

## 11.Doküman

## Erişim Yetkileri

Konu Etiketleri

yetkilendirme , chmod , chattr

# Erişim Yetkileri

Linux sistemlerinin yapısı gereği güvenlik açısından, dosya ve dizinlere ait birçok kısıtlama ve yetkilendirme ayarları vardır. Bu **yetkilerin hepsine sahip olan tek kullanıcı** ise **root** kullanıcısıdır. Bu yüzden sistemi kullanırken eğer root kullanıcısı isek önümüzde uyarıcı bizi kısıtlayıcı bir mekanizma olmayacağından, kimi durumlarda yapacağımız değişiklikler sistemle ilgili büyük sorunlara yol açabilir. Bu yüzden root kullanıcısıyken yapılan işlemlere dikkat etmek gerekir.(Hatta kimi dağıtımlarda root kullanıcı hesabı varsayılan olarak engellenmiştir, ancak istenildiği takdirde ve gereken ayarlar yapıldığında kullanılabilir duruma getirilir.)

Bu girizgahtan sonra, her kullanıcının kendine verilen yetkiler çerçevesince hareket edebildiğini öğrenmiş olduk. Kullanıcıların dosya veya dizinler ile ilgili yapabileceği üç eylem bulunmaktadır. Bunlar;

**okuma(r):** Klasör listesini ve dosya içeriğini görüntüleme.

**yazma(w):** Dosya veya klasör üzerinde değişiklik yapma.

**çalıştırma(x):** Hedef dosyayı çalıştırma veya klasör içerisine erişme.

Aslında bu eylemler sizlere yabancı gelmemeli. Zira daha önce **chmod** komutu yardımı ile gerekli dosyanın iznini değiştirmiş ve dosyamızı çalıştırmayı başarmıştık.

Şimdi mevcut dosyalarımızın ne tür izinleri barındırdığına göz atalım. Bunun için komut satırına ayrıntılı liste yazdırmak üzere **ls -l** komutumuzu veriyoruz.

```

root@taylan:~# ls -l
toplam 32
drwxr-xr-x 3 root root 4096 Eki  3  08:43 Desktop
drwxr-xr-x 2 root root 4096 Eki 26  2018 Documents
-rw-r--r-- 1 root root  0 0ca 15  07:50 dosya1
-rw-r--r-- 1 root root  0 0ca 15  07:50 dosya10
-rw-r--r-- 1 root root  0 0ca 15  07:50 dosya2
-rw-r--r-- 1 root root  0 0ca 15  07:50 dosya3
-rw-r--r-- 1 root root  0 0ca 15  07:50 dosya4
-rw-r--r-- 1 root root  0 0ca 15  07:50 dosya5
-rw-r--r-- 1 root root  0 0ca 15  07:50 dosya6
-rw-r--r-- 1 root root  0 0ca 15  07:50 dosya7
-rw-r--r-- 1 root root  0 0ca 15  07:50 dosya8
-rw-r--r-- 1 root root  0 0ca 15  07:50 dosya9
drwxr-xr-x 2 root root 4096 Kas 15  2018 Downloads
-rw-r--r-- 1 root root  0 0ca 15  08:04 find.py
drwxr-xr-x 2 root root 4096 0ca 15  09:17 Music
drwxr-xr-x 2 root root 4096 0ca 15  05:23 Pictures
drwxr-xr-x 2 root root 4096 Eki 26  2018 Public
-rw-r--r-- 1 root root  0 0ca 15  08:04 resim.jpg
-rw-r--r-- 1 root root  0 0ca 15  08:04 rss.py
-rw-r--r-- 1 root root  0 0ca 15  08:04 sonuc.py
drwxr-xr-x 2 root root 4096 Eki 26  2018 Templates
-rw-r--r-- 1 root root  0 0ca 15  08:04 test.py
drwxr-xr-x 2 root root 4096 Eki 26  2018 Videos

```

Şimdi listemizi inceleyerek bulunan izinleri ele alalım.

**drwxr-xr-x** ve **-rw-r--r--** şeklinde gördüğümüz kısımlar dosya izinlerini ifade ediyor.

Bazı ifadelerin başında olan **d** harfi o ifadenin dizin olduğunu belirtiyor.

Geriye kalan kısımları ayrı ayrı açıklayacak olursak **-** işareti ile ayrılan kısımlar o izine sahip kullanıcı grubunu temsil ediyor. Daha iyi anlamak için **d** harfi hariç **-** işaretini ayırdığımız zaman geri kalan harfleri üç adet üçlü grup haline getirelim;

**rw-r--r--** = **rw** **r-x** **r-x**

**rw-r--r--** = **rw-** **r--** **r--**

Sırayla; birinci harf kümesi dosya sahibinin izinlerini, ikinci harf kümesi grup izinleri ve son küme de diğer kullanıcıların izinlerini belirtir.

d	rw	x	r	-	x	r	-	x	2	root	root	4096	Apr	26	2017	Videos
-	rw	-	r	-	-	r	-	-	1	root	root	0	Jan	9	17:20	deneme
-	rw	-	r	-	-	r	-	-	1	root	root	0	Jan	9	17:20	'deneme
-	rw	-	r	-	-	r	-	-	1	root	root	16	Jan	9	17:22	dosya
-	rw	-	r	-	-	r	-	-	1	root	root	2161	Jan	9	18:07	liste
-	rw	-	r	-	-	r	-	-	1	root	root	0	Jan	9	17:07	metni

Buna göre yukarıdaki dosyalarda bulunan izinleri açıklayacak olursak;

**r** : okuma yetkisi

**w** : yazma yetkisi

**x** : çalıştırma yetkisi

**rw** : dosyanın sahibi olan kullanıcı okuyabilir, yazabilir, çalıştırabilir.

**r-x** : dosya sahibi kullanıcı grubu ile aynı gruba dahil kullanıcılar okuyabilir, çalıştırabilir fakat yazamaz(değişiklik yapamaz).

**r-x** : diğer kullanıcılar okuyabilir, çalıştırabilir fakat yazamaz(değişiklik yapamaz).

## Yetkilerin Değişimi(chmod)

Erişim yetkisini değiştirme işlemini, ancak en yetkili kişi olan root yapabilir. Bu değişim işlemi; daha öncede de kullandığımız **chmod** komutu sayesinde gerçekleştirilir.

**chmod** komutunun parametrelerini tanıyarak örnek verme işlemine geçelim.

**u** : Dosya-dizinin sahibi

**g** : Dosya-dizinin sahibi ile aynı grupta bulunan kullanıcılar

**o** : Diğer kullanıcılar

**a** : Herkese açık.

**=** : Yetki eşitleme

**+** : Yetki ekleme

**-** : Yetki çıkarma

Genel parametreleri gördüğümüze göre gelin birkaç örnek yapalım.

Örnek göstermek adına anlatımı, içerisindeki dosyaların hiç birinde yetkinin bulunmadığı bir klasör üzerinden gerçekleştireceğim.

İlk olarak klasörde yer alan dosyaların herhangi bir yetkiye sahip olmadıklarını teyit etmek için ayrıntılı çıktı almak üzere **ls -l** komutunu kullandık.

```

root@taylan:~/Desktop/yeni_dizin# ls -l
total 28
----- 1 root root 4755 Jan 10 16:44 filazof
----- 1 root root 72 Jan 10 16:46 metin
----- 1 root root 72 Jan 10 16:46 metin_belge
----- 1 root root 72 Jan 10 16:46 metinler
----- 1 root root 28 Jan 10 16:49 test.txt
d----- 3 root root 4096 Jan 10 16:44 yeni
root@taylan:~/Desktop/yeni_dizin#

```

Daha sonra klasörde yer alan tüm dosyalara `*` joker karakteri ile ulaştık ve `chmod +w *` komutumuzu kullanarak **herkese açık** olacak şekilde **yazma(w)** yetkisi verdik.

```

root@taylan:~/Desktop/yeni_dizin# chmod +w *
root@taylan:~/Desktop/yeni_dizin# ls -l
total 28
--W----- 1 root root 4755 Jan 10 16:44 filazof
--W----- 1 root root 72 Jan 10 16:46 metin
--W----- 1 root root 72 Jan 10 16:46 metin_belge
--W----- 1 root root 72 Jan 10 16:46 metinler
--W----- 1 root root 28 Jan 10 16:49 test.txt
d--W----- 3 root root 4096 Jan 10 16:44 yeni
root@taylan:~/Desktop/yeni_dizin#

```

Aynı grupta bulunan kullanıcılar için, yine konumumuzda bulunan tüm dosyaları `*` sayesinde kapsayacak şekilde `g+rx *` komutumuzu verdik.

```

root@taylan:~/Desktop/yeni_dizin# chmod g+rx *
root@taylan:~/Desktop/yeni_dizin# ls -l
total 28
--w-r-x--- 1 root root 4755 Jan 10 16:44 filazof
--w-r-x--- 1 root root 72 Jan 10 16:46 metin
--w-r-x--- 1 root root 72 Jan 10 16:46 metin_belge
--w-r-x--- 1 root root 72 Jan 10 16:46 metinler
--w-r-x--- 1 root root 28 Jan 10 16:49 test.txt
d--w-r-x--- 3 root root 4096 Jan 10 16:44 yeni
root@taylan:~/Desktop/yeni_dizin#

```

Gruptaki kullanıcılara(g), okuma-yazma-çalıştırma yetkisi (rwx), kullanıcıya(u) yazma yetkisi(r), diğer kullanıcılara ise yalnızca çalıştırma yetkisi(x) verdik.



```

root@taylan:~/Desktop/yeni_dizin# chmod g+rwX,u+w,o+X *
root@taylan:~/Desktop/yeni_dizin# ls -l
total 28
-rw-rwx--x 1 root root 4755 Jan 10 16:44 filazof
-rw-rwx--x 1 root root 72 Jan 10 16:46 metin
-rw-rwx--x 1 root root 72 Jan 10 16:46 metin_belge
-rw-rwx--x 1 root root 72 Jan 10 16:46 metinler
-rw-rwx--x 1 root root 28 Jan 10 16:49 test.txt
d-rw-rwx--x 3 root root 4096 Jan 10 16:44 yeni
root@taylan:~/Desktop/yeni_dizin#

```

Ve en son yine bulunduğumuz konumdaki tüm dosyaların yetkilerini kaldırdık.

```

root@taylan:~/Desktop/yeni_dizin# chmod a-rwx *
root@taylan:~/Desktop/yeni_dizin# ls -l
total 28
----- 1 root root 4755 Jan 10 16:44 filazof
----- 1 root root 72 Jan 10 16:46 metin
----- 1 root root 72 Jan 10 16:46 metin_belge
----- 1 root root 72 Jan 10 16:46 metinler
----- 1 root root 28 Jan 10 16:49 test.txt
d----- 3 root root 4096 Jan 10 16:44 yeni
root@taylan:~/Desktop/yeni_dizin#

```

Bu kullanımların dışında yetkilendirme işlemleri daha önceden de rastladığımız ve fark etmeden de olsa kullanmış olduğumuz sayısal şekilde de ifade edilebiliyor.

Bu durumu yetkilerin sayısal karşılığını vererek anlatmaya devam edelim.

Yetki kalıplarının sayısal karşılıkları.

**# dosyanın sahibisahibiyle aynı gruptakiler diğer kullanıcılar**

<b>r</b> 4	4	4
<b>w</b> 2	2	2
<b>x</b> 1	1	1

Yetkilerin sayısal değerlerini kullanarak bir örnek yapalım.

Örneğin biz sadece dosyanın sahibine bütün yetkileri vermek istiyoruz diyelim. Bunun için ilk başta yetki

kalıplarının numara karşılıklarını toplamalıyız. Yani bütün yetkileri vereceğimiz için **r=4** + **w=2** + **x=1** = **toplam sayı 7** etti. Bizler de sadece dosya sahibine bu yetkiyi vermek istediğimizden normalde vereceğimiz **chmod rwx-----** komutumuzu diğer kullanıcılara yetki vermek istemediğimiz için o alanları 0 bırakarak komutu **chmod 700 dosya** şeklinde veriyoruz. Böylelikle sadece dosyanın sahibi tüm yetkilere sahip olmuş oluyor.

```

root@taylan:~/Desktop/yeni_dizin/yeni# ls -l
total 8
d----- 2 root root 4096 Jan  7 10:42 en_yeni
----- 1 root root  72 Jan 10 16:46 metin
root@taylan:~/Desktop/yeni_dizin/yeni# chmod 700 metin
root@taylan:~/Desktop/yeni_dizin/yeni# ls -l
total 8
d----- 2 root root 4096 Jan  7 10:42 en_yeni
-rwx--- 1 root root  72 Jan 10 16:46 metin
root@taylan:~/Desktop/yeni_dizin/yeni#

```

Daha net anlaşılması adına bir örnek daha yapalım. Şimdi de; dosyanın sahibine tüm yetkileri, ortak gruptakilere yalnızca yazma yetkisini, diğer kullanıcılara da sadece okuma yetkisini verelim.

Dosya sahibi kullanıcıya verilecek tüm yetkiler için  $r(4)+w(2)+x(1)=7$  sayısını kullanacağız.

Dosya sahibi ile ortak gruptaki kullanıcılar için vereceğimiz yazma yetkisi için **yazma(w)** karakterinin sayısal karşılığı olan **2** sayısını kullanacağız.

Diğer kullanıcılar için vereceğimiz yalnız okuma yetkisi için ise **okuma(r)** karakterinin sayısal karşılığı olan **4** sayısını kullanacağız.

```

root@taylan:~/Desktop/yeni_dizin/yeni# ls -l
total 8
d----- 2 root root 4096 Jan  7 10:42 en_yeni
----- 1 root root  72 Jan 10 16:46 metin
root@taylan:~/Desktop/yeni_dizin/yeni# chmod 724 metin
root@taylan:~/Desktop/yeni_dizin/yeni# ls -l
total 8
d----- 2 root root 4096 Jan  7 10:42 en_yeni
-rwx-w-r-- 1 root root  72 Jan 10 16:46 metin
root@taylan:~/Desktop/yeni_dizin/yeni#

```

Çıktıdan da anlaşılacağı üzere sayısal karşılıklar istediğimiz yetkilendirme işlemi gerçekleştirdi.

Son bir ayrıntı daha verelim. Eğer verdiğimiz izinlerin o dizinle beraber alt klasörlerinde de etkili olmasını istersek komutumuzu **-R** parametresi ile birlikte kullanmalıyız.

Örneğin bulunduğum konumdaki "**metin**" isimli klasörün erişim yetkilerini listeledim. Sonuç olarak hiçbir yetkinin bulunmadığını belirten **-----** şeklinde bir çıktı geldi.

```

root@taylan:~/Desktop# ls -l
toplam 8
dr---xr-- 2 root root 4096 Mar  4 14:51 dosya
d----- 4 root root 4096 Mar  4 08:18 metin

```

Daha sonra "**metin**" isimli klasörün içerisine girerek oradaki dosya ve dizinlerin erişim izinlerini sorguladım. Sonuç olarak **rw-rw-rw-** şeklinde bütün yetkilere sahip dosya ve dizinlerin olduğunu gördüm.

```

root@taylan:~/Desktop# cd metin/
root@taylan:~/Desktop/metin# ls -l
toplam 8
-rwxrwxrwx 1 root root 0 Mar 4 08:18 belgeleri
drwxrwxrwx 2 root root 4096 Mar 4 08:18 çıktılar
drwxrwxrwx 2 root root 4096 Mar 4 15:12 klasörler
-rwxrwxrwx 1 root root 0 Mar 4 08:18 metin
-rwxrwxrwx 1 root root 0 Mar 4 08:18 metin_dosyaları
-rwxrwxrwx 1 root root 0 Mar 4 08:18 metinler

```

Daha sonra bir üst dizine dönerek ekleyeceğim erişim izinlerinin tüm alt dosyalarda dahil olmak üzere, geçerli olması için komutuma ek olarak `-R` parametresini kullandım ve komutumu `chmod -R 422 metin` şeklinde yazdım.

```

root@taylan:~/Desktop/metin# cd ..
root@taylan:~/Desktop# chmod -R 422 metin/
root@taylan:~/Desktop# ls -l
toplam 8
dr---xr-- 2 root root 4096 Mar 4 14:51 dosya
dr---w--w- 4 root root 4096 Mar 4 08:18 metin
root@taylan:~/Desktop# cd metin/
root@taylan:~/Desktop/metin# ls -l
toplam 8
-r---w--w- 1 root root 0 Mar 4 08:18 belgeleri
dr---w--w- 2 root root 4096 Mar 4 08:18 çıktılar
dr---w--w- 2 root root 4096 Mar 4 15:12 klasörler
-r---w--w- 1 root root 0 Mar 4 08:18 metin
-r---w--w- 1 root root 0 Mar 4 08:18 metin_dosyaları
-r---w--w- 1 root root 0 Mar 4 08:18 metinler

```

Sonuç olarak bütün dosya ve dizinlerde ve alt klasörlerde dahil olmak üzere tüm dosyaların vermiş olduğum yetki erişim izinleri **422** ifadesine karşılık gelen; dosya sahibi için okuma(**r**) , dosya sahibi ile aynı gruptaki kullanıcılar için yazma(**w**) ve diğer kullanıcılar için de yazma(**w**) yetkisi şeklinde yetkilendirildiğini görmüş oldum.

Eğer örneklerle ve açıklamalara rağmen yine de anlamadıysanız ister konuyu tekrar okuyup kendiniz de alıştırmalar yapın isterseniz de bu konuyu şimdilik geçin ihtiyacınız olduğunda burada olduğunu bilerek tekrar göz atın. Seçim sizlere kalmış.

## chattr

Hepimizin başına mutlaka gelen ve çok can sıkıcı bir durum var. Bu durum yanlışlıkla silinen dosyalar. Her nasıl ve neden olursa olsun eğer önemli gördüğümüz dosyalar varsa bir şekilde silinmeden onları koruma altına almamız mümkün. Bizlere bu koruma imkanını veren komut **chattr** komutudur. Aslında **chattr** komutu bir tek silinmeye karşı korumuyor, genel olarak dosyanın değiştirilmesine(silme, değiştirme vs.) engel olmak amacıyla kullanılıyor. Yani bu komutumuz bir nevi ilgili dosyayı dokunulamaz kılıyor. Öyle ki herhangi bir yanlış durumda dosyanın kaybolmasına engel olmak adına **root kullanıcısının bile** değişiklik yapmasına imkan tanımıyor. Komutun kullanım alanına bir örnek daha vererek daha iyi anlamış olalım. Örneğin sistemde bir konfigürasyon dosyasını düzenlediniz ve sistemi yeniden başlattınız, fakat bir bakıyorsunuz ki düzenlediğiniz (değiştirdiğiniz) ayarlar kaybolmuş ve dosya eski haline dönmüş. İşte bu gibi durumlarda sistemin bile ilgili dosyaya müdahale etmesini engellemek için oldukça kullanışlı olan **chattr** komutunu kullanabiliyoruz.

Komutun kullanımı `chattr +i dosya_adı` şeklindedir. Hemen bir örnek yapalım.

```
root@taylan:~/Desktop/yeni_dizin# lsattr
-----e---- ./yeni
-----e---- ./metinler
-----e---- ./test.txt
-----e---- ./filazof
-----e---- ./metin_belge
root@taylan:~/Desktop/yeni_dizin# chattr +i test.txt
root@taylan:~/Desktop/yeni_dizin# lsattr
-----e---- ./yeni
-----e---- ./metinler
--i-----e---- ./test.txt
-----e---- ./filazof
-----e---- ./metin_belge
root@taylan:~/Desktop/yeni_dizin#
root@taylan:~/Desktop/yeni_dizin# rm test.txt
rm: cannot remove 'test.txt': Operation not permitted
root@taylan:~/Desktop/yeni_dizin#
```

Bu tür dosyaları listelemek için `lsattr` komutu kullanılıyor. Bizde dosyaların durumunu daha sonradan karşılaştırabilmek adına ilk olarak konsola `lsattr` komutunu verdik.

`test.txt` isimli dosyamızı `chattr +i test.txt` komutu ile koruma altına aldık.

Daha sonra bunu teyit etmek için `lsattr` komutunu kullandık.

Komutumuzun çıktısında görüldüğü gibi dosyamızın sol tarafında izinler kısmında `-i` şeklinde bir ifade var. İşte bu ifade dosyamızın artık düzenlenemeyecek olduğunun işaretidir. Dosyayı silmeye çalışarak bu durumu teyit ettik.

Eğer bu işlemi geri almak ve dosyamızı üzerinde değişiklikler yapılabilir hale getirmek istersek `chattr -i test.txt` komutumuzu kullanmamız yeterli olacaktır. Örneği aşağıda inceleyebilirsiniz.



```
root@taylan:~/Desktop/yeni_dizin# lsattr
-----e---- ./yeni
-----e---- ./metinler
--i-----e---- ./test.txt
-----e---- ./filazof
-----e---- ./metin_belge
root@taylan:~/Desktop/yeni_dizin# chatter -i test.txt
root@taylan:~/Desktop/yeni_dizin# lsattr
-----e---- ./yeni
-----e---- ./metinler
-----e---- ./test.txt
-----e---- ./filazof
-----e---- ./metin_belge
root@taylan:~/Desktop/yeni_dizin# rm test.txt
root@taylan:~/Desktop/yeni_dizin# lsattr
-----e---- ./yeni
-----e---- ./metinler
-----e---- ./filazof
-----e---- ./metin_belge
root@taylan:~/Desktop/yeni_dizin#
```

Çıktıda da görüldüğü gibi hedef dosyamızın solundaki **-i** işareti yok olmuş. Dolayısıyla **test.txt** isimli dosyamızın artık eski halinde dönerek, düzenlenebilir forma girmiş olduğunu gördük. Ve dosyamızı silerek bu durumu teyit ettik.

## Alıştırmalar Hakkında

Yalnızca okumak yetmez, öğrendiğiniz bilgilerin kalıcı olabilmesi için bolca alıştırmaya yapmalısınız. Doküman içerisindeki bilgileri pekiştirmek için aşağıdaki alıştırmalar ile başlayabilirsiniz. Elbette burada yer alan alıştırmalar faaliyetleri dışında, konuyu öğrendiğinizi hissetmeye kadar kendiniz de bolca pratik yapmayı da ihmal etmeyin lütfen. Aksi halde öğrendiğiniz bilgiler kısa sürede unutulup gidecektir.

"**resimler**" isimli bir klasör oluşturun ve bu klasörün yetkilerini; **dosya sahibi için okuma, yazma ve çalıştırma, dosya sahibi ile aynı gruptaki kullanıcılar için yalnızca çalıştırma ve diğer kullanıcılar için de yalnızca okuma** yetkisini verin.

**resimler** klasörüne **tüm kullanıcılar için yalnızca okuma** izni verin.

**resimler** klasörünün yetkilerini tek seferde **tüm kullanıcılara için yalnızca çalıştırma** yetkisi olarak değiştirin.(Ekleme çıkarma yaparak değil tek seferde istenilen yetkiyi verecek şekilde yapın.)

**resimler** klasörünün **sahibinin** *çalıştırma* yetkisini kaldırın.

**rakamları kullanarak** **resimler** klasörü için **sadece diğer kullanıcılara** *okuma* ve *yazma* yetkisi verin.

**rakamları kullanarak** **resimler** klasörü için **sahibine ve sahibi ile aynı gruptaki kullanıcılara** *yazma* ve *çalıştırma* yetkisi verin.

**rakamları kullanarak** **resimler** klasörü için **tüm kullanıcıların** yetkilerini kaldırın.

**Farklı izinlere sahip 2 klasörün** yetkilerini *birbiri ile aynı* hale getirin.

**Klasörün içerisinde yer alan tüm dosya ve klasörler de dahil olacak şekilde** tek seferde tüm içeriklerin yetkilerini değiştirin. **(Yani yaptığınız değişiklik klasörün alt dizinlerinde yer alan tüm dosya ve klasörleri etkileyecek şekilde olsun. )**

Herhangi bir dosyayı silinmeye karşı korumak için ilgili komutu kullanarak **koruma altına alın**. Ve test etmek için dosyanın izinlerini değiştirip dosyayı silmeye çalışın.

Koruma altına aldığınız **dosyanın korunup korunmadığını** ilgili komutu kullanarak **teyit edin**.

Silinmeye karşı koruma altına aldığınız **dosyanın korumasını** ilgili komutu kullanarak **kaldırın**. Ve test etmek için dosyanın izinlerini değiştirip dosyayı silmeye çalışın.

## Geri Bildirimde Bulunun

Sizlere daha verimli bir kaynak sunabilmemiz için, uygulamada veya dokümantasyonlarda yer alan tüm hata ve eksiklerimizi bize bildirebilirsiniz.

Geri Bildirimde Bulunun