hizmetler kullanılabilir.

5. Kaynaklar Optimize Edilmeli

Kaynaklar oluşturulduktan ve uygulamanın bulut üzerinde sorunsuz çalıştığı doğrulandıktan sonra optimizasyon için neler yapılabileceği incelenmelidir. Bulut sağlayıcılar üzerinde kullanılan kaynakların çoğu belirli bir ücret talep etmektedir. Uygulamanın kaynakları verimli bir şekilde kullanması, kuruluşun ödeyeceği faturanın da azalmasını sağlayacaktır.

Örnek vermek gerekirse, uygulama yoğun istek almadığı sürece fazladan sunucu çalışmasının bir anlamı yoktur. Bu sunucular silinebilir. Daha sonra uygulamanın aldığı istek arttığında sunucu sayısı da artırılabilir. Bu tarz işlemler kolaylıkla otomatize edilebilmektedir.

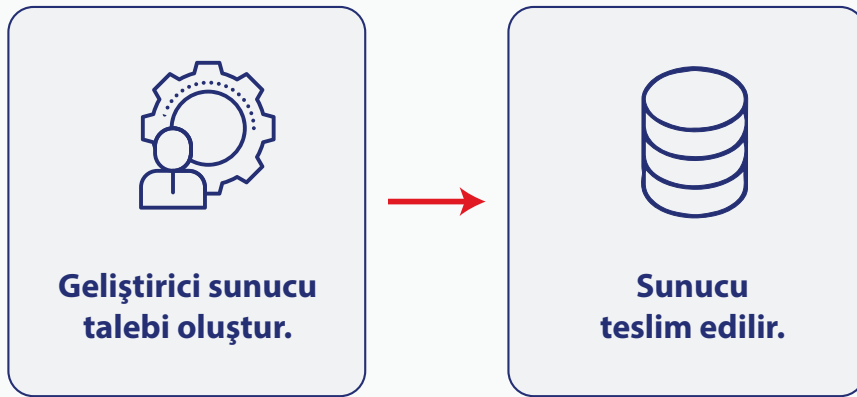
Bunun yanı sıra sunucusuz hizmetler genelde daha ucuz olmaktadır. Uygulamanın kendi işleyişine uygun olan sunucusuz hizmetleri kullanması ücretlendirmenin de düşmesini sağlayacaktır.

Bulut Kullanımının Değerlendirilmesi

Buluta geçiş tamamlandıktan sonra, problem tanımında belirtilen problemlerin nasıl çözüldüğü ve karşılaşılabilecek problemler aşağıda belirtilmiştir.

1. İletişim ve İnsan Faktörü

Uygulamanın altyapısını artık proje ekibi yönettiği için sunucular ve geliştiriciler arasında bir insan faktörü olmayacaktır. Ekip üyeleri başka hiç kimseye bağlı olmadan istediği işi gerçekleştirebilir. Geliştiricinin isteği, sırada bekletilmeden doğrudan başlatılır. Bekleme süresi ise saatler veya günler değil, dakikalardır. Geliştirici istediği özellikleri doğru tanımladığı sürece oluşturulan kaynakta bir eksiklik veya yanlışlık olmamaktadır.



Şekil 4. Bulut Sağlayıcı Kullanan Veri Merkezinde Yaşanan Sunucu Teslim Süreci

2. Güvenlik ve Yetkilendirme

Proje ekibi kendi güvenlik politikasını belirleyeceği için yetkilendirme konusu bir problem olmaktan çıkmaktadır. Bir sunucuya kurulum yapmak için internet erişimi gerekiyorsa, o erişim anında verilebilir. Güvenlik ayarları yanlış yapılandırılmış ise ekip doğrudan müdahale edip düzeltme yapabilir. Hatta bazı bulut sağlayıcılar bu tarz ayarlarda hangi adımın yanlış olduğunu göstermekte, geliştiricilere doğrudan fayda sağlayabilmektedir. Ayrıca proje ekibi, ek bir ayar yapmadan kullanılan servislerin kendi varsayılan güvenlik özelliklerinden de faydalanabilmektedir.

Güvenlik konusunda dikkat edilmesi gereken en önemli konu ekibin bulut sağlayıcının güvenlik sisteminin nasıl çalıştığına hakim olmasıdır. Bulut sağlayıcıların kendi güvenlik önlemleri mevcut olsa da geliştiricilerden kaynaklı bir güvenlik açığı tüm uygulamayı kötü anlamda etkiler. Yanlış ayarlanmış bir güvenlik duvarı, kişilere gereğinden fazla yetki verilmesi, bir kaynağın dış dünyaya açık bırakılması gibi konularda bulut sağlayıcılar üzerine sorumluluk almaz. Tüm sorumluluk geliştirme ekibine aittir. Dikkat edilmediği durumda uygulamadan veri sızdırılması muhtemeldir.

3. Veri Kaybı ve Felaket Kurtarma

Hassas verilerin yedeklenmesi geliştirme ekibi tarafından yapılmaktadır. Bu tarz işlemleri ayarlamak başta geliştiricinin işini zorlaştırıyor gibi görünse de, yapılması birkaç dakika sürer. Bir felaket senaryosu durumunda ise sorumluluk bulut sağlayıcı üzerindedir. Bulut sağlayıcılarda bir bölgede birbirinden uzak birden fazla veri merkezi bulunmaktadır. Eğer bir veri merkezinde güç kesintisi yaşanırsa işlemler diğer bir veri merkezinden devam ettirilir. Planlı kesintileri bulut sağlayıcılar önceden haber vermekte ve geliştiricilerin yapması gereken bir işlem varsa bildirmektedir.

Bir bölgedeki bütün veri merkezlerinin kesintiye uğraması, yani bölgesel bir kesinti meydana gelmesi durumuna karşı önlem almak için, geliştiricilerin bölge seviyesinde yedekleme yapması ve kaynaklarını birden çok bölge üzerinde yönetmesi gerekir. Bölgesel seviyede bir yedekleme ise ek maliyetlere sebep olmaktadır. Ancak bölge çapında bir kesintiye oldukça nadir rastlanmaktadır. Kesintinin olduğu bölgenin kullanılan bölge olması ise daha da düşük bir ihtimaldir. Yine de çok kritik verilerin birden fazla bölgede depolanması faydalı olabilir.

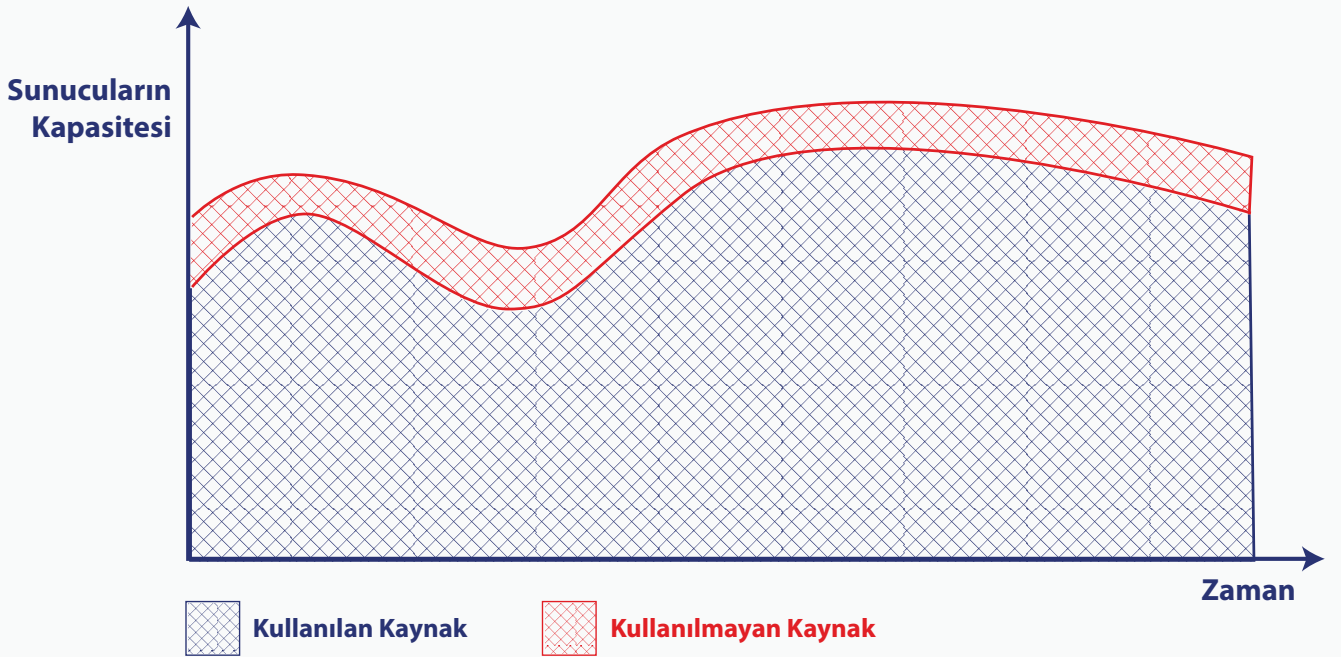
4. Performans ve Kapasite

Kuruluşun kendi veri merkezinin bir kapasitesi olduğu gibi, bulut sağlayıcıların da bir kapasitesi vardır. Ancak bu kapasite tek bir veri merkezine kıyasla çok daha yüksektir. Eğer gerçekten ihtiyaç varsa 12 terabayt RAM bulunduran bir sunucu oluşturup kullanılabilir. Saniyede milyonlarca istek karşılayabilen bir yük dengeleyici oluşturulabilir. Geliştiriciler bir simülasyon gerçekleştirmek

istiyorlarsa gerekli kaynakları tanımlayıp, denemelerini yapıp, ihtiyaçları kalmayınca da bu kaynakları kaldırabilirler. Bulut sağlayıcıların kapasitesinin çok daha fazla olması, uygulamaların yaşam döngüsü için de daha elverişli bir ortam oluşturmaktadır.

5. Maliyet

Projenin sürdürülebilirliği için maliyet önemli bir etkidir. Bulut sağlayıcılar “kullandığın kadar öde” prensibi ile çalıştığı için kuruluş toptan bir donanım satın almak veya ne kadar kapasiteye ihtiyaç duyduğunu önceden tahmin etmek zorunda değildir. Eğer bir kaynağa ihtiyaç kalmadıysa o kaynak kolaylıkla silinebilir ve tekrar ihtiyaç duyulduğu zaman tekrardan oluşturulabilir. Uygulamanın yoğun olmadığı saatlerde ihtiyaç duyulmayan sunucuların silinmesi buna örnek gösterilebilir. Oluşturulan bir sunucunun kapasitesi gereğinden fazla ise bu kapasite kolaylıkla düşürülüp kazanç sağlanabilir.



Şekil 5. Bulut Sağlayıcı Kullanılan Veri Merkezindeki Sunucuların Kaynak Kullanımı

Depolama konusunda ise; çok sık erişilmeyen veriler, indirme süresinin daha uzun olması koşuluyla depolama ücreti daha düşük olan bir alana aktarılabilir.

Sunucusuz hizmetler ise kullanım kolaylığı ile birlikte ücretlendirme konusunda da öne çıkmaktadır. Birçok sunucusuz hizmet, çok yüksek bir limite kadar ücretsiz veya çok düşük ücretlerle kullanılabilir.

Ayrıca bulut sağlayıcılar çok sayıda kuruluşa hizmet verdiği için donanım altyapısı çok büyük olmak zorundadır. Bu da onları toplu olarak çok miktarda donanım satın almaya sevk eder. Çok miktarda

satın alınan donanım daha ucuza mal olmaktadır ve bu ucuzluk bulut sağlayıcının kullanıcılarına da yansımaktadır.

Bu avantajların yanı sıra, bulut sağlayıcılar doğru kullanılmadığı durumda kuruluşlar için büyük maliyetlere sebep olabilmektedir. Deneme amaçlı kurulmuş pahalı bir sunucunun açık unutulması veya uzun dönem kullanılacak şekilde satın alınmış bir sunucunun sonradan değişen ihtiyaçlardan ötürü kullanılmaması gibi senaryolar kuruluşlar için ek maliyetlere yol açar. Optimal şekilde tasarlanmamış bir uygulama mimarisi de büyük maliyetlere yol açabilmektedir.

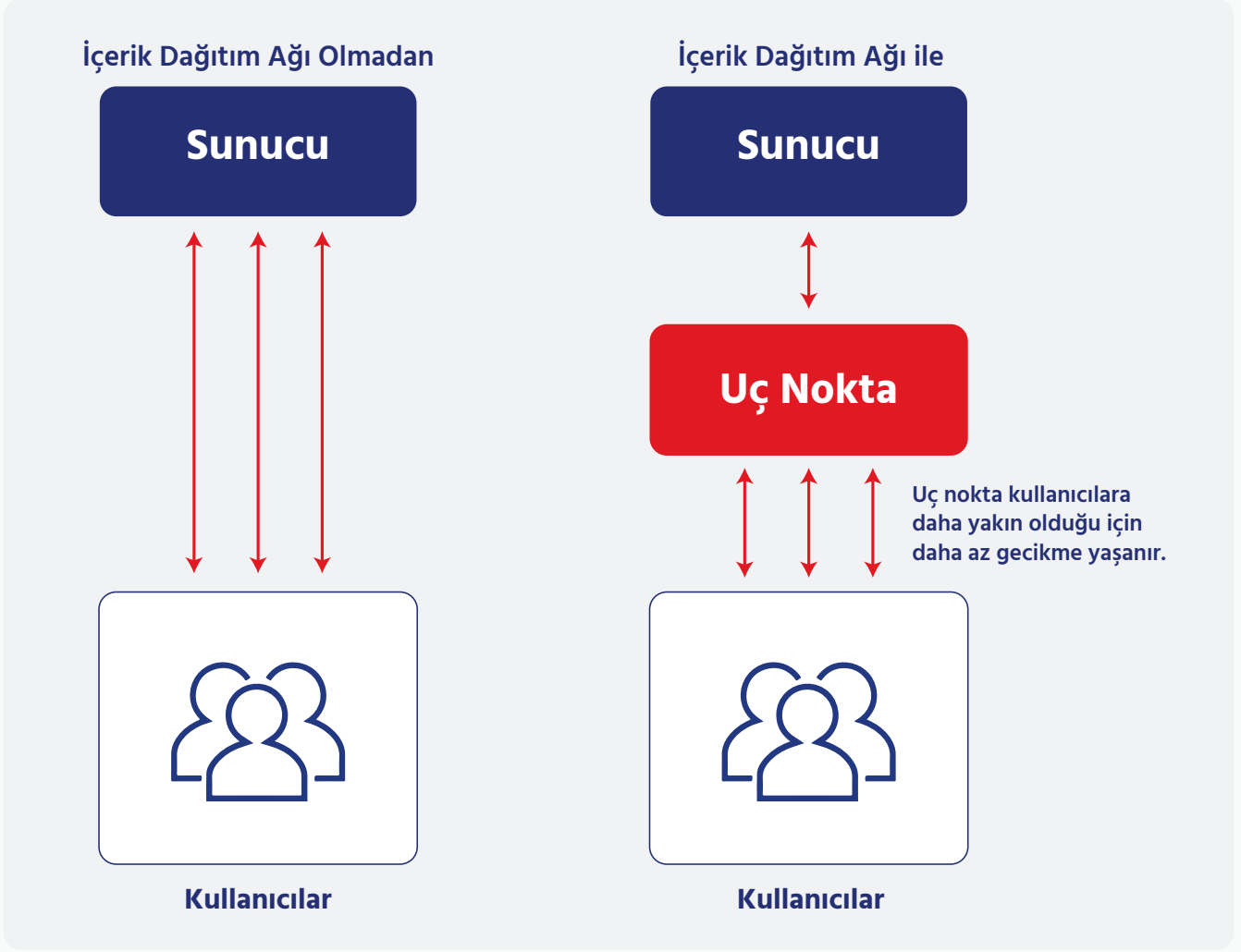


6. Sunulan Hizmetler

Bulut sağlayıcılarda hizmet olarak sadece sunucular değil, sunucusuz adı verilen, sunucu yönetimini devralan ve geliştiriciye sadece gerekli kod yazmayı bırakan yapılar da bulunmaktadır. Hatta yapılmak istenen işlem sadece bir fonksiyondan oluşuyorsa, hizmet olarak fonksiyon olan hizmetler de kullanılabilir.

Çoğu bulut sağlayıcının yapay zekâ, nesnelerin interneti, büyük veri, konteyner orkestrası gibi çok özel alanlar için hazırladığı hizmetler mevcuttur. Bu hizmetlerin en büyük avantajı sunucuların yönetimini bulut sağlayıcının devralmasıdır. Böylece proje ekibi sunucu yönetmeye ve kurulum yapmaya harcayacağı zamanı, geliştirmeye ve iş ihtiyaçlarına yönlendirebilir.

Bulut sağlayıcılar, uygulama ile uygulamanın son kullanıcıları arasındaki gecikmeyi minimuma indirmek için, dünyanın çeşitli yerlerine uç nokta adı verilen merkezler kurarak trafik akışını bu bölgelerden gerçekleştirmektedir. Erişilmek istenen içerik uç nokta üzerinde geçici olarak depolanabilir ve bir sonraki erişimin daha hızlı olması sağlanabilir. Bu sisteme içerik dağıtım ağı adı verilir. Bu yöntem dünyanın her yerinden kullanıcısı olan uygulamalar için oldukça faydalıdır.



Şekil 6. Kullanıcıların Bekleme Süresinin İçerik Dağıtım Ağı Kullanılarak Azaltılması

Bunun yanı sıra sektörde birçok bulut yerel (cloud native) teknoloji kullanılmaktadır. Bu teknolojiler kullanılan bulut sağlayıcıdan bağımsız olarak hizmet sunmayı amaçlamaktadır. Yani bulut kullanıldığı zaman ekibin kullanabileceği hizmetler sadece kullanılan bulut sağlayıcı ile sınırlı kalmaz.

7. Sunucu Bakımı

Eğer sunucuların yönetimi proje ekibinde ise, işletim sistemini güncellemek veya yama yüklemek gibi işlemler tıpkı bulut kullanılmayan senaryoda olabileceği gibi burada da proje ekibinin sorumluluğundadır. Ancak bulut sağlayıcının sağladığı hizmetler kullanıldığı durumda geliştirici ekibi bu tarz işlemlerle uğraşmak zorunda kalmaz. Sunucuların işletim sistemlerinin güncel kalması bulut sağlayıcının sorumluluğundadır ve ekipler bu tarz işlemlerle vakit harcamadan işlerine devam edebilir.

Sonuç ve Öneriler

Sonuç olarak, bulut sağlayıcı kullanmayan veri merkezlerinin karşılaştığı problemler geliştirme süreçlerini aksatmakta ve geliştirme ekibinin motivasyonunu düşürebilmektedir. Bulut sağlayıcılar sunduğu hizmetlerle kuruluşlara özellikle maliyet ve performans açısından büyük faydalar sağlamaktadır.

Ancak, bulut sağlayıcının etkili bir şekilde kullanılması için de ekibin bilgi birikiminin olması gerekmektedir. Aksi takdirde bulut üstünde yanlış yapılan ayarlamalar, özellikle maliyet ve güvenlik konularında kuruluşlara problem çıkarmaktadır.

Kaynakça

1. Albazei, S. (2021, Eylül 20). What to consider when selecting a region for your workloads. AWS. <https://aws.amazon.com/blogs/architecture/what-to-consider-when-selecting-a-region-for-your-workloads/>
2. Albugmi, A. v.d. (2016, Ağustos). Data security in cloud computing. ResearchGate. https://www.researchgate.net/publication/309321387_Data_Security_in_Cloud_Computing
3. AWS Lambda (b.t.) AWS Lambda. AWS. 24 Ağustos 2022 tarihinde <https://aws.amazon.com/tr/lambda/> adresinden alındı.
4. AWS Regions (b.t.). Regions and availability zones. AWS. 24 Ağustos 2022 tarihinde https://aws.amazon.com/about-aws/global-infrastructure/regions_az/?nc1=h_ls adresinden alındı.
5. AWS S3 (b.t.). Amazon S3 fiyatlandırması. AWS. 24 Ağustos 2022 tarihinde <https://aws.amazon.com/tr/s3/pricing/> adresinden alındı.
6. AWS Service Health (b.t.) Service health. AWS. 24 Ağustos 2022 tarihinde <https://health.aws.amazon.com/health/status> adresinden alındı.
7. AWS Snow Ailesi (b.t.). AWS Snow ailesi. AWS. 24 Ağustos 2022 tarihinde <https://aws.amazon.com/tr/snow/> adresinden alındı.
8. AWS SQS (b.t.) Amazon SQS fiyatlandırması. AWS. 24 Ağustos 2022 tarihinde <https://aws.amazon.com/tr/sqs/pricing/> adresinden alındı.
9. AWS Storage Gateway (b.t.). AWS Storage Gateway. AWS. 24 Ağustos 2022 tarihinde <https://aws.amazon.com/tr/storagegateway/> adresinden alındı.
10. AWS Ürünleri (b.t.). AWS cloud ürünleri. AWS. 24 Ağustos 2022 tarihinde https://aws.amazon.com/tr/products/?hp=tile&so-exp=below&aws-products-all.sort-by=item.additionalFields.productNameLowercase&aws-products-all.sort-order=asc&awsfree%3AInvent=*all&awsfree.Free%20Tier%20Type=*all&awsfree.tech-category=*all adresinden alındı.
11. AWS VPC (b.t.) What is VPC reachability analyzer. 24 Ağustos 2022 tarihinde <https://docs.aws.amazon.com/vpc/latest/reachability/what-is-reachability-analyzer.html> adresinden alındı.

12. Azure Front Door (2021, Kasım 11). Microsoft edge locations by abbreviation for Azure Front Door. Azure. <https://docs.microsoft.com/en-us/azure/frontdoor/edge-locations-by-abbreviation>
13. Azure Security (2022, Mart 6). Azure security best practices and patterns. Azure. <https://docs.microsoft.com/en-us/azure/security/fundamentals/best-practices-and-patterns>
14. Cloudflare (b.t.). What is function-as-a-service (FaaS). Cloudflare. 24 Ağustos 2022 tarihinde <https://www.cloudflare.com/learning/serverless/glossary/function-as-a-service-faas/> adresinden alındı.
15. Gancarz, R. (2023, Mayıs 3). Prime Video Switched from Serverless to EC2 and ECS to Save Costs. InfoQ.
16. Glover, C. (2022, Haziran 1). Pegasus Airline breach sees 6.5TB of data left in unsecured AWS bucket. Tech Monitor.
17. Google Cloud (b.t.). Google Cloud documentation. Google Cloud. 24 Ağustos 2022 tarihinde <https://cloud.google.com/docs> adresinden alındı.
18. Jiang, L. v.d. (2020, Ocak). Overview of serverless architecture research. ResearchGate. https://www.researchgate.net/publication/339680821_Overview_Of_Serverless_Architecture_Research
19. Sheth, A. v.d. (2021, Haziran). Research paper on cloud computing. ResearchGate. https://www.researchgate.net/publication/352477780_Research_Paper_on_Cloud_Computing



İşçi Blokları Mahallesi Muhsin Yazıcıoğlu Caddesi No:51/C 06530 Çankaya/ANKARA

+90 (312) 289 92 22 - yte.bilgem@tubitak.gov.tr

TÜBİTAK - BİLGEM Yazılım Teknolojileri Araştırma Enstitüsü (YTE)