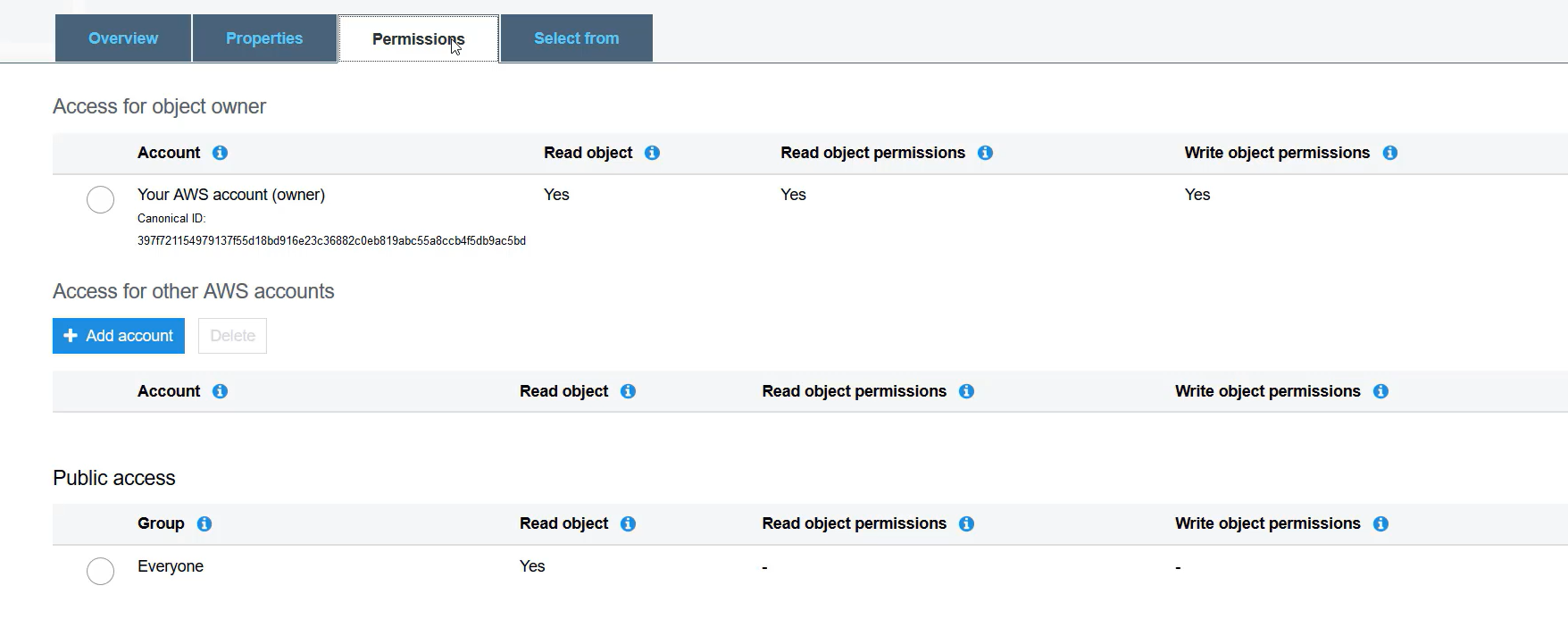


Permission (yetkiler)





yetki 3 tane dir;

ilki bu aws hesabındaki diğer kullanıcılar

bir başka aws hesabında kullanıcılara yetkilendirme

sonuncusu herkese açık bütün dünyadan erişim.

**Bucket üzerinde işlemler; (47. Video 13.dk)**

**Bucket Versioning**

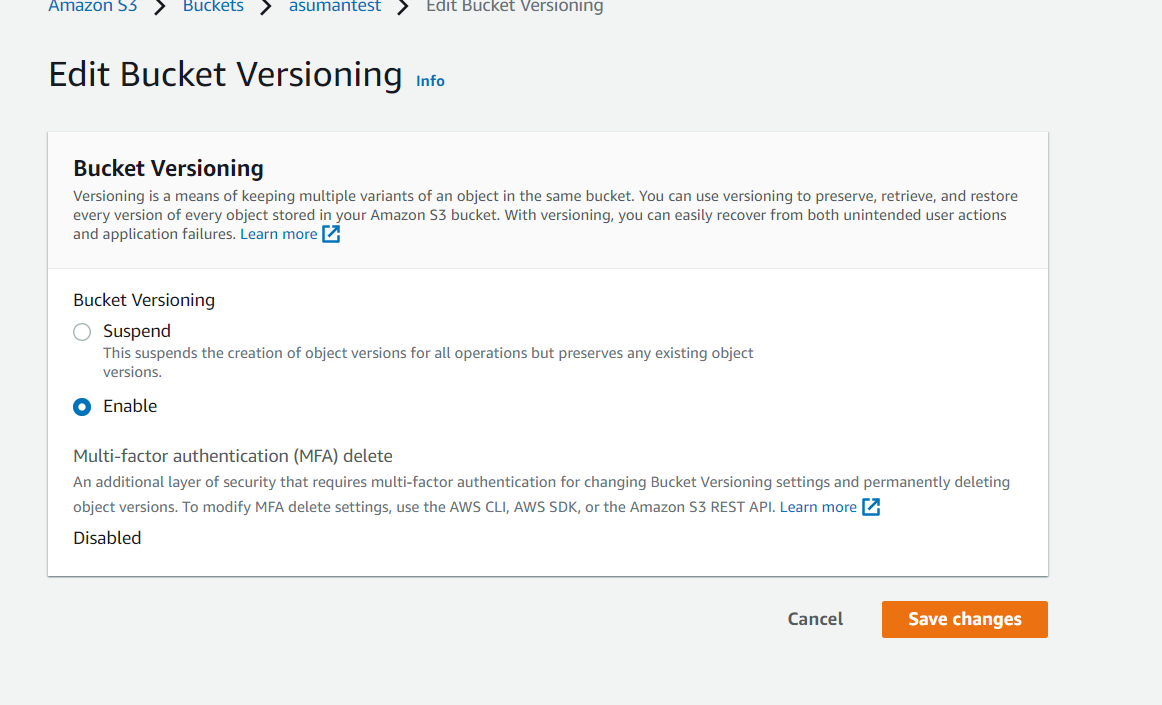
**Enable seçersek bu bucket üzerinde versiyonlama yapmış oluruz.**

Versiyonlamayı bir Bucket üzerinde açtığımız zaman tekrar kapatamıyoruz sadece suspend edebiliyoruz. Sadece durdurabiliriz.

Versiyonlama nedir; s3 bucket bizim dosya object tabanlı dosya depomuz.

Sakladıklarımız yanlışlıkla silindi nasıl geri döndürürüz ? dönmesi imkansız. Bu nedenle s3 ortaya çıktığında beri versiyonlama var. İşte bu versiyonlama her upload her dosyada sürekli tekrarlar versiyonları.

Delete marker denen bir işaret verecek, bu dosyayı çağırsakda gelmeyecek ama bu dosya bucketta bu versiyonlama vesilesi ile saklanmaya devam edecek. Geri getirmek istersek de delete marker ı silebiliriz.



**SERVER ACCESS LOGGİNG**; erişime açık olan objelere kim ne zaman erişmiş vb.durumları yani log kayıtlarını almamızı sağlar.

**STATİC WEBSİTE HOSTİNG;** S3 de dinamik websitelerini tutamayız çünkü S3 bir sunucu değil. (dinamik web sitesi ; facebook gibi arkasında sürekli işlemler olan.) S3 de static web sitelerini tutariz.

**Web sitelerini EC2 de bir sanal makine oluşturup mesela APACHI gibi bir uyg kurup php tabanlı dinamik web sitelerini tutacağız.**

**OBJECT LOCK; Bu object lock u sadece S3 bucket oluştururken devreye alabiliyoruz. Sonradan alamıyoruz.**

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

TAGS; bütçelendirme detayları, faturalandırma detayları yapmamıza olanak sağlamaktadır (departman-satış, callcenter – 120, vb. açmıştık)

Bu tag ı kullanmadan gidip obje bazından yukarıya çıkartarak bucket atayıp sonrada bunu devreye almamız gerekiyor (48.video 2,56 sn)

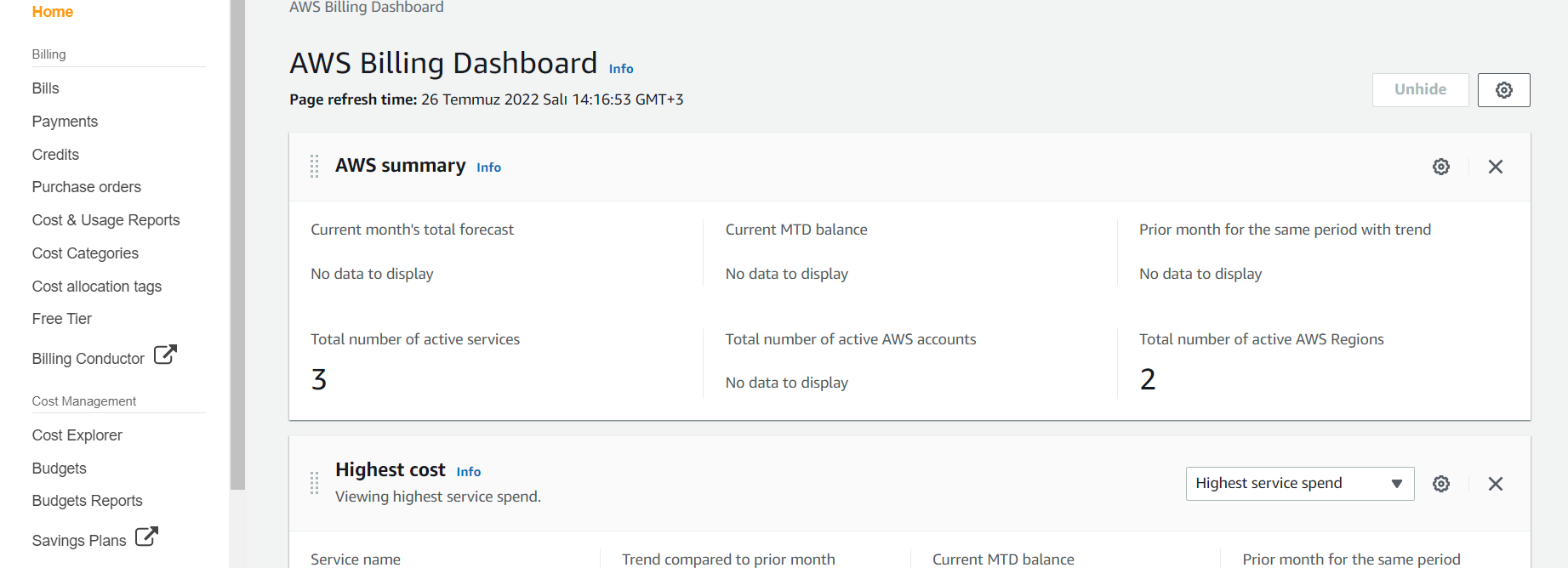
Tag lerin 2 kullanım alanı vardır;

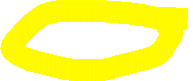
1)dosyaları farklılaştırmak istersek, yani bu aws kaynaklarını farklılaştırmak istersek ve farklı şekilde bir arada toplamak istersek (resource groups konusunda anlatılmıştı)

2) Objelerin üzerine etiket atarken bahsedilmişti; bunları faturalandırma da kullanıyoruz. Faturalandırma detaylarında kullanabiliyoruz.

Faturalandırma detayında kullancaksak Bunu yaparken izlenecek yol;

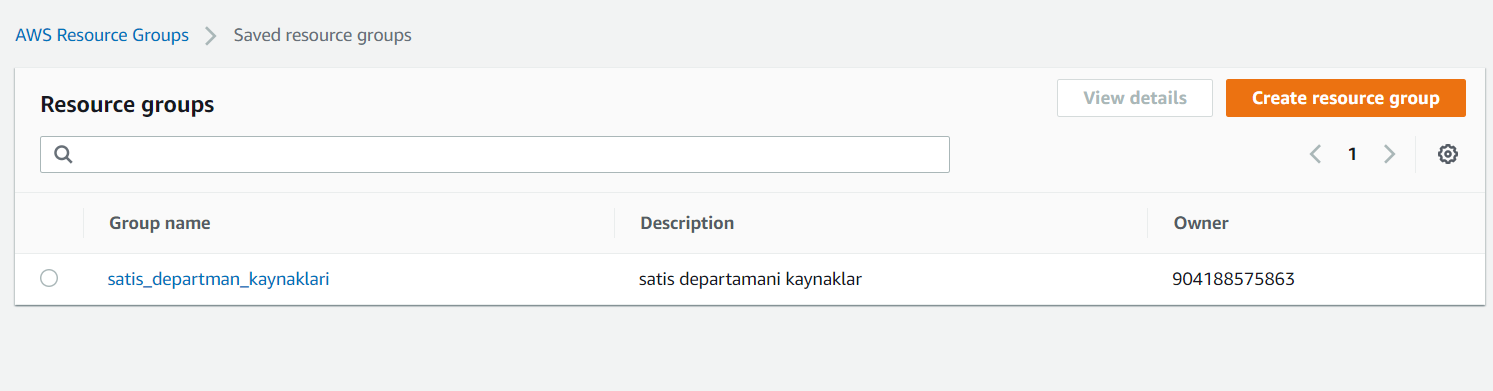
Önce bu Tag Bucket üzerinde devreye alırız 🡪 bucket a bu etiketi atarız 🡪 Billing Dashbord a gelip 🡪 oradan da Cost Allocation Tags ı seçeriz





Tag ataması yaptıktan sonra Resource Group a gelip bir group oluşturup

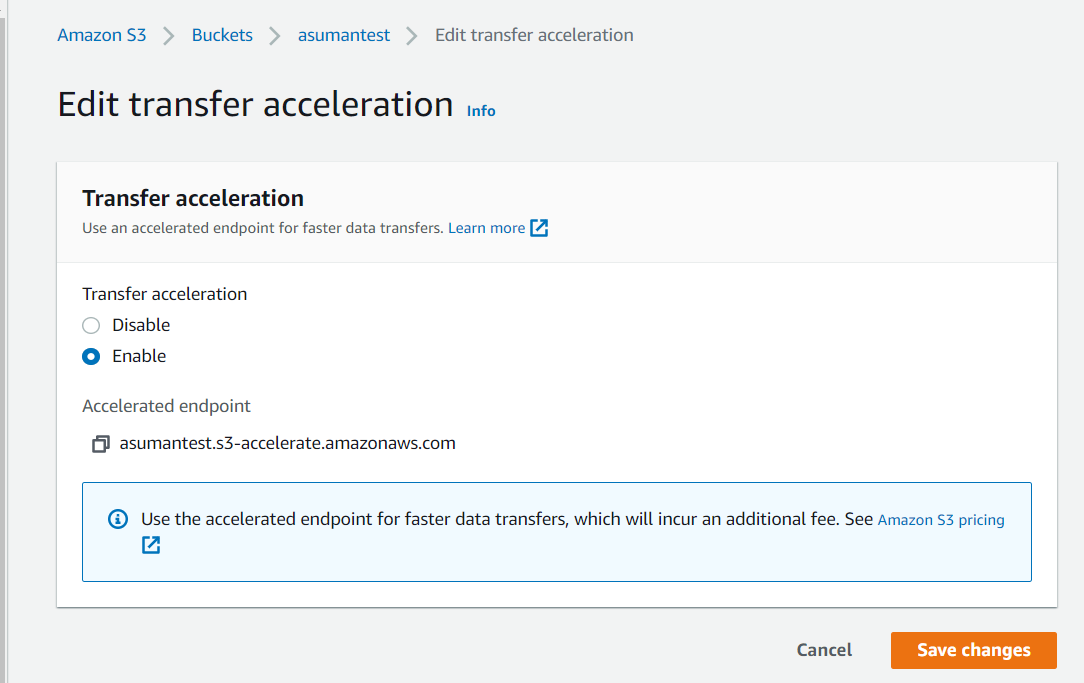
Departmanı IT olan bütün kaynakları listeleyebiliriz. Tag leri bunun için kullanabiliyoruz ya da daha sonra bir uygulama yazıp bu uygulamada bu tag e göre çeşitli şeyler yapabiliyoruz. Tag ler bizim aws kaynaklarında listelemeler yapmamızı sağlar.



**TRANSFERS ACCELERATİON;** S3 bucketi oluşturduk, bu S3 bucket ina 2 terabyte dosya göndereceğiz. Ben Ankara dayım bucket İrlanda dayım. Normalde günler sürecek bir işlem.

Bunun yerine AWS bize TRANFERS ACCELERATİON 🡪 GELELİM

Bunu Enabled yapalım. Bunu s3 bucket URL inden göndermek yerine ENDPOİNT i kullanarak gönderebiliriz. Çok kısa sürede gönderebiliriz, ücrete tabi olarak.

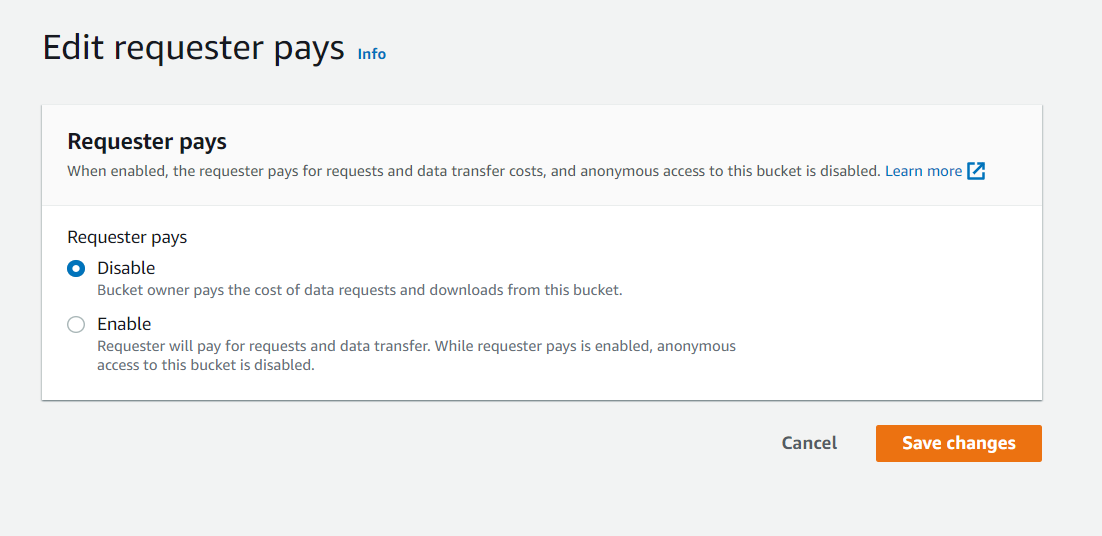


**EVENTS; S3 bucket ına bir dosya upload ettik diyelim. Bu upload işlemi bitince biz istiyoruz ki LAMBDA FONK. Tetiklensin.**

**Lambda fonk. İse bu uploda edilen değişikliği alacak bunun bir 100\*100 piksel bir kopyasını çıkarıp bunu başka bir bucket a yazacak.**

**S3 bucket ındaki bir işlemin AWS servisdeki başka bir işlemi tetiklemesini istiyorsak bu EVENTS i kullanıyoruz.**

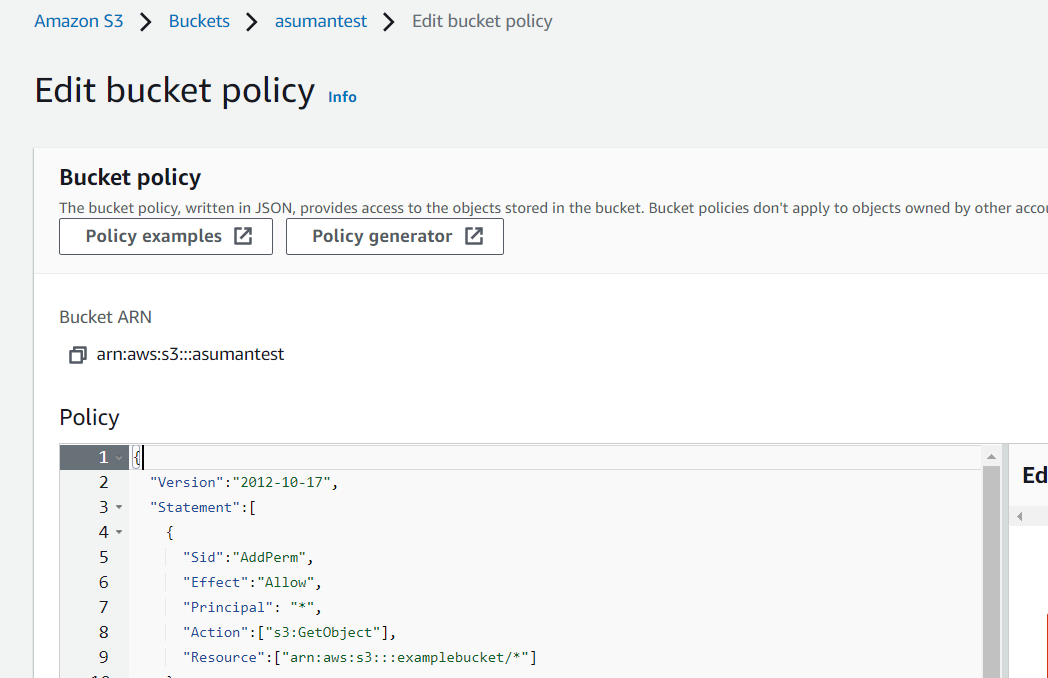
**REQUESTER PAYS;** Bu özellik ile birlikte kullanıma açtığımız dosyayı download edenlerden belli bir ücret alınır. Bu durumda kullanıma açtığımız dosya public ise AWS bunu kapatır. Bu daha çok AWS İN kendisi S3 üzerinden kendi kullanıcıları için bir dosya sunduğunda kullancılardan belli bir ücret karşılığı download etmesine izin vermek için kullanmaktadır. Aşağıda görüldüğü Enable konuma getirmemiz gerekiyor.

****

**PERMİSSİONS**

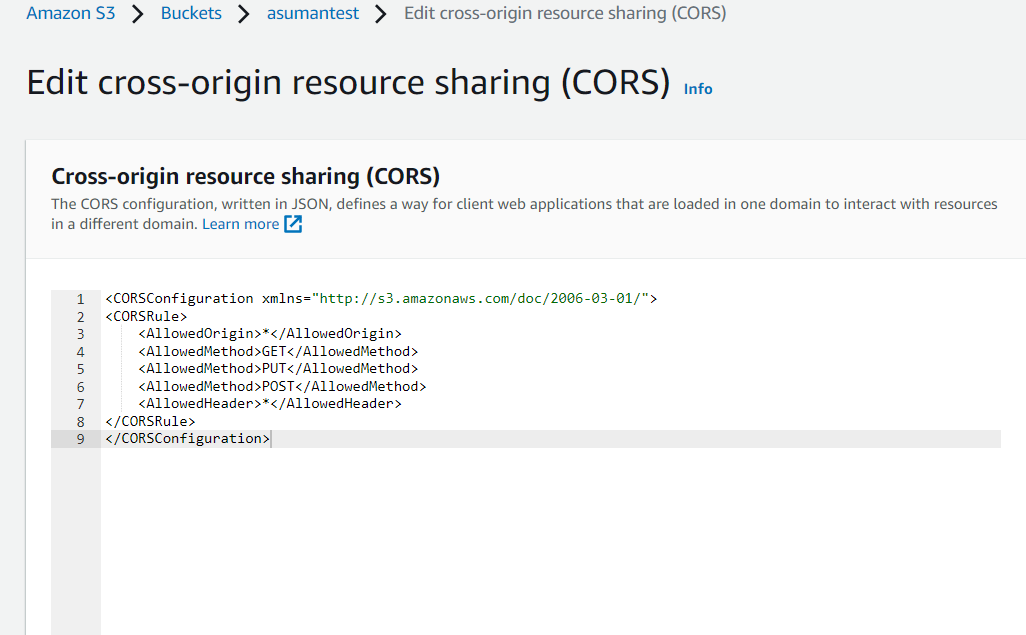
**Access Kontrol List;** Bucket ı kimlerin erişebileceğinin ayarlandığı kısımdır. Access Control List Basit bir yetki işidir.

Karmaşık bir erişim kontrolü yapacaksak bunu **BUCKET POLİCY** den yapacağız.

****

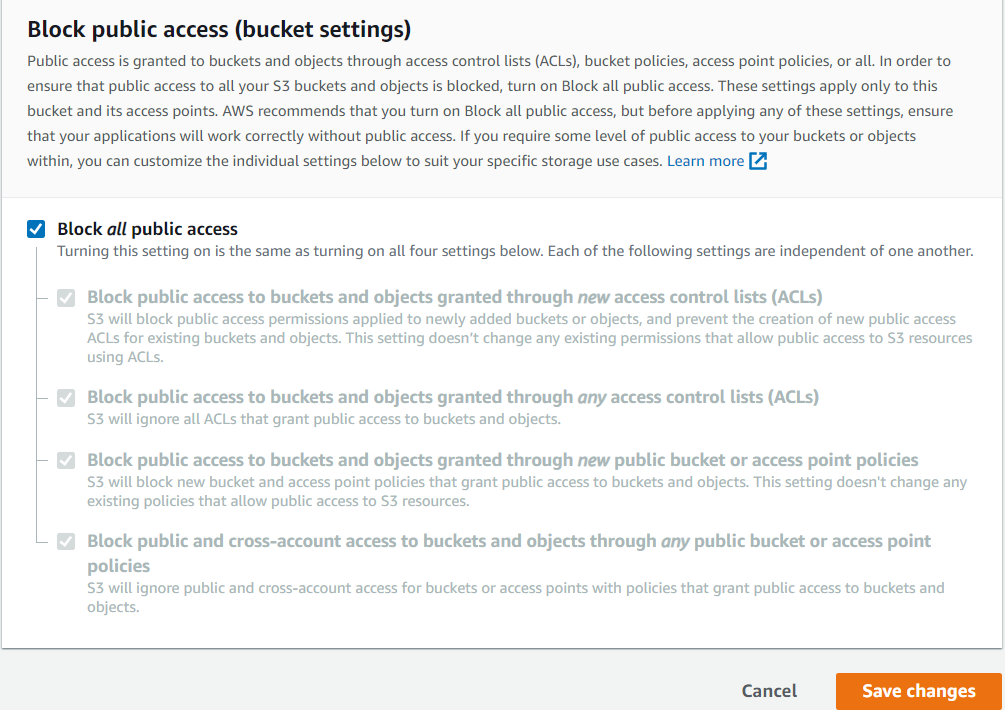
**Cors izinleri;**

[**www.a**](http://www.a) **sitesinden yayınlanan dokümanın** [**www.b**](http://www.b) **sitesinden erişimi anında S3 ün bir sunucu gibi davranmasıdır. Bağlanmak isteyen sitenin izninin olup olmadığını kontrol eden bu kısımdır.**

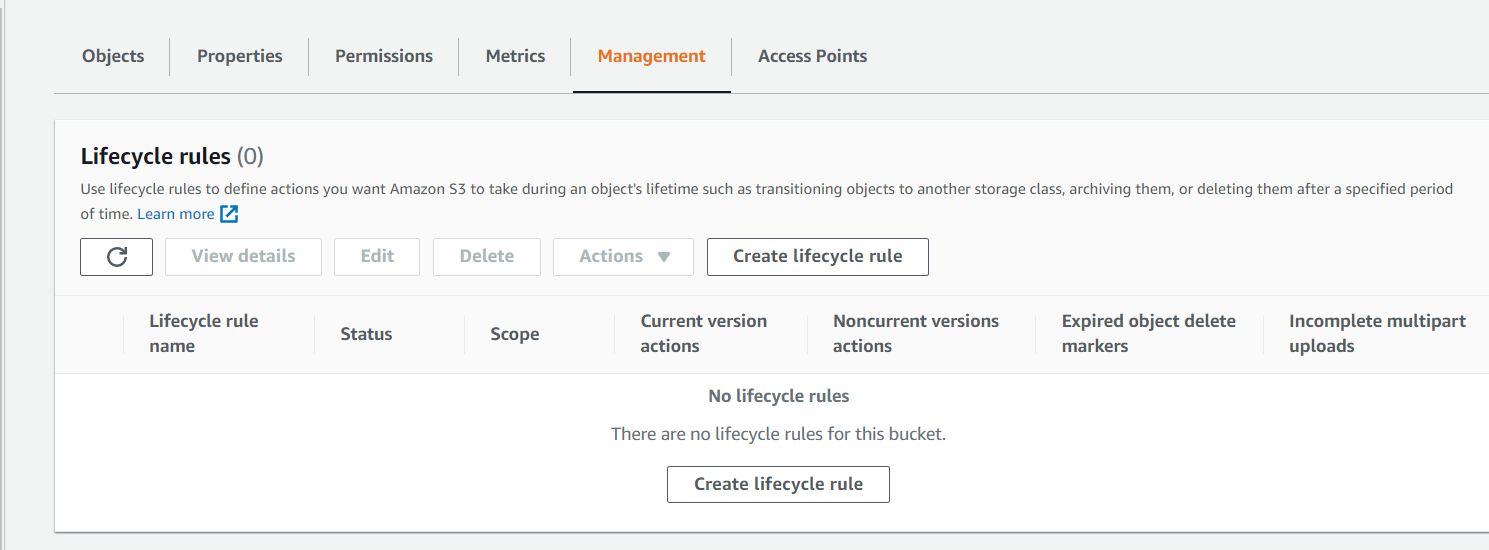
****

**Block public access (bucket settings)**

Yaşanan sıkıntılardan dolayı public Access artık kapatıldı. Hacker lar nedeniyle firma sahipleri çok sorun yaşadığı için artık default olarak kapalı geliyor.

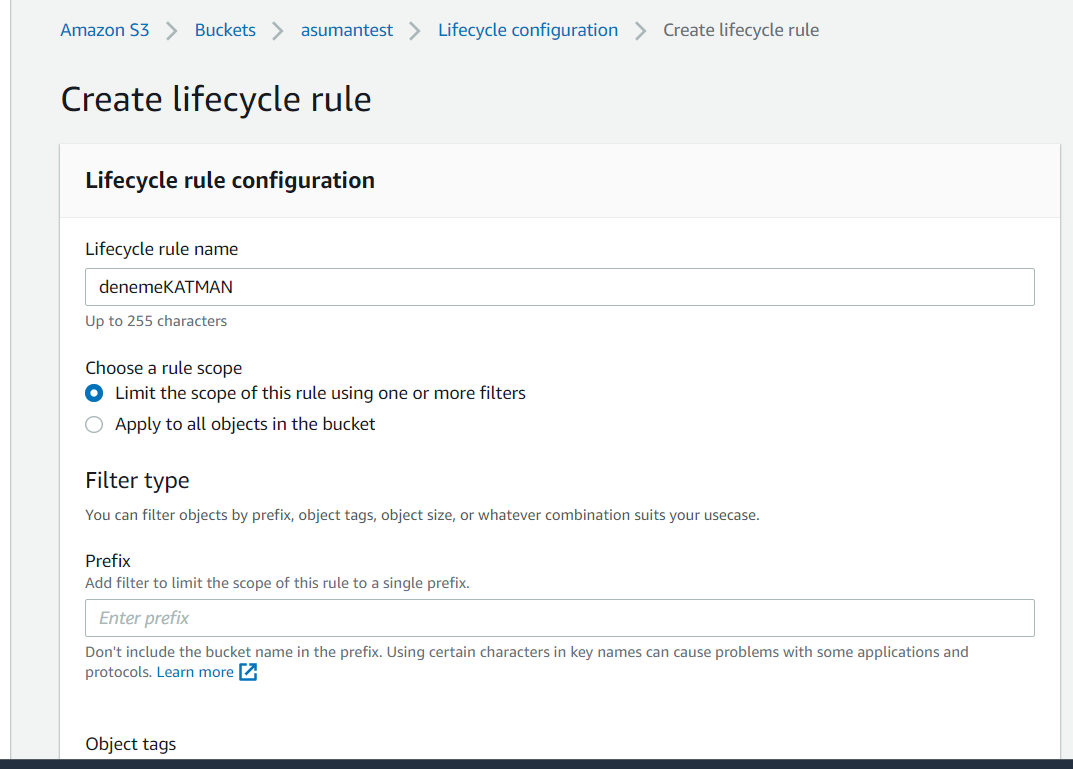
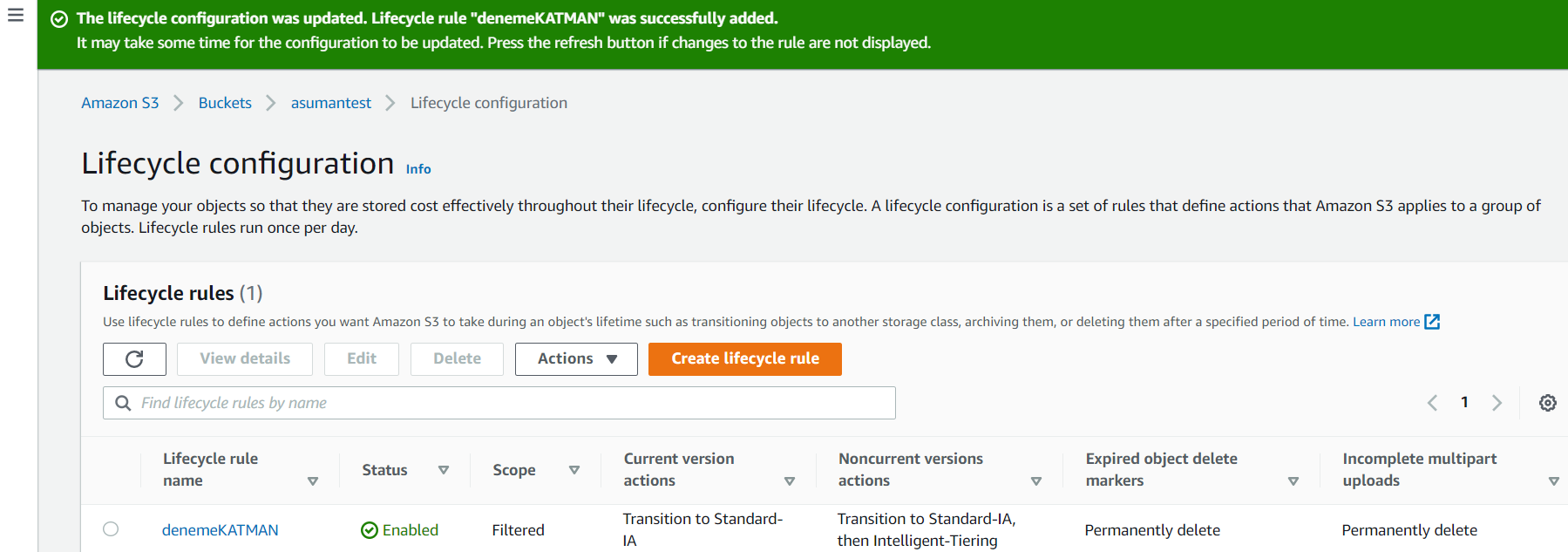


MANAGEMENT

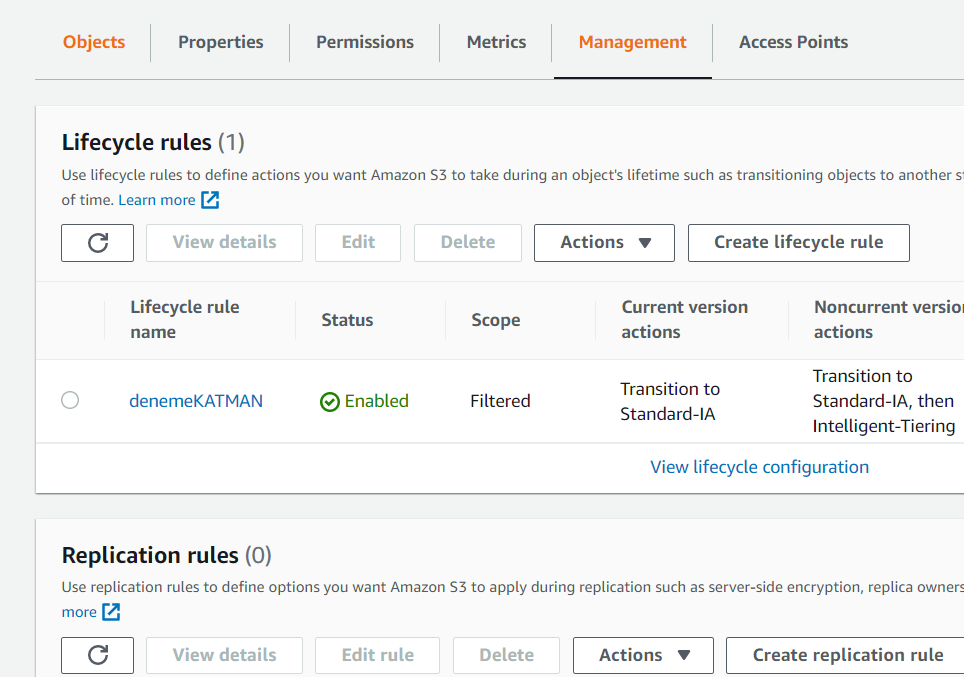


LİFECYCLE;

Biz objelerin katmanlar arasında gezmesini(standart, glaceir gibi) otomatik kurallara bağlayabilirdik, lifecycle bunu otomatik olarak yapar.



Replication rules; her S3 bucket bir region da olur ama biz bucket ı birden fazla region ı expent edip genişletemiyoruz. Ama aws şöyle bir imkan tanıdı. Bir regionda bucket oluştur sonra git Amerika daki region da da bir başka bucket oluştur. Daha sonra gel bu REPLİCATİON da bir kural ekle bu bucket a atılan her dosyayı düzenli olarak Amerikada ki bucket a kaydet. Yedekleme için felaket senaryoları için çok etkili bir çözümdür. Buradaki konu kural farklı region lar olmasıdır.



<https://www.udemy.com/course/bulut-bilisim-temelleri-ve-aws-cozum-mimarligina-giris/learn/lecture/14315024#questions/12169660>

<https://www.udemy.com/course/bulut-bilisim-temelleri-ve-aws-cozum-mimarligina-giris/learn/lecture/14315024#search>

<https://docs.aws.amazon.com/IAM/latest/UserGuide/tutorial_billing.html>

