

EC2

AWS bulutunu oluşturan 130'dan fazla servis var. Ama bu servisler içerisinde önemlisi ve en çok kullanılanı tartışmasız EC2 yani Elastic Compute Cloud dur.

Detaylı bir şekilde anlatılacak ancak basit haliyle EC2: AWS in size Saniyeler içerisinde sanal bir sunucu yaratmanızı sağlayan Servisidir derim. İkinci kısımda Bulut Bilişim tarihini ve Konseptlerini anlatırken Sizlere bulutun hayatımızı ne kadar kolaylaştırdığını Anlatmıştık. İşte o kolaylıklarının büyük çoğunluğunu biz EC2 ve Las Servisler sayesinde yaşıyoruz. Önceden işim için bir sunucu oluşturmak günlerimi Hatta aylarımı alırken Artık tek bir tıkla 2 saniye içerisinde İsteğime uygun sanal makineyi oluşturuyor Ve sadece kullandığım süre boyunca para ödüyorum . EC2 ve benzeri servisler işte tam da bana bunu sağlıyor.

EC2 Özellikleri

Öncelikle sanal makinelerin yani EC2 lerin ücretlendirme ve satış modellerini bir Anlayalım. Arkadaşlar EC2 lar ilk olarak **On-Demand** yani talep üzerine yöntemine göre satın alınabiliyor. Her sanal makina modelinin saatlik bir fiyatı var. İhtiyacınız olan sanal makineleri oluşturuyor, çalıştırıyor ve kullanıyorsunuz. AWS her ay sonunda o ay bu yöntemle çalıştırdığımız makinenin kaç saat çalıştığını hesaplıyor ve ardından bunu saatlik ücretle ile çarparak o kadar bir rakam kesiyor. Örneğin birazdan göreceğimiz t2 mikro model bir sanal makinanın saatlik ücreti 0.01 dolar. Bunu o ay içerisinde 30 saat çalıştırdığımızı düşünelim ay sonunda aws bize 0.3 dolar fatura gönderecek ya da tüm ay boyunca çalıştığını düşününce de ay sonunda 7.5 dolar gibi bir fatura gelecek. Tam anlamıyla kullandığın kadar öde bu yönetimi genelde kısa dönem ihtiyaçlarımız ve test ve benzeri geçici sunucularımız için kullanıyoruz. Fakat seçeneklerimiz On-Demand ile sınırlı değil.

Aws bunun yanında **Reserved Instance “RI”** gibi bir satın alma yöntemi de sunuyor. Bu da aslında aws e 1 veya 3 yıllık taahhüt vererek indirimli fiyattan yararlanmak olarak özetlenebilecek bir satın alma modeli. Örneğin bir web sitem var ve bu Web sitesinin çalıştığı 2-3 adet temel sunucu parkım var. Bu sunucular en az 1 sene boyunca burada hizmet verecekler. aws den bu sunucu tipleri için reserved instance satın alıyorum tamamını ya da bir kısmının önceden ödüyorum az önceki örnekte saatlik 0,01 olan fiyat 0.007 ye düşüyor aws bana yüzde 30 iskonto sağlamış oluyor. Bunun yanında da aynı zamanda bu süre zarfında bu sunucuya her zaman erişebileceğimi garanti etmiş oluyor. bunun yanında aws geçtiğimiz sene içerisinde **scheduled reserved instances** diye bir model daha çıkardı. Reserved instance a çok benzeyen bu modelde satın almayı 24 saat üzerinden değilde belirlediğim gün hafta veya ayın günleri üzerinden yapıyorum. Mesela izim sadece Saat sabah 9-18 arası çalışan bir uygulamam var. önceden buna sadece 24 saatlik reserved instance satın alabiliyordun. Şimdi bunu sadece bu saatler arasında alabiliyorsun.

Üçüncü ve en ilginç satın alma yöntemi ise **Spot instance** dediğimiz tamamen teklif verme yöntemine göre işleyen satın alma yöneti. aws nin kocaman fiziksel bir sunucu parkı olduğunu tahmin edebilirsiniz. aws tüm müşterine bu parkın içinde paylaşımlı hizmet veriyor. belli saatlerde paylaşımlı ortamdaki kullanım oranı yüksek olurken belirli saatlerde ise düşük oluyor. bu sunucu oranının düşük olması aws in boşuna çalışması yani elinde atıl kapasitesi olması anlamına geliyor. aws bu sorunu çözmek adına spot dediğimiz bir fiyatlandırma belirledi. EC2 çeşitlerinin hepsinin bir on-demand yani sabit bir fiyatı var. bir de bu atıl kapasiteye göre oluşan ve saniyelik değişen bir spot fiyatı var. aynen borsa gibi örneğin günlük pig saatte herkes sanal makina alıp çalışırken spot fiyatı ile on-demand fiyatı aynı fakat gece yarısına doğru spot fiyatı on demand fiyatının 5 de birine düşüyor. işte bu yöntemde sen bir hedef fiyat belirleyerek satın alma emri giriyorsun. örneğin fiyat 0.004 ün altında ise makineyi başla. bunun tek kötü tarafı fiyat o hedefi geçerse makine otomatik olarak kapanıyor. Bu size çok saçma gelebilir bu ne işe yarayacak diyebilirsiniz. Gelin size gerçek hayattan kendi yönettiğim projeden örnek vererek anlatayım. daha önce size yazılım geliştirme döngüsünü anlatırken testin önemli bir süreç olduğundan bahsetmiştim. büyük bir firma hayal edin içinde onlarca farklı yazılım ekibin ondan farklı projede çalışıyor toplamda yüzden farklı uygulama versiyonu yaratılıyor. bu yaratılan her versiyonun bir test sunucusuna atılarak çeşitli test araçları ile test edilerek

sonuçların bir yere çıkarılması gerekiyor. Bu da her gün yüzlerce sunucunun yaratılması üstünde çeşitli testlerin çalışması ve tekrar sunucunun devreden çıkarılması anlamına geliyor. bunlar test işlemleri olduğu için sunucunun testin yarısında kapanması ve benzeri şeyler büyük sorun değil. İşte böyle bir ortamda daha önceden on-demand sunucularla yönetirken tüm test alt yapısını spot'a geçirerek aylık on binlerce dolar tasarruf etmiştik. İşte bu tarz kesintinin sıkıntı olmadığı ya da back processing gibi zamanın zamanın değil maliyetin önemli olduğu durumlarda spot bize büyük imkanlar sağlıyor. ayrıca son yıllarda spot satın almaları otomatize eden bir sürü araç çıktı bu araçlar sayesinde tüm production altyapısını spot üzerinden çalıştıran firmalar var. Son satın alma yöntemi **Dedicated host**. EC2 temelde sana kendi kurduğun sunucu parkında fiziksel sunucular üzerinden paylaşımlı olarak sanal makineler satma hizmetidir. yani senin sanal makinenin ile bir başka firmanın sanal makinesi aynı fiziksel sunucu üzerinde duruyor olabilir. iki makine fiziksel olarak aynı makinenin üzerinde durması birbirinin kaynaklarından haberdar değildirler. ama senin bazı durumlarda bunlardan sakın gerekir ve tamamen kendine ait sana tahsis edilmiş yani dedice edilmiş bir sunucudan hizmet alman gerekir. örneğin tabii olduğun regülasyon veya asa bunu zorunlu kılıyordur. ya da kullandığın özel bir yazılım lisanslamayı fiziksel makine üzerine yapıyordur ki sektörde bunu yapan tonlarca firma var. aws sana dedicated host olarak sana fiziksel bir makine tahsis ederek senin kaynaklarını bunun üstünden kullanmanı sağlıyor. işte 4. ve son satın alma şekli de böyle dedice bir host edinmek. arkadaşlar başlangıç olarak özellikle satın alma modelini seçtim çünkü bu tüm iş yapabilme modelini belirliyor ve en önemli konulardan biri haline geliyor. şu ana kadar kısım tamamen iş tarafı ile ilgili yani satın alma ile ilgili ama neyi satın alabildiğinizden bahsetmedik. şimdi neyi satın alabileceğinize bir göz atalım. aws bizlere hemen hemen her ihtiyaç tipine özelleştirilmiş toplamda 5 kategoride 14 değişik sunucu tipinde sanal sunucu kullanma imkanı sunuyor. ilk olarak standart web ve uygulama sunucu ihtiyaçları için kullanabileceğimiz T, M ve yeni çıkan A modellerini barındıran genel amaçlı sunucu kategorisi var. En sık bu tip sunucuları kullanıyoruz. işlemci gücü gereken ve bunun için Optimize edilmiş Compute Optimize kategorisinde C Tipi sunucular var. Memory Optimize kategorisinde ise yüksek Memory gereken data base sunucular ya da inmemorycaching uygulamaları için optimize edilmiş R X Z V U tipi sonuçlar var. nosql Database ya da data warehouse çözümlerinde ihtiyaç duyduğumuz hızlı disk yapıları için özelleşmiş storage optimize kategorisinde ise bizlere D H ve I tipi sunucular hizmet veriyor Son olarak da makine öğrenimi gibi özel işlem gücü gerektiren işler için F P ve G tipi sunucular var bu 5 kategorideki 12 model sizin bu sanal makinelerin üstüne koyacağınız uygulamalar için özelleşmiş modeller Yani siz Eğer standart bir web sunucu kuracaksanız T veya M modelleri bu iş için en uygun olanları ama tutup bir makine öğrenimi modeli koşturacaksanız bu iş için özelleşmiş CPU tabanlı Pmodel kullanmanız en uygun seçenek olacaktır. Bu listede tüm model ailelerini ve ne için özelleştiğini görebiliyorsunuz. bunun yanında daha rahat anlaşılması açısından M ve C ailelerine ait benzer iki modelin CPU gücü ve RAM karşılaştırmalarına bakalım. Gördüğünüz üzere M ailesi daha yüksek RAM'e sahipken C ailesi işlemci gücünü odaklanmış bir sunucu tipi. Bunlar ve bizler bunların içerisinde işimize uygun olanı seçerek işimizi hallediyoruz ama işimiz Sadece bu seçimde bitmiyor. Şimdi Dikkat ettiyseniz bu model harflerinin yanında 1 2 3 4 ve 5 gibi rakamlar da görüyorsunuz Arkadaşlar bu rakamlara da AWS dünyasında jenerasyon deniliyor Örneğin M Ailesinin son jenerasyonu M5. Şimdi bunu şöyle düşünün AWS bu M ailesini ilk yarattığında 2006 senesinde o zamanın Intel işlemcileri ve teknolojileri ile bir sunucu parkı kuruyor daha sonra aradan zaman geçiyor ve Intel yeni işlemciler ve yeni sunucu modelleri çıkarıyor. 2010 senesine gelindiğinde aws bu yeni sunuculardan satın alıp bunlardan oluşan ve M modellerinin ikinci jenerasyonu yani M2'yi devreye alıyor daha güçlü işlemciler daha hızlı bir altyapı ve daha uygun fiyatlı bunları devreye alıp yavaş yavaş kullanıcılarını buna geçmeye teşvik ediyor ve ömrü dolan eski sunucuları yavaş yavaş sistemlerinden kaldırarak M1 ailesini öldürüyorlar aynı şekilde zaman gelip yeni sunucular çıktıkça M3 M4 ve sene başında M5 jenerasyonuna kadar geliyorlar şu anda aws M ailesinden kullanabileceğiniz sunucular M4 ve M5. AWS hala eski M4 jenerasyonu kullanırmaya devam ediyor ama ekranda Gördüğünüz üzere aynı modelin en M4

ile M5 arasında hem performans hem de fiyat farkı var Merak etmeyin şaşırmayın M5 Hem daha performanslı hem de daha ucuz. M4 satın alırsak 6.30 ECU yani elastic Compute unit performansı alıp bunun karşılığında saatini 0,111 dolar öderken M5 alırsak 8 ECU performansına sahip olup karşılığında saat ve 0,107 dolar Ödeyeceğiz. Arkadaşlar bu Eco terimi ile Kafanız karışmasın evde çoğu durumda CPU performansının adlandırmanın belirli başlı bir yöntemi olmadığından buna kendileri ECU adında bir terim uydurmuşlar. Siz bunu işlemciden alacağınız performans olarak okuyun ve Gördüğünüz üzere Yeni jenerasyon Hem daha ucuz hem de daha performanslı. AWS yeni kullanıcıların artık Yeni jenerasyon olan M5 modellerine geçmesini istiyor ki M4 kullanımı tamamen bitsin ve kullanım ömrü dolan sunucuları sistemden çıkarabilsinler ama aynı zamanda M4 kullandırmaya devam ediyorlar Çünkü birçok firma birçok iş yükünü buradaki sunucularda tutuyor bunların M5 e geçmesi için firmalar tarafından birçok test yapılması ve geçişin planlanması gerekiyor. Bu işlemler tamamlanana kadar aws B M4 bu jenerasyonu daha uzun bir süre burada tutmaya devam edecek. şimdi bu ekranda M4 ve M5'in aynı tipteki sunucularını karşılaştırırken yanlarında Large XLarge gibi ifadeler görüyoruz. arkadaşlar geldik Bir diğer sunucu seçeneğine EC2 tiplerinin isimlerindeki bu son kısımda Sunucunun boyutunu belirleyen kısım AWS tüm sunucu ailelerinde her ihtiyaca uygun çeşitli boyutlarda sunucular yaratmış durumda Örneğin Bu listede Gördüğünüz üzere M5 ailesinde Large dan başlayarak 24xlarge'a kadar giden 6 farklı boyutu var. bu model farklılıkları çok basit haliyle aslında aynı sunucu ailesinin farklı RAM CPU depolama ve bant genişliği sağlayan modelleri kendi ihtiyacımıza göre bu modeller arasından bir tanesini seçerek sunucumuzu oluşturuyoruz yani hepsini özetlersek kendimize bir sanal sunucu yaratmak İstedığınız zaman öncelikle bu sanal sunucu hangi iş için kullanılacak ona karar verip o işe uygun bir aile seçecek. hemen ardından Bu ailenin içerisinde işimize en uygun modelini bulacak daha sonra bunun en son jenerasyonunu mu yoksa mevcut iş yükünüz ile uyumlu bir önceki jenerasyonu mu kullanacağız ona karar verecek ve son olarak da hangi boyutta bir makineye ihtiyaç duyduğumuzu kararlaştırıp o modeli seçerek sanal makinamızı oluşturacağız ve böylece ihtiyacımıza uygun bir sanal makine seçmiş olacağız. Tüm bu EC2 modellerini bir arada görüp fiyat ve özellik karşılaştırması yapabileceğimiz EC2instances.info diye güzel bir açık kaynak projesi var bu site aracılığıyla modelleri fiyatlarına bakabilir ve fiyat karşılaştırabilirsiniz konuyla ilgili son bir hatırlatma daha yapmak istiyorum arkadaşlar 2018 senesinin sonuna doğru 2 tane daha EC2 tipi duyurdu ve bunlar diğer modellerden biraz farklı geleneksel olarak Bugüne kadar hep Intel işlemcilerle devam etti. Fakat son yıllarda ARM tabanlı işlemciler ve oldukça popüler durumda bu popülerliğe kayıtsız kalamayan aws ilk defa kendi ARM tabanlı işlemcisini geliştirdi ve bu işlemci ailesi ile oluşturduğu sunucuları A tipi sunucuları olarak adlandırdı yani siz M C V Ya ne bileyim D tipi sunucu seçersiniz. Bunlar Intel işlemcili ama atipi EC2 instanslar ARM tabanlı işlemciler ile geliyor Bunu Aklımızın bir köşesinde not edelim. AWS ortaya koyduğu bir diğer instance de yıllar sonra ilk defa intelin en büyük rakibi AMD ile birlikte çalışarak çıkardıkları AMD serisi oldu. AWS 2018 yılının sonunda M5 ve T3 instance ailelerinin AMD modellerini çıkardı ve bunlara M5A ve T3A dedi. Bunlar birebir Intel işlemcili modellerle aynı fakat AMD her zaman intelden daha uygun fiyatlı olduğundan neredeyse yüzde on daha uygun fiyatlı. AWS 2019 senesi içerisinde bunları denemek istiyor ve deneme eğer deneme başarıyla sonuçlanırsa AMD tabanlı işlemciler ile oluşturulmuş daha fazla instance tipi göreceğiz geldik bölümün sonuna şimdilik bu bölümü sonlandıralım ve Öncelikle burada öğrendiğiniz bilgileri sindirelim.

Hız kesmeden EC2 dünyasını tanımaya devam edelim. ilk bölümde bize sunduğu sanal makine tiplerinden bahsettik ve tüm modelleri öğrendik Ama sanal makine seçiminde bakacağınız tek Kriter bu değil bunun yanında seçim yapabileceğimiz ve konfigüre edebileceğimiz bir Çok değişken var şimdi de gelin onları tanıyalım ilk olarak hep sanal makinelerin cpu ve ram seçenekleri üstünde durduk ama hiç bu sanal makinelerin üzerine kuracağımız işletim sisteminin uygulamanın ve diğeri de datalarımızın duracağı disklerden bahsetmedik. bu alanda aws bize iki temel seçenek sunuyor birincisi Instance Store ve diğeri de

EBS. Bunlardan ilki olan instance Store sanal makinenin üstüne çalıştığı fiziksel sunucu ya direkt bağlı disklerin kullanıldığı toplama yöntemine verilen isim veya SSD veya manyetik yani yıllardır kullandığımız hard disk modellerinden olabiliyor bazı sunucu aileleri bu tipin instance Store depolama modeli desteklerken bazıları Sadece diğer modeli olan ebs i destekliyor. Instance Store depolama modelinin avantajı direkt sanal makinenin bağlı olduğu fiziksel sunucu üzerinde olduğundan yüksek erişim hızlı ve çok düşük gecikme süresi sağlaması Bu nedenle disk giriş çıkış hızının önemli olduğu storage optimize tipi D H ve I ailelerinin bazılarında bu tip depolama çözümleri kullanılıyor dezavantajı ise sanal makinenin bir şekilde kapanırsa Buradaki tüm datayı kaybediyorsun altında yatan fiziksel makineye bir şey olur ya da sen sanal makineyi kapatırsan bu disklerdeki verileri ulaşılmıyorsun. Çünkü aws sana aynı sanal makineyi aynı fiziksel sunucuda tutma garantisini doğal olarak sağlayamıyor. sanal makinalar aws dünyasında bulunan binlerce fiziksel sunucu arasında dolanıp duruyorlar sanal olmasının avantajı da bu zaten. bazen sen farkına bile varmadan senin sanal makine fiziksel sunucusunu değiştiriyor İşte Instance Store bulunduğu sunucuya fiziksel olarak bağlı bir depolama çözümü olduğundan sanal makine Bir şekilde başka bir sunucuya geçmek durumunda kalırsa bu diskleri erişimi kaybediyor ve veriye yetişemiyorlar. gelin hemen Bu diskleri durumunu hem de sanallaştırmayı daha rahat anlamak adına bunları görsel bir diyagramı üzerinde inceleyelim. Sanallaştırma konusunu anlatırken buna kısaca değinmiştik aslında. sanallaştırma Öncelikle 1 adet fiziksel sunucu vardı. Daha sonra bu sunucu üzerine Biz bir sanallaştırma yazılımı kuruyorduk bu katmana Hypervisor diyoruz. Bu Hypervizör bizim bu üstünde koştugu fiziksel makinenin kaynaklarını birçok sanal makineye ortak ve izole olarak kullandırma imkanı sağlıyor. şimdi bu tek bir fiziksel makine olduğunda bu şekilde ama tek bir fiziksel makine ile büyük bir ortam kurmak mümkün değil. Şöyle düşünün eğer bu tek fiziksel makinede bir sorun çıkarsa ne olur. Bunun üstüne çalışan tüm sanal makineler devre dışı kalır. Bunu engellemek için de yöntemler bulmamız gerekiyor o nedenle de ne yapıyoruz ortama ihtiyacımız kadar fiziksel makineye ekliyoruz. aws dünyasında bu makine adedi binlerle ifade ediliyor. Daha sonra da bu tüm fiziksel makinaları birbirleriyle hızlıca haberleşme bilecekleri yüksek hızlı ağ altyapıları ile birbirlerine bağlıyoruz ve tüm makinelerin hypervisor yazılımlarının birbirleriyle ortak hareket edebilecekleri konfigürasyonları ayarlıyoruz. Daha sonra da ortamı Tüm bu fiziksel makinelerin Yüksek hızda erişebileceği ve içerisinde yüzlerce disk bulunan depolama aygıtları ekliyoruz. tüm bu fiziksel cihazlar bu depolama aygıtlarına erişebiliyorlar. Artık sanal makinelerin Tüm dataalarını bu ayrı depolama biriminde tutuyoruz Böylece Eğer böylece fiziksel sunuculardan bir tanesi devre dışı kalırsa onun üstüne çalışan sanal makineler konfigürasyonlarını üstünde kapasite olan diğer fiziksel sunucuya geçirerek oradan kesinti olmadan çalışmaya devam ediyorlar. eğer o sanal makinenin datası o fiziksel makinelerin bir tanesinin üstünde duruyor olsaydı o fiziksel makine Bir Sebep ile devre dışı kalırsa o sanal makine konfigürasyonunu diğer fiziksel makineye taşınmış olsa da ana data o sorunu fiziksel makinede durduğu için çalışmayacaktı. İşte bu tarz fiziksel host üstünde depolama çözümünün kullanımına Instance Store. Bu grafikteki depolama biriminde depolama çözümü kullanıma aws dünyasında EBS yani elastik block storage deniliyor. s3 bölümün de başında üstüne basa basa anlattım file based ve blog based temellidir. Yani bir makineye bağlanarak üstüne işletim sistemi ya da uygulamada kurabildiğiniz tipteki depolama çözümü. EBS yüzde 99.999 erişilebilirlik garantisi sağlayan ve aynı EB içerisinde birden fazla fiziksel cihaza repliki edilen içerisinde SSD ve HDD tabanlı disk altyapılarının bulunduğu bir blog based çözümü. EBS in instance store a göre bir diğer avantajı da snapshot desteğine sahip olması yani ile EBS ile oluşturulan disklerin belirli bir andaki kopyasının kaydedilerek daha sonra bu kopyadan yeni makineler yaratılabilir ya da bu kopyayı yedek olarak kullanma imkanına sahip oluyorsunuz. EBS toplamda 4 farklı depolama imkanı sunuyor. SSD yani solid state disk dediğimiz oldukça hızlı disklerden oluşan General Purpose SSD CP2 genel kullanımlara ve işletim sistemi ana diskleri için uygun bir depolama çözümü. Provisioned IOPS IO1 SSD ile oluşturulmuş ve IOPS değerinin önceden seçilebildiği ve buna göre ücretlendirme sağlayan model. IOPS dediğimizde bir diske saniyede ne kadar okuma ve yazma işleminin yapılabildiğini belirten bir değer. IOPS

un yüksek olması kabaca daha hızlı bir disk anlamına geliyor ve yüksek hız gerektiren Database uygulamaları gibi iş yükleri için kullanılıyor. SSDi dışında HDD yani hard disk drive yani yıllardır kullandığımız katı disk sürücülerle oluşturulmuş ve throw put için Optimize edilmiş iki ayrı model daha mevcut. Throwput da bir depolama sistemine saniyede kaç megabayt veri geçişine izin verildiğini belirten değer yani IOPS u bir arabanın 0'dan 100 kilometreye kaç saniyede çıkabildiğini gösteren değer olarak Throwput da o arabanın maksimum ne kadar kıza çıkabildiğini gösteren değer olarak düşünersek sanırım Kafalar daha az karışır. Bu ikinci hard disk tabanlı depolama yapılarının ilki olan SD1 throwput yönünden Optimize edilmiş durumda ve sık erişilen yüksek hacimli dosya depolama için uygun diskler. Örneğin veri ambarı uygulamaları. son olarak da AWS in artık sistemlerinden yavaş yavaş kaldırdığı Cold HDD dediğimiz SD1 modeli bu en ucuz ve en yavaş model ve log dosyaları gibi dosyaları tutmak için uygun bir depolama imkanı. işte sanal makinemizin modelini belirledikten sonra o model EBS destekli bir modelse en çok kullanılan EC2 ların tamamına yakını sadece EBS destekliyor o sana makinenin üstünde uygulama kuracağımız işletim sistemi uygulama tüm veriler bu disk tiplerinden birinin üstünde duruyor ve bizler sanal makine kurulumunda bunlardan birini seçiyoruz Peki geldik başka bir soruya makine modelimizi belirledik diskinizi de seçtik. Peki bu makineyi ayağa kaldırıp çalıştırma adına bizim bir de bu makinenin içine işletim sistemi kurmamız gerekiyor. Peki bunu nasıl yapacağız tutup da makineye bir şekilde bağlanarak üstüne DVD falan takıp bu işletim sistemi kuracağız işte arkadaşlar geldik şimdi bir instance oluştururken Seçtiğiniz diğer önemli bir Kriter olan işletim sistemi ve platform konusunda yani aws dünyasındaki adıyla Amazon machine Image Kısacası ile amı. arkadaşlar çok kabaca amı içerisinde daha önceden tanımlanmış işletim sistemi ve uygulama dosyaların bulunduğu bir sanal makine şablonu. Siz bir sanal makine oluştururken bu sanal makinenin hangi şablonla oluşturulacağını seçerek o şablon içerisinde sunulan özelliklere göre bir işletim sistemi ve uygulama listesi edebilirsiniz temelde üç çeşit AMI vardır. ilk öncelikle public shared community amı dediğimiz çeşitli topluluklar ve Amazon kendisi tarafından yönetilen amı ları vardır. Örneğin Linux dünyasının en ünlü kombinite Driving dağıtımlarından olan sentosun kendi AMI leri vardır ve bu amı ler sayesinde Sentos yüklü bir sunucu elde edersin. ya da Amazon un kendisinin hazırladığı ve içerisinde temel Linux sunucu işletim sistemi yanında aws cli gibi birkaç aws ile alakalı uygulamanın da yüklü geldiği Amazon Linux ya da diğer bir örnek olarak Windows Server 2016 AMI. bunların tamamı ve daha fazlası herkesin erişimine açık olan AMI lar. bir diğer çeşidi de paid ami dediğimiz ücretli sümülerdir. Bunlar çeşitli firmalar ya da bağımsız de developerlar tarafından yaratılan ve içerisinde işletim sistemi yanında çeşitli uygulamalarında hazır paketlerdir. Örneğin bir saat uygulama yaratıcısı kendi uygulamasını yükle olduğu bir linux sunucu imajı yaratarak bunu Amazon marketplace dediğimiz aws mağazasına koyar ve bir fiyat belirler sen de bu fiyatı kabul ederek bunu satın alırsın ve bu AMI ile kendine sanal makina yaratırsın. bu uygulamayı kullanmaya başlarsın bu tarz yüzlerce AMI mevcuttur ve son olarak da 3. çeşidimiz olarak kendi yarattığımız ve yönettiğimiz AMI larr bulunur. Sen kendi uygulamalarının ve ayarlarının olduğu AMI ler yaratarak bunların halihazırda yüklü olduğu sanal makineler yaratmak için kullanabilirsin.

EC2 Uygulama-1

merhaba arkadaşlar log in olalım ve dünyasına adım atalım arkadaşlar. hemen buraya EC2 yazıyorum ve EC2 yönetim konsolunu açıyorum İşte karşınızda EC2 yönetim konsolu duruyor sol tarafında ile ilgili tüm Temel ve alt servisler ve onların ayarlarını erişebileceğiniz menüler var bunların hepsini detaylıca göreceğiz bu orta kısımda Gördüğünüz üzere bir de kaç tane kaynağımız da olduğundan ve bunların detaylarından bahsediyor hemen altında da bağlı bulunduğumuz ve region ve onu oluşturan availability zone ların durumunun listeliyor hepsi şu anda sorunsuz gözüküyor sağ tarafta da bu regin ile ilgili planlanan bir bakım çalışması olup olmadığını gösteren ekran bulunuyor biz running instance basarak bizim mevcut sanal makinelerimizin listelendiği ekrana geçelim şu anda ekran boş çünkü herhangi bir sanal makine yaratmadık hemen launch instance a tıklayarak ilk sanal makinemizin yaratmaya başlayalım Şimdi arkadaşlar bize ilk

ekranda hangi ami ile bir sanal makine oluşturmak istediğimizi soruyor bir önceki bölümde anlattığım üzere burada üç temel seçeneğimiz var Bunlardan ilki community base bir ami seçme şansımız var. Bunlar bu firmaların ya da komünitelerin resmi ami ları. Örneğin Ubuntu bir sanal makine kurmak istersek Ubuntu firmasının sağladığı ve onlar tarafından update edilen bir ami seçip yolumuza devam edebiliriz. ya da aws market place e geçerek başka firmaların bize ücretli sunduğu ami bir tanesini seçebilme şansımız var. burada yüzlerce farklı yazılım mevcut misal junifer firmasının ürettiği firewall yazılımının yüklü olduğu bir ami seçerek saatlik 0.55 dolarlık bir ücretle bunu kullanma şansına yaşayabiliyoruz son seçenek de kendi yarattığımız bir ami kullanmak buna Daha sonra tekrardan geleceğiz öncelikli bir sanal makinemizin yaratalım bundan sonra daha sonra biz kendimize bir ami yaratacağız ve o da burada daha sonra seçebilmek için çıkacak Ve şimdi de Quick Starta basalım ve Amazon Linux 2 ami ile devam edelim arkadaşlar burada Gördüğünüz üzere bu ami ile ilgili çeşitli özellikleri altında listemiz ilk olarak diyor ki bu ami in root device i yani işletim sisteminin kurulduğu bölüm ebs volume olmak zorunda yani bunu destekliyor hemen ardından da bir virtualization type olarak hbm virtualization type da geliyor ve son olarakta enable yani enhance network enhance enabled seçeneği yes olarak işeretlenmiş yani böyle bu ami ile enhance network un açık olduğu bir sanal makina yaratabiliyoruz. Ben burada kısaca Bu virtualization type olarak belirtilen bölümden bahsetmek istiyorum Arkadaşlar. virtualization type dediğimiz altta bulunan fiziksel sunucu üstünde koşan hypervizör katmanını belirtiyor bu iki çeşit hpm ve para virtual. para virtual artık eski ve aws dünyasında artık Hemen hemen tüm güncel modeller HPM üzerinden geliyor biz hemen bu Amazon Linux ami ini seçerek selecte basarak Yolumuza devam edelim şimdi sırada bu seçtiğimiz ami i ne tipte bir sanal makine konfigürasyon üzerine kuracağımız soruyor tüm EC2 tepler burada listelenmiş durumda Biz ücretsiz seçeneğimiz olan T2 Micro seçerek yolumuza devam edeceğiz arkadaşlar ve next konfigüre instance details ile devam ediyorum şimdi bu kısım önemli arkadaşlar ilk olarak bu konfigürasyonda kaç makine kurmak istediğimizi belirtiyoruz. Biz bir tane makine kurmak istediğimiz için bir seçeneği ile devam edeceğiz. sonrasında bize spot isteğinde bulunup bulunmayacağımızı soruyor burada bir teklif verip fiyat oraya düşerse aç diyebiliriz. Yani biz deriz ki buraya şu anda 0,0038 biz hani 0,0027'ye düşünce aç şeklinde spot isteği gönderebiliriz. Bunu zaten teori kısmında anlatmıştım işte onu buradan açabiliyoruz arkadaşlar. Sonrasında da a kısmı var bunların hepsini vpc altında gördüğümüz daha bir anlamlı olacak. Biz Şimdilik bunları default seçenekleri ile devam ediyoruz. Eğer bizim yarattığımız bir wpc si olsaydı buradan bu wpc seçenekleri seçebiliyoruz kendi vpc mizi. hemen altından da hangi subnet i istediğimizi seçebiliyorduk bunlara geleceğiz arkadaşlar birkaç bölüm Sonra detaylı şekilde ve vpc nin hepsinin ve tabletları neler olduğunu ve nasıl yaratabileceğimizi göreceğiz. hemen bunun altında da auto assign public ip var yani Biz bunu seçersek enable edersek bu makineye bir public ip verecek ve bu sayede biz bu makineye bağlanabiliriz. Bu seçeneğin seçili olması gerekiyor. placement gruba eklemek istiyorsak Onu buradan seçebiliyoruz ve hemen altında da bir rezervasyon yaratmak istiyorsak bir kapasiteli reservation yaratmak istiyorsak onu da buradan ekleyebiliyoruz arkadaşlar şimdi geldik iam rol seçimine. sizlere iam kısmında rolleri anlatırken bir adet s3 lü rol yaratmıştım şimdi bunu hemen buradan hemen seçelim Arkadaşlar bu sayede biz bu makinanın S3 kaynaklarımızı erişebilmesini sağladık bunun ne olduğunu da birazdan makineye bağlandığımız zaman görebileceksiniz hemen altındaki shutdown behavior kısmında da bu makine Eğer işletim sistemi içerisinde bir kapatma komutu ile kapatılırsa aws nin bu makineye ne yapmasını istediğimizi belirtiyoruz. Burada iki tane seçenek var stop ve termanite. stop dersek makine kapanır biz sanal makina parası ödemeyiz sadece bağlı bulunduğu diskin masrafını öderiz ama istediğimiz zaman da makineyi yeniden açar kullanırız Terminate seçersek de bu makine kapatıldığı anda aws bu makine ile ilgili kaynakları tamamen siler. bir sonraki thickbox ise arkadaşlar yani Protect against accenditial termitanion bu seçimde olduğu zaman siz bu makinayı api veya veya management console üzerinden Terminate edemezsiniz koruma kalkana kadar makine silinmez Biz bunu seçelim ve etkilerine daha sonra bakalım bunun hemen altında da arkadaşlar monitöring kısmında detaylı izlemeyi açıp açmak

istemediğimizi soruyor. Şimdi Normalde bu makineler monitöring olarak 5 dakikada bir gözlenir eğer biz bu detail monitöring seçersek bu süre 1 dakikaya düşer bunu isteyip istemediğimi soruyor Tabii ki bunu seçtiğimiz zaman ekstra bir ücret çıkacak. Biz o yüzden şu anda bunu seçmiyoruz arkadaşlar. Tenancy kısmında kısmında da önceki bölümde anlattığım dedicated host mu yoksa paylaşımı bir sistem mi istediğimizi soruyor ve biz paylaşımlı yani share ile devam ediyoruz Bunu hemen altındaki T2/t3 unlimited seçeneği ise bu günün konusu değil arkadaşlar. advanced details kısmında ise bizim için önemli bir hizmet sağlanıyor ben buraya bu sanal makineye ilk kez çalıştırıldığında çalışmasını istediğim Komutları girerek makineyi istediğim işler için kurulduğu anda hazır hale getirebiliyorum. Örneğin Gelin bu makine ilk kurulup açıldığında tüm işletim sistemi ve paket güncellemelerini kendi kendine bizim için hazır hale getirsin. bunun için arkadaşlar hemen buraya alta `#!/bin/bash` enter `yum update -y` komutunu girdim bu komut iki satırlık bir Script. burada biz en çok kullanılan linux shell i olan ve hemen hemen her linux dağıtımında komutların nerede çalıştıracağımızı söylüyor Hemen ardından da Linux paket yöneticisine `update` yani güncelleme komutunu `-y` anahtarı ile giriyoruz bu `-y` anahtarı Eğer `update` sırasında bir soru sorar ve onay isterse Onayla ve bunun için bizden onay bekleme manasında geliyor Neyse çok fazla Kafanız karışmasın linux apayrı bir dünya çözüm mimarlığa doğru ilerlerken linux konusunda uzmanlaşmak da yapmanız gereken işler listesinin en tepesinde bulunuyor olarak sizlere. Bu konuda da yalnız bırakmıyoruz ve çok kısa bir zaman sonra Linux temelleri ve hemen ardından da Linux sistem yöneticiliği kurslarını yayınlıyoruz takipte kalalım. Böylece bu sanal makineyi çalıştırdım da çalışacak komutu da girmiş olduk. Şimdi Next ile devam ederek bu makineye bir disk ekleyelim. arkadaşlar temelde iki çeşit disk ekleyebiliyoruz biri varsayılan olarak Root dediğimiz yani üzerine işletim sistemi kurulacak olan yani bu disk otomatik olarak ekliyor Burada da seçebileceğimiz iki şey var Bir tanesi Arkadaşlar bu diskin boyutu yani kaç gigabyte olacak default olarak 8 biz istersek Örneğin bunu 80 yapabiliriz. Bizim ihtiyacımız için 8 yeterli olacaktır hemen yanında da volume type olarak bunun ne çeşit bir disk olmasını istediğimizi seçebiliyoruz buradaki Volume type ları da bir önceki bölümde anlatmıştım. Bu arada encrypted kısmına dikkatinizi çekmek istiyorum. varsayılan olarak aws EC2 sanal makinelerin Root bölümlerini encrpt edemiyor bir yöntem ama bundan sonra geleceğiz ve bunun önemli bir bilgi olarak kafanıza kaydedin. Çünkü bu bilgi ile ilgili şaşırtma sorular sınavda karşımıza çıkabiliyor. Bu root bölüm ama istersek bu sanal makineye başka disklerde ekleyebiliyoruz. Burada gördüğünüz add New volume bölümüne tıklayarak istediğim kadar bu makineye yeni sanal diskler ekleyebilir bunların boyutlarını örneğin değiştirebilir ve ardından da istersem bunların volume type larını da değiştirebilirim ama bizim işimiz için arkadaşlar tek bir tane Root volume ve 8 GB'lık bir volume gayet yeterli olacaktır Hemen next ile tag kısmına geçiyorum tag kısmına daha önceden kısaca değinmiştik taglar key value pair olarak oluşturulan aws dünyasında yarattığınız kaynaklarını daha rahat takip etmenize olanak sağlayan etiketler örneğin bu makineye name yani isim adında bir key ve ona da ilk sunucu adında Value verelim hemen add tag diyorum buna key name olsun value suda ilk sunucu olsun Böylece buna bir tane tag eklemiş oldum Bir tane daha ekleyebilirim örneği burada şöyle diyelim departmanı da bunun yani bu sanal makine ip departmanına ait olsun hemen Next ile konfigüre Security grupla devam ediyorum geldik security Grup yani güvenlik grubu ayarlarına bunun ve diğer tüm a ve güvenlik ayarlarının detaylarını vpc konusunda göreceğiz O yüzden şimdi çok detay vermeyeceğim ama Security grubu kabaca bu sanal makinenin önünde duran bir güvenlik duvarı olarak düşünün varsayılan olarak bu sanal makine tüm dış dünyadan gelecek trafik için kapalıdır fakat biz yapacağımız işe göre bu makinenin dış dünyadan çeşitli portlar yani kapılar üzerinden erişilmesini isteyebiliriz bunun için öncelikle security Grup yaratıp üstünde ayarlar yaparak bunu makineye atamamız gerekir Bunu da bize bu ekran sağlıyor gelin Öncelikle yeni bir security grup yaratalım ve bir isim ve açıklama girelim hemen create New Security grubu seçiyorum hemen Buna da EC2 tag Grup ismini giriyorum discription kısmı da şöyle diyeyim ilk yarattığımız güvenlik grubu diyelim. Arkadaşlar bunun altında da işte port açma kısmı var add role diyerek yeni kurallar ekleyebilirim hemen buradan çarpı ile bunları silebilirim zaten bir tane varsayılan olarak bir kural var ne diyor type olarak

gördüğünüz gibi bütün aws bize daha önceden birçok servisi tanımlamış durumda Örneğin dns seçersen Protokolü ve port Range otomatik olarak DNS protokolüne göre değişiyor biz bu makineye dış dünyadan SSH erişmek istediğimizden SSH seçiyoruz Protokolü TCP Port u da 22 ve buna kim erişebilirsin? Şimdi burada custom da 0.0.0.0/0 var bu şu manaya geliyor arkadaşlar dünyanın her yerinden buna 22 portundan gelmesine izin ver. Normalde production ortamlarında bunu bu şekilde yapmıyoruz bunu ne yapıyoruz örneğin my ip diyoruz benim şu anda bulunduğum ip den bağlanılmasını sağlıyorum ya da custom diyorum Buraya kendi Örneğin 212 12 12 0/ 24 Network'ün den sadece buraya girilebilirsin diye resdict ediyorum yani kısıtlıyorum .ama bizim işimiz için any where i seçiyoruz 0.0.0.0./0 dan ve ::/0 bu da ip V6 versiyonu bunun. yani dünyanın her yerinden bu tsp 22 Portundan buna erişilmesini sağlıyoruz Hemen buraya iki tane daha Kural ekleyeceğim. Öncelikle bir http kuralı eklemek istiyorum çünkü bu makineyi bir web Server olarak tanımlayacağız daha sonra o şekilde ayarlayacağım hemen bunun altına da http tipi yani güvenlik Protokolü seçiyorum tsp 80 ve 443 hepsine anywhere diyorum ve böylece yeni bir tane Security Grup yarattık bu security Grupa destruction girdik ve hemen altından da ssh http ve https portlarından tüm dünyadan erişilebilmesini sağladık review and launch atıyorum ve son ekranı Geliyorum bu son ekranda Artık tüm seçtiğimiz ayarlarımız bir özetini görüyoruz son bir kontrol yapıp hepsini böyle şu ana kadar Seçtiğiniz tüm ayarları bir kontrol ediyoruz ve her şey yolundaysa da launch a tıklayarak bu sanal makineyi oluşturmaya başlıyoruz Arkadaşlar bu ekranda edebiliriz aws bize bir key pair yani anahtar çift atmamızı söylüyor. aws Buraya dünyasında sanal makinelere bir şifre ile değil de bu anahtar çiftleri ile bağlanıyoruz public olanı aws hesabınıza duracak olan çiftlerin private olanını da sisteminize indirip kaydediyoruz bu dosyayı iyi saklamamız lazım Çünkü bu dosyayı kaybedersek bir daha bu sanal makineye bağlanma şansımız kalmıyor. hemen biz buradan create a new key pair diyelim buna da bir isim verelim Ne diyelim mesela iteğitim diyelim ve bunu download diyelim kaydedelim hemen kaydedelim ve son olarak da launch instance a basalım ve sanal makineyi yaratalım Evet arkadaşlar şu anda sanal makinamız yaratılmaya başlandı makine bir yandan hazır olurken gelin biz bu sanal makinenin tüm özelliklerine bir göz atalım ve menülerini tanıyalım. Öncelikle actions kısmına bir tıklayalım bu ekrandan basitçe arkadaşlar makineyi restart edebiliyorsunuz ya da durdurabiliyorsunuz ya da tamamen silebiliyorsunuz ve instant state kısmında stop yeniden başlat veya tamamen terminated seçeneklerini seçebiliyorsunuz gelin önce terminate işlemini yani makineyi yok etmeyi Deneyelim. Bu seçenek şu anda kapalı. Niye Çünkü biz makineyi oluştururken yanlışlıkla terminate edilmesin seçeneğini işaretledik İşte bu seçenek sayesinde biz bu makine yine buradan ne de api ne de cli üzerinden hiçbir şekilde millet edemiyoruz Yani bu işlemleri yaptıktan sonra terminate demiyorum bunun için Arkadaşlar öncelikle Action change terminatin protection seçip bunu disable etmem gerekir ve tekrar gelip burada instance state determinate e basarsam artık ve bu seçenek seçilebiliyor ve ben bu makineyi terminate edebiliyorum. Bunun dışında da bu instance settings kısmında bu makine ile ilgili birçok ayrı değiştirebiliyoruz Örneğin aim rolünü değiştirebiliyoruz termination protection rolünü Az önce gördüğünüz gibi Shut Down behavior değiştirebiliriz bu bütün makineye oluştururken seçtiğimiz seçenekleri Hemen hemen bir çoğunu buradan instance settings kısmından değiştirebiliyoruz image kısmından da Arkadaşlar bu makine ile ilgili ami yaratıyoruz ki Birazdan buna Geleceğiz şimdi bu arada makinemiz running e düştü yani şu anda makinamız hazır biz istersek bu makineye bağlanabiliriz ama gene öncelikle bu makinenin özelliklerini Bir bakalım Yani bu konsolu tanıyalım Burada neler varmış Bu aws bize ne imkanlar sunuyor bunları bir görelim. Öncelikle descripcion kısmında bu sanal makine ile ilgili tüm bilgilere ulaşabiliyoruz gördüğünüz gibi Özellikle bu public ip kısmı ve buradan kopyalaya bildiğim public ip kısmı bizler için önemli çünkü biz bu makineye bu public ip sayesinde bağlanacağız. Yani bu makine dış dünyadan görünen ip si bu Arkadaşlar. status checks kısmında ise bu sanal makinenin durumunu görebiliyoruz altta çalışan fiziksel sonucu ve hyper vizör katmanında bir sorun oluşursa bunların hepsine buradan bilgi edilebiliyoruz şu anda sistem status checks ve instance status checks ikisinde de herhangi bir sıkıntı yok Eğer sistemde bir sorun olsaydı yani hipervizör de burada bu sytem reachability Check passed

seçeneğini görmeyecektik bir sorun var diyecek de aynı şekilde Eğer bizim Bak sanal makinamızda bir sıkıntı olsaydı da yine bunun altında bu yeşil tag i görmeyecektik Bunun yerine bize gerekli uyarıları verecekti monitoring kısmında bu sanal makine ile ilgili temel verileri ulaşabiliyoruz. Örneğin cpu ıtilization yani işlemci ne kadar kullanılıyor şu anda discreet yani diske ne kadar veri yazılıyor bayt olarak ne kadar veri okunuyor bayt olarak bu tarz bilgileri Yani bütün monitoring bilgilerine buradan erişebiliyoruz biz bu sanal makineyi kurarken bunu detailed monitoring açmamıştık o yüzden bu cloud watch servisi bu sanal makineyi her 5 dakikada bir kontrol edip bu bilgileri alıp bu ekrana yazacak Eğer biz enable detail monitörlük dersek veya baştan seçmiş olsaydık bu 5 dakikalık süre 1 dakikaya düşecekti ve her her 1 dakikada bir biz bu verilerin güncel olanlarını alabilecektik son olarak da etiket kısmına bakalım daha önceden yani makineyi oluştururken oluşturduğumuz etiketler ve name etiketi ve departman etiketi ve value leri değerleri burada görülebiliyor. istersek bu etiketleri burada ekranda Yani bu Liste şeklinde de görebiliyoruz. Name yani isim etiketi zaten burada listeleniyor. Eğer siz name şeklinde bir etiket yaparsanız bu default olarak varsayılan olarak geliyor ama sizin isteğinize göre diğer etiketleriniz ve buraya ekleyebiliyorsunuz hemen buradaki butona tıklayalım ve gördüğünüz gibi sizin kendi yarattığınız tagler de burada eklenebiliyor ve ekledim bu sanal makinenin departman tag i ile etiketlendiği Etiket de burada gösterebiliyor şimdi bu tek bir makinede pek bir mânâ ifade etmeyebilir ama burada 100 tane sanal makineniz olduğunu düşünün bu tarz etiketler Bizim hayatımızı oldukça kolaylaştıracaktır. Şimdi artık makinamız hazır menüleri de tanıdık Şimdi makineye bağlanma sırası geldi ama bu noktada gelin küçük bir mola verelim ve bir sonraki bölümde makineye bağlanarak devam edelim.

Yanına Merhaba arkadaşlar Evet artık makinemiz hazır ve biz makineye bağlanması hazırız Şimdi makineye bağlanma sırası geldi öncelikle bu sistemlere Mac ve Linux makinelerden nasıl bağlanabileceği göstermek istiyorum Ben şu anda Windows bir sistem kullanıyorum ama Windows 2 sene önce Windows sistem for Linux adında Windows altında lüksel kullanmamıza izin veren bir yapı çıkardı bu Mac ve Linux sistemlerde bulabileceğiniz Terme'nin aynısı o yüzden bunun üstünden anlatacağım Öncelikle Az önce indirdiğim Daha doğrusu Bir önceki bölümde indirdiğim ki dosyasında C altında bir klasör Yarattım ki diye onun altına attım şimdi hemen geçeyim Umut ekranıma önceki o klasöre gitmem gerekiyor Sizlerde kendi terminal ellerinizden Bu key dosyasını Kaydettiğiniz klasöre gidin ben ruta düşünüyorum buradan cm'nin altında ceye Buradan da çiğ klasörüne gidiyorum Enes diyorum bakayım dosyam burada mı Evet ip eğitim nokta Pem dosyam burada arkadaşlar benim bu dosyaya Pem dosyasında bir hak vermem gerekiyor ki bu dosya ile bağlanabilirim edebiliriz oluşturduğum sanal makineye vereceğim komutla şu şu Die mod 400 ve dosya adı bu komut her seferinde girmiyoruz Arkadaşlar bu bir seferliğine bu dosya üzerinde gerekli Haklı vermem gerekiyor ve o onu gerçekleştirdiğim adım şimdi bunun hemen ardından artık ama bağlanmaya hazırım ve Öncelikle döneyim management console ve bunun ip adresini kopyalı Yım Çünkü bağlanırken Bu ip adresi üzerinden bağlanacak tekrardan terminal Ekranımda döndüm Şimdi vereceğim komutçu s j bu dosyanın adı hemen ardından ise to User et sanal Makinamın ip adresi gideceğim Arkadaşlar Hemen enter'a basıyorum bana bu fingerprint evlenmek isteyip istemediğini soruyor Yes diyorum Arkadaşlar şu anda Amazon üzerinde oluşturduğum sanal makineye bağlı bilmiş durumdayım içinde terminal üzerinden nasıl bağlanabileceği mizi gösterdim bunu exit ile çıkabiliriz bağlantıyı kesebiliriz arkadaşlar Windows da eğer Windows 10'un Business Pro 10 enterprise versiyonu kullanmıyorsanız Bu şekilde bir Linux şehrine ulaşma imkanınız yok Dolayısıyla bu sanal makineler bu şekilde terminal üzerinden bağlanamıyorsunuz ve bunun için bir tişört vardı araç kullanmamız gerekiyor Bunu da daha önceden size indirmiştik Bu aracın adı pati birazdan patik kullanarak Aynen Az önceki gibi sanal makine müze bağlanacağız ama öncesinde sorun bir işimiz daha kaldı pati pembe dosyalarını desteklemediği için bunu patenin anladığı dosyası haline çevirmemiz lazım onun için de kullanacağımız aracın adı yeni daha önceden indirdiğiniz puttygen dosyası açarak bu hem dosyasını dosyası haline çevireceğiz hemen bunu gerçekleştirelim Öncelikle patıcan

dosyasını buradan bir açalım no diyelim cenin altında koymuştum belki klasörün deydi burada olsaydı seçiyorum ve ip eğitim tam dosyasını seçiyorum şimdi diyor ki bunu düzgün şekilde successful import edebildim tamam diyorum ve şimdi yapacağım işlem şu arkadaşlar Sev private tıklıyorum diyor ki buna herhangi bir şifre girmedim istersek biz buna burada bir şifre gidebilirdik her seferinde bu şifreyi kullanmamız gerekir Bu dosyayı kullanırken ben bunu kullanmak istemiyorum Bir şifre girmek istemiyorum O yüzden Evet ile devam ediyorum şimdi bunu nereye kaydedeyim diyor aynı ki klasörüne ayeti eğitim. dikey olarak kaydediyorum Az önceki gibi burada arkadaşlar Tek seferlik yaptığımız bir iş Yani indirdiğimiz bupam dosyasını bir seferliğine Pink dosyası haline çevirmemiz gerekiyor ve bunu da birazdan pati üzerinde kullanacağız hemen bunu kapatalım cünüp Hadi yarışalım patiği açtım burada tekrardan Az önceki gibi iyisi to 500'dür diyorum ediyorum ip adresimin kopyala yiyeyim hemen ardından da buradan sesi Diyorum ki klasörün içinde bakın gördüğünüz gibi ip eğitim Hatip private şeklinde bir dosyam var az önce yarattın onu seçiyorum tekrar sayısına dönüyorum her seferinde Bunları tek tek girmeyelim diye bunları edebiliriz adında kaydediyorum ve Open diyorum yine Az önce terminalde verdiği gibi bize bir uyarı veriyor Evet ile devam ediyorum ve arkadaşlar Tebrikler Windows üzerinden de sanal makine arıza bağlanmayı başardınız sana makinamız üzerinde çalışmaya hazırız hemen ben bundan da çıkıyorum Çünkü burada biraz daha büyük daha rahat bir gece için Umut üzerinde çalışmak istiyorum tekrardan sanal makine mı bağlanıyorum Arkadaşlar şimdi ilk gireceğim komut tuşu sudo su Bu sayede bu sanal Makine üzerinde Root yani kök kullanıcı izni aldım ve her türlü işlemi yapabileceğim gideceğim komutta Şu gelin yam Apti deneyelim bakalım ne olacak önül Güven innopack Smart for update niye biz böyle Hiçbir abdeyt bulamadık niye hiçbir güncelleme bulamadık sunucu üzerinde Çünkü bir önceki bölümden hatırlarsanız biz bu sanal makineyi oluştururken Advanced vites kısmında buna bir tane sıkıp girmiştik önceki bir şey ben yazmıştık arkasından 1005 demiştik hemen Altına da yan boşluk abdest boşluk 5 ye komutunu vermiştik Bu sayede bu sanal makine ilk oluşturulduğu zaman bu afetleri yani komutu girerek muhabbetleri yaptı o yüzden biz şu anda girdiğimiz zaman gördüğünüz gibi ben yine bu komutu verdim ya madde ettirdin Bana herhangi bir atlet çıkarmak Çünkü zaten afetler yapılmıştı İşten kaçta bu Advanced böyle avantajları var Onu şekilde düşünebilirsin Canım bir sanal makineye atıyorum o sanal makinede bir sürü servis duracağım bir sürü işlem yapacağım bunları bir sıklıkta haline getiriyorum o 0 tane getirdiğim Komutları kısmına giriyorum makine oluşturulduğu zaman bütün servisleri kurmuş oluyor ve ben tekrar makineye gidip şu anda Birazdan yapacağım şekilde tek tek işlemler yapmak zorunda kalmıyorum büyük bir kolaylık ve büyük bir avantaj sağlıyor Biz Hemen işlemlerinize başlayalım ne demiştik Bu makineyi Bir web sunucu olarak kuracağız öyle planlamıştık gelin O nedenle bu sisteme Bir web sunucu yazılımı yükleyelim yükleyeceğimiz yazılımı son dönemde oldukça popüler olan engine.exe yazılımı Normalde bunu şu şekilde yam pardon yan installs ngen.exe şeklinde kuruyorum önceden böyle kuruyordum fakat edebiliriz bu yeni bu dağıtımı değiştirdi ve farklı bir yere aldı O yüzden bu komut yerine şu komutu girmem gerekiyor Amazon 5 Linux bash extras Install diyorum ve engine.exe 1.12 şu anda engine.exe yazılımını eline sunucu yazılımını buraya yüklüyor hemen ye ile onay veriyorum öncelik yüklemesi başarıyla gerçekleştirildi hemen servis engine.exe Start diyeyim ve engine.exe servisini çalıştırayım ekranda bir temiz diyeyim hemen ardından da bu engine.exe servisi her bilgisayar başlatıldığında otomatik olarak çalışsın diye çek konfix komutuyla engine.exe komutuyla Bunun her Bilgisayar yeniden başlatıldı zaman başlamasını sağlar sağladım şimdi bakalım acaba engine.exe yüklememiz Düzgün şekilde gerçekleştirildi mi Bunu anlamanın çok kolay bir yolu var Hemen bu sunucumun public ip sini kopyalıyorum geliyorum HDP: sunucu ip adresi ve Welcome to nginx on Amazon Linux ekran ulaştığımda anlıyorum ki engine.exe sunucu sunucu yazılımını bu sistemi kurabildim ve düzgün şekilde çalışıyor Hatta gelin bunun gerçekten bizim suçumuz dan geldiğini teyit etmek için bu sayfada bazı değişiklikler yapalım bakalım gerçekten de bizim sunucumuz bu mu ve biz bu mesajı Buradan mı alıyoruz tam olarak emin olalım bunun için tekrardan terminale geri dönüyorum hemen bunu bir tekrar tam ekran ekonomi temiz diyeyim şimdi BİM editörümle Ulus ardon users folder onun altındaki şehir

Fold altındaki engine.exe altındaki HD nin altındaki index html dosyasını edit edeceğim Yani benim Az önce buradan gördüm bu dosya Benim sunucumun üzerindeki bu dosya arkadaşlar Ve ben bunu şimdi birteks editörle edit edeceğim hemen gelelim Bunu mesaj Şu tespit kısmını değiştireyim İyi ya basıyorum hemen insört moda giriyorum mesela bunu ait eğitim eğitim HD hemen: yazarak çıkarım tekrardan Browser Ama geri döneyim ve kontrol S5 ile restart yani refresh edeyim gördüğünüz gibi ip eğitim olarak değişti ve ben bu Sunucunun yani benim şu anda bağlandığım Sunucunun gerçekten benim suçum olduğuna emin olmuş oğlum Arkadaşlar bu uygulama kısmında son göstermek istediğim bir şey daha kaldı hatırlayacağınız üzere l'm kısmında uzun uzun rollerden bahsetmiştik bu sanal makineye oluştururken de daha önceden yarattığımız ve iyisi tuz sanal makinelerin güç kaynaklarına erişmesine izin veren rolü bu makineyi a tanıştık bakalım bro çalışıyor mu hemen dönerim terminalde Ekrem'i temizleyelim tekrardan edebiliriz diyelim etüt diyelim Allah diyelim Evet edebiliriz etüt Enes komutuyla bu Ben şu anda İsveç'teki bakır larımı listeye bildim Şimdi tekrardan Ben konsolo geri dönüyorum actions kısmından hemen altından da etek replace l'm role geliyorum bunun altında Bakın şu anda isteği-tire atıldı Ben buradan no rolü seçiyorum Play diyorum Ateş diyorum bunu Close diyerek tekrardan terminale geri dönüyorum Az önce verdiğim komutu tekrar giriyorum u sefer ulaşamadı şimdi bakın hemen altında ne diyor anne waterlogged credentials you can't konfigüre credentials bayrağı nikeit.com söylüyor Biz bölümünde ne yapmıştık lazerden hemen önce komut satırını anlatırken kendi bilgisayarımızda evde konfigüre komutunu girmiştik hemen ardından da Access Basic girip hemen ardından da hangi ricin üzerinde çalışmamız çalışmak istiyorsa koruyacağını seçip output format olarak ceysinin girerek Biz kendi bilgisayarımızda bu konfigürasyonu yapmıştık ve kendi bilgisayarınızdan bize de biliyoruz kaynaklarına sigaraya ulaşabiliyor arkadaşlara şöyle düşünün benim böyle 300 tane sanal makinem var bir hepsinin üzerinde bu konfigürasyonu yapmak çok zor olacaktır bu tahmin edebileceğiniz gibi zor bir işlem İkisi de ben her seferinde o bilgisayarlarda bu evde Bu kez konfigüre komutu girerek Secret ve Access Aydın MYO bilgisayar bilgisayar bir şekilde Ele geçirilirse Ne olur benim evde Buraya üzerinde admin yetki ve sahip olan kullanıcının Access Haydi isteği ve Secret ki başkaları tarafından ele geçirilmiş olur bu yüzden bizim bu tarz verileri bu bilgisayarlarda saklamamız gerekiyor Bunun önüne geçebilmek için bu rol kısmını rol dediğimiz şeyi yarattı ve rollerde Tam da bu işe yarıyor ben bu sanal makineye bir rol atıyorum Bu rol bu şekilde diğer edebiliriz kaynaklarına erişebiliyor ve benim her seferinde kendi kullanıcı Access Aydın ve Secret kimi girmeme gerek kalmıyor yani rollerin amacı bu tekrardan geri dönelim tekrar rolü atmayalım stx gelelim l'm rol ateşe gelelim tekrar seçelim rolümüzü tekrar Play diyelim ve tekrar terminale dönelim de bir kere daha girelim ve gördüğünüz gibi şu anda ben buna bağlı bildim tekrardan yapacağımız son bir işlem daha kaldı Arkadaşlar daha önce etüt bölümünde kendimize güzel bir web sayfası yapıp bunu etüt üzerinden yayınlamıştık dosyalar orada duruyor Gelin bu bestesini yeni yarattığımız sanal makineye taşıyalım Öncelikle şu folder abi Gidelim Arkadaşlar bunu bir ekranı bir temizleyeyim kılıyorlar hemen diyelim OSB altında şehir altında engine.exe altındaki HDM klasörüne HD Elif'le Bakalım burada hangi dosyalar var Gördüğünüz üzere 404 hdm-555 xhtml var bu ara index sileyim bize az önce Web sayfasını gösteren dosya ya da Web sayfasında ana dosyası sildim tekrardan Evet diyeyim daha şu anda bu dosya buradan silindi bunu sildim çünkü biz İsveç'te bulunan web sitemizi buraya taşıyacağız bakalım mp3do dosyalar duruyor oğlumun Neredeler hemen bunun için edebiliyor es3 Elles ve bakımın ismini giriyorum xhtml Klas jpg dosyalarını burada duruyor ve bize de bizim de ihtiyacımız olan dosyalar Bunlar Peki bu dosyaları Gelin bu makineye kopyala Yalım edebiliyorum diyelim es3 diyelim yani kop kop i diyelim Nerede ki et 3'teki: staj batındaki index html dosyasını noktaya Yani buraya kopyala damat edildi hemen bunun bir benzerini de klavye PG için yapalım Cloud jpg dosyasını da buraya kopyala Yalım Böylece bize tuş baktığımızda ki bu dosyaları bu serinin altında şeyinin altında engine.exe ne altındaki HD Emel klasörüne kopyaladık buradaki index html dosyası da artık bizim yeni Web sitemiz oldu ve bakalım acaba bu çalışacak mı dönelim brosuru tekrardan kopyalı Yalım sanal makinemizin ip adresini http sanal makinamız Tebrikler arkadaşlar S3 bakımımızda Duran dosyaları Kendi yarattığımız sanal makinemizin

taşıyarak 1 adet web sitesi yarattık ve bu web sitesini buradan yayınlamayı başardık şimdi şu ana kadar benimle birlikte bu adımları gerçekleştirip Bu siteyi buraya taşı duysanız sizleri gerçekten tebrik ediyorum gerçek anlamı ile Bulut dünyasına adım atmış buluyorsunuz kendinizi ödüllendirmeye hak ettiniz dinlenin ve ardından bana bir sonraki bölümde katılın daha öğreneceğimiz çok şey var arkadaşla

EC2-3

merhaba arkadaşlar artık iş dünyasına iyice ısındığımıza göre EC2 ile ilgili kalan konuları da bitirelim ardından tekrar konsolu dönerek uygulama kısmına devam edeceğiz son bölümde sizlere ilk anlatmak istediğim konu elastik load balancing servisi. bu servise daha önce tüm servisleri anlattığımız bölümde değinmedim çünkü ayrı tek başına kullanabileceğiniz bir servis değil. elastik load balans ya da kısa adıyla elb EC2 ve konteynerlerle kullanabileceğimiz onlara ek bir servis. O nedenle EC2 altında anlatıyoruz adından da anlaşılabilir gibi elb aws nin yönetilen yük dağıtım servisi. bize aws dünyasında yük dağıtımı yapma imkanı sağlıyor gelin büyük dağıtımı işini bir önceki uygulama bölümünde kurduğumuz web sunucu üzerinden anlamaya çalışalım. Şimdi biz bir önceki bölümde kendimize bir web sitesi barındırmak için bir adet sunucu kurduk ve dosyalarımızı bu sürücüyü yükleyerek ilk web sitemizi ayağa kaldırdık bunun bizim için önemli bir web sitesi olduğunu düşünelim. Örneğin bir web mağazası kurduk bu mağazanın da 7/24 saat ayakta kalmasını istiyoruz. Peki şimdi şu sorulara birer cevap bulalım diyelim ki bu kurduğumuz sanal makinede bir sorun çıktı ve bu sanal makine bir şekilde kullanılamaz hale geldi. web sitemiz de bu makinede durduğu için doğal olarak web sitemiz de erişilemez hale geldi. Haydi diyelim bir sonraki uygulama bölümünde göstereceğim şekilde bu makinenin size snapshot unu almıştık ve buradan makineyi tekrar ayağa kaldırmayı başardı en az 1 saat zaman harcadık kurulumu ayarı derken bizim web sitemiz en iyimser tahminle bir saat kapalı kaldı. bu yöntem Gördüğünüz üzere 7 gün 24 saat çalışma mantığına uyumadı. bundan kaçınmak adına ne yapabiliriz mantıklı olan bu sanal makinenin aynısından bir tane daha kurup tüm ayarları önceden yaparak hazır halde bekletmek ana makinada bir şey olursa trafiği bu yeni makineye almak e peki Her seferinde bir o Makina bir bu makine derken sürekli duruma göre manuel müdahalede mi bulacağız Bu da büyük bir iş yükü çıkardı. Gördüğünüz üzere ortada bir sorun var bu sorunumuzun bir yüzü. Bir de şöyle başka bir senaryo düşünün. Ben az önce sanal makinamı t2 mikro olarak kurdum ve öyle çalıştırdım ama bir zaman sonra baktım ki benim web sitemi trafik alıyor Ve bu sanal makinenin kaynakları tamamen tüketildi web siteme gelen kullanıcılar sistemin yavaş olduğundan şikayet etmeye başladı hemen müdahale ettim ve daha güçlü bir sanal makine kurup sistemi ayağa kaldırıp trafiği bu yeni makineye yönlendirdim. Yine bir sürü manuel iş yaptım ama gün geldi sistem büyüdü ve bu makine de yetmemeye başladı gittim daha da güçlü bir makine kurdum yine zaman geçti daha da güçlü su dağıtımı oldu yine kurdum derken gördüğünüz gibi iş çıkırından çıktı. Çok laf ettik ama özetle şunu demek istiyorum Tek bir makineden oluşan servisler kurmanın birçok sıkıntısı var. Bu nedenle hem yedeklilik için hem de kaynak ihtiyacı adına benim aynı hizmeti birden fazla sistem üzerinden vermem gerekiyor işte bunu gerçekleştirebilmemi sağlayan servise load balancer yani yük dağıtıcı diyoruz. arkadaşlar load balancer dediğiniz temelde bir ağ cihazı gelen trafiğinizi artık direk sanal makinelere yönlendirmek yerine load balancer cihazı üzerine düşürüyorsunuz cihazı da sizin belirlediğiniz kurallara göre bu trafiği istediğiniz sayıda cihaza çeşitli kurallara göre yönlendiriyor. load balancer temelde iki komponentli oluşuyor öncelikle listener yani yani dinleyici dediğimiz komponent yapılan ayarlamalara göre belirli bir Port üstünden gelen trafiği dinliyor daha sonra load balancer bu gelen trafiğin nereye hangi kurallara göre göndereceğini konfigürasyonunda bakıyor ve bu konsantrasyonda belirtilen kurallara göre ilgili sunucuya yani hedeflere bu trafiği yönlendiriyor aws dünyasında bu işleri elastik elb bağlantı servisi ile yapıyoruz elb bir temelde üç farklı elb bağlantı servisinden oluşuyor klasik elb en eski load balancer servisi 2009 senesinden bugüne hizmet veriyor ve tüm temel TCP ve http Https web tabanlı load Balancing işlerimizde bize hizmet veriyor 2016 yılından itibaren ise aws klasik yerine geçecek application load balancer ve Network load balancer olmak

üzere 2 yeni servis daha duyurdu. Network load balancer Network terminolojisinde layer for dediğiniz seviyede Çalışıyor yani gelen paketin içeriğine bakmıyor ve üzerindeki kurallara göre temel yönlendirme yapıyor application load Balancer ise layer seven da yani uygulama katmanının da çalışıyor gelen paket içeriğini okuyarak buradaki bilgilere göre de yönlendirme yapabiliyor bunlar ağ Temelli derin konular biz sadece bu aşamada şunu bilelim. Eğer temel tsp tabanlı port üzerinden basit ve hızlı bir load balancer ihtiyacımız varsa bu Network load balancer Ama yok işimiz http https tabanlı ve interaktif web sayfası mobil uygulama, konteyner ve benzeri daha karmaşık işler ise burada kullanacağımız şey application load balancer servisi. bu elb servisiydi. gelelim altında Anlatacağımız diğer bir hizmet olan AutoScaling e. Az önce rot balans ince neden ihtiyacımız olduğunu anlatırken ne demiştik tek makine ile birçok sorunu bozuyor Bu nedenle makine sayısını çoğaltmak lazım Peki ben bu makine sayısını çoğaltma veya eksiltme işini otomatize etmek istersem nasıl mı şimdi gelin yine bir web mağazamız olduğunu düşünelim Bir load balancer kurduk ve hemen arkasında 3 sanal makine yerleştirip işimizi Hallettik ama zaman geldi bu makinaların üçü de kaynaklarını tüketti ve benim 4. Hatta 5. makine kurmam gerekti gittim bu işi manuel hallettim yine büyüdü 6. Makina gerekti gittim Yine manual hallettim bir sürü iş çıktı şimdi hadi bu ayda bir iki defa olsa Eyvallah hemen aksiyon alırım ama şöyle bir senaryo düşünün diyelim ki ben normal zamanda 3 sanal makine ile işimi hallediyor ve Gayet düzgün çalışıyorum ama gün içerisinde öğlen saatlerine doğru sistemde bir yoğunluk oluşuyor ve bu iki makine atıyorum saat 13 ile 16 arasında bana yetmiyor ya da arada bir kampanya yapıyorum hurraaa herkes siteye alışveriş yapmaya geliyor ve kampanya süresince Bana ek sistem işte arkadaşlar aws dünyasında tüm ek yükü karşılamak adına sisteme yeni sanal makine ekleyip yeri geldiğinde sistemden sanal makine çıkarabileceğiniz servise auto scaling servis deniliyor auto scaling Tüm bu ekleme çıkarma işlemini sizin adınıza otomatik hallediyor ve böylece siz size gelen talebin boyutuna göre sisteminizi o talebe cevap verecek şekilde genişletip talep azaldığında da otomatik olarak herhalde daraltabiliyorsunuz auto scaling ile belirli kurallar belirleyip bu kurallara göre sistemde yeni kaynaklar yaratıyorsunuz. Örneğin Eğer benim sana makinaların 5 dakika boyunca %90 dan fazla cpu kaynağı kullanırsa git ortama yeni bir sanal makine Ekle ve bu makine diğer makinelerin cpu kullanımı %30 un altını düşene kadar çalışsın diğer makinalar %30 altına düşerse bu sanal makineyi sil gibi kurallar yaratarak tüm süreci Otomatize etmenize imkan sağlıyor bir sonraki bölümde uygulamalı olarak göreceğiz. hemen o bölüme geçmeden önce ile ilgili son bir şeyden daha placement group lardan bahsetmek istiyorum gerçek hayatta sürekli kullandığımız bir özellik olmasa da aws hem sınavda bu konudan soru soruyor olması nedeniyle hem de bazı durumlarda lazım olduğundan dolayı bunda teorik kısmını son konusu olarak anlatmak istiyorum. yine Senaryo üzerinden anlatalım EC2 dediğimiz şey binlerce fiziksel sunucu üstünde koşan sanal makineler. Sen EC2 üstünde kendine 2 tane sanal makine yarattın diyelim Bu makinelerden bir tanesi atıyorum 4 numaralı fiziksel sunucu üstüne oluşturulurken diğeri yine atıyorum 4000 numaralı fiziksel sunucu üstünde oluşuyor. Bunlar yine de birbirlerine hızlı ağ bağlantıları ile bağlı ve birbirleriyle hızlıca haberleşe biliyorlar ama senin bazı uygulamalarda Cidden bu iki makinenin birbirleriyle çok hızlı ve çok düşük gecikme süresi ile haberleşmeleri gerekiyor. Yani low latency ve high-throughput aws sana bunu placement Grup dediğimiz şeyle ağılıyor. sen bir placement grup yaratarak enhance network destekleyen EC2 tiplerinden sanal makineleri aynı anda bu placement grup içerisinde çalıştırırsan aws bunları aynı fiziksel makine ya da aynı class üzerindeki fiziksel makinelerde çalıştırarak 10 gigabit Per second hızında kesintisiz haberleşmelerini sağlıyor buna classter placement Grups deniliyor burada aws in sana dikkat etmen gereken üç uyarısı var. bir placement grup içerisine çalıştıracağın EC2 instance iplerin aynı olsun yani m4Xlarge c4xlarge lardan oluşan bir grup yapma hepsi m4xLarge olsun bu işi tek bir launch isteğiyle yap yani önce 4 makine çalıştırıp ardından 3 tane daha ekleyeyim deme 7 makinelik bir park kur ve Hepsini aynı anda gruba placement ekleyerek çalıştır isteği tek seferde hallet. 3 Sen bir placement grup içerisindeki bir sanal makine stop edip yeniden start edersen bir ihtimal aynı yerde hata vermeden çalışır fakat hata da verebilir Hata alırsan tüm makinaları stop et yeniden başlat ve böylece aws bunları yeniden bir araya

toplayacaktır. bu 3 şartı bilelim Çünkü sınavda bundan bir tane soru gelebilir bunun yanında senin bu ihtiyacının tam tersi Bir ihtiyacın da olabilir yani bir uygulamanın koştugu 2-3 tane kritik sunucum var Sen bunları bir ihtimal aynı fiziksel sunucu üstünde çalışsın istemeyebilirsin hani fiziksel bir sunucuya bir şey olursa aynı anda hepsi birden aşağı inmesin işte bunu da aws sana spread Place Grup ile sağlıyor buda yeni bir özellik geçen sene devreye alındı.

EC2 Uygulama-2

Merhaba arkadaşlar şimdi hemen kolları sıvayalım ve konsola bağlanarak öğrendiklerimizi uygulayalım ilk uygulama bölümlerinde linux bir sanal makineye atmıştık dedik Şimdi de kendimize bir adet Windows sanal makine yaratalım Bunu hem Windows ortamlar nasıl yaratılıyor ve bağlanılıyor bunu görmek için yaratacağız hem de birazdan ebs disklerle oynamaya başlayacağız yarattığımız ebs volume leri linux makineye bağlanarak komut satırından görmek yerine Windows sistemde grafik ekrandan daha rahat anlayabileceğimiz için bu alıştırmayı Windows üzerinde yapalım. Haydi bir Windows instance yaratalım ama öncesinde gelin daha önceki uygulama bölümünde yaratmış olduğumuz Security gruba Windows sistemlere bağlanırken kullandığımız Remote desktop Protocol un Port olan 33 89 u ekleyelim Bu sayede mevcut Security grubumuzu bu sanal makinede de kullanabileceğiz. hemen buradan Network and Security nin altından Security gruplara geçiyorum bakın bu bizim yarattığımız Security gruptu EC2sec grup buna geliyorum şöyle biraz ekranı yukarı çekeyim Inbound a geliyorum ve Edit e basıyorum burada Bizim daha önceden eklediğimiz 22, http 80, http 433 portları listelenmiş durumda bunları hem ip V4 Hemde ipV6 olarak eklemiş biz aslında bir bunları tek bir satırda eklemiştik ama aws bunları alt alta ip v4 ayrı ip v6 ayrı şekilde bölüyor Şimdi biz buraya bir kural daha ekleyeceğiz hemen buradan listeden kastım tsp Protokolü seçiyoruz arkadaşlar Remote desktop 33 89 TCP protokolünü n33 89 portu üzerinden çalışır. O nedenle Biz de TCP 33 89 u seçtik. Source olarak yani kaynak olarak da bunu yine kolaylık olsun diye 0.0.0.0./0 yani dünyanın her yerinden bu Porta erişilebilirliği şeklinde yapıyoruz Hatta gelin buna Bir açıklamada girelim Windows için rdp portu diyelim ve Save basalım. Böylece security grubumuzun işte 33 89 portunu ekledik bu Security gruba atadığımız adımız sanal makineye 33 89 portu üzerinden bağlanabiliriz Şimdi geldik sanal makineyi yaratma kısmına buradan instance kısmına geliyorum. ve Launch instance diyorum şimdi linux bir makine yaratırken Amazon Linux 2 ye ami seçtik. Windows bir makine yaratırken de Windows yüklü ami Windows'u seçmemiz gerekiyor o yüzden Mesela buraya Search e Windows 2016 yazıyorum ve listeden hemen base image nerede burada Microsoft Windows Server 2016 base image ile devam ediyorum. yine T2 mikro en düşük konfigürasyonu ve ücretsiz olanla devam ediyorum bir tane sanal makine istiyorum diğer herhangi burada değiştireceğim iam role gene atayım EC2S3fullaccessiam önüne atayım diğerlerini default olarak bırakıyorum. Storage yani disk ekleme kısmında burada Root volümü ekliyor otomatik olarak bununla devam ediyorum Şu anda bu ekrandan eklemeyeceğim birazdan bunların hepsini göreceğiz yeni diskler ekleyeceğiz buraya tag ekle diyor hemen bir tane isim ilk Windows olsun bunun adı da. Security grubu ayarla diyor bu sefer yeni bir security grubu yaratmıyorum mevcut Security grubumuz seçiyorum Grup Az önce buna 33 89 portunu da eklemiştir Bu sayede biz bu Windows makineye Remote desktop Protocol üzerinden bağlanabiliriz. review and launch a basıyorum bakıyorum bütün ayarlarım doğru mu bütün yapmak istediklerim burada seçilir mi Evet launch diyorum bu ekranda şimdi bize şeyi soruyor keypair. yeni bir keypair mi yaratmak istiyorsun yoksa mevcut bir keypair mi kullanmak istiyorsun diyor Biz bir önceki uygulama bölümünde bir keypair yaratmıştık onunla devam edeceğiz her seferinde bunu kullanacağım Ama bu noktada aws bize size şunu soruyor diyor ki sen bunu daha önceden yaratmışsın bunun private olanını ile indirmişsin bu dosya hala Sende mi buna emin ol yoksa bu makineye bağlanamzsın diye bir uyarı veriyor Ben de diyorum ki evet bu bende ben bunu erişe biliyorum O yüzden Bu tip box işaretleyip launch instance diyorum şimdi Windows sanal makine yaratılıyor. Bu biraz sürecek Arkadaşlar bunun üzerine de bir 5 dakika beklemeniz gerekecek. Çünkü Windows'un ilk şifresinin oluşturulması bir 5 dakika kadar sürebiliyor o

yüzden bu noktada videoyu durduruyorum makine oluşturulup bu Windows şifresini alabilecek duruma geldiğimiz zaman dönüp devam edeceğim. Evet makine yaratıldı running e geçti. Üzerine de 5 dakika bekledim şimdi actions kısmından Get Windows password e basıyorum burada bana Pemfile ımı daha önceden indirdiğim key dosya mı soruyor. Browse ile bunu c nin Altında key klasörüne atmıştım buradan seçiyorum click Password diyorum. Evet username password mi burada bana indirme ve görme imkanı sağlıyor. Hemen bunları böyle kopyalayın ve public dns i de de kopya alayım buradan hemen new file diyeyim buraya yapıştırıyım elimde dursun bu noktada arkadaşlar Remote desktop client ını açıyorum Windows için Windows tuşu ve R ye basıp Çalıştır geçiyorum buradan mstc yazıyorum tamama tıklıyorum ve uzak masaüstü bağlantısı açıldı mac sistemlerde de bunu Apple App Store'dan indirebiliyorsunuz onu hemen hatırlatalım. public dns imi buraya yapıştırıyorum kullanıcı adı administrator diyor. hemen bağlan diyorum şifremi soruyor şifremi de kopyalıyorum Şifremi de yapıştırıyorum sertifikaya güvenmek istiyor musun diyor Evet diyorum böylelikle Windows yarattığım Windows sanal makineye ilk defa bağlanıyorum. Bu ilk bağlandığımda biraz ayarları yapması gerekiyor o yüzden biraz zaman alıyor sistemin ayağa kalkması için biraz beklememiz gerekiyor. Evet sistemimiz açıldı buradan hemen Windows tuşuna basıyorum. Buradan da Server Manager a gidiyorum lokal server a geliyorum hemen Tools un altında Computer management a geliyorum. Buradan da disk management a geliyorum şimdi bu noktada Arkadaşlar bu sanal Makineyi bir simge durumuna küçültelim ve tekrardan aws konsolumuza geri dönelim hemen burada elastik block storage yani ebs kısmına gelelim burada göstermek istediğim iki tane şey var ve bunların üzerinden bu bölümde devam edeceğiz birincisi volumes İkincisi de snapshot ebs aws dünyasında EC2 sanal makinelerde kullanabileceğimiz diskler oluşturup yönettiğimiz servisin adı volüm kısmında volumes kısmında mevcut sanal makinelerimizde bağlı olarak kullanabileceğimiz tüm diskleri görebilir ve ihtiyacınıza göre yeni disk yaratabiliriz. Bu ilk yarattığım linux Sunucunun root device Bu da ilk yarattığı Windows Sunucunun root device Zaten makinelerle beraber yaratmıştı burada yeni bölümler yaratıp yeni ebs dikler yaratıp bunları makinaları Ata'ya biliyoruz hemen altında volumesün hemen altında snapshot göreceksiniz snapshot kısmında bu disklerin Snapshot adı verilen anlık kopyalarını yönetebiliyoruz Arkadaşlar snapshot bir ebs diskin o andaki kopyasını alıp S3 üzerinde saklamamıza yarayan yöntemin adı Bunu üç temel ihtiyaç için kullanıyoruz Öncelikle snapshotlar bize o diskin Ondaki kopyasını bir yere kaydetme imkanı verdiği için yedekleme amacıyla kullanılıyor bunun yanında Snapshot alıp daha sonra bundan ami yaratma imkanımız oluyor. böylece bir sunucu oluşturup üstünde istediğimiz tüm ayarları yaparak bunun snaps Atını alıp daha sonra bundan oluşan bir ami yaratıp aynı makinenin birebir kopyasını oluşturabiliyoruz bunu birazdan kullanacağız son olarak da alınan Snapshot lardan volümler oluşturuyoruz ve bunu başka sistemlere bağlayıp kullanma imkânına sahip oluyoruz birazdan hepsini göreceğiz ama Öncelikle volümleri bir bakalım mevcut ekranda EC2 sanal makinelerimizde yaratılırken yaratılan bölümlerini olduğunu söylemiştik şimdi gelin bir tane yeni bir volüm yaratalım ve bu volümü Windows makinemize bağlayarak bakalım o Windows makine üzerine bu volüme ı nasıl işleyebiliyor volume e nasıl ulaşabiliyoruz create volüm diyorum Evet yeni bölüm yaratma ekranına açtık ilk olarak bizden Volume types seçmemiz istiyor bunların hepsini Theory bölümünde görmüştük cp2, io1, hdd, true put of the magneting şekilde çeşitli disk çeşitleri var. disk şekillerimiz var bir standart lipito ile devam edeceğiz hemen bunun altında ne büyüklükte bir riske atmak istediğimizi soruyor Biz Şimdilik 20 GB yaratalım ve hemen anlatılda availability zone seçimi var. arkadaşlar Kural bir birdir bir diski bir sanal makineye bağlamak istiyorsanız o Diskle o sanal makine aynı availability zone da olmak zorunda peki bir sanal makinayı bir avalibily zone a nasıl atıyoruz veya bu zone ne biz bunların Network'ünü ağ altyapıların nasıl ayarlıyoruz de bunların tamamının cevabını arkadaşlar birkaç bölüm sonra vpc de göreceğiz Biz buradan servislerden EC2 ya yeni bir konsolda açalım bakalım bizim sanal makinamız availability zone daymış Windows sanal makine mız İrlanda 1-b availability zone da iuwest 1b yüzden bizim de bu volümü Aynen orada yaratmamız gerekiyor Yoksa biz bu volume ü sanal makineye atıyamıyoruz hemen altında thorught ve değeri varsa söylüyor bu general purpose ssd nin herhangi bir

Throughput değerini bize vermiyor şey yapmıyoruz onun seçemiyoruz veya ayarlayamıyoruz snapshot id Şimdilik geçiyorum birazdan Snapchat yarattığımız zaman göreceksiniz. Bakın burada önemli bir bölüm var encryption diye Biz Eğer bunu işaretlersek encrypt these bölümünü seçersek bu disk daha başlangıçta encrypt edebiliyoruz Şimdi ben Linux sanal makine yaratırken ne demiştim Linux sanal makine ilk yaratıldığı ekranda disk seçerken Bu disk non encrypt di ve biz bunu direkt encrypt edemiyorduk Arkadaşlar bir sanal makinenin ilk root device i ilk yaratıldığı anda encrypt edilemez. Bu bir kural daha sonra bunu encrypt edebiliyoruz yönteminde Göreceğiz. Ama daha sonradan yarattığınız ebs volumeleri direk başlangıçta encrypt edebiliyorsunuz bu aklımıza dursun Çünkü daha önceki bölümde de söylediğim üzere Bu bir sınav sorusu olabiliyor size bu ebs disklerin bu encrypt durumunu sınavda sorabiliyor. O yüzden bu Aklımızın bir köşesinde olsun hemen buna bir tek ekleyelim neyim diyelim Bu da ilk ebs volüm Olsun bunda da ve create volume ile volume imizi imizi yaratalım şimdi bu id li bir volüm yarattı. Close diyorum bakın burada daha gelmedi birkaç kez refresh etmem gerekiyor Evet hemen ilk refreshten sonra karşımıza çıktığı ilk ebs volume isimli volume yaratıldı ve state de durumu da available yani erişilebilir bakın seçtim Action a geldim bunu diyorum ki attach volüm Yani bunu bir yere Ata bir yere bağla seçtim nereye bağlayacaksın diyor bunu Hangi instance a bağlayacaksın diyor bir kere tıkladım listede benim bütün sanal makinaları mı bu availability zone daki Yani bunu atabileceğimiz zondaki sanal makinaları listeledi Bunu ilk Windows makineme atmak istiyorum Nereden atmak istiyorsun diyor bu kısmı Şimdilik Boş verin Arkadaşlar bu device kısmını bu karmaşık ve derin bir konu o yüzden bunları default unu Aynen böyle bırakıyoruz ve attach diyorum ve bu disk bu sanal Makine me atadım.Şimdi gördüğünüz gibi Inuse a geçti şimdi tekrardan Windows sanal makineme geri dönüyorum Buradan Remote desktop ile geri dönüyorum disk management a geldim ve bakın burada Gördüğünüz üzere disk bir unknown şeklinde yeni bir disk hemen disk management altında belirdi gelin buna sağ tuşla tıklayalım ve online yapalım. ilk yapmam gereken işlem bu windows'ta Bir disk taktığınız zaman bu Windows a online hale getirmeniz gerekir online diyorum bu Biz ki şu anda online hale getiriyor f5 ile refresh ediyorum hemen tekrardan sağ tuşa tıklıyorum Bu sefer de initialize istiyorum Bu diskini initialize etmek istiyor musun MBR mı GPT olarak mı etmek istiyorsun diyor. Biz bunu GPTi olarak initialize edeceğiz okey diyorum şu anda disk online aldık ve bu disk bu makineye Bir nevi tanıttık artık bu diskin üzerinde çalışmaya başlayabiliriz hemen gelin buradan sağa tuşa bu sefer bu 20 GB'lık bölüme sağa tıklayarak New Simple volüme diyeyim yani basit bir volüm yaratayım Next diyeyim ne kadarını kullanmak istiyorsun bu bölümün diyor tamamını kullanmak istiyorum diyorum. Next diyorum buna bir harf atamak istiyor musun diyor Evet D olarak bir harf okuyalım biliyorsunuz windows'ta her bölümüne C D E şeklinde harfler Atıye biliyorsunuz Next diyorum bunun formatlamak istiyor musun diyor disk ntfs olarak format diyelim bunada isim verelim ilk ebs volume diyelim Windows altında böyle gözüksün hızlı format yapsın hızlı formatlasın performans quick formatı seçiyorum Next diyorum finish diyorum. Evet arkadaşlar ilk diskim yani root device dışında ilk yarattığım disk hazır Ben bunu iyi bir sistemden Yarattım daha sonra bunu bu sanal makineye atardım de geldim Windows disk management in altında bu diske Windows ta tanıttım bakalım o diskimle çalışabilecek miyiz hemen disk pc'ye geleyim Gördüğünüz üzere ilk gibi ebsvolume şeklinde yaratılmış ve bağlanmış hemen mesela burada bir New text dosya yaratayım deneme diyeyim bakın dosya yaratabildim. kaydede bildim Yani diskim şu anda Windows sanal makine takılı ve ben bu disk üzerinde çalışabiliyorum şimdi dönelim tekrardan aws konsoluna. Döndük aws konsoluna arkadaşlar ebs in bize sağladığı en büyük avantajlarından biri de bu disk ihtiyacımıza göre modifiye edebilme şansımızdır bir diğer avantajı da bu disk ihtiyacımıza göre modifiye edebilme şansıdır. işte cloud un bir diğer şansı da bu normalde böyle bir disk kendi ortamımızda fiziksel olarak yaratmamız gerekiyor Yani siparişini vereceğiniz alacağız bunu takacağız ama Diyelim ki biz atıyorum 500 GB'lık bir disk aldık 500 GB disk zamanla doldu Ben bunun üzerine yeni ekleme yapmak istiyorum onu büyütmek istiyorum. Ne yapmam gerekir normal fiziksel ortam değilim disk almam gerekir. Ama sanal ortamda böyle bir yeni disk almama gerek yok Ben İstersem bu disk sanal olarak büyütemiyorum küçültebiliyorum işte gelin şimdi bu senaryoyu yapalım yani

sanal dünyanın nimetlerinden yararlanan ve bu yarattığımız volümü 20 gigabayt olarak her yarattığımız volümü büyütelim buradan Action sa geliyorum modifiye volüm diyorum bakın bu 20 ydi ve benim doldu diyelim Bu artık bana yetmiyor hemen bunu 40 yapıyorum modifiye diyorum Yes diyorum ve Close diyorum şimdi refresh edeceğim bir süre bakın State hemen sarıya düştü bekliyorum. Inouse da hala dönüyorum şimdi Windows sanal makine mı Ama burada Hala 20 GB Niye Çünkü daha Ayarlamasını yapmadım geliyorum buraya geliyorum disk management a Bunu hemen refresh ediyorum bugün gördüğünüz gibi 20 GB'lık diskim yani Windows altında 20 GB olan diskim bir 20 gigabyte daha eklendi ve 40 GB ama işlemlerin bitmedi 2 tane önümde yöntem var İstersem bunu yeni bir tane Simple volüm olarak yaratabilirim yani Aynen burada yarattığım gibi d yi hesabımda e yaparım Bunu yanına ekleyebilirim ya da gelir Bunu extend volüm derim yani genişlet dedim içindeki dosyaları silinmeden bu eklediğim 20 GB ı da buraya eklerim ve görüldüğü Zaten otomatik olarak eklendi ve bunu da 40 GB Total haline getiririm ve bu diski extend etmiş genişletmiş olurum. Bakın önceden 20 GB dı şu anda 40 GB oldu. Windows altında da bakalım şu eski 20 GB gözüküyor refresh ediyorum oldu 40 GB içine giriyorum hala deneme txt dosya burada duruyor yani ne bir dosya kaybına uğradım ne ekstra bir işlem yaptım sadece volume extend ederek bu Windows altında diskimi genişletilmiş oldum arkadaşlar Bunlar gerçek anlamda büyük avantajlar Normalde biz bu avantajları fiziksel ortamlarda yaşayamıyorduk gidiyorduk disk sipariş ediyorduk yenisine ekliyorduk bir sürü karmakarışık işlem yapıyorduk ama gördüğünüz gibi cloud da sadece 2 dakika içinde ben bir diski genişlettim onu da getirip sanal makinenin altında genişleterek hayatıma devam ettim bu gerçek anlamı ile büyük tekrardan dönüyorum konsolla Evet konsola geri döndük. Ama bu noktada Bu bölüme bir son veriyorum arkadaşlar 20 dakikalık süreyi doldurmuş olduk. Böylece Çünkü ortalama her bölümü 20'şer dakika olmasına özen göstermeye çalışıyorum o yüzden şimdilik burada bir mola verelim bir sonraki bölümde burada 3 tane daha yeni disk ekleyerek rate dediğimiz bir yapı var Oraya yapısı nedir Ve bunu nasıl burada yaratabiliriz ne bize ne avantajları var onları görerek kaldığımız yerden devam edeceğiz bir sonraki bölümde görüşmek üzere.

EC2 Uygulama 2 (Devam)

Hızlıca devam edeceğiz ve birazdan makineye 3 tane daha disk yaratıp bunu Windows makine ekleyeceğiz bu disklerle rate dediğimiz yapıyı tanımaya başlayacak ama öncesinde kısaca rate hakkında bir bilgi vermek istiyorum. bir süre Bir makinenin sanal bir makine olduğunu unuttun. İşte bu oluşturduğumuz Windows sunucu fiziksel makine ve ben bu makinenin üstünde fiziksel bir bağladım ve tüm verilerini buldukça kaydediyorum. Aradan 2 sene geçti ve disk bozuldu bu işte hazırladığımız senaryoda Böyle bir durumda ne olur yazınız tüm veriler gider ama gördüğünüz gibi üstünde bir tutmak bana veri güvenliği açısından sıkıntı yaratıyor. bu disklerle ilgili Başıma gelebilecek ilk sıkıntı Ama tek sıkıntı da değil teori kısmında sizlere bu Derslerle ilgili diye bir şeyden bahsetmiş bir ara o değeri olduğundan bahsetmiştim. Örneğin şu an hemen gelin create volüm diyelim bakın burada Gördüğünüz üzere seçtim diske göre iops bir değer veriyor ve bunun bir maksimumun değeri var arkadaşlar hatta kendime Pro Vision iops lu bir değer önceden kendini seçebildiğin bir iopslu kendim seçebildiğim seçersem de Gördüğünüz üzere bunun 64000a yoksa kadar iops olabiliyor maksimumu bu ama bunu da tabii ki disk boyutuna göre seçebiliyoruz vesaire yani şu Kısacası bu iops değerinin de bir maksimum değeri var bunu bilelim ama benim bu volüme üzerine kuracağım uygulama bu maksimum iops değerinden daha fazlasına ihtiyaç duyuyor böyle bir senaryo olduğunu düşünün ki bu senaryo az rastladığımız bir senaryo değil Özellikle Database veri tabanı uygulamalarında iops değeri yani yüksek hızlı diskler çok önemli hale geliyor ve sen bir diskte bir maksimum iops değerine sahipsin ve bazı uygulamaların o maksimum iops dan daha fazla iops a ihtiyaç duyuyor Böyle bir durumda ne yapacaksın Arkadaşlar bunun çözümü rate dediğimiz bir sistem birden fazla diski alıp aynı anda veriyor disklere bölerek yazabiliyorsun. İşte böylece IOPS değerini yükseltir biliyorsun yani Uzun lafın kısası birçok durumda tek bir

Diskle yarattığım çözümler işime yaramıyor ve sektör çok uzun zamandır bu sorunları çözmek adına rate dediğimiz bir teknoloji geliştirdi çok basit anlamıyla 0 1 5 10 gibi çeşitli spesifikasyonları olan birden fazla diski veri koruma ya da hız gibi senaryolarda bir arada kullanma imkanı sağlayan teknolojinin adı. Bizim bilmemiz gereken temelde üç rate çeşidi var rate o dediğimiz konfigürasyonda En az iki disk alınır ve veri bu disklerle eşit şekilde yazılır Bu sayede diyelim ki tek diskten 10000 IOPS performans alınırken 2 Diskle toplamda 20 bin IOPS luk bir performans elde edilir bu rate konfigürasyonunun tek sıkıntısı veri güvenliği olmamasıdır disklerden biri giderse tüm veri kaybolur 2. inceleyeceğimiz raid ise raid 1 diye adlandırılan çeşididir bunda yine en az 2 Disk bulunur birincil diske yazılan veri birebir aynı şekilde diğer diske yazılır bu sayede disklerden biri bozulsa bile verimizin birebir kopyası diğer diskte durduğundan veri kaybı yaşamayız. Son incelediğimiz raid ise raid5 diye adlandırılan ve çeşididir bunda ise ortamda en az üç disk bulunur RAID5 de veri hem disklerle bölünerek yazılır hem de Parite dediğimiz bir parça Bu parçaların kopyasını elde etmenizi sağlayacak veri ortamda dağıtılır Bu sayede disklerden bir tanesi gitse bile diğer iki diskteki pariteler sayesinde bu bozulunan pariteler sayesinde bu bozulan diskteki veriyi yeniden oluşturulabilir RAID5 ile Hem hızlı veri yazma imkanım hem de tek disk bozulduğunda oluşacak data kaybını düşünmezsiniz. Bunu anlatmamın Nedeni de şu arkadaşlar çözüm mimarı olarak bilmeniz gereken bir konu bu elzem Ama bunun yanında AWS sınavında Bununla ilgili şöyle bir soru sorabilir size bir senaryo verir ve bir uygulamayı EC2 de tutacağınızı Ama bu uygulamanın ebs in verdiği maksimuma IOPS değerinden daha yüksek IOPS değerine ihtiyaç duyduğunu bu durumda ne yapmanız gerektiğini sorabilir. Bu sorunun cevabı IBS diskler yaratıp bu disklerle RAID yapmaktır. gelin Bizde aynen bunu yapalım önce üç disk yaratalım Sonra makineden bunu RAID olarak ayarlayalım Evet hızlıca create volüm ile 3 tane de disk atıyorum 20 GB'lık olacak 1b ye bağlı olacak bu ilkinin adı birinci olsun. 20 1b de bir yere birinciye yarattım. Create bölüm diyorum 20 1b de isim olarak da 2. olan 2. yi de yaratıyorum 2. yi de Yarattım Create bölüm diyorum 20 GB 1 b de olacak şekilde ve ismi de 3. olan 2. diski de yaratıyorum. Diskler yaratıldı makineye atamaya hazır 1.den başlayarak attach volüm diyorum ilk windows sunucuma bunu Attach Ediyorum ikinciye ilk windows sunucuma attach ediyorum. İlk Windows sunucuma üçüncüyü Attach Ediyorum. Böylelikle 3 diski de sunucuya bağladım. şimdi sunucuya bağlanmak var. sunucuya bağlandın geldim buraya öyle aşağıya doğru çekiyorum bakın diskler hemen eklenmiş tek tek online çekiyorum online diyorum. Sonra yeniden hepsini 2 3 ve 4 numaralı diski yeni eklediğim diskleri initialize diyorum GBT olarak ediyorum hepsi onunla ile geldi bakın. sağ tuşla tıkladım artık burada bana Spanned, Striped, Mirrored RAID5 olmak üzere yeni seçenekler çıkardı. ne demiştik RAID5 bir volume yaratacağız. Bunu o yüzden bunu tıkladım Next diyorum bütün diskleri buraya ekliyorum disk 2 3 4 toplamda Normalde 60 GB kapasitesi olması gerekiyor ama sonuç olarak 40 GB'lık bir disk yaartıyor neden neden RAID5 böyledir arkadaşlar 3 disk düşünün disklerden 2 tanesinin toplam boyutu kadar 1 bölüm yaratabilirsiniz Çünkü o Üçüncü boyut Son 20 GB Parite dediğimiz şey için ayrılıyor Yani bu veri bu disklerle bölünerek yazılıyor örneğin disk 1 deki verinin bir benzeri de disk 2 ve disk 3 e de yazılıyor böylece disk 1 kaybedilirse disk 2 ve disk 3 den o parite ehsaplanıyor hesaplanarak kaldığı yerden çalışmaya devam edebiliyor Yeni bir disk taktığınız zaman bu nedenle 60 GB'lık bir bölüm 40 GB a düşüyor 20 GB'lık kısımda Parite için saklanır yani Parite oraya yazılıyor o yüzden biz 60 GB 20 şer GB lık 3 Disk seçip bunlardan RAID5 yapmak istediğimizde Total son sonuç 40 GB olacaktır. Next diyelim Buna da e-drive olarak atayalım Aynen bunu da formatlayalım ve volume label olarak RAID5 diyelim Next diyelim Finish diyelim. diyor ki bu diskleri Dynamic diske çevireceğim Çünkü sen bunlarla RAID5 yapıyorsun basic disklerle olmaz Evet diyorum işlemleri bekliyorum bitirmesini. Ve şu anda formatlanıyor ve sağlıklı şekilde RAID5 isimli 1 volume yaratabildim. Gelelim tekrar Windows tan buraya this pc ye gelelim arkadaşlar 40 GB'lık 3 adet diskten yarattığım yeni RAID5 volümün hazır. Ben artık burada yarattığım her türlü dosya bu alttaki 3 diske birden yazılacak ve ben bu disklerden bir tanesini kaybetsem bile ben bu test.txt dosyasına bu volume erişmeye devam edebileceğim işte RAID5'in bana sağladığı avantaj Bu arkadaşlar Böylece volume konusunu da bitirmiş olduk gelin instance lara geri dönelim ve bu Windows

sunucumuzda artık işimiz kalmadığı için bu Windows sunucumuzu buradan terminate edelim yani Windows sunucumuzu siliyoruz artık kapatılıyor shutting Down diyor bekleyelim O biraz zaman olacak Ben burada videoyu durduruyorum. Evet şu anda instance terminate e geldi tekrardan dönelim volumes e arkadaşlar ekranda görebileceğiniz üzere Ben bu makineyi terminate ettikten sonra bu makineye bağlı olan Root device otomatik olarak silindi bakın listede yok Root device ı gitti otomatik olarak sildi. Fakat benim daha sonradan bu ekrandan yarattığın create volume diyerek Yarattığım volümler şu anda burada duruyor Ne yapabilirim istersem ben bunları yeni bir sanal makineyi yaratıp onlara attach edip kullanmaya devam edebilirim. Bu aws nin default davranışdır arkadaşlar yani siz bir makineyi terminate ederseniz yani ana makineye bağlı bulunduğu disk silinir bunun içinde veriniz varsa tamamen silinir fakat daha sonradan yarattığınız ebs volume ler bölümler aynen devam eder siz isterseniz yeni bir sanal makine yaratıp bu bölümleri o sanal makineye bağlayıp oradan devam edebilirsiniz bu aklımızda bulunsun. Çünkü önemli bir durum fakat biz bunları kullanmayacağımız için bunları da sildim boş yere bunlar için para ödemeyelim Evet delete volume diyelim Diskleri de buradan temizleyelim. Evet volume leri bitirdik şimdi geçelim Snapchat kısmına. Arkadaşlar şimdi ben birazdan bir ami yaratmak için ilk bölümde oluşturduğumuz linux sistemi volümünün snapchat ını alacağım ve daha sonra bu snapchat ten ami yaratacağım Fakat bunun için ilk yapmam gereken iş makineyi kapatmak. Peki neden? arkadaşlar Snapchat dediğimiz işlem makinenin o andaki diski'nin birebir kopyasını almak. eğer o diske o anda yazılan bir dosya varsa ya da çalışan bir program ile ilgili işlem yapıyorsa Bu aldığımız snapshot o işlemlerinde o andaki halini yedeklemiş olur Eğer o işlem tamamlanmadıysa ya da diske yazma işlemi daha bitmedi ise bu da bu şekli ile yedeklenmiş olur ve daha sonra ben bu snapshot'ten ne yapmak istersem isteyeyim ister yeni volume yaratmak istersem de yeni sanal makine yaratmak her defasında sorun yaşarım bu nedenle bir diskin Snapshot alırken mutlaka ilk olarak o diske yazılmasını durdurmak gerekir bunun da en basit yöntemi o disk root device ise makineyi kapatmak Root değilse de yine ya makineyi kapatmak ya da diski makineden detach etmektir Bu sadece aws için de geçerli bir durum değil nerde Hangi ortamda bir diskin Snapshot unu alıyorsanız bu geçerlidir gelin O nedenle Öncelikle makinamızı kapatalım hemen instance a gidiyorum burada ilk sunucu seçili zaten Action a geliyorum instate state stop diyorum Ve yes stop basıyorum. Bu işlem biraz sürecek O yüzden videoyu burada durduruyorum. evet makinemiz durduruldu hemen buradan Snapshot diyerek Bu makinenin bağlı olduğu root device ın Snapshot alıyorum volume olarak bu ilk sunucuya bağlı root device ı seçiyorum Bana bir açıklama gir diyor açıklamayı da ilk sunucumuzun Root volume unun snapshot ı diyelim not encrypt söylüyor Root device'ların encrypt olmadığından bahsetmiştik Evet o yüzden bize not encrypt diyor birazdan bunun nasıl encrypt edileceğini yani root device ların nasıl encrypt edileceğini de göreceğiz Bunu hemen Bir tag atayalım name diyelim Bu da ilk Snapshot olsun bu Snapshot'ın tagı create basıyorum ve şu anda Snapshot'ın burada yaratılıyor bak 100'de 9 oldu Birazdan yaratılacak yine videoyu durduruyorum Evet yüzde yüze geldi ve Snapshot ımız yaratıldı. bu işi tekrardan özetlemek istiyorum aws ebs de yarattınız volumelerin o andaki birebir kopyasını alma işlemine snapshot diyoruz ve bu Snapshot s3'te tutuluyor fakat siz s3 de bir bucket a gidip bunun içinde bulamazsınız. Bütün işlemler bu ekran üzerinden yapılıyor ama alt taraftaki bunun Durduğu yer s3. Bu da aklımızın bir tarafında olsun. bakalım bu Snapshot ilgili ne işlemler yapabiliyoruz geliyorum Bu Snapshot ı silebiliyoruz Bu Snapshot'ten bir volume yaratabiliyoruz yani bundan yeni bir volume yaratıp bunu bir başka bir bağlayabiliriz yani o makinenin üzerinden bu diskteki verilere ulaşabiliriz ya da bundan bir imaj yaratabiliriz Yani bir ami yaratabiliriz ve Böylece bu ami içinde bütün özelliklerim olan işte engine.etc yüklenmiştik onun içine bir tane web sayfası atmıştık. Yaptığımız bütün değişikliklerin içinde olduğu bir ami yaratabiliriz o ami dan istediğim kadar sanal makine yaratabiliriz ve bunu kopyalayabiliriz. Başka bir region a veya bu region un altında şimdi bu kısmını göstermek istiyorum çünkü burada önemli bir özellik var bakın ne demiştik Root device ları encrypt edemiyoruz. bu önemli bir bilgi Bu sınavda da karşımıza çıkabilir Ama root device ları encrypt edebilmenin bir yöntemi var O da bu. öncelikle o root device dan root volume den bir Snapshot yaratıyorsunuz. daha sonra bu Snapshot'i kopyalarsanız yani

bu Snapshot'tan yeni bir tane Snapshot yaratırsınız arkadaşlar İşte o Snapshot o yarattığınız yeni Snapshotin encrypt olmasını sağlayabiliyorsunuz ve dolayısıyla bu yeni Snapshot tan bir volume yaratırsınız. Bu volume lü root device yani bu volumü root device olarak atayabilirsiniz İşte o makinenin Root encrypt oluyor Onu buradan sağlıyoruz Hemen onun dışında gelin buradan volume yaratmayı göstereyim bakın hemen bunun ne çeşit bir volüme olacağını soruyor büyüklüğünü ne olacağını soruyor hangi availability zone da olacağını soruyor soruyor Aynen Yeni bir volüme yaratır gibi sıfırdan bir volume yaratır gibi Snapshot dan volüme yaratabilmemizi sağlıyor Böylece bunun volüme type ını değiştirebiliyoruz ve bunun dışında da bizim de şimdi yapacağımız işlem create image Yani budan bir ami image yaratma hemen bununla devam edelim bakalım create image e basalım ve buna Diyelim ki bizim bu bizim base ami olsun Bundan sonra bunu kullanacağım Uzunca bir süre ona bir base ami adı verdik architecture için x86 64 diyoruz. Bu hepimizin kullandığı bugün bütün işlemci aileleri Intel amd nin temel mimarisi onu seçiyoruz. root device name i default bırakıyoruz Ram diski default bırakıyoruz description kısmı Mesela bu bizim ilk sunucudan yarattığımız ami diyelim daha sonra baktığımız zaman görebilelim burada tek değiştireceğimiz şey şu arkadaşlar virtualization type bunu ilk sanal makine yaratırken anlatmıştım. paravirtual ve hpm diye iki seçenek var paravirtual artık çok fazla kalmadı yani o yüzden bunu Hardware assisted virtualization olarak seçiyoruz Evet diğer burada bunu boyutunu istersek değiştirebiliriz volüme type istersek değiştirebiliriz bunları istemiyoruz create diyoruz ve şu anda bizim için yeni bir imaj yaratma isteği gönderildi hemen buradan ami kısmına geliyorum Bakın şu anda bu bizim Snapshot da yeni bir tane base Image yarattı gelin bununla yeni bir sanal makine oluşturalım instance gidelim launch instance diyelim Bu sefer Quick Start an seçmiyorum my ami den seçiyorum burada base ami im az önce yarattığım base ami burada seçili durumda Yani burada bize gösteriyor Bunu hemen seçiyorum T2Micro ile devam edeceğim bir tane sanal makineye atacağım buna da iam rolünü atıyorum diğerlerini default bırakıyorum storage default bırakıyorum Buna da hemen name olarak 2. sunucu diyorum bir tag atıyorum. Bunu daha önceden Yarattığım security grubunun grubuna altına Daha doğrusu Security grubu makineye atıyorum Review and launch diyorum launch a basıyorum ve bu key e sahip olduğumu onayladıktan sonra launch instance diyorum. Ve böylece ilk makinamızın Snapshot aldım Daha sonra bu snapchat'ten bir ami Yarattım ve bu ami dan yeni bir sanal makineye yaratıyorum bu yeni yarattığım sanal makine Bir önceki sanal makinemin birebir aynısı olacak arkadaşlar yani angineye yüklü olacak ve Onun içine de daha önceden s3 de bulundurduğumuz web sitesini taşımıştım o web sitesinin aynısı da bu yeni sanal Makinamda da bulunacak. hemen bu makinenin yaratılmasını bekliyordum ki yaratılmış şu anda gelin bakalım Acaba bunu o angine çalışıyor mu web sitesi yüklü mü Hemen onu görelim ip adresini burada kopyalayalım ve buraya gelip http:// ip adresini yazalım ve İşte karşınızda Web sitesi ne yapmıştık önceki sanal makinede önceki bir komutla angine kurmuştuk Sonra onu başlatmıştık servisi sonra o servisin her bilgisayar başlatıldığında çalışması için ayar yapmıştık gittik daha sonra s3'ten dosyaları çektik o HDML klasörüne attık vesaire bir sürü iş yaptık Ama şu anda hiçbir şey yapmadık sadece bir ami yarattık ve oami dan yeni bir tane sanal Makineyi 2 saniye için de oluşturduk İşte Arkadaşlar bu kendi ami ini yaratmak böyle işlerde bizim çok işimize yarıyor zaten bütün endüstride bunu bu şekilde kullanıyor kendilerine bir base image yaratıyorlar Yani bütün ayarların içinde olduğu bütün güvenlik açıklarının kapatıldığı bütün update yapıldığı kendilerine göre gerekli uygulamaları yüklendiği Daha sonra da bu base ami dan kendine sanal makine yaratıyorlar ve böylece bu diğer işlemleri her seferinde yapmak zorunda kalmıyorlar Biz konsolumuza geri dönerim ilk sunucumuzda bir çalıştıralım o da çalışsın orada dursun restart diyelim Evet ilk sunucumuz da runnig e düştü Böylece çalışıyor Bir sunucumuz ve 2 sunucumuz artık iki tane web sunucumuz var. birine bir şey olursa bu ikincisini kullanabilirim ama bunun için bir sürü yar yapmam lazım demiştim teori bölümünde ve bunun da çözümlü anlatmıştım Arkadaşlar bir sonraki bölümde çözümlerin uygulaması olan load balancer ve auto scaling konularını uygulamalı olarak göreceğiz Şimdilik hoşçakalın.

EC2 UYGULAMA-3

Merhaba arkadaşlar 3. uygulama kısmında Hoş geldiniz hemen hızlıca konsola bağlanarak devam ediyoruz. Evet arkadaşlar Bir önceki uygulama bölümünde 2 tane sunucu kurmuştuk ilk sunucu ve 2. sunucu Bunlar birbirlerinin birebir aynısıydı siz ben sizlere birazdan load balancing konusunu anlatacağım ve bu load Balancing konusunda bu sunucuları bir rot balacer ın arkasına alacağım ve bunların üzerinden bu konuyu anlatacağım fakat Bakın şimdi size bir şey göstermek istiyorum İlk Sunucunun ip adresini kopyaladım burada web sitesi olarak açtım. Bu gördüğünüz web sitesi geldi. İkinci Sunucunun ip adresini kopyaladım bu da Gördüğünüz üzere burada bire bir aynısı geldi neden Çünkü aslında bu makineler birebir aynı makineler birincisinin Snapshot ından yarattığımız ami den ikisncisini yaratmıştık birebir aynı dosyalar var ya birebir aynı site geliyor da ben hangi sunucuda olduğunu göremiyorum bunları rot balancer ların arkasına aldığımız zaman hangi sunucuya gittiğimizi görürsek load balancer konusunu daha iyi anlayabileceğimiz için bu noktada küçük bir ayar yapmak istiyorum gelin şu sunuculara tek tek bağlanalım ve bu sayfa yani index.html üzerinde bize bunu gösteren sayfayı index.html üzerinde değişiklikler yapalım Bu sayede hangi sunucuda olduğumuzu Browser üzerinden bakarken de anlayabilelim ilk sunucumuzun ip adresini bir kopyalayayım sonra geleyim buradan pati dosyasını açıyorum daha önceden bir aws isimli bir Station kaydetmiştik onu load ediyorum şuraya ip adresini kopyalıyorum geri kalan ayarları zaten yapmıştık burada ssh in altında autotantication altında biliyorsunuz aytieğitim.ppk atamıştık bunu ilk uygulamada görmüştük tekrar dönüyoruz section a ve open diyoruz ve diyoruz ve şu anda bir numaralı sunucuma bağlandım hemen sudo su diyorum Root yetkilerini alıyorum bir Clearla ekran temizliyorum lk uygulama kısmında vim editörü ile bir dosya editlemiştik ama VİM biraz karmaşıktır o yüzden bu sefer nanotex editörü ile xhtml dosyasını editleyeceğim Onu içeriğinde değişiklik yapacağım Nano diyorum neredeydi index dosyamız bu /usr/share/nginx/html/index.html dosyasıydı hemen açtım ilk sunucumuzun your title kısmını değiştirelim buradan ve diyelim ki bu “ilk sunucu” olsun Hatta Şuradaki aytitech header ı da mesela aytitech ilk sunucu şeklinde değiştireyim ctrl x ve ondan sonra da y tuşuna basarak ve enter'a basarak bunu kaydettim Bakın şimdi browser a geri döneceğim Bu birinci sunucumdu 1. sunucumuzun ip siydi. refresh ettim header ilk sonucu olarak ve Aytitech ilk sunucu olarak değişti şimdi EC2 magamentkonsoluna geri dönüyorum İkinci Sunucunun ip adresini kopyalıyorum buradan patty e geri geliyorum ve exit ilei de bundan bağlantıyı kopartayım yeniden patığı açayım bu sefer aws session umu tekrar load ediyorum ikinci sunucumuzun ip adresini giriyorum ve Open Diyerek İkinci sunucuma bağlanıyorum tam ekran yapayım bunu daha Bunu da sudo su ile Root yetkileri alayım Clear edeyim az öncekinin aynısı Nano /usr/share/nginx/html/index.html demiştik bunu nanotex editörü ile açtım bunu da ikinci sunucu olarak değiştiriyorum. Aytitech header de aytitech 2. sunucu yapıyorum ctrl x y ve enter ile kaydediyorum tekrardan Browserıma geri dönüyorum ve bunda refresh ediyorum 2. sunucu şimdi Böylece Biz Browser dan da bağlandığımız zaman web sitelerine hangi sunucuda olduğumuzu görebiliyoruz dönelim konsola. Evet teori kısmında anlattım Load balancer ve auto scaling konularında uygulama yapmaya hazırız sunucularımızı ayarladık bunların ikisini de kurduk ilk önce sunucu 2. sunucu Az önce ayarları da yaptık ve gelin bir senaryo üzerinden load Balancing auto scaling konularını anlatalım senaryomuz da şu olsun arkadaşlar bizim web sitemizin 7gün 24saat çalışmasını istiyoruz yani Hiçbir kesinti olmasın ilk adım olan birden fazla sunucu kurarak yedeklilik ihtiyacını karşıladık Yani ilk sunucumuz var 2. sunucumuz var Birine birşey olursa diğerini devreye alabiliriz Ama herhangi bir kesinti olduğunda bu sunucular arasında geçiş yapma işini otomatize etmedik Yani benim manuel yapmam lazım birinci kullanıcılarımı ilk sunucuya yönlendiriyorum buna bir şey olursa benim manuel şeklinde kullanıcılarım 2. sunucuya yönlendirmem lazım işte ilk aşamada bir load balancer kurarak bu iki sunucuyu bu load balancer arkasından hizmet edecek şekilde configure edeceğiz böylece sunuculardan bir tanesi erişilemez olsa bile gelen trafik diğer sunucuya akmaya devam edecek ve bunun yanında da ileride oluşabilecek ek talebi karşılama işini de çok kolaylaştıracamız ama bunu kolaylaştırmak yetmeyecek. Biz Bu talebe göre sunucu ekleme çıkarma işlemi de otomatize etmek isteyeceğiz ve devreye auto scaling servisi

gidecek Hadi bakalım kolları sıvıyalım ve bu işlemleri başlatalım ilk önce bir load balancer oluşturalım hemen buradan gelelim load balancer kısmına gelelim ve bir load balancer yaratmaya başlayalım hemen crate load balancer ile load balancer yaratalım. burada Bize üç tane seçenek sunuyor arkadaşlar application load Balance Network loadcer ve klasik load Balancer. klasik load blancer en eski sunduğu hizmetlerden bir tanesi fakat bunu yavaş yavaş devreden çıkartıyor bunun içindeki özellikleri ikiye böldü network load balancer ve application load balancer olarka iki yeni servis yarattı kabaca application load balancer daha karmaşık load balancing işlemleri için ama bizim işimiz çok basit ve temel TCP tabanlı bir load balancing ise Bu sefer de network load balancer ı kullanıyoruz bizim işimiz şu anda basit ama biz yine de menülerini görebilmek adına application load Balancer ile devam edeceğiz. klasiğe hiç dokunmayacağız Çünkü bu yavaş yavaş artık kaldırılıyor sistemlerden create diyorum ilk olarak bana bir isim vermemi istiyorlar load balancer a buna buna ilk lb dedim yani ilk load balancer ismim bu olsun. hemen altında scheme soruyor internet-facing mi internal mı internal bunların arasındaki fark da şu internet-facing seçersek internetten dış dünyadan gelen trafik karşılayıp İçerideki kaynaklara bu trafiği dağıtabilir internal seçersek bu sefer sadece de içte WPS içerisinde çalışır. Bu WPS içerisindeki load balancer işlerimiz için kullanılabilir bizim senaryoda ne yapıyorduk dış dünyadan gelen trafiği içerideki iki tane web1 ve web2 Yani ilk sunucu ve ikinci sunucumuza dağıtacağız ve internetten bize ulaşacaklar o yüzden Internetfacing'ini seçiyoruz adres dakikada ip v4 ve dualstack var Biz ipv4 seçiyordu.dualstack seçersek hem ip V4 hem ip V6 load balancing işleri için kullanıyor biz ipv4 ve ipv6 üzerinden devam edeceğiz geldik listeners kısmına bunu teori kısmında anlatmıştım. load balancer ların 2 tane komponenti olduğunu söylemiştim bu komponentlerden biri listener bu listenerlardan biri kabaca dış dünyadan bu load balancer hangi Protokolü hangi portu dinleyecek Biz dış dünyadan Http üzerinden bir alacağımız için Http 80 Port u üzerinden dinleme yaptıracağız. diyelim ki https i de açmak istiyoruz yani güvenli trafiği de açmak istiyoruz o zaman https dei buradan seçip 443 ekleyebilirdik. altındaki avalibility zone kısmında ise load balancer ın dış dünyadan dinlediği trafiği içeride hangi avalibility zonlar daki hangi makinaları aktarabileceğimiz seçebiliyoruz daha doğrusu makineyi değil avalibility zone ları seçebiliyoruz bu avalibility zone kısmını teorik olarak basitçe anlattım ama bunu tam anlamıyla anlayabilmek için vpc kısmını da tam amlamamız gerekiyor birkaç bölüm sonra ona gelip bütün bu Networking alt yapılarını anlatacağım arkadaşlar O zaman bu avalibility zone kısmını daha iyi anlayacaksınız ama şimdi Biz buradaki bütün avalibility zone ları seçelim ki bu load balancer ımız içerideki bütün avalibility zone lar daki sistemlerimizde trafiği aktarabilirsin hemen bütün seçenekleri yaptık istersem bir tag de girebilirim buna Hani her zamanki gibi mesela Buna da neyim tag girebilirim ama girmiyorum bunu tag lik bir iş yok şu anda konfigüre security settings diyorum. Bana hemen bir uyarı veriyor Diyor ki sen bunla http trafiğine açtı ama işte tipi güvenliği protokol değildir bunda https i de aç ve bir sertifikalama ile trafiği sağla ve böylece trafiğin güvenli olsun ama Şu anda bir eğitim kapsamında http açmanıza gerek yok o yüzden bu uyarıyı dikkate almıyoruz ve konfigüre Security grubu ile devam ediyorum yeni bir Security grupları yaratmıyorum zaten bir tane yaratmıştım iyisi to EC2 Security Grup isimli EC2-grup ismi ilk başta Yarattım Security Grup ı kullanacağım. Çünkü bu zaten 80 portuna dünyanın her yerinden izin veriyor. O yüzden bu benim işim görecektir bu Security grubu load balancer a bağlıyorum yeni bir tane yaratmıyorum ve geldik target Grup kısmına bir önceki ekranda Daha doğrusu iki önceki ekranda load balancer dış Dünyadan hangi Port üzerinden trafik dinleyeceğini belirtmiştik listener ayarlayıcılar şimdi sırada bu gelen trafiği içerden nasıl dağıtacağını belirleyeceğimiz kısım var bunu aws dünyasında target Grup oluşturarak yapıyoruz yani Hedef grubu. gelin hemen bir isim girelim Buna da diyelim ki ilk target olsun bunun adı da Target taytları 3 tane seçeneğimiz var instance' ip ve lambda function bunların Aralarındaki fark şu instances seçersek bizim oluşturduğumuz EC2 isntance lara Hedef olarak buna kayıt edebiliyoruz ip seçersek manuel ip lerini girerek bu EC2 instance larınveya dış dünyadaki başka nereye load balancer ilk yapacaksak oradaki makinalarına ip adreslerine girerek yapıyoruz ya da load balancer ve arkadaki ve birden fazla lambda fonksiyonunu ayarlayabiliyoruz lambda fonksiyonu ne bunun içinde arkadaşlar 6. kısmı

beklememiz gerekiyor 6. kısım da bunu daha detaylı şekilde anlatıp tüm özelliklerini göreceğiz bizim işimiz için instance' yeterli Çünkü biz instance' yani EC2 üzerine load balancer yapacağız yani aldığımız paketi ise instance lara göndereceğiz burada protokol var aynı şekilde nasıl dış dünyadan listener olarak http80 seçmiştik bu şekilde dinliyordu bizim içerdekii sunucularımızda http 80 portunu dinliyor ve bunu soruyor yani dış dünyadan aldığım trafiği içeride hangi Porta hangi protokol üzerinden göndereyim kısmını buna protokol ve porttan seçiyoruz. listede Http tipi HTTPS var biz http üzerinden devam ettiğimizi söylemiştik. Geldik healt checks kısmına şimdi bu kısmı biraz detaylı andıralım arkadaşlar load balancer lar arkasında bulunan ve trafiği teslim ettiği cihazların düzgün çalışıp çalışmadığını kontrol eder ve eğer düzgün çalışıyorsa bu cihaza trafik gönderir eğer düzgün çalışmıyorsa yani istenilen şekilde çalışmıyorsa da Bu cihaza trafik göndermeyerek kesintileri engeller buna health cechecking yani sağlık kontrolü diyoruz bizim load balancer ımız bu health check işlemini http Protokolü üzerinden yapıyor bizim burada girdiğimiz yere bir istek gönderiyor ve Eğer olumlu cevap alırsa yani şu path a bir istek gönderiyor ve olumlu bir cevap alırsa anlıyor ki O arkadaki sunucu sağlıklı ve ona trafik Göndermeye devam ediyor ama bizim burada belirlediğimiz healt check path indeki url ye gidip buradaki dosyaya gidip bunu bulamazsa veya bundan bir cevap alamazsa da bu sefer o Sunucunun düzgün çalışmadığını anlıyor ve oraya trafik göndermiyor Böylece işte baştan beri yapmak istediğiniz şeyi sağlıyor bize şöyle düşünün Ben bunun arkasına iki tane makine koyacağım bu iki makine ya da bu health check leri yapmasını isteyeceğim bu ayarı yaparak ve bu load balancer bu arkadaki makinaları bu health check ler üzerinden kontrol edecek Eğer makinalardan olumlu cevap dönmüyorsa anlayacak da makinada bir sorun var ve onu devreden çıkartacak ona trafik göndermeyecek ve trafiği diğer makineyi Göndermeye devam edecek Böylece bizim kullanıcılarımızı dış dünyadan bağlanan kullanıcılar hiçbir kesinti yaşamadan Aynen çalışmaya devam edecekler şimdi buradaki protokol tahmin edebileceğiniz üzere yine Http hangi protokoller içerideki makineyi deneyeyim diyor Biz http seçiyoruz ve path yani nereyi deneyeyim diyor Şimdi arkadaşlar bakın bu ilk sunucuda Ben direkt ip adresini yazdığım zaman bağlanabiliyorum yani şöyle gidip buna path/extra/deneme/html yazmadım yani herhangi bir şey girmedim Demek ki benim bunun esas path i bu yani default u bu Ama ben bunu kullanabilirim Ya da buraya bir tane dosya atarım bu sunucuya Örneğin health check isimli bir dosya yaratırım health check isimli isimli dosya yaratırım onu o Html klasörüne koyarım ve onu kontrol etmesini isteyebilirim Eğer benim durumum böyle olsaydı ben bunu buradan buraya bu şekilde girecektim ama bizim böyle bir durumumuz yok index.html var ve o zaten default Path de duruyor o yüzden health check için bu kısım Yani default u bizim için yeterli olacaktır hemen altında da Advanced health check var Bunları bir göz atalım şimdi Dedi ki bu burayı kontrol edecek path kontrol edecek ama bu kontrolün kuralları nasıl olacak işte o kuralları da advanced health check settings altında belirliyoruz arkadaşlar Çünkü burada 5 tane değer var bu değerlere tek tek bakalım healthy trashold şu demek kaç defa bu load balancer arkadaki makineye istek gönderip kaç defanın sonunda bunun sağlıklı olduğuna kanaat getirecek burada ki default değer 5 Yani bu makine 5 defa o İçerdekii makineye sorgu atacak 5 inden de olumlu sonuç alırsa diyecek ki Evet bu makine sağlıklı aynı şekilde bu 5 sorgu gönderdiğinde diyelim ki 2 tane yani Şurada belirlediğiniz değer kadar bu sunucu dönemezse yani 2 kere fail ederse üst üste bu sefer de diyecek ki bu sunucu unhealthy. Peki bu bağlantı kurmaya çalıştı ama bu bağlantıyı tam kuramadı bir Time Out oldu bu timeout olabilmelerini kaç saniye beklesin diyoruz. O da 5 saniye olarak belirlenmiştir default unda. interval kısmı da bu her deneme arasında kaç saniye geçecek onu soruyor yani demedim bu kadar süre bekle 30 bekle 30 saniye bekle Bir daha dene 30 saniye daha bekle Bir daha dene 30 saniye daha bekle şeklinde ilerleyecek Yani bizim bu makineyi bunun arkasına aldığımız zaman bu 5 x 30=150 saniye boyunca buna 150 saniye boyunca 5 defa buna istek gönderecek we 5 inden de olumlu cevap alırsa diyecek ki bu makina healthy ama 2 defa olumsuz cevap alırsa da bu sefer diyecek ki an healthy Peki bu cevap ne? işte bu cevap da arkadaşlar healthy olabilmelerini sağlayan cevap da Http protokolünün kodunu 200 alabilirse bunu healthy olarak görecektir 200'ün dışında bir kod alırsa bunu unhealthy olarak görecektir işte advanced healthy settings bunlar ve

bunların hepsini default olarak bırakıyoruz. Herhangi bir ekstra yapmamıza gerek yok ve hemen bir sonraki ekranda registered Target ile devam ediyoruz bakın burada listede benim 2 sunucum da listelendi çünkü en başta ben bütün availability zone ları seçmiştim oradaki bütün makineleri bana göstermesi gerekiyor. Bizim de 2 tane makinemiz var şu anda açık ilk sunucu ve ikinci sunucu Hemen bunları bu target gruba kayıt edeyim bakın et to registered diye bir buton var Bunları seçtikten sonra bunlara bastım artık bu makinalar bu target grubun altında yani bunlara trafik göndermeye başlayacak son ekrana geçiyorum ve bütün ayarlarını kontrol ediyorum Bütün yaptığım seçenekleri bir kontrol ediyorum bakıyorum ve create basıyorum ve şu anda load balancer yaratılıyor bugün gördüğünüz gibi State provincing diyor 5 dakika bekleyelim Ben burada videoyu durdurayım 5 dakika sonra tekrardan döneyim ve Bu provision olduktan sonra bakalım load balancer nasıl çalışıyor Bir özelliklerine bir göz atalım Evet şimdi aktif oldu yani load balancer oluşturuldu hemen ilk olarak Target Groups a gitmek istiyorum Bakalım yarattığım Target Grup duruyor mu Evet burada duruyor ve bu target grupta da arkadaşlar hemen şöyle ekranı biraz büyütüyüm yukarı çekeyim targetlara geleyim ve bakın registered targets da yani kayıtlı hedeflerde 2 sunucum da duruyor ve statusları şu anda healthy yani bunlarla health checks yapmış ve bunları sağlıklı olarak görmüş ve bunlara trafik göndermeye hazır geelim load balancer ra bakın trafik gönderiyor mu gerçekten onu da bir test edelim nasıl çalışıyor ona bir bakalım şöyle bir yukarı alayım burada dns name i var. Yani bu load Balancer ın dns ismi var. Arkadaşlar elb de sadece dns üzerinden çalışır Yani sizi hiçbir zaman ip adresini göremezsiniz Bu sınavda karşınıza çıkabilir aklınızda olsun elb servisi her zaman size sadece dns ismini verir direkt ip adresi göremezsiniz bunu kopyalıyorum browser a geçiyorum buraya yapııştırıyorum entre basıyorum Bakın şu anda karşımda websitem duruyor ve ikinci sunucuya düştük refresh edeyim tekrar ikinci sunucu bakım bir daha refresh ettim bu sefer ilk sunucuya gitti gördüğünüz gibi ben artık load Balancer a gidiyorum load balancer arkada bir sunucuya 1 ve 2 sunucuya şeklinde trafiği mi dağıtıyor Bu sunuculardan bir tanesi kapansa bile benim artık dış dünyadan kullanıcıların bu url ye gelecekleri için her seferinde bir sunucuya ulaşabilecekler ve ben hiçbir kesinti yaşamadan kullanıcılarına hizmet vermeye devam edeceğim. Hatta gelin bunu da simüle edelim EC2 management konsoluna gidelim instance lardan bir tanesini stop edelim Örneğin 2. sunucuyu stop edelim Action dan instates stop deyim ve bu sunucu kapatılsın. Bakın şimdi bile bir süre beklemem gerekecek Neden ne demiştik kaç kere deneyecekti unhealth olması için 2 defa denediğimde Eğer ikisinde de ulaşamazsa aralarında 30 saniye olacaktı bunu unhealthy olarak işaretleyecekti Bakalım şimdi bu kapanıyor servisler artık Cevap vermemeye başlamıştır gelin biz tekrardan load balancer altında target gruba gidelim ve targetlara bakalım bakın hemen 2. sunucu unused a düştü yani kullanılamaza düştü birazdan bunu refresh etmeye devam edersem buradan unhealthy ye düşecek ve buradan kullanılmayacak zaten şu anda buna trafik göndermemeye başladı load balancer bakın refresh ediyorum her seferinde ilk sunucuya gidiyor hiçbir kesinti yaşamadım 2. sunucu arka tarafta kapandı ama benim websitem şu anda çalışmaya devam ediyor Şimdi tekrar dönelim load balancer a biraz menüsünü inceleyelim description ekranında neler var Ona bir bakalım bütün temel özellikleri burada görebiliyoruz isim dns adresi gibi şu anda aktif mi Nasıl bir load Balancer tarzı bütün ayarları buradan görebiliyoruz yani yaptığımız bütün o başlangıçta seçtiğimiz bütün ayarları buradan görebiliyoruz Hemen listener kısmından bu yarattığımız listener ve onun detaylarını görebiliyoruz istersek yeni listener ekleyebiliyoruz Monitörüng kısmından tahmin edebileceğiniz üzere buna gelen istekleri bunların kaç saniye içinde cevap verdiğini ve http400 ayarı http400 kodunu 4xx kodunu kaç tane ürettiğini veya işte elb nin 5xxx inin count unun bütün monitoring detaylarını buradan görebiliyoruz integrated service konusu şu anda bizim için bir alakası yok ileride bunu altıncı kısımda göreceğimiz servisler var aws config ve Waf gibi bunlarla Entegre edebilmek için olan kısım ve tags de adından da anlaşıldığı gibi etiketler menülerde bu şekilde arkadaşlar hemen tekrardan EC2 instance kısmına geri dönelim load balancer kısmını bitirdik şu ikinci sunucuyu tekrar bir açalım. Evet ve yes diyelim ve hemen ardından da bu bölümü Şimdilik sonlandıralım. Auto scaling i de Bir sonraki bölümde anlatmaya başlayalım Arkadaşlar bir sonraki bölümde görüşmek üzere.

EC2 UYGULAMA 3 (DEVAM)

Arkadaşlar bir önceki bölümde Load balancer kısmını halletmiştik sisteminize gelen trafik kesintisiz olarak garanti altına almıştık ama şimdi ortamda şöyle bir sıkıntımız daha var ya İki cihaz birden bir şekilde giderse yani bu ilk sunucu ve ikinci sunucu aynı anda kapanırsa veya daha da farklı şekilde Bu ilk sunucu ve ikinci sunucu üzerlerine gelen trafik yüzünden fazla trafik yüzünden Cevap veremez duruma gelirse İşte bu gibi durumlarda da süreci otomatize edebileceğimiz auto scaling denilen bir servis var ve bu üçüncü uygulama kısmının ikinci bölümünde bu auto scaling servisini göreceğiz auto scaling servisi bize sanal makinelerimiz ile oluşturduğumuz ideal bir ortam tanımlamamıza imkan veren servis daha sonra bu ideal ortamı auto scaling servisi bizim adımıza sürekli ayakta tutmak için otomatik işlemler yapıyor autoscaling iki temel komponentlerin oluşuyor. Bunlardan ilki launch config aslında launch config sadece auto scaling için değil Şu anda düşünmeyelim launch config şu demek ben bir sanal makina oluştururken ne yapıyorum ami seçiyorum tag giriyorum role Seçiyorum bir sürü ek seçenek seçiyorum ami seçiyorum makine boyutunu seçiyorum vesaire launch config le bu seçtiğimiz şeylerin hepsini daha önceden bir dosya olarak belirliyorum ve auto scaling gibi servislerin bana sormadan bu cevapları bu dosya üzerinden almasını sağlıyorum. Gelin hemen launch config i yaratalım buradan listeden auto scaling e geliyorum ve bunun altında launch configurations kısmına tıklıyorum hemen create launch configuration diyorum yine sanki sıfırdan bir sanal makine yaratıyormuş gibi aynı ekran çıktı buradan İlk seçenek olarak ami seçeceğim ve ben daha önceden Yarattığım base ami ile auto scaling grubu yaratacağım için bu base ami seçiyorum launch config de yine bir sonraki ekranda bana ne boyutta bir makine istediğimi soruyor T2Micro ile devam ediyorum buna bir isim veriyorum bu launch configuration a diyorum ki ilk launch config adı bu olsun herhangi bir spot isteğim de bulunmayacağım iam rolünü seçiyorum EC2-s3 full access monitoringde detail monitöring Açmak istemiyorum istersem buraya yine bir script girerek bu makinelerin ilk çalıştırıldığında işlemler yapmasını sağlayabilir mi bunların hiçbirini yapmıyorum zaten base ami de gerekli bütün ayarları yapmıştım hemen depolama kısmına geçiyorum normal 8 GB'lık bir GP2 bir root device benim için yeterli olacaktır Security gruba geçiyorum daha önceden yarattığım EC2 sec grubu seçiyorum review diyorum ve create launch configuration diyorum Hemen ardından bana Key pair seçtiriyor daha önceden yarattım key pairi seçiyorum ve create launch configuration diyorum ve böylelikle arkadaşlar ilk launch configuration yarattım bakın ne yaptım biz launch configuration yaratırken Aynen bir sanal makine yaratır gibi bütün önceden içmem gereken şeyleri seçtim yani gittim hangi iam rol atanacak onu seçtiğim disk boyutunu seçtiğim Buna Hangi Key pair atanacak onu seçtim Yani aynen bir sanal makine yaratır gibi tek tek adımları geçtim ama sonunda bir sanal makine yaratmadım. Yani bu seçtiğim şeyleri bir dosyaya kaydettim Aslında ve birazdan biz bu dosyadan bir tane auto scaling grubu yaratacağız her seferinde yeni bir makine yaratırken bizim bu seçtiğimiz şeylerle bu makineyi yaratacak yani bize her seferinde sormayacak launch konfiguration bunun için kullanıyor arkadaşlar auto scaling altında kullanabiliyoruz bunu başka servislerde kullanabiliyoruz bu tarz önceden seçmem gereken şeyleri auto scaling la seçebiliyorum şimdi geçelim buradan Yine auto scaling grup sahasında buradan create auto scaling grup dersem launch config yaratabilirim ama bizim grupta geçelim ve buradan create Auto scaling Grup diyelim ve sıfırdan adımları görelim iki tane seçeneğimiz var launch configuration ve Large Template. BU Template arkadaşlar launch configuration in yerine geldi Arkadaşlar artık lauch config yaratmıyoruz launch template yaratıyoruz ama biz launch configuration yarattığımız için onunla devam edeceğiz ilk ekranda bunu seçip hemen ardından da az önce yarattığımız Lauch config i seçiyoruz ve Next ile devam ediyoruz şimdi buna bir isim ver diyor ilk oto scaling Grup diyelim ismimizi verelim grup size bu kısım önemli Arkadaşlar siz bu auto scaling kaç tane makine yaratmak istiyorsanız size onu soruyor biz ortamımızı nasıl kurmuştuk ne demiştik bizim en azından iki tane makineyi ihtiyacımız var. Her seferinde en azından iki tane makine olsun ki hani biri kapanırsa diğeri devam edebilsin. O yüzden bizim instance size ımız yani Bu auto scaling grubun yaratacağı Instance adedi

iki biz bunu seçtik ardından bize vpc seçtiriyor dediğim gibi tekrarlıyorum Bu vpc kısmına geleceğiz Şimdilik çok önemsemedim ve subnet lerin hepsini burada seçelim ve advanced details kısmına basalım. şimdi bu noktada advanced details kısmında bize ilk olarak şunu soruyor sen auto scaling ile yaartacağın makinaları Çünkü burada iki tane makine yaratacağız bunları daha önceden yarattığım bir load balans arkasına load Balance arkasına otomatik olarak eklemek istiyor musun. Evet istiyorsan git buradan recieve trafic from one or more load balancers ı seç hemen ardından da biz klasik load balancer değil de ne yapmıştık application load balancer yaratmıştık onların altında da target Grup vardı git o target grubu seç. Böylece ben bu her yer yarattığım auto scaling grup ile her yarattığım bu auto scaling grup ile her yarattığım makinayı bu load balansın arkasına ekleyim Böylece gördüğünüz gibi süreci Otomotize etmiş oluyorum. load balancer Yarattım ardından auto scaling grubu yaratıyorum buradaki makinalara bir şey olsa bile bu auto scaling Grup her seferinde En azından iki tane makineyi orada açık tutacak ve bunları da load balancer ın Ardında arkasına ekleyecek burada altında help Check type diyor bunların sağlıklı olup olmadığını yine load balancer da anlattığım şekilde çalışıp çalışmadığını neye göre kontrol edeyim diyor EC2 üzerinden mi elb mi bunları bunları çek etsin diyor Biz bu çeki zaten elb de yaptırdığımız için elb bunları çek etsin diyor geri kalanlara dokunmuyorum arkadaşlar bunları standart bırakıyorum ve configure policy ile devam ediyorum şimdi bu kısım önemli autoscaling grupta Burada iki tane seçenek var arkadaşlar. keep this grup at this initial size veya use scaling policises to adjust capity of this group ilk seçeneği seçersek arkadaşlar auto scaling grubun ilk yapmasını istediğimiz şeyi yapmasını sağlarız yani her seferinde Burada en azından iki tane sanal makine bulunmasını garanti eder. yani Ben bunu seçtim ileri ileri dedim Bütün ayarları tamamladım auto scaling grup gidecek bana iki tane makine yaratacak Ben bir şekilde bu iki tane makineyi terminate edersem gidecek yeniden 2 tane makine yaratacak bir tanesinde bir sıkıntı çıkarsa gidecek Onu silecek yeni bir tane yaratacak ve her seferinde Burada en azından iki tane makine olmasını Garanti edecek. Fakat benim istediğim şey bu değildi. Biz ne demiştik teori bölümünde biz bunları yük gelirse dışarıdan çok fazla istek gelirse yeni makinalar eklesin çok fazla istek azalırsa Normal haline dönerse de bu eklediği makinaları silsin yani arada autoscaling yapsın bunu da işte Arkadaşlar bu ikinci seçenekle seçiyoruz ve bana diyor ki bunları bu makinaları hangi aralıkta yaratayım Ben bakın Burada 2 ile 4 arası diyeceğim şimdi bu şu manaya geliyor burada her seferinde En azından iki tane makine bulunmasını sağla ama birazdan şu altta gireceğim ayara göre buradaki durum oluşursa 4 taneye kadar Buraya yeni makine ekle bu alttaki durum ortadan kalkarsa da tekrardan iki taneye kadar geri dön işte arkadaşlar auto scaling yani otomatik olarak bunları yukarı doğru Scale etmek ve aşağı doğru scale etmek işini buradaki bu ikinci seçenek ile yapıyoruz biz 2 ile 4 arasında dedik ve bunu hangi kurala göre yapacağız bakın buradan metrik type diye 4 tane seçenek çıkacak karşımıza application load balancer reusest count per target, average cpu utilization, average network on, average network out. ilk seçenek şu demek load balancer a gelen istek sayısına göre ben bunu seçersem ve dersem ki buraya 90 atıyorum veya 200 eğer load balancer a aynı anda iki yüzden fazla kullanıcı istek gönderirse bu yeni bir tane sanal makine yaratmaya başlayacak diyelimi 300 oldu 400 oldu 4 taneye kadar sanal makine yaratmaya devam edecek fakat diyelim akşam oldu benim web sitemin trafiği düştü 200'ün altına indi kullanıcı sayısı yani dış dünyadan Benim web siteme bağlanan kullanıcı sayısı 200'ün altına indi bu sefer de gidecek arkadaşlar bu yarattığı iki tane makineyi silecek ve sadece ortamda 2 tane makine bırakacak yani otomatik yüke göre load Balance edecek Biz bunu denemek adını avegara cpu utilization ile de devam edelim Buna da bir değer olarak 30 girelim de bu iki makine Normalde ortamdaki makinaların cpu ları ortalama yüzde 30'un altında ise normal bırak yüzde 30'u geçerse buraya makine eklemeye başla şeklinde değer giriyorum. Bunu şimdi çok basit şekliyle yaptık 30 dedik 30'un üstü 30 altı şeklinde yaptık Ama ben burada scalie the auto scaling group using step or simple scaling policies tıklarsam da bu sefer bunu arttırma ve azaltma olarak ayarlayabiliyrum yani %90 üzerine geçerse ekle Ama yüzde otuzunu altına inerse de ortalama cpu kullanımı Bu sefer de kaldır şeklinde daha detaylı ayarlama yapabiliyorum ama benim işim için şimdilik bu average cpu utilization un yüzde 30'u Üstü altı

kısmı görebilmemiz açısından yeterli O yüzden şimdilik bunu seçerek devam edeceğim ve konfigüre notifications ile devam ediyorum eğer ben bu autoscaling her seferinde yeni bir makine yarattığında veya makineye ortamdaki kaldırdığında bana bir bildirim göndermesini istiyorsan bir Email atmasını istiyorsam burada Ad notification diyerek SMS servisinin 6. kısım da göreceğiz detaylarını o servis aracılığıyla bana mail atılmasını sağlayabilirim. Bizim şimdilik bunlarla işimiz yok konfigüre tag diyorum yine tag gireyim bunlara mesela name diyeyim Buna da diyeyim ki autoscaling grup ile yaratılan şekilde bir tag gireyim burada tag new instances a tıklarsam da bu Yarattığım name ve key value pair i gidecek bütün yarattığın sınıfları atayacak biz bunun olmasını istiyoruz bütün yarattığı instanceları bu şekilde eklesin diyoruz o yüzden bu da işaretli olacak bir Ve basıyorum ve create auto scaling grup diyorum şimdi auto scaling grubum yaratıldı bakın Desired iki Yani ben burada kaç tane makine olmasını istiyorum minimum en az kaç tane makine olsun istiyorum maksimum toplamda kaç makine olmasını istiyorum az önce yarattığımız kural buydu instance kısmında da şu anda bu auto scaling grubunun altına kaç tane makine var şu anda sıfır birazdan bunu refresh etmeye başladığınızda gördüğünüz gibi şu anda 2 oldu Çünkü dönersem dashboard a Buradan da instance sa tıklarsam Bakın iki tane hemen sanal makine yarattı ve bu şu anda bunları ayağa kaldırmaya çalışıyor. auto scaling grup tekrardan dönüyorum auto scaling gruba. Geliyorum buradan scaling Activity history ye iki tane successful şeklindeki tane yani status successful Düzgün şekilde yaratabildin diyor iki tane EC2 instance yarattı. Hemen bakalım biz bunları load balancer ın arkasına eklemesini istemiştik load balancer ın arkasına beklemiş mi target gruba geliyorum Açılmasını bekliyorum targets a geliyorum Evet bu yeni yarattığı auto scaling grup ile yaratılan sunucuları da load balancer arkasına eklemiş Yani ben buradan tekrardan load Balancer a gelip load Balancer ın dns ismini ismini kopyalayıp buraya Gidersem Bakın bu dört makineye artık her seferinde tek tek load balancer üzerinden erişebileceğim peki auto scaling grup tam anlamıyla ne işe yarıyor bunu görmek için arkadaşlar tekrardan instance a geliyorum ve bu auto scaling grup ile yaratılan makinalardan bir tanesini terminate ediyorum yani bir şey oldu bu makinede bir sorun çıktı ben ve ben bu makineyi kapattım ortamdaki bir sıkıntı oldu bu makine kapatıldı. Mesela servis durdu mesai bunun içindeki engine.exe servisine bir sıkıntı oldu ve servis durdu ve artık cevap vermiyor. Bu makine artık ortamda yok bakın Şimdi tekrardan auto scaling gruba döneceğim hemen scaling polices değil de Activity History e geleceğim ve burada biraz bekleyeceğim Arkadaşlar şimdi o makinenin Kapanmasını bekliyorum ve aradan da 300 saniye geçmesi gerekiyor çünkü auto scaling grubu kurarken Advanced settings de 300 değeri görmüştük help chech için bu 300 değeri burada bu makinenin ne kadar beklemesini istediğimizi söylüyor yani Ne kadarlık bir süre boyunca beklesin Eğer bir sorun varsa ona göre aksiyon alsın diye yani 3 saniye beklememiz gerekiyor 3 saniye sonra burada bu sanal makinenin kapatıldığını göreceğiz ve yeni bir sanal makine yaratacak bakın hemen gördü zaten şu anda EC2 instance ın terminate edildiğini gördü ve elb den bağlantıların düşürülmesini bekliyor Bu işlem tamamlandıktan sonra bakın ne yaptı launching a new EC2 yeni bir tane instance yaratmaya başladı hemen bu instance a geelim ve bunu görelim bakın hemen Burada yeni bir EC2 instance yarattı ve bunu devreye almaya çalışıyor biraz Bekleyeceğiz ve Daha sonrasında bu EC2 instance yaratılacak ve onu da elb nin arkasına atılacak Şimdi geelim Kafamızdaki bir soruya O sorunun cevabını bulmaya biz tekrar dönelim auto scaling gruba ne kadar dedik kaç tane makine olmasını istiyoruz demiştik 2 minimumda kaç olacak 2 maksimumda kaç olacak 2 şu anda kaç tane var 3 bu birazdan 2 olacak çünkü bir tanesini kapattık ama az önce söyledim 300 saniye mevzusundan dolayı Hala sende duruyor birazdan bunu refresh ettiğimde Burası da 2 kalacak O yeni makine yaratıldığında yani burada 2 gözüküyor onu söylemek istiyorum ama intance a geliyorum benim burada 4 tane sunucum var arkadaşlar. 1 2 3 4 ilk sunucu ve ikinci sunucu ortamda varken autoscaling Grup gitti 2 tane daha sunucu yarattın Neden şimdi kurallardan bir tanesi şu arkadaşlar auto scaling grup sizin mevcut ortamınıza bakmaz yani o mevcut ortamda hangi makineler var Siz bunları da bunun içine katmak istediniz mi istemediniz mi ilk yarattığınız zaman auto scaling grup bunu önemsemez ortamda sizin istiyorsanız 200 tane sanal makinamız olsun Siz Buna dediyseniz 2 tane sanal makine yarat o gider iki tane

sanal makine yaratır Çünkü bu bizim ilk sunucu ve ikinci sunucu Biz bu işi yapmak için kurduğumuz sunucular olmasına rağmen auto scaling grup bunların bu iş için kurulduğunu bilmiyor bizim bunları autoscaling gruba tanıtmamız gerekiyor Onu nasıl yapacağız İlk sunucuya tıklıyorum Action sa geliyorum Buradan instance settings e geliyorum ve hemen bakalım attach to Auto scaling Grup seçeneğini seçiyorum mevcut bir auto scaling grubu Gruba ekle seçeneğini seçtim yarattığı auto scaling grubu seçtim ve attach dedim ve böylelikle benim daha önceden yarattığım Bu sunucuyu da autoscaling grubun içine eklemiş oldum ikinci sunucuya geliyorum Aynı işlemi yapıyorum instance Setting attach to auto scaling Grup diyorum mevcut bir auto scaling gruba listeden auto scaling grubunu seçiyorum Attach diyorum bunu da şu anda bizim yaratmış olduğumuz auto scaling gruba ekledi fakat dönüyorum auto scaling gruba şimdi ne oldu desired 4 minimum 2 maksimum 2 niye Desired 4 oldu Biz başlangıçta 2 seçmiştik biz ortamda En azından iki tane makine olmasını istiyorduk ama ne yaptık sonradan 2 tane de biz manuel ekledik işte bu manuel eklediğiniz arkadaşlar bu desired Yani istediğiniz sayıyı artırır bu aklımızda olsun bunda Action Stan Edite tıklayarak tekrardan desired kapasitesi 2 dersem ve save basarsam şimdi biraz beklersen olacaklar şu autoscaling Grup bakacak burada kaç tane sanal makine var çalışan 1 2 3 4 ve bakın hemen bunların ismini de auto scaling grup ile yaratılan şeklinde değiştirdi Çünkü biz bunları auto scaling grubunun altına almıştık gidecek bakacak burada kaç tane çalışan sanal makine var 4 biz tane sanal makine çalışmasını istiyoruz Optimum durumda 2 ne zaman biz bunların 4e kadar çıkmasını istiyoruz Eğer cpu kullanımları yüzde 30'un üzerinde ise şu an da bunların cpu kullanımları ortalama %30 üzerinde mi hemen bakalım Bakın şu anda neredeyse neredeyse hiç kullanılmıyor yüzde 0,15 neredeyse hiç kullanılmıyor neredeyse Ortalama 100 de 1 falan kullanılıyor Demek ki şu anda benim yüzde 30'un üzerinde bir ortalama cpu kullanımı yok o zaman auto scaling Grup biraz bekledikten sonra bu sana o makinalardan 2 tanesini kapatmaya başlayacak ve ortamda 2 tane sanal makine bırakacak ki benim bu sana makinalarının ortalama %30 cpu kullanımı yapmaya başlamasına kadar. Eğer böyle bir durum olursa da gidecek ortama yeni sanal makine ekleyecek ki birazdan bunları da göreceğiz Şimdi arkadaşlar bu noktada video durduruyorum bir süre Çünkü bu auto scaling bu durumu algılamasını Beklemek istiyorum auto scaling bu durumu algıladıktan sonra döneceğim ve buradaki sanal makinelerin kapanmış olduğunu sizlere göstereceğim ve hemen ardından da bu sanal makinalara bağlanarak bu sanal makinalarda bir süre bir yük yaratmak istiyorum yani cpu kullanımı %30 dan fazla hale getirmek istiyorum O zaman da dönüp buraya bakacağız ki auto scaling Grup Yeni sanal makine olarak eklemiş hemen durduruyorum birazdan görüşürüz Evet yaklaşık 5 dakika kadar bekledim ve buradaki 2 tane sanal makineye gitti termate etti Bunu da görmüş olduk tekrar gidelim auto scaling gruba ve bakalım buradaki işlemleri neler yapmış hemen aktive history ye geliyorum şunu biraz daha yukarı çekeyim bakın gitti buradan 2 tane sanal makineyi termate etti Çünkü daha önceden Biz ne yapmıştık Bunun iki tane instance ortalama Desire instance iki olmasını söylemiştik ama ortamda 4 tane vardı ve cpu kullanım yüzde 30 un altındaydı O yüzden iki taneyi instance ı otomatik olarak terminate etti auto scaling grubu nasıl çalıştığını anladık Ama bu cpu kısmını da bir görelim Yani biz bu makinelere bir load atalayalım Yani bu makineler biraz çalıştı ve cpu yüzde 30'dan fazla kullansınlar Daha sonra da bekleyelim auto scaling grubun bu duruma karşı yeni makineler ekleyerek bu sistemi rahatlattığını kendi gözlerimizle görelim bunun için de bi tane stres tool diye tool var arkadaşlar linux dünyasında bu stres toolu makinada çalıştırdığınız zaman ve belirli parametrelerle cpu kullanımı yaratıyor Yani bunu simule edeceğiz Çünkü benim şu anda bu makineler dış dünyadan 20000 tane kullanıcının aynı anda bağlanmasını simüle edebilme imkanım yok o yüzden Makine üzerinde simule edeceğim Bu makinelere bir stres tool yükleyeceğim bu tool aracılığıyla da bu makinelerin yüzde 30'dan fazla cpu kullanımı yapmalarını sağlayacağım şu anda makinelerin ip sini kopyalayıp putty üzerinden bağlanayım Hemen load diyeyim ilk defa bağlandığım için soruyor Clear diyorum Su dosu diyorum bir daha Clear diyorum buradan hemen fedora kütüphanesinden bu stres tool unu yükleyeceğim Komutları Sizlerde eğitimin kaynaklar kısmında bulabilirsiniz hemen bunu yüklüyorum hemen arkasından da stres aracını yüklüyorum bekliyorum yüklenmesini ve şu anda yüklendi Aynı işlemi diğer

makinada da yapacağım yine yeniden bir putty çalıştırırım. Bunun üzerinde stresi çalıştırırım Ondan sonra diğer makineye bağlanayım. Hatta bunu da time out 20000 diyeyim şu anda stres bu makinada çalışmaya başladı Yani bu makinada cpu kullanımı oluşmaya başladı bu makineden bağlantıyı kesiyorum şimdi diğer makineye bağlanmak için putty yeniden açıyorum diğer makinanın çalışan diğer makinenin ip adresine bakıyorum kopyaladım load dedim kopyaladım load dedim kopyaladım açıyorum sudosuyla root yetkisi alıyorum tekrardan text Dosyalarımaya dönüyorum bunları yapıştırıyorum hemen burada kopyalayıp stresle kuruyorum install ediyorum stres tool kuruldu uygulama hemen stres uygulamasını çalıştırıyorum ve bunda cpu kullanımı yaratmasını sağlıyorum dönüyorum şimdi ise EC2 konsoluma Bakalım burada monitöründen cpu kullanımı oluşmaya başladı mı Evet cpu kullanımı oluşmaya başladı. Şu anda yüzde altı cpu kullanıyor bu birazdan çoğalacak her 5 dakikada bir kontrol ettiği için bunu 5 dakika sonra tekrar görebileceğiz. Bunda bakıyorum cpu oluşmaya başladı mı bunda da oluşmaya başladı. Hatta bu ikisini birden seçersem aynı anda ikisini birden görebilirim ne kadarlık cpu kullanıyor Şimdi bunu bunları bir süre bekleyelim Arkadaşlar bu bir süre cpu kullanımı yaratsın bu makinalarda ve hemen ardından da auto scaling Grup Bu yüzde otuzu geçtiği zaman ve belirli bir süre devam ettiğinde bunları tespit edecek ve bu tespit ettikten sonra da burada yeni sanal makineler yaratmaya başlayacak şimdi burada videoyu durduruyorum o işlemler başladığı zaman geri dönüp sizlere bunları göstereceğim eve tekrar geri döndüm Şu anda ortalama cpu yüzde 30 üzerine çıktı Aslında ben bir hata yaptım gittim sanal makinalardan bir tanesinden bağlantıyı kopardım O yüzden oradaki stres çalışmadı ama diğer makine de şu anda ekranda görebildiğiniz üzere yüzde yüze yakın bir cpu kullanımı oluştu bu cpu kullanımı ikisinin ortalaması da yüzde 30'un üzerinde olduğundan dolayı hemen iki tane sanal Makina yarattığı Çünkü ortamda bu yükü dengelemesi gerekiyor ve biz auto scaling gruba bunu söylemiştik gelin bakın auto scaling Grup sa geçelim ve hemen ardından da Activity history oraya gelem ve launch instance diyor Bu yeni bir tane instance' yarattı Daha doğrusu 2 tane instance' yarattı ve burada instance sayısını 4'e çıkardı ikiden dörde çıkartıyorum diye de bize uyarı verdi İşte arkadaşlar auto scaling grup bu tarz işlemleri otomatize etmemizi sağlayan servisin adı böylelikle de EC2 konusunda sonuna gelmiş olduk Yani tüm teoriyi gördük ardından da burada 3 bölüm şeklinde uygulamasını yaptık ve EC2 ile ilgili tüm kavramları görmüş olduk şimdi sırada EC2 ile bağlantılı ama ayrı bir servis olan efs var ve bir sonraki bölümde bu nefes servisini göreceğiz Bir sonraki bölümde görüşmek üzere arkadaşlar