

Bulut Depolama Türleri

- **Doğrudan Bağlı Depolama (Direct Attached Storage):** Bulut tabanlı sunucuya doğrudan sunulan depolama. Genellikle işletim sistemi (operating system) için kullanılır ve geçici (ephemeral) bir yapıya sahiptir.
- **Dosya Depolama (File Storage):** Standart ethernet ağı üzerinden sunucu düğümlerine (compute nodes) bağlanan depolama. NFS (Network File System) kullanır ve birden fazla sunucu tarafından erişilebilir.

Blok Depolama (Block Storage)

- Yüksek hızlı fiber bağlantılarla sunucu düğümlerine sunulan depolama. Veritabanları (databases) gibi yüksek hız gerektiren uygulamalar için uygundur. Genellikle "volumes" (hacimler) olarak sağlanır ve yalnızca bir sunucuya bağlanabilir.

Nesne Depolama (Object Storage)

- Bir API (Application Programming Interface) üzerinden erişilen depolama türüdür. En ucuz ve en yavaş depolama türüdür, ancak sınırsız (infinite) boyut sunar. Belge (documents), video, yedekleme (backups) gibi yapılandırılmamış veri türleri için idealdir.

Veri Yedekleme (Data Backup)

- Dosya ve blok depolama için anlık görüntü (snapshot) alma yöntemi. Anlık görüntüler, depolamanın belirli bir anındaki görüntüsünü alır ve veri kaybını önlemek için kullanılabilir.

File Storage Nedir?

- **File Storage**, bir **compute node** (hesap düğümü) ile bağlantılı olarak erişilebilen ve veri depolamak için kullanılan bir depolama türüdür.
- **Direct attached storage** (doğrudan bağlı depolama) ile karşılaştırıldığında, genellikle daha az maliyetli, daha dayanıklı ve daha az disk yönetimi gerektirir.

Depolama Yapısı ve Güvenlik

- **File Storage**, fiziksel disklerin ayrı bir donanımda bulunduğu **storage appliances** (depolama cihazları) üzerinden bağlanır.
- Bu cihazlar, veri güvenliğini sağlamak için **encryption in transit** (iletim sırasında şifreleme) gibi hizmetler sunar.

Ağ ve Performans

- **File Storage**, bir **ethernet network** (ethernet ağı) üzerinden bağlanır ve bu ağ, e-posta almak veya interneti gezmek için kullanılan aynı türdendir.
- Ancak, ethernet ağlarının hızı değişkenlik gösterebilir; bu nedenle, **File Storage** genellikle yüksek hız gerektirmeyen iş yükleri için kullanılır.

Uygulama Alanları

- **File Storage**, birden fazla **compute node** (hesap düğümü) ile aynı anda bağlanabilir, bu da onu ortak depolama gereksinimleri için ideal hale getirir.
- Örnek uygulamalar arasında **departmental file share** (departman dosya paylaşımı) ve **landing zone** (geliş alanı) yer alır.

IOPS Kapasitesi

- **IOPS** (Input/Output Operations Per Second), disklerin veri okuma ve yazma hızını ifade eder.
- Uygulamanız için yeterli IOPS değeri sağlanmazsa, depolama bir darboğaz oluşturabilir ve uygulamanızın yavaş çalışmasına neden olabilir.

Block Storage (Blok Depolama) ile **File Storage** arasındaki farkları ve hangi durumlarda birinin diğerine tercih edileceğini ele alacak bir sonraki bölüme geçiş yapmaktadır.

Block Storage Nedir?

- **Block Storage**, verileri **chunks** (parçalar) halinde depolar ve her bir parçayı benzersiz bir adres altında saklar.
- **Compute node** (hesap düğümü) ile bağlantı gerektirir ve genellikle yüksek hızlı **fiber optic networks** (fiber optik ağlar) üzerinden bağlanır.

Performans ve Güvenlik

- **Block Storage**, düşük gecikme (low latency) gerektiren uygulamalar için idealdir, bu nedenle **databases** (veritabanları) ve **mail servers** (posta sunucuları) gibi yüksek performans gerektiren iş yükleri için tercih edilir.
- **Encryption at rest** (dinlenme sırasında şifreleme) ve **encryption in transit** (iletim sırasında şifreleme) gibi güvenlik özellikleri sunar.

File Storage ile Karşılaştırma

- **File Storage**, birden fazla **compute node** (hesap düğümü) ile aynı anda bağlanabilirken, **Block Storage** yalnızca bir **compute node** ile bağlanabilir.
- **File Storage**, daha düşük maliyetli ve daha esnek bir çözüm sunar, ancak hız değişkenliği gösterebilir.

IOPS Kapasitesi

- Her iki depolama türü için de **IOPS** (Input/Output Operations Per Second) kapasitesinin dikkate alınması önemlidir.
- Bulut sağlayıcıları, depolama sağlarken IOPS özelliklerini belirlemenize ve gerektiğinde ayarlamanıza olanak tanır.

Object Storage Nedir?

- **Object Storage**, belirli bir **compute node** (hesap düğümü) ile bağlantı gerektirmeden veri depolamak için kullanılan bir hizmettir. Veriler, bir **API** (Uygulama Programlama Arayüzü) aracılığıyla yüklenir ve yönetilir.

- Maliyet açısından, genellikle diğer bulut depolama seçeneklerinden daha ucuzdur; gigabayt başına maliyet genellikle birkaç ABD sentidir.

Sınırsız Depolama Kapasitesi

- **Object Storage**, sınırsız bir depolama kapasitesine sahiptir. Kullanıcılar, ihtiyaç duydukları kadar veri yükleyebilir ve yalnızca kullandıkları gigabayt için ödeme yaparlar.
- **Buckets** (kova) adı verilen yapılar içinde veriler depolanır. Her kova, anlamlı isimler alabilir ve farklı nesne türleri için farklı kovalar oluşturulabilir.

Metadata ve Erişim

- Her nesne, bir **metadata** (veri hakkında veri) ile birlikte depolanır. Bu, nesnenin konumunu bulmak ve erişmek için yardımcı olur.
- Kovalar, belirli bir boyut tanımlamadan oluşturulabilir ve depolama sağlayıcısı yeterli kapasiteyi garanti eder.

Dayanıklılık ve Erişilebilirlik

- Bulut sağlayıcıları, farklı dayanıklılık seviyelerine sahip kovalar sunar. Örneğin, bazıları yalnızca bir veri merkezinde depolama yaparken, diğerleri verileri birden fazla veri merkezinde saklar.
- Yüksek erişilebilirlik ve dayanıklılık sunan seçenekler genellikle daha maliyetlidir.

Uygulama Alanları

- **Object Storage**, büyük miktarda yapılandırılmamış veriyi depolamak için idealdir. Statik veriler (örneğin, metin dosyaları, ses ve video dosyaları) için uygundur.
- Ancak, işletim sistemleri veya veritabanları gibi dinamik içeriklerin depolanması için uygun değildir.

Bu video, **Object Storage**'ın temel özelliklerini ve diğer depolama türleriyle olan farklarını özetlemektedir.

Bu video, **Object Storage** (Nesne Depolama), **Data Tiers** (Veri Katmanları) ve **Object Storage APIs** (Nesne Depolama API'leri) üzerine odaklanmaktadır.

Depolama Katmanları

- **Standart Katman** (Standard Tier): Sık erişilen nesnelerin depolandığı yerdir ve genellikle en yüksek gigabayt başına maliyete sahiptir.
- **Arşiv Katmanı** (Archive Tier): Ayda bir veya iki kez erişilen belgelerin depolandığı, daha düşük maliyetli bir depolama seçeneğidir.

Erişim ve Maliyet

- **Soğuk Arşiv Katmanı** (Cold Vault Tier): Yılda sadece bir veya iki kez erişilen verilerin depolandığı, çok düşük maliyetli bir seçenektir.

- **Otomatik Arşivleme** (Automatic Archiving): Erişilmeyen nesnelerin daha ucuz bir depolama katmanına otomatik olarak taşınmasını sağlayan kurallar oluşturulabilir.

API ve Kullanım

- **API** (Application Programming Interface): Nesne depolama, bir API aracılığıyla erişilir; en yaygın olanı **S3 API**'dir.
- **Veri Kurtarma** (Data Retrieval): Nesne depolama, dosya veya blok depolama ile karşılaştırıldığında daha yavaş bir erişim sunar ve bazı durumlarda veri kurtarma saatler alabilir.

Yedekleme ve Kurtarma

- **Yedekleme** (Backup) ve **Felaket Kurtarma** (Disaster Recovery): Nesne depolama, off-site tape (şerit) çözümlerinin yerini alarak veri kurtarma süresini azaltabilir.

CDN Nedir?

- **Content Delivery Network** (CDN), kullanıcıların coğrafi konumlarına göre web sitesi içeriğinin önbelleğe alınmış (cached) kopyalarını sunan dağıtılmış bir sunucu ağıdır.
- CDN, web sitenizin daha hızlı yüklenmesini sağlar ve kullanıcı deneyimini iyileştirir.

CDN'in Çalışma Prensipleri

- Kullanıcılar, uzak bir sunucuya (örneğin, Dallas'taki bir sunucuya) erişmeye çalıştıklarında, bu mesafe gecikmelere (latency) neden olur. Örneğin, Sydney'den Dallas'a olan mesafe yaklaşık 8,600 mil (13,840 km)'dir.
- CDN, içeriği dünya genelinde birçok noktada depolayarak, kullanıcıların en yakın sunucudan (edge server) içeriği almasını sağlar. Bu, gecikmeyi (latency) azaltır.

Faydaları

- **Hız** (Speed): Kullanıcılar, içeriği daha hızlı alır çünkü en yakın sunucudan (edge server) erişim sağlanır.
- **Trafik Azaltma** (Traffic Reduction): Dallas'taki ana sunucu üzerindeki yük azalır, bu da daha iyi bir performans sağlar.
- **Güvenlik** (Security): Kullanıcılar doğrudan ana sunucu ile iletişim kurmadığı için, güvenlik riskleri (security risks) azalır.