

## 12.Doküman

## Kullanıcı İşlemleri

Konu Etiketleri

grup yönetimi , id , groupadd , kullanıcı yönetimi , adduser , useradd, kullanıcı silmek , deluser , gpasswd , aktif pasif hesap , usermod , kimlik değişimi , su , sudo , exit

## Kullanıcı İşlemleri

Linux'ta birçok kullanıcı hesabının olabildiğini ve bunlar içinde en yetkilinin root kullanıcısı olduğunu önceki derslerimizde tecrübe etmiştik. Ancak sistemde root kullanıcısı olmadan da root kullanıcısının yetkilerine sahip olmamız mümkün. Bu imkanı bize **Linux Grup Yönetimi** veriyor.

### Grup Yönetimi

Aynı grupta yer alan kullanıcılar bizim tanımlamamızla aynı haklara sahip olabiliyorlar.

Biraz daha ayrıntılı vermek gerekirse **Linux** ve **UNIX** sistemlerindeki kullanıcılar bir veya daha fazla gruba aşağıdaki nedenlerle atanır:

- Dosyaları veya diğer kaynakları Grup yönetimi sayesinde ilgili kullanıcılarla paylaşarak, sistemde erişim sınırlamalarıyla güvenlik sağlaması.
- Kullanıcı yönetim ve denetiminde kolaylık sağlıyor olması.
- Grup üyeliği, bu gruba izin verilen dosyalara, dizinlere veya cihazlara özel erişim sağlar. Bu madde baştaki madde ile benzer amaca hizmet ederek tamamen kontrolü elde tutmayı sağlamak için kullanılır.

Yani genel olarak **Grup sistemini** bir çeşit **kontrol mekanizması** gibi düşünebilirsiniz.

Bu kısa bilgilendirmeden sonra gelin grup yönetimi ile ilgili işlemlere geçelim.

Bir grup oluşturulduğunda bu grubun bilgisi **/etc** dizini içerisinde yer alan **group** isimli dosyada tutuluyor. Yani mevcut grupları görüntülemek istersek **/etc** dizini içerisinde yer alan **group** dosyasına bakmamız gerekir. Bunun için **less /etc/group** , **more /etc/group** , **cat /etc/group** komutlarından herhangi birini kullanabiliriz. Çıktıları aşağıdaki şekilde olacaktır.(Çıktıyı uzun olması nedeni ile aşağıdaki çıktının görselini keserek verdim.)

```
root@taylan:~# more /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
- -More- - (11%)
```

Hemen bu çıktıda yer alan kısımlara bir açıklık getirelim. Bunun için bir grubu bölüm bölüm numaralandırarak açıklayalım. Ben bu durum için örnek bir grup belirtiyorum.

```
cdrom:x:24:taylan,dev_usr,neo
```

1      2   3                      4

Belirtilen numaralı kısımların açıklamaları:

**1. Grup\_ismi :** Gruba verilen isimdir.

**2. Parola :** Parola alanını belirtiyor. Genelde parola kullanılmaz ancak kimi durumlarda kullanıldığı oluyor, bizim çıktımızda da **x** ile belirtilen alan parola kısmının boş olduğunu belirtiyor. Bu parola belirleme işlemi çok sık kullanılsa da, ayrıcalıklı gruplarda uygulamak için yararlıdır.

**3. Grup Kimliği (GID) :** Atanan grup kimliğini(grup numarasını) belirtiyor.

**4. Grup Listesi :** Grubun üyesi olan kullanıcıların kullanıcı adlarının bir listesidir. Kullanıcı adları, virgülle ayrılmış şekilde belirtiliyor.

Bu alanların neleri ifade ettiğini öğrendiğimize göre, gelin hangi kullanıcının hangi gruba dahil olduğunu nasıl öğrenebiliriz onu görelim.

## Kullanıcı Gruplarını Sorgulamak

Kullanıcıların ait olduğu grupları görmek istersek komut satırına `id kullanıcı_adı` şeklinde komut vermemiz yeterli olacaktır. Ben "**burak**" kullanıcı hesabı için sorgulama yapmak üzere konsola `id burak` şeklinde komutumu veriyorum.

```
root@taylan:~# id burak
uid=1000(burak) gid=1000(burak) gruplar=1000(burak),1001(ad)
```

Gördüğünüz gibi "**burak**" kullanıcı hesabı için burada; **uid**(user id /kullanıcı numarası), **gid**(group id/grup numarası) ve dahil olduğu **gruplar** listelenmiş oldu.

Burada yeri gelmişken çıktıda da görülen **uid**(user id/kullanıcı numarası) ve **gid**(group id/grup numarası) kavramlarına değinelim. Bu numara aralıkları kullanıcı hesabına göre değişiklik gösteriyor. Yani kullanıcı çeşidine göre numaraları üç temel gruba ayırabiliriz. O da şu şekilde ;

- **root kullanıcısı** : UID=0, GID=0
- **sistem kullanıcısı** : UID=1 - 499, GID=1 - 499
- **normal kullanıcı** : UID=500 < X, GID=500 < X (Buradaki X ifadeleri 500'den büyük tüm sayıları temsil etmektedir.)

Ayrıca `id` komutunun birçok parametresi vardır bazıları birazdan göreceğimiz grup oluşturma kısmındaki parametrelerden oluşuyor. Detaylı bilgi için man sayfasına bakabilirsiniz. Ben yine de örnek olması açısından birkaç parametresini gösteriyorum ve daha sonra grup oluşturma ile konumuza devam ediyoruz.

**g :** `id -g kullanıcı_adı` belirtilen kullanıcının **grup numarasını(gid)** verecektir.

**u :** `id -u kullanıcı_adı` belirtilen kullanıcının **kullanıcı numarasını(uid)** verecektir.

**G :** `id -G kullanıcı_adı` belirtilen kullanıcının **dahil olduğu tüm grupları(groups)** verecektir.

```
root@taylan:~# id burak
uid=1000(burak) gid=1000(burak) gruplar=1000(burak),1001(ad)
root@taylan:~# id -g burak
1000
root@taylan:~# id -u burak
1000
root@taylan:~# id -G burak
1000 1001
```

Yukarıdaki parametreler haricindekiler için man kılavuz sayfasına bakın lütfen.

## Sıra geldi yeni grup oluşturmaya..

Yeni bir grup oluşturmak istersek `groupadd yeni_grup_adı` şeklinde komutumuzu kullanırız.

Ben örnek olması açısından "**yeni**" isimli bir grup oluşturmak için komut satırına `groupadd yeni` şeklinde komutumu veriyorum.

```
root@taylan:~# groupadd yeni
root@taylan:~#
```

Ve oluşturduğumuz grubu sorgulamak için grup bilgilerinin tutulduğu dosyaya bakmak üzere `cat /etc/group | grep grup_adı` komutunu ya da `tail -n 1 /etc/group` kullanabiliriz.

```
root@taylan:~# tail -n 1 /etc/group
yeni:x:1005:
root@taylan:~# cat /etc/group | grep yeni
yeni:x:1005:
```

Ayrıca grup oluşturulurken kullanılabilecek bazı parametreler var. **Bunlar;**

**g** : Grup id belirleme. İstediğiniz numarayı başka bir gruba ait numara ile aynı olmayacak şekilde verebilirsiniz.

```
root@taylan:~# groupadd -g 1234 GNU
root@taylan:~# tail -n 1 /etc/group
GNU:x:1234:
```

Eğer aynı grup id ile başka bir grup eklemek istersek konsol bize "**bu id ye sahip başka bir grubun halihazırda bulunduğu**" uyarısını verecektir. Dolayısı ile grup ekleme işlemi başarısız olacaktır.

```
root@taylan:~# groupadd -g 1234 Linus
groupadd: GID '1234' already exists
```

**f** : işlemi hatalar olsa bile zorlayarak tamamlar. Genelde bu kullanım sorunlar çıkardığı için pek tavsiye edilmez.

Parametreler bunlar ile sınırlı değil ancak sizler `man groupadd` ve `groupadd --help` komutları yardımı ile diğer parametreleri de keşfedebilirsiniz.

Ayrıca oluşturduğunuz grupları silmek isterseniz konsola `groupdel grup_adı` şeklinde komut yazmanız yeterli olacaktır.

```
root@taylan:~# cat /etc/group | grep GNU
GNU:x:1234:
root@taylan:~# groupdel GNU
root@taylan:~# cat /etc/group | grep GNU
root@taylan:~#
```

## Kullanıcı Yönetimi

Sistemimize yeni bir kullanıcı eklemek için kullanabileceğimiz iki farklı komut var. Bunlar `adduser` ve `useradd` komutlarıdır.

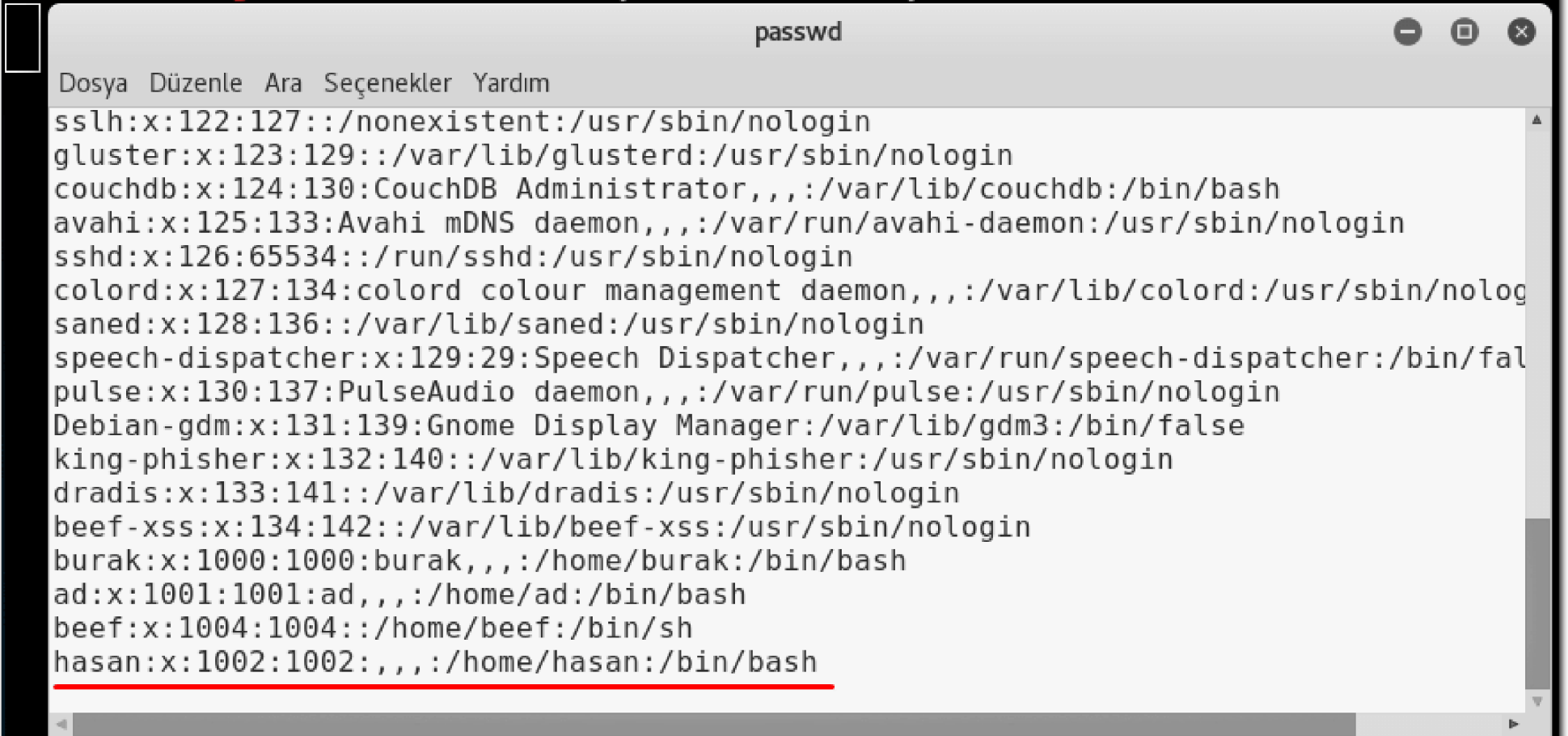
### adduser

Yeni kullanıcı eklemek için komutumuzu `adduser kullanıcı_adı` şeklinde veriyoruz. Böylelikle oluşturduğumuz kullanıcıya ait ev dizini `home/kullanıcı_adı` şeklinde otomatik olarak oluşmuş oluyor. Ben örnek olaması açısından "**hasan**" isimli bir kullanıcı hesabı eklemek üzere konsola `adduser hasan` komutunu veriyorum.

```
root@taylan:~# adduser hasan
Adding user `hasan' ...
Adding new group `hasan' (1002) ...
Adding new user `hasan' (1002) with group `hasan' ...
Creating home directory `/home/hasan' ...
Copying files from `/etc/skel' ...
Yeni parolayı girin:
Yeni parolayı tekrar girin:
passwd: şifre başarıyla güncellendi
hasan için kullanıcı bilgileri değiştiriliyor
Yeni değeri girin, veya varsayılan değer için ENTER'a basın
    Tam İsim []:
    Oda Numarası []:
    İş Telefonu []:
    Ev Telefonu []:
    Diğer []:
Is the information correct? [Y/n] y
root@taylan:~#
```

Çıktıları inceleyecek olursak ev dizini otomatik olarak oluşturulmuş ve "**hasan**" isimli kullanıcı "**hasan**" grubuna dahil edilmiş. Yeni eklemiş olduğumuz "**hasan**" isimli kullanıcı hesabını teyit etmek için; Sistemde kullanıcı hesapları ile ilgili bilgilerin tutulduğu `etc` dizini içerisindeki `passwd` dosyaya bakalım.

```
root@taylan:~# leafpad /etc/passwd
```



```
passwd
Dosya Düzenle Ara Seçenekler Yardım
sshd:x:122:127:./nonexistent:/usr/sbin/nologin
gluster:x:123:129:./var/lib/glusterd:/usr/sbin/nologin
couchdb:x:124:130:CouchDB Administrator,,,:/var/lib/couchdb:/bin/bash
avahi:x:125:133:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/usr/sbin/nologin
sshd:x:126:65534:./run/sshd:/usr/sbin/nologin
colord:x:127:134:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
saned:x:128:136:./var/lib/saned:/usr/sbin/nologin
speech-dispatcher:x:129:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
pulse:x:130:137:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
Debian-gdm:x:131:139:Gnome Display Manager:/var/lib/gdm3:/bin/false
king-phisher:x:132:140:./var/lib/king-phisher:/usr/sbin/nologin
dradis:x:133:141:./var/lib/dradis:/usr/sbin/nologin
beef-xss:x:134:142:./var/lib/beef-xss:/usr/sbin/nologin
burak:x:1000:1000:burak,,,:/home/burak:/bin/bash
ad:x:1001:1001:ad,,,:/home/ad:/bin/bash
beef:x:1004:1004:./home/beef:/bin/sh
hasan:x:1002:1002:,,,:/home/hasan:/bin/bash
```

Dosyanın en alt satırına baktığımızda yeni eklemiş olduğumuz "**hasan**" isimli kullanıcı hesabının eklenmiş olduğunu görüyoruz. Son olarak da `home` dizinine "**hasan**" kullanıcı hesabı için dosya oluşturulmuş mu buna bakalım.



```
root@taylan:~# cd /home && ls
ad ahmet ayşe burak hasan lost+found
root@taylan:~/home#
```

Çıktıda görüldüğü gibi **home** dizininde oluşturmuş olduğum "**hasan**" isminde kullanıcı dosyası bulunuyor.

## useradd

Kullanıcı oluşturmak için bahsettiğimiz 2. komut olan **useradd** komutu kullanımı; **useradd -m kullanıcı\_adı** şeklindedir. Burada komutumuz ile birlikte kullanmış olduğumuz **m** parametresi ile, oluşturduğumuz kullanıcıya ait ev dizininin otomatik oluşmasını sağladık. Birde son olarak kullanıcıya ait bir parola belirlemek üzere **passwd kullanıcı\_adı** şeklinde komutumuzu vererek parolamızı oluşturuyoruz.

Parola oluşturma işleminde kullandığımız **passwd kullanıcı\_adı** şeklindeki komut bütün hesaplar için şifre güncelleme işleminde de kullanılıyor. Ve bir bilgi daha; bu parola bilgileri **/etc** dizini altındaki **shadow** isimli dosyada şifreli şekilde tutuluyor.

```
root@taylan:~# leafpad /etc/shadow

shadow

Dosya Düzenle Ara Seçenekler Yardım
root:$6$HCJm.Tl3$xe4HCHRQmzxkHFJt3xzhonq86o/9ixwjJCG70KHkJcsQ7TMpmtgkEz150je34hK7lloN5UKuEE/bNwS6a8XXq.:17562:0:99999:7:::
daemon*:17479:0:99999:7:::
bin*:17479:0:99999:7:::
sys*:17479:0:99999:7:::
sync*:17479:0:99999:7:::
games*:17479:0:99999:7:::
man*:17479:0:99999:7:::
lp*:17479:0:99999:7:::
mail*:17479:0:99999:7:::
news*:17479:0:99999:7:::
uucp*:17479:0:99999:7:::
proxy*:17479:0:99999:7:::
www-data*:17479:0:99999:7:::
backup*:17479:0:99999:7:::
list*:17479:0:99999:7:::
irc*:17479:0:99999:7:::
gnats*:17479:0:99999:7:::
nobody*:17479:0:99999:7:::
systemd-timesync*:17479:0:99999:7:::
```

Ben örnek olması açısından "**ali**" isimli bir kullanıcı hesabı oluşturmak üzere komutumu **useradd -m ali** şeklinde verdikten sonra "**ali**" isimli kullanıcı hesabı için şifre belirlemek üzere **passwd ali** komutunu kullandım.

```
root@taylan:~# useradd -m ali
root@taylan:~# passwd ali
Yeni parolayı girin:
Yeni parolayı tekrar girin:
passwd: şifre başarıyla güncellendi
root@taylan:~#
```

Görüldüğü gibi "**ali**" isimli kullanıcı hesabımız başarılı şekilde oluşturulmuş oldu. Şimdi **home** dizinini ve **passwd** dosyasına bakarak bu durumu teyit edelim.

```
root@taylan:~# cd /home && ls
ad ahmet ali ayşe burak hasan lost+found
root@taylan:/home#
root@taylan:/home# tail -n 1 /etc/passwd
ali:x:1005:1006::/home/ali:/bin/sh
root@taylan:/home#
```

Çıktı sonuçlarında bir problem olmadığına göre konumuza, kullanıcı silme işlemi ile devam edebiliriz.

## Kullanıcı Silmek

Oluşturulan kullanıcı hesabını ev dizini ile birlikte silmek için `deluser --remove-home kullanıcı_adi` komutunu kullanmamız yeterli olacaktır.

```
root@taylan:/home# deluser --remove-home ali
Looking for files to backup/remove ...
Removing files ...
Removing user `ali' ...
Warning: group `ali' has no more members.
Done.
root@taylan:/home# deluser --remove-home hasan
Looking for files to backup/remove ...
Removing files ...
Removing user `hasan' ...
Warning: group `hasan' has no more members.
Done.
```

Kullanıcı silme işlemlerini kontrol etmek adına `home` dizinini kontrol edelim.

```
root@taylan:/home# ls
ad ahmet ayşe burak lost+found
root@taylan:/home#
```

Çıktıda "**hasan**" ya da "**ali**" bulunmadığına göre kullanıcı silme işlemimiz başarılı şekilde tamamlanmış oldu.

## Gruplara Kullanıcı Ekleme-Çıkarma İşlemi

Yeni grup ve yeni kullanıcı hesabı nasıl oluşturulur bunları gördük. Şimdi sıra geldi gruba yeni kullanıcı ekleme ve gruptan kullanıcı çıkarma işlemlerine.

**Gruba Kullanıcı Ekleme** Var olan bir gruba yeni bir kullanıcı eklemek için `gpasswd` komutunun `a` parametresini kullanarak, komutumuzu `gpasswd -a kullanıcı_adi ekleneceği_grup_adi` şeklinde kullanmamız yeterlidir.

```
root@taylan:~# tail -n 5 /etc/group
beef-xss:x:142:
burak:x:1000:beef
ad:x:1001:burak
beef:x:1004:
yeni:x:1005:
root@taylan:~# gpasswd -a burak yeni
burak kullanıcısı yeni grubuna ekleniyor
root@taylan:~# tail -n 5 /etc/group
beef-xss:x:142:
burak:x:1000:beef
ad:x:1001:burak
beef:x:1004:
yeni:x:1005:burak
root@taylan:~#
```

### Gruptan Kullanıcı Çıkarmak

Eğer gruba dahil olan bir kullanıcıyı gruptan çıkarmak istersek de `gpasswd` komutunun `d` parametresini kullanarak, komutu `gpasswd -d kullanıcı_adı çıkarılacağı_grup_adı` şeklinde kullanmamız gerekmektedir.



```

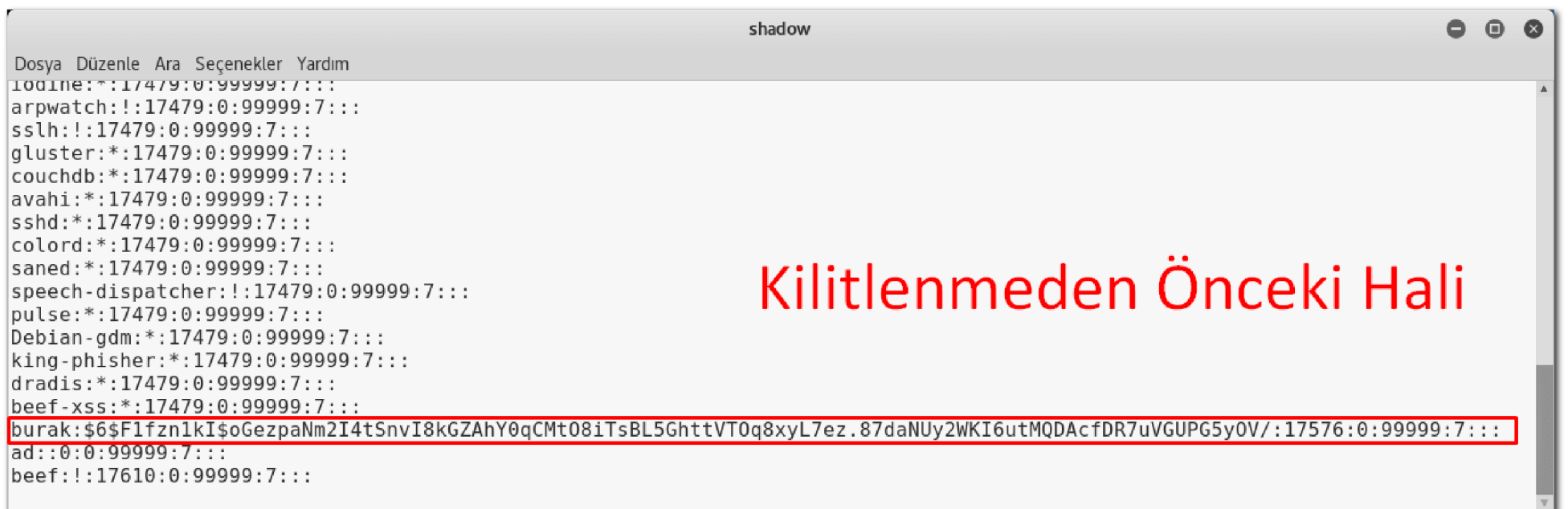
root@taylan:~# tail -n 5 /etc/group
beef-xss:x:142:
burak:x:1000:beef
ad:x:1001:burak
beef:x:1004:
yeni:x:1005:burak
root@taylan:~#
root@taylan:~# gpasswd -d burak yeni
burak kullanıcısı yeni grubundan çıkarılıyor
root@taylan:~# tail -n 5 /etc/group
beef-xss:x:142:
burak:x:1000:beef
ad:x:1001:burak
beef:x:1004:
yeni:x:1005:
root@taylan:~#

```

## Aktif-Pasif Hesap Ayarlama

Varolan bir kullanıcı hesabını kilitlemek için `usermod -L kullanıcı_adı` komutunu kullanırız. Buradaki kilitlemekten kasıt kullanıcının sisteme giriş yapmasını engellemek için parolasının kilitlenmesidir. Parolayı devre dışı bırakmak için, sistem otomatik olarak şifreli parolanın önüne '!' işareti koyar ve kullanıcı hesabına giriş yapamaz.

Bu durumu denemek için öncelikle kullanıcı hesaplarının parolalarının şifreli bir şekilde tutulduğu dosya olan `shadow` dosyasına bakalım.



Kilitlenmeden Önceki Hali

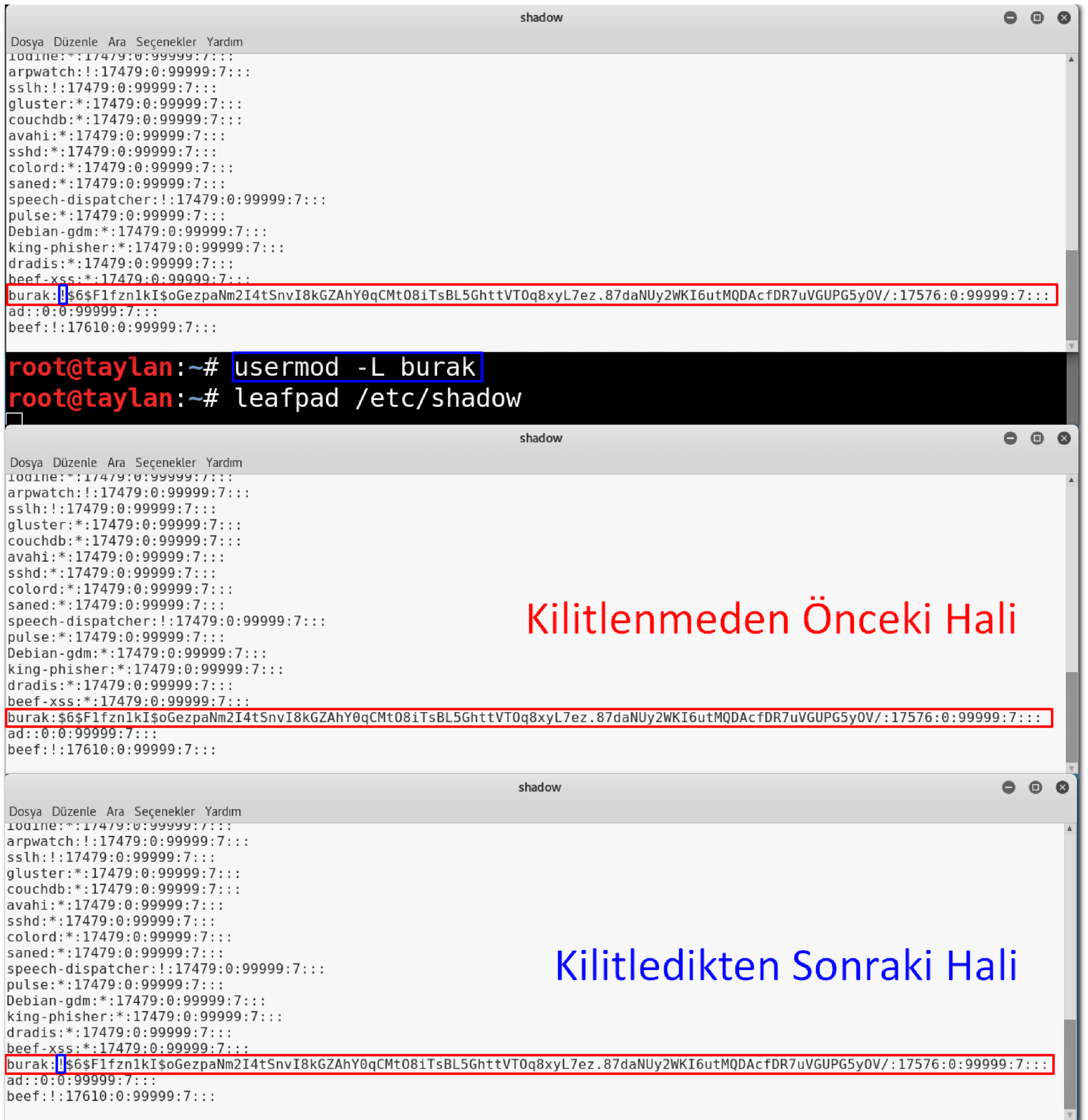
```

Dosya Düzenle Ara Seçenekler Yardım
shadow
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
arpwatch:x:6:6:arpwatch:/usr/sbin:/usr/sbin/nologin
sshd:x:42:42:ssh:/usr/sbin:/usr/sbin/nologin
gluster:x:49999:49999:gluster:/usr/sbin:/usr/sbin/nologin
couchdb:x:59999:59999:couchdb:/usr/sbin:/usr/sbin/nologin
avahi:x:58:58:avahi:/usr/sbin:/usr/sbin/nologin
sshd:x:42:42:ssh:/usr/sbin:/usr/sbin/nologin
colord:x:35:35:colord:/usr/sbin:/usr/sbin/nologin
saned:x:42:42:saned:/usr/sbin:/usr/sbin/nologin
speech-dispatcher:x:42:42:speech-dispatcher:/usr/sbin:/usr/sbin/nologin
pulse:x:42:42:pulse:/usr/sbin:/usr/sbin/nologin
Debian-gdm:x:42:42:gdm:/usr/sbin:/usr/sbin/nologin
king-phisher:x:42:42:king-phisher:/usr/sbin:/usr/sbin/nologin
dradis:x:42:42:dradis:/usr/sbin:/usr/sbin/nologin
beef-xss:x:142:142:beef-xss:/usr/sbin:/usr/sbin/nologin
burak:$6$F1fzn1kI$0GezpaNm2I4tSnvI8kGZAHy0qCMt08iTsBL5GhttVT0q8xyL7ez.87daNUy2WKI6utMQDAcfDR7uVGUPG5y0V/:17576:0:99999:7:::
ad::0:0:99999:7:::
beef::17610:0:99999:7:::

```

"burak" isimli kullanıcının parolasını yukarıdaki şekildedir. Daha sonra karşılaştırma yaparken tekrar ele alacağız.

Konsol ekranına `usermod -L burak` yazarsam "burak" kullanıcısının hesabını kilitlemiş olurum. Bu kilitleme işlemini; kullanıcının hesaba girişte kullandığı parolanın önüne sistem tarafından otomatik olarak getirilen `!` (ünlem işareti) sayesinde gerçekleşmektedir. Bu durumu teyit etmek için tekrar `shadow` dosyasına bakalım ve önceki hali ile sonraki halini kıyaslayarak görelim.



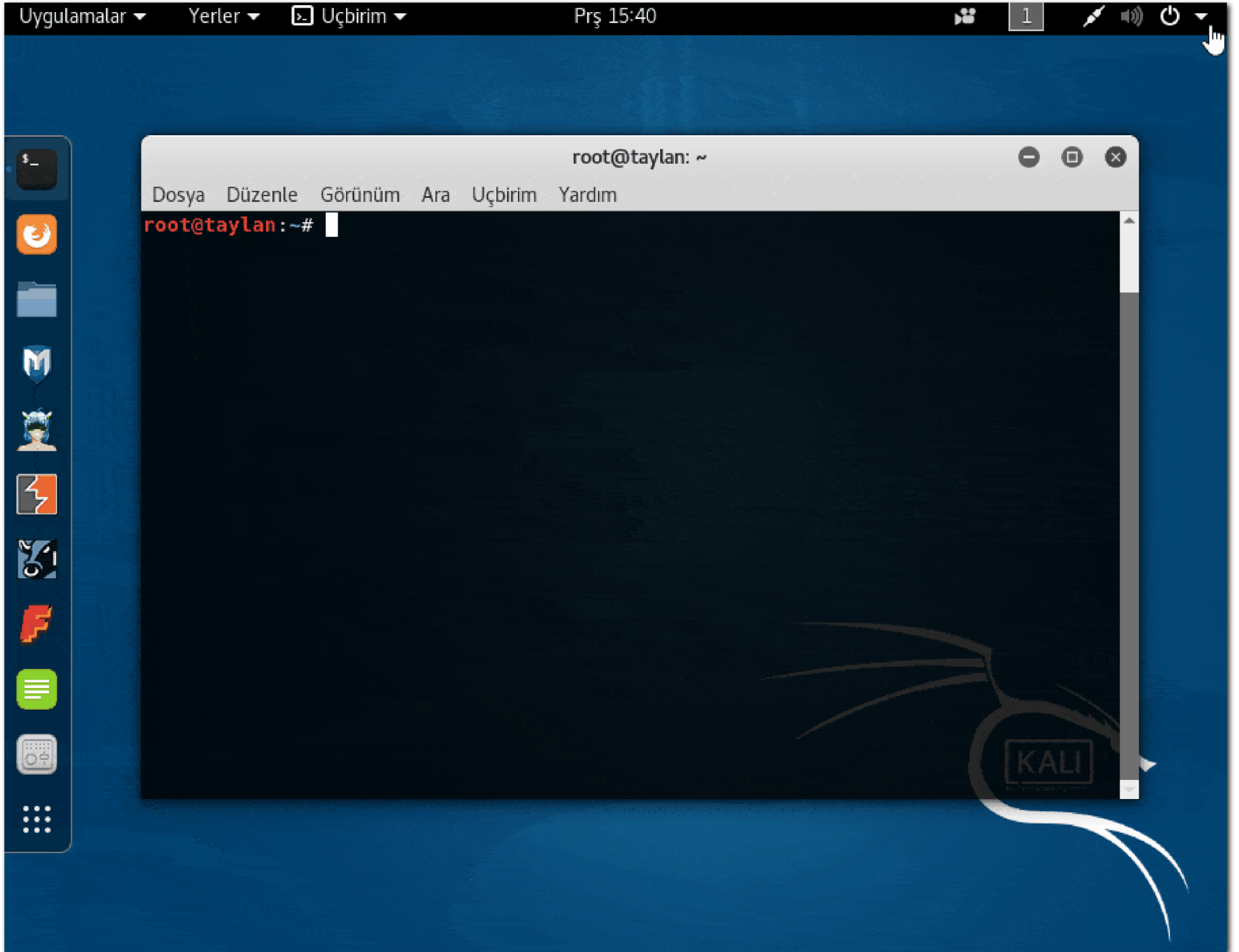
Gördüğünüz gibi hesabı kilitleme işleminde; hesaba girişte kullanılan parolanın önüne otomatik olarak gelen **!** (ünlem işareti) hesaba girmeyi engelleyerek hesabı kilitlemiş oluyor.

Pasif durumdaki hesabı aktif hale getirmek için ise **usermod -U kullanıcı\_adı** komutu kullanılır. Aynı şekilde hesabın tekrar aktif hale getirilmesi de devre dışı bırakılmış olan kullanıcı parolasının önündeki '!' işaretinin kaldırılmasıyla gerçekleşir.

```
shadow
Dosya Düzenle Ara Seçenekler Yardım
iodine:!:17479:0:99999:7:::
arpwatch:!:17479:0:99999:7:::
sslh:!:17479:0:99999:7:::
gluster:!:17479:0:99999:7:::
couchdb:!:17479:0:99999:7:::
avahi:!:17479:0:99999:7:::
sshd:!:17479:0:99999:7:::
colord:!:17479:0:99999:7:::
sane:!:17479:0:99999:7:::
speech-dispatcher:!:17479:0:99999:7:::
pulse:!:17479:0:99999:7:::
Debian-gdm:!:17479:0:99999:7:::
king-phisher:!:17479:0:99999:7:::
dradis:!:17479:0:99999:7:::
beef-xss:!:17479:0:99999:7:::
burak:$6$F1fzn1kI$0GezpaNm2I4tSnvI8kGZAHy0qCMt08iTsBL5GhttVT0q8xyL7ez.87daNUy2WKI6utMQDAcfDR7uVGUPG5y0V/:17576:0:99999:7:::
ad:!:0:0:99999:7:::
beef:!:17610:0:99999:7:::

root@taylan:~# usermod -U burak
root@taylan:~# leafpad /etc/shadow
```

Eğer anlamadığınız bir nokta varsa bu durumu GIF ile anlattım, dikkatlice takip ederseniz gayet net anlaşılacaktır.



## Kimlik Değişimi

Örneğin ben "**burak**" kullanıcısıyken, yapmak istediğim işlem ancak "**root**" kullanıcısının yetki alanındaysa ve ben "**root**" hesabının parolasını biliyorsa, "**root**" hesabının kimliğine bürünerek o işlemi gerçekleştirebilirim.

Geçiş işlemleri için **su** komutunu kullanıyoruz. Komutun kullanımı ile ilgili iki farklı durum var, bunlar:

- **su kullanıcı\_adi** : diğer kullanıcı kimliğine geçiş yapar.
- **su - kullanıcı\_adi** : diğer kullanıcı kimliğine geçiş yapar ve direk olarak geçiş yapılan kullanıcının kabuğunda çalışmaya başlar.



Bu durum en iyi örnekler ile açıklanabilir. İlk önce **"root"** kullanıcısıyken **"burak"** isimli kullanıcı hesabına giriş yapacağım daha sonra **"burak"** isimli kullanıcıdan **"root"** hesabına giriş işlemini gerçekleştireceğim. Adımları sırasıyla takip edin lütfen.

Hemen mevcut kullanıcı oturumunu daha önce öğrenmiş olduğumuz **whoami** komutu ile sorgulayalım.

```
root@taylan:~# whoami
root
root@taylan:~#
```

**"root"** kullanıcısıyken **"burak"** hesabına geçiş yapmak için **su burak** komutunu kullanıyoruz. Ve **whoami** komutu ile geçiş durumunu kontrol ediyoruz.

```
root@taylan:~# su burak
burak@taylan:/root$ whoami
burak
burak@taylan:/root$
```

Geçiş yaptığımız **"burak"** hesabından **"root"** hesabına dönmek için **exit** komutunu kullanmamız yeterli.

```
burak@taylan:/root$ exit
exit
root@taylan:~# whoami
root
root@taylan:~#
```

Şimdi **su - burak** komutu kullanarak geçiş yapacağımız kullanıcı kabuğunda çalışalım.

```
root@taylan:~# su - burak
burak@taylan:~$ whoami
burak
burak@taylan:~$
```

Görsellere dikkatlice bakacak olursanız; **su burak** komutunu kullanarak geçiş yaptığımızda **burak@taylan:/root\$** olan konsol ismi **su - burak** komutunu kullandığımızda direk olarak geçiş yaptığımız hesabın kabuğunda çalışmaya başlayarak **burak@taylan:~\$** şeklinde oldu. Yani **su - burak** komutu bize tıpkı o hesap oturumunu açmışız gibi tepki verdi. Aşağıda çıktıları tekrar bakarsanız aradaki farkı daha net anlayabilirsiniz.

```
root@taylan:~# su burak
burak@taylan:/root$ exit
exit
root@taylan:~# su - burak
burak@taylan:~$ exit
çıkış
root@taylan:~#
```

Fark etmiş olacaksınız ki başta geçiş yapabilmemiz için geçeceğimiz hesabın parolasını bilmemiz gerektiğini söylemiştim. Ancak "**burak**" hesabına yaptığımız geçişlerde parola sorulmadı. Bunun sebebi "**burak**" hesabının normal kullanıcı olmasıdır. Şimdi bu durumu birde "**burak**" hesabındayken "**root**" kullanıcısı için iki farklı kullanımda da deneyelim.

```
burak@taylan:~$ su root
Parola:
root@taylan:/home/burak# whoami
root
root@taylan:/home/burak#
```

```
burak@taylan:~$ su - root
Parola:
root@taylan:~# whoami
root
root@taylan:~#
```

Gördüğünüz gibi yönetici hesabına erişmek için bizlerden parola bilgisi istendi. Bu durum da başta açıkladığımız yetkili hesaplara geçişlerde şifre sorma olayını açıklıyor.

### Alıştırmalar Hakkında

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



**Linus** isminde **yeni bir kullanıcı hesabı oluşturun** ve **home** dizinine bakarak teyit edin.

**GNU** isminde **1515** grup numarasına(gid) sahip **yeni bir grup oluşturun** ve grubun oluşup oluşmadığını ilgili dosyaya bakarak teyit edin.

Oluşturmuş olduğunuz **GNU** isimli guruba yeni oluşturmuş olduğunuz **Linus** kullanıcı **hesabını ekleyin**.

**Linus** isimli **kullanıcının dahil olduğu grubu konsoldan** ilgili komut ile **sorgulayın**.(uid(user id/kullanıcı numarası), gid(group id/grup numarası) ve dahil olduğu grupları konsoldan listeleyin.)

**GNU** isimli gruba eklemiş olduğunuz **Linus** kullanıcı **hesabını gruptan çıkarın** ve bu durumu(gruptan çıkarılıp çıkarılmadığını) teyit edin.

**GNU** isimli **gurubu silin ve silme işlemini de teyit edin**.

**Linus** isimli kullanıcı **hesabına giriş yapın**(oturum açın) ve **Linus** kullanıcı hesabındayken **root** kullanıcı hesabına **kimlik değişimi** ile konsoldan **geçiş yapın**. Geçiş yaptıktan sonra **geçiş** ilgili komut(hangi kullanıcı oturumunda çalışmakta olduğumuzu bildiren komut) ile **teyit edin**. Ve daha sonra **root** kullanıcı hesabından ilgili komut ile **çıkış yaparak tekrar Linus kullanıcı hesabına dönün** ve **geçiş** ilgili komutla **teyit edin**.

**Linus** kullanıcı hesabındayken ilgili komut ile konsoldan **root** kullanıcı **hesabının kimliğine geçiş yapıp direk olarak root kullanıcı hesabının kabuğundan çalışmaya başlayın**. Ve bu durumu ilgili komut ile teyit edin.(Burada önemli nokta **geçiş yapılan kullanıcının kabuğunda çalışmaktır**.)

**Linus** kullanıcı hesabındayken **oturumu kapatıp root** kullanıcı **hesabına geçin** ve **root** kullanıcı hesabındayken **Linus** kullanıcı **hesabını kilitleyin**. Ve **hesabın kilitlenip kilitlenmediğini de teyit edin**.

Kilitlemiş olduğunuz **Linus** kullanıcı **hesabının kilidini açın** ve **hesaba giriş yaparak açılıp açılmadığını teyit edin**.

# Geri Bildirimde Bulunun

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

Geri Bildirimde Bulunun