

Linux

UBUNTU 20.04

ÖZGÜR AYDIN

1 - LINUX DOSYA SİSTEMİ VE UBUNTU KURULUMU	8
Linux Dosya Sistemi Yapısı	8
Linux Dosya Tipleri.....	8
Linux İşletim Sistemi Kurulumu	10
Putty ile İşletim Sistemine Erişmek.....	19
2 - TEMEL LINUX KOMUTLARI	21
Terminoloji.....	21
Shell Komut Çeşitleri - Alias, Internal & External Commands, which, type, \$PATH.....	22
Linux'ta Kabuk Başlatma Dosyalarını ve Kullanıcı Profillerini Anlama	22
man Kullanımı	24
info kullanımı.....	26
help kullanımı.....	26
tab kullanımı.....	26
whatis / whereis	27
Temel Komutlar – pwd, cd, ls, history, clear	27
Temel Komutlar - touch, cat, cp, mkdir, rm, rmdir, mv, rename.....	29
Temel Komutlar - file, time, uptime, cal	32
I/O Redirection - stdin, stdout, stderr	32
Operatörler - ‘’, “”, ;, &, &&, , , #, ~	34
Değişkenler	36
Bash Değişkenleri	37
Environment Değişkenleri.....	37
Shell Değişkeni - \$	38
Wildcard * ? []	38
3 - DOSYALAR İLE ÇALIŞMAK.....	39
Shell Ortamı ve Kullanıcı Profili	39
Dosya Editörlerinin Kullanımı	40
Dosya İçerğini Görüntüleme - less, more, cat, head, tail	42
Filtreler - grep, cut, sort, wc, tr, uniq, sed, awk	44
İçerik ve Dosya Arama Komutları - find, locate	46
Hardlink & Soft Link Kullanımı.....	49
4 - İŞLETİM SİSTEMİ BAĞLANTI TIPLERİ VE SERVİS KONTROL	51
İşletim Sistemine Login Olma - TeleTYpewriter (TTY), Pseudo Terminal Slave (PTS)	51
Komutlar - w, who, whoami, last, id, su, su –.....	52
İşletim Sistemini Shutdown ve Reboot Etmek	54
Servisleri Kontrol Etmek - systemctl, enable, disable.....	54
5 - SSH Yapılandırması	55

SSH Servisi Nedir?	55
SSH Servisinin Yapılandırılması ve Güvenli Hale Getirilmesi.....	56
Kullanıcı sınırlaması ve root kullanıcısını kapatmak.....	56
Port Değişikliği.....	57
Belirli bir adrese SSH isteklerine izin vermek	57
Belirli adreslerden SSH erişimine izin vermek	58
Host Bazlı Erişim Engelleme	58
Boş Parolalar ile Erişimi Engelleme	59
Açık Kalan SSH Oturumlarını Belirli Bir Süre Sonra Kapatma.....	59
Erişim Denemesini Sınırlama	59
SSH 2 Versiyonunu Kullanmak.....	59
LoginGraceTime	60
SSH Key Yapılandırması	60
SFTP (Secure File Transfer Protocol).....	62
6 – DOSYA VE DİZİNLERDE İZİN YÖNETİMİ	63
Dosya Sahipliklerini ve İzinleri Anlama	63
Erişim Modları	64
Kullanıcı ve Grup Sahipliğini Değiştirme - chown ve chgrp.....	65
Dosya ve Dizin İzin İşlemleri - chmod	67
SUID (Set User ID), SGID (Set Group ID) ve Sticky Bit.....	68
ACL Yönetimi	71
7 - KULLANICI VE GRUP YÖNETİMİ.....	75
Kullanıcı Tipleri	75
Kullanıcı Yönetimi - useradd, userdel	77
Parola Yönetimi – passwd, chage	78
Parola Yönetimi için PAM Modülü Kullanımı.....	80
Grup Yönetimi – groupadd, groupdel	81
Usermod	82
Sudo – Sudoers File	83
8 - İŞLEM(PROCESS) YÖNETİMİ.....	86
İşlem Yönetimini Anlama -ps	86
Foreground ve Background İşlemleri	88
İşlem Önceliğini Ayarlama - nice, renice	89
İşlem Sonlandırma - kill, killall	90
9 - SİSTEM PERFORMANSINI GÖRÜNTÜLEME	91
Performans Ölçüm Araçları – top, free, watch, vmstat, htop, sysstat	91
top	91

free.....	92
watch free	92
vmstat.....	93
htop.....	93
iostat ve pidstat.....	94
Komutlar - lspci, lscpu, lshw, lsblk, lsmod	95
Cockpit Kullanımı	96
10 - Zaman Senkronizasyonu.....	97
NTP Nedir?.....	97
Komutlar - date, hwclock.....	98
Komutlar - timedatectl.....	99
Chrony ile NTP Senkronizasyonu.....	100
11 - YAZILIM YÖNETİMİ	103
apt install/update/upgrade/remove/purge/search/list.....	103
Kaynaktan Yazılım İndirme - wget, dpkg.....	105
.deb Paketi İle Kurulum Yapmak.....	106
Sıkıştırılmış Dosyalar İle Kurulum Yapmak.....	107
Mirror Değiştirmek.....	108
12 – Boot (Ön İndirme) İşlemi Anlama ve Yönetme.....	108
Boot İşlem Sırasını Anlama	108
MBR ve GPT nedir?	109
GRUB 2 Nedir?.....	110
GRUB2 Dosyaları.....	111
Systemd Nedir?	113
Service Unit.....	117
Target Unit.....	119
Recovery Menü.....	120
Root Parolası Resetleme	121
13 - Kernel Yönetimi.....	124
Linux Kernel Nedir?	124
Kernel Modüllerini Yönetme	125
Modülde Değişiklik Yapmak	127
Kernel Versiyonunu Anlama	128
Kernel Versiyonunu Güncelleme	129
Eski Kernel Versiyonlarını Kaldırmak.....	130
Kernel Downgrade	131
Komutlar - uname, hostnamectl, dmesg	131

Donanım Tanımlama	133
BIOS (Basic Input/Output System).....	133
UEFI (Unified Extensible Firmware Interface)	133
Aygıt Arayüzleri.....	133
Donanım Dizinleri	134
Aygıtları Tanımlamak.....	135
Donanım Aygıtları için Driver Kontrolü.....	136
14 - NETWORK.....	136
TCP/IP Protokolünü ve IP Adreslerini Anlama	136
MAC Adreslerini Anlama	138
Temel İletişim Port ve Protokollerİ	138
IP Adresi ve Interface Yönetimi.....	138
Network Ayarlarının Yapılandırılması ve Doğrulanması	139
DHCP Yapılandırması	139
Static IP Yapılandırması.....	140
Komutlar	141
ifconfig	141
ip	141
ping.....	144
traceroute.....	145
mtr.....	146
nslookup.....	146
dig	148
networkctl.....	149
netstat.....	149
ss	151
Hostname, hosts ve resolve.conf Dosyasının Düzenlenmesi	151
Routing Table.....	153
15 - Disk Yönetimi	155
Linux'da Aygit İsimleri.....	155
Dosya Sistemleri.....	155
Komutlar – lsblk, df, du	156
lsblk.....	156
df	157
du	158
fdisk, fstab ve partprobe nedir?.....	159

fdisk	159
partprobe	159
Disk Ekleme - fdisk.....	160
Disk Genişletme	164
Mevcut Diskde İkincil Bölüm Oluşturmak	165
GPT Bölüm Oluşturmak - gdisk.....	167
Sistem Diskini Genişletme - Gparted	168
Swap Alanı Oluşturmak ve Genişletmek	171
Diski Sunucudan Kaldırmak	173
16 - LVM (Logical Volumes) Yönetimi.....	174
LVM Nedir?	174
Extents - LE ve PE Mapping.....	175
Linear Logical Volumes	176
Striped Logical Volume	176
Mirrored Logical Volume	177
LVM Yapılandırma Dosyası.....	177
LVM Yapısını Oluşturmak (Creating)	178
Volume Gruba Disk Eklemek	183
LVM'i Yeniden Boyutlandırmak (Resizing).....	185
Logical Volume boyutunu artırmak (lvmextend)	185
Physical Volume'un boyutunu artırmak (pvextend).....	185
Logical Volume Boyutunun Azaltmak.....	189
Striped Volume Oluşturmak.....	190
Mirrored Volume Oluşturmak	192
Remove – Logical Volume ve Volume Group.....	192
Recover Failed Disk.....	193
Snapshot Volume	195
LVM Snapshot Restore	196
LVM Özellikleri.....	196
17 - Firewall Yapılandırması	197
Firewall Nedir?	197
ufw Yapılandırması	198
Kural Ekleme ve Silme	199
Ufw logging.....	200
19 - AppArmor Yönetimi.....	201
AppArmor Nedir?.....	201

AppArmor Yapılandırması.....	203
19 - BACKUP ve RESTORE.....	206
Backup Nedir?	206
RPO - RTO nedir?.....	207
Backup Çeşitleri? – Full, Incremental, Sentetik, Differential.....	208
tar, scp, dump Kullanımı	209
tar ile arşivleme.....	209
tar ile arşivden çıkarma	211
Arşive Dosya Ekleme.....	212
Arşivi Uzak bir Sunucuya Gönderme - scp	212
dump Kullanımı.....	213
Rsync	216
21 - LOGLARI ANLAMAK	217
Log Dosyalarını Okumak.....	218
Logları Anlamak.....	219
journald ve journalctl.....	219
Rsyslogd	221
Uygulama Loglarının Rsyslog ile Toplanması	223
Log Döngüsünü Ayarlama – logrotate	224
21 - GÖREVLERİ ZAMANLAMA	226
Cron Servisini Yönetme.....	226
Cron Zamanlamasını Anlama.....	227
Cron Configurasyon Dosyasını Düzenleme	227
Örnek Bir Görevi Zamanlamak.....	228

1 - LINUX DOSYA SİSTEMİ VE UBUNTU KURULUMU

Linux Dosya Sistemi Yapısı

Bir Linux sistemini yönetmek için varsayılan dizinlere aşina olmanız gereklidir. Linux'da Filesystem Hierarchy Standard (FHS)'na göre tüm dosyalar ve dizinler, farklı fiziksel veya sanal cihazlarda depolanmış olsalar bile / kök dizini yani root altında görünür. Bu dizine sadece root kullanıcısı yazma hakkına sahiptir.

/	/bin	Temel komut dosyalarını barındıran dizindir. ping, ls, cp, cat vs.
	/boot	Kernel'in boot etmesi gereken bütün dosya ve dizinleri içerir. Bu dizinin silinmesi halinde sistem bir daha açılamaz hale gelir.
	/dev	Fiziksel donanıma erişim gereklili dosyaları barındırır. Boot sırasında kullanılır. Örn: /dev/sda
	/etc	Sistemde kullanılan tüm servis ve programlarının yapılandırma dosyalarını içerir. Örn: /etc/resolv.conf
	/home	Lokal kullanıcıların dosyalarının barındırıldığı ana dizindir.
	/lib, /lib64	/boot, /bin ve /sbin tarafından kullanılan kütüphaneleri barındırırlar.
	/media, /mnt	Dosya sistemine dahil edilen harici depolama cihazları kullanılır.
	/opt	Uygulamaların opsyonel olarak kullandığı paketler için kullanılır.
	/proc	Sistem çalışan işlemler ve kaynaklar hakkında bilgileri içerir. Örn: /proc/uptime
	/root	Root kullanıcısının dosya dizinidir. / ana dizin ile karıştırılmamalıdır.
	/snap	Uygulamalar için yeni nesil paket sistemidir.
	/run	En son boot işleminde sonra oluşan kullanıcı ve işlem bilgilerini içerir.
	/sbin	Sistem administratorleri tarafından kullanılan komut dosyalarını barındırır. Örn: iptables, reboot, fdisk, ifconfig
	/srv	HTTP, NFS, FTP gibi servislerle ilişkili veriler için kullanılır.
	/tmp	Geçici dosyaların depolanması için kullanılır. Boot sırasında bu alandaki dosyalar silinir.
	/usr	Programların, dosyaları ve dokümanları için kullanılan alt dizinlerini barındıran dizindir.
	/var	Dinamik olarak boyutu değişen log, mailbox gibi dosyaları barındıran dizindir.

Linux Dosya Tipleri

Linux dosya sisteminde gezinirken farklı dosya türleriyle karşılaşacağız. En çok kullanılanlar regular file diye tabir ettigimiz normal dosyalar ve dizinlerdir. Bununla birlikte, Linux işletim sisteminde, başka dosya türleri de bulunmaktadır.

Dosya tiplerini tanımlamak için ls -lah komutunu kullanabiliriz. Komut çıktısının en başındaki sembol bizi dosyanın tipini anlamamıza yardımcı olacaktır.

```
ozgur@ubuntu:~$ ls -lah
total 72K
drwxr-xr-x 5 ozgur ozgur 4.0K Jan 31 16:12 .
drwxr-xr-x 3 root  root  4.0K Jan 25 21:25 ..
-rw-rw-r-- 1 ozgur ozgur    0 Jan 31 15:26 abc1
-rw-rw-r-- 1 ozgur ozgur    0 Jan 31 15:26 abc2
```

- : Regular File

d : Directory

c : Character Device File

b : Block Device File

s : Local Socket File

p : Named Pipe

l : Symbolic Link

Regular File

Normal dosya, Linux sisteminde bulunan en yaygın dosya türüdür. Metin dosyaları, resimler, kitaplıklar vb. farklı dosyaları yönetir. Touch komutuyla normal bir dosya oluşturabilirsiniz.

Directory

Directory yani dizinler, Linux'ta bulunan en yaygın ikinci dosya türüdür. Directory'i mkdir komutuyla oluşturulabilirsiniz.

Character device

Character ve blok aygıt dosyaları, kullanıcıların ve programların, donanım çevre aygıtlarıyla iletişim kurmasına olanak tanır.

```
ozgur@ubuntu:/dev$ ls -lah | grep vcs
crw-rw---- 1 root tty      7,   0 Feb  2 01:26 vcs
crw-rw---- 1 root tty      7,   1 Feb  2 01:26 vcs1
```

Block device

Harddisk, bellek gibi depolama cihazları için kullanılır.

```
ozgur@ubuntu:/dev$ ls -lah |grep sda
brw-rw---- 1 root disk     8,   0 Feb  2 01:26 sda
brw-rw---- 1 root disk     8,   1 Feb  2 01:26 sda1
```

Local Domain Sockets

İşlemler arasındaki iletişim için kullanılır. Genellikle X windows, syslog vb. hizmetler tarafından kullanılırlar.

```
root@client:/# ls -ltr /run/systemd/journal/syslog
srw-rw-rw- 1 root root 0 Feb 25 06:08 /run/systemd/journal/syslog
```

Named Pipes

İki local işlem arasında iletişime izin verir.

```
root@client:/# ls -ltr /run/initctl
prw----- 1 root root 0 Feb 25 06:08 /run/initctl
```

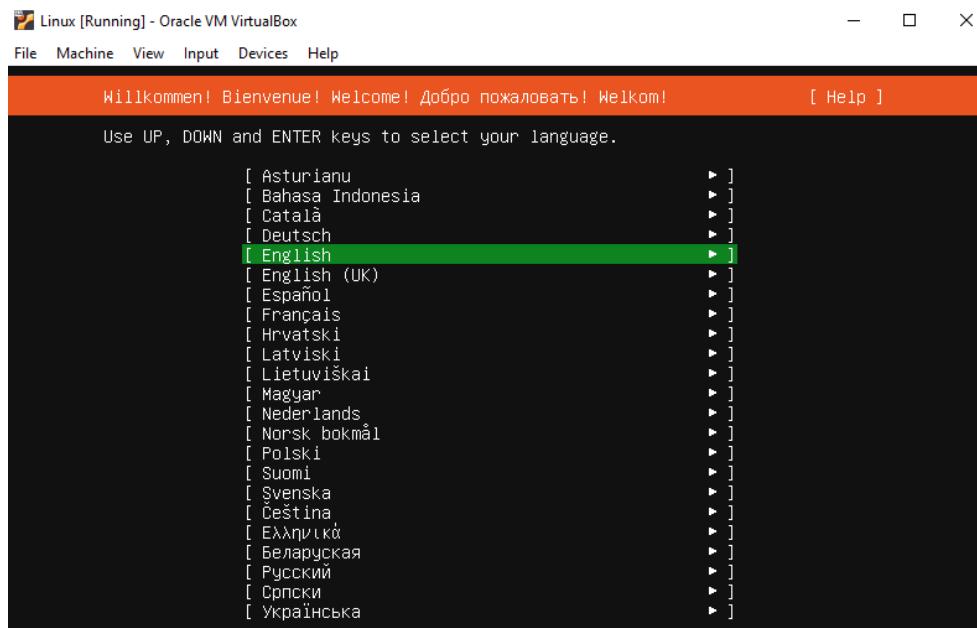
Symbolic Link

Bir dizin, dosya veya uygulama için oluşturulan kısayollar diyebiliriz. İlerleyen konularda farklarını ve kullanım şekillerini göreceğiz.

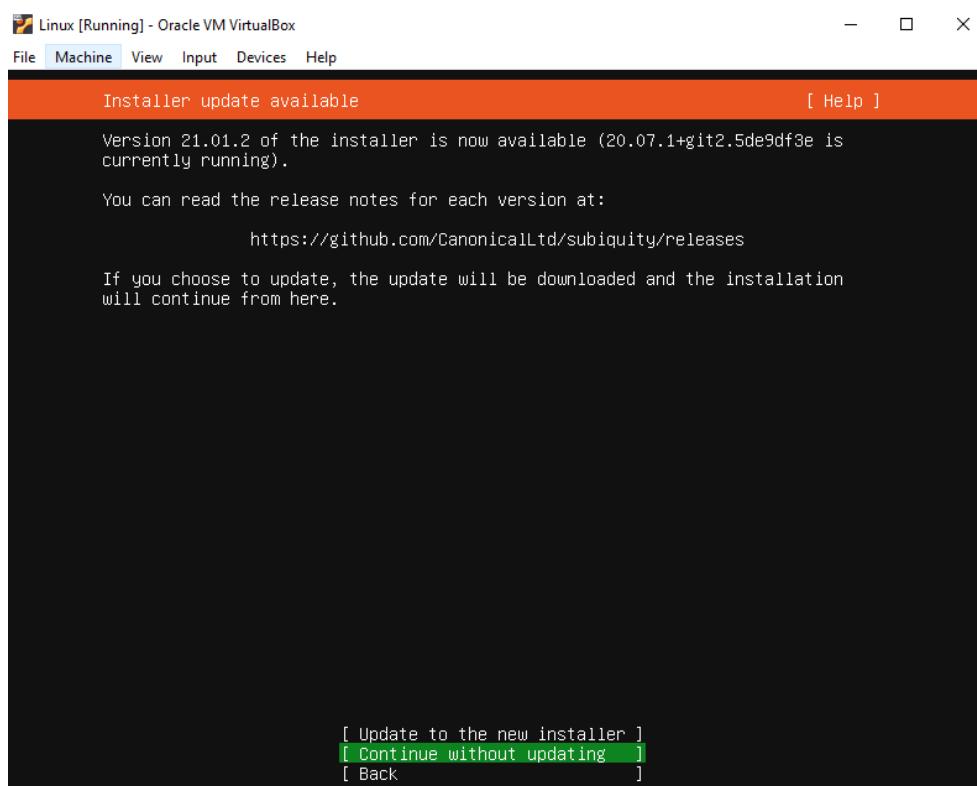
```
ozgur@ubuntu:/dev$ ls -lah /dev/log
lrwxrwxrwx 1 root root 28 Feb  2 01:26 /dev/log -> /run/systemd/journal/dev-log
```

Linux İşletim Sistemi Kurulumu

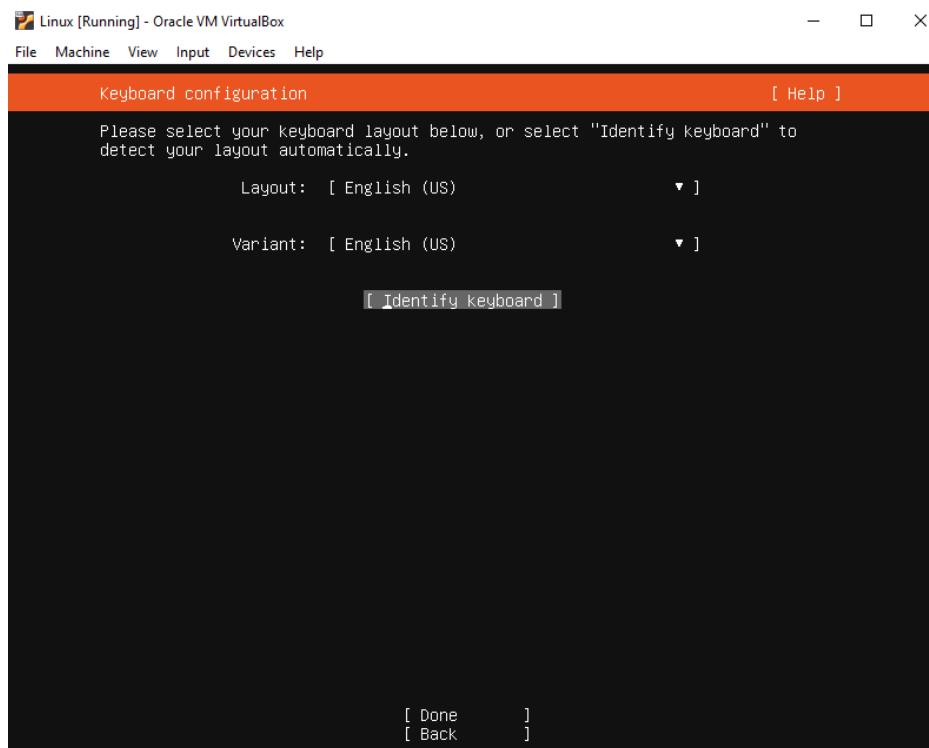
Hem Hyper-V hem de VirtualBox için kurulum işlemleri aynıdır. Her iki platformda da sanal sunucunun CD/DVD Rom'una imaj dosyanız yolunu gösterdiğinizde sunucu ilk açılış sırasında imajdan başlayacak ve siz kurulum sihirbazı ile karşılaşacaktır. Dil seçeneği ile kurulum işlemlerine başlıyoruz. Tavsiyem kuracağınız işletim sistemi ister Windows ister Linux olsun, kurulum dilini İngilizce seçmeniz yönündedir. Bunun nedeni kullanacağınız bazı uygulamaların veya entegre olarak çalıştıracağınız yapıların Türkçe dil desteği olmaması durumunda problemlerle karşılaşmanızın olası olmasıdır. İngilizce seçeneği devam ediyorum. Klavye dili ile ilgili düzenlememiz ilerleyen pencerelerde olacaktır.



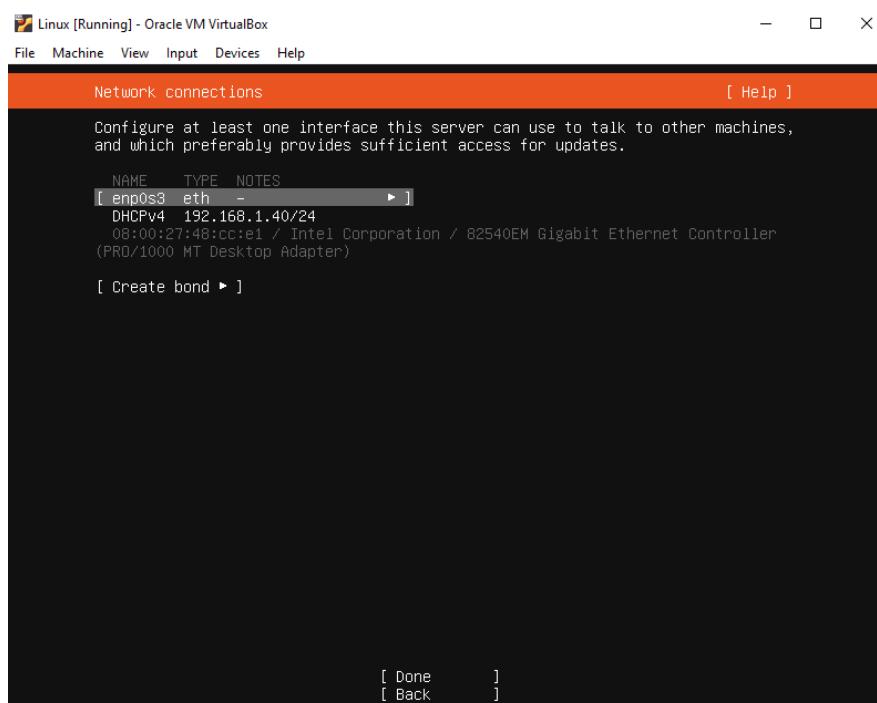
İmajınızın daha güncel bir versiyonu çıktıysa güncelleme yapıp yapılmayacağı sorulur. Güncelleme yapmadan devam ediyoruz.



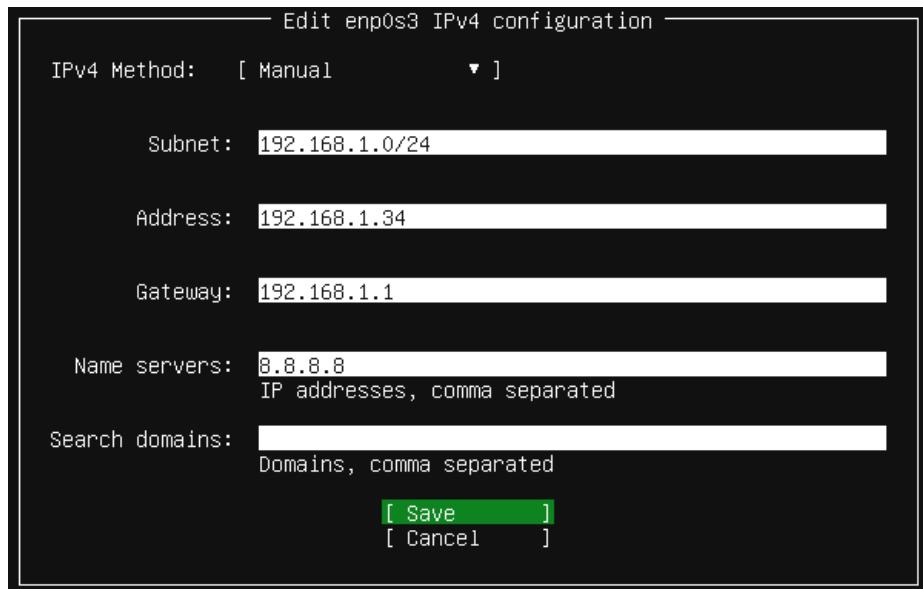
Klavye dili ile ilgili Identify Keyboard seçeneğiyle klavyemizdeki belirli tuşların sorgulaması yapılır ve kullandığımız klavye tanımlanarak otomatik olarak seçim ekranına getirilir.



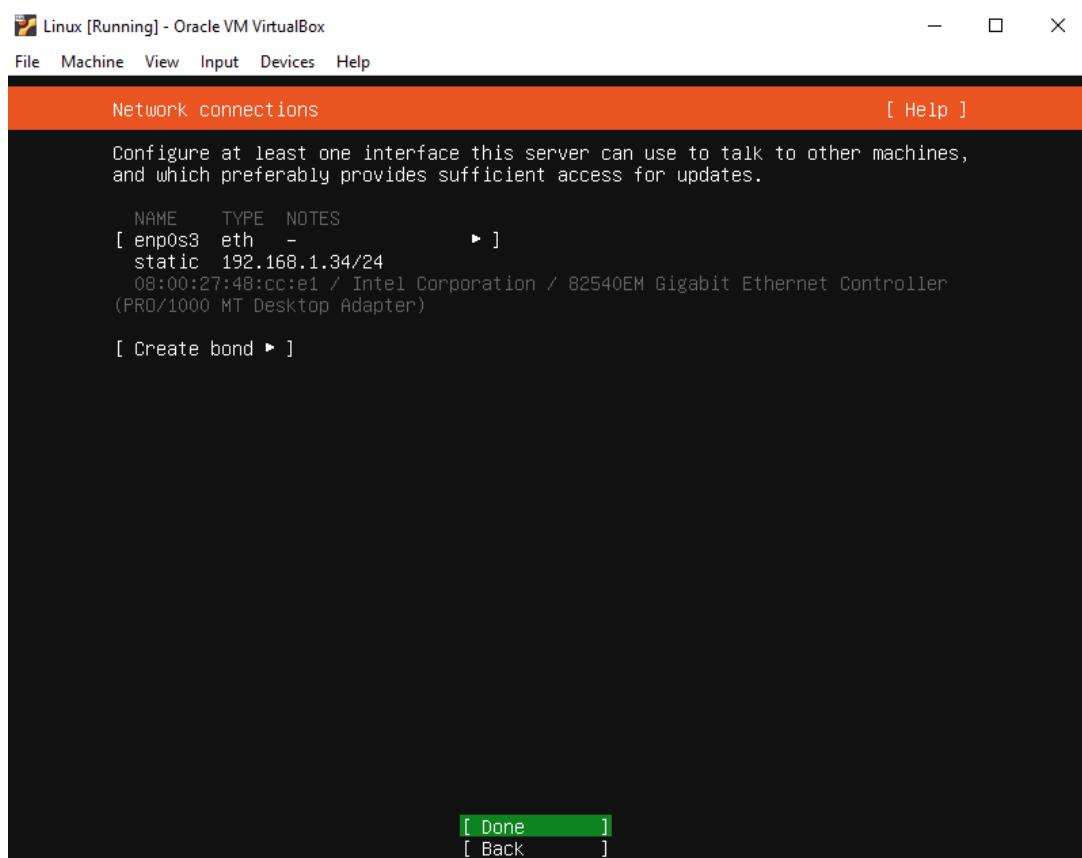
İlk kurulum sırasında IP adresiniz DHCP protokolü tarafından otomatik olarak atanır. DHCPv4 ile başlayan satırı dikkat ederseniz bize atanın IP 192.168.1.40'dır. /24 ibaresi ise Subnet Mask'ı ifade eder. Eğitimimiz kapsamında olmamakla birlikte bu mask ile 192.168.1.0 networküne sahip olduğumuzu ve 192.168.1.1 Getway olmak üzere (ki elbette değişimelidir ancak ev kullanıcıları için genelde 1 gateway'dir yani interne çıkış sağladığımız kapı) 192.168.1.2'den başlayıp – 192.168.1.254'e kadar olan tüm IP leri kullanabileceğimizi anlıyoruz. 192.168.1.255 adresi Broadcast adresi olduğu için kullanılamaz. Broadcast adresi, aynı network içerisinde bulunan cihazların birbirlerine çağrı yapması için kullanılır. Biz IP adresimizi el ile atayacağız. Yani sabit bir IP adresi kullanacağız.



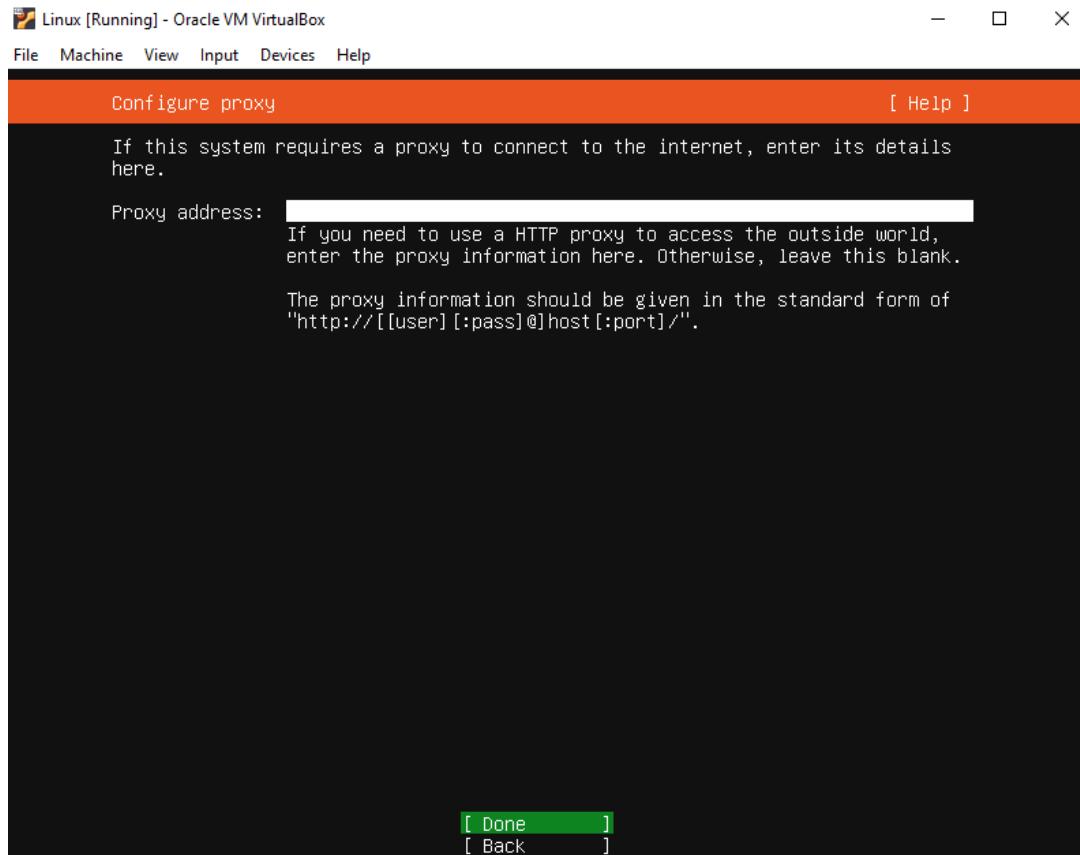
Sabit IP vermek için enp0s3 ismiyle başlayan Ethernet interface'ini seçip enter'a basıyoruz. Edit IPv4 seçip, IPv4 Method seçeneğini Automatic(DHCP) den Manual e çeviriyoruz. Karşımıza IP bilgilerimizi gireceğimiz bir çıkıyor. Subnet, Address, Gateway, Name Servers bilgilerini girip save ile kaydediyoruz.



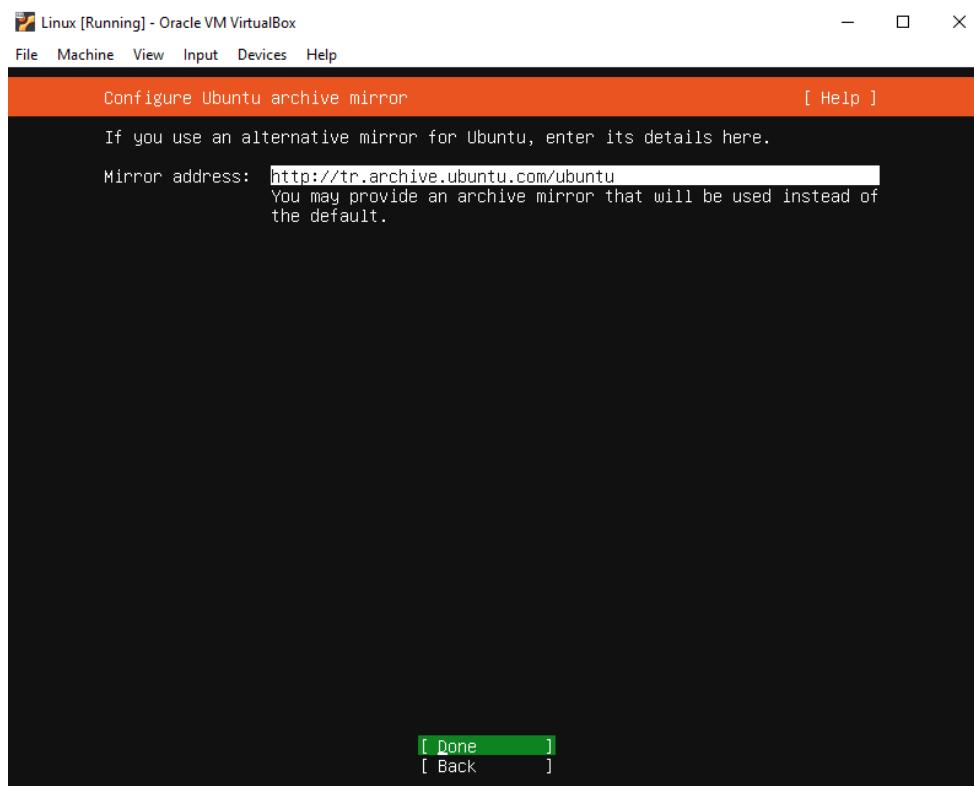
Kayıt işlemi tamamlandıktan sonra IP mizin static olarak değiştiğini görüyoruz.



Proxy adresi belirtmeden devam ediyoruz.

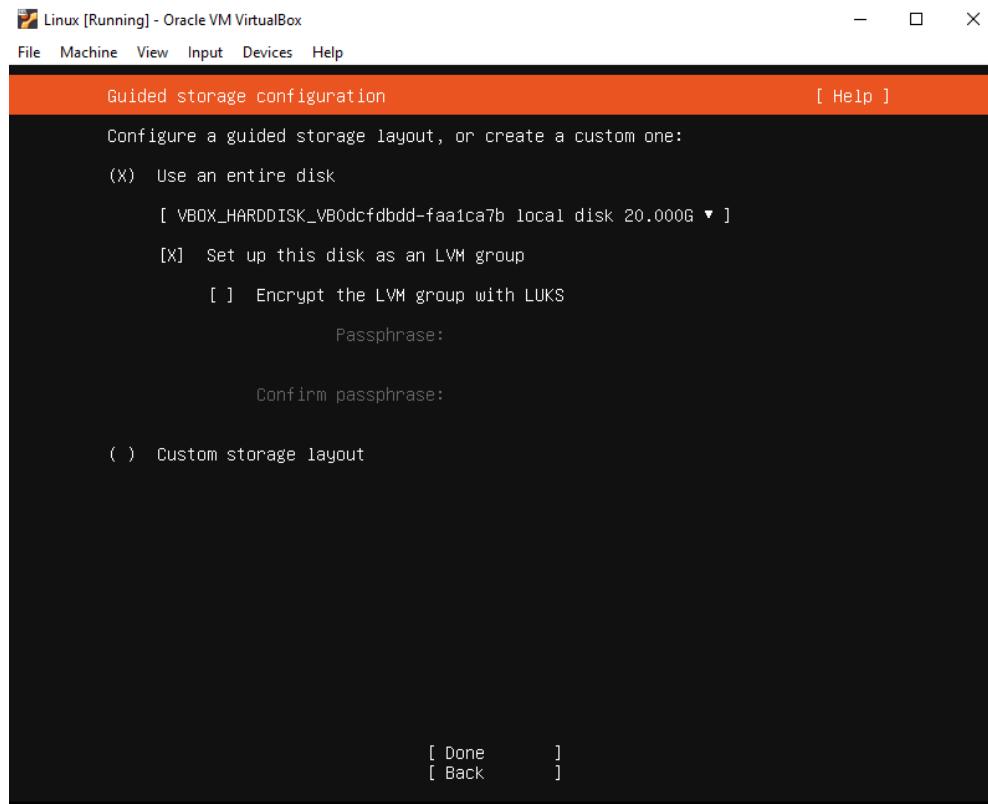


Mirror adresi varsayılan olarak bırakıp devam ediyoruz.

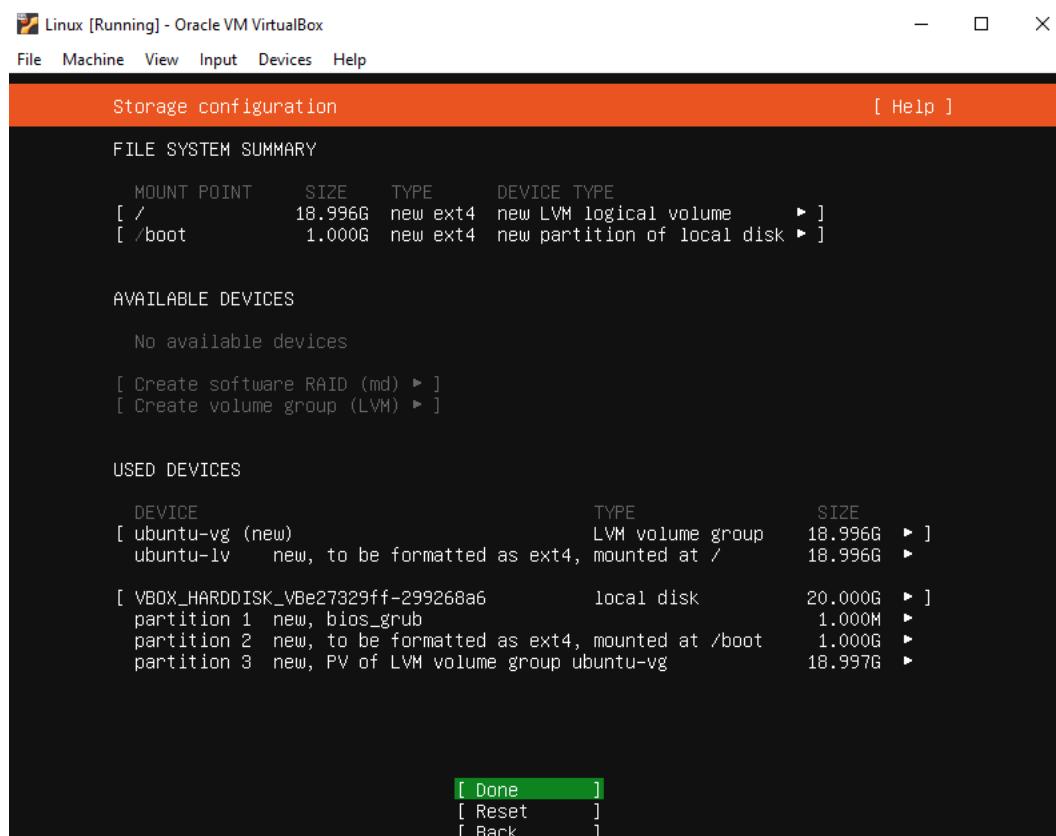


Disk konfigürasyonunu tüm diski kullan olarak seçili bırakıp Logical Volume Manager (LVM)'in bütün ayarları otomatik yapmasını sağlıyoruz. Custom Storage Layout seçeneği disk yapılandırmasında root, boot, swap, home, var, lib ve usr dizinlerinin ayarlarını manuel olarak yapmamıza imkan tanıyor. İlerleyen bölümlerde bu konuya hem disk yapılandırmalarında hem de LVM'de deşinmekle birlikte LVM disk yönetimi konusunda çok

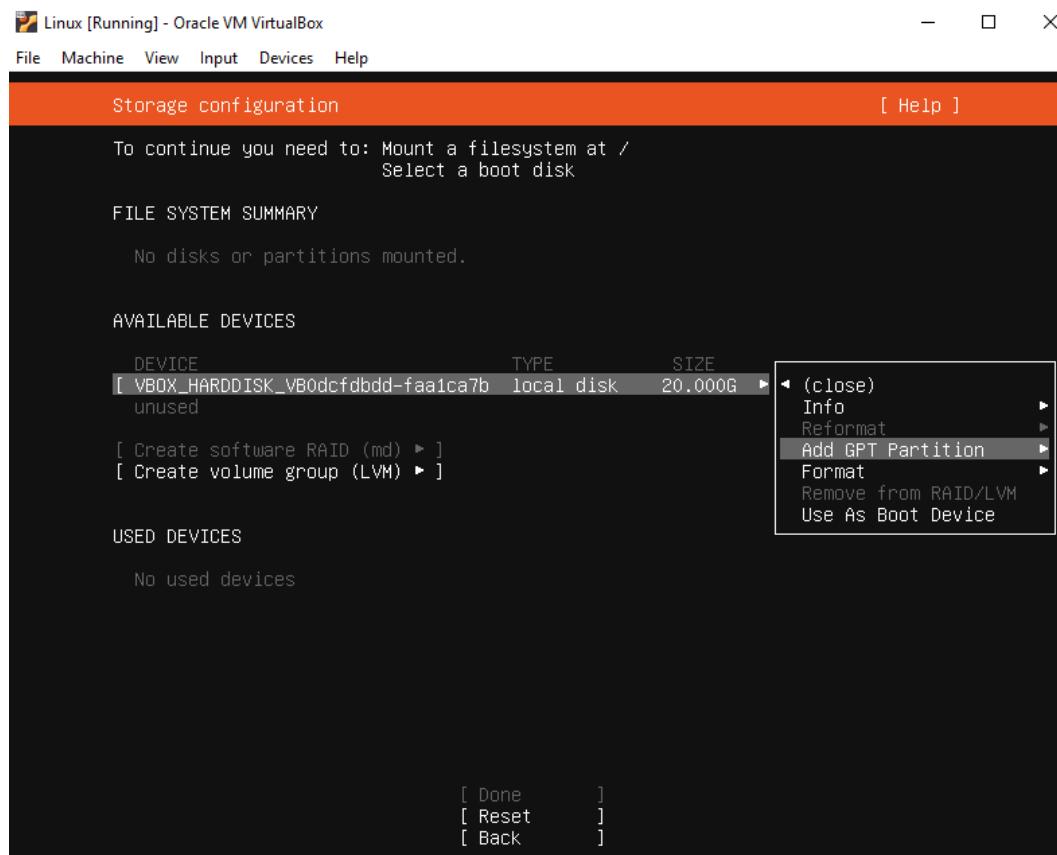
büyük kolaylıklar sağlıyor. Elbette her ikisinin de farklı kullanım alanları bulunmaktadır. LVM seçerek devam edeceğiz ancak Custom Storage Layout seçeneklerini ve ayalarını da görelim.



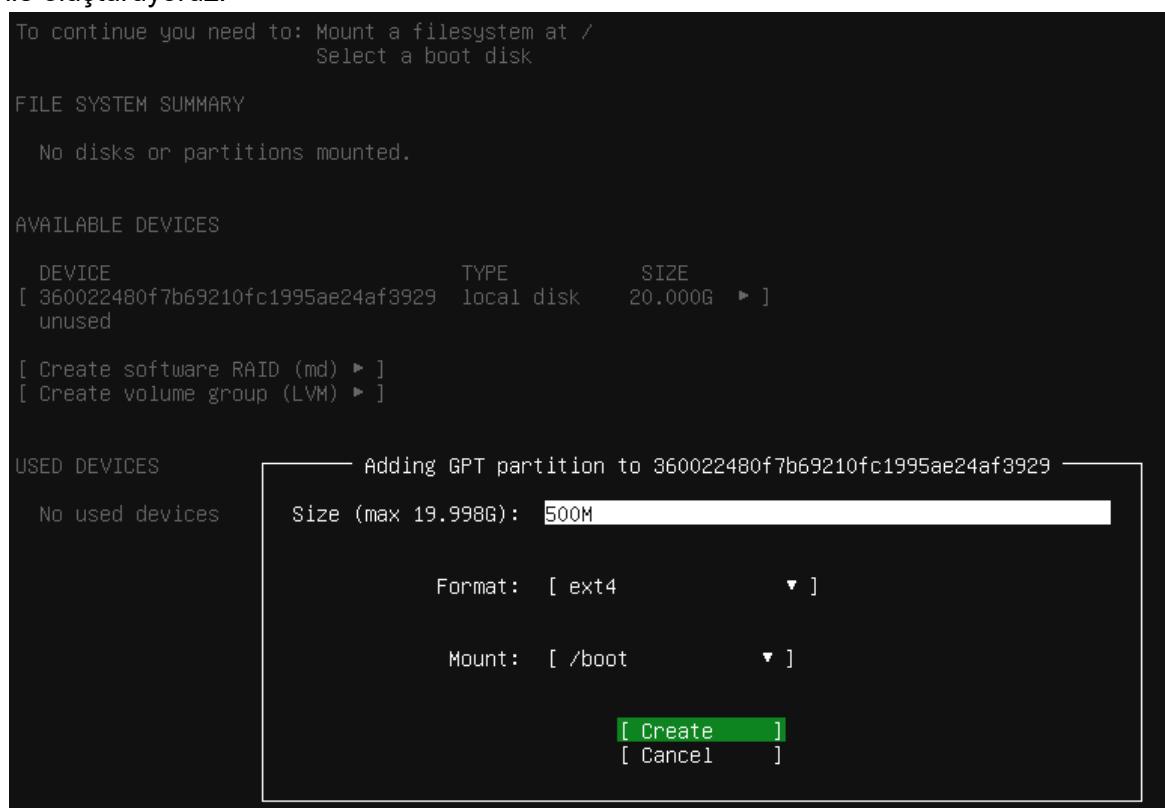
Dilerseniz LVM tarafındanki bölümlerde de değişiklik yapabiliyoruz. Varsayılan olarak bırakıp devam ediyoruz.



Eğer disk bölümlerini el ile oluşturmak istersek klavye ile sanal diskin üzerine geldikten sonra Add GPT Partition ile alan oluşturma işlemine başlıyoruz.



Ekrandaki görüntüden de görüleceği üzere root ve boot alanlarını oluşturduk.
Oluşturacağınız alanın, Size ile boyutunu, mount ile hangi dizine bağlanacağını seçip create ile oluşturuyoruz.



Yapilandırmamız ekran görüntüsündeki gibi oluyor. Değişiklik yapmak isterseniz Reset ile tüm ayarları silip en başa dönenbilirsiniz veya sadece belirli bir dizin için değişiklik yapabilirsiniz. Bölümleri oluşturma sırası önemli root alanını en son bölüme yerleştirmelisiniz. Root diğer bölümlerin arasında kalırsa disk artırımı yapmak isterseniz bu işlemi gerçekleştiremezsiniz.

```
FILE SYSTEM SUMMARY

MOUNT POINT      SIZE      TYPE      DEVICE TYPE
[ /              19.508G  new ext4  new partition of local disk ▶ ]
[ /boot          500.000M new ext4  new partition of local disk ▶ ]

AVAILABLE DEVICES

No available devices

[ Create software RAID (md) ▶ ]
[ Create volume group (LVM) ▶ ]

USED DEVICES

DEVICE                      TYPE      SIZE
[ 360022480f7b69210fc1995ae24af3929    local disk  20.000G ▶ ]
partition 1  new, bios_grub                  1.000M ▶
partition 2  new, to be formatted as ext4, mounted at /boot  500.000M ▶
partition 3  new, to be formatted as ext4, mounted at /       19.508G ▶
```

Disk yapılandırmasını tamamladıktan sonra Done ile devam ediyoruz. Karşımıza diskin formatlanacağına dair bir uyarı çıkıyor. Yapılandırmamızın doğru olduğundan emin isek devam ediyoruz.

```
Storage configuration

FILE SYSTEM SUMMARY

MOUNT POINT      SIZE      TYPE      DEVICE TYPE
[ /              19.508G  new ext4  new partition of local disk ▶ ]
[ /boot          500.000M new ext4  new partition of local disk ▶ ]

AVAILABLE DEVICES

No available devices

[ Create software RAID (md) ▶ ]
[ Create volume group (LVM) ▶ ]

USED DEVICES

DEVICE
[ 360022480f7b692
partition 1  ne
partition 2  ne
partition 3  ne
```

— Confirm destructive action —

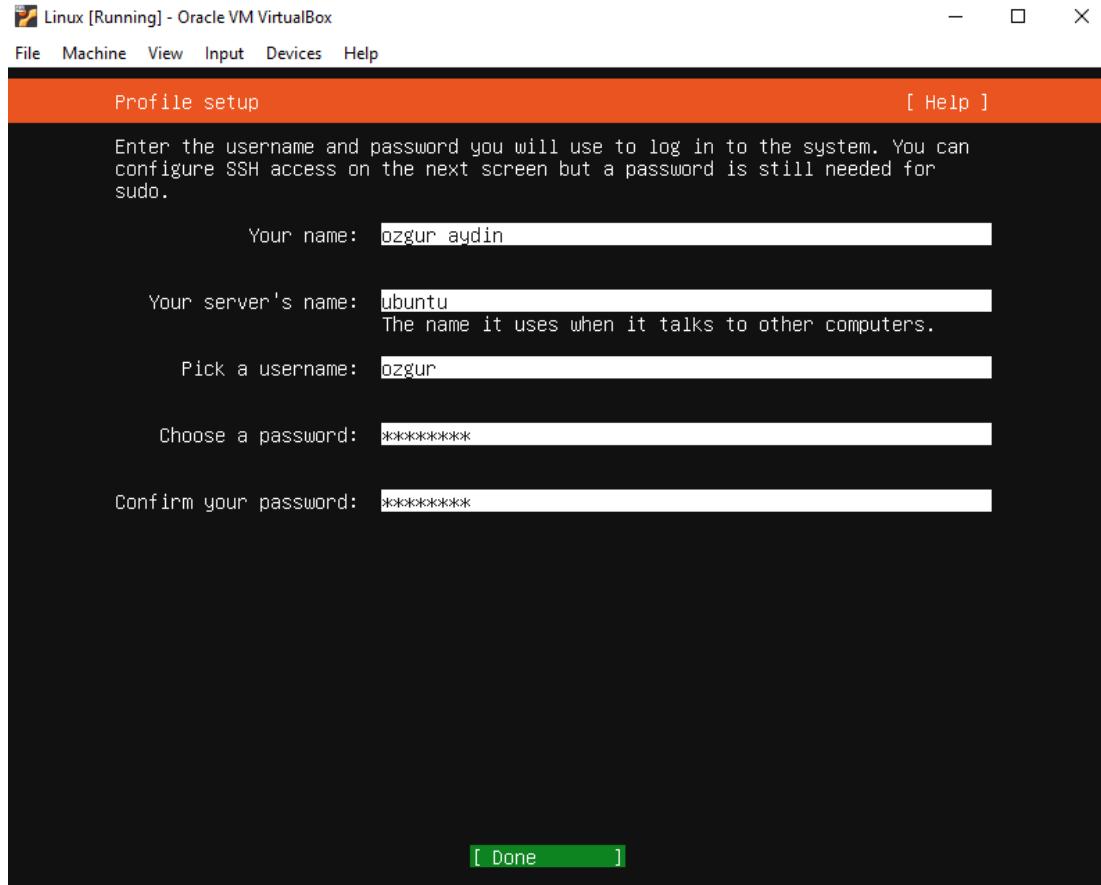
Selecting Continue below will begin the installation process and result in the loss of data on the disks selected to be formatted.

You will not be able to return to this or a previous screen once the installation has started.

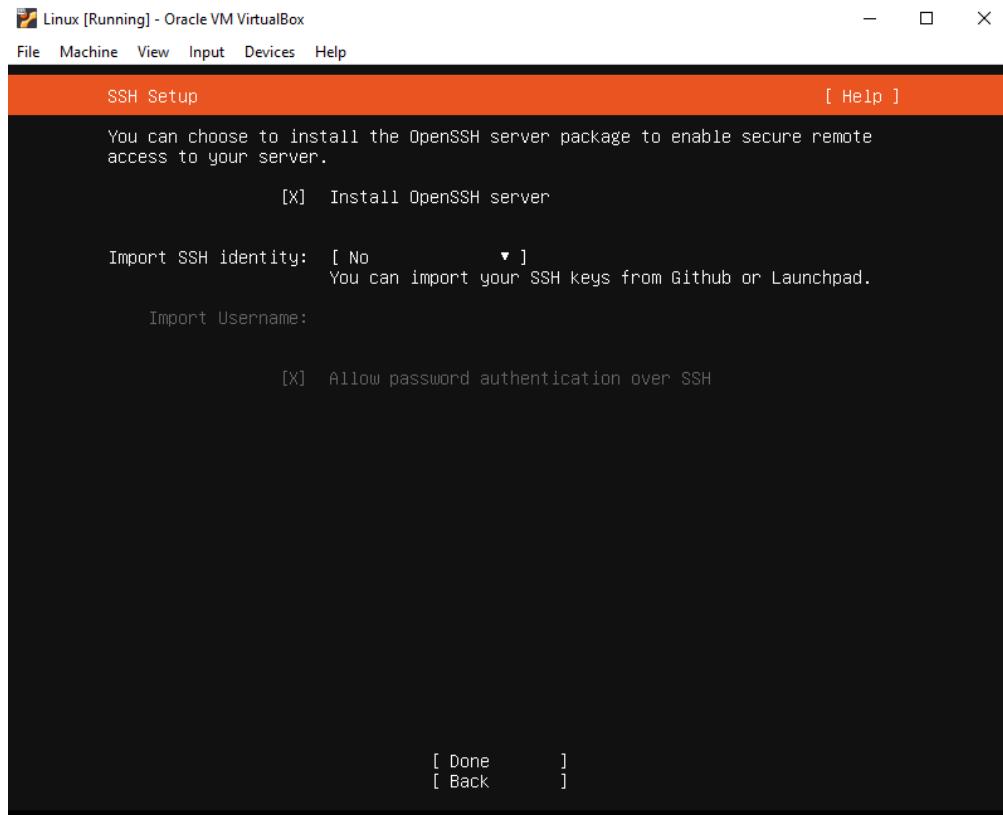
Are you sure you want to continue?

[No]
[Continue]

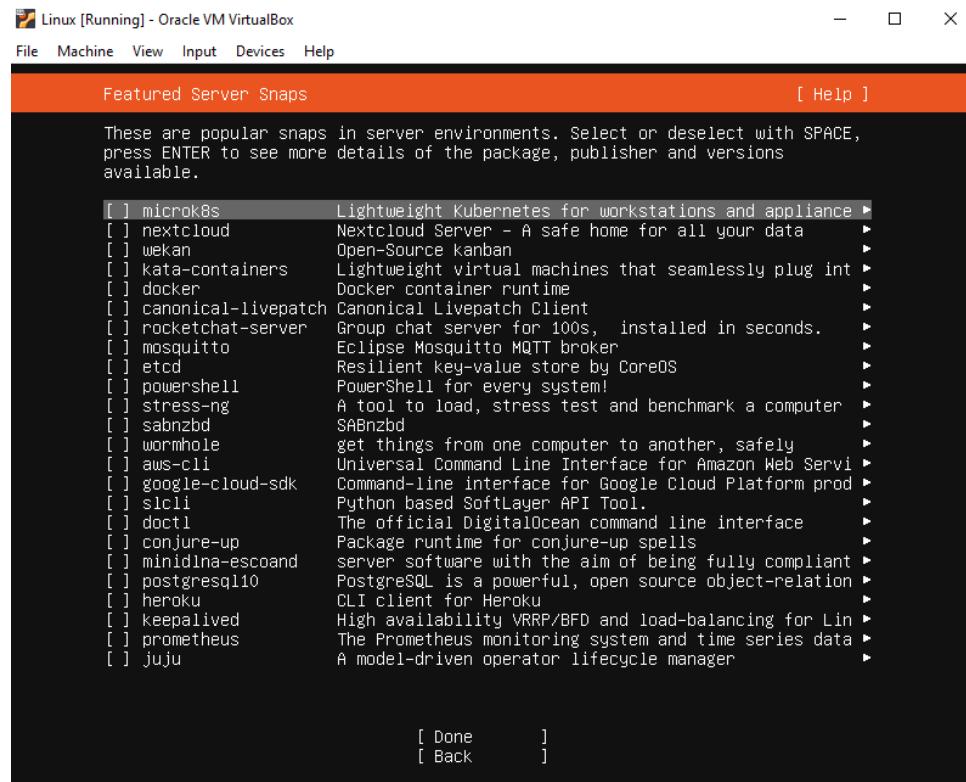
Sunucumuza erişebilmek için kullanıcı bilgilerini, sunucu adını ve parola bilgisini giriyoruz. Username bölümü sizin sunucuya login olurken kullanacağınız kullanıcı adınız olacak. Internete erişim sağlayan bir sunucu dakikalar içinde Brute Force Attack saldırılara maruz kalabilir. Bu nedenle sunucularda kullanılacak parolaları güçlü yapmakta fayda var.



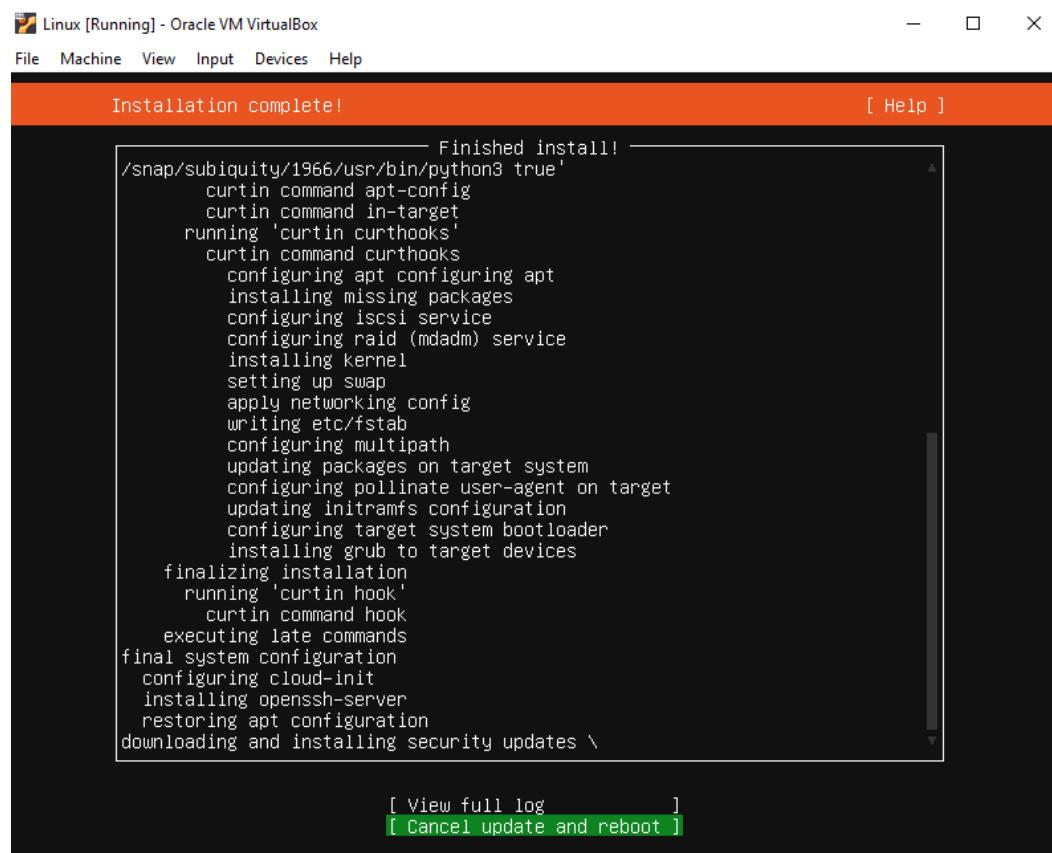
Sunucumuza erişim sağlamak için SSH uygulaması kurmamız gerekiyor. Linux sunucularda kullanılan en yaygın uygulama OpenSSH 'tir. İşaretleyip kurulum sırasında yüklenmesini sağlıyoruz.



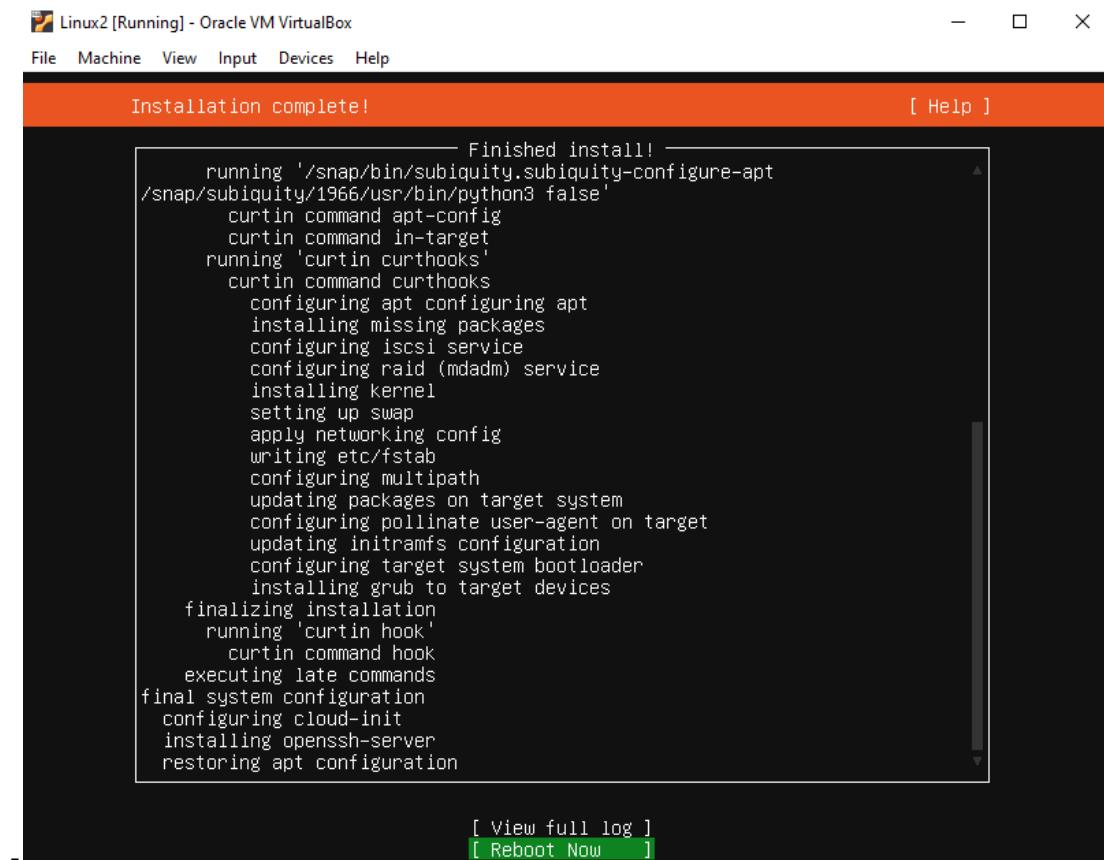
Ubuntu çeşitli uygulamalar ve ortamlar için destek sağlar. Listede Ubuntu üzerine kurulan en yaygın uygulamalar görülür ancak çok daha fazlası vardır. Seçim yapmadan done ile kurulumu başlıyoruz.



Kurulum tamamlandığında güncellemelerin bitmesini bekleyebilirsiniz. Cancel update and reboot seçeneği ile sunucunuzu yeniden başlatıp login olarak kullanmaya başlayabilirsiniz.



Güncellemeler tamamlandıında Reboot Now ile sunucuyu yeniden başlatıp kullanıcı bilgileriniz ile login olabilirsiniz.

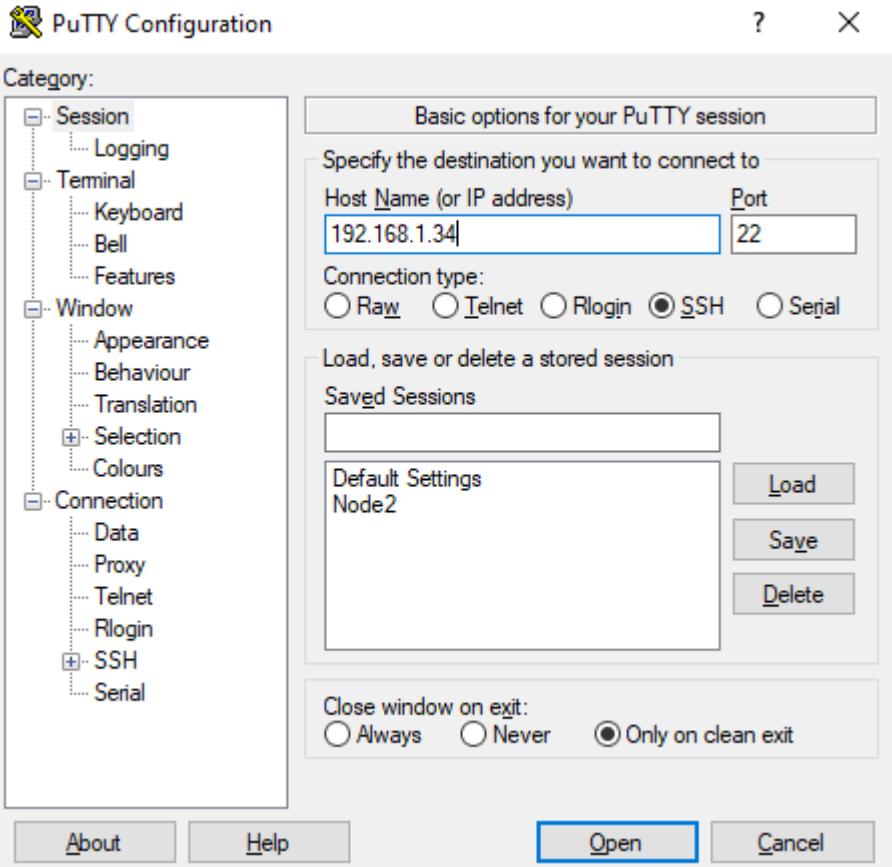


Putty ile İşletim Sistemine Erişmek

Sunucu kurulumu tamamlandıktan sonra olurda verdığınız IP'yi unutursanız konsol ekranından kurulum sırasında belirlediğimiz kullanıcı adı ve şifre ile sunucumuza login oluyoruz. ip a komutu ile sunucumuza bağlı ağ adaptörlerini (network interface) ve bilgilerini görüntüleyiyoruz. 2:Enp0s3 ile başlayan satırı dikkat edecek olursak IP mizin 192.168.1.34 olduğu görülmüyor. Sizin tarafınızda bu IP farklı olacaktır. Aynı kontrolü sizde yaparak IP nizi öğrenebilirisiniz.

```
ozgur@ubuntu:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:48:cc:e1 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.34/24 brd 192.168.1.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe48:cc:e1/64 scope link
        valid_lft forever preferred_lft forever
```

www.putty.org adresinden PUTTY uygulamasını bilgisayarınıza indirip kurduktan sonra sunucu IP'si ve varsayılan bağlantı portu olan 22 ile sunucunuza Secure Shell (SSH) bağlantısı yapabilirsiniz.



Window kullanıcıları isterlerse Powershell üzerinden ssh kullanıcı_adınız@sunucu_ip_adresi ile sunucuya erişim sağlayabilirler. SSH ile bağlantı sorunu yaşarsanız eğer sshd_config dosyasında bazı düzenlemeler yapmak gerekiyor. Bunun için SSH Yapılandırması bölümünden faydalananabilirsiniz.

```
PS C:\Windows\system32> ssh ozgur@192.168.1.34
The authenticity of host '192.168.1.34 (192.168.1.34)' can't be established.
ECDSA key fingerprint is SHA256:MQ81C9dB/NMD91CBpGFeePrAQGHa+j2ygv3/hA2dYtI.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.34' (ECDSA) to the list of known hosts.
ozgur@192.168.1.34's password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-65-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue 02 Feb 2021 09:35:55 PM UTC

System load: 0.0          Processes:           105
Usage of /home: 0.3% of 1.91GB  Users logged in:      1
Memory usage: 20%          IPv4 address for enp0s3: 192.168.1.34
Swap usage:   0%         

Last login: Tue Feb  2 20:37:16 2021
ozgur@ubuntu:~$
```

2 - TEMEL LINUX KOMUTLARI

Terminoloji

Archive : Birden çok dosya içeren, genellikle depolama alanından tasarruf etmek için sıkıştırılmış tek bir büyük dosya. Genellikle bilgisayarlar arasında aktarımı kolaylaştırmak için oluşturulur. Örnek arşiv biçimleri TAR, ZIP, RAR vb.

Background Process : Kullanıcı girişi olmadan çalışan bir programlardır.

Bash : Bourne-Again Shell sözcüklerinin kısaltılmıştır. Linux sistemler için yazılan komutları yorumlayan komut dili yorumlayıcısıdır.

Binaries : Yürütilebilir programlara derlenmiş kaynak kodlarıdır.

Boot : Bilgisayarı başlatmak ve komut satırından bazı temel programları çalışıtmak için bir işletim sistemi içeren alandır.

Cron : Belirtilen görevleri belirli bir zaman veya aralıkta yürüten bir Linux arka plan programıdır.

Dpkg (Debian Paket Yöneticisi) : Debian Linux'ta bulunan ancak diğer dağıtımlarla uyumlu, Internet yüklemeleri için bir paketleme ve yükleme aracıdır.

File System : Bir işletim sisteminin içeriğine, bir disk veya başka bir depolama ortamının nasıl erişip yorumlayacağını söyleyen bir dizi programdır.

Foreground Process : Kullanıcının şu anda etkileşimde olduğu programdır.

FTP (File Transfer Protocol) : Diğer bilgisayarlara dosya aktarma yöntemidir.

GNU (GNU is Not Unix) Project : Massachusetts Institute of Technology (MIT) Free Software Foundation (FSF)nın tescilli UNIX uygulamalarına alternatifler geliştirme ve teşvik etme çabasıdır.

GPL (GNU General Public License) : Genel kullanım için sunulan kamu lisansıdır.

Kernel : Bir bilgisayardaki bütün kaynakların yönetimini sağlayan işletim sistemi nin çekirdeğidir.

Log : Uygulama ve sistem mesajlarını veya hataları saklamak için tutulan kayıtlardır.

Port : İşletim sistemi üzerinde çalışan bir uygulamanın, dış dünyaya erişimini sağlama veya dışarıdan gelen isteklere cevap verebilmesi için kullandığı bağlantı noktasıdır.

Shell : İşletim sisteme, komutlar ile müdahale edilmesine izin veren komut satırı arayüzüdür.

Shell Script : Otomatik olarak çalışması için ayarlanan komutları barındıran komut dosyasıdır.

SuperUser: Root kullanıcısı ile aynı anlama gelir.

Swap : Verileri geçici olarak RAM'dan (Random Access Memory) disk'e veya tersi yönde taşımak için kullanılan sanal bir hafızadır. Takas dosyası olarak da adlandırılır.

Syslog : Bütün sistem mesaj ve hatalarının depolandığı alandır.

Shell Komut Çeşitleri - Alias, Internal & External Commands, which, type, \$PATH

Shell'in amacı, komutların yürütülebileceği bir ortam sağlamasıdır. Shell, bir kullanıcının girdiği komutu doğru şekilde yorumlamakla ilgilenir. Bunu yapmak için, üç farklı komut çeşidini kullanır.

- Alias
- Internal Komutlar
- External Komutlar

Alias (takma ad), kullanıcı tarafından tanımlanan komutlardır. ls -lah komutu gizli dosyalar dahil, dizin altındaki dosyaların ayrıntısını gösterir. ll = 'ls –lah' ile ll kullanarak komuta alias ataması yapabiliriz.

Internal komutlar, shell'in parçası olan komutlardır. Bir komutun çeşidini görmek için **type** komutu kullanılır.

```
ozgur@ubuntu:~$ type time
time is a shell keyword
ozgur@ubuntu:~$ type ls
ls is aliased to `ls --color=auto'
ozgur@ubuntu:~$ type cp
cp is /usr/bin/cp
```

External komutları aramak için **\$PATH** değişkenini kullanırız. Bu değişken, bir kullanıcı komutu girdiği zaman girilen komutun varsayılan olarak nerede çalıştırılacağını tanımlar. Komut satırında cp, date vs komutları girdiğinizde direk çalışır. Bunun nedeni PATH'de bu dosya dizini tanımlanmıştır. Eğer .profile dosyasındaki kayıtlı PATH'in satır sonunda bulunan bin'den sonra :/tmp yazarsak bu dizin içinde çalıştırılabilir bir şey varsa /tmp dizinine gitmeden herhangi bir yerde çalıştırabiliriz.

```
ozgur@ubuntu:~$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
```

Shell'in komutu çalıştırmak için kullanacağı dizini görmek için **which** kullanılır.

```
ozgur@ubuntu:~$ which fdisk
/usr/sbin/fdisk
```

Geçerli bir dizine sahip olmayan komutlar için komut başına ./ işareteti getirilir.

Linux'ta Kabuk Başlatma Dosyalarını ve Kullanıcı Profillerini Anlama

<https://www.tecmint.com/set-unset-environment-variables-in-linux/>

Linux birden fazla kullanıcının sistem üzerindeki kaynaklara erişmesine izin verecek şekilde tasarlanmıştır. Linux'da kullanıcının eriştiği ortam, global ve personal olarak iki şekilde oluşturulur. Normalde, bir Linux sistemiyle çalışmanın temel yöntemi shell'dir. Bir kullanıcı oturum açtıktan sonra başlatma sırasında okuduğu belirli dosyalara bağlı olarak bir ortam oluşturur.

Shell başlatıldığı zaman, sistem kullanıcıı için önceden tanımlanmış dosyalar okunarak bir ortam oluşturulur. /etc dizini altındaki bulunan /etc/profile ve /etc/bashrc dizinleri sistem üzerindeki tüm kullanıcılar uygulanan global yapılandırma dosyalarını içerir.

Belirli bir kullanıcı ortamı ise kullanıcının home dizininde bulunan kullanıcıya ait dizindeki .profile, .bashrc dosyaları okunarak oluşturulur.

Interactive ve Non-Interactive Shell nedir?

Kullanıcı sisteme giriş yapmaya çalıştığında /etc/passwd içinde bulunan kimlik bilgileri doğrulanır ve sonrasında /etc/profile veya ~/.profile dosyası okunarak başlatılan shell'e interactive shell denir. Non-Interactive shell ise kullanıcı girişü gerekmeksiz bir script çalıştığı zaman çağrılır.

Global Shell Başlangıç Dosyaları

Global ortam değişkenlerinin yapılandırıldığı dosya **/etc/profile** dosyasıdır.

Global ortamda değişiklikler yapmak için kullanılan script dosyaları **/etc/profile.d/** dizini altındadır.

```
ozgur@test:/etc$ ls -lah | grep profile
-rw-r--r-- 1 root root      581 Dec  5  2019 profile
drwxr-xr-x 2 root root     4.0K Mar  8 15:15 profile.d
```

Tüm sistem kullanıcıları için geçerli olan diğer yapılandırmaları **/etc/bash.bashrc** dosyasını içindedir.

Kullanıcı Shell Başlangıç Dosyaları

Home dizini bir kullanıcının dosya, dizin ve programlarının saklandığı alandır. Sunucuya login olduktan sonra kullanıcının ilk bulunduğu dizin home dizinidir. Bütün sistem dosyalarının bulunduğu root dizinine geçmek için / slash işaretini kullanılır.

Kullanıcı oturumunu açtığında, kullanıcının ayarları ve çalışma ortamı, kullanıcının home dizininde bulunan başlangıç dosyaları tarafından belirlenir.

Başlangıç Dosyaları

```
ozgur@client:~$ pwd
/home/ozgur
ozgur@client:~$ ls -lah
total 32K
drwxr-xr-x 4 ozgur ozgur 4.0K Feb 24 10:36 .
drwxr-xr-x 3 root  root  4.0K Feb 14 13:33 ..
-rw----- 1 ozgur ozgur  954 Feb 23 21:56 .bash_history
-rw-r--r-- 1 ozgur ozgur 220 Feb 25 2020 .bash_logout
-rw-r--r-- 1 ozgur ozgur 3.7K Feb 25 2020 .bashrc
drwx----- 2 ozgur ozgur 4.0K Feb 14 13:33 .cache
-rw-r--r-- 1 ozgur ozgur  807 Feb 25 2020 .profile
drwx----- 2 ozgur ozgur 4.0K Feb 20 13:15 .ssh
-rw-r--r-- 1 ozgur ozgur    0 Feb 14 13:33 .sudo_as_admin_successful
```

.bash_history dosyası oturumunuz açıkken girdiğiniz komutlar, oturumunuzu kapatırken bu dosyaya yazılır. Bu şekilde girdiğiniz komutlar hafızada tutulur. Önceden yaptığınız işlemleri veya kullanıp hatırlamadığınız komutlara bakmak için çok faydalıdır. Oturumunuz, uygun olmayan bir şekilde kapanırsa komutlar kaydedilmez.

.bash_logout dosyası sunucudan çıkışken geçici dosyaları sil, login süresini kaydet gibi yapılmasını istediğiniz komutlar yerleştirilerek çalıştırılır. Sunucudan çıkışken bu dosya otomatik olarak çalıştırılır. En sık kullanılan komut clear 'dır.

.bashrc dosyası, sunucuda açık bir oturumunuz varsa ve yeni bir terminal ekranı açarak sunucuya ikinci bir bağlantı yaptığımda çalıştırılır. Her oturum açtığınızda çalışarak, oturumunuz için yapılan ayarları getirir.

.profile dosyası, sunucuya console veya uzak bağlantı yapılarak giriş sağlandığında çalıştırılır. Kullanıcı için tanımlanan ortam değişkenlerini ve yapılandırmalarını tutar.

```
Last login: Wed Feb 24 07:40:08 2021 from 192.168.1.37
Merhaba bashrc
Merhaba Linux
ozgur@client:~$ sudo su -
[sudo] password for ozgur:
root@client:~# su ozgur
Merhaba bashrc
ozgur@client:/root$ /bin/bash
Merhaba bashrc
```

.ssh oturumunuzda kullandığınız SSH anahtarlarının bulunduğu dizindir.

.cache dosyası kullanıcıların ön bellekteki dosyalarını saklar.

.bashrc veya .profile dosyalarına alias eklemek için .bashrc veya .profile dosyalarının en alt satırına alias'ları yerleştirebiliriz. Netstat –an komutu için aşağıdaki gibi bir alias girelim.

Alias yakala='netstat -an'

/bin/bash/ komutu ile .bashrc'yi yeniden çalıştırıyoruz.

```
ozgur@client:~$ /bin/bash
Merhaba bashrc
ozgur@client:~$ yakala
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address          Foreign Address        State
tcp      0      0 127.0.0.53:53            0.0.0.0:*
tcp      0      0 0.0.0.0:22              0.0.0.0:*
tcp      0      64 192.168.1.39:22         192.168.1.37:54722 ESTABLISHED
```

man Kullanımı

man, maual kelimesinden gelen, bir komutun nasıl kullanılacağını anlatan kılavuzdur. Komut ile birlikte kullanılacak değişkenleri, değişkenlerin göstereceği sonuçları ve komut örneklerini man sayfalarında bulabiliriz.

man sayfalarından yararlanmak için man ve ardından bilgi almak istediğimiz komutu yazmamız yeterlidir.

man cp

CP(1) User Commands

NAME

cp - copy files and directories

SYNOPSIS

cp [OPTION]... [-T] SOURCE DEST
cp [OPTION]... SOURCE... DIRECTORY
cp [OPTION]... -t DIRECTORY SOURCE...

DESCRIPTION

Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.

Mandatory arguments to long options are mandatory for short options too.

-a, --archive
same as -dR --preserve=all

--attributes-only
don't copy the file data, just the attributes

--backup[=CONTROL]
make a backup of each existing destination file

Anahtar bir kelime üzerinden bir man sayfasına ulaşmak isterseniz -k (keyword) değişkenini kullanmanız gereklidir. Anahtar kelimenizin partition olduğunu varsayıyalım.

man –k partition komutuyla partition kelimesinin geçtiği man sayfaları görüntülenir. Komutların sonundaki (1) ve (8) sayfaların kategori numaralarıdır. (1) Çalıştırılabilir program veya komut satırı anlamına gelirken, (8) sistem yöneticileri tarafından kullanılan komutlardır.

```
ozgur@ubuntu:~$ man -k partition
addpart (8)           - tell the kernel about the existence of a partition
cfdisk (8)            - display or manipulate a disk partition table
cgdisk (8)            - Curses-based GUID partition table (GPT) manipulator
delpart (8)           - tell the kernel to forget about a partition
fdisk (8)             - manipulate disk partition table
fixparts (8)          - MBR partition table repair utility
gdisk (8)             - Interactive GUID partition table (GPT) manipulator
growpart (1)          - extend a partition in a partition table to fill available space
kpartx (8)            - Create device maps from partition tables.
parted (8)            - a partition manipulation program
partprobe (8)          - inform the OS of partition table changes
partx (8)              - tell the kernel about the presence and numbering of on-disk partitions
resizepart (8)         - tell the kernel about the new size of a partition
sfdisk (8)             - display or manipulate a disk partition table
sgdisk (8)             - Command-line GUID partition table (GPT) manipulator for Linux and Unix
systemd-gpt-auto-generator (8) - Generator for automatically discovering and mounting root, /home,
```

Eğer kategoriler içinden bir arama yapmak istersek;

man -k password | grep 8 komutu ile sadece sistem yöneticileri tarafından kullanılan sayfalar içinde arama yapmış oluruz.

```
ozgur@ubuntu:~$ man -k password | grep 8
chpasswd (8)          - update group passwords in batch mode
chpasswd (8)          - update passwords in batch mode
cpgr (8)              - copy with locking the given file to the password or group file
cppw (8)              - copy with locking the given file to the password or group file
grpconv (8)           - convert to and from shadow passwords and groups
grpunconv (8)         - convert to and from shadow passwords and groups
pam_pwhistory (8)    - PAM module to remember last passwords
pam_unix (8)          - Module for traditional password authentication
pwck (8)              - verify integrity of password files
pwconv (8)             - convert to and from shadow passwords and groups
```

man sayfaları bir veritabanında tutulur. Veritabanını güncellemek için herhangi bir argüman kullanmadan sadece komut satırına **mandb** yazmamız yeterlidir.

info kullanımı

man sayfaları haricinde info ile de destek alabileceğiniz sayfalar bulunmaktadır. Bazı komutlar için man sayfaları yeterli olmadığındaysa info sayfalarına da bakmak size yardımcı olabilir. fdisk komutıyla ilgili bir bilgi almak istersek komutumuz aşağıdaki gibi olacaktır. Sayfa çıktısı man sayfaları gibi olacaktır. Çıktıda sağ ve sol üst tarafta (8) rakamını göreceksiniz. Bunun anlamı sistem yöneticileri için kullanılan bir komut olduğunu.

info fdisk

```
EDISK(8)                               System Administration      FDISK(8)

NAME
    fdisk - manipulate disk partition table

SYNOPSIS
    fdisk [options] device
    fdisk -l [device...]

DESCRIPTION
    fdisk is a dialog-driven program for creation and manipulation of partition tables. It
    understands GPT, MBR, Sun, SGI and BSD partition tables.

    Block devices can be divided into one or more logical disks called partitions. This division
    is recorded in the partition table, usually found in sector 0 of the disk. (In the
    BSD world one talks about 'disk slices' and a 'disklabel').
```

help kullanımı

man ve info sayfaları bazı durumlarda hem uzun hem de karmaşık gelebilir. Komut hakkında hızlı bir şekilde yardım isterseniz help kullanabilirsiniz.

```
ozgur@ubuntu:~$ cp --help
Usage: cp [OPTION]... [-T] SOURCE DEST
       or: cp [OPTION]... SOURCE... DIRECTORY
       or: cp [OPTION]... -t DIRECTORY SOURCE...
Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.

Mandatory arguments to long options are mandatory for short options too.
  -a, --archive                  same as -dR --preserve=all
  --attributes-only              don't copy the file data, just the attributes
  --backup[=CONTROL]             make a backup of each existing destination file
  -b                            like --backup but does not accept an argument
  --copy-contents                copy contents of special files when recursive
  -d                            same as --no-dereference --preserve=links
```

tab kullanımı

tab tuşu hem komutları hem de dizinlerin tamamını yazmadan otomatik olarak tamamlamasını sağlar. Örneğin pass yazıp tab tuşuna basın ardından kullanıcı adınızın ilk harfini yazıp tab tuşuna tekrar bastığınızda hem komut hem de kullanıcı adınız otomatik olarak tamamlanarak passwd kullanıcı_adınız şeklinde bir komut oluşacaktır.

Baş bir örneğimizde SSH yapılandırma dosyamıza gidelim. nano / işaretini koyduktan sonra e yaz + tab, ssh yaz + tab ve sshd yaz + tab ile hızlı bir şekilde yapılandırma dosyasına gidebiliriz.

Yine cd (change directory) ile bulunduğunuz dizinde iki defa tab tuşuna arka arkaya basarsanız bulunduğunuz dizinin alt dizinlerini görebilirsiniz.

```
ozgur@ubuntu:~$ cd /etc/
alternatives/          iproute2/           rc1.d/
apparmor/              iscsi/             rc2.d/
apparmor.d/            kernel/            rc3.d/
apport/                landscape/        rc4.d/
apt/                  ldap/              rc5.d/
```

whatis / whereis

whatis Bir komutun tanım çıktısını verir. whereis ise komutun yolunu gösterir.

```
ozgur@ubuntu:~$ whatis route
tc-route (8)          - route traffic control filter
ozgur@ubuntu:~$ whatis ufw
ufw (8)               - program for managing a netfilter firewall
ozgur@ubuntu:~$ whereis ufw
ufw: /usr/sbin/ufw /usr/lib/ufw /etc/ufw /usr/share/ufw /usr/share/man/man8/ufw.8.gz
ozgur@ubuntu:~$ whereis route
route:
```

Temel Komutlar – pwd, cd, ls, history, clear

pwd (Print Working Directory) : Bulundığınız dizini gösterir.

```
ozgur@ubuntu:~$ pwd
/home/ozgur
```

cd (Change Directory) : Dizinler arasında gezinmek için cd komutu kullanılır. Hiyerarşik yapılandırmaya göre bir gezinti gerçekleştirebilirsiniz. Örneğin cd /etc/ssh ile ulaştığınız dizinin altında başka bir dizin altında bulunan dizine erişemezsiz. Bunu yapmak için root dizininde başlayarak dizin değiştirme işlemeye gitmeniz gereklidir.

root@ubuntu:/etc/ssh# Ekran çıktısında /etc/ss h dizini altında olduğumuz görülüyor. Bu dizin altında direkt olarak /usr/share dizini altında bulunan bir dizine gidemeyez. Bunu yapmak için root dan başlayıp gideceğimiz dizinin yolunu göstermemiz gereklidir.

```
root@ubuntu:/etc/ssh# cd /usr/share/
```

Komut satırına tek başına **cd** yazıp enter'a bastığınızda login olduğunuz kullanıcının /home dizinine dönersiniz.

```
ozgur@ubuntu:/etc$ cd /etc/ssh
ozgur@ubuntu:/etc/ssh$ cd
ozgur@ubuntu:~$ pwd
/home/ozgur
```

cd .. nokta ile bir önceki dizine geri dönersiniz.

```
ozgur@ubuntu:~$ cd /etc/ssh
ozgur@ubuntu:/etc/ssh$ cd ..
ozgur@ubuntu:/etc$
```

cd – ile çıkışmış olduğunuz bir önceki dizine geri dönersiniz.

```
ozgur@ubuntu:/etc/ssh$ cd
ozgur@ubuntu:~$ pwd
/home/ozgur
ozgur@ubuntu:~$ cd -
/etc/ssh
ozgur@ubuntu:/etc/ssh$
```

/ işaretini dizinlerin arasındaki hiyerarşî için kullanılır. / işaretini kullanmadan bir dizine gidemezsiniz.

```
ozgur@ubuntu:~$ cd etc  
-bash: cd: etc: No such file or directory
```

ls (list) : Dizinlerin içeriklerini listeler. ls komutu ile birlikte kullanılacak değişkenler listenin çıktısını daha ayrıntılı olarak gösterecektir.

ls -a gizli dosyalar dahil tüm dosyaları gösterirken, -l ile ayrıntılı bir liste görürsünüz. -h dosya boyutunun anlaşılabilir şekilde bir çıktısını verir.

Komut satırına ls -lah yazıp enter'a basarsak bulunduğuuz dizin altındaki dosyaların izinlerinin durumu, sahipleri, boyutu, oluşturulma zamanları ve isimlerini görürüz. Mavi renkli olan test isimli dosya haricindeki dosyalar gizli dosyadır.

```
ozgur@ubuntu:~$ ls -lah  
total 32K  
drwxr-xr-x 4 ozgur ozgur 4.0K Feb  3 10:57 .  
drwxr-xr-x 3 root  root  4.0K Feb  2 22:22 ..  
-rw----- 1 ozgur ozgur   18 Feb  2 23:03 .bash_history  
-rw-r--r-- 1 ozgur ozgur  220 Feb 25 2020 .bash_logout  
-rw-r--r-- 1 ozgur ozgur  3.7K Feb 25 2020 .bashrc  
drwx----- 2 ozgur ozgur 4.0K Feb  2 23:03 .cache  
-rw-r--r-- 1 ozgur ozgur  807 Feb 25 2020 .profile  
-rw-r--r-- 1 ozgur ozgur    0 Feb  2 23:03 .sudo_as_admin_successful  
drwxrwxr-x 2 ozgur ozgur 4.0K Feb  3 10:57 test
```

history : Komut satırında girdiğimiz komutları saklar. Komut satırına history yazdığında login olduğunuz kullanıcı ile girdiğiniz tüm komutları görebilirsiniz. History daha önceden yaptığınız bir işlemi hatırlamadığınızda neler yaptığınız konusunda size faydalı bilgiler sunabilir. History çıktısındaki komut başında bulunan sayının başına ! koyup yazdığında komutu gerçekleştirmiş olursunuz. Örneğin çıktıda 1. sırada nano /etc/ssh/sshd_config komutunu yazmak yerine komut satırına !1 yazabilirisiniz. Diğer bir yöntem, mouse ile bir seçim yaptıktan sonra bu seçim hafızaya alınır. Komut satırına mouse'un sağ tuşuna tıklayarak hafızaya aldığı komutu yapıştırabilirsiniz.

```
ozgur@ubuntu:~$ history  
1  nano /etc/ssh/sshd_config  
2  cd /usr/bin/  
3  cd  
4  mkdir test  
5  rm -rf test  
6  pwd  
7  history
```

History içerisinde çok fazla kayıt olabilir. Bu kayıtların içerisinde aramak için shell ekranında Ctrl+R 'ye basıp aramak istediğiniz komutu yazabilirisiniz. HISTSIZE değişkeni ile history de tutulacak boyutu ayarlayabilirsiniz.

```
ozgur@ubuntu:~$ echo $HISTSIZE  
1000  
ozgur@ubuntu:~$ HISTSIZE=2000  
ozgur@ubuntu:~$ echo $HISTSIZE  
2000
```

clear : Açık olan oturumdaki ekranı temizlemek için kullanılır. Ekrانınızda çok fazla işlem yaptınız ve artık bu rahatsızlık verecek seviyeye geldiyse clear ile temiz bir komut satırına döndürülürsünüz.

Temel Komutlar - touch, cat, cp, mkdir, rm, rmdir, mv, rename

touch : Boş bir dosya yaratmanın en basit yoludur.

touch testfile

touch testdosyasi.txt

touch test{1..5} komutu adı test ile başlayan 1'den 5'e kadar sıralanan 5 adet boş dosya yaratır.

```
ozgur@ubuntu:~$ ls
ozgur@ubuntu:~$ touch testfile
ozgur@ubuntu:~$ ls
testfile
ozgur@ubuntu:~$ touch testdosyasi.txt
ozgur@ubuntu:~$ ls
testdosyasi.txt  testfile
ozgur@ubuntu:~$ touch test{1..5}
ozgur@ubuntu:~$ ls
test1  test2  test3  test4  test5  testdosyasi.txt  testfile
```

cat : Dosya içeriğini komut satırında read-only olarak görüntüler.

```
ozgur@ubuntu:~$ echo MERHABA DÜNYA > test1
ozgur@ubuntu:~$ cat test1
MERHABA DÜNYA
ozgur@ubuntu:~$
```

cp (Copy) : Bir dosyayının kopyasını oluşturmak için kullanılır. Bir konfigürasyon dosyasında değişiklik yapmadan öncebu komut ile mutlaka bir kopyasını oluşturmanızı tavsiye ederim. Herhangi bir şeyin ters gitmesi durumunda yedek dosyanızdan düzeltme imkanınız olur.

cp test test.backup

```
ozgur@ubuntu:~$ ls
test
ozgur@ubuntu:~$ cp test test.backup
ozgur@ubuntu:~$ ls
test  test.backup
```

Dosyanızı başka bir dizine kopyalamak isterseniz dizin yolunu göstermeniz gereklidir. Dosyanızın bir kopyası belirttiğiniz dizine kopyalanır.

cp test backupfiles

```
ozgur@ubuntu:~$ mkdir backupfiles
ozgur@ubuntu:~$ cp test backupfiles
ozgur@ubuntu:~$ ls
backupfiles  test  test.backup
ozgur@ubuntu:~$ ls backupfiles/
test
```

Bir dizin içindeki bütün dosyaların ve dizinlerin başka bir dizine kopyasını almak isterseniz **-r** argümanını kullanmalısınız.

cp -r backupfiles/ newbackupfiles

```
ozgur@ubuntu:~$ ls
backupfiles test test.backup
ozgur@ubuntu:~$ cp -r backupfiles/ newbackupfiles
ozgur@ubuntu:~$ ls
backupfiles newbackupfiles test test.backup
```

Sadece dizin içerisindeki dosyaları bir klasöre kopyalamak isterseniz kopyalamak istediğiniz dosyaları yan yana yazabilirsiniz.

cp test test.backup files

```
ozgur@ubuntu:~$ ls
backupfiles newbackupfiles test test.backup
ozgur@ubuntu:~$ mkdir files
ozgur@ubuntu:~$ cp test test.backup files
ozgur@ubuntu:~$ ls files/
test test.backup
```

mkdir : Linux'da dizin yaratmak için kullanılır.

mkdir dosyalarım

```
ozgur@ubuntu:~$ mkdir dosyalarim
ozgur@ubuntu:~$ ls
dosyalarim
```

Eğer alt alta parent dizinler oluşturmak isterseniz -p argümanını kullanmamız gereklidir.

mkdir -p dosyalarim/dosyalar/dosya

```
ozgur@ubuntu:~$ rm -rf *
ozgur@ubuntu:~$ mkdir -p dosyalarim/dosyalar/dosya
ozgur@ubuntu:~$ ls
dosyalarim
ozgur@ubuntu:~$ cd dosyalarim/
ozgur@ubuntu:~/dosyalarim$ ls
dosyalar
ozgur@ubuntu:~/dosyalarim$ cd dosyalar/
ozgur@ubuntu:~/dosyalarim/dosyalar$ ls
dosya
```

rm (Remove) : Bir dosyayı silmek için kullanılır. Dikkatli kullanılması gereklidir. **rm** ile sildiğiniz dosyaları bir daha geri döndüremezsiniz.

rm test1

```
ozgur@ubuntu:~$ touch test{1..3}
ozgur@ubuntu:~$ ls
test1 test2 test3
ozgur@ubuntu:~$ rm test1
ozgur@ubuntu:~$ ls
test2 test3
```

rm -i komutuyla dosyanın silinip silinmeyeceğini soran bildirim alınmasını sağlayabilirsiniz.

```
ozgur@ubuntu:~$ rm -i test2
rm: remove regular empty file 'test2'? yes
ozgur@ubuntu:~$ ls
test3
ozgur@ubuntu:~$ rm -i test3
rm: remove regular empty file 'test3'? n
ozgur@ubuntu:~$ ls
test3
```

rm -rf komutu ile dosya ve dizinler dahil bütün her şeyi silersiniz. -r recursive ve -f force anlamındadır.

```
ozgur@ubuntu:~$ rm test
rm: cannot remove 'test': Is a directory
ozgur@ubuntu:~$ rm -rf test
ozgur@ubuntu:~$ ls
test3
```

rmdir (Remove Directory) : Bir dizini silmek isterseniz rmdir komutunu kullanabilirsiniz. Yine -p argümanı ile alt dizinler dahil bir silme işlemi de gerçekleştirebilirsiniz.

```
ozgur@ubuntu:~$ mkdir -p dosyalarim/dosyalar
ozgur@ubuntu:~$ ls
dosyalarim  test3
ozgur@ubuntu:~$ cd dosyalarim/
ozgur@ubuntu:~/dosyalarim$ ls
dosyalar
ozgur@ubuntu:~/dosyalarim$ cd
ozgur@ubuntu:~$ rmdir dosyalarim/dosyalar
ozgur@ubuntu:~$ ls
dosyalarim  test3
ozgur@ubuntu:~$ cd dosyalarim/
ozgur@ubuntu:~/dosyalarim$ ls
ozgur@ubuntu:~/dosyalarim$
```

mv (Move) : Bir dosya veya dizini başka bir konuma taşımak için kullanılır. Windows'daki Kes komutunun aynısıdır. mv ile taşıdığınız dosyaya bir daha eski konumundan ulaşamazsınız.

mv test.txt dosyalarim

```
ozgur@ubuntu:~$ mkdir dosyalarim
ozgur@ubuntu:~$ touch test.txt
ozgur@ubuntu:~$ ls
dosyalarim  test.txt
ozgur@ubuntu:~$ mv test.txt dosyalarim/
ozgur@ubuntu:~$ ls
dosyalarim
```

mv komutunu isim değiştirmek içinde kullanabiliriz.

mv test.txt deneme.txt

```
ozgur@ubuntu:~$ touch test.txt
ozgur@ubuntu:~$ ls
test.txt
ozgur@ubuntu:~$ mv test.txt deneme.txt
ozgur@ubuntu:~$ ls
deneme.txt
```

Temel Komutlar - file, time, uptime, cal

file : Dosyaların tiplerini tanımlamak için kullanılır. -s argümanı ile özel dosyalar hakkında bilgi verir.

```
ozgur@ubuntu:~$ file test1.txt
test1.txt: empty
ozgur@ubuntu:~$ file test2.txt
test2.txt: UTF-8 Unicode text
ozgur@ubuntu:~$ file /dev/sda
/dev/sda: block special (8/0)
ozgur@ubuntu:~$ file -s /dev/sda
/dev/sda: DOS/MBR boot sector, extended partition table (last)
```

time : Bir komutun çalışma süresinin ne kadar sürecekğini gösterir.

```
ozgur@ubuntu:~$ time tar -cf test.tar test
real    0m0.004s
user    0m0.003s
sys     0m0.000s
```

uptime : Sunucunun açıldığı zamanı, ne kadar süredir açık olduğunu, login olan kullanıcı sayısını gösterir.

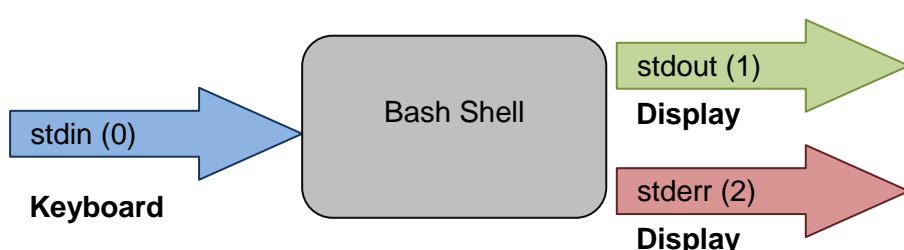
```
ozgur@ubuntu:~$ uptime
21:09:24 up 2:12, 2 users, load average: 0.00, 0.00, 0.00
```

cal : Takvimi gösterir.

```
ozgur@ubuntu:~$ cal
February 2021
Su Mo Tu We Th Fr Sa
      1  2  3  4  5  6
  7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28
```

I/O Redirection - stdin, stdout, stderr

Komut satırından yapılan girişlerin ve ekrandaki çıktıların yönlendirme işlemidir. Temel olarak stdin(0), stdout(1) ve stderr(2) olarak 3 yön vardır.



stdout > : işaretin bir ekran çıktısını bir dosyaya yönlendirmek için kullanılır.

```
ozgur@ubuntu:~$ echo Merhaba Dünya  
Merhaba Dünya  
ozgur@ubuntu:~$ echo Merhaba Dünya > test.txt  
ozgur@ubuntu:~$ ls  
test.txt  
ozgur@ubuntu:~$ cat test.txt  
Merhaba Dünya
```

> işaretini dosyanın içindelerinin üzerine yazar. Eğer dosya içerisinde daha önceden bir veri varsa bu bilgiler silinir ve yönlendirdiğiniz bilgiler yazılır.

```
ozgur@ubuntu:~$ echo Merhaba Dünya  
Merhaba Dünya  
ozgur@ubuntu:~$ echo Merhaba Dünya > test.txt  
ozgur@ubuntu:~$ ls  
test.txt  
ozgur@ubuntu:~$ cat test.txt  
Merhaba Dünya  
ozgur@ubuntu:~$ echo Linux Eğitimi > test.txt  
ozgur@ubuntu:~$ cat test.txt  
Linux Eğitimi
```

Yönlendirme işaretini kullanırken yanlış bir komut yazarsanız dosyanın içi silinir.

```
ozgur@ubuntu:~$ cho Linux Eğitimi > test.txt  
Command 'cho' not found, did you mean:  
  
  command 'echo' from deb coreutils (8.30-3ubuntu2)  
  command 'who' from deb coreutils (8.30-3ubuntu2)  
  
Try: sudo apt install <deb name>  
  
ozgur@ubuntu:~$ cat test.txt  
ozgur@ubuntu:~$
```

set -o noclobber komutunu kullanırsanız yanlışlıkla dosyaların üzerine yazmanın önüne geçebilirsiniz.

```
ozgur@ubuntu:~$ cat test.txt  
Merhaba Dünya  
ozgur@ubuntu:~$ set -o noclobber  
ozgur@ubuntu:~$ echo Linux Eğitimi > test.txt  
-bash: test.txt: cannot overwrite existing file
```

noclobber ayarlamانıza rağmen dosyanın üzerine yazmak isterseniz > işaretinden sonra | kullanarak içeriği değiştirebilirsiniz.

```
ozgur@ubuntu:~$ cat test.txt  
Merhaba Dünya  
ozgur@ubuntu:~$ set -o noclobber  
ozgur@ubuntu:~$ echo Linux Eğitimi > test.txt  
-bash: test.txt: cannot overwrite existing file  
ozgur@ubuntu:~$ echo Linux Eğitimi >| test.txt  
ozgur@ubuntu:~$ cat test.txt  
Linux Eğitimi
```

Dosya içeriğini değiştirmeden yeni veriyi bir alt satırda eklemek istersek >> işaretini kullanmalıyız.

```
ozgur@ubuntu:~$ echo Merhaba Dünya >> test.txt
ozgur@ubuntu:~$ cat test.txt
Linux Eğitimi
Merhaba Dünya
```

Stdout log çıktılarını veya sistem ile ilgili özet verileri bir dosyaya yazdırınmak için kullanılabilir.

Stderr 2> : Hata mesajlarını ekrana bastırmadan dosya içeriğine yazdırınmak için kullanılır. Aşağıdaki çıktıda **ls -lh** komutu ile aldığımız çıktıyı **test.txt** dosyasının içerisinde yazdırıldı. Ardından yanlış bir komut girerek ekrana hata dönmescini sağladık. Devamında aynı hatalı komutu **2>** olacak şekilde yapıp çalışırdık. Ekrana bir hata dönmemi ancak hata çıktısı dosya içerisinde yazıldı.

```
ozgur@ubuntu:~$ ls -lh > test.txt
ozgur@ubuntu:~$ cat test.txt
total 0
-rw-rw-r-- 1 ozgur ozgur 0 Feb  3 22:04 test.txt
ozgur@ubuntu:~$ ls lh > test.txt
ls: cannot access 'lh': No such file or directory
ozgur@ubuntu:~$ ls lh 2>> test.txt
ozgur@ubuntu:~$ cat test.txt
ls: cannot access 'lh': No such file or directory
ozgur@ubuntu:~$
```

Yukarıdaki örnekte dosya içerisindeki veriyi kaybettik. **2>** simgesini **2>>** olarak değiştirip yazdığınıza dosya içeriği olduğu gibi kaldı ancak alt satırda hata mesajı da eklendi.

```
ozgur@ubuntu:~$ ls -lh > test.txt
ozgur@ubuntu:~$ cat test.txt
total 0
-rw-rw-r-- 1 ozgur ozgur 0 Feb  3 22:09 test.txt
ozgur@ubuntu:~$ ls lh 2>> test.txt
ozgur@ubuntu:~$ cat test.txt
total 0
-rw-rw-r-- 1 ozgur ozgur 0 Feb  3 22:09 test.txt
ls: cannot access 'lh': No such file or directory
```

Stdin < : Temel olarak **cat** ile aynı anlama gelir. Dosya içerisinde bulunan verileri ekrana basar. Aşağıdaki ekran çıktısında linuxeğitimi base64 ile kodlayarak ekrana bastırdık. Ardından decode ederek yeniden görüntüledik.

```
ozgur@ubuntu:~$ base64 <<< linuxeğitimi
bGludXhlxJ9pdGltaQo=
ozgur@ubuntu:~$ base64 -d <<< bGludXhlxJ9pdGltaQo=
linuxeğitimi
```

Operatörler - ‘’, “”, ;, &, &&, |, ||, #, \, ~

Burada gösterdiğimizden çok daha fazla operatör vardır. Bash Script yazımında sıkılıkla kullanılırlar. Genel olarak kullanılanlar hakkında bilgi sahibi olmak faydalı olacaktır.

“ ve “ ”

Komut satırında yapacağınız girişlerin arasındaki boşlukların kaldırılmasını engeller, ayrık isimli dosya ve dizin adları oluşturmanıza veya bu dosya ve dizinleri kullanmanızı sağlar.

```
ozgur@ubuntu:~$ mkdir 'benim dosyalarim'  
ozgur@ubuntu:~$ mkdir "senin dosyalarin"  
ozgur@ubuntu:~$ ls  
'benim dosyalarim'  'senin dosyalarin'  
ozgur@ubuntu:~$ echo 'Merhaba      Dünya' > "benim kayitlarim"  
ozgur@ubuntu:~$ ls  
'benim dosyalarim'  'benim kayitlarim'  'senin dosyalarin'  
ozgur@ubuntu:~$ cat "benim kayitlarim"  
Merhaba      Dünya
```

; noktalı virgül

İki ayrı komut serisini sırayla çalıştırır. Komutların başarılı olarak çalışıp çalışmaması bir diğer komutu etkilemez.

```
ozgur@ubuntu:~$ echo Merhaba ; ls -lh  
Merhaba  
total 12K  
drwxrwxr-x 2 ozgur ozgur 4.0K Feb  4 08:39 'benim dosyalarim'  
-rw-rw-r-- 1 ozgur ozgur  116 Feb  4 08:57 'benim kayitlarim'  
drwxrwxr-x 2 ozgur ozgur 4.0K Feb  4 08:39 'senin dosyalarin'
```

& Ampersand

Komutun sonuna koyularak komutun arka planda çalışması sağlanır. Aşağıdaki ekran çıktısında bir alanı klonlamak için bir komut başlattık ancak ön yüzde çalışmaya başladı. Ctrl+C ile komutu sonlandırdık. Aynı komutu bu sefer sonuna & işaretini koyarak çalıştırıldı. Bu sefer arka planda çalışmaya başladı ve biz 1280 process ID'si ile çalıştığını bildirdi. Fg komutu ile komutun ön planda çalışmasını sağlayıp yine Ctrl+C ile işlemi sonlandırdık.

```
ozgur@ubuntu:~$ dd if=/dev/zero of=/dev/null  
^C19648963+0 records in  
19648962+0 records out  
10060268544 bytes (10 GB, 9.4 GiB) copied, 14.2322 s, 707 MB/s  
  
ozgur@ubuntu:~$ dd if=/dev/zero of=/dev/null &  
[1] 1280  
ozgur@ubuntu:~$ fg  
dd if=/dev/zero of=/dev/null  
^C9649216+0 records in  
9649215+0 records out  
4940398080 bytes (4.9 GB, 4.6 GiB) copied, 6.87988 s, 718 MB/s
```

Noktalı virgül ile aynı şekilde de kullanılır.

&& Çift Ampersand

Ve anlamında kullanılır. Arka arkaya yazılan komutlardan kendinden önce gelen başarılı olursa çalıştırılır.

```
ozgur@ubuntu:~$ cd test && ls -lh /tmp/  
-bash: cd: test: No such file or directory
```

| pipe

Çıktı veren komutlarla kullanıldığından çok faydalıdır. Çıktıların içinde istediğiniz bir veriyi yakalamak, çıktı boyutu ayarlamak için komutlardan önce kullanılır.

```
ozgur@ubuntu:~$ tail -f /var/log/auth.log | grep Failed
Feb  4 09:40:44 ubuntu sshd[1360]: Failed password for invalid user fadf from 192.168.1.38 port 54845 ssh2
Feb  4 09:41:17 ubuntu sshd[1360]: Failed password for invalid user fadf from 192.168.1.38 port 54845 ssh2
Feb  4 09:41:21 ubuntu sshd[1360]: Failed password for invalid user fadf from 192.168.1.38 port 54845 ssh2
```

|| Çift Pipe

Veya anlamında kullanılır. Arka arkaya yazılan komutlardan birisi başarısız olursa diğerinin çalışması sağlar. Tek başına kullanımı pek anlamı olmama da diğer operatörlerle kullanılıncaya faydalıdır.

```
ozgur@ubuntu:~$ cd test || cd 'benim dosyalarim' && echo Merhaba > deneme.txt && pwd && ls -lah
-bash: cd: test: No such file or directory
/home/ozgur/benim dosyalarim
total 12K
drwxrwxr-x 2 ozgur ozgur 4.0K Feb  4 09:30 .
drwxr-xr-x 6 ozgur ozgur 4.0K Feb  4 08:41 ..
-rw-rw-r-- 1 ozgur ozgur    8 Feb  4 09:30 deneme.txt
```

Diyez

Girilen komuta veya yapılandırma dosyalarının içeresine yorum satırı girmek için kullanılır. History ile geçmiş girdiğimiz komutlara baktığımızda neyi ne için yaptığımızı hatırlamak için iyi bir yöntemdir veya başka bir administrator'ün yaptığınız işlemleri anamasına kolaylık sağlayacaktır.

```
ozgur@ubuntu:~$ mkdir test #test isimli bir dizin oluşturduk
ozgur@ubuntu:~$ cd test # test dizinine girdik
ozgur@ubuntu:~/test$ echo Linux Eğitimi > test.txt #test.txt isimli bir dosyasının içine veri yazdık.
```

\ backslash

Komutları veya karakterleri birleştirmek için kullanılır.

```
ozgur@ubuntu:~$ mkdir 'onun dosyalari'
ozgur@ubuntu:~$ ls
'onun dosyalari'
ozgur@ubuntu:~$ cd onun\ dosyalari/
ozgur@ubuntu:~/onun dosyalari$ echo Linux \#\?\;\{\Linux #?;{
```

~ Tilde

Tilde işaretinden sonra sunucuda bulunan kullanıcılarından birini yazdığınızda o kullanıcının home dizinine gidersiniz.

```
root@client:~# cd ~ozgur
root@client:/home/ozgur#
```

Değişkenler

İşlem yapılmak istenen veriyi hafızada tutmak için değişkenler kullanılır. Değişkenlerde sayı, harf, tarih gibi bir çok biçimde veri tutabiliriz. Değişkenlerin büyük harfle olması tavsiye edilir. Birden fazla değer alacak bir değişkende veriler arasında : olmalıdır. Linux'de üç çeşit değişken mevcuttur.

```
#!/bin/bash
BACKUP=/home/ozgur/backup-$ (date +%Y%m%d).tar
tar -cvf $BACKUP /home/ozgur/test
```

Bash Değişkenleri

Komut satırında kullanıcı tarafından oluşturulan değişkenlerdir. Herhangi bir değer alabilir. Bash değişkenler herhangi bir program içindeki değişkenlerle birlikte kullanılamaz. Shell ekranında oluşturulan değişkenler oturum kapatılınca kaybolur.

```
MRB="Merhaba Linux!"
```

```
root@client:~# MRB="Merhaba Linux!"
root@client:~# echo $MRB
Merhaba Linux!
```

```
root@client:~# A=45
root@client:~# B=67
root@client:~# C=`expr $A + $B`
root@client:~# echo $C
112
root@client:~#
```

Environment Değişkenleri

Linux sistemlerde ortam değişkenleri shell veya subshell de başlatılan uygulamalar tarafından kullanılabilen dinamik olarak adlandırılmış değerlerdir. Değişkeni tanımlamak için export komutu kullanılır.

```
root@client:~# A=10
root@client:~# cat test.sh
#!/bin/bash
B=20
C=30
D=`expr $A + $B + $C`
echo $D
root@client:~# ./test.sh
50
root@client:~# export A=10
root@client:~# ./test.sh
60
root@client:~#
```

Environment değişkenleri görmek için komut satırına **env** komutu yazılır.

```
ozgur@client:~$ env
SHELL=/bin/bash
PWD=/home/ozgur
LOGNAME=ozgur
XDG_SESSION_TYPE=tty
MOTD_SHOWN=pam
HOME=/home/ozgur
LANG=en_US.UTF-8
LS_COLORS=rs=0:di=01;34:ln=01;36
=40;33;01:or=40;31;01:mi=00:su=3
4:ex=01;32:*.tar=01;31:*.tgz=01;
31:*.lzh=01;31:*.lzm
```

Bu değişkenleri kalıcı olarak tutmak için .profile veya .bashrc dosyalarının içlerine tanımlayabiliriz.

```
Last login: Wed Feb 24 19:41:22 2021 from 192.168.1.37
Merhaba bashrc
Merhaba Linux
ozgur@client:~$ sudo su -
[sudo] password for ozgur:
root@client:~# ls
snap test.sh
root@client:~# ./test.sh
60
root@client:~# cat test.sh
#!/bin/bash
B=20
C=30
D=`expr $A + $B + $C`
echo $D
root@client:~# echo $A
10
```

Sistem Değişkenleri

Linux Bash Shell tarafından oluşturulan değişkenlerdir. İşletim sistemi yükleniği zaman bir çok sistem değişkeni de beraberinde oluşturulur.

set | less komutuyla sunucuda tanımlı tüm değişkenler görüntülenir.

Shell Değişkeni - \$

\$ işaretinin değişkenlere atanmış verilerin kullanılması için kullanılır. Komut satırında **env** (environment) yazdığınızda işletim sistemi tarafından atanmış değişkenlerin bir kısmını görebilirsiniz. Bu değişkenleri \$Değişken_ismi şeklinde kullanabilirsiniz.

```
ozgur@ubuntu:~$ echo $USER
ozgur
ozgur@ubuntu:~$ echo $SSH_CONNECTION
192.168.1.38 53782 192.168.1.40 22
```

Kendi değişkeninizi oluşturarak da kullanabilirsiniz.

```
ozgur@ubuntu:~$ IPim=192.168.1.40
ozgur@ubuntu:~$ echo 'Bilgisayarımın IP'si $IPim 'dir'
Bilgisayarımın IPsi 192.168.1.40 dir
```

* Değişkenler küçük büyük harfe duyarlıdır.

* unset ile değişkeni silebilirsiniz.

\$LANG ile işletim sistemi dilini öğrenebilirsiniz.

```
ozgur@ubuntu:~$ echo $LANG
en_US.UTF-8
```

Wildcard * ? []

* (**Asteriks**), işaretini dizin ve dosya işlemlerinde komutlarla birlikte kullanarak kendinize kolaylık sağlayabilirsiniz.

Bir dizin altındaki ismi file ile başlayan her şeyi silmek isterseniz **rm -rf file*** şeklinde kullanabilirsiniz.

```
ozgur@ubuntu:~$ ls
deneme1  deneme2  deneme3  deneme4  deneme5  file1  file2  file3  file4  file5
ozgur@ubuntu:~$ rm -rf file*
ozgur@ubuntu:~$ ls
deneme1  deneme2  deneme3  deneme4  deneme5
```

Bir dizin altındaki her şeyi silmek isterseniz **rm -rf *** yapabilirsiniz.

```
ozgur@ubuntu:~$ ls
deneme1  deneme2  deneme3  deneme4  deneme5  file1  file2  file3  file4  file5
ozgur@ubuntu:~$ rm -rf *
ozgur@ubuntu:~$ ls
ozgur@ubuntu:~$
```

? (**Soru İşareti**), sadece bir karakteri baz alır. Örneğimizde çeşitli adlarda dizinlerimiz mevcut. Bu dizinlerden Y ile başlayıp K ile bitenlerin listelenmesini sağlıyoruz.

```
ozgur@ubuntu:~$ ls
YAK  YAN  YAR  YOK  YÜK
ozgur@ubuntu:~$ ls -lah Y?K
-rw-rw-r-- 1 ozgur ozgur 0 Feb  4 22:48 YAK
-rw-rw-r-- 1 ozgur ozgur 0 Feb  4 22:48 YOK
-rw-rw-r-- 1 ozgur ozgur 0 Feb  4 22:48 YÜK
```

[] (**Köşeli Parantez**), yine dizinleri görüntüleme de filtre maksatlı kullanabiliriz. Örneğimizde sonu K ve N ile biten dizinlerin görüntülenmesini sağlıyoruz.

```
ozgur@ubuntu:~$ ls
YAK  YAN  YAR  YOK  YÜK
ozgur@ubuntu:~$ ls -lah YA[K,N]
-rw-rw-r-- 1 ozgur ozgur 0 Feb  4 22:48 YAK
-rw-rw-r-- 1 ozgur ozgur 0 Feb  4 22:48 YAN
```

3 - DOSYALAR İLE ÇALIŞMAK

Shell Ortamı ve Kullanıcı Profili

Kullanıcınız ile sunucuya login olduğunuzda login olduğunuz kullanıcının home dizinde komut satırına başlarsınız. Home dizini kullanıcının dosya, dizin ve program dosyalarını saklar.

```
[1] login as: ozgur
[2] ozgur@192.168.1.34's password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-65-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information disabled due to load higher than 1.0

94 updates can be installed immediately.
3 of these updates are security updates.
To see these additional updates run: apt list --upgradable

Last login: Tue Feb  2 21:35:55 2021 from 192.168.1.38
ozgur@ubuntu:~$ pwd
/home/ozgur
```

Home dizininden root dizinine geçmek için / kullanılır.

```
ozgur@ubuntu:~$ cd /
ozgur@ubuntu:$ pwd
/
```

Home dizin altında bazı gizli dosyalar vardır. Bu dosyalar kullanıcı oluşturulduğunda otomatik olarak oluşturulur.

```
ozgur@ubuntu:~$ ls -lah
total 28K
drwxr-xr-x 3 ozgur ozgur 4.0K Feb  2 23:02 .
drwxr-xr-x 4 root  root  4.0K Feb  2 20:34 ..
-rw----- 1 ozgur ozgur   541 Feb  2 23:02 .bash_history
-rw-r--r-- 1 ozgur ozgur  220 Feb 25 2020 .bash_logout
-rw-r--r-- 1 ozgur ozgur  3.7K Feb 25 2020 .bashrc
drwx----- 2 ozgur ozgur 4.0K Feb  2 20:37 .cache
-rw-r--r-- 1 ozgur ozgur  807 Feb 25 2020 .profile
```

Sunucu içerisinde oturum açan her kullanıcının yürüttüğü görevleri, /etc dizini altındaki profile dosyası tanımlar.

.bash_history dosyası kapanmış oturumlarınızda girdığınız komutların listesini tutar. Mevcut oturumunuzda girdığınız komutlar siz oturumunuza kapattığınızda bu dosyaya yazılır.
\$HISTFILE değişkeni dosya konumunu tutarken, bu dosyada tutulacak komut sayısını \$HISTFILESIZE değişkeni ile belirleriz.

```
ozgur@ubuntu:~$ echo $HISTFILE
/home/ozgur/.bash_history
ozgur@ubuntu:~$ echo $HISTFILESIZE
2000
```

.bash_logout dosyası oturum kapatma prosedürlerini içerir. Sistemden çıkış yaparken bu dosya çalıştırılır

.bashrc dosyası kullanıcı yeni bir oturum açtığında çalıştırılır.

.profile dosyası kullanıcı ortamını yapılandırmak için kullanılır.

Dosya Editörlerinin Kullanımı

Ubuntu içerisinde varsayılan vi, vim ve nano dosya editörleri vardır. Bazı küçük farklılıklar haricinde temel olarak hepsinin kullanımı aynıdır. Her biri için internet üzerinden cheat sheet diye tabir ettiğimiz en sık kullanılan komutları gösteren pdf veya görsel dosyalar mevcuttur.

Kullanmak istediğiniz dosya editör uygulamasının ismini ve düzenleyeceğiniz dosya adını yazıp enter'a basarak düzenleyiciye erişebilirsiniz.

- * Vi için **vi test.txt**
- * Vim için **vim test.txt**
- * Nano için **nano test.txt**

Dosyaları düzenlemeden önce dosyanın sahibi olduğunuzdan veya root yetkili bir kullanıcı ile düzenleme yaptığınızdan emin olun. Diğer türlü dosyaya girip bir çok düzenleme yaptıktan sonra yetkiniz olmadığından değişiklik yapamazsınız. Eğer böyle bir durum ile

karşılaşırsanız yaptığınız değişikliği mouse ile seçip içeriği geçici olarak hafızaya alın. Yetkili kullanıcı ile girdikten sonra hafızaya aldığınız içeriği mouse'un sağ tuşu ile yapıştırın.

Kullanımı en kolay Nano'dur Herhangi bir tuşa basmadan direk dosya içerisindeki değişiklik yapabilirsiniz. Klavyenizin yön tuşları ile dosya içinde gezebilirsiniz.

nano -B dosya_adınız komutu ile orijinal dosyanızın bir yedeği dosya adının sonuna ~ işaretini getirerek oluşturular. Yaptığınız değişikliklerden sonra eski dosyaya erişmek isterseniz ~ işaretini ile biten dosyadan erişebilirsiniz.

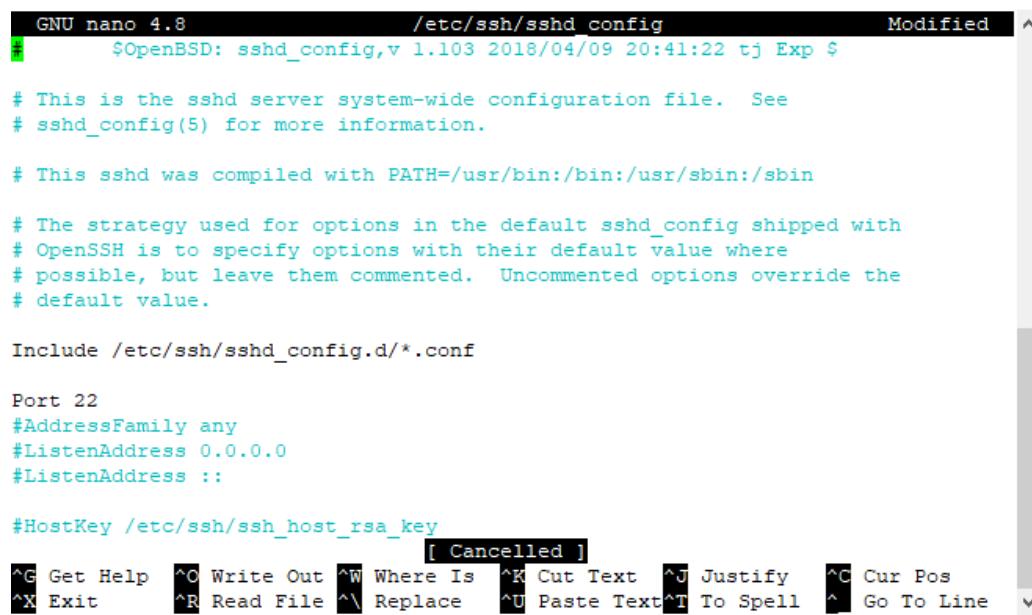
```
ozgur@ubuntu:~$ touch test.txt
ozgur@ubuntu:~$ nano -B test.txt
ozgur@ubuntu:~$ ls
test.txt  test.txt~
```

nano -l dosya_adınız komutu ile dosya içerisindeki satırların başında satır numarası gösterilir.



```
GNU nano 4.8                               test.txt
1 Merhaba
2 Linux
3 Dünya
4
```

Aşağıdaki ekran görüntüsünde en alta bulunan seçenekleri CTRL tuş kombinasyonuyla kullanabilirsiniz. Ctrl + O farklı kaydetken, Ctrl + W dosya içinde arama yapmanızı sağlar. Ctrl + X ile dosyadan çıkmadan önce kayıt edip etmeyeceğiniz sorulduğundan yanıt gerçekleştirilebilir ve dosyadan çıkarılır. Ctrl + _ kombinasyonu ile satır numarası kullanılarak değişiklik yapılmak istenen satır gidilir.



```
GNU nano 4.8                               /etc/ssh/sshd_config      Modified ^
# OpenBSD: sshd_config,v 1.103 2018/04/09 20:41:22 tj Exp $

# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
[ Cancelled ]
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos
^X Exit     ^R Read File  ^\ Replace   ^U Paste Text  ^T To Spell  ^ Go To Line  ^
```

Vim ile dosyanıza eriştiğten sonra düzenlemek yapmak için i (insert) tuşuna basmanız gereklidir. Değişiklik yapıldıktan sonra çıkmak için ESC tuşuna basıp dosya kayıt edilecekse :w! kombinasyonu, edilmeyecekse :q! kombinasyonu kullanılır. Dosyadan çıkmadan kayıt için :w! kombinasyonu kullanılır.

Vi ve Vim Cheat Sheet

i	İmlecin olduğu yerden yazmaya başlar.
---	---------------------------------------

I	İmlecin bulunduğu satırın başından yazmaya başlar.
1G	İlk satıra gider.
G	Son satıra gider.
A	Satır sonuna gider.
dd	İmlecin bulunduğu satırı siler.
:x	Değişiklikleri kaydeder ve çıkar.
:q	Değişiklikleri kaydetmeden çıkar.
:wq!	Değişiklikleri kaydeder ve çıkar.
/	Dosya içinde arama yapar.

Nano Cheat Sheet	
Ctrl+S	Dosyadan çıkmadan kaydeder.
Ctrl+X	Dosyadan çıkar.
Ctrl+A	Satır başı yapar.
Ctrl+E	Satır sonu yapar.
Ctrl+Y	Bir sayfa yukarı çıkar.
Ctrl+V	Bir sayfa aşağı iner.

!!!Önemli!!!!: Özellikle yapılandırma dosyalarında bir değişiklik yapmadan önce mutlaka cp komutu ile bir kopyasını alın. İstemediğiniz durumlarda yedek dosyanızdan geriye dönme şansınız olur.

Dosya İçerliğini Görüntüleme - less, more, cat, head, tail

less, **more** ve **cat** komutları bir dosyanın içeriğini görüntülemek için kullanılır. **less** komutu ile görüntülemek istediğiniz dosyanın ilk satırından başlayarak görüntüleme sağlanır. Yukarı aşağı tuşları ile dosya içinde gezilebilir. İçeriği uzun olan dosyalarda kullanılabilir.

```
ozgur@ubuntu:~$ less /var/log/syslog
Feb  2 20:34:43 ubuntu kernel: [    0.000000] Linux version 5.4.0-65-generic (bu
ilddd@lcy01-amd64-018) (gcc version 9.3.0 (Ubuntu 9.3.0-17ubuntul~20.04)) #73-Ubu
ntu SMP Mon Jan 18 17:25:17 UTC 2021 (Ubuntu 5.4.0-65.73-generic 5.4.78)
Feb  2 20:34:43 ubuntu kernel: [    0.000000] Command line: BOOT_IMAGE=/vmlinuz-
5.4.0-65-generic root=UUID=3ela8564-3424-4112-971e-3a645e658bbf ro maybe-ubiquit
y
Feb  2 20:34:43 ubuntu kernel: [    0.000000] KERNEL supported cpus:
Feb  2 20:34:43 ubuntu kernel: [    0.000000]   Intel GenuineIntel
Feb  2 20:34:43 ubuntu kernel: [    0.000000]   AMD AuthenticAMD
Feb  2 20:34:43 ubuntu kernel: [    0.000000]   Hygon HygonGenuine
Feb  2 20:34:43 ubuntu kernel: [    0.000000]   Centaur CentaurHauls
```

/ (slash) ile dosya içinde arama yapılabilir. Aramak istediğiniz kelimenin başına / (slash) işaretini koyup enter basarak arama yapabilirsiniz.

```
Feb  2 20:34:43 ubuntu kernel: [ 1.768265] pci 0000:00:06.0: reg 0x10: [mem 0
xf0804000-0xf0804fff]
/failed
```

q tuşu ile gösterimden çıkabilirsiniz.

more ile yine dosyanın en başından itibaren gösterime başlanır ancak enter ve boşluk tuşu ile daha fazla satıra bakılabilir. Enter tuşuna her basıldığından 1 satır kaydırılırken, boşluk tuşu ile ekranın tamamı kaydırılacak şekilde içerik görüntülenir.

cat komutu dosya içeriğini ekranda en son satırından itibaren olacak şekilde ekrana yazdırır. İçeriği uzun olan dosyalarda pek kullanılabilir.

```
ozgur@ubuntu:~$ cat test.txt
Merhaba
Linux
Dünya
Merhaba
```

cat (concatenate) aynı zamanda birden fazla dosya içeriğini bir dosyada toplama yeteneğine sahiptir.

```
ozgur@ubuntu:~$ echo Ubuntu > test1
ozgur@ubuntu:~$ echo Centos > test2
ozgur@ubuntu:~$ echo Debian > test3
ozgur@ubuntu:~$ echo GNU > test4
ozgur@ubuntu:~$ cat test1 test2 test3 test4 >testall
ozgur@ubuntu:~$ cat testall
Ubuntu
Centos
Debian
GNU
```

head, varsayılan olarak bir dosyanın ilk on satırını gösterir. Ancak satır sayısı ayarlanabilir.

```
ozgur@ubuntu:~$ head test.txt
Merhaba
Linux
Dünya
Merhaba
Ubuntu
Centos
Unix
Windows
Docker
Kubernetes
ozgur@ubuntu:~$ head -3 test.txt
Merhaba
Linux
Dünya
```

tail, varsayılan olarak en son 10 satırı gösterir. Head 'a olduğu gibi satır sayısı ayarlanabilir. tail -f komutu ilk çalıştırıldığında son 10 satırı göstermekle birlikte canlı olarak dosyaya yazılan verileri ekrana yazdırır.

```
ozgur@ubuntu:~$ tail -f /var/log/auth.log
Feb  4 15:26:29 ubuntu sshd[609]: Server listening on 0.0.0.0 port 22.
Feb  4 15:26:29 ubuntu sshd[609]: Server listening on :: port 22.
Feb  4 15:27:41 ubuntu sshd[737]: Accepted password for ozgur from 192.168.1.38 port 57005 ssh2
Feb  4 15:27:41 ubuntu sshd[737]: pam_unix(sshd:session): session opened for user ozgur by (uid=0)
Feb  4 15:27:41 ubuntu systemd-logind[571]: New session 1 of user ozgur.
Feb  4 15:27:41 ubuntu systemd: pam_unix(systemd-user:session): session opened for user ozgur by (uid=0)
Feb  4 16:17:01 ubuntu CRON[948]: pam_unix(cron:session): session opened for user root by (uid=0)
Feb  4 16:17:01 ubuntu CRON[948]: pam_unix(cron:session): session closed for user root
Feb  4 17:17:01 ubuntu CRON[985]: pam_unix(cron:session): session opened for user root by (uid=0)
Feb  4 17:17:01 ubuntu CRON[985]: pam_unix(cron:session): session closed for user root
```

```
ozgur@ubuntu:~$ tail test.txt
Ubuntu
Centos
Unix
Windows
Docker
Kubernetes
Apple
Zabbix
MySQL
MariaDB
ozgur@ubuntu:~$ tail -3 test.txt
Zabbix
MySQL
MariaDB
```

Filtreler - grep, cut, sort, wc, tr, uniq, sed, awk

grep, en çok kullanılan filtredir. Bir ekran çıktısı veya dosya içeriği içinde istediğiniz bir tanımı grep ile filtreleyebilirsiniz.

```
ozgur@ubuntu:~$ cat test.txt | grep Linux #Genel Kullanım
Linux
ozgur@ubuntu:~$ grep Ubuntu test.txt #cat komutunu kullanmadan
Ubuntu
ozgur@ubuntu:~$ grep U test.txt #Büyük U içeren satırlar
Ubuntu
Unix
ozgur@ubuntu:~$ grep -i u test.txt #Büyük küçük harf ayrimı yapmadan
Linux
Ubuntu
Unix
Kubernetes
ozgur@ubuntu:~$ grep -v U test.txt #Büyük U harfi içermeyenler
Merhaba
Linux
Dünya
```

cut, bir dosyanın içerisindeki kolonları seçip filtreleyebilir. Aşağıdaki komutumuzda /etc/passwd isimli dosyadan –f1,7 ile satırın 1nci ve 7nci sütununu, head -4 ile en son 4 satırını getirmesini sağladık.

```
ozgur@ubuntu:~$ cut -d: -f1,7 /etc/passwd | head -4
root:/bin/bash
daemon:/usr/sbin/nologin
bin:/usr/sbin/nologin
sys:/usr/sbin/nologin
```

tr, bir dosya içerisindeki karakterleri değiştirmek için kullanılır. Ekran çıktısında a'dan başlayıp z'ye kadar olan bütün harfleri büyük harfe çevirerek ekrana getir dedik. **tr** sadece ekran çıktısında değişiklik yapar. Dosya içerisinde değişiklik yapmaz.

```
ozgur@ubuntu:~$ cat test.txt| tr 'a-z' 'A-Z' | head -4
MERHABA
LINUX
DÜNYA
MERHABA
```

wc, bir dosya içerisindeki satır, kelime ve karakterleri saymak için kullanılır.

```
ozgur@ubuntu:~$ wc test.txt #Dosyanın, sırayla satır, kelime ve karakter sayı özetini verir.
15 17 134 test.txt
ozgur@ubuntu:~$ wc -l test.txt #Dosyadaki satır sayısını sayar.
15 test.txt
ozgur@ubuntu:~$ wc -w test.txt #Dosyadaki kelime sayısını sayar.
17 test.txt
ozgur@ubuntu:~$ wc -c test.txt #Dosyadaki karakter sayısını sayar.
134 test.txt
```

sort, varsayılan olarak dosya içeriğini alfabetik sıraya dizer. Ancak belirli sütunlara göre de bu sıralamayı yapabilir.

```
ozgur@ubuntu:~$ tail -5 test.txt | sort #Varsayılan sıralama
GMC,Savana 1500,1999
Honda,Accord,1988
Hyundai,Santa Fe,2009
Kia,Optima,2005
Lexus,RX,2005
ozgur@ubuntu:~$ tail -5 test.txt | sort -k2 #2nci sütuna göre sıralama
Honda,Accord,1988
Kia,Optima,2005
Lexus,RX,2005
GMC,Savana 1500,1999
Hyundai,Santa Fe,2009
```

uniq, dosya içerisindeki tekrar eden verileri kaldırarak ekranda gösterimini sağlar. Aynı zamanda hangi veriden kaç tekrar ettiğini de sayabilir.

```
ozgur@ubuntu:~$ sort test1.txt | uniq
ADANA
HATAY
ISTANBUL
KARS
TOKAT
TRABZON
ozgur@ubuntu:~$ sort test1.txt | uniq -c
 1 ADANA
 1 HATAY
 2 ISTANBUL
 1 KARS
 1 TOKAT
 2 TRABZON
```

sed, akan veriyi düzenlemek için kullanılır. Aşağıdaki komutta Honda ibresini Mercedes ile değiştirerek ekrana yansittık. Ancak 's/Honda/Mercedes/' komutu sadece 1nci satırda değişiklik yapar. İçerigin içindeki tüm Hondaların değişimini istersek 's/Honda/Mercedes/g' yazmalıyız. g ile tüm içeriğin içini manipüle edebiliriz.

```
ozgur@ubuntu:~$ cat test.txt
Aston Martin,V8 Vantage,2008
Honda,Civic,2000
Land Rover,Discovery,2000
GMC,Savana 1500,1999
Honda,Accord,1988
Lexus,RX,2005
ozgur@ubuntu:~$ cat test.txt | sed 's/Honda/Mercedes/'
Aston Martin,V8 Vantage,2008
Mercedes,Civic,2000
Land Rover,Discovery,2000
GMC,Savana 1500,1999
Mercedes,Accord,1988
Lexus,RX,2005
```

Diger bir kullanımda Honda olan satırları silerek ekrana içeriği getirdik.

```
ozgur@ubuntu:~$ cat test.txt | sed '/Honda/d'
Aston Martin,V8 Vantage,2008
Land Rover,Discovery,2000
GMC,Savana 1500,1999
Lexus,RX,2005
```

awk, dosya veya işlemler içindeki veriyi analiz etmek için kullanılan diğer bir araçtır. Ekran çıktısında satır ve sütunlardan oluşan verimizin 1nci sütunu gösterdik. İkinci komutta ise tail -4 ile son 4 veriyi görüntüledik.

```
ozgur@ubuntu:~$ cat test.txt
AstonMartin      V8 Vantage      2008
Honda            Civic           2000
LandRover        Discovery       2000
GMC              Savana1500     1999
Honda            Accord          1988
Lexus             RX              2005
ozgur@ubuntu:~$ cat test.txt | awk '{print $1}'
AstonMartin
Honda
LandRover
GMC
Honda
Lexus
ozgur@ubuntu:~$ cat test.txt | awk '{print $1}' | tail -4
LandRover
GMC
Honda
Lexus
```

İçerik ve Dosya Arama Komutları - find, locate

find, Linux dosya sisteminin içerisinde aradığını bulmak için kullanılan en etkili komutlardan biridir. Dosya tipleri, tarih, izin, grup, dosya boyutu gibi kriterlere göre dosya sisteminde aramalar yapabilir. Herhangi bir paket yüklenmeden Ubuntu içerisinde varsayılan olarak yüklü gelir ve basit syntaxı ile kullanımı kolaydır.

find . -name *.gz komutuyla . yani bulunduğu dizinde uzantısı gz olan içerik ekranda listelenir.

```
root@client:/home/ozgur# find . -name *.gz
./test.tar.gz
root@client:/home/ozgur#
```

find / -name *.bak komutuyla root dizini altında adı fark etmez ama uzantısı bak olan dosyalar listelenir. Bu şekilde arşiv, yedek gibi bulmaya çalıştığınız önemli dosyaları arayabilirsiniz.

```
root@client:~# find / -name *.bak
/etc/apache2/apache2.conf.bak
```

find /etc -type f -name “*.conf” komutumuzun anlamı, /etc klasörü altında tipi file ve uzantısı conf olan dosyaları listeledir.

```
ozgur@ubuntu:~$ find /etc -type f -name "*.conf"
/etc/pam.conf
/etc/security/group.conf
/etc/security/access.conf
/etc/security/namespace.conf
/etc/security/sepermit.conf
```

find / -type d -name "oyun" komutumuzun anlamı, kök dizinden başlayarak tipi dizin ve oyun kelimesi ile başlayan dizinleri getir.

find /home/ozgur/ -type f -name "test*" > arama.txt komutumuzla, çıktı sonuçlarını arama.txt isimli bir dosyaya yazdırıldı.

```
ozgur@ubuntu:~$ find /home/ozgur/ -type f -name "test*" > arama.txt
ozgur@ubuntu:~$ cat arama.txt
/home/ozgur/test1
/home/ozgur/test4
/home/ozgur/test.txt
/home/ozgur/testall
/home/ozgur/test.txt~
```

find / -size +100M komutuyla boyutu 100 Mb'dan yüksek olarak dosyalar listelenir. Dosya sisteminizde yer kaplayan dosyaları bulmak için güzel bir yöntemdir.

```
root@client:~# find / -size +100M
/--diff
/j
/proc/kcore
```

find / -type f -perm 0777 komutuyla farkına varmadan full izin verdığınız dosyaları bulup güvenlik önlemleri alabilirsiniz.

```
root@client:/home/ozgur# find / -type f -perm 0777
/home/ozgur/test.txt
```

find /home -type f -perm 0777 -print -exec chmod 0640 {} \; komutuyla full yetki verdiğiniz dosyaların yetkilerini düzenleyebilirsiniz. exec argümanından farklı komutlar kullanarak örneğin **rm** komutuyla dosyaları silebilirsiniz.

```
root@client:/home/ozgur# chmod 0777 test{1..3}.txt
root@client:/home/ozgur# ls -ltr
total 0
-rw-r--r-- 1 root root 0 Feb 24 08:31 test5.txt
-rw-r--r-- 1 root root 0 Feb 24 08:31 test4.txt
-rwxrwxrwx 1 root root 0 Feb 24 08:31 test3.txt
-rwxrwxrwx 1 root root 0 Feb 24 08:31 test2.txt
-rwxrwxrwx 1 root root 0 Feb 24 08:31 test1.txt
root@client:/home/ozgur# find /home -type f -perm 0777 -print -exec chmod 0640 {} \;
/home/ozgur/test3.txt
/home/ozgur/test1.txt
/home/ozgur/test2.txt
root@client:/home/ozgur# ls -ltr
total 0
-rw-r--r-- 1 root root 0 Feb 24 08:31 test5.txt
-rw-r--r-- 1 root root 0 Feb 24 08:31 test4.txt
-rw-r----- 1 root root 0 Feb 24 08:31 test3.txt
-rw-r----- 1 root root 0 Feb 24 08:31 test2.txt
-rw-r----- 1 root root 0 Feb 24 08:31 test1.txt
```

find . -type f -name “.*” komutuyla gizli olan dosyaları listeleyebilirsiniz.

```
root@client:~# find . -type f -name ".*"  
./.profile  
./selected_editor  
./.bashrc  
./viminfo  
./.bash_history  
./.lessshst
```

find . -type f -user ozgur komutuyla bir kullanıcıya ait dosyaları listeleyebilirsiniz.

Arama sırasında büyük küçük harf ayrimı yapılır. Eğer **-name** argümanına **-i** argümanını eklerseniz ayrim yapmadan içerik listelenir.

```
root@client:/home/ozgur# ls  
scptest.txt  test.tar.gz  test.txt  Test.txt  
root@client:/home/ozgur# find . -name test.txt  
./test.txt  
root@client:/home/ozgur# find . -iname test.txt  
./test.txt  
./Test.txt
```

find /home -mtime 3 komutuyla son üç gün içerisinde değiştirilmiş dosyaları listeleriz. İsterseniz son 30 dakika gibi bir aramayı **-mmin** argümanı ile yapabilirsiniz. Bu komutla sunucunuzda şüphelendiğiniz içerik, yetki gibi dosya ve dizinler üzerindeki değişimleri takip edebilirsiniz.

```
root@client:~# find /home -mtime 3  
/home  
/home/ozgur  
/home/ozgur/test4.txt  
/home/ozgur/.bash_logout  
/home/ozgur/.sudo_as_admin_successful  
/home/ozgur/test3.txt  
/home/ozgur/.profile  
/home/ozgur/test5.txt  
/home/ozgur/.ssh  
/home/ozgur/.bashrc  
/home/ozgur/test1.txt  
/home/ozgur/.bash_history  
/home/ozgur/.cache  
/home/ozgur/.cache/motd.legal-displayed  
/home/ozgur/test2.txt
```

Eğer aramalarınızda “Permission denied” ibaresi görürseniz ibareyi gördüğünüz dosya üzerinde yetkiniz olmadığından içeriği aranamadı anlamındadır. Root, root yetkisi olmayan veya dosya/dizin sahipliği olmayan kullanıcılar bu ibareyle karşılaşır.

locate, find'dan farklı bir araçtır. Kullanmak için ISO veya internet üzerinden mlocate paketini yüklemek gereklidir. **sudo apt install mlocate -y** komutu ile internetten yüklenebilir. Dosyaların yerini belirlemek için indeksleri kullanır.

```
ozgur@ubuntu:~$ locate test.txt  
/home/ozgur/test.txt  
/home/ozgur/test.txt~  
/usr/share/doc/screen/terminfo/test.txt.gz
```

Arama sırasında yeni oluşturduğunuz dosya veya dizinleri göremiyorsanız indekslerin güncellenmediğinden kaynaklanır. **updatedb** ile güncellemelerini sağlayabilirsiniz.

-l argümanı ile ekrana getirilen sonuç sayısını sınırlayabilirsiniz.

```
ozgur@ubuntu:~$ locate -l 1 test.txt  
/home/ozgur/test.txt
```

Hardlink & Soft Link Kullanımı

Hardlink, oluşturulduğu zaman oluşturulduğu dosyanın aynı izin ve sahipliğiyle bir kopyası oluşturulur. Asıl dosya değiştirse hardlink'deki içerikde değişir. Asıl dosya silinse bile hardlink içindeki veriler kalır. Hardlink genelde yedekleme maksadıyla kullanılır. Ancak asıl dosyanın içerisindeki veriler silinirse hardlinkde de silinir. Hardlink oluşturulduğu dosyanın birebir aynısıdır.

Hardlink'leri farklı partition'lar arasında oluşturamayız. Örneğin /dev/sda1 içindeki bir dosya veya dizini /dev/sda2 de bir yere hardlink ile bağlayamayız.

Örneğimizde test.txt dosyası için bir hardlink oluşturduk. Ardından test.txt dosyasına bir satırlık veri ekledik. Hardlink oluşturduğumuz dosyayı görüntülediğimizde test.txt'ye eklediğimiz içeriğin bu dosyaya da eklendiğini gördük.

Devamında text.txt dosyasını sildik. Hardlink oluşturduğumuz dosyanın hala var olduğunu ve içeriğinde aynı şekilde korunduğunu gördük.

```
ozgur@ubuntu:~$ ls
test.txt
ozgur@ubuntu:~$ cat test.txt
Audi,5000S,1984
Toyota,Celica,1978
Daewoo,Lanos,2001
Hyundai,Sonata,2002
Dodge,Ram 2500 Club,1999
Kia,Rio,2010
Dodge,Ram 1500,2002
Dodge,Stealth,1995
ozgur@ubuntu:~$ ln test.txt hardlink_test.txt
ozgur@ubuntu:~$ echo Buick,Rendezvous,2004 >> test.txt
ozgur@ubuntu:~$ cat hardlink_test.txt
Audi,5000S,1984
Toyota,Celica,1978
Daewoo,Lanos,2001
Hyundai,Sonata,2002
Dodge,Ram 2500 Club,1999
Kia,Rio,2010
Dodge,Ram 1500,2002
Dodge,Stealth,1995
Buick,Rendezvous,2004
ozgur@ubuntu:~$ ls
hardlink_test.txt test.txt
ozgur@ubuntu:~$ rm -rf test.txt
ozgur@ubuntu:~$ ls
hardlink_test.txt
ozgur@ubuntu:~$ cat hardlink_test.txt
Audi,5000S,1984
Toyota,Celica,1978
Daewoo,Lanos,2001
Hyundai,Sonata,2002
Dodge,Ram 2500 Club,1999
Kia,Rio,2010
Dodge,Ram 1500,2002
Dodge,Stealth,1995
Buick,Rendezvous,2004
```

Softlink, Windows'daki kısayolun aynısıdır. Partition'lar arası softlink oluşturabilir. Bir ismi başka bir isme bağlamak için kullanılır. Bu kısayol üzerinden dosyaya erişilebilir ve içerik değiştirilebilir. Ancak asıl dosya silinirse, soft linkin hiçbir anlamı kalmaz.

Örneğimizde /etc klasörü altındaki test.txt dosyasına /home/ozgur dizini altında bir softlink oluşturduk. Softlink'in içeriğini görüntülediğimizde test.txt'nin içeriğini gördük. Ardından asıl dosya olan test.txt dosyasını sildik. Softlink'i kontrol ettiğimizde renginin değiştiğini görüntüledik. Bunun anlamı softlink'in bağlı olduğu dosya silindi ve softlink artık kullanılamaz.

```
ozgur@ubuntu:~$ ln -s /etc/test.txt softlink_test.txt
ozgur@ubuntu:~$ ls
softlink test.txt
ozgur@ubuntu:~$ cat softlink_test.txt
Audi,5000S,1984
Toyota,Celica,1978
Daewoo,Lanos,2001
Hyundai,Sonata,2002
Dodge,Ram 2500 Club,1999
Kia,Rio,2010
Dodge,Ram 1500,2002
Dodge,Stealth,1995
Buick,Rendezvous,2004
ozgur@ubuntu:~$ sudo rm -rf /etc/test.txt
ozgur@ubuntu:~$ ls
softlink test.txt
```

Hem softlink hem de hardlink'leri silmek için **rm** komutunu kullanabilirsiniz.

4 - İŞLETİM SİSTEMİ BAĞLANTI TİPLERİ VE SERVİS KONTROL

İşletim Sistemine Login Olma - TeleTYpewriter (TTY), Pseudo Terminal Slave (PTS)

Bir işletim sistemine konsol veya uzak bağlantı (terminal) ile login olabiliriz. Konsol bağlantısı, sunucuya direk olan bağlantıdır. Sunucuya klavye, mouse ve monitör bağlı olarak erişim sağladığımız erişim çeşididir. Sanal sunucu tarafında da konsol bağlantısı aynı mantıkla çalışır. Sunucuya uzaktan erişim sağlayamadığımız durumlarda sanallaştırma ortamındaki konsol bağlantısı ile sanki sunucuya klavye, mouse ve monitör bağlamış gibi oluruz. Bu erişim yöntemi ile sunucunun BIOS'una (Basic Input Output System), boot menüsüne erişim sağlarız.

Uzak bağlantı ise Linux sunucularda Telnet ve SSH bağlantısı ile sağlanır. Telnet güvensiz bir bağlantı yöntemidir. Ağlı dinleyen bir saldırgan tarafından Telnet üzerinden iletilen bilgiler kopyalanabilir. Bu bilgiler ile dinlenen ağa sizlabilir. SSH bağlantısı ise sunucu ile aradaki bağlantıyı şifreleyerek güvenli bir bağlantı oluşturur. Bu şekilde ağlı dinleyen saldırgan anlamsız şifrelenmiş verileri ele geçirse bile amacına ulaşamaz. Ancak varsayılan SSH bağlantısının da bazı zafiyetleri vardır. İlerleyen konularda bu zafiyetlerin nasıl giderileceğine detaylıca bakacağız.

Bir sunucuya konsol ile bağlantı sağlandığında tty şeklinde görünür. Aşağıdaki ekran çıktısında ozgur adlı kullanıcının konsol bağlantısı kullanarak saat 08:50'de sisteme login olduğunu görüyoruz. Bağlantı şeklimiz sanal konsol bağlantısı olduğu için tty1 olarak çıktı aldık. Fiziksel olarak bir bağlantı sağlamasaydık ttys olarak görecektik.

```
ozgur@ubuntu:~$ w
 08:50:52 up 20 min,  1 user,  load average: 0.00, 0.00, 0.03
USER   TTY      FROM          LOGIN@    IDLE   JCPU   PCPU WHAT
ozgur   tty1     -           08:50    3.00s  0.08s  0.00s w
```

Sunucuya SSH ile bağlantı yaptığımızda ise bağlantı şeklini pts olarak görürüz. Ekran çıktısında ozgur adlı kullanıcının SSH bağlantısı kullanarak 192.168.1.38 IP'sinden saat 09:01'de sisteme login olduğunu görüyoruz. Tty bağlantısında FROM bölümünün boştu çünkü sunucuya direk erişim sağlamıştık.

```
ozgur@ubuntu:~$ w
 09:01:45 up 31 min, 1 user, load average: 0.07, 0.02, 0.00
USER      TTY      FROM          LOGIN@    IDLE      JCPU      PCPU WHAT
ozgur     pts/0    192.168.1.38    09:01   1.00s  0.04s  0.00s w
```

Komutlar - w, who, whoami, last, id, su, su –

Sistem üzerindeki kullanıcılar hakkında bilgi almak için kullanılan bazı komutlar vardır.

w , sisteme login olan kullanıcıların ayrıntılı listesini verir. Ekran çıktısına bakıldığından, komutun 09:32:30'da çalıştırıldığı, 1 saat 2 dakikadır sistemin açık olduğu ve 2 kişinin sisteme login olduğu görülmektedir. Kullanıcılarından aydın olan konsol bağlantısı ile 09:31'de bağlantı sağlamışken, ozgur kullanıcısı 192.168.1.38 IP'sinden 09:01'de sisteme giriş yapmıştır.

```
ozgur@ubuntu:~$ w
 09:32:30 up 1:02, 2 users, load average: 0.31, 0.12, 0.04
USER      TTY      FROM          LOGIN@    IDLE      JCPU      PCPU WHAT
aydin     tty1      -           09:31   54.00s  0.01s  0.00s -sh
ozgur     pts/0    192.168.1.38    09:01   4.00s  0.04s  0.00s w
```

who , biraz daha özet bir bilgi ile sadece sisteme giriş yapmış (login) kullanıcılar hakkında bilgi verir.

```
ozgur@ubuntu:~$ who
aydin     tty1      2021-02-05 09:31
ozgur     pts/0    2021-02-05 09:01 (192.168.1.38)
```

whoami , sisteme girmiş olduğunuz kullanıcının kim olduğu ve bağlantı bilgilerini verir. Whoami birleşik yazıldığında sadece kullanıcı ismi çıktısını verirken, ayrı yazıldığında kullanıcı, bağlantı şekli, zamanı ve IP adres bilgilerini verir.

```
ozgur@ubuntu:~$ whoami
ozgur
ozgur@ubuntu:~$ who am i
ozgur     pts/0    2021-02-05 09:01 (192.168.1.38)
```

last komutu sistemde oturum açıp kapatan kullanıcı, sistemin açılış ve kapanış bilgilerini gösterir.

```
root@test:~# last
ozgur     pts/0    192.168.1.39    Mon Mar 15 14:30  still logged in
reboot   system boot 5.4.0-66-generic Mon Mar 15 14:29  still running
ozgur     tty1      Sat Mar 13 23:10 - down      (00:00)
reboot   system boot 5.4.0-66-generic Sat Mar 13 23:10 - 23:10  (00:00)
ozgur     pts/0    192.168.1.34    Sat Mar 13 21:10 - 23:09  (01:58)
```

last -x komutu ile sistem çalışma düzeyleri ve kapatma bilgileri gösterilir.

```
root@test:~# last -x
ozgur     pts/0    192.168.1.39    Mon Mar 15 14:30  still logged in
runlevel (to lvl 5) 5.4.0-66-generic Mon Mar 15 14:29  still running
reboot   system boot 5.4.0-66-generic Mon Mar 15 14:29  still running
shutdown system down 5.4.0-66-generic Sat Mar 13 23:10 - 14:29 (1+15:18)
ozgur     tty1      Sat Mar 13 23:10 - down      (00:00)
runlevel (to lvl 5) 5.4.0-66-generic Sat Mar 13 23:10 - 23:10  (00:00)
```

last username komutu ile belirli bir kullanıcı hakkında login bilgisi alınır.

```
root@test:~# last ozgur
ozgur      pts/0        192.168.1.39      Mon Mar 15 14:30  still logged in
ozgur      tty1          Mon Mar 13 23:10 - down    (00:00)
ozgur      pts/0        192.168.1.34      Sat Mar 13 21:10 - 23:09  (01:58)
ozgur      pts/0        192.168.1.34      Fri Mar 12 17:10 - 20:21  (03:11)
ozgur      tty1          Fri Mar 12 17:09 - down    (03:11)
```

lastb komutu başarısız erişim girişimlerinin listesini gösterir.

```
root@test:~# lastb
192.168. ssh:notty      192.168.1.34      Mon Mar  8 15:59 - 15:59  (00:00)
192.168. ssh:notty      192.168.1.34      Mon Mar  8 15:59 - 15:59  (00:00)
```

lastlog komutu sistem üzerindeki kullanıcıların en son ne zaman login olduklarını listeler.

Username	Port	From	Latest
root			**Never logged in**
daemon			**Never logged in**
bin			**Never logged in**
sys			**Never logged in**
sync			**Never logged in**
games			**Never logged in**
man			**Never logged in**
lp			**Never logged in**

id, sırasıyla kullanıcının id'sini, bağlı olduğu birincil grubu ve ait olduğu grup listesini görüntüler. Ekran çıktısında aydin kullanıcısının grup listesine dikkat edildiğinde yeni adlı bir gruba daha üye olduğu görülür.

```
ozgur@ubuntu:~$ id ozkan
uid=1002(ozkan) gid=1003(ozkan) groups=1003(ozkan)
ozgur@ubuntu:~$ id aydin
uid=1001(aydin) gid=1001(aydin) groups=1001(aydin),1002(yeni)
```

su, komutu kullanıcılar arası geçiş için kullanılır. ozgur kullanıcı ile su komutunu kullanarak aydin kullanıcı olarak login olabiliriz.

```
ozgur@ubuntu:~$ whoami
ozgur
ozgur@ubuntu:~$ su aydin
Password:
```

Ayrıca **su**, **root** kullanıcısının adıdır. **sudo su** yazarak root kullanıcına geçiş yapabilirsiniz.

```
ozgur@ubuntu:~$ whoami
ozgur
ozgur@ubuntu:~$ sudo su
root@ubuntu:/home/ozgur#
```

su - , komutu ile login olduğunuz kullanıcının home dizininden başlayarak erişim sağlarsınız.
– argümanını kullanmazsanız diğer kullanıcı ile giriş yaparsınız ancak hala kendi home dizinizde işlem yapıyor olursunuz.

```
ozgur@ubuntu:~$ su - aydin
Password:
$ pwd
/home/aydin
$ exit
ozgur@ubuntu:~$ su aydin
Password:
$ pwd
/home/ozgur
```

İşletim Sistemini Shutdown ve Reboot Etmek

Eğer işletim sistemini shutdown veya reboot etmek isterseniz shutdown komutunu kullanmalısınız.

shutdown komutunu tek başına kullanırsanız 1 dakika sonrasında planlanmış bir kapatma işlemi gerçekleşir. shutdown -c komutu ile bu işlemi iptal edebilirsiniz.

```
ozgur@ubuntu:~$ sudo shutdown
[sudo] password for ozgur:
Shutdown scheduled for Fri 2021-02-05 10:34:37 UTC, use 'shutdown -c' to cancel.
ozgur@ubuntu:~$ sudo shutdown -c
```

shutdown -P komutu sunucunun fisini çekmekle aynı etkiye sahiptir. Arka planda çalışan uygulamaların hiçbir kapanma prosedürlerini yerini getiremeden sunucu direkt power-off edilir. Test ortamı olarak kullandığınız bir sunucu ise bu komut kullanılabilir ancak production dediğimiz çalışan hizmetleri barındıran bir sunucuda kullanılmamalıdır.

shutdown -H komutu sunucuyu komut girildiği andan itibaren kapatma işlemine sokar. Bu komut girildikten sonra arka plandaki servisler kapatılarak sunucu power-off durumuna geçer.

shutdown -r komutu sunucuyu reboot (yeniden başlatmak) etmek için kullanılır

Ayrıca **reboot** komutu da sunucuyu kapatıp yeniden başlatma işlemlerinde kullanılabilir.

reboot --halt komutu **shutdown -H** ile aynı etkiye sahiptir.

reboot -p komutu **shutdown -P** komutu ile aynıdır.

reboot komutu tek başına sunucuyu yeniden başlatır.

Servisleri Kontrol Etmek - systemctl, enable, disable

Linux sunucu içerisinde çeşitli görevleri yerine getirmek için kullanılan servisler vardır. Servislerin yapılandırma dosyalarında değişiklik yaptığınızda değişikliklerin gerçekleşmesi için ilgili servisi yeniden başlatmanız gereklidir. Sunucunuz yeniden açıldığında ise kritik öneme sahip servislerinizin doğru çalışıp çalışmadığından emin olmak için onları kontrol etmeniz gereklidir. Servisle ilgili yapacağınız güncelleme işlemlerinde ise geçici olarak servisi durdurmanız gerekebilir. Servisleri kontrol etmek için **systemctl** komutunu kullanabiliriz.

systemctl status nginx komutu bize nginx servisinin durumu hakkında bilgi verir. Komut çıktısını özetleyecek olursak 3133 PID'i (Process ID) ile nginx servisinin 3 dakikadır active ve çalışıyor olduğu görüntülenir. En alt kısmında da log dosyasındaki en son kayıtlar gösterilir.

```
root@ubuntu:/home/ozgur# systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
  Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
  Active: active (running) since Fri 2021-02-05 11:44:48 UTC; 3min 1s ago
    Docs: man:nginx(8)
   Process: 3121 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, stat
  Process: 3130 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUC>
 Main PID: 3133 (nginx)
    Tasks: 2 (limit: 1074)
   Memory: 4.4M
      CGroup: /system.slice/nginx.service
          └─3133 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
              ├─3134 nginx: worker process

Feb 05 11:44:48 ubuntu systemd[1]: Starting A high performance web server and a reverse proxy server...
Feb 05 11:44:48 ubuntu systemd[1]: Started A high performance web server and a reverse proxy server.
lines 1-15/15 (END)
```

systemctl restart nginx komutu nginx servisinin yeniden başlamasını sağlayacaktır.

systemctl stop nginx komutu ile nginx servisi duracaktır. Eğer servisi durdurursanız nginx bağlantınız sonlanacaktır. Servisi yeniden aktif hale getirmek için konsol bağlantısı yapmanız gereklidir.

systemctl start nginx komutu ile nginx servisi başlatılır.

Enable ve Disable kavramları sunucu yeniden başlatıldığından bir servisin otomatik olarak yeniden başlatılıp başlatılmayacağını belirler.

systemctl enable nginx komutu nginx servisinin sunucu açılırken otomatik olarak açılması için kullanılır.

systemctl disable nginx komutu ise sunucu açılırken nginx servisinin otomatik olarak başlamamasını sağlar. nginx servisini kullanmak isterseniz **systemctl start nginx** komutuyla başlatmanız gereklidir.

5 - SSH Yapılandırması

SSH Servisi Nedir?

SSH protokolü, network üzerinden güvenli ve şifreli bir şekilde iletişim kurmamızı sağlar. SSH ile sadece sunuculara değil switch ve router gibi cihazlara da erişim sağlanır. Ubuntu, varsayılan olarak OpenBSD SSH uygulamasını kullanır. Uygulama ücretsizdir ve yapılandırması da oldukça basittir. Birçok farklı platformla uyumlu olduğu için yaygın bir şekilde kullanılır.

SSH servisini kullanmadığımız nedeni network trafiğini sürekli dinleyen ve takip eden bilgisayar korsanı tabir ettiğimiz saldırganların (man in the middle attacks), FTP (File Transfer Protocol) ve Telnet gibi şifresiz protokollerden geçen verilere çok kolay erişmesinden dolayıdır. SSH ile iki cihaz arasında şifreli bir iletişim başlatılır. Bir saldırgan trafiği dinlese dahi şifreli verilerle karşılaşacağından anlamlı bilgiler elde edemez.

SSH ile FTP protokolü Secure FTP hale gelir ve dosya transferi sırasında gönderilen veriler şifrelenerek karşı tarafa ilettilir.

SSH1 ve SSH2 protokolleri vardır. SSH1'in zayıflıkları nedeniyle SSH2 yayınlanmıştır. Yapılandırma dosyasında yapılacak değişiklerle sadece SSH2'nin kullanılması sağlanabilir.

OpenBSD verileri şifrelemek için 3DES, AES gibi şifreleme algoritmalarını kullanır.

SSH Servisinin Yapılandırılması ve Güvenli Hale Getirilmesi

SSH servisini yapılandırmak için `/etc/ssh/sshd_conf` dosyasını düzenlememiz gerekiyor. SSH varsayılan olarak root kullanıcısı için açıktır ve 22 numaralı port'u kullanmaktadır. Varsayılan ayarlar ile SSH servisini kullanmak güvenli değildir. SSH servisi iletişimi şifreler ancak varsayılan ayarlar sizin dışınızda saldırganlar tarafından da biliniyor. Bir sunucunun SSH servisi açıldıktan sonra dakikalar içerisinde Brute Force Attacklarına maruz kalır. Eğer şifreniz kolay tahmin edilebilir şifre ise saldırgan sunucuya kolay bir şekilde ele geçirebilir ki saldırganların sürekli zafiyet bularak sunuculara erişim sağlamaya çalışıklarından bahsetmemeye gerek yok sanırım.

Root kullanımı bilgi güvenliği açısından da mahzurludur. Root kullanıcısı sisteme bütün dosyalara erişip değişiklik yapma yetkisine sahiptir. root kullanıcısını kullanan Junior bir sistem admin, sunucuya istemeden de olsa zarar verebilir. Sistem yöneticilerinin tamamı root ile sunucuya girdiğini düşünün. Çok kritik bir değişiklik yapıldı. Kimin yaptığına nasıl bulacaksınız. Bu gibi nedenlerle root kullanıcısının SSH erişimi kapatmakla birlikte sistem üzerinden de kapatılması gereklidir.

Bir sunucunun SSH servisini güvenli hale getirmek için olmazsa olmaz bazı adımlar vardır.

- Belirli bir kullanıcıya SSH izni verip root kullanıcısının SSH erişimi kapatılmalıdır.
- Varsayılan SSH erişim portu değiştirilmelidir.
- SSH erişimi parola kullanılmadan SSH Key kullanarak yapılmalıdır.
- SSH erişimi mümkünse VPN üzerinden sağlanmalıdır. Eğer böyle bir imkan yoksa belirli IP'lere izin verilerek erişim sağlanmalıdır.

Kullanıcı sınırlaması ve root kullanıcısını kapatmak

İşte ilk olarak root kullanıcısının SSH erişimini kapatıp root yetkilerine sahip bir kullanıcı oluşturmakla başlıyoruz. Root kullanıcısını kapatmak saldırganların amacına ulaşmasını engelleyecek savunma stratejilerinden bir tanesidir. Öncelikle ozguraydin isimli bir kullanıcı oluşturarak işe başlıyorum.

useradd -m ozguraydin komutuyla kullanıcı oluşturuyoruz. **-m** belirteci oluşturduğumuz kullanıcı için aynı zamanda home klasörü altında bir dosya oluştur anlamına geliyor.

passwd ozguraydin komutu oluşturduğumuz kullanıcıya parola atıyoruz. Kullanıcılarla vereceğiniz parolaların karmaşık ve tahmin edilmesi zor olması gerektiğini hatırlatmama gerek yok sanırım :)

usermod -aG sudo ozguraydin komutu ile admin grubuna kullanıcıyı dahil ediyoruz.

SSH bağlantısı için **/etc/ssh/sshd_config** dosyasını düzenliyoruz. Bu dosyada Root kullanıcısının erişim sağlamasına izin veren **PermitRootLogin** satırını no olarak değiştiriyoruz. Hemen alt satırda **AllowUsers ozguraydin** ifadesini ekleyerek yeni oluşturduğumuzu kullanıcıya izin veriyoruz. Dosyayı kaydedip çıkıyoruz.

```
#LoginGraceTime 2m
PermitRootLogin no
AllowUsers ozguraydin
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

Son olarak systemctl restart sshd ile SSH servisini yeniden başlatıyoruz.

root kullanıcısı ile giriş yapmaya çalıştığımızda erişim sağlayamıyoruz.

```
[root@192.168.1.41 ~]# login as: root
[root@192.168.1.41 ~]# root@192.168.1.41's password:
[root@192.168.1.41 ~]# Access denied
[root@192.168.1.41 ~]# root@192.168.1.41's password: [REDACTED]
```

Dilerseniz **AllowGroups** parametresi sadece izin verdığınız grubun üyelerine izin verebilirsiniz.

```
#LoginGraceTime 2m
PermitRootLogin no
AllowUsers ozguraydin
AllowGroups yeni
#StrictModes yes
```

Belirli kullanıcıları veya grubu **DenyUsers** veya **DenyGroups** parametresi ile de engelleyebilirsiniz.

Kullanıcı veya gruplara, izin verme ve engelleme bir politika belirlenerek kullanılmalıdır. Eğer sunucunuzda 2 kullanıcı varsa AllowUsers kullanmak iyi bir seçenektedir. Eğer çok fazla kullanıcınız varsa izin vereğiniz kullanıcıları gruplayarak AllowGroups kullanabilirsiniz. Veya birçok kullanıcınız var ama bazılarının erişimini engellemek istiyorsunuz. Bu sefer DenyUsers iyi bir seçenek olabilir.

Port Değişikliği

Diğer önemli değişiklik SSH bağlantısının portu varsayılan olarak 22'dir. Portu değiştirek saldırganların 22 portundan sunucuya erişimini engelleyeceğiz. Elbette dışarıya açık olan bir sunucuya nmap taraması yapan bir saldırgan, sunucunun SSH erişimi için kullandığı portu bulacaktır. Ancak amaç saldırgana kolay yem olmamaktır. Bunun için /etc/ssh/sshd_config dosyamızın altına gidiyoruz. Port numaramızı değiştiriyoruz. Burada kullanacağımız portun sunucu içerisinde başka bir uygulama tarafından kullanılmadığına dikkat etmek gerekiyor.

```
Include /etc/ssh/sshd_config.d/*.conf

Port 5722
#AddressFamily any
#Listen 0.0.0.0
#ListenAddress ::
```

Yine **systemctl restart sshd** ile SSH servisini yeniden başlatıyoruz.

Artık yeni belirlediğimiz portu kullanarak sunucumuza erişim sağlayabiliriz. Elbette bu değişiklikten sonra firewall kullanıyorsanız orada da gerekli kural değişikliklerini yapmanız gerekiyor.

Belirli bir adrese SSH isteklerine izin vermek

Sunucunuz birden fazla IP'ye sahip olabilir. Sunucu varsayılan olarak bütün adreslerden SSH erişimini dinler.

```
tcp        0      0  0.0.0.0:5722          0.0.0.0:*               LISTEN
```

Belirli bir adres üzerinden bu erişimi sağlamak istiyorsanız ListenAddress satırını düzenleyebilirsiniz.

```
Protocol 2

Port 5722
#AddressFamily any
ListenAddress 192.168.1.41
#ListenAddress ::
```

Bu şekilde sunucunuz sadece belirtiğiniz IP üzerinden SSH portunu dinleyecektir.

```
tcp      0      0 192.168.1.41:5722      0.0.0.0:*      LISTEN
```

Belirli adreslerden SSH erişimine izin vermek

Sadece belirlediğiniz adreslerden SSH erişim sağlamak istiyorsanız, burada sunucu güvenlik duvarı çok işe yarayacaktır.

Ubuntu üzerinden örnek vereceğiz ancak diğer işletim sistemlerinde de benzer mantık geçerlidir. Ubuntu'da güvenlik duvari olarak ufw kullanılıyor. Siz isterseniz iptables veya bildiğiniz beğendiğiniz birini kurabilirsiniz. Ufw yi ilk önce active hale getiriyoruz.

sudo ufw enable

Sonrasında sadece belirttiğim IP'den erişim sağlayacağım kuralı giriyorum. Dilerseniz burada bir subnet'te girebilirsiniz.

sudo ufw allow from 192.168.1.209 to any port 5722

Diğer IP'lerden erişimi engellemek için bir kural daha ekliyorum.

ufw deny from any to any port 5722

Sonuçta güvenlik duvarındaki kurallar aşağıdaki gibi görünüyor. Bu şekilde sadece 192.168.1.209 adresinden SSH erişimine izin vermiş olduk.

```
root@ubuntu:/# ufw status numbered
Status: active

          To             Action    From
          --             ----    ---
[ 1] 5722           ALLOW IN   192.168.1.209
[ 2] 5722           DENY IN   Anywhere
[ 3] 5722 (v6)     DENY IN   Anywhere (v6)
```

Host Bazlı Erişim Engelleme

Belirli nedenlerle sunucu içerisindeki güvenlik duvarını kullanmıyorsunuz. O zaman host bazlı erişim engelleme ile belirlediğimiz IP'ler üzerinden erişim sağlayabiliriz. **/etc/hosts.allow** dosyasına giriş yapıyoruz. **sshd : izin_vereceğimiz_ip** satırını ekliyoruz. Ardından **/etc/hosts.deny** dosyasına **sshd : *** satırını ekliyoruz. Bu şekilde sshd servisine belirlediğimiz IP'nin erişimini sağlarken diğer IP'lerden erişimi engellemiş oluyoruz.

```
# /etc/hosts.allow: list of hosts that are allowed to access the system.
#                               See the manual pages hosts_access(5) and hosts_options(5).
#
# Example:      ALL: LOCAL @some_netgroup
#                  ALL: .foobar.edu EXCEPT terminalserver.foobar.edu
#
# If you're going to protect the portmapper use the name "rpcbind" for the
# daemon name. See rpcbind(8) and rpc.mountd(8) for further information.
#
sshd : 192.168.1.209
```

Boş Parolalar ile Erişimi Engelleme

Farkına varmadan parolasını boş geçerek oluşturduğunuz kullanıcılar olabilir. Bu tip kullanıcıların erişimini engellemek için sshd_config dosyamızdaki **PermitEmptyPasswords** satır değerini no yapabiliriz.

```
# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
PermitEmptyPasswords no
```

Açık Kalan SSH Oturumlarını Belirli Bir Süre Sonra Kapatma

Birçok sunucu ile uğraşırken oturumuzun uzun süre işlem yapmadan açık kaldığı zamanlar olabiliyor veya ekran kilidine almadan bilgisayarımızın başından ayrılabiliyoruz. Güvenlik zafiyeti oluşturmamak adına sunuculardaki boşta duran oturumların belirli bir süre sonra kapanmasını sağlayabiliyoruz. Yine sshd_config dosyamızdaki **ClientAliveInterval** değerini saniye cinsinden ayarlayabiliyoruz. Süre dolduğundan sunucu otomatik olarak oturumu sonlandıracaktır.

```
ClientAliveInterval 300
```

Erişim Denemesini Sınırlama

Varsayılan olarak istediğiniz kadar parola denemesi yaparak sunucuya erişim sağlayabiliyorsınız. Ancak saldırganlar bu zafiyeti kullanarak sunucuya brute-force saldıruları düzenleyebiliyorlar. Parola deneme sayısı belirleyerek belirli bir denemeden sonra SSH bağlantısını otomatik olarak sonlandırabiliyorsınız. sshd_config dosyasındaki **MaxAuthTries** değerini belirlediğiniz sayıda değiştirebilirsiniz. Genel itibarıyle 3 sayısı iyidir. 3 denemeden sonra sunucuya bağlanamayan birisinin zaten o sunucuda işi yoktur :)

```
PermitRootLogin no
AllowUsers ozguraydin
#StrictModes yes
MaxAuthTries 3
```

SSH 2 Versiyonunu Kullanmak

SSH'ın 1 nci versiyonunda bir çok zafiyet olması nedeniyle 2 nci versiyonu çıkarıldı. Varsayılan olarak sunucunun 2 nci versiyonu kullanmasını sshd_config dosyamıza ekleyeceğimiz Protocol parametresi ile sağlıyoruz. Bu şekilde versiyon 1 ile gelecek istekleri engellemiş oluyoruz.

```
Protocol 2

Port 5722
#AddressFamily any
ListenAddress 192.168.1.41
#ListenAddress ::
```

LoginGraceTime

LoginGraceTime ile oturum açma isteği yapıldıktan sonra şifre girilmesi için verilecek süre belirlenir. GraceTime ile sunucunuza yapılan boş SSH isteklerini otomatik olarak kapatırırsınız. Ekran çıktısında 192.168.1.42 IP adresinden 2 adet SSH bağlantısı yapılmış. Bu bağlantılar login olmuş bağlantılar değil. Login olmak için parola girilmesini bekliyor. Pek olası değil ama bu şekilde 10000 lerce bağlantı kurup sunucu portlarını meşgul edilebilir.

ozgur@ubuntu:~\$ netstat -an grep 22					
tcp	0	0 0.0.0.0:22	0.0.0.0:*		LISTEN
tcp	0	64 192.168.1.34:22	192.168.1.38:62285		ESTABLISHED
tcp	0	0 192.168.1.34:22	192.168.1.42:40834		ESTABLISHED
tcp	0	0 192.168.1.34:22	192.168.1.42:40832		ESTABLISHED
tcp6	0	0 :::22	:::*		LISTEN

LoginGraceTime 'ı makul bir süreye ayarlamak bu gibi olayların önüne geçecektir. Varsayılan olarak 2 dakika'ya ayarlıdır.

```
LoginGraceTime 2m  
PermitRootLogin no
```

SSH Key Yapılandırması

Sunucunuza bağlanması en güvenli yolu bir SSH Key kullanmaktır. SSH Key kullandığınızda sunucuya parolasız erişim sağlayabilirsiniz. Ayrıca sshd_config dosyası içerisindeki parola ile ilgili parametreleri değiştirerek sunucuya parola ile erişimi tamamen kapatabilirsiniz.

Bir SSH Key oluşturduğunuzda Public ve Private olarak iki key oluşur. Public Key bağlanmak istediğiniz sunucuya yüklenir. Private Key ise bağlanacağınız bilgisayarda saklanır.

Sunucumuza bağlanacağımız bilgisayar üzerinden **ssh-keygen** komutu ile bir SSH key oluşturuyoruz. Passphrase (Parola) kısmını boş bırakmamanızı tavsiye ediyorum. Buraya girdiğiniz parolayı unutmayın. Boş bırakırsanız sadece SSH Key dosyası ile erişim sağlarsanız ancak bir parola belirleyerek key dosyasını eline geçiren birinin sadece dosyayı kullanarak erişimini engellemiştir olursunuz. Test amaçlı bir Centos sunucuda SSH key'i oluşturuyorum.

```
[root@centos7 ~]# ssh-keygen  
Generating public/private rsa key pair.  
Enter file in which to save the key (/root/.ssh/id_rsa):  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in /root/.ssh/id_rsa.  
Your public key has been saved in /root/.ssh/id_rsa.pub.  
The key fingerprint is:  
SHA256:Drjp0bGtDKPEAdTI7zq3bXwGtLl0vIyHpj8jQjM/NWk root@centos7  
The key's randomart image is:  
+---[RSA 2048]---+  
| ..o |  
| .o . |  
| .. |  
| .. o |  
| o o = S |  
| . * * E o |  
| = @ B X . |  
| + =. @ @ + |  
| +.=o@.+ |  
+---[SHA256]---+
```

ssh-copy-id ozguraydin@192.168.1.41 -p 5722 (portu değiştirmiştir çünkü) komutu ile Public Key'i erişim sağlamak istediğimiz sunucuya kopyalıyoruz.

```
[root@centos7 ~]# ssh-copy-id -p 5722 ozguraydin@192.168.1.41
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompt
ed now it is to install the new keys
ozguraydin@192.168.1.42's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh -p '5722' 'ozguraydin@192.168.1.41'"
and check to make sure that only the key(s) you wanted were added.
```

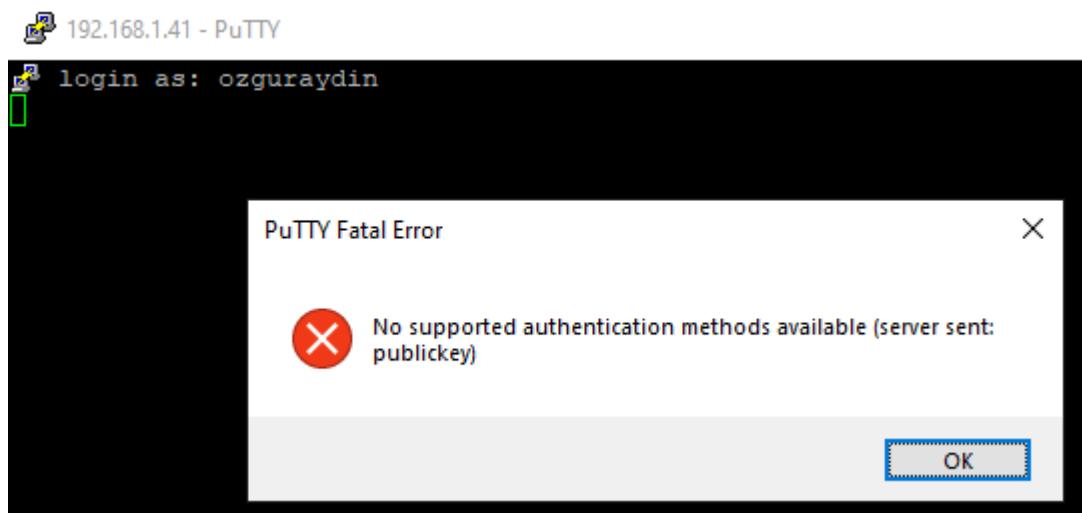
Ardından Centos sunucumuzdan SSH bağlantısı deniyoruz. Key için girdiğimiz parolamızı yazarak sunucumuza erişim sağlıyoruz.

```
[root@centos7 ~]# ssh -p 5722 ozguraydin@192.168.1.41
Enter passphrase for key '/root/.ssh/id_rsa': █
```

Son olarak sshd_config dosyamız içerisinde **PasswordAuthentication** değeri no yaparak parolalı erişimi engelliyoruz.

```
# Example of overriding settings on a per-user basis
#Match User anoncvs
#    X11Forwarding no
#    AllowTcpForwarding no
#    PermitTTY no
#    ForceCommand cvs server
PasswordAuthentication no
```

Parola ile erişim sağlamaya çalıştığımızda Public key ile erişim sağlamamız gerektiğini bildiren bir uyarı alıyoruz.



Windows bir cihazdan powershell ile ssh yapmak istersek yine bir ssh key oluşturuyoruz.

```

PS C:\Windows\system32> ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\Fury/.ssh/id_rsa):
C:\Users\Fury/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\Fury/.ssh/id_rsa.
Your public key has been saved in C:\Users\Fury/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:W4XAeqRF/3JXYh02zsd/Yx12Kp0IoC7wCR62/nxxK4E fury@DESKTOP-M374FAC
The key's randomart image is:
+---[RSA 2048]----+
| oo   o |
| +o . . o |
| =. o . = . |
| o... o + = |
| = ...S + o +o+ |
| o *Eoo .o + +.= |
| o + .+.. o +o+ |
| . . .o . .. o |
| ..o.. . |
+---[SHA256]-----+
PS C:\Windows\system32>

```

Ardında scp ile public key'i sunucumuza kopyalıyoruz.

```

scp -P 5722 C:\Users\Fury/.ssh/id_rsa.pub
ozguraydin@192.168.1.41:/home/ozgur/.ssh/authorized_keys

```

```

PS C:\Windows\system32> scp -P 5722 C:\Users\Fury/.ssh/id_rsa.pub ozguraydin@192.168.1.41:/home/ozguraydin/.ssh/authorized_keys
ozguraydin@192.168.1.41's password:
id_rsa.pub                                              100%  403    130.5KB/s   00:00
PS C:\Windows\system32>

```

ssh -p 5722 ozguraydin@192.168.1.41 ile key kullanarak sunucumuza Windows Client üzerinden bağlanıyoruz.

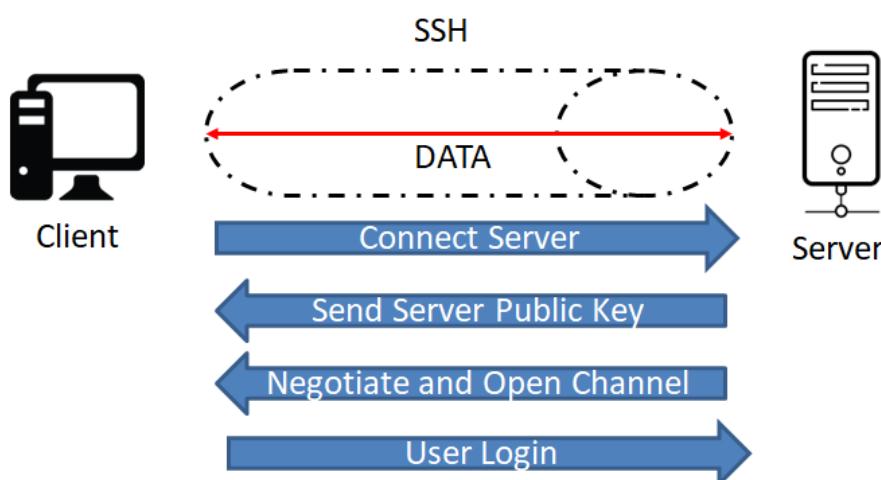
```

PS C:\Windows\system32> ssh ozguraydin@192.168.1.41 -p 5722
Enter passphrase for key 'C:\Users\Fury/.ssh/id_rsa':

```

SFTP (Secure File Transfer Protocol)

Sunucunuz ile uzak sunucu arasında dosya, ses veya video biçiminde olabilecek verileri güvenli bir şekilde aktarmak için kullanılan bir araçtır. Aktarım protokolü FTP (File Transfer Protocol), güvenlik protokülü ise SSH'tir.



En bilindik uygulamalar WINScp ve FileZilla'dır.

6 – DOSYA VE DİZİNLERDE İZİN YÖNETİMİ

Ubuntu, geleneksel olarak UNIX dosya sahipliği ve izin sistemini kullanır. Sistemdeki her şey bir dosya olarak ele alınır ve tüm dosyalara (dizinler ve aygitlar dahil) bir veya daha fazla okuma, yazma ve yürütme izni atanabilir. Bir dosyanın güvenliği, bu izinlerden ve dosya sahipliğiyle sağlanır. Sistem yöneticilerinin görevi bu izinleri yöneterek kullanıcıların dosya erişim kontrolünü sağlayarak ilgizsiz kişilerin hassas dosyalara erişimini engellemektir.

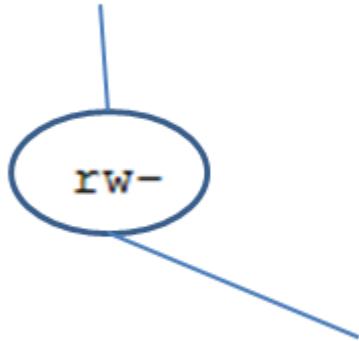
Dosya Sahipliklerini ve İzinleri Anlama

Linux bir sistemde tüm dosyaların ve dizinlerin bir sahibi vardır. Dosya ve dizin üzerinde işlemler yapılrken sahiplik ve izinler dikkate alınır. Bir dosyanın sahibi değilseniz dosyayı okuyamaz ve değiştiremezsiniz veya dosyaya read yani okuma izniniz varsa o dosya üzerinde bir değişiklik yapamazsınız.

Genel olarak sistem yöneticileri el alışkanlığı veya uğraşmak istemediklerinden dosya veya dizinlerin sahipliğini root kullanıcısına ve izinleri de full olarak verirler. Ancak bu davranış işlem yapılan dosyaya istenmeyen kişilerin erişerek verileri çalmasına veya zarar vermesine neden olabilir. Bu nedenle dosya ve dizinlerin izinlerini değiştirmeden önce sahipliklerini ve izinleri anlamak önemlidir.

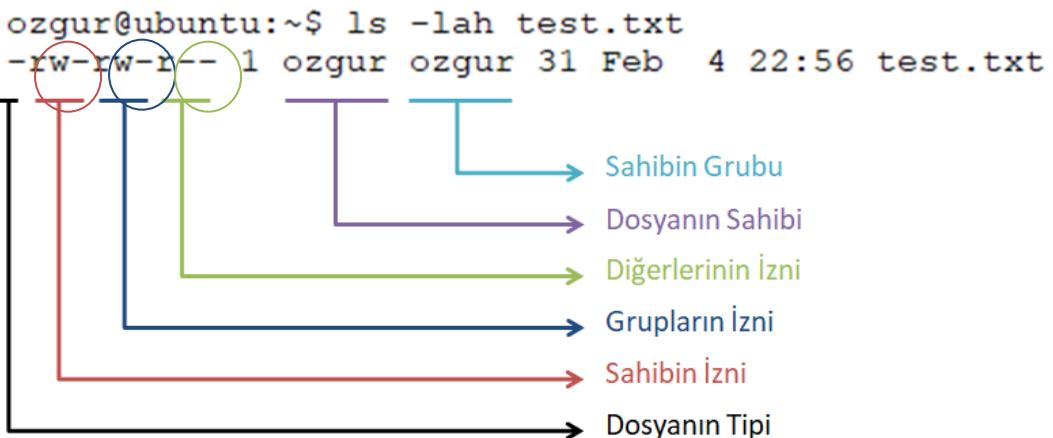
Bir dosyanın veya dizinin read, write ve execute olmak üzere 3 farklı izni bulunur. – ise izin yok anlamındadır.

```
ozgur@ubuntu:~$ ls -lah test.1
-rw-rw-r-- 1 ozgur ozgur 31 Fe
```



- r - Read – Okuma İzni
- w - Write – Yazma İzni
- x- Execute – İşlem Yapma İzni
- (İzin yok) Deny

Okuma, yazma ve işlem yapma izinleri 3 gruptan oluşur. Her bir grup birbirinden bağımsız böülümlere izin verir. İlk bölüm dosya sahibinin izinlerini tanımlar. İkinci bölüm ise dosyanın sahibi olan gruba verilen izinlerdir. Son bölüm ise sistemde kalan diğer kullanıcılaraya verilen izinlerdir.



Erişim Modları

İzinleri vermek için numara veya semboller kullanılır.

Sembolik Mod
(Symbolic Mode)

W ← → 4

r ← → 2

x ← → 1

Numerik Mod
(Absolute Mode)

Bir dosya üzerinde nümerik değerleri kullanarak izinler tanımlanırken verilecek izinlerin sayısal değerleri toplanır.

Numara	İzin Tipi	Hesaplama	Sembol
0	Deny	0	---
1	Execute	1	--x
2	Write	2	-w-
3	Execute+Write	1+2	-wx
4	Read	4	r--
5	Read+Execute	4+1	r-x
6	Read+Write	4+2	rw-
7	Read+Write+Execute	4+2+1	rwx

Sembolik değerler kullanarak izin verileceğe operatörlere ve işaretlere ihtiyaç vardır.

Operatör veya İşaret	Tanım
+	Dosya veya dizine izin ekler.
-	Dosya veya dizinden izni kaldırır.
=	Daha önceden ayarlanan izinlerin üzerine yazar.
u	Kullanıcı \ Sahip (User \ Owner)
g	Grup (Group)
o	Diğerleri (Other)
a	Herkes (all)

Örneğimizde ozgur kullanıcısının home dizininde test.txt isimli dosyanın izinlerine baktığımızda, ozgur kullanıcı için yazma ve okuma iznine, ozgur grubu ve diğerleri (other) için okuma iznine sahiptir. Devamında aydin kullanıcı ile login olduktan sonra aydin kullanıcısının home dizininde bir dosya oluşturduk. Dosyanın izinlerine baktığımızda izinlerin test.txt ile yanı olduğunu ancak grubunun farklı olduğunu görüyoruz.

```
ozgur@ubuntu:~$ pwd
/home/ozgur
ozgur@ubuntu:~$ id
uid=1000(ozgur) gid=1000(ozgur) groups=1000(ozgur),4(adm),24(cdrom),27(sudo),30(dip),46(plugd
ev),116(lxd)
ozgur@ubuntu:~$ ls -ltr
total 0
-rw-rw-r-- 1 ozgur ozgur 0 Feb  5 21:05 test.txt
ozgur@ubuntu:~$ su - aydin
Password:
$ pwd
/home/aydin
$ touch testaydin.txt
$ ls -ltr
total 0
-rw-r--r-- 1 aydin network 0 Feb  5 21:07 testaydin.txt
```

Kullanıcı ve Grup Sahipliğini Değiştirme - chown ve chgrp

Bir dosyanın sahipliğini değiştirmek için chown komutu kullanılır. chown komutu ile birlikte dosyanın yeni sahibi ve grubunu yazdıktan sonra sahipliği değişecek dosya belirtilir.

chown username:groupname filename

```
ozgur@ubuntu:~$ ls -lah test.txt
-rwxrw-r-- 1 ozgur system 31 Feb  4 22:56 test.txt
ozgur@ubuntu:~$ sudo chown aydin:network test.txt
ozgur@ubuntu:~$ ls -lah test.txt
-rwxrw-r-- 1 aydin network 31 Feb  4 22:56 test.txt
```

Bir dosyanın sahipliğini değiştirmek için root veya root yetkisine sahip bir kullanıcı olmalısınız. Örneğimizde ozkan kullanıcıyı sahibi olduğu dosyanın sahipliğini değiştiremedi. Ozgur kullanıcı root yetkisine sahip bir kullanıcı olduğu için sahipliği değiştrebildi.

```
ozkan@ubuntu:~$ ls -ltr
total 0
-rw-r--r-- 1 ozkan system 0 Feb  5 22:31 test.txt
ozkan@ubuntu:~$ chown aydin:system test.txt
chown: changing ownership of 'test.txt': Operation not permitted
ozkan@ubuntu:~$ su - ozgur
Password:
ozgur@ubuntu:~$ sudo chown aydin:system test.txt
ozgur@ubuntu:~$ ls -ltr /home/ozkan/test.txt
-rw-r--r-- 1 ozkan system 0 Feb  5 22:31 /home/ozkan/test.txt
```

Bir dizinin sahipliğini değiştirirken eğer alt dizinleri var ise sahipliğin dizinlere aktarılması için -R (Recursively) argümanını kullanmalısınız.

Örneğimizde ozgur kullanıcısının test isimli bir dosyası ve alt dosyaları mevcut. Sadece chown komutunu kullanarak test dosyasının sahipliğini ozkan kullanıcısına atadı ancak alt dizinlerde bu atama gerçekleşmedi. -R argümanı ile komutu yeniden çalıştırıldığında alt dizin ve dosyaların sahipliği ozkan kullanıcısına geçmiş oldu.

```
ozgur@ubuntu:~$ mkdir test
ozgur@ubuntu:~$ cd test
ozgur@ubuntu:~/test$ mkdir -p 1 2 3
ozgur@ubuntu:~/test$ touch test{1..5}
ozgur@ubuntu:~/test$ ls -ltr
total 12
drwxrwxr-x 2 ozgur ozgur 4096 Feb  5 22:36 3
drwxrwxr-x 2 ozgur ozgur 4096 Feb  5 22:36 2
drwxrwxr-x 2 ozgur ozgur 4096 Feb  5 22:36 1
-rw-rw-r-- 1 ozgur ozgur     0 Feb  5 22:36 test5
-rw-rw-r-- 1 ozgur ozgur     0 Feb  5 22:36 test4
-rw-rw-r-- 1 ozgur ozgur     0 Feb  5 22:36 test3
-rw-rw-r-- 1 ozgur ozgur     0 Feb  5 22:36 test2
-rw-rw-r-- 1 ozgur ozgur     0 Feb  5 22:36 test1
ozgur@ubuntu:~/test$ cd
ozgur@ubuntu:~$ ls -ltr
total 4
drwxrwxr-x 5 ozgur ozgur 4096 Feb  5 22:36 test
ozgur@ubuntu:~$ sudo chown ozkan:system test
ozgur@ubuntu:~$ ls -ltr
total 4
drwxrwxr-x 5 ozkan system 4096 Feb  5 22:36 test
ozgur@ubuntu:~$ cd test
ozgur@ubuntu:~/test$ ls -ltr
total 12
drwxrwxr-x 2 ozgur ozgur 4096 Feb  5 22:36 3
drwxrwxr-x 2 ozgur ozgur 4096 Feb  5 22:36 2
drwxrwxr-x 2 ozgur ozgur 4096 Feb  5 22:36 1
-rw-rw-r-- 1 ozgur ozgur     0 Feb  5 22:36 test5
-rw-rw-r-- 1 ozgur ozgur     0 Feb  5 22:36 test4
-rw-rw-r-- 1 ozgur ozgur     0 Feb  5 22:36 test3
-rw-rw-r-- 1 ozgur ozgur     0 Feb  5 22:36 test2
-rw-rw-r-- 1 ozgur ozgur     0 Feb  5 22:36 test1
ozgur@ubuntu:~/test$ cd
ozgur@ubuntu:~$ sudo chown -R ozkan:system test
ozgur@ubuntu:~$ ls -ltr test/
total 12
drwxrwxr-x 2 ozkan system 4096 Feb  5 22:36 3
drwxrwxr-x 2 ozkan system 4096 Feb  5 22:36 2
drwxrwxr-x 2 ozkan system 4096 Feb  5 22:36 1
-rw-rw-r-- 1 ozkan system     0 Feb  5 22:36 test5
-rw-rw-r-- 1 ozkan system     0 Feb  5 22:36 test4
-rw-rw-r-- 1 ozkan system     0 Feb  5 22:36 test3
-rw-rw-r-- 1 ozkan system     0 Feb  5 22:36 test2
-rw-rw-r-- 1 ozkan system     0 Feb  5 22:36 test1
```

chgrp komutu dosya ve dizinin grup sahipliğini değiştirmek için kullanılır.

sudo chgrp system script

```
ozgur@ubuntu:~$ ls -ltr script
-rw-r-----+ 1 ozgur ozgur 41 Feb  6 13:02 script
ozgur@ubuntu:~$ sudo chgrp system script
[sudo] password for ozgur:
ozgur@ubuntu:~$ ls -ltr script
-rw-r-----+ 1 ozkan system 41 Feb  6 13:02 script
```

Dosya ve Dizin İzin İşlemleri - chmod

Bir dosyanın izinlerini değiştirmek için chmod komutu kullanılır. Örneğimizde aydin kullanıcısı test.txt isimli bir dosya oluşturdu ve ardından chmod ile dosyanın izinlerini 640 olarak yani kendisi için okuma ve yazma (6), grubu için sadece okuma (4) ve diğerleri için deny (0) olarak belirledi. Ardından aydin kullanıcısı ile aynı grupta olan ozkan kullanıcısı dosyayı okudu ancak farklı grupta olan ozgur kullanıcısı dosyayı okumaya çalıştığında Permission denied hatası aldı ve dosyayı okumasına izin verilmedi.

```
$ pwd
/home/aydin
$ echo benim dosyam > test.txt
$ ls -ltr
total 4
-rw-r--r-- 1 aydin system 13 Feb  5 21:38 test.txt
$ chmod 640 test.txt
$ ls -ltr
total 4
-rw-r----- 1 aydin system 13 Feb  5 21:38 test.txt
$ su - ozkan
Password:
$ pwd
/home/ozkan
$ cat /home/aydin/test.txt
benim dosyam
$ su - ozgur
Password:
ozgur@ubuntu:~$ cat /home/aydin/test.txt
cat: /home/aydin/test.txt: Permission denied
ozgur@ubuntu:~$
```

Execute iznini tek başına veremezsiniz. Bir kullanıcı bir dosyayı execute edebilmek için read yetkisine de sahip olması gereklidir. Örneğimizde aydin kullanıcısı ekrana bir çıktı basan script isimli bir dosya hazırlayıp yetkilerini kendisi için tam(7), grup için execute(1) ve diğerleri için read ve execute (5) veriyor. Ozkan kullanıcısı aydin ile aynı grupta olmasına rağmen sadece execute yetkisi olması nedeniyle scripti çalıştırılamıyor. Ozgur kullanıcısı ise other yani diğer kullanıcı yetkileri okuma ve yazma olması nedeniyle scripti çalıştırabiliyor

```
aydin@ubuntu:~$ ls -ltr
total 4
-rw-r--r-- 1 aydin system 41 Feb  5 22:14 script
aydin@ubuntu:~$ chmod 715 script
aydin@ubuntu:~$ ls -ltr
total 4
-rwx--xr-x 1 aydin system 41 Feb  5 22:14 script
aydin@ubuntu:~$ su - ozkan
Password:
ozkan@ubuntu:~$ cd /home/aydin
ozkan@ubuntu:/home/aydin$ ./script
/bin/bash: ./script: Permission denied
ozkan@ubuntu:/home/aydin$ su - ozgur
Password:
ozgur@ubuntu:~$ cd /home/aydin/
ozgur@ubuntu:/home/aydin$ ./script
Merhaba Linux
Fri 05 Feb 2021 10:17:28 PM UTC
ozgur@ubuntu:/home/aydin$ █
```

Sembolik izinler ise yine **chmod** komutuyla kullanılır ancak izin verme işlemleri operatörler ve işaretler ile olur.

```
ozgur@ubuntu:~$ touch test
ozgur@ubuntu:~$ ls -ltr
total 0
-rw-rw-r-- 1 ozgur ozgur 0 Feb  5 22:49 test
ozgur@ubuntu:~$ chmod o+w test #Other'a yazma yetkisi ekledik.
ozgur@ubuntu:~$ ls -ltr
total 0
-rw-rw-rw- 1 ozgur ozgur 0 Feb  5 22:49 test
ozgur@ubuntu:~$ chmod g+x test #Group'a execute yetkisi ekledik.
ozgur@ubuntu:~$ ls -ltr
total 0
-rw-rwxrw- 1 ozgur ozgur 0 Feb  5 22:49 test
ozgur@ubuntu:~$ chmod g-w test #Group'tan okuma yetkisini kaldırdık.
ozgur@ubuntu:~$ ls -ltr
total 0
-rw-r-xrw- 1 ozgur ozgur 0 Feb  5 22:49 test
ozgur@ubuntu:~$ chmod o=x test #Other'a execute yetkisi verirken diğer yetkileri kaldırdık.
ozgur@ubuntu:~$ ls -ltr
total 0
-rw-r-x--x 1 ozgur ozgur 0 Feb  5 22:49 test
ozgur@ubuntu:~$ chmod g-rx test #Group'tan okuma ve yazma yetkisini kaldırdık.
ozgur@ubuntu:~$ ls -ltr
total 0
-rw-----x 1 ozgur ozgur 0 Feb  5 22:49 test
ozgur@ubuntu:~$ chmod a+rwx test #Herkese bütün yetkileri verdik.
ozgur@ubuntu:~$ ls -ltr
total 0
-rwxrwxrwx 1 ozgur ozgur 0 Feb  5 22:49 test
ozgur@ubuntu:~$ chmod o-rwx test #Other'dan tüm yetkileri kaldırdık.
ozgur@ubuntu:~$ ls -ltr
total 0
-rwxrwx--- 1 ozgur ozgur 0 Feb  5 22:49 test
ozgur@ubuntu:~$ chmod a+rwx test #Herkese bütün yetkileri verdik.
ozgur@ubuntu:~$ chmod go-rwx test #Other'dan tüm yetkileri kaldırdık.
ozgur@ubuntu:~$ ls -ltr
total 0
-rwx----- 1 ozgur ozgur 0 Feb  5 22:49 test
ozgur@ubuntu:~$ chmod g+rwx, o+r test #Group'a okuma yazma, Other'a okuma yetkisi verdik.
ozgur@ubuntu:~$ ls -ltr
total 0
-rwxrw-r-- 1 ozgur ozgur 0 Feb  5 22:49 test
```

SUID (Set User ID), SGID (Set Group ID) ve Sticky Bit

Normal bir kullanıcı kendi sahip olduğu dosyalar hariç başka bir kaynağa erişemez ve uygulama çalıştırılamaz. Örneğin bir kullanıcı şifre değiştirmek istesin. Yeni bir şifrenin

/etc/shadow dosyasına yazılması gereklidir ancak bu dosyanın izinlerine bakarsak sahibi root'tur ve diğer kullanıcılar için hiçbir izin verilmemiştir. Ancak bir kullanıcının şifresini değiştirmek istemesi çok normaldir. Bu problemin çözümünü SUID ile sağlıyoruz. SUID bazı dosyaların kullanıcılar tarafından çalıştırılması için özel izin sağlamaktadır. /usr/bin/passwd nin izinlerine dikkat edilirse execute iznini görmemiz gereken yerde s ibaresi ile özel izin atandığını görüyoruz. Bu şekilde şifresini değiştirmek isteyen kullanıcıya passwd komutu için geçici olarak yetki atanarak şifresini değiştirmesi sağlanıyor. SUID bazı sistem dosyalarına varsayılan ayar olarak uygulanır.

```
ozgur@ubuntu:~$ ls -ltr /etc/shadow
-rw-r----- 1 root shadow 1317 Feb  5 11:40 /etc/shadow
ozgur@ubuntu:~$ ls -ltr /usr/bin/passwd
-rwsr-xr-x 1 root root 68208 May 28 2020 /usr/bin/passwd
```

Diğer bir özel izin SGID'dir. SGID ile izin verilen grup kullanıcıları grup sahibinin yetkisine sahip olur. SUID izinde olduğu gibi, SGID bazı sistem dosyalarına varsayılan ayar olarak uygulanır.

```
root@ubuntu:/tmp# mkdir testdizini
root@ubuntu:/tmp# ls -ltr
total 20
drwxr-xr-x 2 root root 4096 Feb  6 10:08 testdizini
root@ubuntu:/tmp# chmod 2777 testdizini/
root@ubuntu:/tmp# ls -ltr
total 20
drwxrwsrwx 2 root root 4096 Feb  6 10:08 testdizini
root@ubuntu:/tmp# su - aydin
aydin@ubuntu:~$ cd /tmp/testdizini/
aydin@ubuntu:/tmp/testdizini$ touch test.txt
aydin@ubuntu:/tmp/testdizini$ ls -ltr
total 0
-rw-r--r-- 1 aydin root 0 Feb  6 10:13 test.txt
aydin@ubuntu:/tmp/testdizini$
```

Sticky Bit için temp klasörünü örnek verebiliriz. Other kullanıcılar için t harfi koyularak bu dizin üzerindeki execute yetkileri kaldırılmıştır. Bu şekilde kullanıcıların birbirlerinin dosyalarını silmesi engellenir.

```
ozgur@ubuntu:~$ ls -ltr / | grep tmp
drwxrwxrwt 11 root root 4096 Feb  6 00:05 tmp
```

Örneğimizde tmp dizin altında ozgur ve aydin kullanıcılarına ait birer dosya var. tmp altında tüm kullanıcı, grup ve diğerleri tam yetkili olmasına rağmen aydin kullanıcısı ile dosyayı silmeye çalıştığında Sticky Bit sayesinde izin verilmedi.

```

aydin@ubuntu:/tmp$ ls -ltr
total 20
-rw-rw-r-- 1 ozgur ozgur      0 Feb   6 10:29 ozgurfile
-rw-r--r-- 1 aydin system      0 Feb   6 10:29 aydinfile
aydin@ubuntu:/tmp$ rm ozgurfile
rm: remove write-protected regular empty file 'ozgurfile'? y
rm: cannot remove 'ozgurfile': Operation not permitted
aydin@ubuntu:/tmp$ rm aydinfile
aydin@ubuntu:/tmp$ ls -ltr
total 20
-rw-rw-r-- 1 ozgur ozgur      0 Feb   6 10:29 ozgurfile
aydin@ubuntu:/tmp$ █

```

Izin	Numeric	Sembolik	Tanım
SUID	4	u+s	Kullanıcı dosya sahibinin izinleri ile işlem yapar.
SGID	2	g+s	Kullanıcı grup sahibinin izinleri ile işlem yapar.
Sticky Bit	1	o+t	Other kullanıcıları dosya içeriğini silemez.

SUID, SGID ve Sticky Bit için izinler Sembolik izinlere benzer şekilde atanır.

chmod u+s filename #Kullanıcılar için özel izin

chmod g+s filename #Grup için özel izin

chmod o+t filename #Sticky Bit

```

ozgur@ubuntu:~$ touch test
ozgur@ubuntu:~$ chmod u+s test
ozgur@ubuntu:~$ ls -ltr
total 0
-rwSr-wr-- 1 ozgur ozgur 0 Feb  6 00:13 test
ozgur@ubuntu:~$ chmod g+s test
ozgur@ubuntu:~$ ls -ltr
total 0
-rwSr-wrSr-- 1 ozgur ozgur 0 Feb  6 00:13 test
ozgur@ubuntu:~$ chmod o+t test
ozgur@ubuntu:~$ ls -ltr
total 0
-rwSr-wrSr-T 1 ozgur ozgur 0 Feb  6 00:13 test
ozgur@ubuntu:~$ █

```

Ayrıca nümerik olarak da atanabilir.

```
ozgur@ubuntu:~$ touch testnumerik
ozgur@ubuntu:~$ chmod 4664 testnumerik
ozgur@ubuntu:~$ ls -ltr
total 0
-rwSr-wr-- 1 ozgur ozgur 0 Feb  6 00:17 testnumerik
ozgur@ubuntu:~$ chmod 2664 testnumerik
ozgur@ubuntu:~$ ls -ltr
total 0
-rw-rwSr-- 1 ozgur ozgur 0 Feb  6 00:17 testnumerik
ozgur@ubuntu:~$ chmod 1664 testnumerik
ozgur@ubuntu:~$ ls -ltr
total 0
-rw-rw-r-T 1 ozgur ozgur 0 Feb  6 00:17 testnumerik
ozgur@ubuntu:~$ chmod 6664 testnumerik
ozgur@ubuntu:~$ ls -ltr
total 0
-rwSrwsr-- 1 ozgur ozgur 0 Feb  6 00:17 testnumerik
ozgur@ubuntu:~$ chmod 7664 testnumerik
ozgur@ubuntu:~$ ls -ltr
total 0
-rwSrwsr-T 1 ozgur ozgur 0 Feb  6 00:17 testnumerik
ozgur@ubuntu:~$
```

ACL Yönetimi

İzinleri kullanarak kullanıcılar ve gruplara yetki verdik ancak bir dosya ve dizine birden fazla kullanıcı veya grup için vermek gerekirse Access Control List (ACL) leri kullanmak gereklidir. ACL ile dosya dizinlerinin içerisinde gelişmiş izin düzenlemeleri yapabilirsiniz.

Ubuntu üzerinde ACL yüklü olmayabilir. **sudo apt install acl -y** komutu ile ACL'i yükleyebilirsiniz.

Bir dosya oluşturulduğunda ACL uygulanmaz.

getfacl komutu dizin veya dosya üzerindeki ACL listesini gösterir. ACL'in ayarlanmadığı gösterimde **ls -ltr** komutu çıkışına benzer ancak dikey bir görünüm çıktısı ekrana gelir.

ACL, XFS dosya sisteminde disable durumdayken Ext4 dosya sisteminde varsayılan olarak enable durumdadır.

```
ozgur@ubuntu:~$ ls -ltr
total 4
-rw-r--r-- 1 ozgur ozgur 41 Feb  6 13:02 script
ozgur@ubuntu:~$ getfacl script
# file: script
# owner: ozgur
# group: ozgur
user::rw-
group::r--
other::r--
```

ACL **setfacl** komutu ile uygulanır.

setfacl -m u:aydin:rx script komutunun açılımı script adlı dosyanın ACL listesini güncelleyerek (-m) user (u) **aydin** olan kullanıcıya okuma ve uygulama (rx) izinlerini ver.

Bir dosya ve dizinde ACL uygulandığını ls -ltr çıktısındaki izinler bölümünün sonundaki + işaretinden anlarız.

```
ozgur@ubuntu:~$ setfacl -m u:aydin:rx script
ozgur@ubuntu:~$ getfacl script
# file: script
# owner: ozgur
# group: ozgur
user::rw-
user:aydin:r-x      +
group::r--
mask::r-x
other::r--

ozgur@ubuntu:~$ ls -ltr script
-rw-r-xr--+ 1 ozgur ozgur 41 Feb  6 13:02 script
```

ACL tanımladıktan sonra mask adlı bir satır oluştuguna dikkat edin. Mask ACL ile atanmış kullanıcı ve gruplar ile ilişkilidir. Diğer izinlerle bir ilişkisi yoktur ve bu izinlerde herhangi bir değişiklik yol açmaz.

Mask'ın amacı kullanıcılara veya guruplara verdığınız izinleri gerekiğinde tek yerden sınırlamanız içindir. Mask'a atadığınız izinlerin full olması kullanıcı veya guruplara yetkinin devrolmasını sağlamaz. Örneğimizde mask için read ve execute yetkisi tanımladık ancak ACL ile izin atadığımız kullanıcıda read yetkisi olduğu için dosyayı çalıştırılamadı. Sonrasında kullanıcıya ACL ile execute yetkisi verince dosyayı çalıştırabildi. Son olarak mask'i sadece read olarak değiştirdik ve kullanıcının execute yetkisi olmasına rağmen dosyayı çalıştırımadı.

Kullanıcılara ACL ile yetki atarken atadığınız izinlere göre maskin kapsamı değişecektir. Ozgur kullanıcısına rwx olarak izin atadıysanız mask rwx olarak güncellenecektir. Ancak maskı rw olarak güncellerseniz ozgur kullanıcısının yetkisi rwx olarak kalacak ancak mask'dan x kaldırıldığı için herhangi bir dosyayı execute edemeyecektir.

```
ozgur@ubuntu:~$ setfacl -m mask:rx script
ozgur@ubuntu:~$ ls -ltr
total 4
-rw-r-xr--+ 1 ozgur ozgur 41 Feb  6 13:02 script
ozgur@ubuntu:~$ getfacl script
# file: script
# owner: ozgur
# group: ozgur
user::rw-
user:aydin:r--
group::r--
mask::r-x
other::r--

ozgur@ubuntu:~$ ./script
-bash: ./script: Permission denied
ozgur@ubuntu:~$ su - aydin
Password:
aydin@ubuntu:~$ cd /home/ozgur
aydin@ubuntu:/home/ozgur$ ./script
-bash: ./script: Permission denied
aydin@ubuntu:/home/ozgur$ exit
logout
ozgur@ubuntu:~$ setfacl -m u:aydin:rx script
ozgur@ubuntu:~$ su - aydin
Password:
aydin@ubuntu:~$ cd /home/ozgur
aydin@ubuntu:/home/ozgur$ ./script
Merhaba Linux
Sat 06 Feb 2021 01:42:31 PM UTC
aydin@ubuntu:/home/ozgur$ getfacl script
# file: script
# owner: ozgur
# group: ozgur
user::rw-
user:aydin:r-x
group::r--
mask::r-x
other::r--

aydin@ubuntu:/home/ozgur$ exit
logout
ozgur@ubuntu:~$ setfacl -m mask:r script
ozgur@ubuntu:~$ su aydin
Password:
aydin@ubuntu:/home/ozgur$ ./script
bash: ./script: Permission denied
aydin@ubuntu:/home/ozgur$
```

```
ozgur@ubuntu:~$ getfacl script
# file: script
# owner: ozgur
# group: ozgur
user::rw-
user:aydin:r-x          #effective:r--
group::r--
mask::r--
other::r--
```

Bir dizine ACL atamak isterseniz –d argümanını kullanmanız gereklidir.

setfacl -d -m u:ozkan:r dosyalarim

```
ozgur@ubuntu:~$ mkdir dosyalarim
ozgur@ubuntu:~$ setfacl -d -m u:ozkan:rw dosyalarim/
ozgur@ubuntu:~$ getfacl dosyalarim/
# file: dosyalarim/
# owner: ozgur
# group: ozgur
user::rwx
group::rwx
other::r-x
default:user::rwx
default:user:ozkan:rw-
default:group::rwx
default:mask::rwx
default:other::r-x
```

Dosya veya dizinden bir ACL kuralını kaldırmak isterseniz -b argümanı kullanmalısınız.

setfacl -b script

```
ozgur@ubuntu:~$ getfacl script
# file: script
# owner: ozgur
# group: ozgur
user::rwx
user:aydin:r-x
group::r--
mask::r-x
other::---

ozgur@ubuntu:~$ setfacl -b script
ozgur@ubuntu:~$ getfacl script
# file: script
# owner: ozgur
# group: ozgur
user::rwx
group::r--
other::---
```

Eğer ACL'den bir kullanıcı veya gurubun iznini kaldırıracaksanız –x argümanını kullanmalısınız.

setfacl -x u:aydin script

aydin ve ozkan kullanıcılarına ACL ile izin verilmiştir. –x argümanı ile aydin kullanıcısının yetkileri kaldırılmıştır.

```
ozgur@ubuntu:~$ ^C
ozgur@ubuntu:~$ setfacl -m u:aydin:rx script
ozgur@ubuntu:~$ setfacl -m u:ozkan:r script
ozgur@ubuntu:~$ getfacl script
# file: script
# owner: ozgur
# group: ozgur
user::rwx
user:aydin:r-x
user:ozkan:r--
group::r--
mask::r-x
other::---

ozgur@ubuntu:~$ setfacl -x u:aydin script
ozgur@ubuntu:~$ getfacl script
# file: script
# owner: ozgur
# group: ozgur
user::rwx
user:ozkan:r--
group::r--
mask::r--
other::---
```

7 - KULLANICI VE GRUP YÖNETİMİ

Sunucular içinde kullanıcılar olmadan çalışmak mümkün değildir. İster 1 ister 100 kullanıcıınız olsun kullanıcıları nasıl yöneteceğinizi öğrenmek önemlidir. Kullanıcıların sayısına bağlı olmaksızın kullanıcı yönetimi temelde aynıdır. Kullanıcı yönetimiyle sistem ve dizin erişimi, parola ilkeleri, disk kotası gibi ilkeleri yapılandırırırsınız.

Kullanıcı Tipleri

Linux bir sisteme super user, normal kullanıcı ve sistem kullanıcı olmak üzere 3 farklı kullanıcı tipi vardır. Sisteme erişecek olan tüm kullanıcıların bir hesabı olmalıdır. Linux sistemlerde kullanıcı hesap bilgileri kullanıcı tipine bakılmaksızın /etc/passwd dosyasında depolanır. Dosya içerisinde kullanıcı adı, parola, kullanıcı, grup ID'si, kullanıcı adı, iletişim bilgisi, kullanıcının /home dizininin yolu ve varsayılan komut dizini gibi bilgiler yer alır. Dosya içerisindeki alanlar : işaretü ile birbirinden ayrılır.

Parola bilgisi X işaretü ile gösterilir. Bunun anlamı kullanıcı bir parolaya sahiptir. Kullanıcı parolaları /etc/shadow dosyası içinde şifrelenmiş halde tutulur.

```
ozgur@ubuntu:/ $ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
```

```
root@ubuntu:~# cat /etc/shadow
root:*:18474:0:99999:7:::
daemon:*:18474:0:99999:7:::
bin:*:18474:0:99999:7:::
sys:*:18474:0:99999:7:::
```

Sunucu içinde sadece bir tane super/root kullanıcı olabilir. Bu kullanıcı sistem üzerindeki tüm kaynaklara sınırsız erişime sahiptir.

```
root@ubuntu:~# id root  
uid=0(root) gid=0(root) groups=0(root)
```

Root kullanıcı ile sistem üzerinde sürekli olarak işlem yapmak tehlikelidir. Yanlışlıkla bir dosyayı silebilir veya bir servisi durdurabilirsiniz. Bu nedenle normal bir kullanıcı hesabı oluşturup bu hesaba root yetkisi atamak iyi bir fikirdir. Böylece ihtiyaç halinde root olarak çalışıp işimiz bittiğinde normal kullanıcı hesabına dönebiliriz.

Ubuntu'da root kullanıcı ilk kurulumda devre dışıdır. root kullanıcısını manuel olarak ayarlamak gereklidir. Test ortamında root kullanıcısı istediğiniz yapabilirsiniz ancak çalışan bir yapıda root ile yapacağınız bir hatanın dönüşü olmayabilir.

Linux sistemlerde root olarak bir komut çalıştırırmak için sudo komutu kullanılır. Eğer kritik bir komut kullanıyorsanız ve root yetkisi ile çalıştırılmıyorsanız o komut gerçekleştirilmeyecektir.

```
ozgur@ubuntu:~$ apt update  
Reading package lists... Done  
E: Could not open lock file /var/lib/apt/lists/lock - open (13: Permission denied)  
E: Unable to lock directory /var/lib/apt/lists/  
W: Problem unlinking the file /var/cache/apt/pkgcache.bin - RemoveCaches (13: Permission denied)  
W: Problem unlinking the file /var/cache/apt/srcpkgcache.bin - RemoveCaches (13: Permission denied)  
ozgur@ubuntu:~$ sudo apt update  
0% [Working]
```

root 'un olduğu oturuma geçmeniz gerekirse sudo su –i veya sudo su – komutu kullanılabilir. Eğer – veya –i kullanmazsanız root'a erişim sağladığınız kullanıcının home dizini üzerinden işlemleri gerçekleştirirsınız. Eğer root'un home dizininde bir işlem yapacaksanız bu durum karışıklığa neden olacaktır. root 'a giriş yaptığımızda satır sonu \$ işaretinden # işaretine dönüşür.

```
ozgur@ubuntu:~$ sudo su  
root@ubuntu:/home/ozgur# exit  
exit  
ozgur@ubuntu:~$ sudo -i  
root@ubuntu:~# exit  
logout  
ozgur@ubuntu:~$ sudo su -  
root@ubuntu:~#
```

Normal kullanıcılar /home dizininde kendilerine ait olan klasör üzerinde işlem yapabilir. Sistem çapında değişiklik yapmasına izin verilmez. Sistem yöneticileri izin yönetimini kullanarak kullanıcılar farklı izinler de atayabilir veya mevcut izinlerini engelleyebilir.

Sistem kullanıcı ise bir şahıs olmayıp sistem üzerindeki çeşitli hizmetleri çalıştıran bir yönetici hesabıdır. Sistem kullanıcılarının bir dizini veya şifresi olmamakla birlikte oturum açmalarına da izin verilmez. Bu kullanıcıların yetkilerinin düzenlenmesi tavsiye edilmez.

```
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin  
bin:x:2:2:bin:/bin:/usr/sbin/nologin  
sys:x:3:3:sys:/dev:/usr/sbin/nologin
```

Sistem üzerindeki UDI (User ID) ve GID (Group ID) değerleri kullanıcıların alfabetik isimlerini temsil ederler. root kullanıcısının UID ve GID değerleri 0:0 ile gösterilir. Normal kullanıcıların UID değerleri 1000'den başlar.

Bir kullanıcının sisteme login olmasını engellemek için /etc/passwd dosyası içindeki kullanıcının bulunduğu satır sonuna **/sbin/nologin** ve **/bin/false** değeri atanır.

Kullanıcı Yönetimi - useradd, userdel

Bir sisteme kullanıcı eklemek için useradd komutu kullanılır. useradd komutu kullanıldığında /etc/passwd dizini altına eklenir. useradd komutu tek başına kullanılabilir ancak aldığı argümanlarla kullanmak daha uygundur. Örnek vererek açıklayalım.

useradd hasan komutu ile herhangi bir özelleştirme yapmadan kullanıcımızı direk ekledik. UID'yi sıradaki ID numarasını atayarak oluşturdu. GID, yine hasan diye bir grup oluşturarak sıradaki GID numarasını atadı. Kullanıcı için bir açıklama girmedik. Home dizini belirlemedik ve komut satırında shell ekranına düşmesi için ayarlandı. Ayrıca hasan için bir kullanıcı dizini de otomatik olarak oluşturulmadı. /home/hasan diye bir dizine gitmek isterseniz böyle bir dizin yok. Sizin manuel oluşturmanız gereklidir.

```
useradd -m -g network -c "Network Admin" -u 2000 -s "/bin/bash" -d /home/ahmet ahmet
```

komutu ile

-m ile kullanıcı adıyla bir home dizini altında bir dizin oluşturduk.

-g ile kullanıcıyı network isimli bir gruba ekledik.

-c ile kullanıcı için bir açıklama girdik.

-u ile bir UID atadık.

-s ile komut satırının bash olarak ayarlanması sağladık.

-d ile home dizinini de kullanıcının dizinini ayarladık.

-p ile kullanıcıya parola ataması yaptık.

Eğer test ortamındaysanız kullanıcıları nasıl atadığınız çok önemli değildir. Ancak canlı ortamda sistem yönetici olarak tüm yapılandırmayı bir standarda göre yapmak görevinizdir.

```
root@ubuntu:/# useradd hasan
root@ubuntu:/# cat /etc/passwd | grep -i hasan
hasan:x:1003:1007::/home/hasan:/bin/sh
root@ubuntu:/# useradd -m -g network -c "Network Admin" -u 2001 -s "/bin/bash" -d /home/mehmet -p Q1w2e3r4 mehmet
root@ubuntu:/# cat /etc/passwd | grep -i mehmet
mehmet:x:2001:1005:Network Admin:/home/mehmet:/bin/bash
root@ubuntu:/# ls -ltr /home/
total 32
drwx----- 2 root      root      16384 Feb  2 19:53 lost+found
drwxr-xr-x  2 ozkan    network   4096 Feb  6 00:33 ozkan
drwxr-xr-x  4 aydin    aydin    4096 Feb  6 00:33 aydin
drwxr-xr-x  7 ozgur    ozgur    4096 Feb  6 19:05 ozgur
drwxr-xr-x  2 mehmet   network   4096 Feb  7 13:07 mehmet
```

userdel komutu ile kullanıcı silinir ancak sadece kullanıcı silinir. Kullanıcının kendisi ve verilerinin tamamını silemek istersek -r argümanını kullanmak gereklidir.

```
userdel -r mehmet
```

```
root@ubuntu:/# userdel ahmet
root@ubuntu:/# ls -ltr /home/
total 36
drwx----- 2 root    root    16384 Feb  2 19:53 lost+found
drwxr-xr-x  2 ozkan   network  4096 Feb  6 00:33 ozkan
drwxr-xr-x  4 aydin   aydin    4096 Feb  6 00:33 aydin
drwxr-xr-x  7 ozgur   ozgur    4096 Feb  6 19:05 ozgur
drwxr-xr-x  2 mehmet  network  4096 Feb  7 13:07 mehmet
drwxr-xr-x  2 2009   network  4096 Feb  7 13:10 ahmet
root@ubuntu:/# userdel -r mehmet
userdel: mehmet mail spool (/var/mail/mehmet) not found
root@ubuntu:/# ls -ltr /home/
total 32
drwx----- 2 root    root    16384 Feb  2 19:53 lost+found
drwxr-xr-x  2 ozkan   network  4096 Feb  6 00:33 ozkan
drwxr-xr-x  4 aydin   aydin    4096 Feb  6 00:33 aydin
drwxr-xr-x  7 ozgur   ozgur    4096 Feb  6 19:05 ozgur
drwxr-xr-x  2 2009   network  4096 Feb  7 13:10 ahmet
```

Parola Yönetimi – passwd, chage

Kullanıcı bilgileri /etc/passwd dosyasında depolanır. Bu dosya kontrol edildiğinde parolanın bulunduğu kısımda x işaretü görülebilir. X işaretini parolanın shadow'udur. Bunun anlamı kullanıcının bir parolasının olduğunu göstermektedir. Kullanıcı parolaları güvenlik nedeniyle /etc/shadow dosyası içinde şifreli bir şekilde tutulur. root ve sistem kullanıcılarının sisteme erişim izinleri olmadığından parola bölgümleri * şekliyle gösterilir. ozgur ve aydin kullanıcılarının parolaları şifrelenmiştir. Eğer bir kullanıcının parola bölümünde ! işaretü varsa o kullanıcıya parola atanmamış demektir.

```
root@ubuntu:/# cat /etc/shadow
root:*:18474:0:99999:7:::
daemon:*:18474:0:99999:7:::
bin:*:18474:0:99999:7:::
sys:*:18474:0:99999:7:::
sync:*:18474:0:99999:7:::
games:*:18474:0:99999:7:::
man:*:18474:0:99999:7:::
lp:*:18474:0:99999:7:::
mail:*:18474:0:99999:7:::
news:*:18474:0:99999:7:::
uucp:*:18474:0:99999:7:::
proxy:*:18474:0:99999:7:::
www-data:*:18474:0:99999:7:::
backup:*:18474:0:99999:7:::
list:*:18474:0:99999:7:::
irc:*:18474:0:99999:7:::
gnats:*:18474:0:99999:7:::
nobody:*:18474:0:99999:7:::
systemd-network:*:18474:0:99999:7:::
systemd-resolve:*:18474:0:99999:7:::
systemd-timesync:*:18474:0:99999:7:::
messagebus:*:18474:0:99999:7:::
syslog:*:18474:0:99999:7:::
_apt:*:18474:0:99999:7:::
tss:*:18474:0:99999:7:::
uuidd:*:18474:0:99999:7:::
tcpdump:*:18474:0:99999:7:::
landscape:*:18474:0:99999:7:::
pollinate:*:18474:0:99999:7:::
systemd-coredump:!:18660:::::
ozgur:$6$Xz7PKTV.6jL5d8LD$1cb.v40r4sXbRMGiPxbygc88Lc8ubHrRo.HcQEbiWYQWKedhy
p/Ij7q7uns6HBwWIHADA4q.I35NxqaqOikEp.:18660:0:99999:7:::
lxd:!:18660:::::
dnsmasq:*:18660:0:99999:7:::
aydin:$6$msUi/M8MyPiEqo6m$/dsahMfCR7IcdA7M/sheo1Xlm7dyavGJG/cxmMFjwIjfZIN2
oBLOJzfO8yHBHJ5jLKL7jcVty50dy6180xWU.:18663:0:99999:7:::
```

passwd dosyası kolonlara ayrılır. Sırasıyla;

Kullanıcı adı,

Parolanın şifreli hali,

Parolanızı değiştirdiğiniz gün ile 1 Ocak 1970 günü arasındaki gün farkıdır. Bu değer gün sayısı olarak parolanın değiştiği günü gösterir.

Parolanın değiştirilebileceği gün sayısıdır. Varsayılan değer 0'dır. Bu şekilde kullanıcı istediği zaman parolasını değiştirebilir. Bir gün sayısı atarsanız parola değişikliği için en son parola değiştirdiği tarihten atadığınız gün sayısı kadar zaman geçmesi gereklidir.

Parolanın değiştirilmesinin zorunlu kalınacağı gün sayısıdır.

Parolanın sona ermesinden önce kullanıcının uyarılacağı gün sayısıdır. Parola değişim süresi 10 kala kullanıcıya şifre değiştirmesini hatırlatmak için kullanılır.

Parolanın sona ermesinden sonra hesabın devre dışı bırakılacağı gün sayısıdır. Kullanıcı parola değişim süresini dikkate almadan şifresini değiştirmeden. Belirlenen gün sayısı sonra hesabı devredışı edilir.

Hesabın devre dışı bırakılacağı gün sayısıdır.

passwd username ile kullanıcıya parola atanır.

```
root@ubuntu:/# passwd hasan
New password:
Retype new password:
passwd: password updated successfully
```

Bir kullanıcının hesabı **-l** ile kilitlenebilir. Kilitlenen hesaba giriş yapılamaz. **-u** ile kilit kaldırılır. Kilitlenen hesabın /etc/shadow klasöründeki şifre bölümünün başına ! işaretü getirilir. Kilitleme işlemi bir nevi parola değişimi ile yapılır.

```
root@ubuntu:/# passwd -l hasan
passwd: password expiry information changed.
root@ubuntu:/# cat /etc/shadow | grep -i hasan
hasan:!$6$n8gwjGT84lpGU6xF$PG1fMqkPhc/chT8VzVLZ8oA1FbRTPEax79iF.Skl/O6m93w.
1mwvk1f.zf/PZQC1EfZWdT/uWhGu.TOn2oCac1:18665:0:99999:7:::
root@ubuntu:/# passwd -u hasan
passwd: password expiry information changed.
root@ubuntu:/# cat /etc/shadow | grep -i hasan
hasan:$6$n8gwjGT84lpGU6xF$PG1fMqkPhc/chT8VzVLZ8oA1FbRTPEax79iF.Skl/O6m93w.1
mwvk1f.zf/PZQC1EfZWdT/uWhGu.TOn2oCac1:18665:0:99999:7:::
```

chage komutu parola polikası ilgili bilgileri görüntülemenizi ve değişiklikler yapmanızı sağlar. Aynı değişiklikleri shadow dosyasını manuel olarak düzenleyerek de yapabilirsiniz. Bu komut ile çalışmadan önce shadow dosyanızın yedeğini almakta fayda var. Yanlışlıkla bütün kullanıcıların (root, system, user) parola politikalarını aynı yaparak istenmeyen durumlara yol açabilirsiniz.

```

root@ubuntu:~# chage -l ahmet
Last password change : Feb 07, 2021
Password expires : never
Password inactive : never
Account expires : never
Minimum number of days between password change : 0
Maximum number of days between password change : 99999
Number of days of warning before password expires : 7
root@ubuntu:~# chage -M 30 ahmet #Parolanın expire olma süresi
root@ubuntu:~# chage -l ahmet
Last password change : Feb 07, 2021
Password expires : Mar 09, 2021
Password inactive : never
Account expires : never
Minimum number of days between password change : 0
Maximum number of days between password change : 30
Number of days of warning before password expires : 7
root@ubuntu:~# chage -I 10 ahmet #Expire süresinden sonra parolanın devredisi kalma süresi
root@ubuntu:~# chage -l ahmet
Last password change : Feb 07, 2021
Password expires : Mar 09, 2021
Password inactive : Mar 19, 2021
Account expires : never
Minimum number of days between password change : 0
Maximum number of days between password change : 30
Number of days of warning before password expires : 7
root@ubuntu:~# chage -E "2021-06-15" ahmet #Hesabın expire olacağı zaman
root@ubuntu:~# chage -l ahmet
Last password change : Feb 07, 2021
Password expires : Mar 09, 2021
Password inactive : Mar 19, 2021
Account expires : Jun 15, 2021
Minimum number of days between password change : 0
Maximum number of days between password change : 30
Number of days of warning before password expires : 7
root@ubuntu:~# chage -W 5 ahmet #Expire süresinden önce uyarının başlayacağı gün sayısı
root@ubuntu:~# chage -l ahmet
Last password change : Feb 07, 2021
Password expires : Mar 09, 2021
Password inactive : Mar 19, 2021
Account expires : Jun 15, 2021
Minimum number of days between password change : 0
Maximum number of days between password change : 30
Number of days of warning before password expires : 5

```

Parola Yönetimi için PAM Modülü Kullanımı

Her ne kadar passwd içerisinde belirli ayarları yapıyor olsakta parolaların daha karmaşık olması için PAM modülü kullanılabilir. PAM modülü, parola değişimi, expire süresi gibi ayarların yapılmasına da izin verir. PAM modülünü kullanmak için yüklenmelidir.

apt -y install libpam-pwquality

PAM modülü parola yönetimi için temel olarak 2 farklı dosyaya sahiptir.

/etc/login.defs dosyası ile parolanın sona erme süresi (PASS_MAX_DAYS), sona erme süresinden önceki hatırlatılacak gün sayısı (PASS_WARN_AGE) ve parolanın kullanılabileceği gün sayısı (PASS_MIN_DAYS) gibi ayarlar yapılır. Yine bu ayarların uygulanacağı kullanıcı UID değerleri de bu dosyada belirtilir.

```

# Password aging controls:

#
#      PASS_MAX_DAYS   Maximum number of days a password may be used
#      PASS_MIN_DAYS   Minimum number of days allowed between password changes
#      PASS_WARN_AGE    Number of days warning given before password expires

PASS_MAX_DAYS    99999
PASS_MIN_DAYS    0
PASS_WARN_AGE    7

#
# Min/max values for automatic uid selection in useradd
#
UID_MIN           1000
UID_MAX           60000

```

/etc/security/pwquality.conf dosyası ile kullanıcılara karmaşık parolalar oluşturmasını sağlamak için parola uzunluğu, sayısı ve harf sayısı, parolanın içermeyeceği karakterler gibi yapılandırmalar yapılabilir.

Değişken	Açıklama
usercheck=1	Parola için kullanıcı isminin kontrolünü yapar
maxrepeat=1	Yeni şifrede kullanılabilecek ardışık karakter sayısı
minclass=3	Şifre içinde bulunacak minimum karakter sınıfını (Büyük, küçük harf, sayı ve özel karakterler) tanımlar.
ocredit=1	Şifrede bulunacak maksimum özel karakter sayısı
ucredit=1	Şifrede bulunacak maksimum büyük harf sayısı
lcredit=1	Şifrede bulunacak maksimum küçük harf sayısı
dcredit=1	Şifrede bulunacak maksimum rakam sayısı
minlen=8	Şifrenin minimum karakter uzunluğu
difok=1	Yeni şifrede, eski şifrenin sahip olduğu karakterlerden kullanabilme sayısı

```
root@test:~# grep "^[^#*/;]" /etc/security/pwquality.conf
difok = 1
minlen = 8
dcredit = 3
ucredit = 1
lcredit = 3
ocredit = 1
minclass = 4
maxrepeat = 1
usercheck = 1
```

Grup Yönetimi – groupadd, groupdel

Gruplar, kullanıcıları yönetmeyi çok daha kolay hale getirerek her kullanıcıya ayrı ayrı izin atmak zorunda kalmadan, çok sayıda kullanıcıya hızlı bir şekilde izin vermek veya iptal etmek için kullanılır. Yöneticiler gruplara dizin oluşturarak grup kullanıcılarının ortak bir alanda çalışma yapmasını kolay bir şekilde sağlayabilir.

Sistem üzerindeki gruplar /etc/group dosyasında tutulur.adm grubunda syslog ve ozgur kullanıcılarının olduğu görülür.

cat /etc/group

```
ozgur@ubuntu:~$ cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,ozgur
```

groupadd komutu grup eklemek için kullanılır.

groupdel komutu grup silmek için kullanılır.

gpasswd komutu gruba bir parola oluşturur.

useradd -G Kullanıcı oluştururken bir gruba kullanıcıya ekler.

deluser - gruptan kullanıcı silmek için kullanılır.

```
root@ubuntu:~# id ozkan
uid=1002(ozkan) gid=1005(network) groups=1005(network)
root@ubuntu:~# groupadd devops
root@ubuntu:~# usermod -G devops ozkan
root@ubuntu:~# id ozkan
uid=1002(ozkan) gid=1005(network) groups=1005(network),1006(devops)
root@ubuntu:~# deluser ozkan devops
Removing user 'ozkan' from group 'devops' ...
Done.
root@ubuntu:~# id ozkan
uid=1002(ozkan) gid=1005(network) groups=1005(network)
root@ubuntu:~# groupdel devops
```

Usermod

Linux sistemlerde varolan kullanıcılar üzerinde değişiklik yapmak için kullanılan araçtır. Usermod farklı argümanlar alarak kullanıcı adı, grubu, dizini gibi bilgileri değiştirebilir.

-c argümanı ile kullanıcıya bir açıklama, -d ile dizin değişikliği ve -s ile shell özelliklerini değiştirdik.

```
root@ubuntu:~# useradd ahmet
root@ubuntu:~# cat /etc/passwd | grep ahmet
ahmet:x:1004:1008::/home/ahmet:/bin/sh
root@ubuntu:~# usermod -c "Backup Admin" -d /home/ahmet2 -s "/bin/bash" ahmet
root@ubuntu:~# cat /etc/passwd | grep ahmet
ahmet:x:1004:1008:Backup Admin:/home/ahmet2:/bin/bash
```

Kullanıcıları gruplara atarken -g argümanı ile birincil grubunu değiştirebilirsiniz. -G argümanı ile gruba atarken birincil grubu sabit kalır ve atamak istediğiniz grup ikincil olarak yer alır.

```
root@ubuntu:~# id aydin
uid=1001(aydin) gid=1004(system) groups=1004(system),1002(yeni)
root@ubuntu:~# usermod -g network aydin
root@ubuntu:~# id aydin
uid=1001(aydin) gid=1005(network) groups=1005(network),1002(yeni)
root@ubuntu:~# usermod -g network -G system aydin
root@ubuntu:~# id aydin
uid=1001(aydin) gid=1005(network) groups=1005(network),1004(system)
```

usermod -l komutu bir kullanıcının ismini değiştirmek için kullanılır. İsim değişikliği yapmadan önce kullanıcının çalıştığı tüm processler durdurulmalıdır. Bu değişikliği yaptığınızda kullanıcı root, ssh erişimi gibi izinleri eski kullanıcı ismi ayarlandığından bu izinleri kaybeder. Servislerin veya yetkilerin yapılandırma dosyalarını tekrar düzenlemek gereklidir.

```
root@ubuntu:~# id hasan
uid=1003(hasan) gid=1007(hasan) groups=1007(hasan)
root@ubuntu:~# usermod -l hasan2 hasan
root@ubuntu:~# id hasan
id: 'hasan': no such user
root@ubuntu:~# id hasan2
uid=1003(hasan2) gid=1007(hasan) groups=1007(hasan)
```

Kullanıcı hesabını kilitlemek için -L , kilidi kaldırmak için -U argümanını kullanabilirsiniz.

```
root@ubuntu:~# usermod -L ahmet
root@ubuntu:~# cat /etc/shadow | grep -i ahmet
ahmet:$6$rc9YEEWtGDynKAuV$1BIixleBiqAhcb.lbndpsGM/
0:99999:7:::
root@ubuntu:~# usermod -U ahmet
root@ubuntu:~# cat /etc/shadow | grep -i ahmet
ahmet:$6$rc9YEEWtGDynKAuV$1BIixleBiqAhcb.lbndpsGM/.r
:99999:7:::
root@ubuntu:~#
```

Sudo – Sudoers File

sudo komutu bir komutu execute etme yetkisi ile (kill, mount, adduser gibi) çalıştırılmak için kullanılır. Bu komut ile root olarak sisteme login olmanızı veya root kullanıcısının sisteme login olması için erişim izini vermenize gerek kalmaz.

Ancak sudo komutunu normal bir kullanıcı kullanırsa komut hata vererek işlem gerçekleşmeyecek ve bu işlem sistem loglarına düşecektir. Bunun nedeni kullanıcının root yetkisine sahip olmamasıdır.

```
$ whoami
hasan
$ apt update
Reading package lists... Done
E: Could not open lock file /var/lib/apt/lists/lock - open (13: Permission denied)
E: Unable to lock directory /var/lib/apt/lists/
W: Problem unlinking the file /var/cache/apt/pkgcache.bin - RemoveCaches (13: Permission denied)
W: Problem unlinking the file /var/cache/apt/srcpkgcache.bin - RemoveCaches (13: Permission denied)
```

Bir kullanıcı sudo komutunu girdiğinde sistem sudoers dosyasını kontrol ederek komutu çalıştırılan kullanıcının yetkilerine bakar. Eğer yetki varsa parola sorar ve işlemi gerçekleştirir.

```
ozgur@ubuntu:~$ sudo apt update
[sudo] password for ozgur:
Hit:1 http://tr.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://tr.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://tr.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:4 http://tr.archive.ubuntu.com/ubuntu focal-security InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
89 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

Sudo ile bir kullanıcının işlem yapabilmesi için sudoers dosyası içinden kullanıcıya yetki verilmelidir. Kısaca bu dosya kullanıcının çalıştırabileceği komutların listesidir.

Sudoers içindeki tanımlamalar Alias'lar yardımıyla yapılır. Alias'lar birden çok girdiyi depolayabilen değişkenlerdir. User, Runas, Host ve Cmnd Alias olarak 4 çeşitlerdir.

User_Alias SYSTEM = hasan, ahmet veya User_Alias SOFTWARE = mehmet, aylin

Runas_Alias OPR = root, admin veya Runas_Alias DB = mysql, sybase

Host_Alias ITNET = 192.168.1.0/24 veya Host_Alias CNET = 192.168.2.0/25

Cmnd_Alias DISK = /usr/bin/mount, /usr/sbin/fdisk, /usr/sbin/mkfs

Sudoers dosyasında olup olmadığını ve çalıştırabileceğiniz komutları kontrol etmek için **sudo -l** komutu kullanılır.

```

ozgur@test:~$ sudo -l
Matching Defaults entries for ozgur on test:
  env_reset, mail_badpass,
  secure_path=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:snap/bin

User ozgur may run the following commands on test:
  (ALL : ALL)  ALL

```

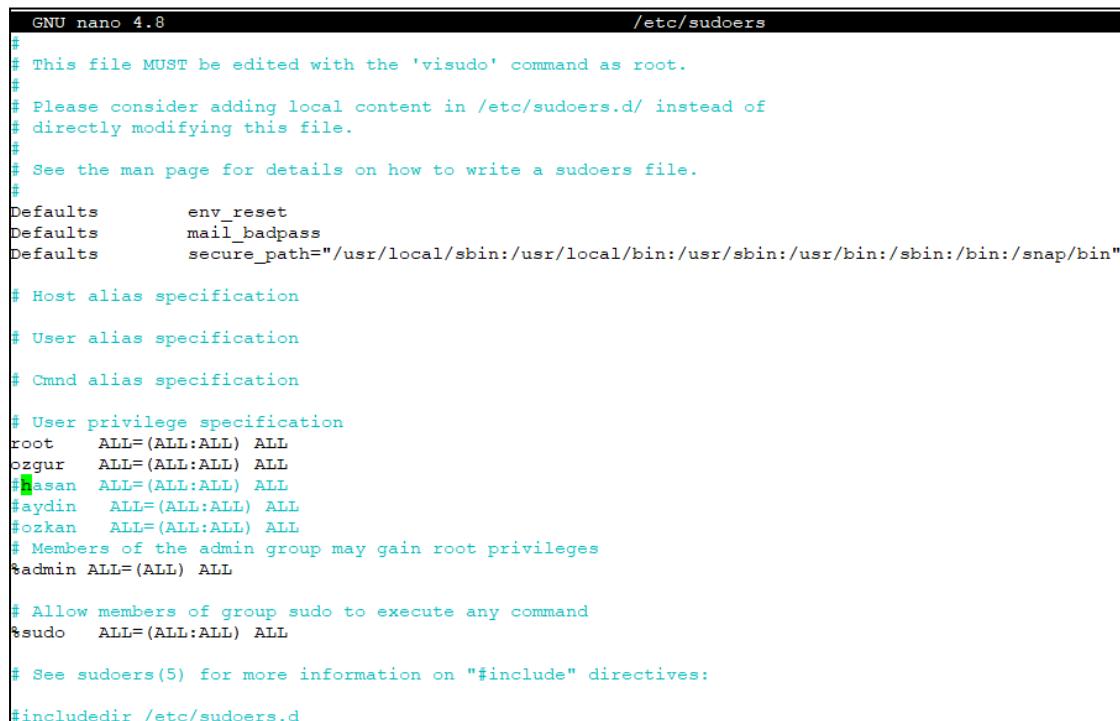
Eğer kullanıcı sudoers içinde değilse bir uyarı mesajıyla komut sonlandırılır.

```

$ sudo -l
[sudo] password for ozkan:
Sorry, user ozkan may not run sudo on test.

```

sudoers dosyasına erişim sağlamak için **sudo visudo** veya **nano /etc/sudoers** komutlarını kullanabilirsiniz. Bu dosyada basit olarak ozgur kullanıcısına bütün yetkileri verdik.



```

GNU nano 4.8                               /etc/sudoers
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL
ozgur   ALL=(ALL:ALL) ALL
#asan   ALL=(ALL:ALL) ALL
#aydin  ALL=(ALL:ALL) ALL
#ozkan  ALL=(ALL:ALL) ALL
# Members of the admin group may gain root privileges
%admin  ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:
#include /etc/sudoers.d

```

root	ALL=(ALL:ALL)	ALL – Kullanıcı adı
root	ALL=(ALL:ALL)	ALL – Sunucunun tamamındaki yetkilendirme
root	ALL=(ALL:ALL)	ALL – Kullanıcıları tanımlar.
root	ALL=(ALL:ALL)	ALL – Grupları tanımlar.
root	ALL=(ALL:ALL)	ALL – Komutlarının tamamını tanımlar.

Sudoers içindeki satırı yorumlayacak olursak root kullanıcısı hostun tamamında bütün komutları çalıştırır.

Aliasları satır içindeki 4 alana tanımlayarak izinler sağlayabiliriz. Örneği sistem departmanı için ADMIN isimli bir Alias oluşturduk. Komutlar içinde SRV isimli bir Alias oluşturduk. İzin tanımlayacağımız bölümde kullanıcı kısmına ADMIN ve komut kısmına SRV yazarsak yazdığımız kural ADMIN Alias'ı içindeki tüm kullanıcılar için geçerli olarak SRV içindeki komutları çalıştırmasını sağlayacaktır.

```
Cmnd_Alias SRV = /sbin/service
User_Alias ADMIN = ozkan

ADMIN ALL=(ALL) SRV
```

Sudoers dosyası içinde en alt satırda %wheel ALL=(ALL:ALL) NOPASSWD:

ALL şeklinde bir satır eklersek bu grubu atanan kullanıcılar şifre girmeden bütün komutları çalıştırabilir. wheel grubu Centos sunucularda kullanılan bir gruptur. Ancak wheel burada bizim için sadece grubun ismidir. Kullanıcıları sudo grubuna da ekleyebilirsiniz.

```
root@ubuntu:~# sudo usermod -aG sudo ozkan
root@ubuntu:~# id ozkan
uid=1002(ozkan) gid=1005(network) groups=1005(network),27(sudo),1006(devops)
```

Aynı şekilde bir kullanıcının şifre girmemesini istersek kullanıcı satırının en sonundaki ALL'dan önce NOPASSWD: argümanını yazmamız gereklidir.

Kullanıcılara her zaman bütün yetkiler verilmelidir. Sadece belirli kaynaklar için root yetkisi verilebilir. Hasan kullanıcısı normal bir kullanıcı ve sudo netplan apply komutu ile networkü restart etmek istiyorum. Sudoers dosyasında hasan yok şeklinde uyarı alıyorum.

```
$ sudo netplan apply
[sudo] password for hasan:
hasan is not in the sudoers file. This incident will be reported.
```

Ardından hasan kullanıcısını sudo netplan apply komutunu çalıştırması için sudoers dosyasına ekliyoruz.

```
hasan  ALL=(ALL:ALL) NOPASSWD: /usr/sbin/netplan apply
```

Komutu tekrar çalıştırıldığımızda hasan kullanıcısı şifre girmeden servisi restart edebiliyor.

```
$ sudo netplan apply
[sudo] password for hasan:
hasan is not in the sudoers file. This incident will be reported.
$ sudo netplan apply
$
```

hasan ALL=(ALL:ALL) /usr/bin/mount, /usr/sbin/ifconfig gibi bir satırla sadece belirli komutlar için kullanıcılara izinler tanımlayabiliyoruz.

```
root    ALL=(ALL:ALL) ALL
ozkan   ALL=(ALL:ALL) /sbin/fdisk
ahmet   ALL=(ALL:ALL) /usr/bin/systemctl restart sshd
```

Sudo -l komutıyla ozkan kullanıcısının yetkilerine baktığımızda sadece fdisk komutunu çalıştırabileceği görünüyor.

```
$ sudo -l
Matching Defaults entries for ozkan on test:
  env_reset, mail_badpass,
  secure_path=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin

User ozkan may run the following commands on test:
  (ALL : ALL) /sbin/fdisk
```

Yetki verilen komutları çalıştırmadan önce sudoers içerisinde kullanıcı satırındaki komut tanımlarının başına NOPASSWD: yazarsak kullanıcı şifre girmeden komutları çalıştırabilir.

8 - İŞLEM(PROCESS) YÖNETİMİ

İşlem Yönetimini Anlama -ps

Linux sunucuda iki tip işlem gerçekleşir. Bunlardan bir tanesi shell'den (komut satırından) kullanıcı tarafından komutlarla başlatılan işlemler diğeri ise sunucu başladığında root yetkisiyle arka planda çalışan işlemlerdir. Bir işlem başlatıldığında aynı anda bu işleme bağlı bir çok iş parçacığı (thread) çalışmaya başlar. Linux sistemler bu işlemleri ve iş parçacıkları yönetmek için çeşitli araçlara sahiptir.

Bu araçlarla hem işlemlerin takibi yapılırken hem de kullandığı kaynakların durumları gözlemlenir. Kritik durumlarda işlemlere müdahale edilerek sunucunun stabil çalışması sağlanır.

İşlemlerin takibi için **ps (Process)** komutu kullanılır. **ps** aldığı argümanlarla çeşitli çıktılar vererek sistem yöneticilerine kolaylıklar sağlar.

ps -a komutu ile sunucuya yapılan tüm bağlantılar görüntülenir.

```
ozgur@ubuntu:~$ ps -a
 PID TTY          TIME CMD
 901 pts/1    00:00:00 bash
 1419 pts/0    00:00:00 ps
```

ps -e komutu işlem listesi gösterilir.

```
ozgur@ubuntu:~$ ps -e
 PID TTY          TIME CMD
  1 ?        00:00:07 systemd
  2 ?        00:00:00 kthreadd
  3 ?        00:00:00 rcu_gp
  4 ?        00:00:00 rcu_par_gp
  6 ?        00:00:00 kworker/0:0H-kblockd
```

ps -aux işlem listesini detaylandırır.

```
ozgur@ubuntu:~$ ps -aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START  TIME COMMAND
root      1  0.0  1.2 103348 12856 ?        Ss   12:54  0:07 /sbin/init maybe-ubiquity
root      2  0.0  0.0     0     0 ?        S    12:54  0:00 [kthreadd]
root      3  0.0  0.0     0     0 ?        I<   12:54  0:00 [rcu_gp]
root      4  0.0  0.0     0     0 ?        I<   12:54  0:00 [rcu_par_gp]
root      6  0.0  0.0     0     0 ?        I<   12:54  0:00 [kworker/0:0H-kblockd]
```

USER : İşlemi çalıştırınan kullanıcıyı gösterir.

PID (Process ID) : İşlemenin ID'sidir. İşlemi durdurup, başlatmak ve takip etmek için bu ID kullanılır.

%CPU : İşlemci kullanım miktarını gösterir.

%MEM : Bellek kullanım miktarını gösterir.

VSZ (Virtual Memory Size) : İşlemenin erişebileceğini bellek miktarıdır.

RSS (Resident Set Size) : İşleme atanan bellek miktarıdır.

TTY : İşlemenin gerçekleştiği terminal tipini gösterir.

STAT (Status) : İşlemenin durumunu gösterir.

TIME : İşlemenin CPU kullanım süresini gösterir.

COMMAND : İşlemenin kendisini gösterir.

ps -ao user,comm,pid,%cpu komutunda -o argümanı ile ekran çıktısı özelleştirilir.

```
ozgur@ubuntu:~$ ps -ao user,comm,pid,%cpu
USER      COMMAND          PID %CPU
ozgur    bash            901  0.0
ozgur    ps             1428  0.0
ozgur@ubuntu:~$ ^C
ozgur@ubuntu:~$
```

ps -ef|less komutuyla bir PID'nin Parent PID numarasını görebilirsiniz.

```
ozgur@ubuntu:~$ ps -ef|less
UID      PID  PPID  C STIME TTY          TIME CMD
root      1     0  0 12:54 ?        00:00:07 /sbin/init maybe-ubiquity
root      2     0  0 12:54 ?        00:00:00 [kthreadd]
root      3     2  0 12:54 ?        00:00:00 [rcu_gp]
root      4     2  0 12:54 ?        00:00:00 [rcu_par_gp]
root      6     2  0 12:54 ?        00:00:00 [kworker/0:0H-kblockd]
root      8     2  0 12:54 ?        00:00:00 [mm_percpu_wq]
```

PPID ile izlediğiniz işlemin hangi işlem tarafından çalıştırıldığını görebilirsiniz. Ozgur kullanıcısı 1762 PID'li bir işlem gerçekleştiriyor ve bu işlemin PPID'si 1128. PID'si 1128 olan işlemi takip edersek PPID'si 1127 olan bir işlem tarafından çalıştırıldığı görülüyor. PID'si 1127 olan işlemi takip edersek 1047 PPID'li sshd servisi tarafından çalıştırıldığı görülüyor. Bu şekilde takip edildiğinde PPID'si 1 olan ve root tarafından çalıştırılan sshd servisine ulaşmış oluyoruz. Bu şekilde işlemleri takip ederek hangi servisi hangi servisle ilişkisi olduğunu bulabiliyoruz.

```
root      611      1  0 12:55 ?        00:00:00 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 start
ups
root      616      1  0 12:55 ?        00:00:00 /usr/bin/python3 /usr/share/unattended-upgrades/unat
tended-upgrade-shutdown --wait-for-signal
root      617      1  0 12:55 ?        00:00:00 /usr/lib/polkit-1/polkitd --no-debug
root      624      1  0 12:55 ?        00:00:00 nginx: master process /usr/sbin/nginx -g daemon on;
master_process on;
www-data   625      624  0 12:55 ?        00:00:00 nginx: worker process
ozgur     889      1  0 12:59 ?
ozgur     896     889  0 12:59 ?        00:00:00 (sd-pam)
ozgur     901     606  0 12:59 ttys1    00:00:00 -bash
root     1046      2  0 13:00 ?        00:02:13 [kworker/0:0-events]
root     1047     611  0 13:00 ?        00:00:00 sshd: ozgur [priv]
ozgur    1127     1047  0 13:01 ?        00:00:06 sshd: ozgur@pts/0
ozgur    1128     1127  0 13:01 pts/0    00:00:00 -bash
root     1733      2  0 17:28 ?        00:00:00 [kworker/0:2]
root     1746      2  0 17:47 ?        00:00:00 [kworker/u2:2-events_unbound]
root     1747      2  0 17:53 ?        00:00:00 [kworker/u2:1-events_power_efficient]
root     1760      2  0 17:58 ?        00:00:00 [kworker/u2:0-events_power_efficient]
ozgur    1761     1128  0 17:58 pts/0    00:00:00 ps -ef
ozgur    1762     1128  0 17:58 pts/0    00:00:00 less
```

ps aux --sort=-%pid komutu ile çıktıları belirli bir sıraya göre listeleyebilirsiniz. Örnek çıktıkla işlem sırasına göre bir çıktı aldık. Ancak **-%cpu** veya **-%mem** değerlerine göre alacağımız çıktılarla kaynakları en çok hangi uygulamaların tükettiğini görebiliriz.

```
ozgur@ubuntu:~$ ps -aux --sort=-pid
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START  TIME COMMAND
ozgur    1706  0.0  0.3  9216  3676 pts/0    R+   17:19  0:00 ps -aux --sort=-pid
ozgur    1681  0.0  0.5  8276  5296 pts/1    Ss+  17:17  0:00 -bash
ozgur    1680  0.0  0.6  13920  6152 ?       S    17:17  0:00 sshd: ozgur@pts/1
root    1535  0.0  0.9  13920  9148 ?       Ss   17:16  0:00 sshd: ozgur [priv]
root    1514  0.0  0.0     0     0 ?       I    17:13  0:00 [kworker/u2:1-events_power_efficient]
root    1512  0.0  0.0     0     0 ?       I    17:03  0:00 [kworker/u2:0-events_unbound]
```

ps aux | grep ozgur komutu ile bir kullanıcının çalıştırduğu işlemleri takip edebiliriz.

```
ozgur@ubuntu:~$ ps aux | grep ozgur
ozgur      889  0.0  0.9  18668  9904 ?          Ss   12:59  0:00 /lib/systemd/systemd --user
ozgur      896  0.0  0.3 103212  3380 ?          S    12:59  0:00 (sd-pam)
ozgur      901  0.0  0.5   8264  5284 tty1        S+   12:59  0:00 -bash
root     1047  0.0  0.9  13928  9104 ?          Ss   13:00  0:00 sshd: ozgur [priv]
ozgur     1127  0.0  0.6  13928  6096 ?          S    13:01  0:06 sshd: ozgur@pts/0
ozgur     1128  0.0  0.5   8408  5536 pts/0       Ss   13:01  0:00 -bash
ozgur     1765  0.0  0.3   8876  3504 pts/0       R+   18:09  0:00 ps aux
ozgur     1766  0.0  0.0   6432   736 pts/0       S+   18:09  0:00 grep --color=auto ozgur
```

STAT kolonundaki harf değerleri işlemin durumu hakkında bize bilgi verir.

D Uykuda olan işlemler

R Çalışan İşlemler

S Çalışmak için bir işlemin tamamlanmasını beklerken uykuda olan işlemler

T Durmuş olan işlemler

X Ölmüş olan işlemler

Z Zombi işlemler

STAT kolunundaki harflerin yanındaki ikinci değerler (BSD Format)

< Öncelikli İşlem

N Düşük öncelikli İşlem

s Oturum lideridir. Oturumdaki ilk süreç anlamına da gelir.

I Çoklu işlem

+ Ön yüzde çalışan işlem.

Adı geçmişen Zombie Process, yürütülmesi tamamlanan ancak hala bir giriş için bekleyen ve sistem tarafından sonlandırılamayan işlemlerdir. Eğer zombie process'ler sisteminizde fazla ise bu diğer process'lerin çalışmasına engel olacaktır. Bu nedenle zombie process'lerin tespit edilip kill -s SIGCHLD PID ile öldürülmeleri gereklidir.

Foreground ve Background İşlemleri

Kullanıcı tarafından çalıştırılan her işlem **foreground** işlemidir. Klavyeden komutu gönderip ekrandan çıktıları alıp ön yüzde gerçekleştirdiğimiz işlemler de diyebiliriz.

Background, kullanıcı tarafından komut satırında herhangi bir işlem yapmadan çalışan işlemlerdir. Eğer bir background işleminde klavyede bir işlem gerçekleşmesi gerekiyorsa bekler. Background işlemlerin avantajı arka planda çalışarak kullanıcının başka bir komut girmek için işlemin tamamlanmasını beklememesidir.

Bir işlemi arka planda çalıştırmak için komut sonuna & işareteti koyulur. & işareteti ile bir komutu çalıştırıldığınızda çıktı olarak bir PID verilir. Bu PID ile işlemi takip edebilir veya **fg** komutuyla işlemi ön yüze alabilirsiniz.

Örneğimizde test.txt isimli dosyaya her saniyede bir Linux yazması için & işareteti ile arka planda bir döngü başlattık. Ekrana, bu işlemin PID numarası yazıldı. İşlemi **fg (foreground)** komutu ile önyüze aldık. Bu işlem devam ederken aynı oturumla herhangi bir işlem yapmamız mümkün değil. Ctrl+Z ile işlemi durduruyorum. ps aux komutuyla 1799 PID'li işlemin durumuna baktığımızda S+ ibaresini görüyoruz. Bu, ön yüzde çalışıp uykuda olan bir

İşlem anlamındadır. fg komutu ile tekrar işlemi ön yüze alıp Ctrl+C ile işlemi tamamen durduruyorum. Jobs komutu ile kontrol ettiğimde arka planda çalışan bir işin olmadığı görülüyor.

```
ozgur@ubuntu:~$ (while true; do echo -n "Linux " >> test.txt; sleep 1; done) &
[1] 1799
ozgur@ubuntu:~$ fg
( while true; do
    echo -n "Linux " >> test.txt; sleep 1;
done )
^Z
[1]+  Stopped                  ( while true; do
    echo -n "Linux " >> test.txt; sleep 1;
done )
ozgur@ubuntu:~$ ps -aux | grep 1799
ozgur      1799  0.0  0.3  8408  3572 pts/0      T     18:21   0:00 -bash
ozgur      1812  0.0  0.0   6300   736 pts/0      S+    18:22   0:00 grep --color=auto 1799
ozgur@ubuntu:~$ fg
( while true; do
    echo -n "Linux " >> test.txt; sleep 1;
done )
^C
ozgur@ubuntu:~$ jobs
```

İşlem Önceliğini Ayarlama - nice, renice

Çalışan işlemleri sınırlayabilir veya diğer işlemlere göre önceliklendirerek kaynak kullanımlarını daha etkin hale getirebilirsiniz. İşlemler, öncelik değerleri 0 olarak çalıştırılırlar. nice ile işlemleri çalıştırmadan önce maksimum -20 (en yüksek öncelik değeri) ile minimum 19 (en düşük öncelik değeri) arasında öncelik değeri verebilirsiniz.

sudo nice -n 7 dd if=/dev/zero of=/dev/null

sudo renice 11 -p 1939

Örneğimizde işlem önceliği 7 olan dd ile diske boş veri yazılması için bir işlem başlattık. Ardından renice ile işlem önceliğini değiştirdik.

```
ozgur@ubuntu:~$ sudo nice -n 7 dd if=/dev/zero of=/dev/null &
[6] 1938
ozgur@ubuntu:~$ ps -all
F S  UID      PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
4 S  1000     901    606  0 80    0 - 2066 poll_s  tty1      00:00:00 bash
0 S  1000    1895   1128  0 80    0 - 1369 hrtme  pts/0      00:00:00 sleep
4 S    0    1898   1128  0 80    0 - 2311 -       pts/0      00:00:00 sudo
0 S  1000    1899   1128  0 80    0 - 1369 hrtme  pts/0      00:00:00 sleep
4 R    0    1900   1898  81 85    5 - 1380 -       pts/0      00:05:25 dd
4 S    0    1938   1128  0 80    0 - 2311 -       pts/0      00:00:00 sudo
4 R    0    1939   1938  33 87    7 - 1380 -       pts/0      00:00:01 dd
0 R  1000    1940   1128  0 80    0 - 2199 -       pts/0      00:00:00 ps
ozgur@ubuntu:~$ sudo renice 11 -p 1939
1939 (process ID) old priority 7, new priority 11
ozgur@ubuntu:~$ ps -all
F S  UID      PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
4 S  1000     901    606  0 80    0 - 2066 poll_s  tty1      00:00:00 bash
0 S  1000    1895   1128  0 80    0 - 1369 hrtme  pts/0      00:00:00 sleep
4 S    0    1898   1128  0 80    0 - 2311 -       pts/0      00:00:00 sudo
0 S  1000    1899   1128  0 80    0 - 1369 hrtme  pts/0      00:00:00 sleep
4 R    0    1900   1898  80 85    5 - 1380 -       pts/0      00:05:44 dd
4 S    0    1938   1128  0 80    0 - 2311 -       pts/0      00:00:00 sudo
4 R    0    1939   1938  36 91   11 - 1380 -       pts/0      00:00:12 dd
0 R  1000    1943   1128  0 80    0 - 2199 -       pts/0      00:00:00 ps
```

İşlem Sonlandırma - kill, killall

kill komutu ile çalışan işlemleri sonlandırmak için kullanılır. Sahibi olduğunuz işlemleri direk **kill** komutu ile sonlandırırken diğer işlemler için root yetkisine sahip bir kullanıcı olmanız gereklidir.

kill 1791

kill -9 1791

Örneğimizde dd komutuyla başlattığımız işlemin PID sini bularak kill komutuyla sonlandırdık.

```
ozgur@ubuntu:~$ ps -aux | grep dd
root      2  0.0  0.0    0   0 ?          S   12:54  0:00 [kthreadd]
root     94  0.0  0.0    0   0 ?          I<  12:54  0:00 [ipv6_addrconf]
message+ 555  0.0  0.4  7584  4884 ?          Ss  12:55  0:00 /usr/bin/dbus-daemon
ss=systemd: --nofork --nopidfile --systemd-activation --syslog-only
root    1898  0.0  0.4  9244  4704 pts/0    S   19:06  0:00 sudo nice -n 5 dd if=/dev/zero of=/dev/null
root    1900 68.1  0.0  5520   784 pts/0    RN   19:06 26:10 dd if=/dev/zero of=/dev/null
ozgur    1972  0.0  0.0  6432   740 pts/0    S+  19:45  0:00 grep --color=auto dd
ozgur@ubuntu:~$ sudo kill 1900
[sudo] password for ozgur:
ozgur@ubuntu:~$ ps -aux | grep dd
root      2  0.0  0.0    0   0 ?          S   12:54  0:00 [kthreadd]
root     94  0.0  0.0    0   0 ?          I<  12:54  0:00 [ipv6_addrconf]
message+ 555  0.0  0.4  7584  4884 ?          Ss  12:55  0:00 /usr/bin/dbus-daemon
ss=systemd: --nofork --nopidfile --systemd-activation --syslog-only
ozgur    1976  0.0  0.0  6432   736 pts/0    S+  19:45  0:00 grep --color=auto dd
[4]-  Terminated                  sudo nice -n 5 dd if=/dev/zero of=/dev/null
```

Bazen programlar kill komutunu engeller veya kabul etmezler. Bu durumlarda kill komutuna, görevi yerine getirecek sinyaller eklenir.

kill -15 Bu komut SIGTERM sinyali göndererek verilerin diske yazılıp bellek temizlendikten sonra işlemin (PID) temiz bir şekilde kapatılmasını söyler.

kill -1 Bu komut SIGHUP sinyali göndererek SIGTERM işleminde olduğu gibi işlemin sonlanmasını ve işlemin restart edilmesini söyler.

kill -2 Bu komut SIGINT sinyali gönderir. Ctrl+C kombinasyonun eşit bir sinyal göndererek işlemi sonlandırmaya çalışır.

kill -11 Bu komut SIGSEGV sinyali göndererek işlemin diske verileri yazmadan sonlanmasına neden olur. Ancak işlemin neden hatalı davranışlığı ile ilgili kayıtları tutan bir dump dosyası oluşturur.

kill -9 Bu komut SIGKILL sinyali göndererek işlemin hiçbir şey yapmadan sonlanmasını sağlar. İşlem sonlandığında işleme ait hiçbir loga (işlem günlüğü) ulaşılabilir.

killall komutu bir kullanıcıya ait tüm işlemleri sonlandırmak için kullanılır.

sudo killall -u username

Örneğimizde aydin kullanıcısının işlemlerini ozgur kullanıcısıyla root yetkisi olarak durduruyoruz.

```

ozgur@ubuntu:~$ ps -aux | grep aydin
root      1989  0.0  0.8  13924  9020 ?
aydin     2001  0.3  0.9  18672  9952 ?
aydin     2004  0.0  0.4 104420  4576 ?
aydin     2137  0.3  0.6  13924  6188 ?
aydin     2138  0.0  0.5   8308  5144 pts/1
aydin     2143  100  0.0   5520   788 pts/1
ozgur    2148  0.0  0.0   6432   736 pts/0
ozgur    2163  0.0  0.0   6432   736 pts/0
ozgur@ubuntu:~$ sudo killall -u aydin
[sudo] password for ozgur:
ozgur@ubuntu:~$ ps -aux | grep aydin
ozgur    2163  0.0  0.0   6432   736 pts/0
ozgur@ubuntu:~$ grep --color=auto aydin

```

9 - SİSTEM PERFORMANSINI GÖRÜNTÜLEME

Sistem performansını takip etmek sistem yöneticileri için önemli bir görevdir. Günümüzde birçok yapı sunucularının performans değerlerini çeşitli monitoring araçlarıyla takip etse de anlık olarak sunucu içerisinde yapmamız gereken kontroller olabilir. Linux sunucular için top aracı en iyi araçlardan biridir.

Performans Ölçüm Araçları – top, free, watch, vmstat, htop, sysstat

top aracı bütün Linux sistemlerin vazgeçilmez performas görüntüleme aracıdır. Top gerçek zamanlı olarak sunucu içindeki bütün olandan bitenden haberdadır.

top aracı adı üstünde top komutu ile çalışır. Top komutu foreground yani ön yüzde çalışan bir komuttur. Top ile performansa bakarken eğer işlem de yapmak isterseniz ikinci bir oturum daha başlatmanız gereklidir. Top işlem listesini ekrana getirirken en çok kaynak kullanımından en az kaynak kullanımına göre işlemleri sıralayıp ekrana getirir.

```

top - 23:39:37 up 1:14, 2 users, load average: 0.00, 0.00, 0.00
Tasks: 99 total, 1 running, 98 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.3 sy, 0.0 ni, 99.3 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st
MiB Mem : 981.3 total, 532.7 free, 140.6 used, 308.0 buff/cache
MiB Swap: 1962.0 total, 1962.0 free, 0.0 used. 692.9 avail Mem

```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1068	root	20	0	0	0	0	I	0.7	0.0	0:24.62	kworker/0:0-events
1042	ozgur	20	0	13928	6220	4732	S	0.3	0.6	0:08.30	sshd
1	root	20	0	102000	11760	8668	S	0.0	1.2	0:05.95	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd

1 ncı sırada sunucunun ne zaman açıldığı ve ne kadar süredir açık olduğu görülür. Hemen yanında sisteme login olmuş kullanıcı sayısı ve load average bilgileri yer alır.

Load average sırasıyla sunucunun son 1, 5 ve 15 dakikadaki yüzde cinsinde yük miktarını gösterir. Soldan başlayacak olursak ortalama yük miktarı aşağıdaki gibi olur.

Son 1 dakikada 0.01 = %1

Son 5 dakikada 0.14 = %14

Son 15 dakikada 0.12 = %12

2 ncı sırada toplam çalışan görev sayısı ve çalışan görevlerin durumları görüntülenir.

3 ncı sırada işlemci performans göstergeleri (indicator) yer alır. Kısalmaların anlamları;

Gösterge	Açıklaması
us	Kullanıcıların çalıştırıldığı işlemlerin yüzdesi
sy	Kernel mode'da sistem çağrıları ve driver erişimleri için harcanan süre.
ni	Öncelik verilen işlemler için harcanan zamanın yüzdesi
id	İşlemcinin boşta geçirdiği sürenin yüzdesi

wa	İşlemcinin disk istekleri için beklediği süredir. Yüksek değerler düşük storage performansını işaret eder.
hi	İşlemcinin donanım kesintileri için harcadığı süredir. Bu değerin yüksek olması donanım tarafında problemlerin olduğunu göstergesidir.
si	İşlemcinin yazılım tarafındaki kesintiler için harcadığı süredir. Bu değerin yüksek olması yazılım tarafında problemlerin olduğunu gösterir.
st	Eğer bir sanallaştırma ortamınız varsa buradaki değer sanal sunucuların kullanım yüzdesini gösterir.

4 ncü ve 5 ncı sıra, sırasıyla fiziksel ve sanal (swap) bellek kullanımılarını gösterir.

top çalıştırktan sonra f tuşuna basarsanız top çıktısındaki işlem görünüm kolonlarını değiştirebilirsiniz. Klavyenin yön tuşları ile göstermek veya kaldırmak istediğiniz alanı boşluk tuşıyla işaretlemeniz yeterlidir. Bu ekranındaki düzenleme tamamen sistem yöneticilerinin kendi ihtiyaçlarına göre yapılmalıdır.

```
Fields Management for window 1:Def, whose current sort field is %CPU
  Navigate with Up/Dn, Right selects for move then <Enter> or Left commits,
  'd' or <Space> toggles display, 's' sets sort. Use 'q' or <Esc> to end!

* PID      = Process Id          CGROUPS = Control Groups
* USER     = Effective User Name SUPGIDS = Supp Groups IDs
* PR       = Priority             SUPGRPS = Supp Groups Names
* NI       = Nice Value           TGID   = Thread Group Id
* VIRT     = Virtual Image (KiB) OOMa   = OOMEM Adjustment
* RES      = Resident Size (KiB) OOMs   = OOMEM Score current
* SHR      = Shared Memory (KiB) ENVIRON = Environment vars
* S        = Process Status       vMj    = Major Faults delta
* %CPU    = CPU Usage             vMn    = Minor Faults delta
* %MEM    = Memory Usage (RES)   USED   = Res+Swap Size (KiB)
* TIME+   = CPU Time, hundredths nsIPC  = IPC namespace Inode
* COMMAND = Command Name/Line   nsMNT  = MNT namespace Inode
* PPID    = Parent Process pid  nsNET   = NET namespace Inode
* UID     = Effective User Id   nsPID   = PID namespace Inode
* RUID    = Real User Id        nsUSER  = USER namespace Inode
* RUSER   = Real User Name      nsUTS   = UTS namespace Inode
```

free komutu bellek kullanımıyla ilgili bilgiler verir. Mem fiziksel bellek, Swap ise Virtual Bellek miktarını gösterir. free komutu ile çeşitli argümanlar kullanarak miktarların farklı ölçü birimleriyle gösterilmesi sağlanabilir.

```
ozgur@ubuntu:~$ free --mega
              total        used        free      shared  buff/cache   available
Mem:         1028         152        559            1        316        721
Swap:        2057          0       2057
ozgur@ubuntu:~$ free -t
              total        used        free      shared  buff/cache   available
Mem:      1004848      145508      550032      1052      309308      708168
Swap:      2009084          0      2009084
Total:     3013932      145508      2559116
ozgur@ubuntu:~$ free -tm
              total        used        free      shared  buff/cache   available
Mem:          981         142         537            1        302        691
Swap:        1961          0       1961
Total:      2943         142        2499
ozgur@ubuntu:~$ free -m
              total        used        free      shared  buff/cache   available
Mem:          981         142         537            1        302        691
Swap:        1961          0       1961
```

watch free komutu da bellek kullanımıyla ilgili bilgiler verir.

Every 2.0s: free							ubuntu: Sat Feb 6 23:22:52 2021						
	total	used	free	shared	buff/cache	available		total	used	free	shared	buff/cache	available
Mem:	1004848	145672	549808	1052	309368	708008							
Swap:	2009084	0	2009084										

vmstat ise diğer bir yararlı komuttur. Sunucunun en son reboot ettiği zamandan itibaren ortalama bellek, I/O ve CPU kullanımı raporlar.

ozgur@ubuntu:~\$ vmstat																
procs -----memory----- ---swap-- -----io---- -system-- -----cpu----																
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa	st
1	0	0	549816	34100	275320	0	0	68	5	35	78	0	2	98	0	0

Ayrıca zaman aralıkları belirleyerek istenilen sayıda örnek toplayıp karşılaştırmalı bir rapor da alabilirsiniz. Vmstat 5 5 in anlamı her 5 saniye bir olmak üzere 5 adet örnek topla.

ozgur@ubuntu:~\$ vmstat 5 5																
procs -----memory----- ---swap-- -----io---- -system-- -----cpu----																
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa	st
2	0	0	542648	34152	280688	0	0	65	5	34	75	0	1	98	0	0
0	0	0	542640	34152	280688	0	0	0	0	20	30	0	0	100	0	0
0	0	0	544152	34152	280688	0	0	0	0	19	29	0	0	100	0	0
0	0	0	544152	34152	280688	0	0	0	0	18	28	0	0	100	0	0
0	0	0	544152	34152	280688	0	0	0	0	19	31	0	0	100	0	0

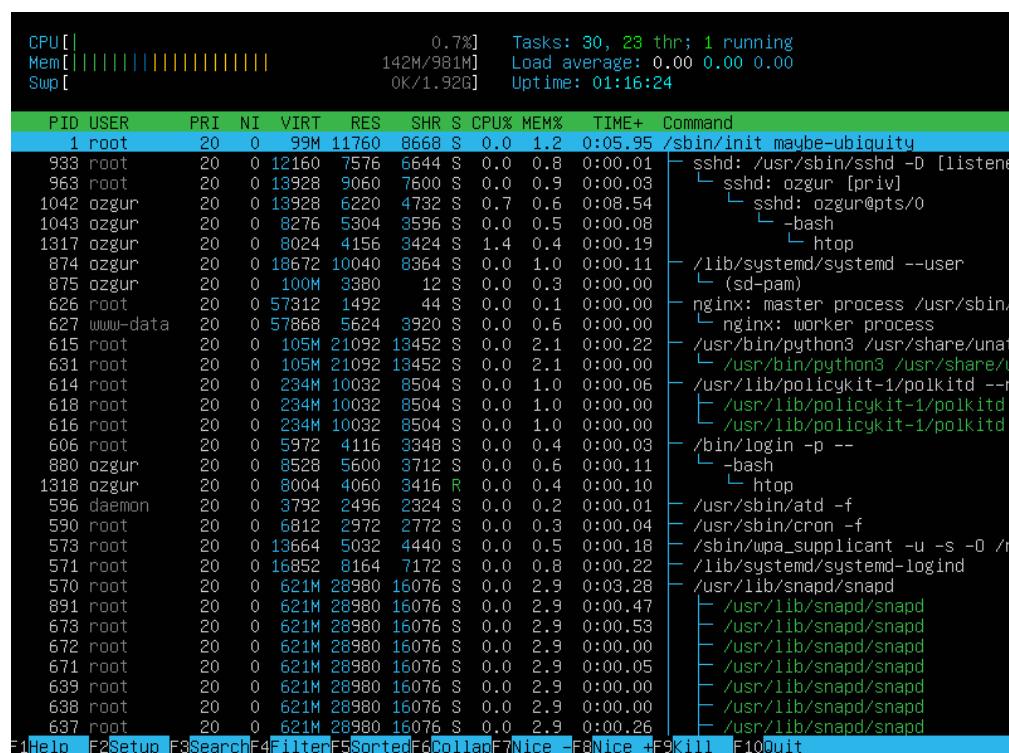
htop ise top 'a nazaran daha renkli ve ek özelliklerini olan bir araçtır. Ancak temel olarak top ile aynı bilgileri verir.

F5 tuşuna bastığınızda işlemlerin ağaç yapısındaki görüntüsünü elde edersiniz. Bu çalışan işlemin kök dizinine kadar gitmenize izin verir.

F6 ile ekranda gösterilen kolonlara göre işlemleri sıraya dizebilirisiniz.

F9 ile belirlediğiniz işlemi sonlandırabilirsiniz.

F10 ile htop'dan çıkışabilirsiniz.



iostat ve pidstat

iostat ve pidstat araçlarını kullanmak için sunucunuzda sysstat aracının yüklü olması gerekiyor. Systat sistem performansını monitor eden shell'de çalışan en iyi monitoring araçlarından bir tanesidir. Disk bölgelerinin giriş çıkış istatistiklerini (I/O), process'lerin kaynak kullanımlarını ve sistemdeki tüm olayları toplayıp raporlayabilir.

Eğer yüklü değilse **apt install sysstat** komutuyla yükleyebilirsiniz.

```
root@ubuntu:~# apt install sysstat
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libsensors-config libsensors5
Suggested packages:
  lm-sensors isag
The following NEW packages will be installed:
```

iostat komutuyla CPU ve disk okuma yazma hızlarını görebiliyoruz. idle bölümü boşta duran CPU kaynağını gösteriyor. Bu sanal sunucu için 2 CPU kaynak kullanmışız. Buradan çıkarımlar kaynak azaltımı yaparak tasarruf sağlayabiliyoruz.

```
root@ubuntu:~# iostat
Linux 5.4.0-65-generic (ubuntu)        02/16/2021      _x86_64_      (2 CPU)

avg-cpu: %user  %nice %system %iowait  %steal   %idle
          0.05    0.03    0.06    0.02    0.00   99.84

Device      tps    kB_read/s    kB_wrtn/s    kB_dscd/s    kB_read    kB_wrtn    kB_dscd
dm-0       0.01     0.24        0.00      45.87      5245         4    996756
fd0        0.00     0.00        0.00       0.00        20         0        0
loop0      0.07     0.08        0.00       0.00      1794         0        0
loop1      0.00     0.02        0.00       0.00      337         0        0
loop2      0.00     0.05        0.00       0.00      1054         0        0
loop3      0.01     0.02        0.00       0.00      408         0        0
loop4      0.00     0.05        0.00       0.00      1095         0        0
loop5      0.00     0.02        0.00       0.00      338         0        0
loop6      0.00     0.00        0.00       0.00        1         0        0
sda        0.58    17.65       7.59     803.86    383478    164828   17467276
sdb        0.01     0.29        0.00      45.87      6305         4    996756
sdc        0.01     0.06        0.00       0.00      1312         0        0
```

pidstat processlerin kaynak kullanımını gösterir.

```
root@ubuntu:~# pidstat
Linux 5.4.0-65-generic (ubuntu)        02/16/2021      _x86_64_      (2 CPU)

11:10:24 AM   UID      PID  %usr %system  %guest   %wait   %CPU   CPU  Command
11:10:24 AM     0        1  0.01   0.05    0.00    0.00   0.05      1  systemd
11:10:24 AM     0        2  0.00   0.00    0.00    0.00   0.00      0  kthreadd
11:10:24 AM     0        9  0.00   0.00    0.00    0.00   0.00      0  ksoftirqd/0
```

-d argümanı her iki stat komutu için belirlenecek zaman ve tekrar kadar sistemden örnek toplar.

iostat -d 20 2 komutuyla 20 saniyede bir 2 örnek toplamış olacağız. Hatta grep ile belirli bir aygit için bu işlemi yapalım. Komutu çalıştırıldıktan sonra başka bir pencerede **dd if=/dev/zero of=/dev/null** komutuyla diskte bir işlem yapalım ve iki örnek arasındaki farkı görelim.

```
root@ubuntu:~# iostat -d 20 3|grep sd
sda        0.56    15.65      6.95    711.37    384250    170688   17467276
sdb        0.01     0.26      0.00    40.59      6305         4    996756
sdc        0.01     0.08      0.00      0.00      1996         0        0
sda        0.10     0.00      0.40      0.00        0         8        0
sdb        0.00     0.00      0.00      0.00        0         0        0
sdc        0.00     0.00      0.00      0.00        0         0        0
```

pidstat -r 5 2 komutu 5 saniye 2 örnek toplayarak ortalamasını alıp ekrana yansıtır.

root@ubuntu:~# pidstat -r 5 2								
Linux 5.4.0-65-generic (ubuntu)			02/16/2021			_x86_64_		(2 CPU)
12:01:28 PM	UID	PID	minflt/s	majflt/s	VSZ	RSS	%MEM	Command
12:01:33 PM	33	3671	12.18	0.00	1211480	5640	0.29	apache2
12:01:33 PM	0	4213	187.23	0.00	11180	5476	0.28	pidstat
12:01:33 PM	UID	PID	minflt/s	majflt/s	VSZ	RSS	%MEM	Command
12:01:38 PM	33	3671	1.60	0.00	1211480	5640	0.29	apache2
12:01:38 PM	0	4213	20.60	0.00	11180	6004	0.30	pidstat
Average:	UID	PID	minflt/s	majflt/s	VSZ	RSS	%MEM	Command
Average:	33	3671	6.89	0.00	1211480	5640	0.29	apache2
Average:	0	4213	104.00	0.00	11180	5740	0.29	pidstat

Komutlar - lspci, lscpu, lshw, lsccsi, lsmem

ls komutunu dizin içeriğini ve detaylarına bakmak için kullandık ancak ls komutu yazıp tab tuşuna iki defa basıldığında aslında yapabildiklerinin daha fazla olduğunu göreceksiniz.

root@ubuntu:~# ls
ls
lsb_release
lsattr
lscpu
lsblk
lshw
lsinitramfs
lslogins
lsns
lspgpot
lsmem
lsmod
lsof
lsusb
lspci

lslogins : Sisteme login olmuş kullanıcıları gösterir.

lsns : Sistem üzerindeki erişilebilir namespace'leri listeler. Konteynirizasyon işleriyle uğraşacak arkadaşlar namespace kavramı çok duyacak.

lscpu : İşlemci mimarisi, modeli, hızı, ailesi, core sayısı, ön belleği gibi detaylı bir çıktı verir.

lsmem : RAM ile ilgili bilgiler verir.

lsof : Sistem açık olan dosya ve işlemleri listelemeye yarar. lsof ile hangi servisleri hangi dosya ve dizinlerle iletişimde olduğu kim tarafından hangi port üzerinden çalıştığını görebiliriz.

lsof -i aktif olan bağlantıları listeler.

root@ubuntu:~# lsof -i								
COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE	NAME
systemd-r	785	systemd-resolve	12u	IPv4	21919	0t0	localhost:domain	
systemd-r	785	systemd-resolve	13u	IPv4	21920	0t0	TCP localhost:domain (LISTEN)	
sshd	1169	root	4u	IPv4	26805	0t0	TCP ubuntu:ssh->192.168.1.37:64892 (ESTABLISHED)	
sshd	1321	ozgur	4u	IPv4	26805	0t0	TCP ubuntu:ssh->192.168.1.37:64892 (ESTABLISHED)	
sshd	2010	root	3u	IPv4	32211	0t0	TCP *:ssh (LISTEN)	
sshd	2010	root	4u	IPv6	32213	0t0	TCP *:ssh (LISTEN)	
apache2	2361	root	4u	IPv6	33142	0t0	TCP *:http (LISTEN)	
apache2	2362	www-data	4u	IPv6	33142	0t0	TCP *:http (LISTEN)	
apache2	2363	www-data	4u	IPv6	33142	0t0	TCP *:http (LISTEN)	

lsof -u root komutuyla root kullanıcısının yaptığı işlemleri gösterir.

lsof +d /var/log/ komutu ile belirtilen dizine erişen servis ve işlemleri listeler.

root@ubuntu:~# lsof +d /var/log/								
COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE	NAME
rsyslogd	2465	syslog	7w	REG	8,51	38264	262297	/var/log/syslog
rsyslogd	2465	syslog	8w	REG	8,51	32041	263791	/var/log/apache.log
rsyslogd	2465	syslog	9w	REG	8,51	20972	264718	/var/log/auth.log

lsof -i :22 komutuyla sunucunun 22 numaralı portunda çalışan servis ve kurulan bağlantıları görebiliriz.

```
root@ubuntu:~# lsof -i :22
COMMAND  PID  USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
sshd    1169  root    4u  IPv4  26805      0t0  TCP  ubuntu:ssh->192.168.1.37:64892 (ESTABLISHED)
sshd    1321 ozgur   4u  IPv4  26805      0t0  TCP  ubuntu:ssh->192.168.1.37:64892 (ESTABLISHED)
sshd    2010  root    3u  IPv4  32211      0t0  TCP  *:ssh (LISTEN)
sshd    2010  root    4u  IPv6  32213      0t0  TCP  *:ssh (LISTEN)
```

lsof -i @192.168.1.37 belirttiğiniz IP'nin sunucuda eriştiği servisleri görebilirsiniz.

```
root@ubuntu:~# lsof -i @192.168.1.37
COMMAND  PID  USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
sshd    1169  root    4u  IPv4  26805      0t0  TCP  ubuntu:ssh->192.168.1.37:64892 (ESTABLISHED)
sshd    1321 ozgur   4u  IPv4  26805      0t0  TCP  ubuntu:ssh->192.168.1.37:64892 (ESTABLISHED)
```

Sunucunuza bağladığınız yeni donanınızı aşağıdaki komutları kullanarak sisteme doğru şekilde bağlanıp bağlanmadığını görebilirsiniz.

lshw : Sunucu donanımı hakkında detaylı bilgiler verir.

lspci : PCI veriyolu üzerinden bağlı aygıtlar hakkında bilgi verir.

Cockpit Kullanımı

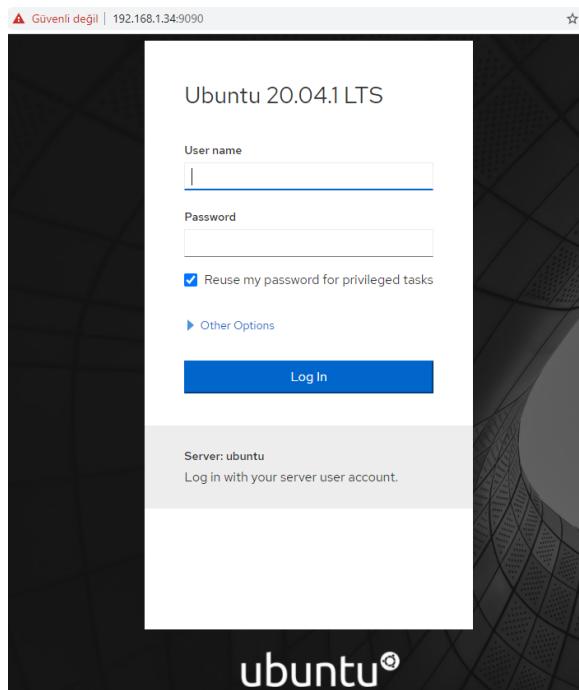
Cockpit, linux (Redhat, Centos, Ubuntu, Debian, CoreOS, Fedora) sunucuları yönetmek için web tabanlı bir arayüzdür. Sistem kaynaklarının izlenmesine ve yapılacak konfigürasyonun kolaylıkla ayarlanması sağlar. Kullanılmadığı zaman herhangi bir kaynak tüketmez ve üzerinde veri barındırmaz.

Cockpit uygulamasını **sudo apt install cockpit -y** komutu ile yükliyoruz.

sudo systemctl start cockpit komutu ile uygulamayı başlatıyoruz.

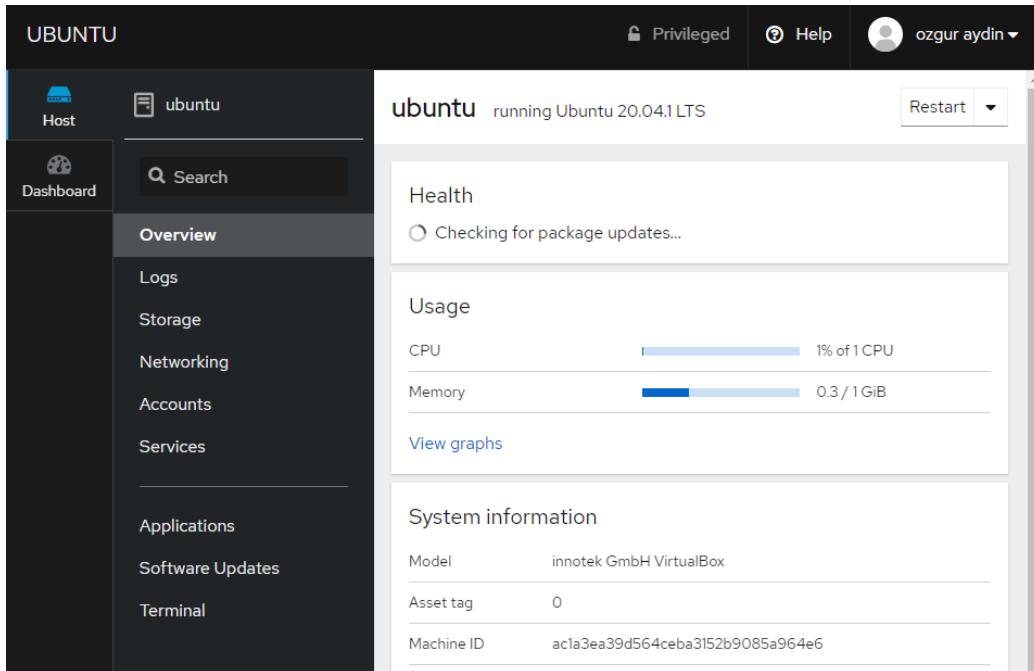
sudo systemctl enable cockpit komutu ile sunucu yeniden başladığında uygulamanın da otomatik olarak başlamasını sağlıyoruz.

Bilgisayarlarınızın browserinden http://sunucu_ipiniz:9090 adresine gidelim.



Sunucu erişim bilgileriniz ile giriş yapın. İlk ekranda kaynak kullanım miktarlarını görürsünüz. Ayrıca view graphs a basarsanız grafiksel olarak kaynak kullanımlarını izlersiniz. Bu ekrandan log, disk alanınız, network ayarlarınız, kullanıcı hesapları, servisler ve updateleri yönetebilirsiniz. Sunucuyu web üzerinden restart ve shutdown edebilirsiniz.

Terminal ile web üzerinden sunucuya komut satırıyla erişim sağlayabilirsiniz. Network ve firewall ayarlarını yaparsanız bu ekrana yönetiminizdeki diğer sunucuları da bağlayabilirsiniz.



Web üzerindeki terminal ekranından erişim sağladığımızda bağlantı tipimizin web cons olduğunu görebilirsiniz.

```
ozgur@ubuntu:~$ w
23:54:22 up 1:29, 3 users,  load average: 0.23, 0.15, 0.08
USER   TTY      FROM           LOGIN@    IDLE   JCPU   PCPU WHAT
ozgur   tty1     -          22:26  13:00  4.49s  4.35s htop
ozgur   pts/0    192.168.1.38  22:30  7:42   0.10s  0.10s -bash
ozgur   web cons -          23:49  0.00s  0.00s  0.01s /usr/lib/
```

10 - Zaman Senkronizasyonu

NTP Nedir?

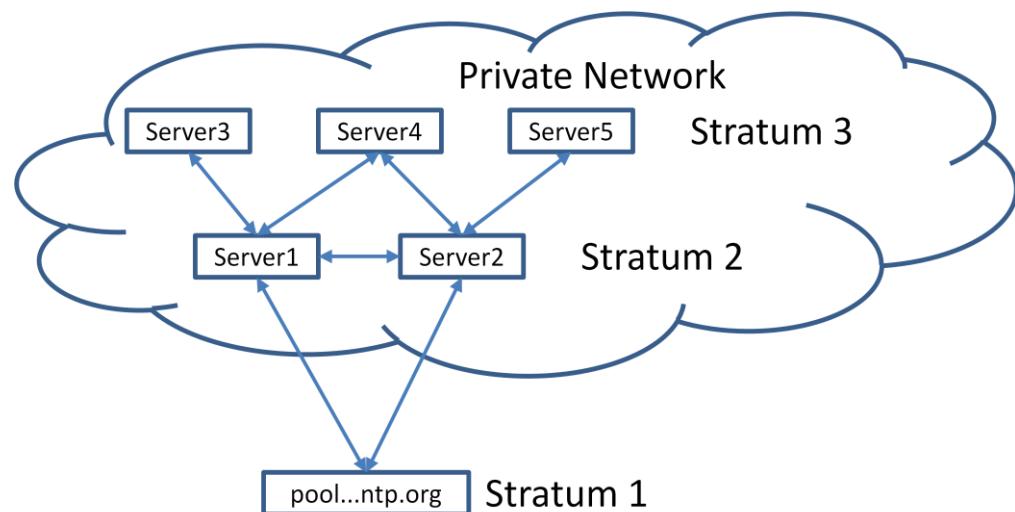
The Network Time Protocol (NTP), bilgisayar sistemi saatini bir ağ üzerinden otomatik olarak senkronize etmek için kullanılan bir protokoldür. Bir sunucuda zamanın doğru şekilde senkronize edilmesi önemlidir. Farklı sunucularda hizmetlerle birlikte çalışırken zaman damgaları hayatı bir rol oynar. Düzungün bir şekilde senkronize edilmemiş bir sunucuda özellikle veritabanı servisleri doğru çalışmamayacaktır. Kimlik doğrulama işlemlerinde zamanın doğru senkronize edilmemesi kullanıcıların sisteme erişmelerinde problem yaşamalarına neden olacaktır. Sistem loglarının yanlış zaman ile damgalanması, logların anlamalarını yitirmesi ile sonuçlanabilir.

Zamanı senkronizasyonu için sunucular harici bir NTP (Network Time Protocol) serverdan zaman bilgisini öğrenmeleri için yapılandırılırlar. En güvenilir zaman kaynağı atomik saatlerdir. Güvenilirliği tarif ederken katman kavramından bahsetmek yerinde olur. Sunucular zamanı öğrenmek için her zaman ana kaynağa gidemezler. Zamanı ilk katmandan öğrenen

sunucuların zaman bilgisi diğer sunuculara göre daha doğru olacaktır. Diğer katmanlarda olan sunucular, zamanı öğrenmek için daha fazla katman geçiklerinden ve bu geçişlerinde çok küçükte olsa zaman harcadıklarından dolayı kaynak ile aralarında zaman farkı oluşacaktır.

Örneğin bankacılık sisteminde 1 saniye içinde onbinlerce işlem yapılıyor. Bu işlemlerde oluşacak salise farkları bile çok büyük problemlere neden olacaktır.

Konumuza ilk olarak date, hwclock ve timedatectl araçlarının kullanımından bahsettiğinden sonra NTP server ve client yapılandırmalarıyla açıkkık getirmeye çalışacağız.



Komutlar - date, hwclock

date, sunucunun tarih ve zaman bilgisini gösterir. Aldığı argümanlarla farklı kullanım şekilleri vardır.

```
root@ubuntu:~# date
Sun 07 Feb 2021 07:35:37 PM UTC
```

sudo date -s komutuyla zamanı değiştirebilirsiniz.

```
root@ubuntu:~# date -s "07 FEB 2021 23:30:00"
Sun 07 Feb 2021 11:30:00 PM UTC
```

```
root@ubuntu:~# date +"%d-%m-%y"
28-03-21
```

```
root@ubuntu:~# date +"%d-%m-%y"/"%T"
28-03-21/23:16:59
```

hwclock, komutu bir diğer zaman ile ilgili işlemleri gerçekleştirebileceğimiz araçtır. --set argümanı ile saatı ayarlabilirsiniz.

```
root@ubuntu:~# hwclock
2021-02-07 23:51:38.741992+03:00
root@ubuntu:~# hwclock --set --date="2021-02-07 23:50:06"
```

Timezone'u değiştirmenin en garanti yolu bağlı olduğunuz bölgenin timezone dosyası ile localtime dosyanızı değiştirmek.

```
root@ubuntu:~# cp /usr/share/zoneinfo/Europe/Istanbul /etc/localtime
root@ubuntu:~# date
Sun 07 Feb 2021 09:11:34 PM +03
```

Komutlar - timedatectl

timedatectl zaman hakkında date komutuna göre daha detaylı bilgi verir ve zaman bilgisini yönetmek için diğer komutlara göre daha kullanışlıdır. Eğer sunucunuzda bir NTP servisi varsa zamanı bu servis ile senkronize eder.

timedatectl status

```
root@ubuntu:~# timedatectl status
          Local time: Sun 2021-02-07 19:36:06 UTC
          Universal time: Sun 2021-02-07 19:36:06 UTC
                  RTC time: Sun 2021-02-07 19:36:07
                  Time zone: Etc/UTC (UTC, +0000)
System clock synchronized: yes
          NTP service: active
      RTC in local TZ: no
```

Zamanı değiştirmek için NTP servisini durdurmanız gereklidir.

timedatectl set-ntp false

```
root@ubuntu:~# timedatectl set-ntp false
root@ubuntu:~# timedatectl
          Local time: Sun 2021-02-07 21:26:28 +03
          Universal time: Sun 2021-02-07 18:26:28 UTC
                  RTC time: Sun 2021-02-07 21:00:29
                  Time zone: Etc/UTC (+03, +0300)
System clock synchronized: no
          NTP service: inactive
      RTC in local TZ: no
```

Zamanı ayarlamak için set-time parametresi kullanılır.

timedatectl set-time "2021-01-31 17:30:00"

```
root@ubuntu:~# timedatectl set-time "2021-01-31 17:30:00"
root@ubuntu:~# timedatectl
          Local time: Sun 2021-01-31 17:30:02 +03
          Universal time: Sun 2021-01-31 14:30:02 UTC
                  RTC time: Sun 2021-01-31 14:30:03
                  Time zone: Etc/UTC (+03, +0300)
System clock synchronized: no
          NTP service: inactive
      RTC in local TZ: no
```

Time zone ları bulmak için list-timezones parametresi kullanılır.

timedatectl list-timezones | grep Ist

```
root@ubuntu:~# timedatectl list-timezones | grep Ist
Europe/Istanbul
```

Time zone değiştirmek için set-timezone parametresi kullanılır.

timedatectl set-timezone Europe/Istanbul

```
root@ubuntu:~# timedatectl set-timezone Europe/Istanbul
root@ubuntu:~# timedatectl
        Local time: Sun 2021-01-31 17:32:04 +03
        Universal time: Sun 2021-01-31 14:32:04 UTC
                  RTC time: Sun 2021-01-31 14:32:05
                 Time zone: Europe/Istanbul (+03, +0300)
System clock synchronized: no
          NTP service: inactive
      RTC in local TZ: no
```

Lokal zamana göre değişiklik yapmak için set-local-rtc parametresi kullanılır. Tekrar UTC (Universal Time) olarak ayarlamak için 1 değeri 0 yapılarak komut yeniden çalıştırılır.

timedatectl set-local-rtc 1

```
root@ubuntu:~# timedatectl set-local-rtc 1
root@ubuntu:~# timedatectl
        Local time: Mon 2021-02-08 00:20:37 +03
        Universal time: Sun 2021-02-07 21:20:37 UTC
                  RTC time: Mon 2021-02-08 00:20:36
                 Time zone: Europe/Istanbul (+03, +0300)
System clock synchronized: yes
          NTP service: active
      RTC in local TZ: yes

Warning: The system is configured to read the RTC time in the local time zone.
This mode cannot be fully supported. It will create various problems
with time zone changes and daylight saving time adjustments. The RTC
time is never updated, it relies on external facilities to maintain it.
If at all possible, use RTC in UTC by calling
'timedatectl set-local-rtc 0'.
```

Chrony ile NTP Senkranizayonu

NTP server olarak ntpd ve chronyd servisleri yaygın olarak kullanılmaktadır. Her iki serviste aynı işi yapmakla birlik aralarında ufak yapılandırma farklılıklarları vardır. Ancak konunun temeline hakim olan bir sistem yönetici için hangisini kullandığının bir anlamı yoktur. Yapacağımız yapılandırmada Red Hat sunucularda da kullanılan chronyd servisini ele alacağız.

İlk olarak chrony servisini sunucumuza yükliyoruz.

sudo apt install chrony -y

Servis yüklenikten sonra başlatıyoruz.

systemctl start chronyd

Servis durumunu kontrol ediyoruz.

systemctl status chronyd

```
root@ubuntu:~# systemctl status chronyd
● chrony.service - chrony, an NTP client/server
   Loaded: loaded (/lib/systemd/system/chrony.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2021-02-08 01:38:05 +03; 1min 17s ago
```

chronyc komutu ile ikinci bir kontrol yaparak servisin çalışıyor ve kaynaklara bağlı olduğunu doğruluyoruz.

chronyc activity

```
root@ubuntu:~# chronyc activity
200 OK
8 sources online
0 sources offline
0 sources doing burst (return to online)
0 sources doing burst (return to offline)
0 sources with unknown address
```

Zamanı senkronize ettiğimiz NTP serverları ve değerlerini görüntüleyyoruz.

chronyc sources -v

```
root@ubuntu:~# chronyc sources -v
210 Number of sources = 8

.-- Source mode '^' = server, '=' = peer, '#' = local clock.
/- .- Source state '*' = current synced, '+' = combined , '-' = not combined,
| / '?' = unreachable, 'x' = time may be in error, '~' = time too variable.
|| | .- xxxx [ yyyy ] +/- zzzz
|| | | xxxx = adjusted offset,
|| | | | yyyy = measured offset,
|| | | | zzzz = estimated error.
|| | |
|| | |
MS Name/IP address      Stratum Poll Reach LastRx Last sample
-----
^? golem.canonical.com    2    7   340   470  +5059us[+4645us] +/-  79ms
^- alphyn.canonical.com   2    7   301    15  +5707us[+5707us] +/- 134ms
^? pugot.canonical.com    2    7   340   471  +1416us[+1002us] +/-  75ms
^? chilipepper.canonical.com  2    7   340   469  +4201us[+4201us] +/-  79ms
^? time.cloudflare.com     0    7     0     -  +0ns[+0ns] +/-   0ns
^+ ntp1.home4u.ch          2    7   340   472  -1183us[-2316us] +/-  59ms
^+ 195.21.59.161            2    7   340   472  +2172us[+1759us] +/-  80ms
^* ntp2.home4u.ch          1    6   300   471  -2993us[-3406us] +/-  43ms
```

Kaynakların durumlarını görüntüleyelim.

chronyc sourcestats -v

```
root@ubuntu:~# chronyc sourcestats -v
210 Number of sources = 8

      .- Number of sample points in measurement set.
      /   .- Number of residual runs with same sign.
      |   /   .- Length of measurement set (time).
      |   |   /   .- Est. clock freq error (ppm).
      |   |   |   /   .- Est. error in freq.
      |   |   |   |   /   .- Est. offset.
      |   |   |   |   |   |   On the -.
      |   |   |   |   |   |   samples. \
      |   |   |   |   |   |
Name/IP Address      NP  NR  Span  Frequency  Freq Skew  Offset  Std Dev
-----
golem.canonical.com  4   4    8    +110.219   6924.452  +64ms  1290us
alphyn.canonical.com 5   4   462   -1.855     7.802  +5879us  307us
pugot.canonical.com  4   3    8   -258.540  10708.124  -140ms  1758us
chilipepper.canonical.com  4   4    6   -99.670   553.706  -49ms   53us
time.cloudflare.com   1   0    0    -0.576    2000.000  +0ns   4000ms
ntp1.home4u.ch        4   3    7   -161.961   3980.081  -89ms   812us
195.21.59.161         4   3    9   +56.252   1157.711  +34ms   175us
ntp2.home4u.ch        5   3   525   +0.466    3.909  -1971us  148us
```

Sunucunun hangi sunucuya senkronize olduğunu görelim.

chronyc tracking

```

root@ubuntu:~# chronyc tracking
Reference ID      : 3E0CAD0C (ntp2.home4u.ch)
Stratum          : 2
Ref time (UTC)   : Sun Feb 07 22:48:02 2021
System time      : 0.000001965 seconds fast of NTP time
Last offset      : -0.002271715 seconds
RMS offset       : 0.002271715 seconds
Frequency        : 9.902 ppm fast
Residual freq   : +0.005 ppm
Skew             : 2.645 ppm
Root delay       : 0.070599593 seconds
Root dispersion  : 0.004585291 seconds
Update interval  : 64.9 seconds
Leap status      : Normal

```

Chrony'nin yapılandırma dosyası /etc/chrony dizini altında bulunan chrony.conf dosyasıdır. Listedekilerin haricinde başka sunucular eklemek istersek bu dosyayı yapılandırmalıyız. Text editörümüzle dosyaya girip Türkiye için kullanılan genel NTP serverlarının tanımlarını dosyaya yapıştırıyoruz.

nano /etc/chrony/chrony.conf

```

GNU nano 4.8                               /etc/chrony/chrony.conf                         Modified
# Welcome to the chrony configuration file. See chrony.conf(5) for more
# information about usable directives.

# This will use (up to):
# - 4 sources from ntp.ubuntu.com which some are ipv6 enabled
# - 2 sources from 2.ubuntu.pool.ntp.org which is ipv6 enabled as well
# - 1 source from [01].ubuntu.pool.ntp.org each (ipv4 only atm)
# This means by default, up to 6 dual-stack and up to 2 additional IPv4-only
# sources will be used.
# At the same time it retains some protection against one of the entries being
# down (compare to just using one of the lines). See (LP: #1754358) for the
# discussion.
#
# About using servers from the NTP Pool Project in general see (LP: #104525).
# Approved by Ubuntu Technical Board on 2011-02-08.
# See http://www.pool.ntp.org/join.html for more information.
#pool ntp.ubuntu.com          iburst maxsources 4
#pool 0.ubuntu.pool.ntp.org    iburst maxsources 1
#pool 1.ubuntu.pool.ntp.org    iburst maxsources 1
#pool 2.ubuntu.pool.ntp.org    iburst maxsources 2
server 0.tr.pool.ntp.org
server 1.tr.pool.ntp.org
server 2.tr.pool.ntp.org
server 3.tr.pool.ntp.org

```

chronyd servisini yeniden başlatıyoruz.

systemctl restart chronyd

NTP serverları tekrar kontrol edelim.

```

root@ubuntu:~# chronyc sources -v
210 Number of sources = 4

    .-- Source mode '^' = server, '=' = peer, '#' = local clock.
    / .- Source state '*' = current synced, '+' = combined , '-' = not combined,
    | / '?' = unreachable, 'x' = time may be in error, '~' = time too variable.
    ||                                .- xxxx [ yyyy ] +/- zzzz
    ||          Reachability register (octal) -.           | xxxx = adjusted offset,
    ||          Log2(Polling interval) --.           |         yyyy = measured offset,
    ||          \           |           |           zzzz = estimated error.
    ||          |           |
    ||          |           \

MS Name/IP address          Stratum Poll Reach LastRx Last sample
=====

^? time.cloudflare.com        3   6     5    27    +25ms[  +25ms] +/-   55ms
^? 195.21.59.161              2   6     5    26    +4302us[+4302us] +/-   66ms
^? testntp1.superonline.net   1   6     5    37    +26ms[  +26ms] +/-   44ms
^? host-213-14-25-177.rever>  2   6     2    139   +5047us[+5047us] +/-   61ms

```

11 - YAZILIM YÖNETİMİ

Linux sistemlerde yazılımların kurulumlarını paket yöneticileri gerçekleştirir. Linux dağıtımlarına göre farklı paket yöneticileri olsa da en bilindik olan apt ve rpm'dir. Rpm paket yöneticileri RedHat, Centos, Fedora sistemlerinden kullanılırken Ubuntu gibi debian tabanlı sistemlerde apt kullanılır.

Paket yöneticileri uygulamaların güncellliğini, yazılımlar kurulurken gerekli tüm bağımlılık şartlarının uygun olup olmadığını, uygulamaların indirileceği konumları güncellemek ve paketlerin güvenli bir sağlayıcıdan indirmek gibi görevleri vardır. Eğer paket yöneticileri olmasaydı sunucuya kurulacak her uygulama için uygulama kaynağını bulup, bağımlılıklarını sağlayıp sisteme kurmamız gerekecekti.

apt install/update/upgrade/remove/purge/search/list

Advanced Package Tool kelimelerinin kısaltması olan apt Debian tabanlı Linux sistemlerde kullanılır. APT, paketleri otomatik olarak bulmak ve indirmek için tasarlanmıştır. APT, internet kesintilerinden kaynaklanan uygulama yükleme kesintilerini problem giderildiğinde devam ettirmeyi de yönetebilir.

apt-get ve apt komutları temelde aynı amaca hizmet etseler de apt komutu daha fonksiyonel ve kullanması daha kolay bir komuttur.

Yazılımlar paketlerinin depolandığı yolları barındıran repository indexlerini güncellemek ve yeni sürümlerini olup olmadığını denetlemek için apt komutunu kullanırız.

sudo apt update

```
ozgur@ubuntu:~$ sudo apt update
Hit:1 http://tr.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://tr.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://tr.archive.ubuntu.com/ubuntu focal-backports InRelease
Get:4 http://tr.archive.ubuntu.com/ubuntu focal-security InRelease [109 kB]
Get:5 http://tr.archive.ubuntu.com/ubuntu focal-security/main amd64 Packages [482 kB]
Get:6 http://tr.archive.ubuntu.com/ubuntu focal-security/main amd64 c-n-f Metadata [6,164 B]
Get:8 http://tr.archive.ubuntu.com/ubuntu focal-security/universe amd64 c-n-f Metadata [9,784 B]
Fetched 125 kB in 7s (19.2 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
```

Yazılımları kontrol ederek sistem üzerindeki tüm yazılımların güncellemelerini yapar. İlk olarak paket listelerini kontrol ederek uygulamaların güncelleme sıralamalarını denetler eğer güncelleme varsa yüklenecek, kaldırılacak ve güncellenecek paketleri gösterir. Sonrasında güncellenecek, indirilecek ve kaldırılacak uygulama sayısı ve gerekli disk alanını bildirerek işlem için sistem yöneticisinden onay alınır. Komut sonunda -y argümanı ile onaylama işlemini en başta yapabilirsiniz. Böylece siz kahvenizi almaya gittiğinizde güncellemeler tamamlanmış olur :) Elbette kritik bir sunucu ise istenmeyen sonuçların ortaya çıkmaması için onay işleminden önce değişiklik yapılacak uygulamalar incelenmelidir. **apt upgrade komutu sadece paket yükleme ve güncelleme işlemi yapar.**

sudo apt upgrade

```
ozgur@ubuntu:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following NEW packages will be installed:
  motd-news-config python3-pexpect python3-ptyprocess
The following packages will be upgraded:
  alsa-ucm-conf apt apt-utils base-files bcache-tools bind9-dnsutils bind9-host
  bind9-libs bolt bsduutils busybox-initramfs busybox-static cloud-init
  command-not-found cryptsetup cryptsetup-bin cryptsetup-initramfs cryptsetup-run
  fdisck finalrd initramfs-tools initramfs-tools-bin initramfs-tools-core
  language-selector-common libapt-pkg6.0 libasound2 libasound2-data libblkid1
  libc-bin libc6 libcryptsetup12 libdns-export1109 libdrm-common libdrm2 libefiboot1
  libefivar1 libfdisk1 libglib2.0-0 libglib2.0-bin libglib2.0-data libisc-export1105
  libldap-2.4-2 libldap-common liblzma5 libmount1 libnetplan0 libpam-modules
  libpam-modules-bin libpam-runtime libpam0g libplymouth5 libsmartcols1 libudev1
  libuuid1 locales lshw lsof mdadm mount netplan.io open-vm-tools plymouth
  plymouth-theme-ubuntu-text python-apt-common python3-apt python3-commandnotfound
  python3-distupgrade python3-software-properties python3-update-manager rsyslog
  snapd software-properties-common sosreport ubuntu-minimal
  ubuntu-release-upgrader-core ubuntu-server ubuntu-standard udev unattended-upgrades
  update-manager-core update-notifier-common util-linux uid-runtime xz-utils zlib1g
85 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 52.2 MB of archives.
After this operation, 7,460 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Eğer uygulamaların kullanmadığı paketlerin kaldırılmasıyla birlikte bir upgrade işlemi yapmak isterseniz komutumuz;

sudo apt full-upgrade

Tek bir uygulamayı upgrade edecekseñiz komutumuz;

sudo apt upgrade net-tools

```
ozgur@ubuntu:~$ sudo apt upgrade net-tools
[sudo] password for ozgur:
Reading package lists... Done
Building dependency tree
Reading state information... Done
net-tools is already the newest version (1.60+git20180626.aebd88e-1ubuntul).
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

Sadece kullanılmayan bütün pekatleri otomatik olarak kalırmak istersek;

sudo apt autoremove

Sistemimize bir uygulama yüklemek için install argümanını kullanmalıyız. Eğer yükleyeceğimiz uygulamaya ait özel bir paket varsa uygulama_isminden sonra paket ismini yazmalıyız.

sudo apt install uygulama_ismi

Sistemden bir uygulama kaldırmak için remove argümanı kullanılır. apt remove komutu uygulamanın yapılandırma dosyalarını geride bırakır. Böylece uygulamayı tekrar geri yükleyebilirsiniz.

sudo apt remove uygulama_ismi

Uygulamayı ve dosyalarını tamamen kaldırmak için purge argümanı kullanılır.

sudo apt purge uygulama_ismi

Sistem yüklü uygulamaları görmek için list argümanı kullanılır.

sudo apt list

Yüklemek istediğiniz bir uygulamayı aramak için search argümanı kullanılmalıdır. Örneğin netstat aracını kullanacaksınız ancak Ubuntu ilk kurulduğunda bu araç yoktur. net-tools uygulaması ile bu araç gelir. net-tools'u hatırlayamadınız. apt search netstat yazdığında yüklemeniz gereken uygulama size listelenir.

sudo apt search uygulama_ismi veya anımsatacak bir isim

```
ozgur@ubuntu:~$ apt search netstat
Sorting... Done
Full Text Search... Done
bwm-ng/focal 0.6.2-1 amd64
    small and simple console-based bandwidth monitor

gnome-nettool/focal 3.8.1-3 amd64
    network information tool for GNOME

net-tools/focal,now 1.60+git20180626.aebd88e-1ubuntul amd64 [installed]
    NET-3 networking toolkit
```

Kaynaktan Yazılım Yükleme - wget, dpkg

Sunucuya yüklemek istediğiniz uygulamalar Ubuntu repository'lerinde (repo) olmayabilir. Genelde bu uygulamaların web sitelerinde kaynak kodları sıkıştırılmış tar.gz dosyaları veya deb paketleri halinde kullanıcılar sunulur. Bu dosyaları sunucunuza indirerek uygulamalarınızı çalıştırabilirsiniz.

Uygulamayı sftp, ftp veya scp ile belirli bir konumdan sunucunuza gönderebilirsiniz veya uygulama web üzerinden erişilebilir bir konumdaysa **wget** komutu uygulamayı sunucunuza indirmenize yardımcı olur. Wget HTTP, HTTPS ve FTP protokollerini kullanarak dosya indirme işlemini ücretsiz bir araçtır.

Dpkg (Debian Package) ise .deb paketlerini yüklemek için kullanılan bir araçtır. Apt gibi yeteneklere sahip değildir. Apt bütün paket yönetimini otomatize ederken dpkg ile bu işlemleri manuel yapmanız gereklidir. dpkg genelde uygulamanın eski sürümlerine ihtiyaç olduğunda veya repolarda bulunmayan bir uygulamayı sisteme kurmak istediğiniz zaman kullanılmaktadır.

wget'in aldığı bazı kullanışlı argümanlar ise;

-c yarı kalan indirmeyi tamamlar.

wget -c https://examples.xyz/file.tar.gz

-i bir dosya içerisinde yazılan dosya kaynaklarını giderek çoklu indirmek yapar.

wget -i indirilecek.txt

-O indirilen dosyanın ismini değiştirir.

wget -O dosyalar.tar.gz https://examples.xyz/file.tar.gz

-P ile bir dosya yolu belirterek uygulamayı dosya yoluna indirirsiniz.

wget -P dosyalarim/arsiv/ https://examples.xyz/file.tar.gz

-tries ile internet bağlantısında oluşacak bir kesintide indirmeyi tekrar denemek kullanılır. Örneğimizde 10 defa deneme yapacaktır.

wget -tries=10 https://examples.xyz/file.tar.gz

--limit-rate ile indirme hızınızı sınırlayabilirsiniz.

wget -limit-rate=1m https://examples.xyz/file.tar.gz

.deb Paketi İle Kurulum Yapmak

İlk önce wget ile mysql uygulamasının işletim sistemimize uygun olan deb paketini /tmp klasörü altına indirelim.

İşletim sistemimizin versiyon bilgisini **cat /proc/version** komutu ile görebiliriz. İndireceğimiz paketlerin işletim sisteminizle uyumlu olmasına dikkat edin.

```
ozgur@ubuntu:/tmp$ cat /proc/version
Linux version 5.4.0-65-generic (buildd@lcy01-amd64-018) (gcc version 9.3.0 (Ubuntu 9.3.0-17ubuntu1~20.04)) #73-Ubuntu SMP Mon Jan 18 17:25:17 UTC 2021
```

wget aracıyla deb paketini indiriyoruz.

wget -c https://dev.mysql.com/get/mysql-apt-config_0.8.16-1_all.deb

```
ozgur@ubuntu:/tmp$ wget -c https://dev.mysql.com/get/mysql-apt-config_0.8.16-1_all.deb
--2021-02-09 00:01:35-- https://dev.mysql.com/get/mysql-apt-config_0.8.16-1_all.deb
Resolving dev.mysql.com (dev.mysql.com)... 137.254.60.11
Connecting to dev.mysql.com (dev.mysql.com)|137.254.60.11|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://repo.mysql.com//mysql-apt-config_0.8.16-1_all.deb [following]
--2021-02-09 00:01:36-- https://repo.mysql.com//mysql-apt-config_0.8.16-1_all.deb
Resolving repo.mysql.com (repo.mysql.com)... 23.64.143.224
Connecting to repo.mysql.com (repo.mysql.com)|23.64.143.224|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 35528 (35K) [application/x-debian-package]
Saving to: 'mysql-apt-config_0.8.16-1_all.deb'

mysql-apt-config_0.8.16 100%[=====] 34.70K --.-KB/s   in 0.1s
2021-02-09 00:01:37 (238 KB/s) - 'mysql-apt-config_0.8.16-1_all.deb' saved [35528/35528]

ozgur@ubuntu:/tmp$ ls -ltr
total 56
-rw-rw-r-- 1 ozgur ozgur 35528 Nov  5 16:51 mysql-apt-config_0.8.16-1_all.deb
```

dpkg komutu ile MySQL repo paketini kuruyoruz.

sudo dpkg -i mysql-apt-config_0.8.16-1_all.deb

```
ozgur@ubuntu:/tmp$ sudo dpkg -i mysql-apt-config_0.8.16-1_all.deb
(Reading database ... 72658 files and directories currently installed.)
Preparing to unpack mysql-apt-config_0.8.16-1_all.deb ...
Unpacking mysql-apt-config (0.8.16-1) over (0.8.9-1) ...
Setting up mysql-apt-config (0.8.16-1) ...
Warning: apt-key should not be used in scripts (called from postinst maintainer script of the package mysql-apt-config)
OK
```

sudo apt update komutu ile paket bilgisini güncelliyoruz. Bu şekilde uygulamayı yüklerken bizim belirlediğimiz paket içeriği dikkate alınarak gerekli yazılımlar indirilip kurulacak.

```
ozgur@ubuntu:/tmp$ sudo apt update
Hit:1 http://tr.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://tr.archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:3 http://tr.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:4 http://tr.archive.ubuntu.com/ubuntu focal-security InRelease
Get:5 http://repo.mysql.com/apt/ubuntu focal InRelease [12.2 kB]
Get:6 http://repo.mysql.com/apt/ubuntu focal/mysql-8.0 Sources [961 B]
Get:7 http://repo.mysql.com/apt/ubuntu focal/mysql-apt-config amd64 Packages [566 B]
Get:8 http://repo.mysql.com/apt/ubuntu focal/mysql-8.0 amd64 Packages [7,640 B]
Get:9 http://repo.mysql.com/apt/ubuntu focal/mysql-tools amd64 Packages [6,742 B]
Fetched 28.1 kB in 13s (2,160 B/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
```

sudo apt install mysql-server komutu ile uygulamamızı yükliyoruz.

```
ozgur@ubuntu:/tmp$ sudo apt install mysql-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libmecab2 mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client mysql-common
  mysql-community-client mysql-community-client-core mysql-community-client-plugins
  mysql-community-server mysql-community-server-core
The following NEW packages will be installed:
  libmecab2 mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client mysql-common
  mysql-community-client mysql-community-client-core mysql-community-client-plugins
  mysql-community-server mysql-community-server-core mysql-server
0 upgraded, 12 newly installed, 0 to remove and 0 not upgraded.
Need to get 35.1 MB of archives.
After this operation, 319 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

systemctl status mysql komutu ile servisin durumuna bakalım.

```
ozgur@ubuntu:/tmp$ systemctl status mysql
● mysql.service - MySQL Community Server
  Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
  Active: active (running) since Tue 2021-02-09 00:17:05 +03; 2s ago
```

Sıkıştırılmış Dosyalar İle Kurulum Yapmak

Sıkıştırılmış dosyalarla kurulum yaparken genellikle firmalar kurulum talimatlarını içeren dokümanlar hazırlarlar. Uygulamaların sıkıştırılmış hallerini açtıktan sonra kimi uygulama bir script dosyası ile yükleme işlemini başlatırken kimi uygulamalar da çalıştırılacakları dizinin altına kopyalanmak ister. Uygulamamızın tar.gz uzantılı olan halinin linkini belirledikten sonra wget aracı ile indiriyoruz. Farklı sıkıştırılma teknikleri de desteklenir.

wget -c paketinyolu.tar.gz komutu ile indiriyoruz.

İndirdiğimiz dosyayı, sıkıştırma teknüğine göre ister bulunduğu alana ister komut sonuna çıkarılacağı yolu göstererek export ediyoruz. Devamında uygulamanın kurulum talimatlarına göre ilerliyoruz.

tar zxvf packagename.tar.gz -C

tar zxvf packagename.tgz -C

tar jxvf packagename.bz -C

tar jxvf packagename.tar.bz2 -C

Mirror Değiştirmek

Mirror'lar sunucu üzerine yüklenecek yazılımların çekileceği adreslerdir. Eğer herhangi bir nedenden dolayı mirrorlar cevap vermez ise sunucuya yazılım yüklenemez, upgrade ve update işlemleri yapılamaz. Temel olarak “Temporary failure resolving” gibi bir hata ile karşılaşılır.

Eğer DNS çözümlemesinde bir problem yok ise mirror yolunu sağlam bir adres ile değiştirerek işlemlere devam edilebilir. Mirror'lar /etc/apt/sources.list dosyasında bulunur.

cp /etc/apt/sources.list /etc/apt/sources.list.bak komutuyla dosyanın bir yedeği alınır.

nano /etc/apt/sources.list komutuyla dosya içine girilerek indirme yapmak istediğiniz adres girilir. Kullanılmayacak olanlar kaldırılabilir veya # işareteti ile yorum satırına çevrilebilir.

```
root@test: ~
GNU nano 4.8                                     /etc/apt/sources.list
# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to
# newer versions of the distribution.
# deb http://tr.archive.ubuntu.com/ubuntu focal main restricted
# deb-src http://tr.archive.ubuntu.com/ubuntu focal main restricted
deb http://archive.ubuntu.com/ubuntu focal main restricted
```

12 – Boot (Ön Yükleme) İşlemi Anlama ve Yönetme

Bilgisayarınızı açtıktan sonra , Linux çekirdeğinin yüklenmesinden başlayarak oturum açma ekranına gelene kadar çalıştırılan kod ve uygulamaların bütününe boot işlemi denir. Boot sırasında gerçekleşen olayları anlamak karşılaşacağınız problemleri çözme konusunda yardımcı olacaktır.

Boot İşlem Sırasını Anlama

BIOS (Basic Input Output System) bilgisayar donanımı ve yazılımı arasında köprü gören ve anakart üzerindeki bir yongada depolanan bir uygulamadır. Bilgisayırlı ilk açığınızda BIOS, işletim sisteminin çekirdeğinin nereden olduğu, nasıl yüklenmesi ve nasıl başlatılması gerekiği talimatlarını içeren kodları (bootloader veya bootcode) arar. Devamında ön yüklemenin yapılacağı CD, DVD, USB, HDD gibi bir önyükleme birimi arar. BIOS, birincil (primary) disk içinde MBR (Master Boot Recorder) denilen ön yükleyici kodu ve partition (bölüm) tablosunu barındıran kodu bulur ve belleğe yükler. BIOS ön yükleyici bulup kodu belleğe yükledikten sonra sistemin kontrolü önyükleyiciye geçer. Ubuntu varsayılan önyükleyici uygulaması GRUB2'dir.

Sonrasında ön yükleyici diskte bulunan Kernel'i bulur ve belleğe yükler. İşlem tamamlandıktan sonra kontrol sistem kaynaklarını yöneten Kernel'e geçer. Kernel yüklemesinden sonra servislerin başlatılması için systemd çalıştırılır. Systemd, default.target olarak tanımlanan servislerin nasıl çalışacağını bildiren unit'leri çalıştırır.



MBR ve GPT nedir?

MBR (Master Boot Record) sabit disk düzenini tanımlamak için kullanılan bir bölümleme şemasıdır. MBR, bir önyükleyici ve bir bölüm tablosu dahil olmak üzere bir bilgisayarı başlatmak için gereken her şeyi içerir. MBR bir disk üzerinde en fazla 4 bölüm oluşturabilir ve bölüm tarafından kullanılabilecek maksimum boyut 2 TB ile sınırlıdır.

GPT (GUID (Global Unique) Partition Table) ise MBR'ın bölümleme ve 2 TB'lık boyut sınırlarına çözüm olarak meydana geldi. BIOS'un yerini alan yeni nesil UEFI de (Unified Extensible Firmware Interface) GPT'yi kullanır. Yeni nesil sunucular artık BIOS değil UEFI ile gelmektedir.

GPT bir disk üzerinden 128 bölüm oluşturabilir. 2 TB'lık sınır yoktur. 8 zebibyte (ZiB) kadar bölüm oluşturulabilir.

MBR'da partition table single point of failure durumdadır yani tektir. Silinirse bütün partition bilgileri kaybolduğu için verilerinizde kaybolur. Ancak GPT'de partition table yedeklidir.

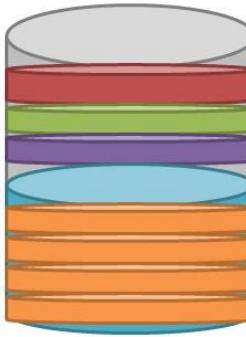
GPT, partition tablosunda hata veya bozulma olup olmadığını kontrol eden bir algoritma (checksum) kullanır.

	MBR	GPT
Maksimum Bölüm Boyutu	2 TiB	8 ZiB
Bölüm Sayısı	4	128

MBR için en iyi çözüm 4 tane partition oluşturmaktansa 3 primary partition ve 1 extended partition oluşturmaktır. Bu şekilde extended bölüme içerisinde 15 tane logical partition oluşturulabilir.

Eğer MBR'da 4 tane partition varken 5 ncı partition oluşturmak isterseniz diskde yeterli alan yok uyarısı ile işleminiz iptal edilir.

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sdc1		2048	1000000	997953	487.3M	83	Linux
/dev/sdc2		1001472	2000000	998529	487.6M	83	Linux
/dev/sdc3		2000896	3000000	999105	487.9M	83	Linux
/dev/sdc4		3000320	6291455	3291136	1.6G	5	Extended
/dev/sdc5		3002368	4000000	997633	487.1M	83	Linux
/dev/sdc6		4003840	5000000	996161	486.4M	83	Linux
/dev/sdc7		5003264	5500000	496737	242.6M	83	Linux
/dev/sdc8		5502976	6291455	788480	385M	83	Linux



fdisk –l komutuyla disklerin sektör, boyut, tip ve şemaları hakkında bilgi alınır.

```
Disk /dev/sda: 22 GiB, 23622320128 bytes, 46137344 sectors
Disk model: Virtual Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disklabel type: gpt
Disk identifier: 06A493A0-4B40-428C-8538-E9E16DA3AE23

Device Start End Sectors Size Type
/dev/sda1 2048 4095 2048 1M BIOS boot
/dev/sda2 4096 1028095 1024000 500M Linux filesystem
/dev/sda3 1028096 46135295 45107200 21.5G Linux filesystem
```

```
Disk /dev/sdb: 2 GiB, 2147483648 bytes, 4194304 sectors
Disk model: Virtual Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
```

GRUB 2 Nedir?

GRUB 2 (Grand Unified Bootloader) işletim sisteminden bağımsız bir önyükleyicidir. GRUB 2, BIOS'tan görevi aldıktan sonra Linux Kernel'i belleğe yükleyerek çalıştırır. Kernel yüklemesi tamamlandıktan sonra GRUB2 nin görevi tamamlanır. Bilgisayarınızı başlattıktan sonra ilk olarak etkileşim içine girdiğiniz menü GRUB2 menüsüdür. Bu menüyü kullanarak kurtarma, boot, root şifresi değişikliği gibi işlemler gerçekleştirilir. GRUB2 birden fazla kernel'i destekleyebilir ve boot işlemi sırasında kerneller arasında seçim yapılmasına izin verir. GRUB2 genellikle problemsiz bir şekilde çalışır ve herhangi bir müdahaleye ihtiyaç duymaz.

GRUB 2, **/etc/grub.d/** dosyasında bulunan scriptleri ve **/etc/default/grub** dosyasındaki ayarları çalıştırarak GRUB2 menüsünü (**grub.cfg**) oluşturur. Bu dosya, update-grub komutu her çalıştırıldığında otomatik olarak yeniden oluşturulur. Bu komut, varsayılan olarak mevcut Ubuntu işletim sisteminin durumunu belirleyen ve bulunursa GRUB 2 menüsüne eklenen diğer işletim sistemleri için sistemde arama yapan komut dosyalarını etkinleştirir. Update-grub komutu, root ayrıcalıklarına sahip bir kullanıcı tarafından çalıştırılabilir.

GRUB2'nin yüklü olduğu konumu ve UUID'yi bulmak için:

```
sudo grub-probe -t device /boot/grub
```

```
sudo grub-probe -t fs_uuid /boot/grub
```

```
ozgur@ubuntu:~$ sudo grub-probe -t device /boot/grub  
/dev/sda2  
ozgur@ubuntu:~$ sudo grub-probe -t fs_uuid /boot/grub  
f4c6fc21-29d9-4589-8e71-e7abac23cf7b
```

GRUB2 Dosyaları

/boot/grub/grub.cfg GRUB2 nin ana yapılandırma dosyasıdır. /etc/grub.d dizini altındaki dosyaların yapılandırması bu dosya içinde toplanır. Ancak bu dosya üzerinden direkt yapılmaz.

/etc/default/grub Bu dosyalar root yetkisine sahip bir kullanıcı tarafından düzenlenlenebilir ve menü güncellendiğinde grub.cfg'ye eklenir.

/etc/grub.d/ Bu dizindeki scriptler update-grub (grub.cfg'nin güncellenmesi için çalıştırılır.) komutunun yürütülmesi sırasında okunur ve talimatları /boot/grub/grub.cfg içine dahil edilir.

GRUB2 için değişiklik /etc/default/grub ve /etc/grub.d/ dosyalarında yapılır. update-grub ile güncellenerek değişiklikler /boot/grub/grub.cfg'ye uygulanır.

```
root@ubuntu:~# cat /etc/default/grub  
# If you change this file, run 'update-grub' afterwards to update  
# /boot/grub/grub.cfg.  
# For full documentation of the options in this file, see:  
#   info -f grub -n 'Simple configuration'  
  
GRUB_DEFAULT=0  
GRUB_TIMEOUT_STYLE=hidden  
GRUB_TIMEOUT=0  
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`  
GRUB_CMDLINE_LINUX_DEFAULT="maybe-ubiquity"  
GRUB_CMDLINE_LINUX=""  
  
# Uncomment to enable BadRAM filtering, modify to suit your needs  
# This works with Linux (no patch required) and with any kernel that obtains  
# the memory map information from GRUB (GNU Mach, kernel of FreeBSD ...)  
#GRUB_BADRAM="0x01234567,0xfefefefe,0x89abcdef,0xefefefef"  
  
# Uncomment to disable graphical terminal (grub-pc only)  
#GRUB_TERMINAL=console  
  
# The resolution used on graphical terminal  
# note that you can use only modes which your graphic card supports via VBE  
# you can see them in real GRUB with the command 'vbeinfo'  
#GRUB_GFXMODE=640x480  
  
# Uncomment if you don't want GRUB to pass "root=UUID=xxx" parameter to Linux  
#GRUB_DISABLE_LINUX_UUID=true  
  
# Uncomment to disable generation of recovery mode menu entries  
#GRUB_DISABLE_RECOVERY="true"  
  
# Uncomment to get a beep at grub start  
#GRUB_INIT_TUNE="480 440 1"
```

Grub menüsü boot sırasında otomatik olarak gizlenmişse GRUB_TIMEOUT değeri değiştirilerek menünün belirttiğiniz zaman kadar ekranда kalmasını sağlayabilirsiniz. Değişiklik yapmak için grub dosyasını metin editörümüz ile açıyoruz.

sudo nano /etc/default/grub

GRUB_TIMEOUT değerinin karşısına -1 yazarak başlangıçta GRUB menüsünün bizim seçim yapmamız için açık kalmasını sağlayacak şekilde ayarlayabiliriz.

```
GNU nano 4.8          /etc/default/grub          Modified
# If you change this file, run 'update-grub' afterwards to update
# /boot/grub/grub.cfg.
# For full documentation of the options in this file, see:
#   info -f grub -n 'Simple configuration'

GRUB_DEFAULT=0
GRUB_TIMEOUT_STYLE=hidden
GRUB_TIMEOUT=-1
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
GRUB_CMDLINE_LINUX_DEFAULT="maybe-ubiquity"
GRUB_CMDLINE_LINUX=""

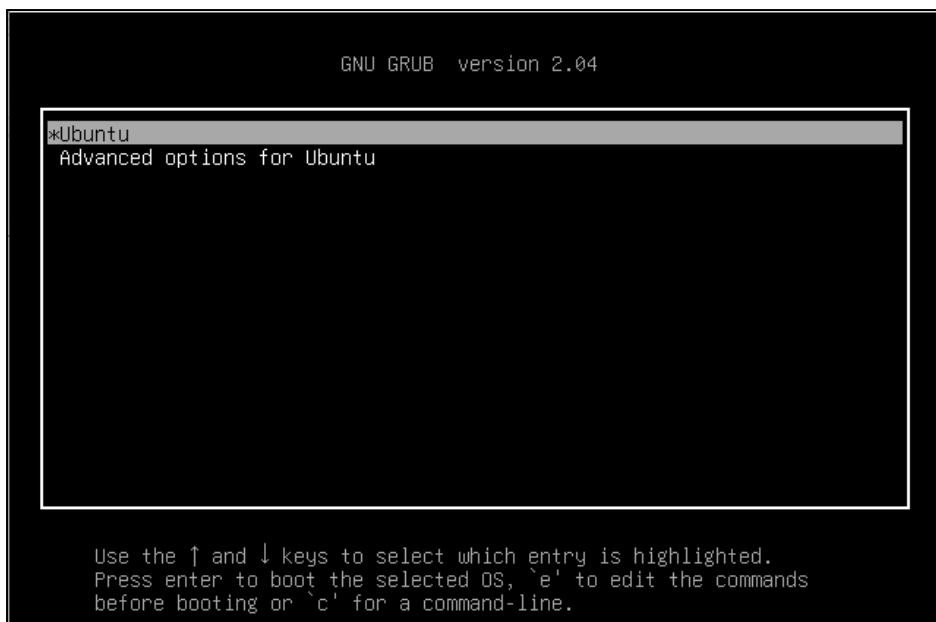
# Uncomment to enable BadRAM filtering, modify to suit your needs
# This works with Linux (no patch required) and with any kernel that obtains
# the memory map information from GRUB (GNU Mach, kernel of FreeBSD ...)
#GRUB_BADRAM="0x01234567,0xfefefefe,0x89abcdef,0xefefefef"

# Uncomment to disable graphical terminal (grub-pc only)
#GRUB_TERMINAL=console
```

Yaptığımız değişikliğin **grub.cfg** dosyasına yazılması için **update-grub** ile güncelliyoruz.

```
root@ubuntu:~# update-grub
Sourcing file '/etc/default/grub'
Sourcing file '/etc/default/grub.d/init-select.cfg'
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-5.4.0-42-generic
Found initrd image: /boot/initrd.img-5.4.0-42-generic
done
root@ubuntu:~#
```

Sunucumuzu reboot ettiğimizde başlangıç ekranında GRUB2 menüsü bizim seçim yapmamızı bekleyecek.



GRUB menüsüne grub dosyasında değişiklik yapmadan geçmek isterseniz ilk açılış sırasında SHIFT tuşuna basılı tutmalısınız.

Grub Bash ekranına geçmek için GRUB menüsünde c tuşuna basılmalıdır.

```
GNU GRUB version 2.04

Minimal BASH-like line editing is supported. For the first word,
TAB lists possible command completions. Anywhere else TAB lists
possible device or file completions. ESC at any time exits.

grub> _
```

ls komutuyla sunucu içindeki diskler görüntülenir. Diskler 0'dan başlayarak numaralandırılır. 0 numarasına sahip disk ilk diskimizdir.

```
grub> ls
(hd0) (hd0,gpt3) (hd0,gpt2) (hd0,gpt1) (hd1) (hd2) (hd3) (hd4)
```

Boot diski yine ls komutuyla bulabiliyoruz. Boot disk içerisinde kernel dosyaları bulunur.

```
grub> ls (hd0,gpt2)/
lost+found/ config-5.4.0-66-generic initrd.img.old vmlinuz.old
System.map-5.4.0-66-generic vmlinuz-5.4.0-66-generic initrd.img vmlinuz
grub/ vmlinuz-5.4.0-67-generic initrd.img-5.4.0-66-generic
config-5.4.0-67-generic System.map-5.4.0-67-generic
initrd.img-5.4.0-67-generic
```

Eğer grub.cfg dosyamızın içeriğini cat komutuyla görüntüleyebiliyoruz. Grub.cfg dosyası boot dizininde olduğunda erişim sağlayabiliyoruz.

```
grub> ls (hd0,gpt2)/grub/grub.cfg
grub.cfg
```

```
grub> cat (hd0,gpt2)/grub/grub.cfg
```

Acil bir durumda Grub satırından bir kernel'i seçerek sistem disklerindeki verilere **initramfs** üzerinden ulaşmak istersek linux16 komutunu kullanabiliyoruz.

```
grub> linux16 /vmlinuz-5.4.0-67-generic root=/dev/sda2
grub> initrd16 /initrd.img-5.4.0-67-generic
grub> _
```

boot komutu ile seçilen kernel'den sistem başlatılır.

```
grub> initrd16 /initrd.img-5.4.0-67-generic
grub> boot_
```

Systemd Nedir?

Servislerin başlatılma işlemi sırasında hangi hizmetlerin başlatılacağı, klavyeden veri girişi, sistem kaynaklarının kontrolü vb. işlevleri Ubuntu'da systemd tarafından kontrol edilir. Klasik sistemlerde (init) belirli bir sıra ile hizmetler başlatılırken, Systemd birbirine bağlı olmayan hizmetleri aynı anda başlatma yeteneğine sahiptir. Systemd, systemctl, journalctl, notify, analyze, cgls, cgtop, logind ve nspawn gibi araçlarla sistemi ve servisleri yönetir.

Systemd, deamon ve process lerin çalışması için mekanizmalar sağlar. Deamon'lar arka planda çalışan uygulamalardır. Bu servisler, servis isimlerinin sonlarına d harfi alır (ssh, sshd gibi). Deamon'lar sistem başlatılırken boot sırasında başlatılır ve sistem kapatılana yada

manuel olarak kapatılana kadar çalışmaya devam eder. Process'ler ise meydana gelen tüm işlemleri kapsar ve her bir process'e bir PID (Process ID) atanır.

Ubuntu 14.04 ve önceki sistemlerde servislerin kontrolü initd tarafından sağlanmaktadır. Ancak günümüzde servis kontrolü systemd tarafından yapıldığından initd ve ona bağlı olan runlevel'ler tarafından bahsetmeyeceğiz. Runlevel'lar yerini Target'lara bırakmıştır. Ubuntu 20.04 'de /etc dizini altını kontrol ettiğinizde initd hala vardır. Ancak initd ile çalıştırığınız tüm servisler ve fazlası systemd tarafından tek bir arayüz kullanılarak çalıştırıldığından initd ye ihtiyaç duyulmamaktadır.

Systemd, systemctl komutunu kullanarak bir servisi durdurup başlatabilir, durumunu gözlemebilirsiniz.

Komut	Tanımı
systemctl start servis_adi	Servisi Başlatır.
systemctl stop servis_adi	Servisi Durdurur.
systemctl restart servis_adi	Servisi Yeniden Başlatır.
systemctl reload servis_adi	Servisi yeniden başlatmak yerine yapılandırma dosyalarını yeniden yükler.
systemctl status servis_adi	Servisin durumunu gösterir.
systemctl enable servis_adi	Servisin başlangıçta systemd tarafından başlatılmasını sağlar.
systemctl disable servis_adi	Servisin başlangıçta systemd tarafından başlatılmamasını sağlar.
systemctl halt	Sistemi Kapatır.
systemctl reboot	Sistemi Reboot eder.

Systemctl status komutunun çıktısındaki status alanının aldığı değerler aşağıdaki gibidir. Bu değer birimin aktiflik durumu ve mevcut durumuyla birleştirerek ekrana getirir.

Status	Tanım
Loaded	Unit file aktif ve işleniyor.
Active (running)	Bir veya daha fazla işlem çalışıyor.
Active (exited)	Tek seferlik yapılandırma başarıyla tamamlandı.
Active (waiting)	Bir işlem için beklemeye.
Inactive	Çalışmıyor.
Enabled	Boot işlemi sırasında başlayacak.
Disabled	Boot işlemi sırasında başlamayacak.
Static	Başka bir unit tarafından otomatik çalıştırılır. Enable edilemez.

Systemd, systemctl komutunu kullanarak servisleri kontrol etse de sadece servis kontrolü yapmaz. Aynı zamanda socket, device, mount, swap gibi birimleri de yönetir. Bu birimler Unit olarak adlandırılır.

Unit Tipi	Dosya Uzantısı	Açıklaması
Service	.service	Sistem üzerinde çalışan servisler
Target	.target	Systemd unit grubu
Automount	.automount	Automount edilecek dosya ve dizinler.
Device	.device	Kernel'in tanıdığı cihaz dosyaları.
Mount	.mount	Mount edilecek dosya sistemleri
Path	.path	Dosya sistemindeki dosya veya dizinler

Scope	.scope	Sonradan çalıştırılan harici uygulamalar.
Slice	.slice	Sistem süreçlerini yöneten unit dosyaları.
Snapshot	.snapshot	Systemd'nin snapshotları
Socket	.socket	Uygulamaların birbirleriyle iletişim kurması için yöntem oluşturur.
Swap	.swap	Swap dosyası
Timer	.timer	Systemd timer

Unit tiplerine göre bir liste görüntülemek için **systemctl list-units --type=service** komutu kullanılır. type argümanından sonra listelenmek istenen unit tipi yazılır. Örneğimizde servis unitler listelenmiştir.

```
root@ubuntu:~# systemctl list-units --type=service
 _UNIT           LOAD   ACTIVE SUB   DESCRIPTION
accounts-daemon.service loaded active running Accounts Service
apparmor.service    loaded active exited Load AppArmor prof>
apport.service     loaded active exited LSB: automatic cra>
atd.service        loaded active running Deferred execution>
blk-availability.service loaded active exited Availability of bl>
chrony.service     loaded active running chrony, an NTP cli>
cloud-config.service loaded active exited Apply the settings>
cloud-final.service loaded active exited Execute cloud user>
cloud-init-local.service loaded active exited Initial cloud-init>
cloud-init.service    loaded active exited Initial cloud-init>
console-setup.service loaded active exited Set console font a>
cron.service       loaded active running Regular background>
```

UNIT : Birim İsmi

LOAD : Ünenin yapılandırmasının systemd tarafından ayrıstırılıp ayrıstırılmadığı bilgisidir. Yüklenen birimlerin konfigürasyonu bellekte tutulur.

ACTIVE: Birimin aktif olup olmama durumunu gösterir.

SUB : Birimin mevcut çalışma durumunu gösterir.

DESCRIPTION : Birim hakkında kısa bir bilgi verir.

Targets ları görüntülemek için aşağıdaki komutu çalıştırabiliriz.

systemctl list-units --type=target

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
all.target	not-found	inactive	dead	all.target
basic.target	loaded	active	active	Basic System
blockdev@dev-disk-by\x2duuid-	loaded	inactive	dead	Block Device Preparation for /dev/disk/b
blockdev@dev-disk-by\x2duuid-	loaded	inactive	dead	Block Device Preparation for /dev/disk/b
blockdev@dev-disk-by\x2duuid-	loaded	inactive	dead	Block Device Preparation for /dev/disk/b
blockdev@dev-loop0.target	loaded	inactive	dead	Block Device Preparation for /dev/loop0
blockdev@dev-loop1.target	loaded	inactive	dead	Block Device Preparation for /dev/loop1
blockdev@dev-loop2.target	loaded	inactive	dead	Block Device Preparation for /dev/loop2
blockdev@dev-loop3.target	loaded	inactive	dead	Block Device Preparation for /dev/loop3
blockdev@dev-loop4.target	loaded	inactive	dead	Block Device Preparation for /dev/loop4
blockdev@dev-loop5.target	loaded	inactive	dead	Block Device Preparation for /dev/loop5
blockdev@dev-sda3.target	loaded	inactive	dead	Block Device Preparation for /dev/sda3
blockdev@dev-sda4.target	loaded	inactive	dead	Block Device Preparation for /dev/sda4
cloud-config.target	loaded	active	active	Cloud-config availability
cloud-init.target	loaded	active	active	Cloud-init target
cryptsetup.target	loaded	active	active	Local Encrypted Volumes
emergency.target	loaded	inactive	dead	Emergency Mode
getty-pre.target	loaded	inactive	dead	Login Prompts (Pre)
getty.target	loaded	active	active	Login Prompts
graphical.target	loaded	active	active	Graphical Interface
local-fs-pre.target	loaded	active	active	Local File Systems (Pre)
local-fs.target	loaded	active	active	Local File Systems
multi-user.target	loaded	active	active	Multi-User System
network-online.target	loaded	active	active	Network is Online
network-pre.target	loaded	active	active	Network (Pre)
network.target	loaded	active	active	Network
nss-lookup.target	loaded	active	active	Host and Network Name Lookups
nss-user-lookup.target	loaded	active	active	User and Group Name Lookups
paths.target	loaded	active	active	Paths
remote-fs-pre.target	loaded	active	active	Remote File Systems (Pre)
remote-fs.target	loaded	active	active	Remote File Systems
rescue.target	loaded	inactive	dead	Rescue Mode
shutdown.target	loaded	inactive	dead	Shutdown
slices.target	loaded	active	active	Slices
sockets.target	loaded	active	active	Sockets
sound.target	loaded	active	active	Sound Card
swap.target	loaded	active	active	Swap
sysinit.target	loaded	active	active	System Initialization
time-set.target	loaded	active	active	System Time Set
time-sync.target	loaded	active	active	System Time Synchronized
timers.target	loaded	active	active	Timers
umount.target	loaded	inactive	dead	Unmount All Filesystems

Çalışan çalışmayan tüm sistem unitlerin listesini görmek için --all argümanı kullanılmalıdır.

Systemctl list-units -all

--state argümanı ile filtreleme yaparak çıktılar elde edebilirsiniz.

Systemctl unmask servis_ismi komutuyla masked olmuş bir birimi enabled hale getirebilirsiniz.

Systemd'nin yüklemeye çalıştığı tüm birim dosyalarını görmek için list-unit-files argümanı kullanılır.

UNIT FILE	STATE	VENDOR PRESET
proc-sys-fs-binfmt_misc.automount	static	enabled
.mount	generated	enabled
boot.mount	generated	enabled
dev-hugepages.mount	static	enabled
dev-mqueue.mount	static	enabled
home.mount	generated	enabled
proc-sys-fs-binfmt_misc.mount	disabled	enabled
snap-core18-1944.mount	enabled	enabled
snap-core18-1988.mount	enabled	enabled
snap-lxd-16099.mount	enabled	enabled
snap-lxd-19188.mount	enabled	enabled
snap-snapd-10707.mount	enabled	enabled
snap-snapd-11036.mount	enabled	enabled
sys-fs-fuse-connections.mount	static	enabled
sys-kernel-config.mount	static	enabled
sys-kernel-debug.mount	static	enabled
sys-kernel-tracing.mount	static	enabled

Service Unit

Systemd servisleri başlatmak için birim dosyaları kullanıldığından bahsetmiştim. Bu dosyalar /etc/systemd/system altında bulunur. Örneğin sshd.service için kullanılan dosyayı incelersek Unit, Service ve Install olarak 3 bölüm görürüz.

systemctl cat sshd

Unit, servisin çalışması için gereken bağımlılıkları içerir. After ifadesi sshd başlatılmadan önce network.target ve auditd.service in çalışması gerektiğini belirtir. Yani after bu iki servis başladıkten sonra başla demektir. Before ifadesi de yer alabilir. Before da belirtilen servisten önce çalış demektir.

Service, servisin nasıl başlatılıp durdurulacağını, izinleri tanımlar.

Install, birimin çalışma durumunu tanımlar.

```

ozgur@ubuntu:~$ systemctl cat sshd
# /lib/systemd/system/sshd.service
[Unit]
Description=OpenBSD Secure Shell server
Documentation=man:sshd(8) man:sshd_config(5)
After=network.target auditd.service
ConditionPathExists=!/etc/ssh/sshd_not_to_be_run

[Service]
EnvironmentFile=-/etc/default/ssh
ExecStartPre=/usr/sbin/sshd -t
ExecStart=/usr/sbin/sshd -D $SSH_OPTS
ExecReload=/usr/sbin/sshd -t
ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
Restart=on-failure
RestartPreventExitStatus=255
Type=notify
RuntimeDirectory=sshd
RuntimeDirectoryMode=0755

[Install]
WantedBy=multi-user.target
Alias=sshd.service
ozgur@ubuntu:~$ █

```

Unit dosyalarının içerisinde **nano** ile değişiklik yapabilirsiniz. Unitlerin dosya yolu **systemctl cat unit_adi** komut çıktısının ilk satırında belirtilir. Değişiklik yaptıktan sonra **systemctl daemon-reload** komutuyla Unit'leri yeniden yüklenir.

Systemctl birimlerinin çoğu durumda bağımlılıkları vardır. Bazı birimler, diğer birimlere bağlı olarak başlatılır ve belirli bir birimin talep edildiği bir olay, başka bir birimin başlamasını tetikleyebilir. Servisin bağlı olduğu bağımlılık listesine ulaşmak için komutumuz aşağıdaki gibidir.

systemctl list-dependencies sshd

```

ozgur@ubuntu:~$ systemctl list-dependencies sshd
sshd.service
└─.mount
   └─system.slice
      └─sysinit.target
         ├─apparmor.service
         ├─blk-availability.service
         ├─dev-hugepages.mount
         ├─dev-mqueue.mount
         ├─finalrd.service
         ├─keyboard-setup.service
         ├─kmod-static-nodes.service
         ├─lvm2-lvmpolld.socket
         ├─lvm2-monitor.service
         └─multipathd.service

```

Bağımlilikler dışında, bazı birimlerin diğer birimlerle çatışmaları (Conflicts) vardır. Ufw ve Firewalld, crond ve ntpd gibi servisler birbirleriyle çakıştığı için beraber çalışmazlar. Bu servislerin birbirlerine bağlılığı olarak örnek gösterbiliriz.

Bir birimin özelliklerini görmek için show argümanı kullanılır.

```
root@ubuntu:~# systemctl show sshd
Type=notify
Restart=on-failure
NotifyAccess=main
RestartUSec=100ms
TimeoutStartUSec=1min 30s
TimeoutStopUSec=1min 30s
TimeoutAbortUSec=1min 30s
RuntimeMaxUSec=infinity
```

Target Unit

Target'lar servislerin çalışma durumlarını belirleyen, birçok servis unitden oluşan ve unitlerin doğru sırada ve zamanda yüklenmesini sağlayan birimlerdir. Bir target'ın en basit tanımı unitlerden oluşan gruptur. **systemctl enable** ve **disable** komutlarıyla servislerin otomatik olarak hangi hedefleri çalışması gereği ayarlanır ve böylece siz bir servis için hangi servis için hangi hedef çalıştırılması diye düşünmezsiniz.

```
root@ubuntu:~# systemctl cat multi-user.target
# /lib/systemd/system/multi-user.target
# SPDX-License-Identifier: LGPL-2.1+
#
# This file is part of systemd.
#
# systemd is free software; you can redistribute it and/or modify it
# under the terms of the GNU Lesser General Public License as published by
# the Free Software Foundation; either version 2.1 of the License, or
# (at your option) any later version.

[Unit]
Description=Multi-User System
Documentation=man:systemd.special(7)
Requires=basic.target
Conflicts=rescue.service rescue.target
After=basic.target rescue.service rescue.target
AllowIsolate=yes
```

Çalışan hedefleri görmek için **systemctl --type=target** komutu kullanılır.

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
basic.target	loaded	active	active	Basic System
cloud-config.target	loaded	active	active	Cloud-config availability
cloud-init.target	loaded	active	active	Cloud-init target
cryptsetup.target	loaded	active	active	Local Encrypted Volumes
getty.target	loaded	active	active	Login Prompts
graphical.target	loaded	active	active	Graphical Interface
local-fs-pre.target	loaded	active	active	Local File Systems (Pre)
local-fs.target	loaded	active	active	Local File Systems
multi-user.target	loaded	active	active	Multi-User System
network-online.target	loaded	active	active	Network is Online
network-pre.target	loaded	active	active	Network (Pre)
network.target	loaded	active	active	Network
nss-lookup.target	loaded	active	active	Host and Network Name Lookups
nss-user-lookup.target	loaded	active	active	User and Group Name Lookups
paths.target	loaded	active	active	Paths
remote-fs-pre.target	loaded	active	active	Remote File Systems (Pre)
remote-fs.target	loaded	active	active	Remote File Systems
slices.target	loaded	active	active	Slices
sockets.target	loaded	active	active	Sockets
swap.target	loaded	active	active	Swap
sysinit.target	loaded	active	active	System Initialization
time-set.target	loaded	active	active	System Time Set
time-sync.target	loaded	active	active	System Time Synchronized
timers.target	loaded	active	active	Timers

Recovery Menü

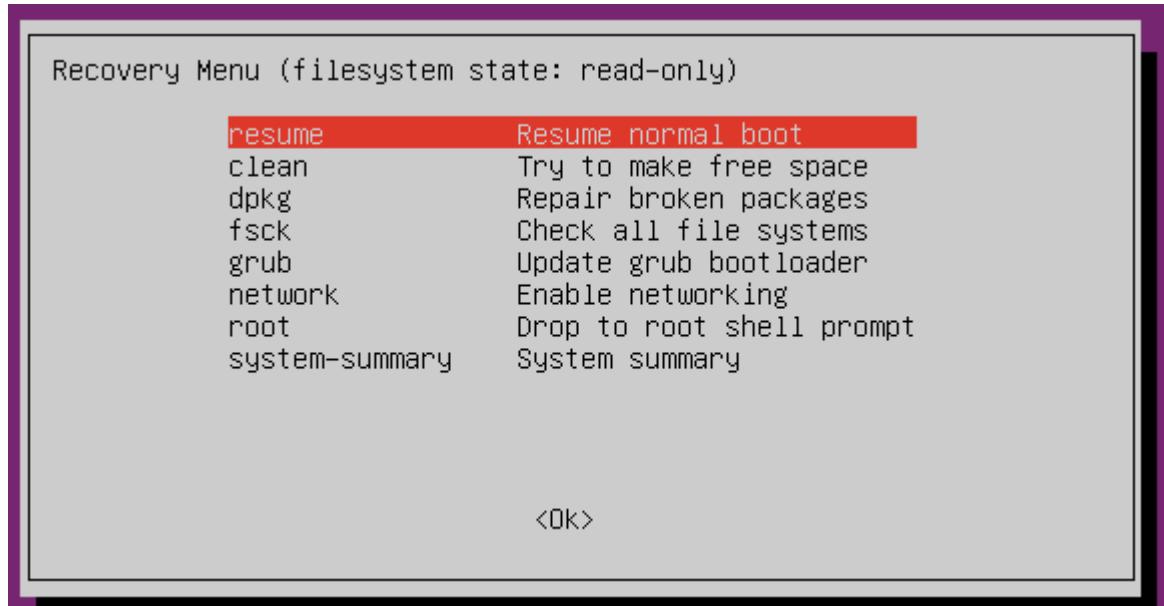
Recovery menü, sunucu açılışında meydana gelecek problemlere çözüm sağlamak için kullanılır. Elbette üst seviye problemler için burada anlatılanlardan farklı araçlar ve yöntemler kullanılmaktadır. Bu konuda sunucumuzun GRUB menüsünden Advanced Options bölümündeki araçları inceleyeceğiz. GRUB menüsü açılışında klavyenin tuşlarıyla Advanced Options for Ubuntu seçeneği ile kernel listesinin bulunduğu alana geçiş yapıyoruz.



Kernel menüsünde, sistemi ilk yüklediğinizde sadece bir kernel ve recovery mode'u olur. En son yüklenen ve güncel olan kernel en üst kısımda olur. Eğer kernel update'inden sonra bir problem yaşarsanız bir önceki kernel'den sistemi başlatabilir veya recovery mode'u kullanarak onarım seçeneklerini deneyebilirsiniz.

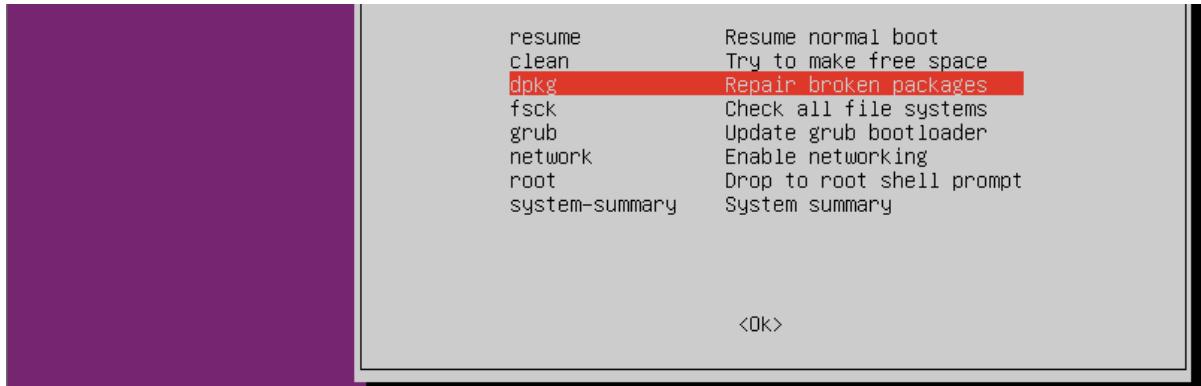


Recovery Menü alanındaki araçlardan bahsedecek olursak;



clean : Sistem bölümünde boş alan açmak için kullanılır. Bu işlem için apt autoremove komutunu kullanarak kullanılmayan paketleri, güncellemeleri silerek yer açmaya çalışır. Aynı işi bakım modunda root olarak shell'e düşüp siz de yapabilirsiniz.

dpkg : Sistemde probleme neden olan paketleri kaldırılması veya yeniden yüklenmesi için kullanılır. Bu özelliği kullanmak için internet bağlantısı gereklidir. Recovery Menüsündeki network bölümünden erişim ayalarını düzenleyebilirsiniz.



```
Reading cache
Reading package lists... Done
Building dependency tree
Reading state information... Done

Calculating the changes

Do you want to start the upgrade?

1 new package is going to be installed.

You have to download a total of 41.5 k. This download will take about
1 second with a 1Mbit DSL connection and about 5 seconds with a 56k
modem.

 Continue [yN] Details [d]
```

fsck : Disk üzerinde oluşacak hatalı alanları düzeltmek için kullanılan çok yararlı bir araçtır.

grub : GRUB menüsünü günceller.

root : root yetkisi ile bakım modundan shell'e düşersiniz. Böylece yaşadığınız problemle ilgili araştırma ve problemi gidermek için düzenlemeler yapma şansınız olur.

system-summary : Sistem üzerindeki dosya sistemi, disk, memory, network hakkında bilgi verir.

Root Parolası Resetleme

Sunucunuzun root şifresini unutmuş olabilirsiniz. Root şifresini değiştirmek için sunucumuzu yeniden başlatğı GRUB menüsüne ulaşıyoruz.



GRUB parametrelerini düzenlemek için 'e' ye basıyoruz. Yön tuşlarıyla alt satırında linux ile başlayan satırın sonundaki ro (read-only) olan bölümü siliyoruz.

```
GNU GRUB version 2.04

insmod part_gpt
insmod ext2
set root='hd0,gpt2'
if [ $feature_platform_search_hint = xy ]; then
    search --no-floppy --fs-uuid --set=root --hint-bios=hd0,gpt2 --hint-efi=hd0,gpt2 --hint-baremetal=ahci0,gpt2 f4c6fc21-29d9-4589-8e71-e7abac23cf7b
else
    search --no-floppy --fs-uuid --set=root f4c6fc21-29d9-4589-8e71-e7abac23cf7b
fi
linux /vmlinuz-5.4.0-42-generic root=/dev/mapper/ubuntu--vg-ubuntu--lv ro maybe-ubiquity_
initrd /initrd.img-5.4.0-42-generic

↑
```

Minimum Emacs-like screen editing is supported. TAB lists completions. Press Ctrl-x or F10 to boot, Ctrl-c or F2 for a command-line or ESC to discard edits and return to the GRUB menu.

Ro olan kısmını rw init=/bin/bash olarak değiştiriyoruz.

```
GNU GRUB version 2.04

insmod part_gpt
insmod ext2
set root='hd0,gpt2'
if [ $feature_platform_search_hint = xy ]; then
    search --no-floppy --fs-uuid --set=root --hint-bios=hd0,gpt2 --hint-efi=hd0,gpt2 --hint-baremetal=ahci0,gpt2 f4c6fc21-29d9-4589-8e71-e7abac23cf7b
else
    search --no-floppy --fs-uuid --set=root f4c6fc21-29d9-4589-8e71-e7abac23cf7b
fi
linux /vmlinuz-5.4.0-42-generic root=/dev/mapper/ubuntu--vg-ubuntu--lv rw init=/bin/bash_
initrd /initrd.img-5.4.0-42-generic

↑
```

Minimum Emacs-like screen editing is supported. TAB lists completions. Press Ctrl-x or F10 to boot, Ctrl-c or F2 for a command-line or ESC to discard edits and return to the GRUB menu.

Ctrl+X veya F10 ile değişikliklerin uygulanmasını sağlıyoruz. Boot işlemi devam ediyor ve bash ekranına düşüyoruz.

```
[ 8.164977] [drm] Initialized vmwgfx 2.15.0 20180704 for 0000:00:02.0 on minor 0
[ 9.097227] e1000 0000:00:03.0 eth0: (PCI:33MHz:32-bit) 08:00:27:9a:f1:cc
[ 9.097271] e1000 0000:00:03.0 eth0: Intel(R) PRO/1000 Network Connection
[ 9.100290] e1000 0000:00:03.0 enp0s3: renamed from eth0
Begin: Loading essential drivers ... [  9.325533] raid6: sse2x4  gen()  8167 MB/s
[ 9.389304] raid6: sse2x4  xor()  5444 MB/s
[ 9.445398] raid6: sse2x2  gen()  8433 MB/s
[ 9.497528] raid6: sse2x2  xor()  5387 MB/s
[ 9.561958] raid6: sse2x1  gen()  6183 MB/s
[ 9.625374] raid6: sse2x1  xor()  4140 MB/s
[ 9.625402] raid6: using algorithm sse2x2 gen() 8433 MB/s
[ 9.626044] raid6: ... xor() 5387 MB/s, rmw enabled
[ 9.626593] raid6: using ssse3x2 recovery algorithm
[ 9.650696] xor: measuring software checksum speed
[ 9.753733] prefetch64-sse: 30739.000 MB/sec
[ 9.861434] generic_sse: 34349.000 MB/sec
[ 9.862026] xor: using function: generic_sse (34349.000 MB/sec)
[ 9.885166] async_tx: api initialized (async)
done.
Begin: Running /scripts/init-premount ... done.
Begin: Mounting root file system ... Begin: Running /scripts/local-top ... done.
Begin: Running /scripts/local-premount ... [ 10.061121] Btrfs loaded, crc32c=crc32c-intel
Scanning for Btrfs filesystems
done.
Warning: fsck not present, so skipping root file system
[ 10.296944] EXT4-fs (dm-0): mounted filesystem with ordered data mode. Opts: (null)
done.
Begin: Running /scripts/local-bottom ... done.
Begin: Running /scripts/init-bottom ... done.
bash: cannot set terminal process group (-1): Inappropriate ioctl for device
bash: no job control in this shell
root@(none):/# [ 123.954465] random: crng init done
[ 123.955060] random: 7 urandom warning(s) missed due to ratelimiting
root@(none):/#
```

mount | grep -w / komutunu çalıştırarak root dosya sisteminin okuma ve yazma erişim haklarına sahip olduğunu doğruluyoruz.

```
root@(none):/# mount | grep -w /
/dev/mapper/ubuntu--vg-ubuntu--lv on / type ext4 (rw,relatime)
```

passwd komutu ile şifremizi değiştiriyoruz.

```
root@(none):/#
root@(none):/# mount | grep -w /
/dev/mapper/ubuntu--vg-ubuntu--lv on / type ext4 (rw,relatime)
root@(none):/# passwd
New password:
Retype new password:
passwd: password updated successfully
```

Yaptığımız değişiklikleri uygulayarak GRUB menüsünün boot işleminin devam etmesini sağlıyoruz.

exec /sbin/init

```
root@(none):/# exec /sbin/init
```

Bu değişiklikten sonra root kullanıcısının erişimi açılır. Eğer normalde root kullanıcısını dışardan erişime kapattıysanız /etc/passwd içerisinde root kullanıcısının satır sonuna /nologin eklemeniz gereklidir.

Farkına varmadan kullanıcınızın sudoers dosyasından root yetkisini kaldırın. Root'un da parolası yok. Aynı yöntemle root parolası oluşturmayıp kullanıcınızı sudoers altına ekleyerek sunucuya erişebilirsiniz.

13 - Kernel Yönetimi

Kernel, bir işletim sistemi içinde yer alan süreçleri ve süreç etkileşimlerini yöneten karmaşık bir yazılım parçasıdır. Kullanıcıların kernel ile pek işleri olmamakla birlikte Kernel'in yönettiği uygulamalarla çalışır.

Linux Kernel Nedir?

Linux çekirdeği, işletim sisteminin kalbidir. Kullanıcının çalıştığı shell ortamı ile donanım arasındaki katmandır. Çekirdek aynı zamanda temel işletim sistemi görevlerini yerine getirir. Kernel tarafından gerçekleştirilen işletim sistemi görevleri, farklı kernel iş parçacıkları tarafından gerçekleştirilir. Kernel iş parçacıkları ps aux komutunda bulunan köşeli parantez içindedir.

```
ozgur@ubuntu:~$ ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root        1  1.0  1.1 102352 11796 ?          Ss  18:05  0:06 /sbin/init ma
root        2  0.0  0.0     0     0 ?          S   18:05  0:00 [kthreadd]
root        3  0.0  0.0     0     0 ?          I<  18:05  0:00 [rcu gp]
```

Linux çekirdeğinin bir diğer önemli görevi de donanımın başlatılmasıdır. Her donanım parçası belirli özellikler içerir ve bu özelliklerini kullanmak için de bir sürücünün (driver) yüklenmesi gereklidir. Eğer uygun olmayan bir sürücü sisteme yüklenirse kernel düzgün bir şekilde çalışmamayabilir. Problemi gidermek için sürücüyü sistemden kaldırmanız gereklidir.

Linux çekirdeğinin kaynak kodu (source code), çekirdek kaynak ağacı (kernel source code) adı verilen bir grup dizinde tutulur. Çekirdek kaynak ağacının yapısı önemlidir çünkü çekirdeği derleme (compile)(oluşturma) süreci otomatize edilmiştir ve bu otomasyon make aracı tarafından yorumlanan komut dosyaları tarafından kontrol edilir. Makefiles olarak bilinen bu komut dosyaları, çekirdek kodunun parçalarını belirli yerlerde bulmayı beklerler, eğer bulamazlarsa çalışmazlar. Kernel kaynak dosyaları /usr/src/Linux-headers~ altında bulunur. Genelde kernel ile programcılar ve uygulama geliştiriciler çalışır.

```
ozgur@ubuntu:~$ ls -ltr /usr/src/linux-headers-5.4.0-65
total 168
-rw-r--r-- 1 root root 1321 Nov 25 2019 Kbuild
-rw-r--r-- 1 root root 62408 Jan 18 19:31 Makefile
-rw-r--r-- 1 root root 620 Jan 18 19:31 Kconfig
drwxr-xr-x 27 root root 4096 Feb 2 23:00 include
drwxr-xr-x 14 root root 4096 Feb 2 23:00 Documentation
drwxr-xr-x 27 root root 4096 Feb 2 23:00 arch
drwxr-xr-x 137 root root 4096 Feb 2 23:00 drivers
drwxr-xr-x 4 root root 4096 Feb 2 23:00 crypto
drwxr-xr-x 2 root root 4096 Feb 2 23:00 certs
drwxr-xr-x 3 root root 4096 Feb 2 23:00 block
drwxr-xr-x 77 root root 4096 Feb 2 23:00 fs
drwxr-xr-x 70 root root 4096 Feb 2 23:00 net
drwxr-xr-x 3 root root 4096 Feb 2 23:00 mm
drwxr-xr-x 19 root root 4096 Feb 2 23:00 lib
drwxr-xr-x 17 root root 4096 Feb 2 23:00 kernel
drwxr-xr-x 2 root root 4096 Feb 2 23:00 ipc
drwxr-xr-x 2 root root 4096 Feb 2 23:00 init
drwxr-xr-x 29 root root 4096 Feb 2 23:00 samples
drwxr-xr-x 33 root root 4096 Feb 2 23:00 tools
drwxr-xr-x 26 root root 4096 Feb 2 23:00 sound
drwxr-xr-x 12 root root 4096 Feb 2 23:00 security
drwxr-xr-x 15 root root 12288 Feb 2 23:00 scripts
drwxr-xr-x 4 root root 4096 Feb 2 23:00 virt
drwxr-xr-x 3 root root 4096 Feb 2 23:00 usr
drwxr-xr-x 5 root root 4096 Feb 2 23:00 ubuntu
ozgur@ubuntu:~$
```

Kernel kurulduğunda, bu dosyalar otomatik olarak yerleştirilir. Kernel kaynaklarına yama (patch) geçtiğinizde, bu dosyalar otomatik olarak değiştirilir. Genel itibariyle sistem yöneticilerinin kernel ile ilgili değişiklik veya müdahalelesine gerek yoktur.

Donanımlar, aygit sürücülerini tarafından kontrol edilir. Sürücüler ise donanımlara, kullanıcı ile nasıl iletişim kurulacağını söyler. Kernel, kendisinin bir parçası olarak derlenmiş sürücülere sahiptir ve ihtiyaç olan sürücüler modül (module) olarak derlenir. Önyükleme işlemi sırasında yalnızca derlenen aygit sürücülerini kullanılır; modül (module) sürücüler sistem başlatıldıktan sonra kullanılabilir.

Kernel Modüllerini Yönetme

Modüllerin, donanımların ihtiyacı olan sürücülerin derlenmesiyle oluştuğunu öğrendik. Normalde modüller otomatik olarak yüklenir ancak bazı durumlarda bir modülü izleme, yükleme ve kaldırma görevlerini yönetmek için özel araçlar kullanmamız gerekebilir. Genellikle sisteme eklenen yeni donanım için bir sürücü modülüne ihtiyaç duyulur.

lsmod : Yüklü olan modülleri gösterir. Modül listesi uzun olduğundan pipe ile komut arkasına less koymak ekran çıktısını takip etmemi kolaylaştırır. Çıktıda sırasıyla modülün ismi, boyutu, başka modüller tarafından kullanılma sayısı ve kullanan modülün ismini gösterir. Modüller **/sys/module/** dizini altında tutulur.

```
ozgur@ubuntu:~$ lsmod|less
Module           Size  Used by
dm_multipath    32768  0
scsi_dh_rdac    16384  0
scsi_dh_emc     16384  0
scsi_dh_alua    20480  0
```

Bir modül dizini içerisinde farklı erişim yetkilerine sahip dizin ve dosyalar bulunur.

```
root@ubuntu:/sys/module# cd e1000
root@ubuntu:/sys/module/e1000# ll
total 0
drwxr-xr-x  7 root root   0 Feb 15 23:32 .
drwxr-xr-x 157 root root   0 Feb 15 23:32 ..
-r--r--r--  1 root root 4096 Feb 15 23:36 coresize
drwxr-xr-x  2 root root   0 Feb 15 23:39 drivers/
drwxr-xr-x  2 root root   0 Feb 15 23:36 holders/
-r--r--r--  1 root root 4096 Feb 15 23:39 initsize
-r--r--r--  1 root root 4096 Feb 15 23:33 initstate
drwxr-xr-x  2 root root   0 Feb 15 23:39 notes/
drwxr-xr-x  2 root root   0 Feb 15 23:39 parameters/
-r--r--r--  1 root root 4096 Feb 15 23:36 refcnt
drwxr-xr-x  2 root root   0 Feb 15 23:39 sections/
-r--r--r--  1 root root 4096 Feb 15 23:39 srcversion
-r--r--r--  1 root root 4096 Feb 15 23:39 taint
--w-----  1 root root 4096 Feb 15 23:32 uevent
-r--r--r--  1 root root 4096 Feb 15 23:39 version
```

Sunucumuzda varolan network adaptörünün driver'ını bulmak istersek **/sys/class/net/enp0s3/device/** yolunu izlemeliyiz.

```
root@ubuntu:/sys/class/net/enp0s3/device# ll
total 0
drwxr-xr-x 5 root root 0 Feb 15 23:32 .
drwxr-xr-x 15 root root 0 Feb 15 23:32 ..
-r--r--r-- 1 root root 4096 Feb 15 23:32 ari_enabled
-rw-r--r-- 1 root root 4096 Feb 15 23:33 broken_parity_status
-r--r--r-- 1 root root 4096 Feb 15 23:33 class
-rw-r--r-- 1 root root 256 Feb 15 23:32 config
-r--r--r-- 1 root root 4096 Feb 15 23:33 consistent_dma_mask_bits
-rw-r--r-- 1 root root 4096 Feb 15 23:33 d3cold_allowed
-r--r--r-- 1 root root 4096 Feb 15 23:32 device
-r--r--r-- 1 root root 4096 Feb 15 23:33 dma_mask_bits
lrxwxrwxrwx 1 root root 0 Feb 15 23:32 driver -> ../../bus/pci/drivers/e1000/
```

modprobe : Modprobe ile modül yükleme işlemi otomatik olarak gerçekleştirilir. Modül kaldırmak için -r argümanı kullanılır.

```
root@ubuntu:/sys/module# lsmod | grep st
vboxguest 348160 0
zstd_compress 167936 1 btrfs
root@ubuntu:/sys/module# modprobe st
root@ubuntu:/sys/module# lsmod | grep st
st 61440 0
vboxguest 348160 0
zstd_compress 167936 1 btrfs
root@ubuntu:/sys/module# modprobe -r st
root@ubuntu:/sys/module# lsmod | grep st
vboxguest 348160 0
zstd_compress 167936 1 btrfs
```

modinfo : Modül hakkında bilgi verir. Modülün driver bilgisini bulmak için faydalıdır. Filename bölümü modülün driver'ının yolunu gösterir. **modinfo -p e1000** komutu modülün parametrelerini getirir.

```
ozgur@ubuntu:~$ modinfo e1000|less
filename: /lib/modules/5.4.0-65-generic/kernel/drivers/net/ethernet/intel/e1000/e1000.ko
version: 7.3.21-k8-NAPI
license: GPL v2
description: Intel(R) PRO/1000 Network Driver
author: Intel Corporation, <linux.nics@intel.com>
srcversion: FCB88217EDA1AA26ACC1A02
alias: pci:v00008086d00002E6Esv*sd*bc*sc*i*
```

insmod : Çalışan kernel'e bir module yüklemek için kullanılır.

rmmmod : Modülü kaldırmak için kullanılır.

depmod : Bazı modüllerin çalışmak için başka modüllere ihtiyaç duyabilir. Bu komut modüller için bir bağımlılık dosyası oluşturur.

Komut	Açıklama
lsmod	Sistemde çalışan modüller gösterir.
insmod	Kernel'e modül yüklemek için kullanılır.
rmmmod	Kernel'den modül kaldırmak için kullanılır.
depmod	Modüller'in başka modüllerle çalışmasını sağlar.
modprobe	Modül yükleme işlemlerini otomatik gerçekleştirir.
modinfo	Modüller hakkında bilgi verir.

/etc/modprobe.conf : Kernel modül değişkenlerini içeren modprobe ve depmod control dosyasıdır.

Kernel içindeki modüllere ulaşmak için **/usr/lib/modules/5.4.0-65-generic/kernel/** dizinine gitmek gereklidir. Burada modüller çeşitlerine göre dizinlere ayrılmıştır. Modinfo komutuyla bir modülün driver'ını ararsak bu dizine geleceğiz.

```
root@ubuntu:/usr/lib/modules/5.4.0-65-generic/kernel# ll
total 68
drwxr-xr-x 17 root root 4096 Feb 2 23:00 .
drwxr-xr-x  5 root root 4096 Feb 2 23:01 ..
drwxr-xr-x  3 root root 4096 Feb 2 23:00 arch/
drwxr-xr-x  2 root root 4096 Feb 2 23:00 block/
drwxr-xr-x  4 root root 4096 Feb 2 23:00 crypto/
drwxr-xr-x 103 root root 4096 Feb 2 23:00 drivers/
drwxr-xr-x  54 root root 4096 Feb 2 23:00 fs/
drwxr-xr-x  3 root root 4096 Feb 2 23:00 kernel/
drwxr-xr-x 10 root root 4096 Feb 2 23:00 lib/
drwxr-xr-x  2 root root 4096 Feb 2 23:00 mm/
drwxr-xr-x  59 root root 4096 Feb 2 23:00 net/
drwxr-xr-x  3 root root 4096 Feb 2 23:00 samples/
drwxr-xr-x 15 root root 4096 Feb 2 23:00 sound/
drwxr-xr-x  4 root root 4096 Feb 2 23:00 ubuntu/
drwxr-xr-x  2 root root 4096 Feb 2 23:00 virtualbox-guest/
drwxr-xr-x  2 root root 4096 Feb 2 23:00 wireguard/
drwxr-xr-x  2 root root 4096 Feb 2 23:00 zfs/
```

Modülde Değişiklik Yapmak

Bir modülde değişiklik yapmak için modprobe aracını kullandığımızdan bahsettiğimiz Video isimli bir modülün allow_duplicates isimli parametresini değiştireceğiz.

```
cd /sys/module/video/parameters #Parametrelerin olduğu dizini gidiyoruz.
```

```
ls -ltr #Parametreleri görüntüleyiyoruz.
```

```
cat allow_duplicates #Değişiklik yapacağımız parameter değerine bakıyoruz.
```

```
lsmod |grep video #Modülün yüklü olma durumuna bakıyoruz.
```

```
modprobe -r video #Modül yüklü ise durduruyoruz.
```

```
modprobe -v video allow_duplicates=Y #Parametre değerini değiştiriyoruz.
```

```
modprobe video #Modülü tekrar yükliyoruz.
```

```

root@ubuntu:/sys/module/video# cd parameters/
root@ubuntu:/sys/module/video/parameters# ll
total 0
drwxr-xr-x 2 root root 0 Feb 16 00:04 .
drwxr-xr-x 6 root root 0 Feb 15 23:32 ..
-rw-r--r-- 1 root root 4096 Feb 16 01:08 allow_duplicates
-rw-r--r-- 1 root root 4096 Feb 16 01:08 brightness_switch_enabled
-rw-r--r-- 1 root root 4096 Feb 16 01:08 device_id_scheme
-rw-r--r-- 1 root root 4096 Feb 16 01:08 disable_backlight_sysfs_if
-rw-r--r-- 1 root root 4096 Feb 16 01:08 hw_changes_brightness
-rw-r--r-- 1 root root 4096 Feb 16 01:08 only_lcd
-rw-r--r-- 1 root root 4096 Feb 16 01:08 report_key_events
root@ubuntu:/sys/module/video/parameters# cat allow_duplicates
N
root@ubuntu:/sys/module/video/parameters# lsmod |grep video
video buf2_vmalloc 20480 1 usbtv
video buf2_memops 20480 1 video buf2_vmalloc
video buf2_v412 24576 1 usbtv
video buf2_common 49152 2 usbtv,video buf2_v412
video dev 225280 3 usbtv,video buf2_v412,video buf2_common
mc 53248 3 video dev,video buf2_v412,video buf2_common
video 49152 0
root@ubuntu:/sys/module/video/parameters# modprobe -r video
root@ubuntu:/sys/module/video/parameters# modprobe -v video allow_duplicates=Y
insmod /lib/modules/5.4.0-65-generic/kernel/drivers/acpi/video.ko allow_duplicates=Y
root@ubuntu:/sys/module/video/parameters# modprobe video
root@ubuntu:/sys/module/video/parameters# cat allow_duplicates
Y

```

/etc/modprobe.d dizini altına oluşturulacak bir conf dosyası ile de modül üzerinde değişiklik yapılabilir. Aşağıdaki örnekte **usbtv** isimli bir modülün içerisinde bir conf dosyası kullanarak driver'in içerisinde **test_value=8** isimli bir parameter yerleştirdik.

```

root@ubuntu:~# modinfo -p usbtv
root@ubuntu:~# lsmod|grep usbt
root@ubuntu:~# cd /etc/modprobe.d/
root@ubuntu:/etc/modprobe.d# ll
total 44
drwxr-xr-x 2 root root 4096 Feb 8 21:09 .
drwxr-xr-x 111 root root 4096 Feb 14 17:53 ..
-rw-r--r-- 1 root root 154 Feb 16 2020 amd64-microcode-blacklist.conf
-rw-r--r-- 1 root root 325 Mar 12 2020 blacklist-ath_pci.conf
-rw-r--r-- 1 root root 1518 Mar 12 2020 blacklist.conf
-rw-r--r-- 1 root root 210 Mar 12 2020 blacklist-firmware.conf
-rw-r--r-- 1 root root 677 Mar 12 2020 blacklist-framebuffer.conf
-rw-r--r-- 1 root root 583 Mar 12 2020 blacklist-rare-network.conf
-rw-r--r-- 1 root root 154 Nov 12 02:24 intel-microcode-blacklist.conf
-rw-r--r-- 1 root root 347 Mar 12 2020 iwlwifi.conf
-rw-r--r-- 1 root root 379 Jan 23 2020 mdadm.conf
root@ubuntu:/etc/modprobe.d# nano usbtv.conf
root@ubuntu:/etc/modprobe.d# modinfo -p usbtv
root@ubuntu:/etc/modprobe.d# nano usbtv.conf
root@ubuntu:/etc/modprobe.d# modprobe -v usbtv
insmod /lib/modules/5.4.0-65-generic/kernel/drivers/media/mc/mc.ko
insmod /lib/modules/5.4.0-65-generic/kernel/drivers/media/v4l2-core/video dev.ko
insmod /lib/modules/5.4.0-65-generic/kernel/drivers/media/common/video buf2/video buf2-common.ko
insmod /lib/modules/5.4.0-65-generic/kernel/drivers/media/common/video buf2/video buf2-v412.ko
insmod /lib/modules/5.4.0-65-generic/kernel/drivers/media/common/video buf2/video buf2-memops.ko
insmod /lib/modules/5.4.0-65-generic/kernel/drivers/media/common/video buf2/video buf2-vmalloc.ko
insmod /lib/modules/5.4.0-65-generic/kernel/drivers/media/usb/usbtv/usbtv.ko test_value=8

```

modprobe -r usbtv ile modülü kaldırıp oluşturduğumuz conf dosyasını sildiğimizde module varsayılan ayarlarına geri döner. Yukarıdaki örneği eğitim maksatlı yaptık. Modüller üzerinde değişiklik yaparken dikkatli olunmalı, sisteminizdeki donanımları hangi driver'ı kullandığı iyi bilinmeli ve değişiklik yapılacak modül hakkında detaylı bilgi edinildikten sonra işlemlere geçilmelidir.

Kernel Versiyonunu Anlama

Kernel'e yeni özellikler eklendikçe ve hatalar giderildikçe yeni versiyonlar çıkar. Versiyon numaları yapılan değişikliklerin sırasını takip etmek için vardır. Sahip olduğunuz yazılım,

çıkan yeni versiyonla daha iyi çalışacak diye bir garanti yoktur. Yeni versiyonun artılarını ve eksilerini inceleyip anladıkta sonra sisteminize uygulamanız en doğrusudur.

Kernel versiyon numarası 5 bölüme ayrıılır.

cat /proc/version_signature

```
root@ubuntu:/tmp# cat /proc/version_signature
Ubuntu 5.4.0-65.73-generic 5.4.78
```

5.4 Kernel versiyonunu tanımlar.

0 upstream kernel versiyonlarını tanımlamak için kullanılan eski bir parametredir.

-65 Bu kernel için ABI (Application Binary Interface) versiyonudur.

.15 Bu kernel için upload sayısıdır.

5.4.78 Kernel'in ana versiyon numarasıdır.

Kernel Versiyonunu Güncelleme

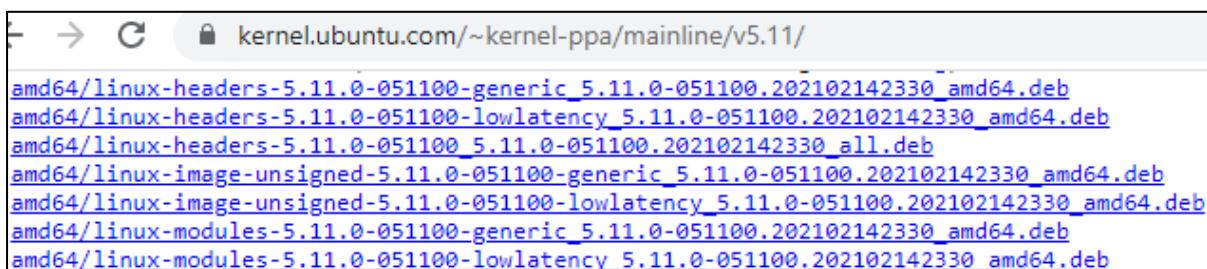
İlk olarak yapacağınız ister kernel ister işletim sistemi güncellemesi olsun, sunucunuzun yedeğini kesinlikle alın. Önemli dosyalarınızı yedekleyin. Mümkünse bir snapshot'ını alın. Sistemin bir daha ayağa kalkmaması gibi bir durumda bu yedekler hayat kurtarır. Yedeksiz iş yapmayın. Önce emniyet sonra hareket :)

Sunucunun kernel versiyonunu kontrol ediyoruz.

```
ozgur@ubuntu:~$ cat /proc/version_signature
Ubuntu 5.4.0-65.73-generic 5.4.78
```

Linux Kernel versiyonları açık kaynak olması nedeniyle herkesin kullanımına Ücretsiz sunulmuştur. Eğer var olan kernel versiyonunuza güncellemek isterseniz

<https://kernel.ubuntu.com/~kernel-ppa/mainline/> adresinden sistemimize uygun kernel'i indiriyoruz. Burda all ve generic paketleri indiriyoruz.



Bağlantı adreslerini kernel.txt isimli bir dosyaya kopyalıyoruz.

```
GNU nano 4.8                                     kernel.txt
https://kernel.ubuntu.com/~kernel-ppa/mainline/v5.11/amd64/linux-headers-5.11.0-051100-generic>
https://kernel.ubuntu.com/~kernel-ppa/mainline/v5.11/amd64/linux-headers-5.11.0-051100_5.11.0->
https://kernel.ubuntu.com/~kernel-ppa/mainline/v5.11/amd64/linux-image-unsigned-5.11.0-051100->
https://kernel.ubuntu.com/~kernel-ppa/mainline/v5.11/amd64/linux-modules-5.11.0-051100-generic>
```

5.5.1 kernel versiyonunu ait headers, image ve modulleri **wget -i kernel.txt** ile sunucumuza indiriyoruz. **-i** argümanı txt dosyasının içindeki tüm linklerdeki veriyi indirecek.

```
root@ubuntu:/tmp# wget -i kernel.txt
--2021-02-15 12:44:11-- https://kernel.ubuntu.com/~kernel-ppa/mainline/v5.11/amd64/linux-headers-5.11.0-051100-generic_5.11.0-051100.202102142330_amd64.deb
Resolving kernel.ubuntu.com (kernel.ubuntu.com) ... 91.189.94.216
Connecting to kernel.ubuntu.com (kernel.ubuntu.com)|91.189.94.216|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1241808 (1.2M) [application/x-debian-package]
Saving to: 'linux-headers-5.11.0-051100-generic_5.11.0-051100.202102142330_amd64.deb'

linux-headers-5.11.0-05 100%[=====] 1.18M 768KB/s in 1.6s
```

ls komutuyla dosyayı kontrol ediyoruz.

```
root@ubuntu:/tmp# ls
kernel.txt
linux-headers-5.11.0-051100_5.11.0-051100.202102142330_all.deb
linux-headers-5.11.0-051100-generic_5.11.0-051100.202102142330_amd64.deb
linux-image-unsigned-5.11.0-051100-generic_5.11.0-051100.202102142330_amd64.deb
linux-modules-5.11.0-051100-generic_5.11.0-051100.202102142330_amd64.deb
```

dpkg -i *.deb komutuyla kernel paketini yükliyoruz ve yükleme tamamlandıktan sonra sunucuyu reboot ediyoruz.

```
root@ubuntu:/tmp# dpkg -i *.deb
Selecting previously unselected package linux-headers-5.11.0-051100.
(Reading database ... 150489 files and directories currently installed.)
Preparing to unpack linux-headers-5.11.0-051100_5.11.0-051100.202102142330_all.deb ...
Unpacking linux-headers-5.11.0-051100 (5.11.0-051100.202102142330) ...
Selecting previously unselected package linux-headers-5.11.0-051100-generic.
Preparing to unpack linux-headers-5.11.0-051100-generic_5.11.0-051100.202102142330_amd64.deb .
.
Unpacking linux-headers-5.11.0-051100-generic (5.11.0-051100.202102142330) ...
```

Paket yüklemesi tamamlandıktan sonra sunucuyu reboot ediyoruz ve yeni kernel versiyonuna güncelleme işlemimiz tamamlanıyor.

```
root@ubuntu:~# cat /proc/version
Linux version 5.11.0-051100-generic (kernel@kathleen) (gcc (Ubuntu 10.2.0-13ubuntu1) 10.2.0, GL
U_ld (GNU Binutils for Ubuntu) 2.35.1) #202102142330 SMP Sun Feb 14 23:33:21 UTC 2021
```

Uyarı : Yaptığımız güncelleme tamamen eğitim maksatlı olup çalışan bir sistem üzerinde güncelleme yapmadan önce yüklenecek kernel ile nasıl değişiklikler oluyor, sisteminize nasıl etkileri olur şeklinde araştırmalar yapmanız gerekiyor. Aksi durumda tatsız durumlarla karşı karşıya kalabilirsiniz.

Eski Kernel Versiyonlarını Kaldırmak

dpkg --list | grep linux-image komutuyla sunucuda bulunan kernelleri görüntüleyiyoruz.

```
root@ubuntu:~# dpkg --list | grep linux-image
ii  linux-image-5.4.0-65-generic          5.4.0-65.73
ii  linux-image-generic                  5.4.0.65.68
ii  linux-image-unsigned-5.11.0-051100-generic 5.11.0-051100.202102142330
ii  linux-image-unsigned-5.5.1-050501-generic 5.5.1-050501.202002011032
ii  linux-image-unsigned-5.5.1-050501-lowlatency 5.5.1-050501.202002011032
```

cat /proc/version veya **uname -a** komutuyla sistemin çalıştığı kernel'i kontrol ediyoruz.

```
root@ubuntu:~# uname -a
Linux ubuntu 5.11.0-051100-generic #
root@ubuntu:~# cat /proc/version
Linux version 5.11.0-051100-generic
```

Dpkg çıktısındaki eski olan versiyonları **apt purge** komutuyla kaldırıyoruz.

```
root@ubuntu:~# apt purge linux-image-5.4.0-65-generic linux-image-generic linux-image-unsigned-5.5.1-050501-generic
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  amd64-microcode intel-microcode iucode-tool libdbus-glib-1-2 linux-headers-generic thermald
Use 'apt autoremove' to remove them.
The following additional packages will be installed:
  linux-image-unsigned-5.4.0-65-generic
Suggested packages:
  fdutils linux-doc | linux-source-5.4.0 linux-tools
The following packages will be REMOVED:
  linux-generic* linux-image-5.4.0-65-generic* linux-image-generic* linux-image-unsigned-5.5.1-050501-generic*
  linux-image-unsigned-5.5.1-050501-lowlatency* linux-modules-extra-5.4.0-65-generic*
```

Son olarak **dpkg --list | grep linux-image** komutuyla kernel durumunu kontrol ediyoruz.

```
root@ubuntu:~# dpkg --list | grep linux-image
ii  linux-image-unsigned-5.11.0-051100-generic 5.11.0-051100.202102142330
```

apt --purge autoremove komutuyla eski ve kullanılmayan paketleri kaldırabilirsiniz.

```
root@ubuntu:~# apt --purge autoremove
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  apache2-bin* apache2-data* apache2-utils* checkpolicy* fontconfig-config*
  fonts-dejavu-core* libapr1* libaprutil1* libaprutil1-dbd-sqlite3*
  libaprutil1-ldap* libfontconfig1* libgd3* libjansson4* libjbig0*
  libjpeg-turbo8* libjpeg8* liblua5.2-0* libnginx-mod-http-image-filter*
  libnginx-mod-http-xslt-filter* libnginx-mod-mail* libnginx-mod-stream*
  libtiff5* libwebp6* libxpm4* nginx-common* nginx-cote* python3-audit*
  python3-ipy* python3-selinux* python3-semanage* python3-sepolgen*
  python3-sepolicy* semodule-utils* ssl-cert*
0 upgraded, 0 newly installed, 34 to remove and 0 not upgraded.
After this operation, 18.5 MB disk space will be freed.
Do you want to continue? [Y/n] Y
```

Kernel Downgrade

İlk yöntem açılışta GRUB menüsündeki Advanced seçeneklerden sistemde yüklü olan bir kernel ile sistemi açabilirsiniz.

İkinci yöntem downgrade edeceğiniz kernel'i <https://kernel.ubuntu.com/~kernel-ppa/mainline/> adresinden indirip **dpkg -i *.deb** komutuyla kuruyoruz. Sistemi reboot edip yeniden başlangıç sırasında GRUB menüsünden yüklediğimiz kernel'i seçiyoruz.

İndireceğiniz kernel paketleri sisteminizde çalışan uygulamalar ile uyumlu olmalıdır.

Komutlar - uname, hostnamectl, dmesg

uname komutu işletim sistemi ve donanımı hakkında bilgi veren bir araçtır. Sırayla sayarsak Kernel adı, host ismi, kernel, kernel versiyonu, donanım ismi, işlemci yapısı, donanım platformu ve işletim sistemi bilgilerini döndürür.

uname -a

```
ozgur@ubuntu:~$ uname -a
Linux ubuntu 5.4.0-65-generic #73-Ubuntu SMP Mon Jan 18 17:25:17 UTC 2021 x86_64
x86_64 GNU/Linux
```

hostnamectl komutu uname komutuna göre işletim sistemi ve donanımı hakkında daha düzenli bir çıktı verir.

```
ozgur@ubuntu:~$ hostnamectl
      Static hostname: ubuntu
      Icon name: computer-vm
      Chassis: vm
      Machine ID: ac1a3ea39d564ceba3152b9085a964e6
      Boot ID: 8f72d4075fe94e5f980f01ce8a0acb72
  Virtualization: oracle
Operating System: Ubuntu 20.04.2 LTS
      Kernel: Linux 5.4.0-65-generic
Architecture: x86-64
```

Hostnamectl komutu set komutu alarak sunucu ismi, yer adı, şasi tipi bilgileri ayarlamانıza imkan tanır.

dmesg, /var/log/dmesg dizini altında tutulan kernel loglarının çıktısını ekrana döndürür. Bu loglar sistem üzerindeki servis ve donanım sürücülerinin durumlarını ve hatalarını gösterir. dmesg çıktısı ekrana ilk geldiğinde çok karmaşık gelebilir. Ancak aldığı argümanlar sayesinde aradığınızı bulmanız kolaylaşır.

Dmesg logları olaylara göre seviyelendirilebilir. Ayrıca dmesg komut çıktısındaki zaman bölümü (yeşil alan) bizler için çok anlamlı değildir. Anlayabileceğimiz bir zaman çevirmek için -T argümanı kullanılır.

Dmesg --level=emerg,alert,crit -T

```
root@ubuntu:~# dmesg --level=err,warn -T
[Thu Feb 11 18:05:30 2021] [Firmware Bug]: TSC doesn't count with P0 frequency!
[Thu Feb 11 18:05:36 2021] platform eisa.0: EISA: Cannot allocate resource for mainboard
[Thu Feb 11 18:05:36 2021] platform eisa.0: Cannot allocate resource for EISA slot 1
[Thu Feb 11 18:05:36 2021] platform eisa.0: Cannot allocate resource for EISA slot 2
[Thu Feb 11 18:05:36 2021] platform eisa.0: Cannot allocate resource for EISA slot 3
[Thu Feb 11 18:05:36 2021] platform eisa.0: Cannot allocate resource for EISA slot 4
[Thu Feb 11 18:05:36 2021] platform eisa.0: Cannot allocate resource for EISA slot 5
[Thu Feb 11 18:05:36 2021] platform eisa.0: Cannot allocate resource for EISA slot 6
[Thu Feb 11 18:05:36 2021] platform eisa.0: Cannot allocate resource for EISA slot 7
[Thu Feb 11 18:05:36 2021] platform eisa.0: Cannot allocate resource for EISA slot 8
[Thu Feb 11 18:05:37 2021] [drm:vmw_host_log [vmwgfx]] *ERROR* Failed to send host log message.
[Thu Feb 11 18:05:37 2021] [drm:vmw_host_log [vmwgfx]] *ERROR* Failed to send host log message.
[Thu Feb 11 18:05:43 2021] vboxguest: loading out-of-tree module taints kernel.
[Thu Feb 11 18:05:43 2021] vgdrvHeartbeatInit: Setting up heartbeat to trigger every 2000 milliseconds
[Thu Feb 11 18:05:43 2021] vboxguest: Successfully loaded version 6.1.10_Ubuntu
[Thu Feb 11 18:05:43 2021] vboxguest: misc device minor 58, IRQ 20, I/O port d040, MMIO at 00000000f0400000 (size 0x400000)
[Thu Feb 11 18:05:46 2021] ext4 filesystem being mounted at /boot supports timestamps until 2038 (0x7fffffff)
[Thu Feb 11 23:28:19 2021] kaudited_printk_skb: 20 callbacks suppressed
```

Belirli bir hizmete göre arayabilirsiniz.

dmesg --facility=syslog

```
root@ubuntu:~# dmesg --facility=syslog
[ 11.937393] systemd-journald[319]: Received client request to flush runtime journal.
```

Ne aradığınızı biliyorsanız **dmesg** ile birlikte **grep** komutu da çok işe yarar. Örneğimizde **enp0s3** isimli sunucu interface'sının loglarını görüyoruz.

```
root@ubuntu:~# dmesg | grep enp0s3
[ 8.367030] e1000 0000:00:03.0 enp0s3: renamed from eth0
[ 20.319777] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
[ 20.359039] IPv6: ADDRCONF(NETDEV_CHANGE): enp0s3: link becomes ready
[17681.636852] e1000: enp0s3 NIC Link is Down
[17683.657771] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control: RX
```

Donanım Tanımlama

BIOS (Basic Input/Output System)

Sunucunun sahip olduğu donanımın yapılandırılması ve işletim sisteminin başlatılacağı aygıtin seçimi için kullanılan basit arayüzlü bir uygulamadır. BIOS, işletim sistemini yüklemek veya çalıştmak için bir donanım üzerindeki küçük bir program olan bir önyükleme programını çalıştırır. Önyükleme programı işletim sisteminin bulunduğu yeri gösteren bir yapılandırma dosyasına sahiptir. Hatta ön yükleme programı, çoklu işletim sistemi bulunan cihazlarda bir menü oluşturularak (Örneğin GRUB 2) kullanıcıya istediği işletim sisteminin başlatması için imkân verilir.

BIOS, ön yükleme programının bulunduğu kaynağı harici veya dahili bir disk, USB bellek, CD-ROM veya network üzerinde bulunan bir sunucu olarak seçebilir. Bir sabit sürücüden önyükleme yaparken, ön yüklemenin yapılacakı sabit sürücüyü belirlemelisiniz, BIOS'un önyükleme programını buradan yüklemesi gereklidir. Bu işlem, Master Boot Record (MBR) tanımlanarak yapılır. MBR, sistemdeki ilk sabit disk bölümündeki ilk sektördedir. BIOS, MBR'yi arar ve burada depolanan ön yükleme programını bellekten okur. Önyükleme yükleyici programı, sistemde kurulu ayrı bir bölümde depolanan gerçek işletim sistemi çekirdek dosyasının konumunu gösterir.

UEFI (Unified Extensible Firmware Interface)

BIOS'un sahip olduğu tek bir diskte bir önyükleme programı, önyükleme program boyutunu düşük olması gibi sınırlamaları aşmak için UEFI geliştirilmiştir. Modern sunucu ve bilgisayar sistemlerinde UEFI bir standart haline gelmiştir. UEFI, herhangi bir boyuttaki önyükleme programıyla birlikte birden çok işletim sistemi için birden çok önyükleme yükleyici programı saklama becerisine sahiptir. önyükleme yükleyici programlarını depolamak için EFI Sistem Bölümü (ESP) adı verilen özel bir disk bölümünü belirler. ESP, önyükleme yükleyici programlarını depolamak için Windows İşletim Sistemlerinde eski File Allocation Table (FAT) dosya sistemini kullanır. Linux için /boot /efi dizinine mount edilir.

Aygıt Arayüzleri

Linux bir sisteme bağladığınız her cihaz, sistem donanımıyla iletişim kurmak için bir bağlantı protokolü kullanır. Linux çekirdek yazılımı, bu protokollerı kullanarak donanım aygıtına nasıl veri gönderileceğini ve alınacağını bilir. Aygıtları, sisteme bağlamak için temel olarak Peripheral Component Interconnect (PCI), Universal Serial Bus (USB) ve General Purpose Input/Output (GPIO) protokollerini kullanılır.

PCI Kartlar

Sunucu veya bilgisayar anakartına aygıtların bağlanması için kullanılan standarttır. Sunucu sabit diskleri (SATA, SCSI), harici disk kartları (HBA), network arayüzleri, kablosuz kartlar,

ekran kartları, ses kartları vb. bir çok donanım kartı bu standarı kullanır. Çoğu kart, tak çıkar özelliğle herhangi bir sunucu kapatma veya yeniden başlatma işlemi yapmadan kullanılabilir.

USB Arayüzü

Universal Serial Bus (USB), kullanım kolaylığı ve yüksek hızlı veri iletişim özelliğiyle bir çok alanda kullanılmaktadır. Çeşitli sürümlere sahip olsa da en güncel sürümü olan USB 3.2, 20 Gbps'ye kadar bir veri aktarım hızına sahiptir. USB arayüzüni kullanarak harici bellekleri, sabit sürücüler, yazıcıları, klavyeleri, ve mouse gibi çevre donanımları sunucu ve bilgisayarlarla bağlanabilir.

GPIO Arayüzü

Otomasyon projeleri için kullanılan led, sensor veya motor gibi harici aygıtların, Raspberry Pi gibi cihazlar üzerinde çalışan Linux İşletim sistemleri tarafından kontrol edilmesi için tasarlanan arayüzdür. Uygulamalar, GPIO arayüzü sayesinde harici aygıtları kontrol edebilir ve bu aygıtlar üzerindeki basınc, sıcaklık, yükseklik gibi değerleri okuyabilir.

Donanım Dizinleri

/dev Dizini

Linux çekirdeği, bir arayüze bağlı bir aygıt ile veri alışverişi yapmak için /dev dizini altındaki aygıt dosyalarını (device files) kullanır. Bir aygıt ile iletişime geçmek için sadece o aygıt'a ait aygıt dosyası okunmalıdır. Linux sistemine bir donanım eklendiğinde, /dev dizini altında, eklenen donanımı tanımlayan bir dosya oluşturulur. Böylece uygulamalar, bu dosya ile irtibata geçerek donanıma nasıl veri depolayacaklarını veya okuyacaklarını bilirler.

Character Device File ve Block Device File olarak iki ayrı aygıt dosyası tipi vardır. Seri bağlantı tipini kullanan terminal veya USB cihazları için Character dosya tipi oluşturulurken c harfi ile gösterilir. PCI arayüzü ile iletişime geçen kartlar veya diskler için Block dosya tipleri oluşturulur ve b harfi ile tanımlanır.

```
root@test:/dev# ls -al sda* tty1*
brw-rw---- 1 root disk 8,  0 Mar 30 10:13 sda
brw-rw---- 1 root disk 8,  1 Mar 30 10:13 sda1
brw-rw---- 1 root disk 8,  2 Mar 30 10:13 sda2
brw-rw---- 1 root disk 8,  3 Mar 30 10:13 sda3
crw--w---- 1 root tty  4,  1 Mar 30 10:13 tty1
crw--w---- 1 root tty  4, 10 Mar 30 10:13 tty10
crw--w---- 1 root tty  4, 11 Mar 30 10:13 tty11
```

Fiziksel blok cihazları, sanal block cihazlara bağlamak için device mapper fonksiyonu da Linux tarafından kullanılır. Bu sanal cihazlar, Logical Volume Manage (LVM) gibi mantıksal sürücüler oluşturmak için kullanılır. LVM Yapılandırma bölümünde bu kısım detaylı bir şekilde anlatılmıştır.

```
root@test:/dev/mapper# ls -lah
total 0
drwxr-xr-x  2 root root      80 Mar 30 10:13 .
drwxr-xr-x 20 root root     4.2K Mar 30 10:13 ..
crw-----  1 root root 10, 236 Mar 30 10:13 control
lrwxrwxrwx  1 root root      7 Mar 30 10:13 ubuntu--vg-ubuntu--lv -> ../../dm-0
```

/proc Dizini

Linux sistemindeki donanımların durumu ve bilgileri hakkında detaylar bu dizinde bulunur. Linux çekirdeği, sistemindeki donanımın durumunu izlerken /proc dizinindeki dosyaları ve verileri değiştirir. Donanım aygıtlarının ve ayarlarının durumunu görüntülemek için, standart Linux metin komutlarını (cat, grep vb.) kullanarak dosyaların içeriğine bakabilirsiniz.

/proc dizini altında sistem ve aygıtları izlemek ve bilgi almak için bir çok dosya mevcuttur.

```
root@test:/proc# ls
1      173    374    686   8      bus      kmsg      softirqs
10     175    3849   70    81    cgroups  kpagecgroup  stat
103    177    4      71    82    cmdline  kpagecount  swaps
106    178    401    72    822   consoles  kpageflags  sys
11     18     417    729   830   cpuinfo  loadavg   sysrq-trigger
119    182    4207   73    831   crypto   locks    sysvipc
12     183    4453   731   835   devices  mdstat   thread-self
1224   19     4471   74    836   diskstats meminfo  timer_list
14     2      6      742   837   dma      misc    tty
1416   20     643    746   84    driver   modules  uptime
1451   21     647    747   85    execdomains mounts  version
1452   218    648    75    86    fb      mpt    version_signature
15     22     649    753   87    filesystems mtrr    vmallocinfo
1572   23     650    755   88    fs      net    vmstat
1573   233    651    757   89    interrupts pagetypeinfo zoneinfo
1596   234    665    759   9     iomem   partitions
1599   2356   666    76    91    ioports  pressure
16     24     667    764   92    irq     sched_debug
1600   3      668    767   93    kallsyms schedstat
166    302    670    77    acpi   kcore   scsi
168    303    671    78    asound  keys    self
17     3526   672    780   buddyinfo key-users slabinfo
```

Örnek komut olarak **cat /proc/cpuinfo** komutuyla sunucu üzerinde işlemci hakkında bilgi alınabilir.

```
root@test:/proc# cat /proc/cpuinfo
processor       : 0
vendor_id      : AuthenticAMD
cpu family     : 21
model          : 2
model name     : AMD FX(tm)-6300 Six-Core Processor
stepping        : 0
microcode      : 0xffffffff
cpu MHz        : 3500.086
cache size     : 2048 KB
physical id    : 0
siblings        : 1
core id         : 0
cpu cores      : 1
apicid          : 0
initial apicid : 0
fpu             : yes
fpu_exception   : yes
cpuid level    : 13
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fx
sr sse sse2 ht syscall nx fxsr_opt rdtscl lm constant_tsc rep_good nopl nonstop_tsc cpuid extd_apicid ts
c_known_freq pn1 ssse3 sse4_1 sse4_2 hypervisor lahf_lm cr8_legacy 3dnowprefetch ssbd vmmcall arat
bugs            : fxsave_leak sysret_ss_attrs null_seg spectre_v1 spectre_v2 spec_store_bypass
bogomips       : 7000.17
TLB size        : 1536 4K pages
clflush size    : 64
cache_alignment : 64
address sizes   : 48 bits physical, 48 bits virtual
power management:
```

Sistem üzerindeki I/O portlarında çalışan aygıtları görmek için **cat /proc/iports** komutu kullanılır. Normal şartlarda bir portu bir aygit kullanır ancak herhangi bir çakışma görülürse setpci komutu kullanılarak aygitin portu değiştirilmelidir.

/sys Dizini

/proc dizinine benzer bir işlevi olan donanımlar hakkında ek bilgiler sağlayan diğer bir dizindir. Kernel tarafından oluşturulur. Cihaz ve fonksiyonlara göre alt dizinlere ayrıılır.

```
root@test:/sys# ls
block  class  devices  fs      kernel  power
bus    dev    firmware  hypervisor  module
```

Aygıtları Tanımlamak

Linux bir sistemde aygıtları görüntülemek veya problemlerini gidermek için bir çok araç mevcuttur. Bu araçlar çeşitli argümanlar alarak aygitler hakkında detaylı bilgiler alabilir.

Komut	Açıklama
lsdev	Sistem üzerindeki yüklü olan donanımları gösterir.
lsblk	Harddisk gibi bütün block donanımları gösterir.
lspci	PCI arayüzüne bağlı olan network ve HBA kartı gibi aygıtları gösterir.
lsusb	USB arayüzüne bağlı aygıtları gösterir.

Donanım Aygıtları İçin Driver Kontrolü

Günümüzde sunucular bir çok donanımı desteklemektedir. Sunucunuzun donanımlarının doğru bir şekilde desteklenip desteklenmediğini **lspci -k** komutuyla öğreniriz.

```
root@ubuntu:~# lspci -k
00:00.0 Host bridge: Intel Corporation 440BX/ZX/DX - 82443BX/ZX/DX Host bridge (AGP disabled) (rev 03)
00:07.0 ISA bridge: Intel Corporation 82371AB/EB/MB PIIX4 ISA (rev 01)
    Subsystem: Microsoft Corporation 82371AB/EB/MB PIIX4 ISA
00:07.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE (rev 01)
    Kernel driver in use: ata_piix
    Kernel modules: pata_acpi
00:07.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 02)
    Kernel modules: i2c_piix4
00:08.0 VGA compatible controller: Microsoft Corporation Hyper-V virtual VGA
    Kernel driver in use: hyperv_fb
    Kernel modules: hyperv_fb
root@ubuntu:~# cat /proc/version
Linux version 5.11.0-051100-generic (kernel@kathleen) (gcc (Ubuntu 10.2.0-13ubuntu1) 10.2.0, GN
U ld (GNU Binutils for Ubuntu) 2.35.1) #202102142330 SMP Sun Feb 14 23:33:21 UTC 2021
```

Eğer desteklenmeyen bir donanım bulursanız donanımı destekleyen bir kernel modülü yüklemelisiniz. Ancak yükleyeceğiniz kernel modülünde sahip olduğunuz kernel modülünün stabilitesinin bozmadığını kontrol etmelisiniz.

14 - NETWORK

TCP/IP Protokolünü ve IP Adreslerini Anlama

IP (Internet Protocol) protokolü internet üzerindeki adres ve iletişim sistemidir. TCP (Transmission Control Protocol) ise internet üzerindeki verileri bir sıra halinde gönderen ve hata denetimi yaparak veriler tam olarak hedefe ulaşmasını sağlayan protokoldür. Bu iki protokolün birleşimi olan TCP/IP ile günümüzdeki sadece bilgisayarlar değil telefonlar, yazıcılar, drone'lar yani internet'e temas eden tüm aygıtlar birbirleriyle iletişim kurar. İnternete erişim sağlayan daha doğrusu kendi networkü dışındaki başka bir aygit ile iletişim kurmaya çalışan her aygitin benzersiz bir IP adresi olmak zorundadır.

Aslında TCP/IP birçok protokolün bir araya gelmesi ile oluşan bir protokoller bütünüdür. TCP/IP, ağ arayüzü, internet, taşıma ve uygulama olarak 4 katmandan oluşmakla birlikte katmanları daha belirli bir şekilde bölen OSI Referans Modeli ise 7 katman'dan oluşur.

Katmanlar	Katmanların içinde hizmet veren Protokoller
Uygulama (Application) Katmanı	HHTTP/HTTPS, DNS, SMTP, SSH,FTP...
Sunum (Presentation) Katmanı	HTML, DOC, JPEG...
Oturum (Session) Katmanı	SMB, NFS..
Taşıma (Transport)Katmanı	TCP ve UDP
Ağ (Network) Katmanı	IP bilgisini içeren Packet'lerdir. ICMP, ARP vb.
Veri Bağlantısı (Data Link)Katmanı	MAC adresini içeren Frame dediğimiz yapılardır. Ethernet, PPP
Fiziksel (Physical) Katman	Fiziksel bağlantılardır. Fiber, Network Kablosu vb.

IP adresleri IPv4 ve IPv6 olarak iki çeşittir. IPv6 adresler, IPv4 adreslerin tükenmesi nedeniyle ortaya çıkmış olup günümüzde iki adres çeşidine kullanılmakta ve birbirleriyle protokoller yardımıyla sorunsuz haberleşebilmektedir.

IPv4	192.168.1.34	Binary	32 bit ve 4 Öktetden oluşur.
IPv6	fe80:badb:abe01:45bc:34ad:56a3:8862	Hexadecimal	128 bit ve 8 Öktetden oluşur.

IPv4 adreslerinin tükenmesiyle birlikte firmalar internete direk erişim sağlamayan cihazlarının üzerinde private IP dediğimiz IP adreslerini kullanmaya başladı. Eğer cihaz internete erişim sağlamak isterse networkü yönetmekten sorumlu olan router, NAT (Network Address Translation) protokolünü kullanarak private IP'yi public bir IP ile değiştirip cihazın internet üzerindeki kaynaklara erişimi sağlamaktadır.

Aşağıdaki Private IPv4 adresleri internet üzerinde hiçbir yere yönlendirilmezler. Bu IP adresleri sadece LAN (Local Area Network) üzerinde kullanılır.

Class	Network	Network Mask
A Class	10.0.0.0/8	255.0.0.0
B Class	172.16.0.0/12	255.240.0.0
C Class	192.168.0.0/16	255.255.0.0

Bir bilgisayarın hangi ağa ait olduğunu bilmek için bir alt ağ maskesi (Subnet Mask) kullanılır. Alt ağ maskesi, ağ adresinin hangi bölümünün ağı ve hangi bölümün sunucuyu gösterdiğini tanımlar. Bitlerden ve binary sayı sisteminden oluşur.

Decimal	Binary
0	00000000
1	00000001
2	00000010
4	00000100
8	00001000
16	00010000
32	00100000
64	01000000
128	10000000

Decimal	Binary	Hexadecimal	Decimal	Binary	Hexadecimal
0	0000	0	8	1000	8
1	0001	1	9	1001	9
2	0010	2	10	1010	A
3	0011	3	11	1011	B
4	0100	4	12	1100	C
5	0101	5	13	1101	D
6	0110	6	14	1110	E
7	0111	7	15	1111	F

Örneğin 192.168.1.34/24 adresini bitlere göre incelersek Subnetmask'ın 1 olan bölmelerine denk gelen IP adresi kısımları Network'ü gösterir. Burada 192.168.1 bölümü network, 0'dan başayıp sağa doğru giden bölüm ise (34) sunucunun (node, host) adresini gösterir.

IP Adresi	192.168.1.34	11000000.10101000.00000001.00100010	
Subnet Mask	255.255.255.0	11111111.11111111.11111111.00000000	/24

Diğer bir örneğimiz 172.16.144.13 adresidir. Yine subnetmask'ın 1 olan bölümleri network'ü yani 172.16.144.0 ve 13 olan bölümü de adresi gösterir.

IP Adresi	172.16.144.13	10101100.00010000.10010000.00001101	
Subnet Mask	255.255.255.240	11111111.11111111.11111111.11110000	/28

MAC Adreslerini Anlama

Her ağ kartının ayrıca MAC (Media Access Control) adresi olarak bilinen bir adresi vardır. MAC adresleri yerel ağda kullanım içindir (yani, yerel kablo veya LAN, karşılaşılan ilk yönlendiriciye kadar); farklı ağlardaki sunucular arasındaki iletişim için kullanılamazlar. Yine de önemlidirler, çünkü MAC adresleri, sunucuların IP adresine sahip olan ağ kartını bulmasına yardımcı olurlar. 48 bitlerdir. 8 bitlik 6 öktetden oluşurlar. İlk 3 öktet üretici firmayı temsil ederken diğer 3 öktet network kartına özel bir numaradır.

00:0A:B4:56:F5:8E

Temel İletişim Port ve Protokollerı

Sunucular, ağ üzerinde bir web sunucusu veya bir posta sunucusu gibi belirli hizmetler sundukları için kullanılır. Bu hizmetleri birbirlerinden ayırmak için port adresleri kullanılır. HTTP için bağlantı noktası 80 veya SSH sunucusu için bağlantı noktası 22 gibi belirli bir bağlantı noktası adresi vardır ve ağ iletişiminde, gönderen ve alıcı bağlantı noktası adreslerini kullanır. Dolayısıyla, ağ iletişimlerinde yer alan bir kaynak bağlantı noktası adresinin yanı sıra bir hedef bağlantı noktası adresi vardır. En sık kullanılan port ve protokoller;

Port Number	Kullanan Protocol
20	File Transfer Protocol (FTP) Data Transfer
21	File Transfer Protocol (FTP) Command Control
22	Secure Shell (SSH)
23	Telnet - Remote login service, unencrypted text messages
25	Simple Mail Transfer Protocol (SMTP) E-mail Routing
53	Domain Name System (DNS) service
80	Hypertext Transfer Protocol (HTTP) used in World Wide Web
110	Post Office Protocol (POP3) used by e-mail clientser
119	Network News Transfer Protocol (NNTP)
123	Network Time Protocol (NTP)
143	Internet Message Access Protocol (IMAP) Management of Digital Mail
161	Simple Network Management Protocol (SNMP)
194	Internet Relay Chat (IRC)
443	HTTP Secure (HTTPS) HTTP over TLS/SSL

IP Adresi ve Interface Yönetimi

Sunucuya iki method ile IP adresi atanabilir. Bunlardan ilki sabit bir IP adresi olabilir. Çalıştırdığınız sunucu dışarıya örneğin bir DNS, Mail vb. bir servis hizmeti veriyorsa bu sunucunun IP'sinin sabit olması gereklidir.

Diğer yöntem Dynamic Host Configuration Protocol (DHCP) protokolünü kullanarak IP adres atamalarını bir IP havuzu üzerinden otomatik olarak yapmaktadır. Bu yöntem genelde client tarafına IP atamak için kullanılır.

Network Ayarlarının Yapılandırılması ve Doğrulanması

Ubuntu 20.04'ün network ayarları diğer Ubuntu versiyonlarına göre farklı yapılandırılıyor. Network ayarları, önceki versiyonlarda /etc/network/interfaces klasörü altındaki interface'e ait dosyaları yapılandırılıyorken Ubuntu 20.04'de /etc/netplan altındaki yaml dosyası düzenlenerek yapılandırılmaktadır. Bu dosyada düzenleme yapmadan önce yedegini almaktan fayda var.

```
cp /etc/netplan/00-installer-config.yaml /etc/netplan/00-installer-config.yaml.bak
```

DHCP Yapılandırması

Ortamınızda bir DHCP server var ise ve bu sunucu üzerinde IP almak istiyorsanız yapılandırmamız aşağıdaki ekran görüntüsündeki gibi olmalıdır.

```
sudo nano /etc/netplan/00-installer-config.yaml
```

```
[1/1]          /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      dhcp4: true
  version: 2
```

Dosyada düzenleme yapıp değişiklikleri kaydettikten sonra ayarları doğruluyoruz.

```
sudo netplan try
```

```
ozgur@ubuntu:~$ sudo netplan try
[sudo] password for ozgur:
Warning: Stopping systemd-networkd.service, but it can still be activated by:
  systemd-networkd.socket
Do you want to keep these settings?

Press ENTER before the timeout to accept the new configuration

Changes will revert in 115 seconds
Configuration accepted.
```

sudo netplan apply komutu ile ayarları onayladıkten sonra ip -a komutuyla IP bilgisini kontrol ediyoruz. DHCP'den 192.168.1.40 adresi aldığımız görülüyor. Son olarak da dışarıdan bir IP adresine ping atarak erişimimizi denetliyoruz.

```
ozgur@ubuntu:~$ sudo netplan apply
ozgur@ubuntu:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:9a:f1:cc brd ff:ff:ff:ff:ff:ff
        inet 192.168.1.40/24 brd 192.168.1.255 scope global dynamic enp0s3
            valid_lft 259193sec preferred_lft 259193sec
        inet6 fe80::a00:27ff:fe9a:f1cc/64 scope link
            valid_lft forever preferred_lft forever
ozgur@ubuntu:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=113 time=89.7 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=113 time=53.3 ms
```

Static IP Yapılandırması

Yine /etc/netplan altındaki yaml dosyamızın yedeğini alıp metin editörü ile açıyoruz. IP adresi, subnet prefixini, gateway adresimizi ve DNS sunucu adreslerini girdikten sonra dosyayı kaydedip çıkıyoruz.

sudo nano /etc/netplan/00-installer-config.yaml

```
GNU nano 4.8          /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    eth0:
      addresses:
        - 192.168.1.34/24
      gateway4: 192.168.1.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
  version: 2
```

Ayarları doğrulamak için **sudo netplan try** komutunu yazıyoruz. Herşey doğruysa **sudo netplan apply** komutu ile uyguluyoruz. **ip a** komutu ile verdigimiz IP'yi görüntüleyip **ping** ile erişim kontrolü yapıyoruz.

```

root@ubuntu:~# netplan try
Warning: Stopping systemd-networkd.service, but it can still be activated by:
  systemd-networkd.socket
Do you want to keep these settings?

Press ENTER before the timeout to accept the new configuration

Changes will revert in 119 seconds
Configuration accepted.
root@ubuntu:~# netplan apply
root@ubuntu:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
  qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
      valid_lft forever preferred_lft forever
      inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
  qlen 1000
    link/ether 08:00:27:48:cc:e1 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.34/24 brd 192.168.1.255 scope global enp0s3
      valid_lft forever preferred_lft forever
      inet6 fe80::a00:27ff:fe48:cce1/64 scope link
        valid_lft forever preferred_lft forever
root@ubuntu:~# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=7 ttl=113 time=101 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=113 time=54.4 ms
64 bytes from 8.8.8.8: icmp_seq=9 ttl=113 time=54.0 ms

```

Komutlar

ifconfig komutunu Ubuntu'da kullanmak için sunucuda net-tools uygulaması kurulu olmalıdır. **apt install net-tools** komutu ile yükleyebilirsiniz. **ip** ve **ifconfig** komutları genelde network ayarlarını kontrol etmek için kullanılsa da adres, route, link ekleme, silme, interface'leri açıp kapatmak için de kullanılır.

ip a komutu sunucunun IP yapılandırmasını gösterir. **ip addr** komutu da aynı çıktıyi verir.

```

root@ubuntu:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
  link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
  inet 127.0.0.1/8 scope host lo
    valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
      valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
  link/ether 08:00:27:48:cc:e1 brd ff:ff:ff:ff:ff:ff
  inet 192.168.1.34/24 brd 192.168.1.255 scope global enp0s3
    valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe48:cce1/64 scope link
      valid_lft forever preferred_lft forever

```

ip l show komutuyla interfacelerin durumları görüntülenir.

```

root@client:~# ip l show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default
  qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default
  qlen 1000
    link/ether 00:15:5d:02:b5:12 brd ff:ff:ff:ff:ff:ff

```

ip r komutuyla ip bilgisi, default gateway ve ait olunan network görüntülenir.

```

root@client:~# ip r
default via 192.168.1.1 dev eth0 proto static
192.168.1.0/24 dev eth0 proto kernel scope link src 192.168.1.39

```

ifconfig komutu IP yapılandırmasını göstermekle birlikte interface gelen ve giden paketler hakkında da bilgi verir. Networksel bir problemde problemi tespit etmek için **ifconfig** komutunu kullanmak daha uygundur.

```
root@ubuntu:~# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.1.34 netmask 255.255.255.0 broadcast 192.168.1.255
              inet6 fe80::a00:27ff:fe48:cce1 prefixlen 64 scopeid 0x20<link>
                ether 08:00:27:48:cc:e1 txqueuelen 1000 (Ethernet)
                  RX packets 1697 bytes 170513 (170.5 KB)
                  RX errors 0 dropped 6 overruns 0 frame 0
                  TX packets 836 bytes 101932 (101.9 KB)
                  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
              inet6 ::1 prefixlen 128 scopeid 0x10<host>
                loop txqueuelen 1000 (Local Loopback)
                  RX packets 36 bytes 3748 (3.7 KB)
                  RX errors 0 dropped 0 overruns 0 frame 0
                  TX packets 36 bytes 3748 (3.7 KB)
                  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Belirli bir interface için bilgi almak istersek;

ifconfig enp0s3

```
root@ubuntu:~# ifconfig enp0s3
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.1.34 netmask 255.255.255.0 broadcast 192.168.1.255
              inet6 fe80::a00:27ff:fe48:cce1 prefixlen 64 scopeid 0x20<link>
                ether 08:00:27:48:cc:e1 txqueuelen 1000 (Ethernet)
                  RX packets 2263 bytes 217550 (217.5 KB)
                  RX errors 0 dropped 7 overruns 0 frame 0
                  TX packets 1187 bytes 140322 (140.3 KB)
                  TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

ip a show dev enp0s3

```
root@ubuntu:~# ip a show dev enp0s3
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 100
  link/ether 08:00:27:48:cc:e1 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.34/24 brd 192.168.1.255 scope global enp0s3
      valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe48:cce1/64 scope link
      valid_lft forever preferred_lft forever
```

ip a add 192.168.1.44 dev enp0s3 komutu ile interface IP adresi ekleyebiliriz. Add yerine **del** argümanını kullanırsak silebiliriz.

```
ozgur@ubuntu:~$ sudo ip a add 192.168.1.44 dev enp0s3
ozgur@ubuntu:~$
ozgur@ubuntu:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
  link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
      valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
      valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 100
  link/ether 08:00:27:9a:f1:cc brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.39/24 brd 192.168.1.255 scope global dynamic enp0s3
      valid_lft 259131sec preferred_lft 259131sec
    inet 192.168.1.44/32 scope global enp0s3
      valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe9a:f1cc/64 scope link
      valid_lft forever preferred_lft forever
```

ifconfig enp0s3 add 192.168.1.50 komutuyla interface bir IP adresi ekledik. Ifconfig komut çıkışına baktığımızda enp0s3:0 isimli bir sanal interface olduğunu görebiliriz. **del** argümanı ile eklediğimiz IP'yi silebiliriz.

```
root@ubuntu:~# ifconfig enp0s3 add 192.168.1.50
root@ubuntu:~# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.1.39 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::a00:27ff:fe9a:f1cc prefixlen 64 scopeid 0x20<link>
          ether 08:00:27:9a:f1:cc txqueuelen 1000 (Ethernet)
            RX packets 2827 bytes 1206765 (1.2 MB)
            RX errors 0 dropped 9 overruns 0 frame 0
            TX packets 1117 bytes 109841 (109.8 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s3:0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.1.50 netmask 255.255.255.0 broadcast 192.168.1.255
          ether 08:00:27:9a:f1:cc txqueuelen 1000 (Ethernet)

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 108 bytes 8693 (8.6 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 108 bytes 8693 (8.6 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

ifconfig enp0s3 down komutu ile ismini yazdığınız interface'i kapatabilirsiniz. **ifconfig enp0s3** çıkışına dikkat ederseniz IP bilgisinin olmadığı görülmüyor.

ifconfig enp0s3 up komutu ile ismini yazdığınız interface'i açabilirsiniz. **ifconfig enp0s3** çıkışına dikkat ederseniz IP bilgisinin olduğu görülmüyor.

```
root@ubuntu:~# ifconfig enp0s3 down
root@ubuntu:~# ifconfig enp0s3
enp0s3: flags=4098<BROADCAST,MULTICAST> mtu 1500
        ether 08:00:27:9a:f1:cc txqueuelen 1000 (Ethernet)
          RX packets 2689 bytes 1197742 (1.1 MB)
          RX errors 0 dropped 9 overruns 0 frame 0
          TX packets 1092 bytes 107807 (107.8 KB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@ubuntu:~# ifconfig enp0s3 up
root@ubuntu:~# ifconfig enp0s3
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.1.39 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::a00:27ff:fe9a:f1cc prefixlen 64 scopeid 0x20<link>
          ether 08:00:27:9a:f1:cc txqueuelen 1000 (Ethernet)
            RX packets 2698 bytes 1198842 (1.1 MB)
            RX errors 0 dropped 9 overruns 0 frame 0
            TX packets 1101 bytes 108811 (108.8 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

ip link set enp0s3 down komutu ile ismini yazdığınız interface'i kapatabilirsiniz. **ip a** çıkışına dikkat ederseniz IP bilgisinin olmadığı görülmüyor.

ip link set enp0s3 up komutu ile ismini yazdığınız interface'i açabilirsiniz. **ip a** çıkışına dikkat ederseniz IP bilgisinin olduğu görülmüyor.

```
root@ubuntu:~# ip link set enp0s3 down
root@ubuntu:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST> mtu 1500 qdisc fq_codel state DOWN group default qlen 1000
    link/ether 08:00:27:48:cc:e1 brd ff:ff:ff:ff:ff:ff
root@ubuntu:~# ip link set enp0s3 up
root@ubuntu:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:48:cc:e1 brd ff:ff:ff:ff:ff:ff
        inet 192.168.1.34/24 brd 192.168.1.255 scope global enp0s3
            valid_lft forever preferred_lft forever
        inet6 fe80::a00:27ff:fe48:cc:e1/64 scope link tentative
            valid_lft forever preferred_lft forever
```

ip link dev enp0s3 address 08:00:27:3f:a7:33 komutu ile interface'in MAC adresini değiştirebiliriz.

```
root@ubuntu:~# ip link set dev enp0s3 address 08:00:27:3f:a7:33
root@ubuntu:~# ifconfig enp0s3
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::a00:27ff:fe9a:f1cc prefixlen 64 scopeid 0x20<link>
        ether 08:00:27:3f:a7:33 txqueuelen 1000 (Ethernet)
        RX packets 2921 bytes 1212780 (1.2 MB)
        RX errors 0 dropped 12 overruns 0 frame 0
        TX packets 1136 bytes 112628 (112.6 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

ifconfig enp0s3 hw ether 08:00:27:e3:76:a2 komutu ile interface'in MAC adresini değiştirebiliriz.

```
root@ubuntu:~# ifconfig enp0s3 hw ether 08:00:27:e3:76:a2
root@ubuntu:~# ifconfig enp0s3
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet6 fe80::a00:27ff:fe9a:f1cc prefixlen 64 scopeid 0x20<link>
        ether 08:00:27:e3:76:a2 txqueuelen 1000 (Ethernet)
        RX packets 3055 bytes 1222097 (1.2 MB)
        RX errors 0 dropped 13 overruns 0 frame 0
        TX packets 1156 bytes 119120 (119.1 KB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

ping komutu ağdaki diğer cihazlara erişimin olup olmadığını kontrolünü sağlayan bir araçtır.

ping 192.168.1.38 : Hedef IP'ye ping atarak erişimin olup olmadığını kontrol eder.

ping -c 10 192.168.1.38 : Hedef IP'ye 10 ICMP paketi gönderdikten sonra durmayı sağlar.

```
root@ubuntu:~# ping -c 10 192.168.1.38
PING 192.168.1.38 (192.168.1.38) 56(84) bytes of data.
64 bytes from 192.168.1.38: icmp_seq=1 ttl=128 time=1.15 ms
64 bytes from 192.168.1.38: icmp_seq=2 ttl=128 time=0.573 ms
64 bytes from 192.168.1.38: icmp_seq=3 ttl=128 time=3.16 ms
64 bytes from 192.168.1.38: icmp_seq=4 ttl=128 time=0.978 ms
64 bytes from 192.168.1.38: icmp_seq=5 ttl=128 time=0.864 ms
64 bytes from 192.168.1.38: icmp_seq=6 ttl=128 time=0.726 ms
64 bytes from 192.168.1.38: icmp_seq=7 ttl=128 time=0.724 ms
64 bytes from 192.168.1.38: icmp_seq=8 ttl=128 time=0.705 ms
64 bytes from 192.168.1.38: icmp_seq=9 ttl=128 time=0.821 ms
64 bytes from 192.168.1.38: icmp_seq=10 ttl=128 time=0.686 ms

--- 192.168.1.38 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9046ms
rtt min/avg/max/mdev = 0.573/1.039/3.164/0.725 ms
```

ping -D 192.168.1.38 ping çıktısına zaman damgası ekler.

```
root@ubuntu:~# ping -D 192.168.1.38
PING 192.168.1.38 (192.168.1.38) 56(84) bytes of data.
[1613160517.687940] 64 bytes from 192.168.1.38: icmp_seq=1 ttl=128 time=1.48 ms
[1613160518.690446] 64 bytes from 192.168.1.38: icmp_seq=2 ttl=128 time=1.14 ms
[1613160519.691940] 64 bytes from 192.168.1.38: icmp_seq=3 ttl=128 time=0.770 ms
^C
--- 192.168.1.38 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2005ms
rtt min/avg/max/mdev = 0.770/1.131/1.484/0.291 ms
```

ping -I enp0s3 192.168.1.38 : Belirttiğiniz interface'den ICMP paketleri gönderilir.

```
root@ubuntu:~# ping -I enp0s3 192.168.1.38
PING 192.168.1.38 (192.168.1.38) from 192.168.1.34 enp0s3: 56(84) bytes of data.
64 bytes from 192.168.1.38: icmp_seq=1 ttl=128 time=0.776 ms
64 bytes from 192.168.1.38: icmp_seq=2 ttl=128 time=0.733 ms
64 bytes from 192.168.1.38: icmp_seq=3 ttl=128 time=0.705 ms
^C
--- 192.168.1.38 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 0.705/0.738/0.776/0.029 ms
```

traceroute, belirttiğiniz adrese giderken gönderdiğiniz paketlerin uğradığı noktaları bir haritasını çıkarır. Bu komut genelde erişim sağlamaya çalışığınız hedef adresine ulaşamayınca veya erişimde ping süresinin uzun olduğu durumlarda problemin nereden kaynaklandığını tespit etmek kullanılan yararlı bir araçtır. Traceroute, hedef IP adresine varsayılan olarak 3 ICMP paketi gönderir. Paketlerin yol üzerinde geçtiği noktalara erişim süreleri ve sonunda da hedefe olan erişim için kaç nokta geçtiği ve ne kadar sürdüğü ortaya çıkar. Bilinmedik hedeflere alınan trace çıktıları hedefe ulaşmayı bilir. Bunun nedeni hedefin ICMP paketlerine izin vermemesinden kaynaklanır. Traceroute, komutunu kullanmak için sunucuza bu aracı eklemelisiniz. apt install traceroute komutu ile birkaç saniye içinde yüklenir.

traceroute hedef ip veya FQDN (Full Quality Domain Name)

```
root@ubuntu:~# traceroute howtogeek.com
traceroute to howtogeek.com (151.101.194.217), 30 hops max, 60 byte packets
 1 _gateway (192.168.1.1)  2.386 ms  2.205 ms  2.090 ms
 2 212.156.201.88.static.turktelekom.com.tr (212.156.201.88)  23.137 ms  24.504 ms  26.312 ms
 3 81.212.2.123.static.turktelekom.com.tr (81.212.2.123)  29.948 ms  29.704 ms  30.071 ms
 4 34-acibadem-xrs-t2-2---34-acibadem-t3-7.static.turktelekom.com.tr (81.212.215.36)  32.393 ms 34-acibadem-xrs-t2-1---3
 5 * *
 6 301-fra-col-1---34-ebgp-acibadem-sr12e-k.static.turktelekom.com.tr (212.156.101.230)  75.137 ms  58.286 ms  60.481 ms
 7 213.198.83.197 (213.198.83.197)  79.800 ms ffbm-bl-link.telia.net (213.248.79.214)  70.434 ms 213.198.83.197 (213.198.
 83.197)  77.385 ms
 8 217.5.118.22 (217.5.118.22)  402.272 ms  397.858 ms  397.438 ms
```

traceroute –n google.com komutuyla trace çıktısındaki alan adlarının yerine IP döndürülür.

traceroute -q 1 google.com komutu ile gönderilecek paket sayısı ayarlanır.

mtr (my traceroute) hedefe gönderilen paketlerin geçtiği noktalar hakkında canlı olarak bilgi verir. Temel olarak traceroute ve ping araçlarının birleşimidir.

mtr 8.8.8.8

My traceroute [v0.93]							
ubuntu (192.168.1.42) 2021-02-13T08:31:24+0000							
Keys: Help Display mode Restart statistics Order of fields quit							
				Packets Pings			
Host	Loss%	Snt	Last	Avg	Best	Wrst	StDev
1. _gateway	0.0%	130	0.7	1.1	0.6	5.2	0.6
2. 212.156.201.88.static.turkteleko	0.0%	130	23.6	23.4	22.1	40.1	1.7
3. 81.212.2.123.static.turktelekom.	0.8%	129	26.4	25.3	22.6	255.1	20.5
4. 34-acibadem-xrs-t2-2---34-acibad	0.8%	129	23.2	23.8	22.5	26.6	0.7
5. 34-ebgp-acibadem-sr12e-k---34-ac	76.6%	129	23.6	23.7	22.8	25.7	0.6
6. 307-sof-col-2---34-ebgp-acibadem	0.0%	129	33.0	33.6	32.3	68.8	3.2
7. 74.125.52.6	0.0%	129	41.9	43.4	40.4	205.9	14.8
8. 108.170.250.161	0.0%	129	53.3	53.2	52.2	58.3	0.7
9. 108.170.238.171	0.0%	129	52.6	53.4	52.3	56.0	0.6
10. dns.google	0.0%	129	53.4	53.0	52.0	55.9	0.6

Çıktıda packets bölümünden geçilen noktalardaki gönderilen paket sayısı ve kayıp oranları gösterilir. Pings altında sırayla ms cinsinden en gönderilen paketin iletim süresi, ortalama süre, en iyi ve en kötü süre ve en sonda da standart sapma bilgileri gösterilmektedir. Yukarıdaki çıktıya göre 5. Satırda bulunan noktada paket kayıpları olduğu 3 ve 4'üncü satırlarda problem olabileceği görülmektedir.

nslookup komutu DNS çözümleme problemlerinin tespitinde ve DNS kayıtlarının sorgulanmasında kullanılan bir araçtır.

nslookup abc.com komutu abc.com için bir A kaydı döndürür. Eğer alan adından eminsek ve bize bir kayıt dönmüyorsa ilk olarak sorgu yaptığımız alan adından şüphelenerek emin olunan başka adlarda sorgulanır. Sorgulardan beklenen yanıtlar alınamıyorsa sorgu yaptığımız DNS sunucuda problem olabilir. DNS sunucu değiştirerek sorgular tekrar denenir.

DNS çözümleme probleminizin olması sunucunuzun alan adları üzerinden aldığı hizmetlerde aksaklı yaşamamasına neden olur. Sunucunuzda güncelleme, dosya indirme gibi işlemleri yerine getiremezsiniz.

```
root@ubuntu:~# nslookup abc.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   abc.com
Address: 13.224.71.15
Name:   abc.com
Address: 13.224.71.121
Name:   abc.com
Address: 13.224.71.48
Name:   abc.com
Address: 13.224.71.32
```

Bir alan adı için belirli bir kaydı görmek isterseniz -type argümanını kullanabilirsiniz.

```
ozgur@ubuntu:~$ nslookup -type=ns abc.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
abc.com nameserver = ns-736.awsdns-28.net.
abc.com nameserver = ns-318.awsdns-39.com.
abc.com nameserver = ns-1869.awsdns-41.co.uk.
abc.com nameserver = ns-1368.awsdns-43.org.
```

```
ozgur@ubuntu:~$ nslookup -type=mx abc.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
abc.com mail exchanger = 10 abc-com.mail.protection.outlook.com.

Authoritative answers can be found from:
```

Alan adına ait tüm kayıtları görmek –query=any argümanı kullanılır.

nslookup –query=any abc.com

```
ozgur@ubuntu:~$ nslookup -query=any abc.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   abc.com
Address: 143.204.2.22
Name:   abc.com
Address: 143.204.2.21
Name:   abc.com
Address: 143.204.2.112
Name:   abc.com
Address: 143.204.2.98
abc.com nameserver = ns-1368.awsdns-43.org.
abc.com nameserver = ns-1869.awsdns-41.co.uk.
abc.com nameserver = ns-318.awsdns-39.com.
abc.com nameserver = ns-736.awsdns-28.net.
abc.com
    origin = ns-318.awsdns-39.com
    mail addr = awsdns-hostmaster.amazon.com
    serial = 1
    refresh = 7200
    retry = 900
    expire = 1209600
    minimum = 86400
abc.com mail exchanger = 10 abc-com.mail.protection.outlook.com.
abc.com text = "ECZjYXsxe4CrnyGjS8EinRw2keq1hv77Z66acQb6JhwQk14sk4ZGwLt61w4a2htOdmqIJUj1fNCxo6721F0pfq=="
abc.com text = "MS=ms24761496"
abc.com text = "adobe-idp-site-verification=012b7d24aff9766444b9232173abb52ef026139e50aac77c49e02bd5d0dc3916"
abc.com text = "docusign=12a35007-299f-4d83-bd45-4f1963b4e234"
abc.com text = "docusign=53e074c1-b80d-41a1-be73-d444698c3a91"
abc.com text = "v=spf1 include:spf.disney.com -all"
Authoritative answers can be found from:
```

dig (domain information groper) komutu da DNS kayıtlarının sorgulanmasında kullanılan bir araçtır.

dig google.com komutuyla domain adının A kaydıyla sorgulamasını yapıyoruz. Question bölümü bizim yaptığımız sorğu, answer bölümü ise sorgumuza verilen cevaptır.

dig MX google.com komutuyla alan adının MX kaydını sorgulamış oluruz.

```
ozgur@ubuntu:~$ dig google.com

; <>> DiG 9.16.1-Ubuntu <>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 3142
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;google.com.           IN      A

;; ANSWER SECTION:
google.com.        90      IN      A      216.58.207.174

;; Query time: 59 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Sat Feb 13 00:01:31 +03 2021
;; MSG SIZE  rcvd: 55
```

Belirli bir DNS sunucusundan sorgu çalıştırılmak istersek @dns_server_adresi parametresini kullanmalıyız.

dig ubuntu.org @8.8.8.8

```
ozgur@ubuntu:~$ dig ubuntu.org @8.8.8.8

; <>> DiG 9.16.1-Ubuntu <>> ubuntu.org @8.8.8.8
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 39649
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;ubuntu.org.           IN      A

;; ANSWER SECTION:
ubuntu.org.        14399   IN      A      69.16.230.42

;; Query time: 211 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Sat Feb 13 00:15:20 +03 2021
;; MSG SIZE  rcvd: 55
```

dig komutu ile belirli kayıtları sorgulayabilirsiniz. +nocmd, +noall ve +answer argümanları dig çıktısındaki bölgeleri gizleyerek bizim sorguladığımız kayıtla ilgili bilgi dönmesci sağlamaktadır.

dig +nocmd ubuntu.org mx +noall +answer

```
ozgur@ubuntu:~$ dig +nocmd ubuntu.org mx +noall +answer
ubuntu.org.          7084    IN      MX      10 mx156.hostedmxserver.com.
ozgur@ubuntu:~$ dig +nocmd ubuntu.org ns +noall +answer
ubuntu.org.          21589   IN      NS      ns2.parklogic.com.
ubuntu.org.          21589   IN      NS      ns1.parklogic.com.
```

systemd-resolve --status komutu ile DNS sunucu ayarlarını kontrol edebiliriz.

```
Link 2 (enp0s3)
    Current Scopes: DNS
DefaultRoute setting: yes
    LLMNR setting: yes
MulticastDNS setting: no
    DNSOverTLS setting: no
    DNSSEC setting: no
    DNSSEC supported: no
Current DNS Server: 8.8.8.8
    DNS Servers: 8.8.8.8
```

networkctl komutu sunucu üzerindeki interface'lerin listesini vermekle birlikte networkd servisi tarafından yönetilen interfacelerin durumlarını, kongürasyon yenilemelerini, interface'i silme gibi işlemler yapmamıza izin verir. unmanaged olarak ayarlanmış interfaceler üzerinde bu aracı kullanarak işlem yapılamaz.

```
root@ubuntu:~# networkctl
IDX LINK      TYPE      OPERATIONAL SETUP
 1  lo        loopback  carrier    unmanaged
 2  enp0s3    ether     routable  configured
```

netstat (Network Statistics), sunucuya gelen ve giden network bağlantılarını izleyen çok önemli ve faydalı bir araçtır. Netstat sunucudaki hangi portun hangi uygulama ile ilişkilendirildiğini, uygulamaların network erişimlerinin olup olmadığını gösterir.

netstat komutu tek başına çok uzun bir liste çıktısı verir ve bu liste içerisinde aradığınızı bulmak çok zordur. Bu nedenle | işaretini kullanarak netstatla birlikte diğer komutları kullanarak anlamlı çıktılar elde etmek gereklidir.

Linux sistemlerde netstat aracı yüklü değildir. Bu aracı **apt install net-tools** komutu ile yükleyebilirsiniz.

netstat | less komutu ile çıktı içerisinde yukarı aşağı gezebilir veya arama yapabilirsiniz.

```
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address          Foreign Address        State
tcp      0      0 ubuntu:ssh              192.168.1.38:56722  ESTABLISHED
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type      State         I-Node      Path
unix    2      [ ]      DGRAM          26413      /run/user/1000/systemd/notify
unix    3      [ ]      DGRAM          16934      /run/systemd/notify
unix    2      [ ]      DGRAM          16951      /run/systemd/journal/syslog
```

netstat -a komutu bütün portların olduğu bir çıktı verir.

netstat -at sadece TCP protokolünü kullanan portları gösterir.

netstat -au sadece UDP protokolünü kullanan portları gösterir.

netstat -l sadece listening durumunda olan portların çıktısını verir.

netstat -lt sadece TCP kullanan ve listening durumunda olan portların çıktısını verir.

netstat -lu sadece UDP kullanan ve listening durumunda olan portların çıktısını verir.

netstat -s TCP, UDP, ICMP ve IP protokollerinin istatistik çıktısını verir. Yine -t ve -u argümanları ile TCP ve UDP için çıktıları alınabilir.

```
root@ubuntu:~# netstat -s
Ip:
Forwarding: 2
96023 total packets received
0 forwarded
0 incoming packets discarded
94591 incoming packets delivered
63936 requests sent out
20 outgoing packets dropped
Icmp:
8065 ICMP messages received
0 input ICMP message failed
ICMP input histogram:
destination unreachable: 42
timeout in transit: 7138
echo replies: 885
8703 ICMP messages sent
0 ICMP messages failed
ICMP output histogram:
destination unreachable: 40
echo requests: 8663
IcmpMsg:
InType0: 885
InType3: 42
InType11: 7138
OutType3: 40
OutType8: 8663
Tcp:
11 active connection openings
1 passive connection openings
0 failed connection attempts
0 connection resets received
1 connections established
86088 segments received
54951 segments sent out
7 segments retransmitted
0 bad segments received
3 resets sent
Udp:
248 packets received
40 packets to unknown port received
0 packet receive errors
288 packets sent
0 receive buffer errors
0 send buffer errors
IgnoredMulti: 99
```

netstat -tp | less servisleri PID numarası ile gösterir.

```
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State      PID/Program name
tcp     0      0 ubuntu:ssh                192.168.1.38:56722 ESTABLISHED 1734/sshd: ozgur [p]
```

netstat -ac 10 | less komutunda -c argümanı ile 10 saniyede bir çıktıyı otomatik olarak yenileyebilirsiniz.

netstat -r komutu ile routing tablosu görüntülenir.

```
root@ubuntu:~# netstat -r
Kernel IP routing table
Destination     Gateway         Genmask        Flags   MSS Window irtt Iface
default         _gateway       0.0.0.0       UG        0 0          0 eth0
192.168.1.0    0.0.0.0       255.255.255.0 U          0 0          0 eth0
```

netstat -i komutu ile interfacelere gelen giden paket bilgileri görüntülenir.

Kernel Interface table										
Iface	MTU	RX-OK	RX-ERR	RX-DRP	RX-OVR	TX-OK	TX-ERR	TX-DRP	TX-OVR	Flg
eth0	1500	97331	0	2	0	63969	0	0	0	BMRU
lo	65536	228	0	0	0	228	0	0	0	LRU

netstat -ant | grep 22 gibi bir komutla 22nci portlarındaki çıktıyı görebilir veya belirli bir IP için olan bağlantıları görmek için grep den sonra IP yazabilirisiniz.

root@ubuntu:~# netstat -ant grep 22				
tcp	0	0 0.0.0.0:22	0.0.0.0:*	LISTEN
tcp	0	64 192.168.1.42:22	192.168.1.38:56722	ESTABLISHED
tcp6	0	0 :::22	:::*	LISTEN

netstat -ant |grep 22 | grep 'ESTABLISHED' gibi bir komutla grep komutlarını birleştirebilirisiniz.

root@ubuntu:~# netstat -ant grep 22 grep 'ESTABLISHED'				
tcp	0	64 192.168.1.42:22	192.168.1.38:56722	ESTABLISHED

ss komutu netstat yerine kullanılabilen diğer bir komuttur. Kullanım şekli netstat ile aynıdır.

root@ubuntu:~# ss -lt					
State	Recv-Q	Send-Q	Local Address:Port	Peer Address:Port	Process
LISTEN	0	4096	127.0.0.53*:domain	0.0.0.0:*	
LISTEN	0	128	0.0.0.0:ssh	0.0.0.0:*	
LISTEN	0	128	[::]:ssh	[::]:*	

Hostname, hosts ve resolve.conf Dosyasının Düzenlenmesi

Bir sunucu başka bir sunucu ile hostname’ı üzerinden konuşur bu nedenl hostname’ın yapılandırılması önemlidir. DNS üzerinden yapılacak işlemlerde sunucunuzun bir FQDN’ının olması şarttır. Hostname değiştirmek iki yöntem vardır.

/etc/hostname dosyasını metin editörüyle girip istediğimiz ismi yazarak kaydediyoruz.

```
GNU nano 4.8                               /etc/hostname
ubuntu.deneme.com
```

hostnamectl set-hostname ubuntu.deneme.com komutuyla da değiştirebiliriz.

```
root@ubuntu:~# hostnamectl set-hostname ubuntu.deneme.com
root@ubuntu:~# hostnamectl
      Static hostname: ubuntu.deneme.com
                  Icon name: computer-vm
                    Chassis: vm
           Machine ID: ac1a3ea39d564ceba3152b9085a964e6
              Boot ID: 4797416d31684ef890ee16f146d4b107
        Virtualization: oracle
   Operating System: Ubuntu 20.04.2 LTS
                 Kernel: Linux 5.4.0-65-generic
       Architecture: x86-64
```

Hosts dosyası lokal olarak IP adreslerine host isimlerini adresler. Sunucuların FQDN’leri kullanarak birbirleriyle konuşması için /etc/hosts dosyasının içine sunucuların FQDN ve IP adresleri eklenir.

Örnekte test.deneme.com adresine ping atmaya çalışıyoruz ancak böyle bir adres olmadığı için hata aldık. Sonrasında **/etc/hosts** dosyasına girerek test.deneme.com için bir kayıt yapıp dosyayı kaydediyoruz. Tekrar ping attığında kayıt girdiğimiz adrese ping atabiliyoruz.

```
root@ubuntu:~# ping test.deneme.com
ping: test.deneme.com: Name or service not known
root@ubuntu:~# nano /etc/hosts
root@ubuntu:~# ping test.deneme.com
PING test.deneme.com (192.168.1.42) 56(84) bytes of data.
64 bytes from test.deneme.com (192.168.1.42): icmp_seq=1 ttl=64 time=1.59 ms
64 bytes from test.deneme.com (192.168.1.42): icmp_seq=2 ttl=64 time=2.26 ms
```

GNU nano 4.8	/etc/hosts
127.0.0.1 localhost	
127.0.1.1 ubuntu	
192.168.1.34 ubuntu.deneme.com	
192.168.1.42 test.deneme.com	

resolve.conf dosyası sunucunuzu DNS çözümlemesi için kullanacağı DNS sunucularının kaydının tutulduğu dosyadır. Dosyanın içine girdiğinizde systemd-resolved tarafından yönetildiği ve düzenlemememiz konusunda uyarılıyoruz. Bazı sistem yöneticileri bu sayfada farkına varmadan düzenleme yapıyor ancak düzgün bir network yapılandırması ile bu dosyada zaten bir düzenleme yapma ihtiyacınız olmuyor.

sudo nano /etc/resolv.conf

```
GNU nano 4.8          /etc/resolv.conf
# This file is managed by man:systemd-resolved(8). Do not edit.
#
```

Network yapılandırma dosyamız içerisinde DNS (nameservers) adreslerimizi yazdık.

GNU nano 4.8 /etc/netplan/00-installer-config.yaml	
# This is the network config written by 'subiquity'	
network:	
ethernets:	
eth0:	
addresses:	
- 192.168.1.39/24	
gateway4: 192.168.1.1	
nameservers:	
addresses: [8.8.8.8, 8.8.4.4]	
version: 2	

resolvectl status komutuyla DNS sorgulamamızın durumunu kontrol edelim. Ekran çıktısında alt satırlarda interface'in olduğu bölümde ayarladığımız DNS sunucuları görüyoruz.

Link 2 (eth0)	
Current Scopes: DNS	
DefaultRoute setting: yes	
LLMNR setting: yes	
MulticastDNS setting: no	
DNSOverTLS setting: no	
DNSSEC setting: no	
DNSSEC supported: no	
DNS Servers: 8.8.8.8	
8.8.4.4	

Routing Table

Sunucunun paketlerini gönderdiği gateway ve interface'leri gösterir. Routing tablosunda varsayılan olarak default gateway için kayıt girilmiştir. Bu tablo kimin nereye nereden bağlanacağını söyleyen ve yönlendiren bir tablodur.

route -n

Aşağıdaki çıktıda destination kısmında 0.0.0.0 yazmaktadır. Bunun anlamı hedefi herhangi bir network olan bütün istekleri Gateway 192.168.1.1 adresine gönder. İkinci satırda ise 192.168.1.0 networkünü heryere yönlendir. Bu iki satırla sunucunun dışarıya erişimi sınırlımadan bağlı olduğu network'ün gateway'i aracılığıyla sağlanır. Eğer bir ev bilgisayarı kullanıcısıysanız gateway burada sizin modeminiz oluyor.

root@ubuntu:~# route -n							
Kernel IP routing table							
Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	192.168.1.1	0.0.0.0	UG	0	0	0	eth0
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0

Default rota girmek için : **route add default gw 192.168.1.1**

Örnekle açıklayalım. Sunucuya ikinci bir interface ekledik ve netplan dosyamızı düzenleyerek bu interface'e sabit bir IP adresi daha verdik.

```
GNU nano 4.8                               /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    eth0:
      addresses:
        - 192.168.1.42/24
      gateway4: 192.168.1.1
      nameservers:
        addresses:
          - 8.8.8.8
          - 8.8.4.4
    eth1:
      addresses:
        - 192.168.1.70/24
      gateway4: 192.168.1.1
      nameservers:
        addresses:
          - 8.8.8.8
          - 8.8.4.4
  version: 2
```

```

root@ubuntu:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.1.42 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::215:5dff:fe02:b510 prefixlen 64 scopeid 0x20<link>
          ether 00:15:5d:02:b5:10 txqueuelen 1000 (Ethernet)
            RX packets 1422 bytes 161130 (161.1 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 931 bytes 132857 (132.8 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.1.70 netmask 255.255.255.0 broadcast 192.168.1.255
        inet6 fe80::215:5dff:fe02:b511 prefixlen 64 scopeid 0x20<link>
          ether 00:15:5d:02:b5:11 txqueuelen 1000 (Ethernet)
            RX packets 113 bytes 25764 (25.7 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 19 bytes 1506 (1.5 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1000 (Local Loopback)
            RX packets 86 bytes 6548 (6.5 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 86 bytes 6548 (6.5 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

Routing tablomuza baktığımızda iki interface'ımızın olduğu ve her ikisi içinde 2'ser kayıt görüldüğü görülüyor. Belirli bir IP için eth1 interface'ni kullanacak bir route ekliyoruz.

route add 8.8.8.8 gw 192.168.1.1 dev eth1

Routing tablomuza eklendiğini görüyoruz.

```

root@ubuntu:~# route add 8.8.8.8 gw 192.168.1.1 dev eth1
root@ubuntu:~# route -n
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref    Use Iface
0.0.0.0         192.168.1.1   0.0.0.0        UG      0      0        0 eth0
0.0.0.0         192.168.1.1   0.0.0.0        UG      0      0        0 eth1
8.8.8.8         192.168.1.1   255.255.255.255 UGH     0      0        0 eth1
192.168.1.0     0.0.0.0       255.255.255.0   U       0      0        0 eth0
192.168.1.0     0.0.0.0       255.255.255.0   U       0      0        0 eth1

```

curl -I 8.8.8.8 komutuyla bir http isteği gönderiyoruz ve netstat ile kontrol ediyoruz. İsteğin gönderildiği IP'nin 192.168.1.70 IP'li eth1 interface'i olduğunu görüyoruz. Bu şekilde sunucu trafiğini yönlendirme şansımız var. Yaptığımız örnek çok gerçek hayatı uygun olmadı ancak bir sunucunun birden fazla bağlı olduğu gateway olması durumunda network trafiğini yönetebilirsiniz. Örneğin yedekleri gönderdiğiniz bir FTP sunucunuz var. Bu sunucu sizin internal networkünüzde ise internal network'e bakan interface'e bu trafiği yönlendirebilirsiniz.

Active Internet connections (servers and established)					
Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
tcp	0	0	127.0.0.53:53	0.0.0.0:*	LISTEN
tcp	0	0	0.0.0.0:22	0.0.0.0:*	LISTEN
tcp	0	1	192.168.1.70:44064	8.8.8.8:80	SYN_SENT

route del 8.8.8.8 gw 192.168.1.1 dev eth1 route'u silmek için del argümanı kullanılır.

Bir host'u (-host) veya network'ün (-net) erişimini routing tablosu üzerinden engellemek istersek reject argümanını kullanıyoruz.

route add -host 192.168.1.34 reject

route add -net 192.168.1.0 netmask 255.255.255.0 reject (Bir network'ü engellemek)

root@ubuntu:~# route -n Kernel IP routing table							
Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	192.168.1.1	0.0.0.0	UG	0	0	0	eth0
0.0.0.0	192.168.1.1	0.0.0.0	UG	0	0	0	eth1
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
192.168.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
192.168.1.34	-	255.255.255.255	!H	0	-	0	-

15 - Disk Yönetimi

Bir sabit sürücüyü kullanmak için, sabit sürücüde bölümlerin olması gereklidir. Bazı işletim sistemleri her şeyi bir bölüme kurarken (Windows), Linux işletim sistemleri normalde bir sabit diskte boot, root, swap gibi birkaç bölüme sahiptir.

Bilgisayarlar icat edildiğinde, sabit disk düzenini tanımlamak için bir sisteme ihtiyaç vardır. Bu sistem ilk olarak Master Boot Record (MBR) olarak tanımlandı. Daha önceki bölümlerde de bahsettiğimiz üzere MBR, bir önyükleyici ve bir bölüm tablosu dahil olmak üzere bir bilgisayarı başlatmak için gereken her şeyi içerir.

MBR, en fazla 4 bölüm oluşturabiliyordu ve boyutları 2TiB'ı geçemiyordu. Bu nedenle GUID Partition Table (GPT) duyruldu. GPT, 128 bölüm yaratabilir ve bölüm boyutu 8 ZiB'a kadar ulaşabilir.

Günümüzde sunucular hem MBR'ı hem de GPT'yi kullanabilmekle birlikte artılarından dolayı, diskler GPT olarak yapılandırılırlar.

Disk bölümlendirmenin, kullanıcı ve uygulamalar için alan sınırlaması ve veriyi izole etme gibi avantajları vardır.

Linux'da Aygıt İsimleri

Her işletim sisteminde disk ve bölümlerin isimlendirme şekli farklı olabilmekle birlikte Linux sistemlerde cihaz (device) isimleri genellikle aşağıdaki gibidir.

Device Name	Tanımı
/dev/fd0, /dev/fd1	Floopy Drive anlamında fd kullanılır. 0'dan başlar eklenen her floppy drive için sayı bir artar.
/dev/sda, /dev/sdb	SCSI driver kullanan harddisk anlamında sd kullanılır. a ile başlar, eklenen her disk için alfabetik sırayla devam eder.
/dev/sda1, /dev/sda2	Harddiskin bölümlerini temsil eder. Sda isimli cihazın 1'inci bölümü sda1, ikinci bölümü sda2 şeklinde numeratik devam eder.
/dev/scd0 veya /dev/sr0	Optik disk okuyucu olarak tanımlanır.

Dosya Sistemleri

Dosya sistemi bir depolama aygıtı içindeki verinin fiziksel konumunu içiren dizin ve veritabanı olarak değerlendirilebilir. Veriler, depolama aygıtlarının içerisinde dosya ve

dizinler halinde bulunmaktadır. Dosyaların boyutunu, konumunu ve hangi sektörlerin kullanıma hazır olduğunu belirleyen dosya sistemidir.

Bir dosya sistemi sadece dosyaları değil, aynı zamanda verilerin depolandığı sektör blok boyutu, bölüm bilgisi, dosya boyutu, nitelikler, dosya adı, dosya konumu ve dizin hiyerarşisi gibi bilgileri de depolar.

Windows için en çok kullanılan dosya sistemi FAT ve NTFS dosya sistemleri olmakla birlikte Linux sistemler Ext4, Ext3, xfs gibi dosya sistemlerini kullanılır.

Aralarındaki Farklar

- Maksimum Dosya Boyutu
- Maksimum Dosya Sayısı
- Maksimum Partition Boyutu
- Sıkıştırma
- Şifreleme
- Snapshot

Komutlar – lsblk, df, du

lsblk komutu sistemde bağlı bulunan storage ve blokların bir listesini verir. Çıktıyı incelediğimizde sda isimli bir diskimizin olduğu ve 4 bölümden oluştuğu görülmüyor. Ayrıca sr0 isimli bir tane de cdrom var.

Böülümlere baktığımızda sda1 dediğimiz alan disk üzerindeki MBR dediğimiz alandır. Bu alan silinirse sunucunuz reboot olduktan sonra bir daha açılmayacaktır. Çünkü ön yükleyici kod ve böülümler hakkında bilgi bu alanda bulunuyordu.

sda2, root dizininin bulunduğu aslıdan tüm dosya sisteminin bulunduğu alandır.

sda3, boot alanının kernel ve grub'un bulunduğu böülümdür. Bu bölümde silinirse sistem açılmaz.

sda4, home dizini için ayrılan alandır. Sistem üzerinde açılan kullanıcıların kullandığı home dizini için kullanılır.

Sütunları inceleyeceğimizde **NAME** alanı aygıtin ismidir. **MAJ:MIN** (Major:Minor) alanı kernel tarafından aygıtları dahili olarak tanımlamak için kullanılır. 8'in anlamı SCSI disk'tir. **RM**'nin anlamı kaldırılabilir bir aygit olduğunu gösterir. **SIZE**, aygitin boyutu, **RO** ise aygitin read-only olup olmadığını gösterir. **TYPE**, aygitin tipini gösterirken **MOUNTPOINT** ise aygitin bağlı olduğu alanı gösterir.

```
ozgur@ubuntu:~$ lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0    0 55.4M  1 loop /snap/core18/1944
loop1    7:1    0 55.5M  1 loop /snap/core18/1988
loop2    7:2    0 71.3M  1 loop /snap/lxd/16099
loop3    7:3    0 69.9M  1 loop /snap/lxd/19188
loop4    7:4    0 31.1M  1 loop /snap/snapd/11036
loop5    7:5    0 31.1M  1 loop /snap/snapd/10707
sda      8:0    0  20G  0 disk
└─sda1   8:1    0   1M  0 part
└─sda2   8:2    0 12G  0 part /
└─sda3   8:3    0 500M 0 part /boot
└─sda4   8:4    0   2G  0 part /home
sr0     11:0   1 1024M 0 rom
```

lsblk -m komutuyla aygıtlar üzerindeki izinleri görüntüleyebiliriz.

```
ozgur@ubuntu:~$ lsblk -m
NAME   SIZE OWNER GROUP MODE
loop0  55.4M root  disk  brw-rw----
loop1  55.5M root  disk  brw-rw----
loop2  71.3M root  disk  brw-rw----
loop3  69.9M root  disk  brw-rw----
loop4  31.1M root  disk  brw-rw----
loop5  31.1M root  disk  brw-rw----
sda    20G root  disk  brw-rw----
└─sda1   1M root  disk  brw-rw----
└─sda2   12G root  disk  brw-rw----
└─sda3  500M root  disk  brw-rw----
└─sda4   2G root  disk  brw-rw----
sr0   1024M root  cdrom brw-rw----
```

df komutu, sistemdeki mevcut dosya sistemi ve kullanılabilir alanının bilgisini verir. **df** komutu tek başına kullanıldığında alan boyutları bit olarak gösterildiğinden okuması zordur. **-h** argümanı ile anlayacağımız dilden konuşmasını sağlayabiliyoruz.

```
ozgur@ubuntu:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            951M    0  951M   0% /dev
tmpfs           199M  1.1M 198M   1% /run
/dev/sda2        12G  5.2G  6.0G  47% /
tmpfs           994M    0  994M   0% /dev/shm
tmpfs            5.0M    0  5.0M   0% /run/lock
tmpfs           994M    0  994M   0% /sys/fs/cgroup
/dev/sda3       469M 102M  332M  24% /boot
/dev/sda4        2.0G  34M  1.8G   2% /home
/dev/loop0        56M   56M    0 100% /snap/core18/1944
/dev/loop1        56M   56M    0 100% /snap/core18/1988
/dev/loop2        72M   72M    0 100% /snap/lxd/16099
/dev/loop3        70M   70M    0 100% /snap/lxd/19188
/dev/loop4        32M   32M    0 100% /snap/snapd/11036
/dev/loop5        32M   32M    0 100% /snap/snapd/10707
tmpfs           199M    0  199M   0% /run/user/1000
```

Belirli bir dizin hakkında bilgi almak için;

df -h /home

```
ozgur@ubuntu:~$ df -h /home/
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda4        2.0G  34M  1.8G  2% /home
```

Dosya sisteminin tiplerini görmek için -T argümanı kullanılır.

df -hT

```
ozgur@ubuntu:~$ df -hT
Filesystem  Type      Size  Used Avail Use% Mounted on
udev        devtmpfs  951M   0    951M  0% /dev
tmpfs       tmpfs     199M  1.1M  198M  1% /run
/dev/sda2   ext4      12G   5.2G  6.0G  47% /
tmpfs       tmpfs     994M   0    994M  0% /dev/shm
tmpfs       tmpfs     5.0M   0    5.0M  0% /run/lock
tmpfs       tmpfs     994M   0    994M  0% /sys/fs/cgroup
/dev/sda3   ext4      469M  102M  332M  24% /boot
/dev/sda4   ext4      2.0G  34M  1.8G  2% /home
/dev/loop0  squashfs  56M   56M   0  100% /snap/core18/1944
/dev/loop1  squashfs  56M   56M   0  100% /snap/core18/1988
/dev/loop2  squashfs  72M   72M   0  100% /snap/lxd/16099
/dev/loop3  squashfs  70M   70M   0  100% /snap/lxd/19188
/dev/loop4  squashfs  32M   32M   0  100% /snap/snapd/11036
/dev/loop5  squashfs  32M   32M   0  100% /snap/snapd/10707
tmpfs       tmpfs     199M   0    199M  0% /run/user/1000
```

du (Disk Usage) komutu, sunucu üzerindeki dizin ve dosyaların disk kullanım bilgilerini verir.

du -h /home

```
root@ubuntu:~# du -h /home
4.0K  /home/aydin/.local/share/nano
8.0K  /home/aydin/.local/share
12K   /home/aydin/.local
4.0K   /home/aydin/.cache
32K   /home/aydin
16K   /home/lost+found
16K   /home/ahmet
4.0K   /home/ozgur/.local/share/nano
8.0K   /home/ozgur/.local/share
12K   /home/ozgur/.local
8.0K   /home/ozgur/.ssh
4.0K   /home/ozgur/.config/procps
8.0K   /home/ozgur/.config/htop
16K   /home/ozgur/.config
4.0K   /home/ozgur/.cache
28M   /home/ozgur
8.0K   /home/ozkan
28M   /home
```

-s argümanı ile sadece dizin bilgisini gösterir. Alt dizinler hakkında bir çıktı vermez.

du -sh /home

```
root@ubuntu:~# du -sh /home
28M   /home
```

Ekran çıkışısından bir dizini hariç tutarak bilgi almak için exclude argümanını kullanıyoruz.

du -ah --exclude="/home/ozgur" /home/

```
root@ubuntu:~# du -ah --exclude="/home/ozgur" /home/
4.0K    /home/aydin/.local/share/nano
8.0K    /home/aydin/.local/share
12K    /home/aydin/.local
4.0K    /home/aydin/script
4.0K    /home/aydin/.bash_history
0      /home/aydin/.cache/motd.legal-displayed
4.0K    /home/aydin/.cache
4.0K    /home/aydin/.viminfo
32K    /home/aydin
16K    /home/lost+found
4.0K    /home/ahmet/.bashrc
4.0K    /home/ahmet/.bash_logout
4.0K    /home/ahmet/.profile
16K    /home/ahmet
4.0K    /home/ozkan/.bash_history
0      /home/ozkan/test.txt
8.0K    /home/ozkan
76K    /home/
```

Dizinin veya dosyanın en son düzenlendiği zamanı --time argümanı ile öğreniyoruz.

du -sh --time /home

```
root@ubuntu:~# du -sh --time /home
28M    2021-02-13 16:15      /home
```

fdisk, fstab ve partprobe nedir?

fdisk, Linux sistemlerde disk üzerinde yapılan bölüm oluşturma, genişletme, silme ve değiştirme gibi işlemleri yapmak için en çok kullanılan araçtır. fdisk ile bir disk üzerinde en fazla 4 bölüm oluşturabilirsiniz.

fdisk -l komutuyla sistemde varolan disk bölümlerini görebilirsiniz.

Device	Start	End	Sectors	Size	Type
/dev/sda1	2048	4095	2048	1M	BIOS boot
/dev/sda2	4096	25169919	25165824	12G	Linux filesystem
/dev/sda3	25169920	26193919	1024000	500M	Linux filesystem
/dev/sda4	26193920	30388223	4194304	2G	Linux filesystem

Disk ekleme ve genişletme bölümlerinde detaylıca kullanımını göreceğiz.

fstab, Linux işletim sistemlerinde dosya sistemini bir tablosunu tutar.

partprobe, işletim sisteminin bölüm tablosunu yeniden okumasını talep ederek işletim sistemi çekirdeğini bölüm tablosu değişiklikleri hakkında bilgilendiren bir araçtır. Eğer disk yapısında değişiklik yaptınız ve göremiyorsanız partprobe ile kernel'in değişiklikleri okuyup sisteme yansıtmasını isteyebilirsiniz.

Disk Ekleme - fdisk

Fiziksel veya sanal sunucunuza yeni bir disk bağladınız ve bu disk alanını kullanmak istiyorsunuz. Bu bölümde LVM olmayan standart bir disk yapılandırmamasına sahip bir sunucuya disk ekleyip kullanıma alacağız.

lsblk ile sunucumuzda tek bir disk olduğunu görüyoruz.(sda) Ardından sunucumuza diskimizi ekliyoruz. Kullandığınız platforma ve sunucu çeşidine göre değişiklik göstermekle birlikte çoğu sanallaştırma platformu sunucu çalışırken disk eklemenize izin verir. Eğer eklemenize izin vermiyorsa sunucuyu kapatıp yeni bir disk bağlayabilirsiniz.

```
root@ubuntu:~# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0      0 55.4M  1 loop /snap/core18/1944
loop1    7:1      0 55.5M  1 loop /snap/core18/1988
loop2    7:2      0 71.3M  1 loop /snap/1xd/16099
loop3    7:3      0 69.9M  1 loop /snap/1xd/19188
loop4    7:4      0 31.1M  1 loop /snap/snapd/11036
loop5    7:5      0 31.1M  1 loop /snap/snapd/10707
sda     8:0      0  20G  0 disk
└─sda1   8:1      0    1M  0 part
└─sda2   8:2      0   12G  0 part /
└─sda3   8:3      0  500M  0 part /boot
└─sda4   8:4      0    2G  0 part /home
sr0    11:0     1 1024M  0 rom
```

Diski ekledik ve tekrar lsblk komutu ile kontrol ediyoruz. sdb isimli bir görülmüyor.

```
root@ubuntu:~# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0      0 55.5M  1 loop /snap/core18/1988
loop1    7:1      0 55.4M  1 loop /snap/core18/1944
loop2    7:2      0 71.3M  1 loop /snap/1xd/16099
loop3    7:3      0 69.9M  1 loop /snap/1xd/19188
loop4    7:4      0 31.1M  1 loop /snap/snapd/11036
loop5    7:5      0 31.1M  1 loop /snap/snapd/10707
sda     8:0      0  20G  0 disk
└─sda1   8:1      0    1M  0 part
└─sda2   8:2      0   12G  0 part /
└─sda3   8:3      0  500M  0 part /boot
└─sda4   8:4      0    2G  0 part /home
sdb     8:16     1   1G  0 disk
sr0    11:0     1 1024M  0 rom
```

fdisk /dev/sdb komutuyla diskimizi yapılandırmaya başlıyoruz. Eğer seçenekleri görmek isterseniz m tuşuna basarak yardım alabilirsiniz. Yeni bir bölüm ekleyeceğimiz için n tuşuna basarak ilerliyoruz.

```
root@ubuntu:~# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.34).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x89d4a962.

Command (m for help):
```

Primary olarak 4 free ibaresine dikkat edersek, bu alan üzerinde 4 adet disk bölümü oluşturabileceğimiz bildiriliyor. İstersek bir bölümü extended yani genişletilmiş bölüm olarakda yapılandırabiliriz. Primary olarak devam ediyoruz.

```
Command (m for help): n
Partition type
  p  primary (0 primary, 0 extended, 4 free)
  e  extended (container for logical partitions)
Select (default p):
```

Bölüm numarası olarak sırayı bozmadan 1 yazarak devam ediyoruz.

```
Select (default p): p
Partition number (1-4, default 1):
```

Bölümün başlayacağı ve biteceği sektörleri seçmemiz isteniyor. Tüm alanı kullanacağımız için varsayılan değerlerle devam ederek bölümümüzü oluşturuyoruz.

```
Partition number (1-4, default 1): 1
First sector (2048-2097151, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-2097151, default 2097151):

Created a new partition 1 of type 'Linux' and of size 1023 MiB.

Command (m for help):
```

Değişikliklerin kaydedilmesi için w tuşu ile bölüm tablosunu değiştiriyoruz.

```
Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

lsblk komutuyla baktığımızda sdb diskinde sdb1 isimli bir bölüm görüyoruz.

```
root@ubuntu:~# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0    0 55.5M  1 loop /snap/core18/1988
loop1    7:1    0 55.4M  1 loop /snap/core18/1944
loop2    7:2    0 71.3M  1 loop /snap/1xd/16099
loop3    7:3    0 69.9M  1 loop /snap/1xd/19188
loop4    7:4    0 31.1M  1 loop /snap/snapd/11036
loop5    7:5    0 31.1M  1 loop /snap/snapd/10707
sda     8:0    0   20G  0 disk 
└─sda1   8:1    0    1M  0 part 
└─sda2   8:2    0   12G  0 part /
└─sda3   8:3    0  500M  0 part /boot
└─sda4   8:4    0    2G  0 part /home
sdb     8:16   0    1G  0 disk 
└─sdb1   8:17   0 1023M  0 part 
sr0    11:0    1 1024M  0 rom
```

Bölümü oluşturduk ama bir dosya sistemi ile biçimlendirmedik.

mkfs.ext4 /dev/sdb1 komutuyla bölümü ext4 dosya formatı olarak çevirdik.

```

root@ubuntu:/# mkfs.ext4 /dev/sdb1
mke2fs 1.45.5 (07-Jan-2020)
Creating filesystem with 261888 4k blocks and 65536 inodes
Filesystem UUID: cc10086e-a64b-421e-ac6c-8ca0d78deea8
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

```

Bölüm oluşturdu, dosya sistemini de belirledik ancak lsblk çıktısından da göreceğiniz üzere herhangi bir alana bağlı değil. Root dizini altında yeni isimli bir dizin oluşturduk. Bu alanı mount komutuyla yeni isimli dizine bağlayacağız. lsblk çıktısına baktığımızda sdb1'in /yeni dizinine bağlandığını görüyoruz.

mount /dev/sdb1 /yeni

```

root@ubuntu:/# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0    0 55.5M  1 loop /snap/core18/1988
loop1    7:1    0 55.4M  1 loop /snap/core18/1944
loop2    7:2    0 71.3M  1 loop /snap/1xd/16099
loop3    7:3    0 69.9M  1 loop /snap/1xd/19188
loop4    7:4    0 31.1M  1 loop /snap/snapd/11036
loop5    7:5    0 31.1M  1 loop /snap/snapd/10707
sda     8:0    0 20G  0 disk
└─sda1   8:1    0 1M  0 part
└─sda2   8:2    0 12G  0 part /
└─sda3   8:3    0 500M 0 part /boot
└─sda4   8:4    0 2G  0 part /home
sdb     8:16   0 1G  0 disk
└─sdb1   8:17   0 1023M 0 part /yeni
sr0    11:0    1 1024M 0 rom

```

Diskimizi /yeni dizinine bağladık. Fakat kalıcı hale getirmek için fstab içerisine kaydetmeliyiz. Her zaman olduğu gibi fstab'ın .bak uzantılı bir kopyasını alıyoruz.

blkid komutuyla oluşturduğumuz bölümün UUID'sini öğreniyoruz.

```

root@ubuntu:~# blkid
/dev/sda2: UUID="3e1a8564-3424-4112-971e-3a645e658bbf" TYPE="ext4" PARTUUID="c356a443-3f04-4516-a611-9549676
224fe"
/dev/sda3: UUID="b2552b1b-9959-4e06-bf9c-81acadd66d5" TYPE="ext4" PARTUUID="c6464f2a-9032-4405-894e-97faf57
dd43b"
/dev/sda4: UUID="04cbc2de-5ba9-4344-b6c8-edadaa801f76" TYPE="ext4" PARTUUID="86e3e9f9-55be-4761-bc55-70b3fa9
38c88"
/dev/sdb1: UUID="cc10086e-a64b-421e-ac6c-8ca0d78deea8" TYPE="ext4" PARTUUID="89d4a962-01"
/dev/loop0: TYPE="squashfs"
/dev/loop1: TYPE="squashfs"
/dev/loop2: TYPE="squashfs"
/dev/loop3: TYPE="squashfs"
/dev/loop4: TYPE="squashfs"
/dev/loop5: TYPE="squashfs"
/dev/sda1: PARTUUID="a27b8df8-e4fb-42d6-bc2c-87c1e7261625"

```

nano /etc/fstab komutuyla fstab dosyasını düzenliyoruz. UUID, mount edilen yeri, dosya tipini yazıp diğer seçenekleri varsayılan olarak vererek dosyayı kayıt ediyoruz. Kalıcı olarak yeni eklediğimiz disk yeni dizinine mount oluyor.

```

GNU nano 4.8                               /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options>      <dump>  <pass>
# / was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/3e1a8564-3424-4112-971e-3a645e658bbf / ext4 defaults 0 0
# /boot was on /dev/sda3 during curtin installation
/dev/disk/by-uuid/b2552b1b-9959-4e06-bf9c-81acadd66d5 /boot ext4 defaults 0 0
# /home was on /dev/sda4 during curtin installation
/dev/disk/by-uuid/04cbc2de-5ba9-4344-b6c8-edadaa801f76 /home ext4 defaults 0 0
/swap.img    none    swap    sw    0    0
/dev/disk/by-uuid/cc10086e-a64b-421e-ac6c-8ca0d78deea8 /yeni ext4 defaults 0 0

```

Reboot etmeden test etmek umount ediyoruz.

umount /dev/sdb1

Ardından mount -a komutuyla fstab içerisinde mount olması gereken aygıtları tekrar mount ediyoruz. Bu aşamadan sonra alan üzerinde kullanıcı ve grup yetkilendirmeleri yapabilir dilerseniz sadece rootun erişimine bırakabilirsiniz.

```

root@ubuntu:~# umount /dev/sdb1
root@ubuntu:~# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0    0 55.4M  1 loop /snap/core18/1944
loop1    7:1    0 55.5M  1 loop /snap/core18/1988
loop2    7:2    0 69.9M  1 loop /snap/1xd/19188
loop3    7:3    0 71.3M  1 loop /snap/1xd/16099
loop4    7:4    0 31.1M  1 loop /snap/snapd/11036
loop5    7:5    0 31.1M  1 loop /snap/snapd/10707
sda     8:0    0  20G  0 disk
└─sda1   8:1    0    1M  0 part
└─sda2   8:2    0   12G  0 part /
└─sda3   8:3    0  500M  0 part /boot
└─sda4   8:4    0    2G  0 part /home
sdb     8:16   0    1G  0 disk
└─sdb1   8:17   0 1023M 0 part
sr0    11:0   1 1024M 0 rom
root@ubuntu:~# mount -a
root@ubuntu:~# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0    0 55.4M  1 loop /snap/core18/1944
loop1    7:1    0 55.5M  1 loop /snap/core18/1988
loop2    7:2    0 69.9M  1 loop /snap/1xd/19188
loop3    7:3    0 71.3M  1 loop /snap/1xd/16099
loop4    7:4    0 31.1M  1 loop /snap/snapd/11036
loop5    7:5    0 31.1M  1 loop /snap/snapd/10707
sda     8:0    0  20G  0 disk
└─sda1   8:1    0    1M  0 part
└─sda2   8:2    0   12G  0 part /
└─sda3   8:3    0  500M  0 part /boot
└─sda4   8:4    0    2G  0 part /home
sdb     8:16   0    1G  0 disk
└─sdb1   8:17   0 1023M 0 part /yeni
sr0    11:0   1 1024M 0 rom

```

Disk Genişletme

Mevcut disk alanıımız yetmedi ve sanal diskimizi genişlettiğimizi varsayıyalım. Önceki sdb1 diskimizin kapasitesi 1 GB'dı. Biz 2 GB olarak yükseltmek istiyoruz. Sanallaştırma tarafında disk artırımıza sağladık. sdb diskine baktığımızda boyutunun 2G ama sdb1 bölümünün 1 GB olduğunu görüyoruz. 1GB'lık bir alan ekledik ancak kullanılabılır alanımıza dahil olmadı.

```
root@ubuntu:~# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0     0 55.4M  1 loop /snap/core18/1944
loop1    7:1     0 55.5M  1 loop /snap/core18/1988
loop2    7:2     0 71.3M  1 loop /snap/1xd/16099
loop3    7:3     0 31.1M  1 loop /snap/snapd/10707
loop4    7:4     0 69.9M  1 loop /snap/1xd/19188
loop5    7:5     0 31.1M  1 loop /snap/snapd/11036
sda      8:0     0  20G  0 disk 
└─sda1   8:1     0    1M  0 part 
└─sda2   8:2     0   12G  0 part /
└─sda3   8:3     0   500M 0 part /boot
└─sda4   8:4     0    2G  0 part /home
sdb      8:16    0   2G  0 disk 
└─sdb1   8:17    0 1023M 0 part /yeni
sr0     11:0    1 1024M 0 rom
```

ls -ltr komutu ile /yeni dizininin içindeki dosyalara bakıyoruz.

```
root@ubuntu:~# ls -ltr /yeni/
total 16
drwx----- 2 root root 16384 Feb 14 00:16 lost+found
-rw-r--r-- 1 root root     0 Feb 14 00:20 test5
-rw-r--r-- 1 root root     0 Feb 14 00:20 test4
-rw-r--r-- 1 root root     0 Feb 14 00:20 test3
-rw-r--r-- 1 root root     0 Feb 14 00:20 test2
-rw-r--r-- 1 root root     0 Feb 14 00:20 test1
```

fdisk /dev/sdb komutuyla disk yapılandırmamıza başlıyoruz.

d ile /dev/sdb1'i siliyoruz. Bu alandaki değişiklikler w tuşu ile yazmadan uygulanmayacaktır. Bu nedenle işlemler tamamlanana kadar w tuşuna basmıyoruz. Diğer türlü sildiğiniz bölümün içindeki veriler tamamen silinir.

n ile yeni bir bölüm oluşturmaya başlıyoruz. Başlangıç ve bitiş bloklarını varsayılan olarak bırakıyoruz. Burada bitiş bloğu diski genişlettiğimiz için önceki bitiş noktasından daha büyütür. Diskin tamamını sdb1 olarak kullanmak istediğimizden varsayılan olarak bıraktık. Eğer tamamını değil bir bölümünü kullanacak olsaydık hesaplama yaparak bu değeri girmeliydik. Eğer genişletilmiş bölümün tamamını değil bir kısmını kullanacaksanız bu sefer sildiğiniz bölümün kapladığı bloklardan küçük bir alan vermemeye dikkat etmelisiniz. Böyle bir durumda diskde veri varsa kaybedersiniz.

Büyük genişletilmiş alanı sdb1 e verdikten sonra bölüm yaratıldı bilgisi geliyor. Ardından ext4 formatı için önceden oluşmuş imzayı kaldırma isteyip istemeyeceğimizi soruyoruz. Hayır diyerek devam ediyoruz. w ile değişiklikleri bölüm tablosuna yazıyoruz.

```

root@ubuntu:~# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.34).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): d
Selected partition 1
Partition 1 has been deleted.

Command (m for help): n
Partition type
  p  primary (0 primary, 0 extended, 4 free)
  e  extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1):
First sector (2048-4194303, default 2048):
Last sector, +/sectors or +/-size{K,M,G,T,P} (2048-4194303, default 4194303):

Created a new partition 1 of type 'Linux' and of size 2 GiB.
Partition #1 contains a ext4 signature.

Do you want to remove the signature? [Y]es/[N]o: N

Command (m for help): w

The partition table has been altered.
Syncing disks.

```

lsblk komutumuzla baktığımızda alanımızın artmış olduğunu, ls -ltr komutu ile de dosyalarımızın durduğunu görüyoruz.

```

root@ubuntu:~# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0    0 55.4M  1 loop /snap/core18/1944
loop1    7:1    0 55.5M  1 loop /snap/core18/1988
loop2    7:2    0 71.3M  1 loop /snap/lxd/16099
loop3    7:3    0 31.1M  1 loop /snap/snapd/10707
loop4    7:4    0 69.9M  1 loop /snap/lxd/19188
loop5    7:5    0 31.1M  1 loop /snap/snapd/11036
sda     8:0    0   20G  0 disk
└─sda1   8:1    0   1M  0 part
└─sda2   8:2    0  12G  0 part /
└─sda3   8:3    0 500M  0 part /boot
└─sda4   8:4    0   2G  0 part /home
sdb     8:16   0   2G  0 disk
└─sdb1   8:17   0   2G  0 part /yeni
sr0    11:0    1 1024M 0 rom
root@ubuntu:~# ls -ltr /yeni/
total 16
drwx----- 2 root root 16384 Feb 14 00:16 lost+found
-rw-r--r-- 1 root root      0 Feb 14 00:20 test5
-rw-r--r-- 1 root root      0 Feb 14 00:20 test4
-rw-r--r-- 1 root root      0 Feb 14 00:20 test3
-rw-r--r-- 1 root root      0 Feb 14 00:20 test2
-rw-r--r-- 1 root root      0 Feb 14 00:20 test1

```

Mevcut Diskde İkincil Bölüm Oluşturmak

Yeni sdb diskimiz üzerinden devam ediyoruz. Diskimizin boyutunu 1 GB daha artırıyoruz.

lsblk komutumuzla kontrol edip artırdığımız alanı görüntüleyiyoruz.

```
root@ubuntu:~# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0     0 55.4M  1 loop /snap/core18/1944
loop1    7:1     0 71.3M  1 loop /snap/lxd/16099
loop2    7:2     0 69.9M  1 loop /snap/lxd/19188
loop3    7:3     0 55.5M  1 loop /snap/core18/1988
loop4    7:4     0 31.1M  1 loop /snap/snapd/10707
loop5    7:5     0 31.1M  1 loop /snap/snapd/11036
sda      8:0     0 20G   0 disk 
└─sda1   8:1     0 1M    0 part 
  └─sda2   8:2     0 12G   0 part /
    └─sda3   8:3     0 500M  0 part /boot
      └─sda4   8:4     0 2G    0 part /home
sdb      8:16    0 3G    0 disk 
└─sdb1   8:17    0 2G    0 part /yeni
sr0     11:0    1 1024M 0 rom
```

fdisk /dev/sdb komutuyla disk yapılandırmamıza gidiyoruz. n tuşu ile yeni bir bölüm oluşturacağımızı söylüyoruz. 1 primary bölümün olduğunu ve 3 tane daha oluşturabileceğimizi görüyoruz. p tuşu ve 2 numaralı bölümü seçerek devam ediyoruz. First sector alanına dikkat ederseniz 2048'den başlamıyor. Çünkü 2048'den 4194304'e kadar sdb1 kullanıyor. Boş olan sektörleri seçip devam ediyoruz. w ilke bölüm tablomuza yazıyoruz.

```
root@ubuntu:~# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.34).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): n
Partition type
  p  primary (1 primary, 0 extended, 3 free)
  e  extended (container for logical partitions)
Select (default p): p
Partition number (2-4, default 2): 2
First sector (4194304-6291455, default 4194304):
Last sector, +/-sectors or +/-size{K,M,G,T,P}  (4194304-6291455, default 6291455):

Created a new partition 2 of type 'Linux' and of size 1 GiB.

Command (m for help): w
The partition table has been altered.
Syncing disks.
```

lsblk ile kontrol ediyoruz ve sdb2 isimli bölümümüzü görüyoruz. sdb1 gibi dosya sistemi ile biçimlendiriyor, bir klasöre mount edip fstab'a yazdıktan sonra kullanıma hazır.

```
root@ubuntu:~# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0     0 55.4M  1 loop /snap/core18/1944
loop1    7:1     0 71.3M  1 loop /snap/lxd/16099
loop2    7:2     0 69.9M  1 loop /snap/lxd/19188
loop3    7:3     0 55.5M  1 loop /snap/core18/1988
loop4    7:4     0 31.1M  1 loop /snap/snapd/10707
loop5    7:5     0 31.1M  1 loop /snap/snapd/11036
sda      8:0     0 20G   0 disk 
└─sda1   8:1     0 1M    0 part 
  └─sda2   8:2     0 12G   0 part /
    └─sda3   8:3     0 500M  0 part /boot
      └─sda4   8:4     0 2G    0 part /home
sdb      8:16    0 3G    0 disk 
└─sdb1   8:17    0 2G    0 part /yeni
  └─sdb2   8:18    0 1G    0 part
sr0     11:0    1 1024M 0 rom
```

GPT Bölüm Oluşturmak - gdisk

Şimdiye kadar MBR bölüm oluşturarak ilerledik. GPT bölüm oluşturmak için aracımız gdisk'tir. Sunucumuza 3 ncü bir disk ekliyoruz. **lsblk** ile kontrol ettiğimizde sdc isimli diskimizin geldiğini görüyoruz.

```
root@ubuntu:~# lsblk | grep sd
sda      8:0    0    20G  0 disk
└─sda1   8:1    0     1M  0 part
  └─sda2   8:2    0    12G  0 part /
  └─sda3   8:3    0   500M  0 part /boot
  └─sda4   8:4    0     2G  0 part /home
sdb      8:16   0     3G  0 disk
└─sdb1   8:17   0     2G  0 part /yeni
  └─sdb2   8:18   0     1G  0 part
sdc      8:32   0     2G  0 disk
```

gdisk /dev/sdc komutuyla GPT bölüm oluşturmak için sihirbazı başlatıyoruz.

```
root@ubuntu:~# gdisk /dev/sdc
GPT fdisk (gdisk) version 1.0.5

Partition table scan:
  MBR: not present
  BSD: not present
  APM: not present
  GPT: not present

Creating new GPT entries in memory.

Command (? for help):
```

fdisk'deki gibi yeni bölüm oluşturmak için **n** tuşuna basıyoruz. Bölüm numarasını **1** olarak giriyoruz. Dikkat ederseniz 1-128 arası bir numara girebiliyoruz. Çünkü GPT ile bir disk üzerinde 128 tane bölüm oluşturabiliyoruz. Devamında sector alanlarını belirledik. Bize dosya sistemi tipi soruyor ve varsayılan olarak **8300 Linux** yazılıdır. Eğer farklı bir system seçecekseniz **L** tuşu seçenekleri görebilirsiniz. **Linux Filesystem** seçerek devam ediyoruz. Değişiklikleri kaydediyoruz ve sihirbazı tamamlıyoruz.

```
Command (? for help): n
Partition number (1-128, default 1):
First sector (34-4194270, default = 2048) or {+-}size{KMGTP}:
Last sector (2048-4194270, default = 4194270) or {+-}size{KMGTP}:
Current type is 8300 (Linux filesystem)
Hex code or GUID (L to show codes, Enter = 8300):
Changed type of partition to 'Linux filesystem'

Command (? for help): w

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING
PARTITIONS!!

Do you want to proceed? (Y/N): Y
OK; writing new GUID partition table (GPT) to /dev/sdc.
The operation has completed successfully.
```

lsblk komutuyla baktığımızda **sdc1** isimli bölümü görüyoruz. Yine alanı kullanmak için diski formatlamak, bir dizine bağlamak ve fstab içerisinde yazmayı unutmuyoruz.

```
root@ubuntu:~# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0    7:0      0 55.4M  1 loop /snap/core18/1944
loop1    7:1      0 55.5M  1 loop /snap/core18/1988
loop2    7:2      0 71.3M  1 loop /snap/xd/16099
loop3    7:3      0 69.9M  1 loop /snap/xd/19188
loop4    7:4      0 31.1M  1 loop /snap/snapd/11036
loop5    7:5      0 31.1M  1 loop /snap/snapd/10707
sda     8:0      0  20G  0 disk 
└─sda1   8:1      0   1M  0 part 
└─sda2   8:2      0 12G  0 part /
└─sda3   8:3      0 500M 0 part /boot
└─sda4   8:4      0   2G  0 part /home
sdb     8:16     0   3G  0 disk 
└─sdb1   8:17     0   2G  0 part /yeni
└─sdb2   8:18     0   1G  0 part 
sdc     8:32     0   2G  0 disk 
└─sdc1   8:33     0   2G  0 part 
sr0    11:0     1 1024M 0 rom
```

Sistem Diskini Genişletme - Gparted

İşletim sistemi diskini fdisk ile yukarıdaki gibi genişletebiliriz. Ancak GPARTED isimli grafik arayüzüne sahip bir uygulama ile bu işlemi gerçekleştirelim. Gparted ISO dosyasını indirip sunucumuzu bu ISO'dan çalışacak şekilde boot ederek kullanabiliyoruz

Sanallaştırma uygulamamızda diskin boyutunu 2 GB artırdık.

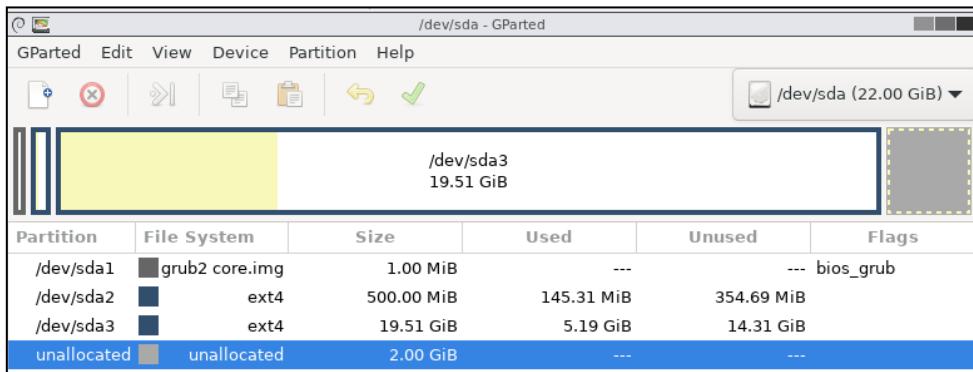
<https://gparted.org/download.php> adresinden gparted uygulamasını indiriyoruz.

Sunucumuzun sanal olan optik sürücüsüne bağlıyor ve sunucuyu CD'den boot edecek şekilde ayarlıyoruz.

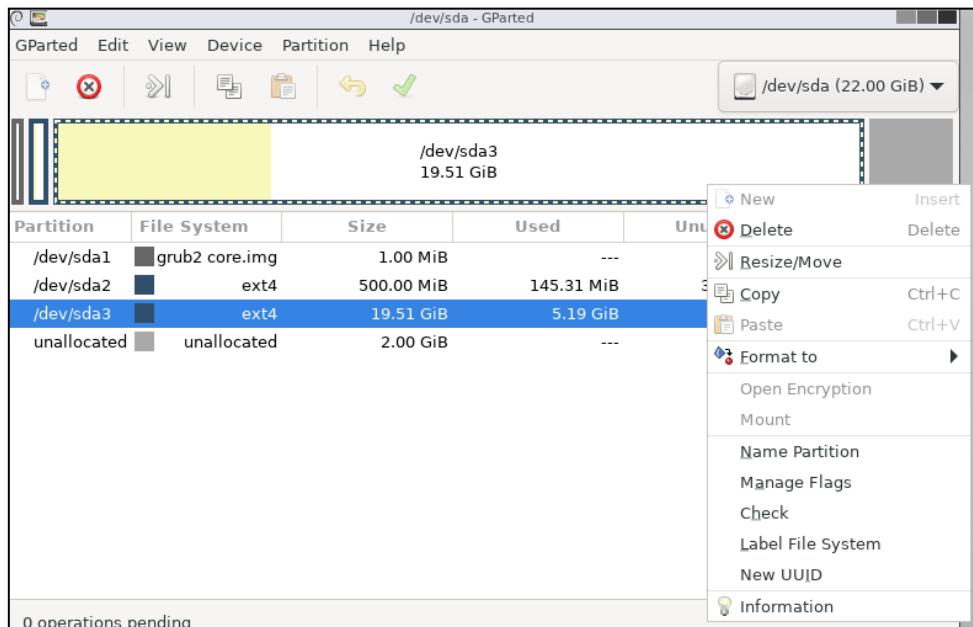
Sunucu boot ettiğinde Gparted uygulamasının ekranı bizi karşılıyor. Gparted Live seçerek ilerliyoruz.



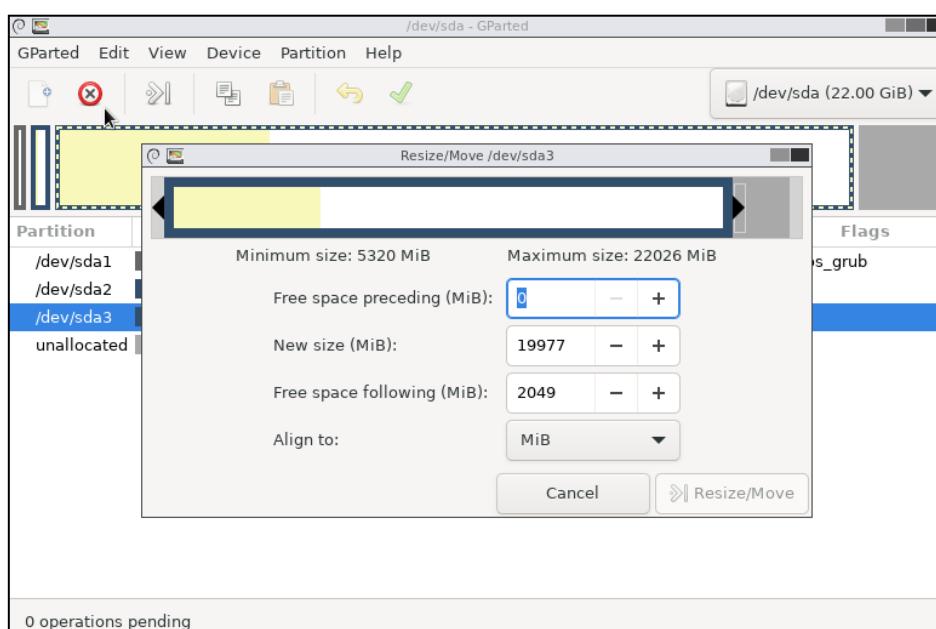
Gparted ekranı açıldıktan sonra root bölümünün /dev/sda3 olduğunu görüyoruz.



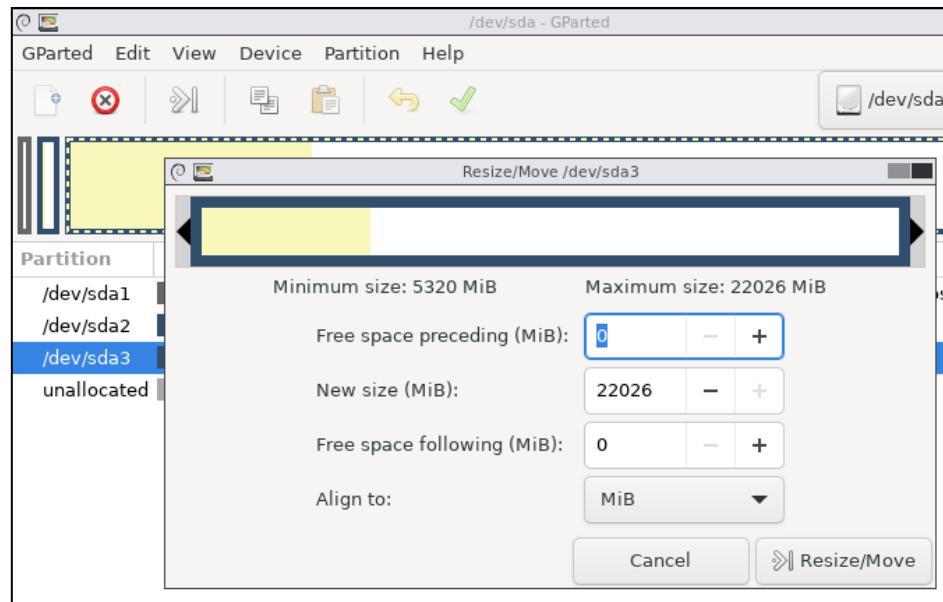
/dev/sda3 bölümüne sağ tıklayarak Resize/Move seçerek diskı genişleteceğimiz pencereye ulaşıyoruz.



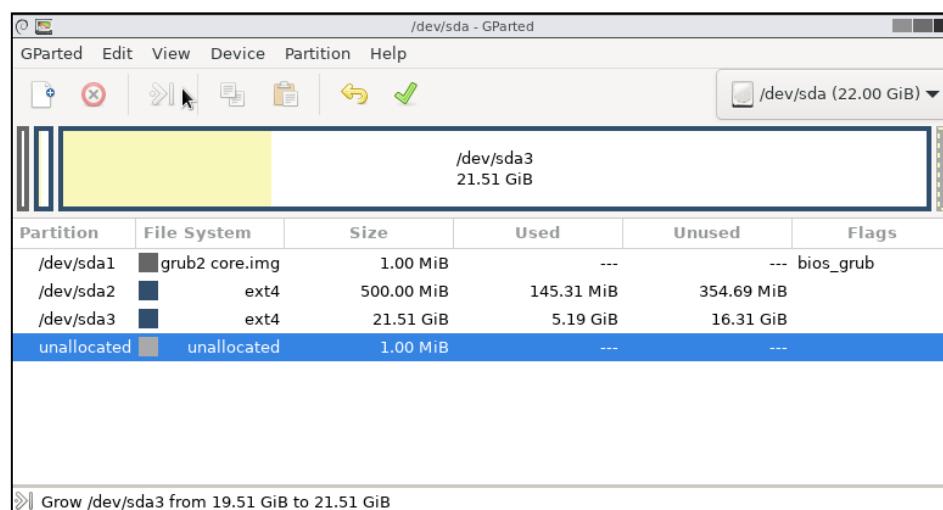
Sarı bölümler diskin dolu olduğu bloklar, beyaz olan alanlar boş ve gri kısım diske atayabileceğimiz alanı tanımlıyor. Mouse ile gri kısımda bulunan ok işaretini tutup en son kısma getiriyoruz.



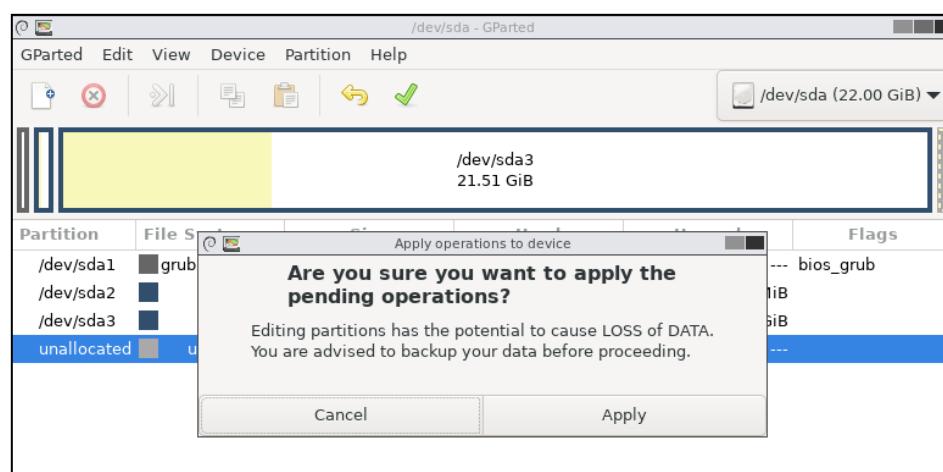
Resize/Move butonuna basarak tanımlamamızı yapıyoruz.



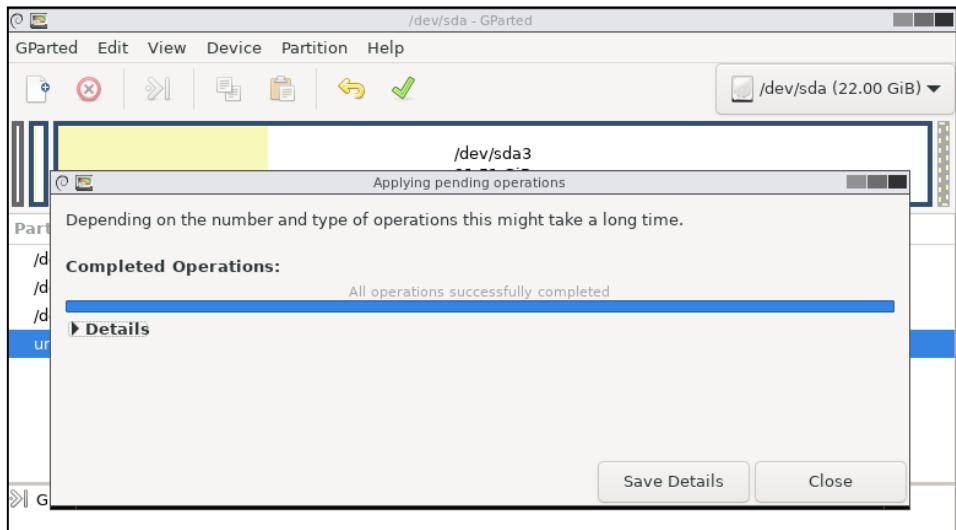
Orta kısmında yeşil olan ok tuşuna basarak yaptığımız değişikliği onaylıyoruz.



Değişiklikler konusunda bir uyarı penceresi çıkıyor. Apply diyerek onaylıyoruz.



Değişiklikler tamamlandığına dair bir bildirim alıyoruz. Close diyerek gparted uygulamasından tamamen çıkararak sunucuyu yeniden başlatıyoruz.



Sunucumuz açıldıktan sonra lsblk ile kontrolleri sağlıyoruz.

```
root@ubuntu:~# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
fd0     2:0    1   4K  0 disk 
loop0   7:0    0 29.9M 1 loop /snap/snapd/8542
loop1   7:1    0  55M 1 loop /snap/core18/1880
loop2   7:2    0 71.3M 1 loop /snap/lxd/16099
sda     8:0    0  22G 0 disk 
|---sda1  8:1    0   1M 0 part 
|---sda2  8:2    0 500M 0 part /boot
|---sda3  8:3    0 21.5G 0 part /
sr0    11:0   1 1024M 0 rom
```

Swap Alanı Oluşturmak ve Genişletmek

Swap, işletim sisteminin artık RAM'de tutamayacağı verileri geçici olarak depolaması için ayrılan sabit sürücüdeki depolama alanıdır. RAM'de yeterli alan kalmadığında, uygulama verilerini tutmak için kullanılır. RAM'de yer kalmadığında geçici olarak swap alanı üzerinden işlem yapmak hayat kurtarıcı olabilir. Sabit disk yavaş olduğu için RAM'deki gibi bir performans alınmaz ama geçici olarak RAM yetersizliği problemine çare olur.

free komutu ile sunucu üzerindeki RAM ve Swap alanlarının bilgisini görebiliriz.

```
root@ubuntu:~# free
              total        used        free      shared  buff/cache   available
Mem:       1969356     1346176      76124        976     547056      475532
Swap:      2097148          524     2096624
```

swapon --show komutuyla swap alanının bilgisi görülür.

```
root@ubuntu:~# swapon --show
NAME      TYPE SIZE USED PRIO
/swapon.img file 2G 524K -2
```

root altında swap bölümü görebiliriz.

```
root@ubuntu:/# ls -ltr | grep swap
-rw----- 1 root root 2057306112 Feb 2 23:01 swap.img
```

Mevcut swap alanımızı swapon ile görüntüledikten sonra

sudo dd if=/dev/zero of=/swapfile bs=1G count=1 komutuyla root altında 1G boyutunda bir swap dosyası oluşturuyoruz.

```
root@ubuntu:/# sudo dd if=/dev/zero of=/swapfile bs=1G count=1
1+0 records in
1+0 records out
1073741824 bytes (1.1 GB, 1.0 GiB) copied, 18.7178 s, 57.4 MB/s
```

chmod 600 /swapfile komutuyla swap dosyasının izinleri ayarlıyoruz.

```
root@ubuntu:/# chmod 600 /swapfile
root@ubuntu:/# ls -ltr /swapfile
-rw----- 1 root root 1073741824 Feb 14 17:44 /swapfile
```

mkswap /swapfile komutuyla alanı swap olarak tanımlıyoruz.

```
root@ubuntu:/# mkswap /swapfile
Setting up swapspace version 1, size = 1024 MiB (1073737728 bytes)
no label, UUID=a89f5e4c-3c7f-4db1-b34e-853110ba2c87
```

swapon /swapfile komutuyla alanı swap alanı olarak açıyoruz.

```
root@ubuntu:/# swapon /swapfile
```

nano /etc/fstab komutuyla swap dosyasını kalıcı hale getiriyoruz.

```
GNU nano 4.8                               /etc/fstab                         Modified
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>      <dump>  <pass>
# / was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/3e1a8564-3424-4112-971e-3a645e658bbf / ext4 defaults 0 0
# /boot was on /dev/sda3 during curtin installation
/dev/disk/by-uuid/b2552b1b-9959-4e06-bf9c-81acadd66d5 /boot ext4 defaults 0 0
# /home was on /dev/sda4 during curtin installation
/dev/disk/by-uuid/04cbc2de-5ba9-4344-b6c8-edadaa801f76 /home ext4 defaults 0 0
/swap.img        none    swap    sw    0    0
/dev/disk/by-uuid/cc10086e-a64b-421e-ac6c-8ca0d78deea8 /yeni ext4 defaults 0 0
/swapfile        none    swap    sw    0    0
```

Son olarak kontrol ediyoruz ve yeni eklediğimiz alan ile swap alanımızın genişlediğini görüyoruz.

```
root@ubuntu:/# free
              total        used         free       shared  buff/cache   available
Mem:       2035060      502604     1107264          1116      425192      1371928
Swap:      3057656           0     3057656
```

swapoff komutu swap alanlarını kapatır. **swapoff -a** komutunu kullanarak bütün alanları kapatıp **swapon -a** komutuyla tüm alanları açabilirsiniz.

swapoff /swapfile komutu ile sadece belirttiğimiz swap alanını kapatmış oluruz. Aynı şekilde **swapon /swapfile** ile bu alanı swap alanı olarak açabiliriz.

e2fsck

e2fsck, ext2 / ext3 / ext4 dosya sistemlerini kontrol etmek için kullanılır. Sunucunuzun uygun olmayan şekilde bir anda kapatılmışsa yada sunucu diskleri uygunsuz bir şekilde çıkarılmışsa dosya sistemi zarar görmüş olabilir. Dosya sistemindeki oluşan hasarları bulmak ve onarmak için kullanılan bir araçtır.

Bir bölümü e2fsck ile kontrol etmek için o bölümün umount olmuş olması gereklidir. e2fsck komutuyla bölümü denetledikten sonra tekrar mount edebiliriz.

```
root@ubuntu:~# umount /dev/sdb1
root@ubuntu:~# e2fsck /dev/sdb1
e2fsck 1.45.5 (07-Jan-2020)
/dev/sdb1: clean, 16/131072 files, 13100/524032 blocks
root@ubuntu:~# mount -a
```

e2fsck -p /dev/sdb1 komutuyla otomatik tamir etme işlemi başlatılır.

e2fsck -n /dev/sdb1 komutuyla sadece bölüm kontrol edilir, herhangi bir değişiklik yapılmaz.

e2fsck -f /dev/sdb1 komutuyla bölüm taramanızı rağmen temiz cevabı alıyorsanız e2fsck'ya -f argümanı ile tarama işlemini yapması için zorlayabilirsiniz.

```
root@ubuntu:~# e2fsck -p /dev/sdb1
/dev/sdb1: clean, 16/131072 files, 13100/524032 blocks
root@ubuntu:~# e2fsck -n /dev/sdb1
e2fsck 1.45.5 (07-Jan-2020)
/dev/sdb1: clean, 16/131072 files, 13100/524032 blocks
root@ubuntu:~# e2fsck -f /dev/sdb1
e2fsck 1.45.5 (07-Jan-2020)
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
/dev/sdb1: 16/131072 files (0.0% non-contiguous), 13100/524032 blocks
```

Diski Sunucudan Kaldırmak

Sunucu üzerindeki disk ile işiniz bitti ve sistemden kaldırma istiyorsunuz. Eğer diskin içindeki veriler tutup saklamak istiyorsanız umount komutunu kullanarak diski bağlı olduğu dizinden ayırayabilirsiniz. Eğer diskin içindeki verileri silip raw hale getirmek istiyorsanız diski formatladığınız fdisk, gdisk veya parted araçları disk üzerindeki bölümleri siliyoruz. Devamında sunucumuzdan kaldırıyoruz.

```

root@ubuntu:/# gdisk /dev/sdc
GPT fdisk (gdisk) version 1.0.5

Partition table scan:
  MBR: protective
  BSD: not present
  APM: not present
  GPT: present

Found valid GPT with protective MBR; using GPT.

Command (? for help): d
Using 1

Command (? for help): w

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING
PARTITIONS!!

Do you want to proceed? (Y/N): Y
OK; writing new GUID partition table (GPT) to /dev/sdc.
The operation has completed successfully.

root@ubuntu:/# lsblk
NAME  MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0  7:0    0   55.4M  1 loop /snap/core18/1944
loop1  7:1    0   55.5M  1 loop /snap/core18/1988
loop2  7:2    0   71.3M  1 loop /snap/ldxd/16099
loop3  7:3    0   69.9M  1 loop /snap/ldxd/19188
loop4  7:4    0   31.1M  1 loop /snap/snapd/11036
loop5  7:5    0   31.1M  1 loop /snap/snapd/10707
sda    8:0    0   20G  0 disk 
└─sda1  8:1    0   1M  0 part 
└─sda2  8:2    0   12G  0 part /
└─sda3  8:3    0   500M 0 part /boot
└─sda4  8:4    0   2G  0 part /home
sdb    8:16   0   3G  0 disk 
└─sdb1  8:17   0   2G  0 part /yeni
└─sdb2  8:18   0   1G  0 part 
sdc    8:32   0   2G  0 disk 
sr0    11:0   1 1024M 0 rom

```

16 - LVM (Logical Volumes) Yönetimi

LVM Nedir?

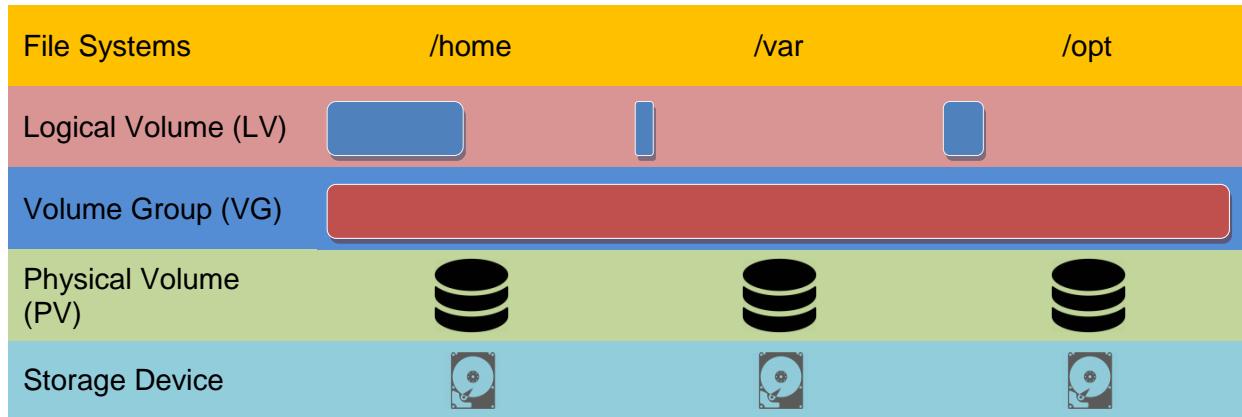
LVM (Logical Volume Management) kullanıcılarla daha kolay ve esnek bir depolama yönetimi deyimi sunar. Klasik disk yönetiminin sınırlamalarını aşarak depolama birimlerini dinamik olarak kolay bir şekilde artırmakla birlikte depolama birimlerin bir havuz altında toplayarak yönetimi kolaylaştırır.

LVM bir veya daha fazla fiziksel disk grubunu bir Volume Group altında toplayarak Volume'ler oluşturur. Eğer Logical Volume'de yer kalmaz ise Volume Group'dan yer alınarak genişletilebilir. Volume Group'ta alan ihtiyacı duyulursa fiziksel veya sanal bir disk eklenerek Volume Group genişletilir.

Logical Volume'lerin boyutları artırılabilir gibi eğer dosya sistemi destekliyorsa boyutu da azaltılabilir.

LVM'in diğer bir artısı da Logical Volume'ün snapshot'ı (anlık görüntüsü) alınarak mevcut durumu korunabilir. İhtiyaç durumunda önceki haline geri dönülebilir. Yedek almak için de kullanılabilir.

LVM mantıksal birimlerini kullanmanın üçüncü önemli avantajı, arızalı donanımı kolayca değiştirme seçeneğidir. Bir sabit disk arızalıysa (bad sector vb.), veriler birim grubu içinde başka bir fiziksel bölüme taşınabilir (pvmove komutu aracılığıyla), arızalı disk daha sonra birim grubundan çıkarılabilir ve yeni bir sabit disk, herhangi bir kesinti olmadan dinamik olarak eklenebilir.



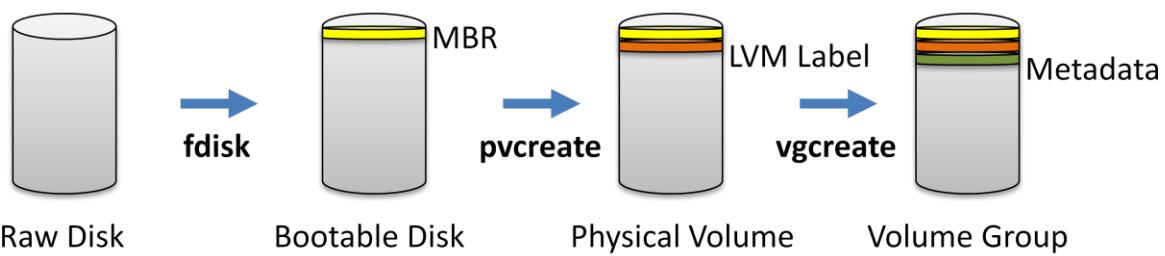
LVM yapısı fiziksel veya sanal diskler üzerine kurgulanmıştır. Bu yapıda sırayla Physical Volume (PV), Volume Group (VG) ve Logical Volume (LV) bulunmaktadır.

Physical Volume : Fiziksel depolama cihazlarını bir araya getiren katmandır. Yönetim için diske bir başlık yazarak katmana dahil eder. Bu katmandaki fiziksel diskler HDD, SSD olabilir. Virtual disklerden de Physical Volume oluşturulabilir.

Volume Group : LVM yapısının merkezidir. Physical Volume'leri birleştirerek bir depolama havuzu oluşturur.

Logical Volume : Volume Group içinden oluşturulan bölümlerdir. Bir nevi fiziksel diskteki bölümler gibi denebilir. Uygulama ve kullanıcı verileri bu katmanla etkileşimde bulunur. Aynı Volume Group üzerindeki Logical Volume'ler birbirlerinden farklı dosya sistemleriyle oluşturulabilir.

Bir Logical Volume oluşturmak için diagramdaki süreç işlenilir.



LVM Label, fiziksel bölgeye atanmış benzersiz bir UUID'dir.

```
PV UUID          DdKLIU-U2Ps-Gg0t-nLDH-k8p2-1MUT-It9Wsg
```

Metadata, Volume Group içinde yer alan fiziksel bölümler, logical bölümler, volumelerin boyutları, ne zaman oluşturuldukları, kaldırıldıkları kısacası Volume Group'la ilgili bütün bilgileri içinde barındırır.

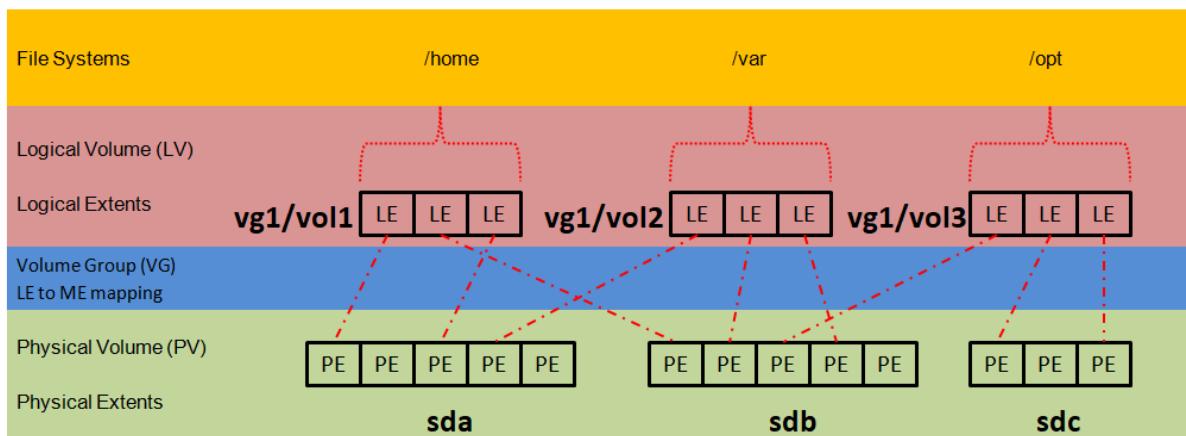
Logical Volume'un linear, striped ve mirrored olarak çeşitleri vardır.

Son olarak LVM yapılandırma dosya ve klasörleri /etc/lvm dizini altında tutulur.

Extents - LE ve PE Mapping

Bir VG içindeki her bir bölüm extents isminde sabit boyutlu küçük parçalara bölünür. Physical Volume'lerdeki parçalara Physical Extents (PE), Logical Volume'lerdeki parçalara Logical

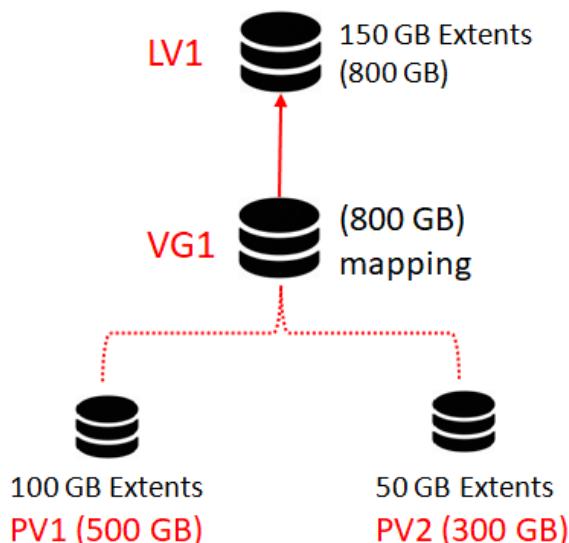
Extents (LE) denir. Logical Volume'lerin kapsamı ve boyutu LE ve PE eşleşmesiyle oluşur. Bu eşleşme sayesinde Logical Volume'ler kolay bir şekilde küçültülp büyütülebilir.



Linear Logical Volumes

Bir Logical Volume'un birden fazla fiziksel diskle oluşmasıdır. Veri ilk önce bir diske yazılmaya başlar. Disk dolduğu zaman diğer diskten verilen alana yazmaya devam eder.

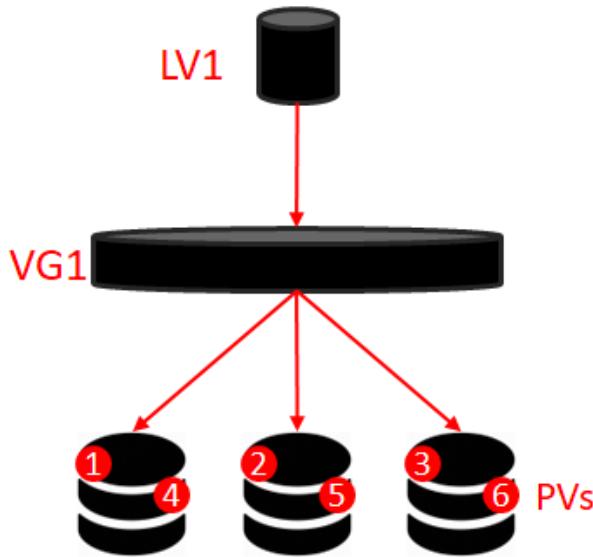
Linear Volume



Striped Logical Volume

Veriyi tek bir diske yazmak yerine belirlenen stripe size (block size diye de adlandırılır.) boyutuna göre birden fazla diske sırayla yazabilen Volume'lerdir. Örneğin, stripe size 128 KB olarak belirlenmişse ve sizin işlem yaptığınız veri, bu boyuttan büyükse verileriniz 128 KB'lık parçalara bölünerek sırayla disklere yazılmaya başlar. Genelde I/O performansını artırmak için kullanılır.

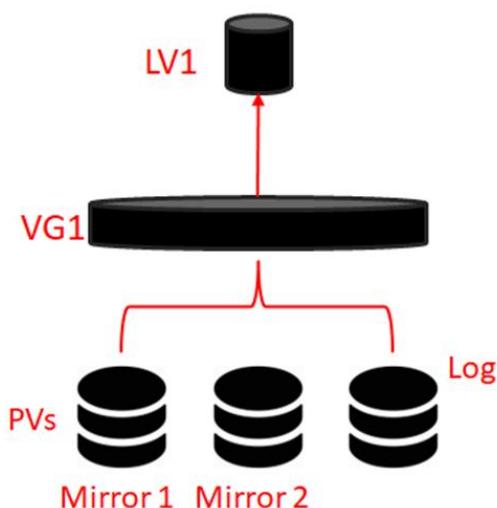
Striped Volume



Mirrored Logical Volume

Birden fazla Physical Volume üzerinde verinin kopyalarını tutar. Bu şekilde Physical Volume'e bağlı bir diskte arıza oluşması durumunda, veri kaybına karşı sizi korur. Mirrored Logical Volume'deki bir fiziksel disk arızalanadığında, volume linear volume olarak erişilebilir hale gelir. Log dosyası, mirror'ların hangi blocklarının senkronize olduğunu tutar.

Mirrored Volume



LVM Yapılandırma Dosyası

LVM'in yapılandırma dosyaları, arşiv ve yedekleri /etc/lvm dizini altında bulunur. Bunlardan en önemlisi archive ve backup dizinleridir.

```
root@client:~# ls -ltr /etc/lvm/
total 116
-rw-r--r-- 1 root root 2301 Feb 13 2020 lvmlocal.conf
-rw-r--r-- 1 root root 102257 Feb 13 2020 lvm.conf
drwxr-xr-x 2 root root 4096 Jul 31 2020 profile
drwx----- 2 root root 4096 Feb 25 19:18 archive
drwx----- 2 root root 4096 Feb 25 19:18 backup
```

Archive dizini içerisinde Volume Group'larda yapılan her türlü değişikliğin kaydı tutulur.

```
root@client:~# ls -ltr /etc/lvm/archive/
total 104
-rw----- 1 root root 1105 Feb 14 18:18 lvmgrup_00000-265431397.vg
-rw----- 1 root root 1107 Feb 14 18:24 lvmgrup_00001-809848792.vg
-rw----- 1 root root 1535 Feb 14 18:59 lvmgrup_00002-1117760818.vg
-rw----- 1 root root 1708 Feb 14 20:27 lvmgrup_00003-1156262348.vg
-rw----- 1 root root 1535 Feb 14 20:35 lvmgrup_00004-1236020261.vg
-rw----- 1 root root 1708 Feb 14 20:37 lvmgrup_00005-783776845.vg
-rw----- 1 root root 1107 Feb 25 12:32 lvmgrups 00000-266345672.vg
```

Backup dosyası Volume Group'la ilgili ayarların bir yedeğidir. LVM üzerindeki metadata verilerinde bir değişiklik olduğunda en son yedekten geri dönülebilir.

```
GNU nano 4.8                               /etc/lvm/backup/testlvm
Generated by LVM2 version 2.03.07(2) (2019-11-30): Thu Feb 25 20:22:58 2021

contents = "Text Format Volume Group"
version = 1

description = "Created *after* executing 'vgcreate testlvm /dev/sdb1 /dev/sdc1 /dev/sdd1'"

creation_host = "client"                  # Linux client 5.4.0-65-generic #73-Ubuntu SMP Mon Jan 18 17:2>
creation_time = 1614284578                # Thu Feb 25 20:22:58 2021

testlvm {
    id = "zAUDAU-5wnf-D18V-P1DR-klhz-68aE-L85uQO"
    seqno = 1
    format = "lvm2"                         # informational
    status = ["RESIZABLE", "READ", "WRITE"]
    flags = []
    extent_size = 8192                      # 4 Megabytes
    max_lv = 0
    max_pv = 0
```

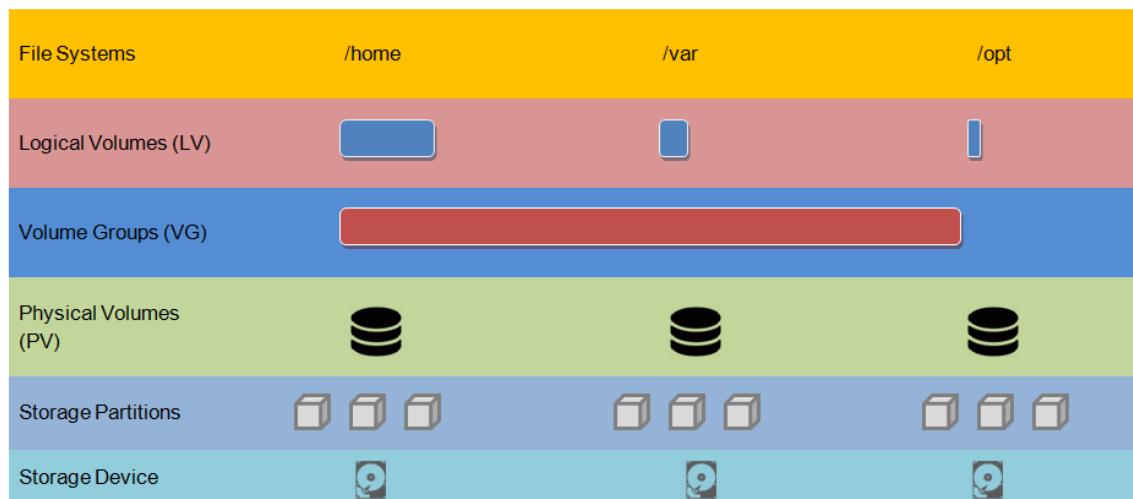
LVM Yapısını Oluşturmak (Creating)

Linux sunucular da ilk kurulum sırasında LVM yapısıyla bütün yapıyı kurabilirsiniz. Eğitimimiz en başında LVM yapısıyla bir sunucu kurulumu yapmıştık. Bu bölümde geleneksel yöntemlerle kurulan bir işletim sisteminde iki disk ekleyerek bir LVM yapısı kuracağız.

Sanal sunucumuza 2 adet disk ekliyoruz. Isblk çıktısı ile sdb ve sdc olarak yeni diskleri görüyoruz. Burada iki yaklaşım mevcut biri tamamen diski fiziksel gruba atamak yada diskin içinden bir bölüm oluşturarak bu bölüm fiziksel grup olarak belirlemek.

Eğer diskinizin tamamını LVM için kullanmayacaksanız diski böülümlere bölüm, böülümleri fiziksel gruba katmak mantıklı olacaktır.

Diskin tamamını kullanacaksanız bölüm oluşturmadan fiziksel gruba diskinizi dahil edebilirsiniz.



Eğer disklerinizi ekledikten sonra sunucuda göremezsiniz partprobe komutuyla disk tablosunu güncelleyin.

```
root@client:/# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
fd0      2:0    1   4K  0 disk 
loop0    7:0    0  55M  1 loop /snap/core18/1880
loop1    7:1    0 55.5M  1 loop /snap/core18/1988
loop2    7:2    0 71.3M  1 loop /snap/lxd/16099
loop3    7:3    0 31.1M  1 loop /snap/snapd/11036
loop4    7:4    0 29.9M  1 loop /snap/snapd/8542
loop5    7:5    0 69.9M  1 loop /snap/lxd/19188
sda      8:0    0  22G  0 disk 
└─sda1   8:1    0   1M  0 part 
  ├─sda2   8:2    0 500M 0 part /boot
  └─sda3   8:3    0 21.5G 0 part /
sdc      8:32   0   3G  0 disk 
sdd      8:48   0   2G  0 disk 
sr0     11:0   1 1024M 0 rom
```

Disk yönetimi bölümünden hatırlayacağınız üzere hemen her iki diskte birer bölüm oluşturuyoruz. Bölümler diskin boyutu kadar olacak. Tek fark bölümün tipini t argümanı ile değiştirip Linux LVM yani 8e olarak değiştiriyoruz.

```
root@client:/# fdisk /dev/sdd

Welcome to fdisk (util-linux 2.34).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Command (m for help): n
Partition type
  p  primary (0 primary, 0 extended, 4 free)
  e  extended (container for logical partitions)
Select (default p):

Using default response p.
Partition number (1-4, default 1):
First sector (2048-4194303, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-4194303, default 4194303):

Created a new partition 1 of type 'Linux' and of size 2 GiB.

Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

İlk olarak sunucuya bağlı olan cihazları ve bölümleri tarıyoruz. Sdb ve sdc disklerini LVM için bağlamıştık. **lvmdiskscan** komut çıktısında da diskleri görüyoruz.

lvmdiskscan

```
root@client:/# lvmdiskscan
/dev/loop0 [      <54.96 MiB]
/dev/loop1 [      55.46 MiB]
/dev/loop2 [      71.27 MiB]
/dev/sda2 [      500.00 MiB]
/dev/loop3 [      <31.09 MiB]
/dev/sda3 [      <21.51 GiB]
/dev/loop4 [      29.88 MiB]
/dev/loop5 [      69.88 MiB]
/dev/sdc1 [      <3.00 GiB]
/dev/sdd1 [      <2.00 GiB]
0 disks
10 partitions
0 LVM physical volume whole disks
0 LVM physical volumes
```

Fiziksel bir grubu yaratmak için pvcreate komutunu kullanıyoruz.

pvcreate /dev/sdc1 /dev/sdd1

```
root@client:/# pvcreate /dev/sdc1 /dev/sdd1
Physical volume "/dev/sdc1" successfully created.
Physical volume "/dev/sdd1" successfully created.
```

Oluşturduğumuz fiziksel grubu doğruluyoruz.

pvs

```
root@client:/# pvs
PV          VG Fmt Attr PSize  PFree
/dev/sdc1    lvm2 ---  <3.00g <3.00g
/dev/sdd1    lvm2 ---  <2.00g <2.00g
```

pvdisplay /dev/sdc1 komutuyla belirttiğimiz physical volume'u görüntüleyebiliyoruz.

Allocatable NO olarak görülür. Çünkü herhangi bir volume group'a bağlamadık. PV UUID değeri ise bu fizikselin bölümün etiketidir.

```
root@client:/# pvdisplay /dev/sdc1
"/dev/sdc1" is a new physical volume of "<3.00 GiB"
--- NEW Physical volume ---
PV Name           /dev/sdc1
VG Name
PV Size          <3.00 GiB
Allocatable       NO
PE Size           0
Total PE          0
Free PE           0
Allocated PE      0
PV UUID           Ly9B46-WRqE-B5ET-BuyI-p48I-xFqr-tzfayZ
```

Fiziksel grubu oluşturuktan sonra Volume Group oluşturuyoruz.

vgcreate lvmgrups /dev/sdc1 /dev/sdd1

```
root@client:/# vgcreate lvmgrups /dev/sdc1 /dev/sdd1
Volume group "lvmgrups" successfully created
```

vgdisplay ile oluşturduğumuz volume grubun detaylarına bakıyoruz. En alttaki PE ifadesi Fiziksel extents ieri işaret eder. Bu volume group üzerinde 1278 adet extents bulunmaktadır. Metadata Areas iki fiziksel disk olması nedeniyle her bir disk için 1 tane tutulduğunda 2

görünür. Metadata Sequence No 1'dir. Çünkü 1 tane volume grup vardır.

```
root@client:/# vgdisplay
--- Volume group ---
VG Name           lvmgrups
System ID
Format            lvm2
Metadata Areas    2
Metadata Sequence No 1
VG Access         read/write
VG Status         resizable
MAX LV
Cur LV
Open LV
Max PV
Cur PV
Act PV
VG Size          4.99 GiB
PE Size           4.00 MiB
Total PE          1278
Alloc PE / Size   0 / 0
Free PE / Size    1278 / 4.99 GiB
VG UUID          f1zU7O-4HxB-TvjZ-1EzL-006Y-o126-xkxDq4
```

pvdisplay /dev/sdc1 komutuyla tekrar fiziksel bölümleri kontrol ettiğimizde allocatable yes olarak görülür. Bu fiziksel bölümde 767 adet PE bulunmaktadır. Allocated PE ise daha herhangi bir volume oluşturmadığımız için 0'dır.

```
root@client:/# pvdisplay /dev/sdc1
--- Physical volume ---
PV Name           /dev/sdc1
VG Name           lvmgrups
PV Size           <3.00 GiB / not usable 3.00 MiB
Allocatable       yes
PE Size           4.00 MiB
Total PE          767
Free PE           767
Allocated PE      0
PV UUID          Ly9B46-WRqE-B5ET-BuyI-p48I-xFqr-tzfayZ
```

Volume grubu doğruluyoruz. Çıktıya dikkat ederseniz İki diskin boyutunun birleşimi kadar alana sahip olduk. Çıktıda PV sütunu fiziksel volume sayısını, LV ise logical bolume sayısını gösterir.

vgs

```
root@client:/# vgs
VG      #PV #LV #SN Attr   VSize  VFree
lvmgrups  2   0   0 wz--n- 4.99g  4.99g
```

Volume gruptan bir Volume oluşturuyoruz. lvmgrup'un içinde 1 GB boyutunda dosyalarım isimli bir volume yaratıyoruz.

lvcreate -L 1G -n dosyalarim lvmgrups

```
root@client:/# lvcreate -L 1G -n dosyalarim lvmgrups
Logical volume "dosyalarim" created.
```

Oluşturduğumuz volume'ü doğruluyoruz.

vgs -o +lv_size,lv_name

```
root@client:/# vgs -o +lv_size,lv_name
VG      #PV #LV #SN Attr   VSize  VFree LSize LV
lvmgrups  2   1   0 wz--n- 4.99g  3.99g 1.00g dosyalarim
```

lvdisplay /dev/lvmgrup/dosyalarim komutuyla oluşturduğumuz logical volume inceleyelim. LE bölümünde 256 ifadesi bu bölümün 256 tane Logical Extents'e sahip olduğunu gösterir.

```
root@client:/# lvdisplay /dev/lvmgrups/dosyalarim
--- Logical volume ---
LV Path          /dev/lvmgrups/dosyalarim
LV Name          dosyalarim
VG Name          lvmgrups
LV UUID          10qDkY-eleu-KUKl-DzB5-q74s-gnm6-Puj93j
LV Write Access  read/write
LV Creation host, time client, 2021-02-25 13:18:07 +0000
LV Status        available
# open           0
LV Size          1.00 GiB
Current LE      256
Segments         1
Allocation       inherit
Read ahead sectors auto
- currently set to 256
Block device    253:1
```

Volume'ü ext4 olarak formatlıyoruz.

mkfs.ext4 /dev/lvmgrups/dosyalarim

```
root@client:/# mkfs.ext4 /dev/lvmgrups/dosyalarim
mke2fs 1.45.5 (07-Jan-2020)
Discarding device blocks: done
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 8d94e0f8-5db6-47a2-b5b4-f3261b5b9f77
Superblock backups stored on blocks:
            32768, 98304, 163840, 229376

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
```

Dosyalarım isimli bir klasör oluşturup volume grupta oluşturduğumuz dosyalarım Volume'ünü bu klasöre bağladık.

```
root@client:/# mount /dev/lvmgrups/dosyalarim /dosyalarim/
```

fstab içerisinde kaydediyoruz. Mkfs komut çıktısındaki UUID numarasını kopyalayıp yapıştırıyoruz. Bu şekilde sunucumuz yeniden başladığında disk alanımız otomatik olarak mount edildiği dizine bağlanacak.

```
GNU nano 4.8                                     /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>   <options>     <dump>   <pass>
# / was on /dev/sda3 during curtin installation
/dev/disk/by-uuid/ee6cca4c-4aa8-4aa0-aa28-fd06a6a8e3f0 / ext4 defaults 0 0
# /boot was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/9cdfaa71-e82c-404e-bc81-322f0a5d2ad8 /boot ext4 defaults 0 0
/swap.img      none      swap    sw      0      0
#/dev/disk/by-uuid/fc059451-6f5c-4c79-8538-25e05cbabf63 /dosyalarim ext4 defaults 0 0
/dev/disk/by-uuid/8d94e0f8-5db6-47a2-b5b4-f3261b5b9f77 /dosyalarim ext4 defaults 0 0
```

df -h çıktılarıyla da yaptığımız işlemleri doğruluyoruz.

```
root@client:~# df -h
Filesystem           Size   Used  Avail Use% Mounted on
udev                 919M    0    919M  0% /dev
tmpfs                193M   1.1M  192M  1% /run
/dev/sda3              22G   5.2G   15G  26% /
tmpfs                962M    0    962M  0% /dev/shm
tmpfs                5.0M    0    5.0M  0% /run/lock
tmpfs                962M    0    962M  0% /sys/fs/cgroup
/dev/sda2              469M  102M   332M  24% /boot
/dev/mapper/lvmgrups-dosyalarim  976M  2.6M   907M  1% /dosyalarim
```

Volume Gruba Disk EklemeK

Volume grubumuzda yer kalmadı ve sunucumuzda fiziksel olduğunu varsayıyalım. İhtiyacımız olan alanı sağlamanın tek yoluda disk artırımı yapmak. Haliyle volume group'ta değişiklik yapmamız gerekecek.

```
root@client:~# lvcreate -L 3.99g -n dosyalarim2 lvmgrups
  Rounding up size to full physical extent 3.99 GiB
  Logical volume "dosyalarim2" created.
root@client:~# vgs
  VG      #PV #LV #SN Attr   VSize  VFree
  lvmgrups   2   2   0 wz--n- 4.99g     0
```

Sunucuya yeni diskimizi takıyoruz. Biz ortamımızda sanal bir disk oluşturup sunucumuza ekliyoruz. sdd olarak disk eklendi.

```
root@client:~# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
fd0        2:0    1   4K  0 disk 
loop0       7:0    0 55.5M 1 loop /snap/core18/1988
loop1       7:1    0 55M  1 loop /snap/core18/1880
loop2       7:2    0 71.3M 1 loop /snap/lxd/16099
loop3       7:3    0 31.1M 1 loop /snap/snapd/11036
loop4       7:4    0 69.9M 1 loop /snap/lxd/19188
loop5       7:5    0 29.9M 1 loop /snap/snapd/8542
sda        8:0    0   22G 0 disk 
└─sda1      8:1    0   1M  0 part 
└─sda2      8:2    0 500M 0 part /boot
└─sda3      8:3    0 21.5G 0 part /
sdb        8:16   0   3G  0 disk 
└─sdb1      8:17   0   3G  0 part 
  ├─lvmgrups-dosyalarim 253:0   0   1G  0 lvm  /dosyalarim
  └─lvmgrups-dosyalarim2 253:1   0   4G  0 lvm  /dosyalarim2
sdc        8:32   0   2G  0 disk 
└─sdc1      8:33   0   2G  0 part 
  └─lvmgrups-dosyalarim2 253:1   0   4G  0 lvm  /dosyalarim2
sdd        8:48   0   1G  0 disk
```

Diskimizde fdisk ile bir bölüm oluşturup Linux LVM (8e) olarak etiketliyoruz.

```
root@client:~# fdisk /dev/sdd

Welcome to fdisk (util-linux 2.34).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x688e99bd.

Command (m for help): n
Partition type
  p   primary (0 primary, 0 extended, 4 free)
  e   extended (container for logical partitions)
Select (default p):

Using default response p.
Partition number (1-4, default 1):
First sector (2048-2097151, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-2097151, default 2097151):
Created a new partition 1 of type 'Linux' and of size 1023 MiB.

Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

pvcreate /dev/sdd1 ile fiziksel volume'u oluşturuyoruz.

```
root@client:~# pvcreate /dev/sdd1
  Physical volume "/dev/sdd1" successfully created.
```

vgextend /dev/lvmgrups /dev/sdd1 komutuyla volume group'a genişletiyoruz.

```
root@client:~# vgextend /dev/lvmgrups /dev/sdd1
  Volume group "lvmgrups" successfully extended
```

vgs komutuyla kontrol ettiğimizde PV'nin 3 olduğunu ve boş alanının disk boyutu kadar arttığını görüyoruz.

```
root@client:~# vgs
  VG      #PV #LV #SN Attr   VSize   VFree
  lvmgrups  3   2   0 wz--n- <5.99g 1020.00m
```

vgdisplay komutuyla da kontrol ettiğimizde fiziksel extentlerin arttığını ve metadata alanını disk sayısı kadar değiştigini görüyoruz.

```
root@client:~# vgdisplay
--- Volume group ---
VG Name          lvmgrups
System ID
Format           lvm2
Metadata Areas   3
Metadata Sequence No 4
VG Access        read/write
VG Status         resizable
MAX LV            0
Cur LV            2
Open LV           2
Max PV            0
Cur PV            3
Act PV            3
VG Size          <5.99 GiB
PE Size           4.00 MiB
Total PE          1533
Alloc PE / Size  1278 / 4.99 GiB
Free  PE / Size  255 / 1020.00 MiB
VG UUID          f1zU7O-4HxB-TvjZ-1EzL-006Y-o126-xkxDq4
```

LVM'i Yeniden Boyutlandırmak (Resizing)

Bu bölümde iki çalışma yapacağız.

Logical Volume boyutunu artırmak (lvextend)

Bir önceki bölümde oluşturduğumuz dosyalarım isimli volume'un boyutunu arttıracıız.

vgs -o +lv_size,lv_name komutuyla kontrol ettiğimizde 4.99GB'lık alanın 3.99 GB'i boş görünüyor. dosyalarım volume'umunun boyutunu 2.5GB yapacağız.

```
root@ubuntu:/# vgs -o +lv_size,lv_name
  VG      #PV #LV #SN Attr   VSize  VFree  LSize  LV
lvmgrup    2   1    0 wz--n- 4.99g  3.99g  1.00g  dosyalarim
```

dosyalarım volume'nün yerini gösterip artırmak istediğimiz miktarı yazarak komutumuzu giriyoruz.

lvextend -L +1.5g /dev/lvmgrup/dosyalarım

```
root@ubuntu:/# lvextend -L +1.5g /dev/lvmgrup/dosyalarim
  Size of logical volume lvmgrup/dosyalarim changed from 1.00 GiB (256 extents) to 2.50 GiB (640 extents).
  Logical volume lvmgrup/dosyalarim successfully resized.
```

resize2fs komutuyla dosya sistemini yeniden boyutlandırıyoruz.

resize2fs /dev/lvmgrup/dosyalarım

```
root@ubuntu:/# resize2fs /dev/lvmgrup/dosyalarim
resize2fs 1.45.5 (07-Jan-2020)
Filesystem at /dev/lvmgrup/dosyalarim is mounted on /dosyalarim; on-line resizing required
old_desc_blocks = 1, new_desc_blocks = 1
The filesystem on /dev/lvmgrup/dosyalarim is now 655360 (4k) blocks long.
```

df -h /dosyalarim/ komutuyla mount edilen dizinin boyutunu kontrol ediyoruz.

```
root@ubuntu:/# df -h /dosyalarim/
Filesystem              Size  Used Avail Use% Mounted on
/dev/mapper/lvmgrup-dosyalarim  2.5G  3.0M  2.4G  1% /dosyalarim
```

Physical Volume'un boyutunu artırmak (pvextend)

LVM yapısıyla kurulmuş bir işletim sisteminin diskini artırıp root bölümünün boyutunu artıracıız.

lsblk komutuyla disk boyutumuzu ve yapımızı görüntüölüyoruz.

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
fd0	2:0	1	4K	0	disk	
loop0	7:0	0	55.5M	1	loop	/snap/core18/1988
loop1	7:1	0	55M	1	loop	/snap/core18/1880
loop2	7:2	0	71.3M	1	loop	/snap/lxd/16099
loop3	7:3	0	69.9M	1	loop	/snap/lxd/19188
loop4	7:4	0	29.9M	1	loop	/snap/snapd/8542
loop5	7:5	0	31.1M	1	loop	/snap/snapd/11036
sda	8:0	0	20G	0	disk	
└─sda1	8:1	0	1M	0	part	
└─sda2	8:2	0	1G	0	part	/boot
└─sda3	8:3	0	19G	0	part	
└─ubuntu--vg-ubuntu--lv	253:0	0	19G	0	lvm	/
sr0	11:0	1	1024M	0	rom	

Diskimizin boyutunu artırıyoruz. sda isimli diskin boyutunu 5 GB artırdık. Volume Group sda3'ün altında yapılanmış durumdadır. Bizde burada işlem yapacağız.

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
fd0	2:0	1	4K	0	disk	
loop0	7:0	0	55.5M	1	loop	/snap/core18/1988
loop1	7:1	0	55M	1	loop	/snap/core18/1880
loop2	7:2	0	31.1M	1	loop	/snap/snapd/11036
loop3	7:3	0	69.9M	1	loop	/snap/lxd/19188
loop4	7:4	0	29.9M	1	loop	/snap/snapd/8542
loop5	7:5	0	71.3M	1	loop	/snap/lxd/16099
sda	8:0	0	25G	0	disk	
└─sda1	8:1	0	1M	0	part	
└─sda2	8:2	0	1G	0	part	/boot
└─sda3	8:3	0	19G	0	part	
└─ubuntu--vg-ubuntu--lv	253:0	0	19G	0	lvm	/
sr0	11:0	1	1024M	0	rom	

İlk olarak sda3'ü fdisk ile genişleteceğiz.

fdisk /dev/sda komutuyla fdisk sihirbazını başlatıyoruz. Ardından p tuşuyla bölümleri listeliyoruz. İkinci satıra dikkat ederseniz sda'nın 25 GB olduğu sektör sayısını gösteriyor. sda3 en son bölümde olduğu için bittiği sector numarasını en büyük değer olan 52428800 'e ayarlayacağız.

```
Command (m for help): p
Disk /dev/sda: 25 GiB, 26843545600 bytes, 52428800 sectors
Disk model: Virtual Disk
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disklabel type: gpt
Disk identifier: 623AEEED6-333D-4A58-BDB5-858EC2254997

Device      Start      End  Sectors Size Type
/dev/sda1    2048     4095     2048   1M BIOS boot
/dev/sda2    4096  2101247  2097152   1G Linux filesystem
/dev/sda3  2101248 41940991 39839744  19G Linux filesystem
```

d tuşu ile bölüm 3 siliyoruz. n ile yeni bir bölüm oluşturuyoruz. Bölüm numarasını giriyoruz. Bölüm 3 yani sda3 ün önceden başladığı sektör numarası ile şimdi ayarlayacağımızın aynı olduğu dikkat edelim. Devamın en büyük sektör numarasını yazarak disk alanının tamamını sda3'e atıyoruz. LVM2 ye ait imza kaldırıyoruz. w ile bölüm tablosuna değişiklikleri uyguluyoruz.

```
Command (m for help): d
Partition number (1-3, default 3):

Partition 3 has been deleted.

Command (m for help): n
Partition number (3-128, default 3):
First sector (2101248-52428766, default 2101248):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2101248-52428766, default 5242876
6):

Created a new partition 3 of type 'Linux filesystem' and of size 24 GiB.
Partition #3 contains a LVM2_member signature.

Do you want to remove the signature? [Y]es/[N]o: N

Command (m for help): w

The partition table has been altered.
Syncing disks.
```

lsblk komutuyla sda3'ün 24GB olduğu görülmüyor.

```
root@ubuntu:~# lsblk
NAME          MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
fd0            2:0    1    4K  0 disk 
loop0           7:0    0 55.5M  1 loop /snap/core18/1988
loop1           7:1    0  55M  1 loop /snap/core18/1880
loop2           7:2    0 31.1M  1 loop /snap/snapd/11036
loop3           7:3    0 69.9M  1 loop /snap/lxd/19188
loop4           7:4    0 29.9M  1 loop /snap/snapd/8542
loop5           7:5    0 71.3M  1 loop /snap/lxd/16099
sda             8:0    0  25G  0 disk 
└─sda1          8:1    0    1M  0 part 
└─sda2          8:2    0    1G  0 part /boot
└─sda3          8:3    0  24G  0 part 
  └─ubuntu--vg-ubuntu--lv 253:0    0 19G  0 lvm  /
sr0            11:0   1 1024M 0 rom
```

Fiziksel Bölümün (PV) boyutunu genişletiyoruz.

pvresize /dev/sda3

```
root@ubuntu:~# pvresize /dev/sda3
  Physical volume "/dev/sda3" changed
  1 physical volume(s) resized or updated / 0 physical volume(s) not resized
```

Fiziksel bölümün boyutunu kontrol ediyoruz.

pvdisplay

```
root@ubuntu:~# pvdisplay
--- Physical volume ---
PV Name              /dev/sda3
VG Name              ubuntu-vg
PV Size              <24.00 GiB / not usable 16.50 KiB
Allocatable          yes
PE Size              4.00 MiB
Total PE             6143
Free PE              1280
Allocated PE         4863
PV UUID              2P3CUX-xbeT-FhIu-Q04z-vkuz-v1zW-hVU1U6
```

pvscan komutuyla boş alanı da görüntülüyoruz.

```
root@ubuntu:~# pvscan
PV /dev/sda3   VG ubuntu-vg      lvm2 [<24.00 GiB / 5.00 GiB free]
Total: 1 [<24.00 GiB] / in use: 1 [<24.00 GiB] / in no VG: 0 [0 ]
```

vgdisplay ile kontrol ettiğimizde Volume Grupta 5 GB'lık boş alanı görüyoruz.

```
root@ubuntu:~# vgdisplay
--- Volume group ---
VG Name              ubuntu-vg
System ID
Format              lvm2
Metadata Areas      1
Metadata Sequence No 3
VG Access           read/write
VG Status            resizable
MAX LV
Cur LV
Open LV
Max PV
Cur PV
Act PV
VG Size              <24.00 GiB
PE Size              4.00 MiB
Total PE             6143
Alloc PE / Size     4863 / <19.00 GiB
Free PE / Size      1280 / 5.00 GiB
VG UUID              CytHYz-AQRa-ZHle-44SI-ORWQ-eFs0-xo1bOG
```

Volume genişletmek için lvextend komutunu kullanıyoruz. Boyut olarak Free Size bölümündeki 1280/5GiB değerinin 1280 olan kısmını girdim. Siz GB cinsinden veya 100%FREE ibaresiyle tüm alanı verip de genişletebilirsiniz. Bu alanda tab tuşıyla ilerleyemeyebilirsiniz. Volume yolunu ile yazmanız gerekebilir.

lvextend -l +1280 /dev/ubuntu-vg/ubuntu-lv

```
root@ubuntu:~# lvextend -l +1280 /dev/ubuntu-vg/ubuntu-lv
  Size of logical volume ubuntu-vg/ubuntu-lv changed from <19.00 GiB (4863 exten
ts) to <24.00 GiB (6143 extents).
  Logical volume ubuntu-vg/ubuntu-lv successfully resized.
```

resize2fs /dev/ubuntu-vg/ubuntu-lv komutuyla dosya sistemini yeniden boyutlandırıyoruz.

```
root@ubuntu:~# resize2fs /dev/ubuntu-vg/ubuntu-lv
resize2fs 1.45.5 (07-Jan-2020)
Filesystem at /dev/ubuntu-vg/ubuntu-lv is mounted on /; on-line resizing required
old_desc_blocks = 3, new_desc_blocks = 3
The filesystem on /dev/ubuntu-vg/ubuntu-lv is now 6290432 (4k) blocks long.
```

df -h ile / (root) bölümümüz boyutuna bakalım.

```
root@ubuntu:~# df -h /
Filesystem           Size  Used Avail Use% Mounted on
/dev/mapper/ubuntu--vg-ubuntu--lv   24G  4.6G  18G  21% /
```

Volume genişletmemiz tamamlanmıştır.

Logical Volume Boyutunun Azaltmak

Belirli bir zaman sonra fark ettinizki çok kullanmadığınız bir bölüme fazla alan vermişsiniz ve diğer bölgelerde de alana ihtiyacınız var. Eğer yapınız LVM ise bu probleme hemen çözüm üretebiliriz. Fazla alan verdığımız bölümün alanını azaltıp önceki konularda yaptığımız gibi istediğimiz lvm bölümünü artıtabiliriz. Bu bölümde bir volume'un boyutunu azaltacağız. Yalnız bu bölüm root olamaz. Çünkü azaltım yapacağımız bölümün unmount yaparak dosya sisteminden ayıracagız. Herşeyden önce bu işlemi yapmadan önce mutlaka verilerinizin bir yedekini farklı bir konuma alın. Disk azaltımı işlemi riskli bir işlemidir ve veri kaybına neden olabilir.

Onceki bölgelerde dosyalarım adından bir volume oluşturup boyutunu da artırılmıştı. Şimdi bu bölümün boyutunu artıralım.

umount /dosyalarım komutuyla volume'u dosya sisteminden ayıriyoruz.

```
root@ubuntu:~# lsblk | grep lvm
└─lvmgrup-dosyalarim 253:0    0  2.5G  0 lvm   /dosyalarim
└─lvmgrup-dosyalarim 253:0    0  2.5G  0 lvm   /dosyalarim
root@ubuntu:~# umount /dosyalarim
root@ubuntu:~# lsblk | grep lvm
└─lvmgrup-dosyalarim 253:0    0  2.5G  0 lvm
└─lvmgrup-dosyalarim 253:0    0  2.5G  0 lvm
```

lvs komutuyla volume boyutunu görüyoruz.

```
root@ubuntu:~# lvs
  LV        VG      Attr          LSize Pool Origin Data%  Meta%  Move Log Cpy%Syn
  c  Convert
  dosyalarim lvmgrup -wi-a----- 2.50g
```

e2fsck -f /dev/lvmgrup/dosyalarim komutuyla dosya sisteminde dosya ve blok sayısının kontrolünü yapıyoruz.

```
root@ubuntu:~# e2fsck -f /dev/lvmgrup/dosyalarim
e2fsck 1.45.5 (07-Jan-2020)
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
/dev/lvmgrup/dosyalarim: 11/163840 files (0.0% non-contiguous), 19252/655360 blocks
```

resize2fs /dev/lvmgrup/dosyalarim 1G komutuyla volume 1GB olacak şekilde dosya sistemine yeniden boyutlandırıyoruz.

```
root@ubuntu:~# resize2fs /dev/lvmgrup/dosyalarim 1G
resize2fs 1.45.5 (07-Jan-2020)
Resizing the filesystem on /dev/lvmgrup/dosyalarim to 262144 (4k) blocks.
The filesystem on /dev/lvmgrup/dosyalarim is now 262144 (4k) blocks long.
```

lvreduce -L -1.5G /dev/lvmgrup/dosyalarim komutuyla bölümü 1.5 GB düşürüyoruz. İçindeki verilerin kaybolabileceği ile ilgili onayımız isteniyor. Elbette bu işlemin bir riski var. Verileriniz silinebilir.

```
root@ubuntu:~# lvreduce -L -1.5G /dev/lvmgrup/dosyalarim
WARNING: Reducing active logical volume to 1.00 GiB.
THIS MAY DESTROY YOUR DATA (filesystem etc.)
Do you really want to reduce lvmgrup/dosyalarim? [y/n]: y
Size of logical volume lvmgrup/dosyalarim changed from 2.50 GiB (640 extents)
to 1.00 GiB (256 extents).
Logical volume lvmgrup/dosyalarim successfully resized.
```

lvdisplay /dev/lvmgrup/dosyalarim komutuyla alanın azaldığını görüyoruz.

```
root@ubuntu:~# lvdisplay /dev/lvmgrup/dosyalarim
--- Logical volume ---
LV Path          /dev/lvmgrup/dosyalarim
LV Name          dosyalarim
VG Name          lvmgrup
LV UUID          weVUog-sfMT-dUan-8f6n-Nmzi-8eKM-d8Y63G
LV Write Access  read/write
LV Creation host, time  ubuntu, 2021-02-14 18:24:53 +0000
LV Status        available
# open           0
LV Size          1.00 GiB
Current LE       256
Segments         1
Allocation       inherit
Read ahead sectors  auto
- currently set to 256
Block device     253:0
```

fstab dosyamıza bu bölüm yazılıydı. Bu nedenle **mount -a** ile dosya sistemine mount ediyoruz. Dosyalarım içindeki verilerimizi kontrol ediyoruz. Hepsi duruyor :)

```
root@ubuntu:~# mount -a
root@ubuntu:~# ls -ltr /dosyalarim/
total 16
drwx----- 2 root root 16384 Feb 14 18:29 lost+found
-rw-r--r-- 1 root root    0 Feb 14 20:36 test5
-rw-r--r-- 1 root root    0 Feb 14 20:36 test4
-rw-r--r-- 1 root root    0 Feb 14 20:36 test3
-rw-r--r-- 1 root root    0 Feb 14 20:36 test2
-rw-r--r-- 1 root root    0 Feb 14 20:36 test1
```

Striped Volume Oluşturmak

Vgs ve pvs komutlarıyla LVM Yapımızı inceliyoruz. Striped volume oluşturmak için 3 diskli bir yapı kurduk. En az iki disk gereklidir. Yapıyı kurma adımları volume grup kurmaya kadar aynıdır. Zaten biz bir volume oluşturacağız. Önceden oluşturduğumuz volume'lerden farklı veriyi sırayla tüm disklere yazması için ayarlayacağız.

```
root@client:~# vgs
  VG      #PV #LV #SN Attr   VSize  VFree
  lvmgrups  3   0    0 wz--n- <8.99g <8.99g
root@client:~# pvs
  PV        VG      Fmt  Attr  PSize  PFree
  /dev/sdb1  lvmgrups lvm2 a--  <3.00g <3.00g
  /dev/sdc1  lvmgrups lvm2 a--  <3.00g <3.00g
  /dev/sdd1  lvmgrups lvm2 a--  <3.00g <3.00g
```

lvcreate -L 3G -i3 -I128 -n volstriped lvmgrups komutunda i3, 3 adet diskimizi ifade ediyor. 3GB'lık alanı 3'e bölerek disklere 1GB olarak yazacak. I128 ise stripe size'ı ifade eder. Bir dosya kaydedildiğinde bölünecek parçanın boyutudur. Bizim örneğimizde 128 KB'tır. Şöyleki 10KB'lık bir dosya kaydettiniz ancak stripe size 128KB olduğu için diskte 128KB yer kaplar. Bu değerin boyutu çalıştığınız veriyle ilgilidir. Çok küçük değerler girerseniz bu sefer disk çok daha küçük parçalara veri yazmaya ve okumaya çalışır. Bu da diskin okuma yazmasını olumsuz etkiler. Diskiniz ve disk kontrollerinin özelliklerine göre bu değer girilmelidir.

Eğer önceden disk içerisinde bir biçimlendirme varsa silinmesi için izin istenir.

```
root@client:~# lvcreate -L 3G -i3 -I128 -n volstriped lvmgrups
WARNING: ext4 signature detected on /dev/lvmgrups/volstriped at offset 1080. Wipe it? [y/n]: y
Wiping ext4 signature on /dev/lvmgrups/volstriped.
Logical volume "volstriped" created.
```

mkfs.ext4 /dev/mapper/lvmgrups-volstriped komutuyla diski formatlıyoruz.

```
root@client:~# mkfs.ext4 /dev/mapper/lvmgrups-volstriped
mke2fs 1.45.5 (07-Jan-2020)
Discarding device blocks: done
Creating filesystem with 786432 4k blocks and 196608 inodes
Filesystem UUID: e1c3b41d-ef67-4cfe-8eab-ca4a2aa0c23b
Superblock backups stored on blocks:
            32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

Dosya sistemimizde bir klasöre mount edelim. Görüleceği üzere 3 bölüme de mount olmuş durumdadır. Bu alana yazmaya başladığında 128KB ilk diske, 2nci 128KB diğer diske, 3ncü 128KB 3'ncü diske ve sonra başa dönerek 1'nci diske yazılacak.

```
root@client:~# mount /dev/mapper/lvmgrups-volstriped /striped/
root@client:~# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
fd0        2:0    1   4K  0 disk
loop0      7:0    0 55.5M 1 loop /snap/core18/1988
loop1      7:1    0 55M  1 loop /snap/core18/1880
loop2      7:2    0 71.3M 1 loop /snap/lxd/16099
loop3      7:3    0 31.1M 1 loop /snap/snapd/11036
loop4      7:4    0 69.9M 1 loop /snap/lxd/19188
loop5      7:5    0 29.9M 1 loop /snap/snapd/8542
sda        8:0    0 22G  0 disk
└─sda1     8:1    0 1M   0 part
└─sda2     8:2    0 500M 0 part /boot
└─sda3     8:3    0 21.5G 0 part /
sdb        8:16   0 3G   0 disk
└─sdb1     8:17   0 3G   0 part
  └─lvmgrups-volstriped 253:0 0 3G   0 lvm  /striped
sdc        8:32   0 3G   0 disk
└─sdc1     8:33   0 3G   0 part
  └─lvmgrups-volstriped 253:0 0 3G   0 lvm  /striped
sdd        8:48   0 3G   0 disk
└─sdd1     8:49   0 3G   0 part
  └─lvmgrups-volstriped 253:0 0 3G   0 lvm  /striped
sr0       11:0   1 1024M 0 rom
```

Mirrored Volume Oluşturmak

Vgs komutuyla baktığımızda önceki konudan her diskimizde 2'ser gb boş olan bulunmaktadır.

```
root@client:~# pvs
PV          VG      Fmt  Attr PSize  PFree
/dev/sdb1   lvmgrups lvm2 a--  <3.00g <2.00g
/dev/sdc1   lvmgrups lvm2 a--  <3.00g <2.00g
/dev/sdd1   lvmgrups lvm2 a--  <3.00g <2.00g
```

lvcreate -L 500M -m1 -n volmirrored lvmgrups komutundaki m1 ile bir tane mirror olacağını belirtiyoruz.

```
root@client:~# lvcreate -L 500M -m1 -n volmirrored lvmgrups
WARNING: ext4 signature detected on /dev/lvmgrups/volmirrored_rmeta_0 at offset 1080. Wipe it? [y/n]: y
Wiping ext4 signature on /dev/lvmgrups/volmirrored_rmeta_0.
Logical volume "volmirrored" created.
```

pvs komutuyla tekrar baktığımızda sdb1 ve sdc1 'den 500M'lik alanın azaldığını görüyoruz.

```
root@client:~# pvs
PV          VG      Fmt  Attr PSize  PFree
/dev/sdb1   lvmgrups lvm2 a--  <3.00g  1.50g
/dev/sdc1   lvmgrups lvm2 a--  <3.00g  1.50g
/dev/sdd1   lvmgrups lvm2 a--  <3.00g <2.00g
```

Yine mkfs ile formatlayıp istediğimiz bir dizine mount edip işlemlerimizi sonlandırıyoruz.

```
sdb                  8:16   0    3G  0 disk
└─sdb1               8:17   0    3G  0 part
  ├─lvmgrups-volstriped     253:1   0    3G  0 lvm  /striped
  ├─lvmgrups-volmirrored_rmeta_0 253:2   0    4M  0 lvm
  │  └─lvmgrups-volmirrored     253:6   0  500M  0 lvm  /mirrored
  ├─lvmgrups-volmirrored_rimage_0 253:3   0  500M  0 lvm
  │  └─lvmgrups-volmirrored     253:6   0  500M  0 lvm  /mirrored
sdc                  8:32   0    3G  0 disk
└─sdc1               8:33   0    3G  0 part
  ├─lvmgrups-volstriped     253:1   0    3G  0 lvm  /striped
  ├─lvmgrups-volmirrored_rmeta_1 253:4   0    4M  0 lvm
  │  └─lvmgrups-volmirrored     253:6   0  500M  0 lvm  /mirrored
  ├─lvmgrups-volmirrored_rimage_1 253:5   0  500M  0 lvm
  │  └─lvmgrups-volmirrored     253:6   0  500M  0 lvm  /mirrored
sdd                  8:48   0    3G  0 disk
└─sdd1               8:49   0    3G  0 part
  └─lvmgrups-volstriped     253:1   0    3G  0 lvm  /striped
```

Remove – Logical Volume ve Volume Group

LVM komponentlerinden birin kaldırılacağı zaman yapanın hiyerarşik olması nedeniyle Logical'dan Physical'a doğru kaldırılır. Yani Physical bir grup kaldırıcasanız o grupta tanımlı bir volume group ve logical volume olmamalıdır.

```
root@client:~# vgs
  VG      #PV #LV #SN Attr   VSize  VFree
  lvmgrups  3   2   0 wz--n- <8.99g 5.00g
root@client:~# lvs
  LV      VG      Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  volmirrored lvmgrups rwi-a-r--- 500.00m
  volstriped  lvmgrups -wi-a----  3.00g
```

Eğer Logical Volume'ler mount edilmiş ise ilk önce unmount edilerek dosya sistemi ile irtibatı kesilmelidir. Sonrasında lvremove volume adı ile kaldırılır.

lvremove /dev/lvmgrups/volmirrored

```
root@client:~# lvremove /dev/lvmgrups/volmirrored
Do you really want to remove and DISCARD active logical volume lvmgrups/volmirrored? [y/n]: y
Logical volume "volmirrored" successfully removed
root@client:~# lvremove /dev/lvmgrups/volstriped
Do you really want to remove and DISCARD active logical volume lvmgrups/volstriped? [y/n]: y
Logical volume "volstriped" successfully removed
```

Devamında volume group kaldırılmalıdır. Eğer gruptan sadece bir physical volume kaldırılacaksa reduce yani azaltma işlemi yapılır.

vgreduce lvmgrups /dev/sdd1

pvs çıktısında görüldüğü üzere sdd1 artık VG'de değil.

```
root@client:~# vgreduce lvmgrups /dev/sdd1
Removed "/dev/sdd1" from volume group "lvmgrups"
root@client:~# vgs
  VG      #PV #LV #SN Attr   VSize  VFree
  lvmgrups    2    0 wz--n-  5.99g  5.99g
root@client:~# pvs
  PV          VG      Fmt  Attr  PSize  PFree
  /dev/sdb1    lvmgrups lvm2 a--  <3.00g <3.00g
  /dev/sdc1    lvmgrups lvm2 a--  <3.00g <3.00g
  /dev/sdd1        lvm2 ---  <3.00g <3.00g
```

Volume Group'u kaldırmak için vgremove komutunu kullanıyoruz. vgs komut çıktısı boş döndü çünkü VG tamamen kaldırıldı.

vgremove lvmgrups

```
root@client:~# vgremove lvmgrups
Volume group "lvmgrups" successfully removed
root@client:~# vgs
```

Recover Failed Disk

LVM yapımız ve özellikleri aşağıdaki gibidir. sdb1 diskinin metadata verisi bir nedenden dolayı silindi.

```
root@client:~# pvs
  PV          VG      Fmt  Attr  PSize  PFree
  /dev/sdb1    testlvm lvm2 a--  <3.00g  2.50g
  /dev/sdc1    testlvm lvm2 a--  <3.00g <2.80g
  /dev/sdd1    testlvm lvm2 a--  <3.00g <3.00g
root@client:~# vgs
  VG      #PV #LV #SN Attr   VSize  VFree
  testlvm    3    2 wz--n-  <8.99g <8.30g
root@client:~# lvs
  LV          VG      Attr   LSize  Pool Origin Data%  Meta%  Move Log Cpy*Sync Convert
  mirrored    testlvm rwi-aor--- 200.00m
               single   -wi-ao---- 300.00m
root@client:~# df -h | grep /dev/mapper/
/dev/mapper/testlvm-mirrored  178M  6.8M  157M  5% /mirror
/dev/mapper/testlvm-single     275M  76M  179M  30% /single
```

dd if = /dev/zero of = /dev/sdb1 bs=1024 count=1 komutuyla sdb1'e boş veri yazarak kullanılmaz hale getirdik.

```
root@client:~# dd if=/dev/zero of=/dev/sdb1 bs=1024 count=1
1+0 records in
1+0 records out
1024 bytes (1.0 kB, 1.0 KiB) copied, 0.000196502 s, 5.2 MB/s
```

pvs komutunu girdiğimde sdb1 için unknown ibaresi var. Unknown olan diskin UUID'si ile durumdan haberdar ediliyoruz.

```
root@client:/# pvs
WARNING: Couldn't find device with uuid McQjdk-3OH7-ktgx-0EHA-OuMP-pQmb-jGodgd.
WARNING: VG testlvm is missing PV McQjdk-3OH7-ktgx-0EHA-OuMP-pQmb-jGodgd (last written to /dev/sdb1).
WARNING: Couldn't find all devices for LV testlvm/single while checking used and assumed devices.
WARNING: Couldn't find all devices for LV testlvm/mirrored_rimage_0 while checking used and assumed devices.
WARNING: Couldn't find all devices for LV testlvm/mirrored_rmeta_0 while checking used and assumed devices.
PV          VG      Fmt  Attr PSize   PFree
/dev/sdc1    testlvm lvm2 a--  <3.00g <2.80g
/dev/sdd1    testlvm lvm2 a--  <3.00g <3.00g
[unknown]   testlvm lvm2 a-m  <3.00g  2.50g
```

İlk olarak dosya sistemine mount olmuş volume'leri umount komutuyla ayıriyoruz.

```
root@client:~# umount /single
root@client:~# umount /mirror
```

Devamında aktif olan volume'leri pasif hale getiriyoruz. a, aktif olanları n ise aktifleri no yap yani deactivate yap anlamındadır.

vgchange -a n --partial

```
root@client:~# vgchange -an --partial
PARTIAL MODE. Incomplete logical volumes will be processed.
WARNING: Couldn't find device with uuid McQjdk-3OH7-ktgx-0EHA-OuMP-pQmb-jGodgd.
WARNING: VG testlvm is missing PV McQjdk-3OH7-ktgx-0EHA-OuMP-pQmb-jGodgd (last written to /dev/sdb1).
WARNING: Couldn't find all devices for LV testlvm/single while checking used and assumed devices.
WARNING: Couldn't find all devices for LV testlvm/mirrored_rimage_0 while checking used and assumed devices.
WARNING: Couldn't find all devices for LV testlvm/mirrored_rmeta_0 while checking used and assumed devices.
0 logical volume(s) in volume group "testlvm" now active
```

Zarar görmüş diskimizin UUID'si ile backupdan bir restore işlemi başlatacağız.

```
pvcreate --uuid "McQjdk-3OH7-ktgx-0EHA-OuMP-pQmb-jGodgd" --restorefile /etc/lvm/backup/testlvm /dev/sdb1
```

```
root@client:~# pvcreate --uuid "McQjdk-3OH7-ktgx-0EHA-OuMP-pQmb-jGodgd" --restorefile /etc/lvm/backup/testlvm /dev/sdb1
WARNING: Couldn't find device with uuid McQjdk-3OH7-ktgx-0EHA-OuMP-pQmb-jGodgd.
WARNING: Couldn't find device with uuid tPjw1V-hE53-BEJU-AMDZ-8Fnz-t7f0-KE4CKp.
WARNING: Couldn't find device with uuid DdKLIU-U2Ps-Gg0t-nLDH-k8p2-lM0T-It9Wsg.
WARNING: Couldn't find device with uuid McQjdk-3OH7-ktgx-0EHA-OuMP-pQmb-jGodgd.
WARNING: VG testlvm is missing PV McQjdk-3OH7-ktgx-0EHA-OuMP-pQmb-jGodgd (last written to /dev/sdb1).
Physical volume "/dev/sdb1" successfully created.
```

Fiziksel bölümü oluşturduk ancak hala volume group hatalı durumu görüyor. Bu nedenle bir restore'da volume group için yapıyoruz.

vgcfgrestore testlvm

```
root@client:~# vgcfgrestore testlvm
Restored volume group testlvm.
```

Ardından logical volume'leri aktif ediyoruz.

```
vgchange -a y testlvm
```

```
root@client:~# vgchange -a y testlvm
 2 logical volume(s) in volume group "testlvm" now active
```

pvs komutuyla kontrolümüzü sağlayalım. Diskimizin unknown durumdan geri döndüğü görülüyor.

```
root@client:~# vgs
  VG      #PV #LV #SN Attr   VSize   VFree
  testlvm  3    2    0 wz--n- <8.99g <8.30g
root@client:~# pvs
  PV          VG      Fmt  Attr PSize   PFree
  /dev/sdb1   testlvm lvm2 a--  <3.00g  2.50g
  /dev/sdc1   testlvm lvm2 a--  <3.00g <2.80g
  /dev/sdd1   testlvm lvm2 a--  <3.00g <3.00g
```

Snapshot Volume

Snapshot Volume, Logical Volume'lerin anlık alınan kopyalarıdır. Bu şekilde snapshot'ı alınan Volume'ın bir kopyası tutulmuş olur. Volume içerisindeki verilerde bir problem olması durumundan snapshotdan geriye dönülebilir. vgs ve lvs komut çıktılarında lvmgrups isimli LVM grubumuz var. Bir adet PV ve bir adet de LV'ye sahiptir. 500MB boyutunda veriler isimli bir tane LV'mız var.

```
root@test:~# vgs
  VG      #PV #LV #SN Attr   VSize   VFree
  lvmgrups  1    1    0 wz--n- <2.00g <1.51g
  ubuntu-vg  1    1    0 wz--n- <9.00g     0
root@test:~# lvs
  LV      VG      Attr       LSize   Pool
  veriler  lvmgrups -wi-ao---- 500.00m
  ubuntu-lv ubuntu-vg -wi-ao---- <9.00g
```

lvcreate -L 100M -s -n veriler_snap /dev/mapper/lvmgrups-veriler komutuyla veriler isimli LV'nin veriler_snap ismiyle bir snapshot'ını alıyoruz. lvs komutuyla kontrol ettiğimizde orijinal LV'si veriler olan bir snapshot LV'nin olduğunu görüyoruz.

```
root@test:~# lvcreate -L 100M -s -n veriler_snap /dev/mapper/lvmgrups-veriler
  Logical volume "veriler_snap" created.
root@test:~# lvs
  LV      VG      Attr       LSize   Pool Origin  Data%  Meta%  Move Log Cpy%Sync Convert
  veriler  lvmgrups  owi-aos--- 500.00m
  veriler_snap lvmgrups  swi-a-s--- 100.00m      veriler 0.01
  ubuntu-lv  ubuntu-vg -wi-ao---- <9.00g
```

Şimdi veriler_snap LV'sini yani snapshot alanımızı genişletip orijinal veriler LV'sine bir dosya ekleyelim.

lvextend -L +200M /dev/mapper/lvmgrups-veriler_snap

```
root@test:~# lvextend -L +200M /dev/mapper/lvmgrups-veriler_snap
  Size of logical volume lvmgrups/veriler_snap changed from 100.00 MiB (25 extents) to 300.00 MiB (75
extents).
  Logical volume lvmgrups/veriler_snap successfully resized.
```

Data sutünündeki değer, orijinal verinin snapshot alanına göre büyümeye oranıdır. Bizim snapshot alanıımız 300MB'dır. Burada değer %9.56 olduğuna göre orijinal LV'de alan 30MB kadar artmış olarak anlayabiliriz. Bu değere göre bir sonraki oluşturulacak snapshot boyutunu ayarlayabiliriz.

```
root@test:~# lvs
  LV        VG      Attr       LSize   Pool Origin  Data%
  veriler    lvmgrups  owi-aos--- 500.00m
  veriler_snap lvmgrups  swi-a-s--- 300.00m          veriler 9.56
```

LVM Snapshot Restore

Direk snapshot LV'sini, sunucu üzerinde oluşturacağınız bir dizine mount edebilirsiniz.

```
mount /dev/mapper/lvmgrups-veriler_snap /snapshot/
```

```
root@test:~# mount /dev/mapper/lvmgrups-veriler_snap /snapshot/
root@test:~# lsblk
NAME           MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0          7:0    0 55.4M  1 loop /snap/core18/1944
loop1          7:1    0 69.9M  1 loop /snap/1xd/19188
loop2          7:2    0 31.1M  1 loop /snap/snapd/10707
loop3          7:3    0 55.5M  1 loop /snap/core18/1988
loop4          7:4    0 32.3M  1 loop /snap/snapd/11107
sda             8:0    0 10G   0 disk 
└─sda1         8:1    0 1M    0 part 
  └─sda2         8:2    0 1G    0 part /boot
  └─sda3         8:3    0 9G    0 part
    └─ubuntu--vg-ubuntu--lv  253:0   0 9G   0 lvm  /
sdb             8:16   0 2G    0 disk 
└─sdb1         8:17   0 2G    0 part
  ├─lvmgrups-veriler-real  253:2   0 500M  0 lvm
  ├─lvmgrups-veriler     253:1   0 500M  0 lvm /dosyalarim
  ├─lvmgrups-veriler_snap 253:4   0 500M  0 lvm /snapshot
  ├─lvmgrups-veriler_snap-cow 253:3   0 300M  0 lvm
  └─lvmgrups-veriler_snap  253:4   0 500M  0 lvm /snapshot
```

Ancak mount ettiğiniz alan snapshot olarak kalacaktır. Eğer orijinal alanınızı tamamen kaybettiyseñiz snapshot'ı convert ederek birleştirmelisiniz. Lvconvert komutu ile --mergesnapshot argümanını kullanıp lvmgrups VG'sindeki, veriler_snap LV'sini orjinali ile birleştiriyoruz.

```
lvconvert --mergesnapshot lvmgrups/veriler_snap
```

```
root@test:~# umount /dosyalarim
root@test:~# lvconvert --mergesnapshot lvmgrups/verilersnap
  Merging of volume lvmgrups/verilersnap started.
  lvmgrups/veriler: Merged: 64.19%
  lvmgrups/veriler: Merged: 100.00%
```

lvs komutuyla kontrol ettiğimizde snapshot'ın kaldırıldığını görüyoruz.

```
root@test:~# lvs
  LV        VG      Attr       LSize   Pool
  veriler    lvmgrups  -wi-a---- 500.00m
  ubuntu-lv  ubuntu-vg -wi-ao--- <9.00g
```

Mount edip ls komutuyla kontrol ettiğimizde snapshotdaki verilerin geri getirildiğini görüyoruz.

```
root@test:~# mount /dev/mapper/lvmgrups-veriler /dosyalarim/
root@test:~# ls -lah /dosyalarim/
total 66M
drwxr-xr-x  3 root root 4.0K Mar  9 19:33 .
drwxr-xr-x 22 root root 4.0K Mar  9 18:40 ..
-rw-r--r--  1 root root 6.6M Mar  9 19:33 data1.txt
-rw-r--r--  1 root root 22M Mar  9 19:33 data2.txt
-rw-r--r--  1 root root 38M Mar  9 19:33 data3.txt
drwx----- 2 root root 16K Mar  9 19:32 lost+found
```

LVM Özellikleri

pvs, vgs ve lvs komut çıktılarındaki Attr sütunun ne anlama geldiğini açıklayalım. Bu alan Volume 'lerin durumları ve çeşitleri hakkında bilgi veriyor.

pvs komutundaki a harfi PV'nin allocatable olduğunu yani atanabilir bir kaynak olduğunu bildiriyor. sdb1, bir VG içinde olmadığı için Attr'de bir harf yok. Ayrıca bir PV'i yanlışlıkla uygun olmayan şekilde silinirse m harfi yani missing olarak işaretlenir.

vgs komutundaki w harfi writeable yani yazılabilir olduğunu gösteriyor. Eğer r harfi varsa read-only yani sadece okunabilir durumdadır. z harfi resizable yani yeniden boyutlandırılabilir. n harfi ise normal anlamındadır ve alanın atanma politikasını belirtir.

lvs komutundaki ilk baştaki – işaretti Volume tipini belirtiyor. Biz Linear bir Volume oluşturduğumuz için – işaretti var. İkinci sıradaki harfler izinleri göstermekle birlikte w harfi writeable yani yazılabilir olduğunu gösteriyor. Üçüncü sıradaki harfler atanma politikalarını gösteriyor. i harfi inherited yani VG'den politikayı miras aldığıını bildiriyor. Beşinci alan Volume'un aktif olduğunu a harfi ile bildirir. Altıncı alan Volume'un açık olduğunu o harfi ile gösterir. Volume'un bir dosya sistemine bağlanmaya hazır olduğu anlamındadır. – işaretti var ise Volume formatlanmamış demektir.

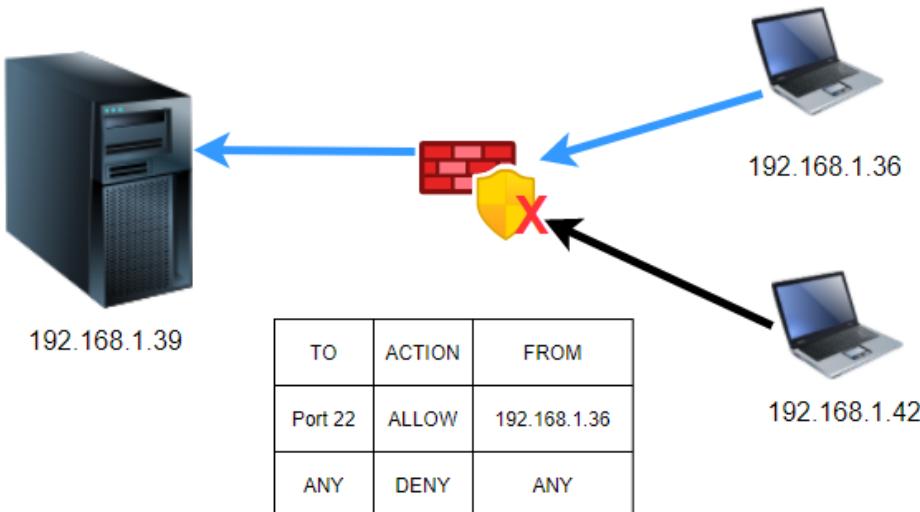
```
ozgur@test:/$ sudo pvs
PV          VG     Fmt Attr PSize  PFree
/dev/sda3   ubuntu-vg lvm2 a-- <9.00g    0
/dev/sdb1   lvmgrups lvm2 --- <2.00g <2.00g
/dev/sdc1   lvmgrups lvm2 a-- <3.00g <1.51g
/dev/sdd1   lvmgrups lvm2 a-- <2.00g <2.00g
ozgur@test:/$ sudo vgs
VG      #PV #LV #SN Attr VSize  VFree
lvmgrups 2   2   0 wz--n- 4.99g  3.50g
ubuntu-vg 1   1   0 wz--n- <9.00g    0
ozgur@test:/$ sudo lvs
LV      VG     Attr LSize  Pool
dosyalarim lvmgrups -wi-ao---- 1.00g
testvolume lvmgrups -wi-a---- 500.00m
ubuntu-lv  ubuntu-vg -wi-ao---- <9.00g
```

17 - Firewall Yapılandırması

Firewall Nedir?

Bir sunucuya gelen veya sunucudan çıkan trafiği sınırlamak için kullanılan bir güvenlik duvarıdır. Firewall, sunucuya gelen veya giden bütün paketleri daha önceden tanımlanmış kurallara ve politikalara göre inceler ve bir pakete izin vererek veya engelleyerek sunucu trafiğini denetler. Herhangi bir ağa bağlı her bilgisayar donanım veya yazılım tabanlı bir güvenlik duvarına sahiptir.

Aşağıdaki resimde 192.168.1.10 sunucusunun firewall'unda sadece 22 portuna 192.168.1.36 IP'sinin erişebilmesi için bir kural var. 192.168.1.36 IP'li bilgisayar sunucunun 22nci porta erişim sağlayabilir. Ancak tablodaki deny kuralı ile 192.168.1.36 IP'sinden gelen isteler hariç tüm istekleri engelleyerek sunucu servislerini güvende tutar.



Ubuntu, varsayılan olarak yüklenen Uncomplicated Firewall (UFW) adlı bir güvenlik duvarı uygulamasına sahiptir. Ufw, bilgisayarınıza erişimle ilgili seçici veya kısıtlayıcı ilkeler uygulamanızı sağlar.

ufw Yapılandırması

ufw, terminalden root yetkisi olan bir kullanıcı ile kullanılır. ufw servisi systemd tarafından kontrol edildiğinden systemctl komut ufw şeklinde servis üzerinde kontrol sağlayabileceğimiz gibi ufw'nin kendine ait komutları vardır.

Komut	Tanım
systemctl status ufw.service	Ufw servisinin durumunu gösterir.
systemctl start ufw.service	Servisi başlatır.
systemctl restart ufw.service	Servisi yeniden başlatır.
systemctl stop ufw.service	Servisi durdurur.
systemctl enable ufw.service	Servisin sistem açılışında başlamasını sağlar.
systemctl disable ufw.service	Servisin sistem açılışında başlamasını durdurur.
ufw reset	Ufw'yi varsayılan döndürür ve devredışı bırakır.
ufw default allow	Ufw gelen giden bütün trafiğe izin verir.
ufw default deny	Ufw gelen trafiği engeller.
ufw show	Ufw ile kullanlanabilecek komutlar gösterilir.
ufw status verbose	Ufw'nin durumunu gösterir.
ufw enable	Ufw'yi aktif hale getirir.
ufw disable	Ufw'yi kapatır.
ufw delete (kural numarası)	Belirtilen numaralı kuralı siler.

```
root@ubuntu:~# systemctl status ufw
● ufw.service - Uncomplicated firewall
   Loaded: loaded (/lib/systemd/system/ufw.service; enabled; vendor preset: enabled)
   Active: active (exited) since Tue 2021-02-16 05:05:06 UTC; 7h ago
     Docs: man:ufw(8)
 Main PID: 381 (code=exited, status=0/SUCCESS)
      Tasks: 0 (limit: 2204)
     Memory: 0B
        CPU: 0
       CGroup: /system.slice/ufw.service

Feb 16 05:05:06 ubuntu systemd[1]: Finished Uncomplicated firewall.
```

```
root@ubuntu:~# ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip
```

ufw yapılandırma dosyaları **/etc/default/ufw** ve **/etc/ufw/sysctl.conf**'dir.

/etc/ufw dizini altındaki kurallar ufw'nin varsayılan ayarlarıdır.

```
root@ubuntu:~# cd /etc/ufw/
root@ubuntu:/etc/ufw# ll
total 56
drwxr-xr-x  3 root root 4096 Feb 16 17:37 .
drwxr-xr-x 96 root root 4096 Feb 16 12:00 ../
-rw-r-----  1 root root  915 Feb 16 13:36 after6.rules
-rw-r-----  1 root root 1126 Jul 31 2020 after.init
-rw-r-----  1 root root 1004 Feb 16 13:36 after.rules
drwxr-xr-x  3 root root 4096 Feb 16 12:00 applications.d/
-rw-r-----  1 root root 6700 Apr  2 2020 before6.rules
-rw-r-----  1 root root 1130 Jul 31 2020 before.init
-rw-r-----  1 root root 2537 Apr  2 2020 before.rules
-rw-r--r--  1 root root 1391 Feb 16 17:21 sysctl.conf
-rw-r--r--  1 root root  313 Feb 16 17:06 ufw.conf
-rw-r-----  1 root root 1385 Feb 16 17:11 user6.rules
-rw-r-----  1 root root 1370 Feb 16 17:42 user.rules
```

Kural Ekleme ve Silme

Kural	Tanımı
ufw allow 22	22 nci porta heryerden erişim izni verir.
ufw deny http	http protokolüne heryerden erişimi engeller.
ufw allow https	https protokolüne heryerden erişime izin verir.
ufw allow 53/udp	53 portuna taşıma protokollerinden TCP ile heryerden erişime izin verir.
ufw allow 5000:5500/tcp	5000 ile 5500 arasındaki portlara taşıma protokollerinden TCP ile heryerden erişime izin verir.
ufw allow from 192.168.1.67	192.168.1.67 IP'sini erişimine izin verir.
ufw deny from 192.168.1.34	192.168.1.67 IP'sini erişimini engeller.
ufw allow from 192.168.1.67 to any port 22	192.168.1.67 IP'sini 22 nci porta erişimine izin verir.
ufw allow from 192.168.1.0/24	192.168.1.0/24 networkündeki bütün IP'lere erişim izni verir.
ufw allow in on eth0 to any port 443	Eth0 interface'de 443 portuna erişime her yerden erişimi izin verir.
ufw delete [kural numarası]	Ufw status komutuyla numarasını öğrendiğin kuralı siler.
ufw delete allow https	https protokolü için verilen izni siler.
ufw deny log 22/tcp	22nci portun TCP protokolünde deny olan logları toplar.
commet 'açıklama'	Komut sonlarına comment argümanı açıklama yapılır.

http için heryerden erişilsin diye kural ekledik.

ufw allow http

```
C:\Users\Fury>curl http://192.168.1.39
curl: (7) Failed to connect to 192.168.1.39 port 80: Timed out
C:\Users\Fury>curl http://192.168.1.39
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
root@ubuntu:~# ufw status
Status: active
root@ubuntu:~# ufw allow http
Rule added
Rule added (v6)
root@ubuntu:~#
```

192.168.1.36 IP'si SSH erişimi deniyor ancak başarısız oluyor. Ufw'ye kural ekleyerek erişime izin veriyoruz.

ufw allow from 192.168.1.36 to any port 22

```
C:\Users\Fury>ssh ozgun@192.168.1.39
ssh: connect to host 192.168.1.39 port 22: Connection timed out

C:\Users\Fury>ssh ozgun@192.168.1.39
ozgun@192.168.1.39's password:

root@ubuntu:~# ufw status numbered
Status: active

To                         Action      From
--                         ----       ---
[ 1] 80/tcp                  ALLOW IN   Anywhere
[ 2] 80/tcp (v6)             ALLOW IN   Anywhere (v6)

root@ubuntu:~# ufw allow from 192.168.1.36 to any port 22
Rule added
```

192.168.1.36 IP'si için verdigimiz izni kaldırırmak istiyoruz.

ufw status numbered #Kuralları listeliyoruz.

ufw delete 2 #2 numaralı kuralı siliyoruz.

```
Last login: Tue Feb 16 17:37:45 2021 from 192.168.1.36
ozgun@ubuntu:~$ exit
logout
Connection to 192.168.1.39 closed.

C:\Users\Fury>ssh ozgun@192.168.1.39
ssh: connect to host 192.168.1.39 port 22: Connection timed out
C:\Users\Fury>

root@ubuntu:~# ufw status numbered
Status: active

To                         Action      From
--                         ----       ---
[ 1] 80/tcp                  ALLOW IN   Anywhere
[ 2] 22                     ALLOW IN   192.168.1.36
[ 3] 80/tcp (v6)             ALLOW IN   Anywhere (v6)

root@ubuntu:~# ufw delete 2
Deleting:
allow from 192.168.1.36 to any port 22
Proceed with operation (y|n)? y
Rule deleted
```

tail -f /var/log/ufw.log komutuyla logları canlı takip ettiğimizde 192.168.1.36 IP'sini blokladığını görüyoruz.

```
Last login: Tue Feb 16 17:37:45 2021 from 192.168.1.36
ozgun@ubuntu:~$ exit
logout
Connection to 192.168.1.39 closed.

C:\Users\Fury>ssh ozgun@192.168.1.39
ssh: connect to host 192.168.1.39 port 22: Connection timed out
0:02:b5:12:44:8a:5b:8b:99:5d:08:00 SRC=192.168.1.36 DST=192.168.1.39 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=62836 DF PROTO=TCP SPT=61963 DPT=22 WINDOW=64240 RES=0x00 SYN URGP=0
Feb 16 17:45:42 ubuntu kernel: [45646.891765] [UFW BLOCK] IN=eth0 OUT= MAC=00:15:5d:02:b5:12:44:8a:5b:8b:99:5d:08:00 SRC=192.168.1.36 DST=192.168.1.39 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=62838 DF PROTO=TCP SPT=61963 DPT=22 WINDOW=64240 RES=0x00 SYN URGP=0
```

Sunucuya ping erişimine kapatmak için **/etc/ufw/sysctl.conf** dosyası içindeki **net/ipv4/icmp_echo_ignore_all=0** satırında 0 değerini 1 yapıp dosyayı kaydediyoruz. Değişikliklerin gerçekleşmesi için systemctl restart ufw ile servisi restart ediyoruz.

```
Pinging 192.168.1.39 with 32 bytes of data:
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Reply from 192.168.1.39: bytes=32 time<1ms TTL=64
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.

GNU nano 4.8          /etc/ufw/sysctl.conf

# Configuration file for setting network variables. Please
# override /etc/sysctl.conf and /etc/sysctl.d. If you prefer
# /etc/sysctl.conf, please adjust IPT_SYSCTL in /etc/default/
# Documentation/networking/ip-sysctl.txt in the kernel source
# information.
#
# Uncomment this to allow this host to route packets between
# net/ipv4/ip_forward=1
#net/ipv6/conf/default/forwarding=1
#net/ipv6/conf/all/forwarding=1

# Disable ICMP redirects. ICMP redirects are rarely used
# for MITM (man-in-the-middle) attacks. Disabling ICMP may cause
# traffic to those sites.
net/ipv4/conf/all/accept_redirects=0
net/ipv4/conf/default/accept_redirects=0
net/ipv6/conf/all/accept_redirects=0
net/ipv6/conf/default/accept_redirects=0

# Ignore bogus ICMP errors
net/ipv4/icmp_echo_ignore_broadcasts=1
net/ipv4/icmp_ignore_bogus_error_responses=1
net/ipv4/icmp_echo_ignore_all=1

# Don't log Martian Packets (impossible addresses)
# packets
```

Ufw logging

Ufw değişik seviyelerde log tutabilir.

Komut	Tanım
ufw logging off	Loglama devredışı kalır.
ufw logging low	Kurallarla eşleşen ve engellenen kuralları loglar.
ufw logging medium	Low seviyeye ek olarak tüm geçersiz paketleri ve yeni bağlantıları loglar.
ufw logging high	Bütün paketleri konfigürasyon dosyasındaki sınırlar dahilinde toplar.
ufw logging full	Herhangi bir sınır olmadan bütün paketleri toplar.

tail -f /var/log/ufw.log komutuyla canlı olarak ufw loglarına bakabilirsiniz.

```
root@ubuntu:~# tail -f /var/log/ufw.log
Feb 16 12:59:59 ubuntu kernel: [28503.804237] [UFW BLOCK] IN=eth0 OUT= MAC=01:00
:5e:00:00:01:60:31:97:3e:b6:50:08:00 SRC=192.168.1.1 DST=224.0.0.1 LEN=36 TOS=0x
00 PREC=0x00 TTL=1 ID=0 DF PROTO=2
```

Loglama seviyesini değiştirmek için ;

ufw logging medium

```
root@ubuntu:~# ufw logging medium
Logging enabled
```

19 - AppArmor Yönetimi

Sunucuya saldırganlar veya yetkisiz erişimlerden korumak için güvenlik duvarları kullanıyoruz. Peki güvenlik duvarını aşan biri normal bir kullanıcı ile sisteminize login olup bütün uygulama ve dosya sistemine erişebilir mi? Cevabı merak ediyorsanız? Evet erişebilir. İşte tam da bu nokta AppArmor, ister kullanıcı olsun ister bir uygulama, erişim sağlamak istenen hedef ile tanımlanmış izinleri karşılaştırır ve yetkisiz bir erişim var ise işlemi durdurur.

Örneğin bir web sitesi doğası gereği tüm dünyaya açıktır. Yazılım ekibinin yazdığı hatalı bir kod, sunucu üzerinde zayıf oluşturabilir veya kullandığınız uygulamanın zayıflıkları de olabilir. Sunucumuzda web hizmetini vermek için nginx kurulu olsun. Saldırgan bir türlü nginx kullanıcısını kullanarak FTP ile sunucuya bağlantı sağlayıp nginx kullanıcısının erişim izni olan dizinlere uygulama kurabilir veya dosyaları silebilir. Ancak AppArmor açık ve yapılandırılmış olsaydı, nginx kullanıcısının normal şartlarda FTP gibi bir yöntemle erişim sağlayamayacağı kurallarda belirli olduğu için saldırının bu girişimi etkisizleştirilecek ve erişimine izin verilmeyecekti

AppArmor Nedir?

AppArmor, zorunlu erişim kontrolü sistemidir. Mandatory Access Control (MAC) olarak bilinir. En iyi bilinen SELinux'un daha basit versiyonudur. AppArmor, önyükleme sırasında çekirdeğe yüklenen profiller sayesinde uygulamaların yalnızca profillerinde belirtildiği kaynaklara erişimini sağlayarak kaynak kullanımını sınırlayan ve ihlalleri loglayarak engelleyen bir uygulamadır.

Sunucu kurulumunu yaptıktan sonra AppArmor varsayılan modda çok az bir profil yapılandırması ile çalışır. **/etc/apparmor.d** dizini altındaki ismi ile anılan uygulamaları kısıtlamak için profilleri barındırır.

Daha fazla profil için **apparmor-profiles** paketini yüklemeniz gereklidir.

```
root@ubuntu:/etc/apparmor.d# ll
total 80
drwxr-xr-x 7 root root 4096 Feb 13 12:41 .
drwxr-xr-x 94 root root 4096 Feb 13 12:45 ../
drwxr-xr-x 4 root root 4096 Jul 31 2020 abstractions/
drwxr-xr-x 2 root root 4096 Jul 31 2020 disable/
drwxr-xr-x 2 root root 4096 Feb 11 2020 force-complain/
drwxr-xr-x 2 root root 4096 Jul 31 2020 local/
-rw-r--r-- 1 root root 1313 May 19 2020 lsb_release
-rw-r--r-- 1 root root 1108 May 19 2020 nvidia_modprobe
-rw-r--r-- 1 root root 3222 Mar 11 2020 sbin.dhclient
drwxr-xr-x 5 root root 4096 Jul 31 2020 tunables/
-rw-r--r-- 1 root root 3202 Feb 25 2020 usr.bin.man
-rw-r--r-- 1 root root 26703 Feb 2 08:21 usr.lib.snapd.snap-confine.real
-rw-r--r-- 1 root root 1575 Feb 11 2020 usr.sbin.rsyslogd
-rw-r--r-- 1 root root 1385 Dec 7 2019 usr.sbin.tcpdump
```

Profil dosyası aslında bir metin dosyasıdır. Örnek dosya tcpdump için olmakla birlikte tcpdump'in çalıştırabileceğii komutlar, erişebileceği dizinler, yönetebileceğii uygulama bilgilerini içerir. Dosya içeriği haricindeki izinlerin dışında uygulamanın başka bir değişiklik yapması AppArmor tarafından engellenir.

```
root@ubuntu:/etc/apparmor.d# cat usr.sbin.tcpdump
# vim:syntax=apparmor
#include <tunables/global>

/usr/sbin/tcpdump {
    #include <abstractions/base>
    #include <abstractions/nameservice>
    #include <abstractions/user-tmp>

    capability net_raw,
    capability setuid,
    capability setgid,
    capability dac_override,
    capability chown,
    network raw,
    network packet,

    # for -D
    @{PROC}/bus/usb/ r,
    @{PROC}/bus/usb/** r,
```

Dizinlerin sonlarındaki r, w,ux gibi ifadeler izinleri ifade eder.

İzin	Tanım
r	read
w	write
ux	unconstrained execute
px	discrete profile execute
ix	inherit execute
m	allow PROT_EXEC with mmap
l	link

Bir uygulama için profil oluşturmayı veya güncellemeyi genprof (generate profile) komutuyla yapıyoruz. Oluşturduğunuz profil dosyasını metin editörüyle düzenleyebilirsiniz.

Komut	Tanım
systemctl status apparmor	Servisi durumu görüntülenir.
systemctl start apparmor	Servisi başlatır.
systemctl stop apparmor	Servisi durdurur.
systemctl restart apparmor	Servisi yeniden başlatır.
apparmor_status	Yükülenen profilleri ve profillere tanımlanmış modları ve process sayısını gösterir.

Sistem yönetici bir uygulamayı iki modda ayarlayabilir.

Mode	Tanımı
Enforce	Kurallar uygulanır, işlem yapılmasına izin verilemez ve syslog'daki tüm ihlaller rapor edilir.
Complain	Sistem herhangi bir kuralı uygulamaz, ancak ihlalleri kaydeder.

AppArmor Yapılandırması

apt install apparmor-profiles apparmor-utils komutuyla profilleri ve profil düzenleme aracını yükliyoruz.

```
root@ubuntu:~# apt install apparmor-profiles
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  apparmor-profiles
0 upgraded, 1 newly installed, 0 to remove and 90 not upgraded.
Need to get 32.7 kB of archives.
After this operation, 358 kB of additional disk space will be used.
Get:1 http://tr.archive.ubuntu.com/ubuntu focal-updates/main amd64 apparmor-profiles all 2.13.3-7ubuntu5.1 [32.7 kB]
Fetched 32.7 kB in 1s (53.0 kB/s)
Selecting previously unselected package apparmor-profiles.
(Reading database ... 71856 files and directories currently installed.)
Preparing to unpack .../apparmor-profiles_2.13.3-7ubuntu5.1_all.deb ...
Unpacking apparmor-profiles (2.13.3-7ubuntu5.1) ...
Setting up apparmor-profiles (2.13.3-7ubuntu5.1) ...
```

apparmor_status komutuyla yüklü profilleri, profillere tanımlanmış modları ve mode durumlarına göre process sayısını görüntüleyiyoruz.

/usr/bin/man satırından örnek verirsek, man komutu için bir profil ve bu profillin enforce modda olduğunu anlıyoruz.

```
root@ubuntu:~# apparmor_status
apparmor module is loaded.
50 profiles are loaded.
31 profiles are in enforce mode.
  /snap/snapd/11036/usr/lib/snapd/snap-confine
  /snap/snapd/11036/usr/lib/snapd/snap-confine//mount-namespace-capture-helper
  /usr/bin/man
  /usr/lib/NetworkManager/nm-dhcp-client.action
  /usr/lib/NetworkManager/nm-dhcp-helper
  /usr/lib/connman/scripts/dhclient-script
  /usr/lib/snapd/snap-confine
  /usr/lib/snapd/snap-confine//mount-namespace-capture-helper
  /usr/sbin/tcpdump
```

chromium_browser için bir profil ve complain modda olduğu görülmüyor.

```
19 profiles are in complain mode.
  /usr/sbin/dnsmasq
  /usr/sbin/dnsmasq//libvirt_leaseshelper
  avahi-daemon
  chromium_browser
  chromium_browser//chromium_browser_sandbox
  chromium_browser//lsb_release
  chromium_browser//xdgsettings
```

Cıktının alındığı an itibariyle belirlenen moddaki profillerde çalışan hizmetler görülür. Örnek olarak man komutunu çalıştırıldığımızda çalışan processler, PID'leri ve modları görülmeyecektir.

```
2 processes have profiles defined.  
2 processes are in enforce mode.  
  /usr/bin/man (2004)  
  /usr/bin/less (2015) /usr/bin/man  
0 processes are in complain mode.  
0 processes are unconfined but have a profile defined.
```

Normal şartlarda AppArmor'un profillerini yapılandırmamanız gerekmektedir ancak geliştirme ortamlarında profillerin modlarını değiştirmeniz gerekebilir.

aa-complain **/usr/bin/man** komutu man için enforce olan modu complain'e çevirir.

```
root@ubuntu:~# aa-complain /usr/bin/man  
Setting /usr/bin/man to complain mode.
```

apparmor_status ile tekrar kontrol ettiğimizde /usr/bin/man complain moda geçmiştir.

```
20 profiles are in complain mode.  
  /usr/bin/man
```

aa-enforce **/usr/bin/man** komutuyla enforce moda geçeriz.

```
root@ubuntu:~# aa-enforce /usr/bin/man  
Setting /usr/bin/man to enforce mode.
```

Profilleri bir metin editörüyle düzenleyebilirsiniz. Profiller **/etc/apparmor.d** dizini altında tutulur.

```

root@ubuntu:/etc/apparmor.d# ls -ltr
total 144
-rw-r--r-- 1 root root 1385 Dec  7  2019 usr.sbin.tcpdump
-rw-r--r-- 1 root root 1575 Feb 11  2020 usr.sbin.rsyslogd
drwxr-xr-x 2 root root 4096 Feb 11  2020 force-complain
-rw-r--r-- 1 root root 3222 Mar 11  2020 sbin.dhclient
-rw-r--r-- 1 root root 1070 May 19  2020 usr.sbin.traceroute
-rw-r--r-- 1 root root  962 May 19  2020 usr.sbin.smbldap-useradd
-rw-r--r-- 1 root root 1925 May 19  2020 usr.sbin.smbd
-rw-r--r-- 1 root root 1358 May 19  2020 usr.sbin.nscd
-rw-r--r-- 1 root root 1019 May 19  2020 usr.sbin.nmbd
-rw-r--r-- 1 root root  990 May 19  2020 usr.sbin.mdnssd
-rw-r--r-- 1 root root 1064 May 19  2020 usr.sbin.identd
-rw-r--r-- 1 root root 4217 May 19  2020 usr.sbin.dnsmasq
-rw-r--r-- 1 root root  949 May 19  2020 usr.sbin.avahi-daemon
-rw-r--r-- 1 root root 10212 May 19  2020 usr.bin.chromium-browser
-rw-r--r-- 1 root root 2035 May 19  2020 sbin.syslog-ng
-rw-r--r-- 1 root root 1306 May 19  2020 sbin.syslogd
-rw-r--r-- 1 root root 1009 May 19  2020 sbin.klogd
-rw-r--r-- 1 root root 1108 May 19  2020 nvidia_modprobe
-rw-r--r-- 1 root root 1313 May 19  2020 lsb_release
-rw-r--r-- 1 root root  858 May 19  2020 bin.ping
drwxr-xr-x 2 root root 4096 Jul 31  2020 disable
drwxr-xr-x 5 root root 4096 Jul 31  2020 tunables
drwxr-xr-x 4 root root 4096 Jul 31  2020 abstractions
-rw-r--r-- 1 root root 26703 Feb  2 08:21 usr.lib.snapd.snap-confine.real
drwxr-xr-x 2 root root 4096 Feb 18 19:04 apache2.d
-rw-r--r-- 1 root root 3202 Feb 18 19:34 usr.bin.man
drwxr-xr-x 2 root root 4096 Feb 18 19:37 local

```

Sunucumuzda Apache kurulu ve biz dosyalarımızın standart konumundan farklı bir yerde tutmak istiyoruz. Bu konumdaki dizin ve dosyalar için yeni bir profil düzenlemek istersek **/etc/apparmor.d/** dizini altında oluşturuyoruz.

nano /etc/apparmor.d/usr.sbin.apache2

```

GNU nano 4.8                                     usr.sbin.apache2
/usr/sbin/apache2

{
/myweb/ r,
/myweb/** rwk,
}

```

```

root@ubuntu:/etc/apparmor.d# ls -ltr | grep apache
drwxr-xr-x 2 root root 4096 Feb 18 20:02 apache2.d
-rw-r--r-- 1 root root   51 Feb 18 20:41 usr.sbin.apache2

```

Kernel'e profili yüklemek için apparmor_parser komutunu kullanıyoruz.

apparmor_parser -r /etc/apparmor.d/usr.sbin.apache2

Değişikliklerden sonra AppArmor'daki profilleri yeniden yükliyoruz.

systemctl reload apparmor

apparmor_status ile profilleri görüntüleyelim. Profil sayısının arttığını ve 3 process için profil tanımlandığını görüyoruz.

```
root@ubuntu:/etc/apparmor.d# apparmor_status
apparmor module is loaded.
51 profiles are loaded.
32 profiles are in enforce mode.
```

```
3 processes are unconfined but have a profile defined.
/usr/sbin/apache2 (779)
/usr/sbin/apache2 (780)
/usr/sbin/apache2 (781)
```

19 - BACKUP ve RESTORE

Backup Nedir?

Backup yani yedekleme sizin veya işletmenizin veri kaybından kaynaklanan uğrayacağınız maddi manevi zararın bir sigortasıdır. Verdiğiniz servisden elde ettiğiniz kar, işlettiğiniz sistemin maliyetinin üzerindeyse verileri yedeklemek gereklidir. Elbette yedeklemenin de bir maliyeti olduğundan bir yedekleme stratejisi belirleyerek ilerlemelidir. Yedeklerin nerede saklanacağı, ne sıklıkla yedekleme işleminin yapılacağı, yedekten ne kadar sürede dönülmesinin bekleniği, yazılım lisansı, donanımınız vb. konular değerlendirilmelidir.

Veri kaybı, donanım arızası, personel hatası, yazılım hatası, donanım driver problemleri, dosya sistemi hataları, hırsızlık, deprem ve yangın gibi doğal afet olayları neticesinde oluşabilir. Bahsettiğimiz durumların hiçbirinde veriniz bir daha geri gelmeyecek şekilde yok olabilir.

Yedekleri güncel, sağlam tutup geri dönme (restore) tatbikatları yaptığınız sürece doğru bir yedekleme stratejisi ile veri kaybından kurtulabilirsiniz.

Yedekleme basit iş gibi görünse de aslında karmaşık bir işledir. Çünkü her sistemin farklı yedekleme çözümlerine ihtiyacı vardır. Uygun çözüm için başlıca belirlenmesi gerekenler;

- Korunacak verinin ne olduğu,
- Veri değişim sıklığı,
- Yedek dosyaların tutulacağı süre,
- Bütçe,

Korunacak veriyi belirleyerek yedeğini alacağınız verinin hacmini anlayabilirsiniz. Veri değişimindeki sıklık yedekleme işlemlerinizin zamanını belirler. Yedek dosyalarının tutulacağı süre hem ne kadar geriye dönük veri tutacağınızı hem de tutulacak yedeklerin boyutunu anlamانıza yardımcı olur. Bütün bunlardan sonra elinizdeki bütçe ile yedekleme stratejinizi dengelemeniz gereklidir.

Yedekleme planı oluştururken bütün yazılımlar yedek alma hızlarının reklamını yaparlar. Ancak önemli olan yedekten ne kadar sürede dönmenizdir. Çünkü yedekler alınırken yedeklere ihtiyacınız yok. Artık sistemlerinize erişemediğinizde yedeklere ihtiyacınız var ve geçen her saniye kayıp demektir. Bu nedenle yedekten dönmenin süresi ve ne kadar önceki veriye dönüleceğinin bilgisi yedekleme stratejisini anahtarlarından biridir.

Yedekleme gerçekten ciddi bir iştir. Alınan yedeklerin monitör edilmesi, doğrulanması, hatalı olanların incelenerek hataların düzeltilmesi, farklı restore senaryolarını çalışarak felaket

anına hazırlıklı olunması çok önemlidir. Çünkü yedeklere ihtiyacınız olduğu gün çalışmaması hem verilerinizin hem de o güne kadar elinizde tuttuğunuz yedeklerin tamamının çöpe gitmesi demektir.

Bir ev kullanıcısı yedeklerini haftada yada ayda bir kere flash belleğe veya DC'ye alırken, orta boyutta bir işletme de daha kritik veriler olabileceği için günlük olarak bir harici diske veya Network Attached Storage (NAS) cihazına yedekleme yapabilir. Daha büyük işletmelerde yedekleme işlemlerini otomatize eden yedekleme yazılımları kullanır. Bu yazılımlar yedekleri Storage Area Network (SAN)'e bağlı storage'lere veya type olarak adlandırılan kasetli sistemlere gönderebilirler. Azure, Amazon gibi bulut tabanlı yedekleme çözümleri de kullanılabilir. Elbette günümüzde bir çok yedekleme çözümü var ancak bu çözümlerin sizin verilerinizin bulunduğu ortama (fiziksel, sanal, bulut) bağlı olarak ihtiyacınızı karşılayıp karşılamayacağını yukarıda bahsettiğimiz kriterlere göre değerlendirmeniz gereklidir.

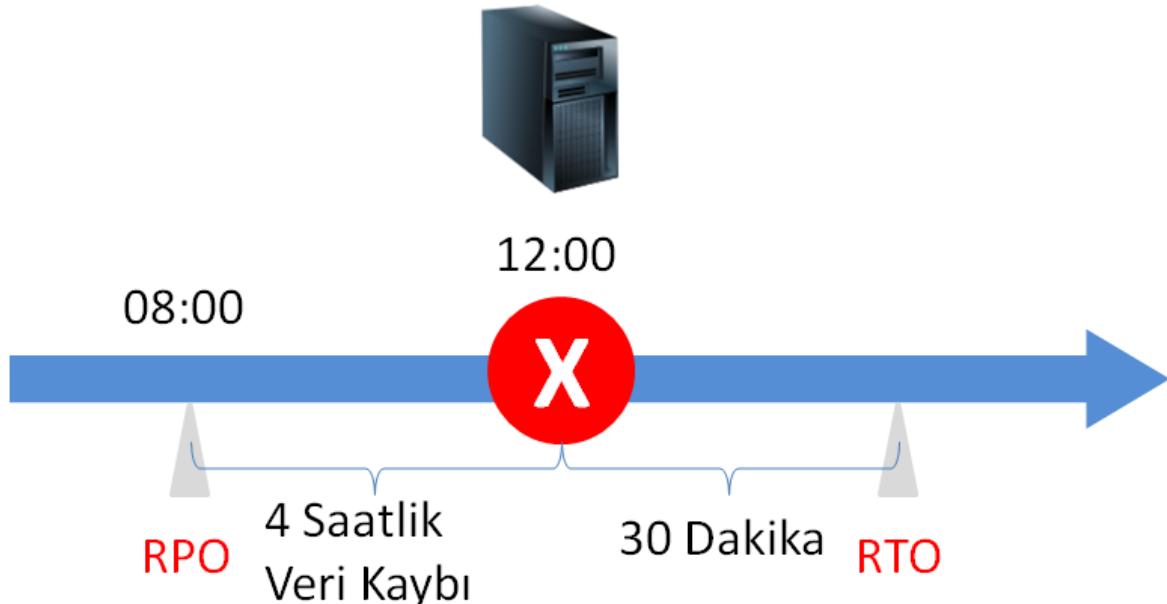
RPO - RTO nedir?

Recovery Point Objective (RPO-Kurtarma Noktası) ve Recovery Time Objective (RTO-Kurtarma Zamanı) kavramları İş sürekliliği ve Felaket Kurtarma (Business Continuity And Disaster Recovery) planını en önemli parametreleridir.

RPO : Zamanda ne kadar geriye gidileceğini belirleyerek en son yedeğin süresini tanımlar. Türkçesi yedeğiniz sabah 08:00'de siz daha mesai başlamadan önce alındı. Öğleye kadar çalıştinız ve yemekten döndükten sonra farkettinizki verileriniz bilgisayar korsanları tarafından şifrelenmiş. Verileri geri almanızında bir yolu yok. Öğlen saat 12:00'den sabah 08:00'a kadar ki 4 saatlik aradaki veri kaybı sizin için ne kadar önemli. Eğer çok önemliyse yedekleme sıklığınızı ve zamanınızı saatlik veya iki saatlik periyodlar olarak düzenlemeniz gereklidir. Bir bankada 1 dakikalık veri kaybı bile felaket olacakken bir tekstil firması için 1 günlük veriyi kaybetmek göze alınabilir. RPO kavramı sizi hayatı döndürecek noktayı belirlemenizi sağlar.

RTO : Verilerinizi kaybettiniz. Ne kadar sürede geri gelmesini bekliyorsunuz? Müşterileriniz hizmet almadan ne kadar süre bu duruma katlanabilir? RTO verilerinizin yükleneceği süreyi belirler. Burada verileriniz boyutu, donanımınız, altyapınız bu süreyi belirleyecek başlıca değişkenlerdir. Çünkü makul bir RTO'unuz yoksa yedekleme işleminiz kısmen başarısız olmuş demektir. Kimi müşteri 2 saatlik hizmet kesintisini sineye çekerken kimisi 15 dakika bile duruma katlanamayabilir.

Elbette mükemmel RPO ve RTO değerlerinin maliyetinin de fazla olacağı ayrı bir handikaptır.



Backup Çeşitleri? – Full, Incremental, Sentez, Differential

Full backup en temel yedekleme yöntemidir. Depolanan alanın tam bir kopyası alınarak farklı bir alanda tutmaktadır. Herhangi bir problemde aldığınız kopyaya geri dönersiniz.

Avantajları :

- Hızlı geri yükleme,
- Veri bütünlüğü,
- Her yedek kendi başına kullanılabilir,

Dezavantajları :

- Yedekleme süresinin uzunluğu,
- Depolama maliyetinin fazla olması,

Büyük verilerin tamamının yedeğini her gün almak maliyetli olacağından bir bütün backup aldıktan sonra kendinden önce alınmış yedeği baz alarak değiştirilmiş dosyaları yedekleme işlemine **incremental backup** diyoruz. Sadece değişimlerin yedeği alındığı için yedekleme süresi ve kapladığı alan full yedeğe göre daha az olacaktır.

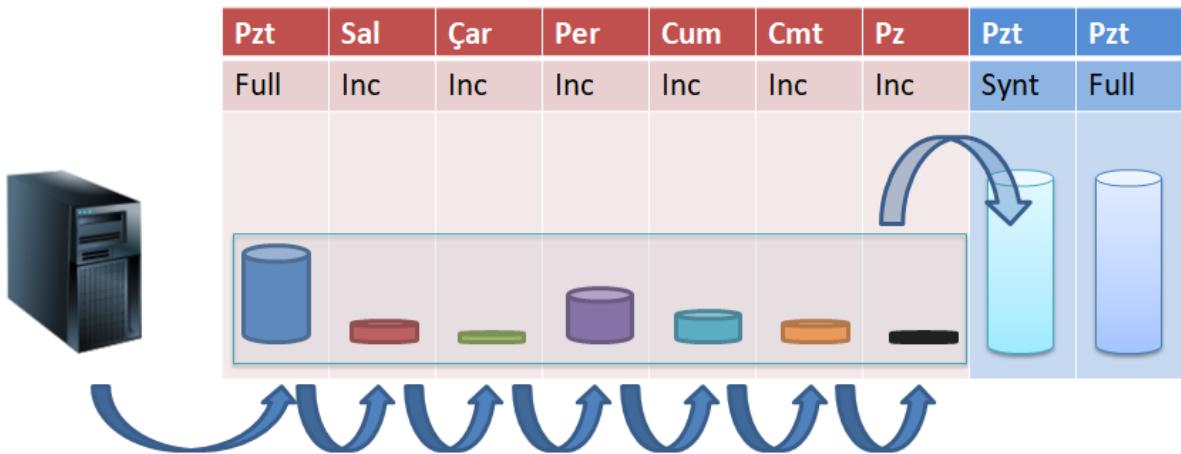
Avantajları :

- Yedekleme süresinin kısaltığı,
- Az alan kaplaması

Dezavantajları :

- Geri yükleme süresinin uzun sürmesi,
- Veri bütünlüğünün sağlanması için kontrol gereklili,

Sentez backup, en son full yedek ile artırımlı yedeklerden gelen verileri bir arşiv dosyasında birleştirir. Bu şekilde Full ve Incremental yedeklemelerin avantajlarından faydalananır.



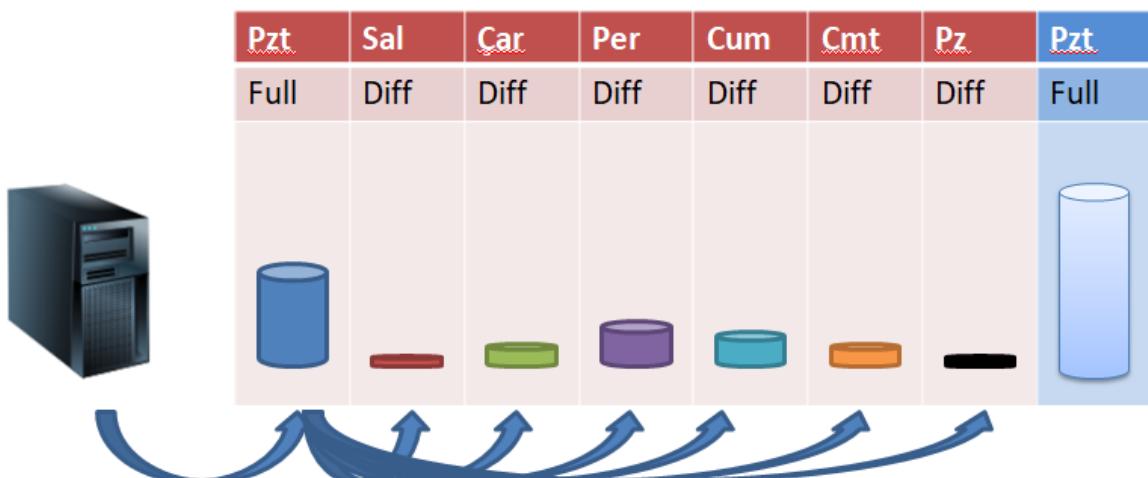
Differential backup, incremental'a benzede differential backup aldığı her fark yedeği için full backup'ı referans alır.

Avantajları :

- Incremental'a göre daha hızlı geri yükleme süresi,

Dezavantajları :

- Incremental'a göre daha fazla depolama alanına ihtiyaç duyar.



tar, scp, dump Kullanımı

tar ile arşivleme

Linux sunucuların tamamında kullanılan ve kullanımı çok kolay olan **tar** arşivleme özelliği sayesinde en temel yedekleme araçlarından bir tanesidir.

Test maksatlı root dizin altında bir dizin ve bu dizin altında iki tane dosya oluşturuyoruz. seq komutuyla dosya içerisinde rasgele veri oluşturduk.

```
mkdir test
```

```
seq 5000000 > /test/testfile1.txt
```

```
seq 15000000 > /test/testfile2.txt
```

```
root@ubuntu:/# mkdir test
root@ubuntu:/# seq 5000000 > /test/testfile1.txt
root@ubuntu:/# seq 15000000 > /test/testfile2.txt

root@ubuntu:#
root@ubuntu:/# ls -lah /test
total 156M
drwxr-xr-x  2 root root 4.0K Feb 19 19:48 .
drwxr-xr-x 22 root root 4.0K Feb 19 19:47 ..
-rw-r--r--  1 root root 38M Feb 19 19:48 testfile1.txt
-rw-r--r--  1 root root 119M Feb 19 19:48 testfile2.txt
```

Yedeğini almak istediğiniz klasör yolunu belirleyip bir arşiv dosyasını oluşturmak için;

tar -cvf /dosyalarim/test.tar test/ komutuyla root altında bulunan test isimli klasörü, ikincil diskin mount olduğu dosyalarım klasörüne arşivledik.

```
root@ubuntu:/# tar -cvf /dosyalarim/test.tar test/
test/
test/testfile1.txt
test/testfile2.txt
```

tar -cvzf /dosyalarim/test2.tar.gz test/ komutunu kullanarak gzip ile arşiv dosyasını sıkıştırmış olduk. -z argümanı bu işlemi yapmamızı sağladı. -z argümanını kullanarak gunzip olarak da dosyamızı arşivleyebiliriz.

```
root@ubuntu:/# tar -cvzf /dosyalarim/test2.tar.gz test/
test/
test/testfile1.txt
test/testfile2.txt
```

tar -cvjf /dosyalarim/test3.tar.bz2 test/ bir önceki komutdaki -z argümanı yerine -j argümanını kullanarak dosyayı daha fazla sıkıştırdık. -J argümanı ile xz, lzip gibi sıkıştırma yöntemleri de kullanılabilir. Ancak pek fazla kullanım yerleri yoktur.

```
root@ubuntu:/# tar -cvjf /dosyalarim/test3.tar.bz2 test/
test/
test/testfile1.txt
test/testfile2.txt
```

Yaptığımız arşiv ve sıkıştırma işlemlerini inceleyecek olursak arşiv dosyası asıl dosya ile aynı boyuttadır. gzip ile dosyaları sıkıştırarak alan tasarrufu sağladık ancak bz2, gzip'den daha iyi sıkıştırma performasına sahip olduğu için veriye çok daha küçük bir boyutta arşivleyip sıkıştırdı. Sıkıştırma yönteminiz ve dosya boyutunuza göre işlem süreniz değişiklik gösterebilir.

```
root@ubuntu:/# ls -lah /dosyalarim/
total 218M
drwxr-xr-x  2 root root 4.0K Feb 19 19:51 .
drwxr-xr-x 22 root root 4.0K Feb 19 19:49 ..
-rw-r--r--  1 root root 42M Feb 19 19:49 test2.tar.gz
-rw-r--r--  1 root root 22M Feb 19 19:51 test3.tar.bz2
-rw-r--r--  1 root root 156M Feb 19 19:49 test.tar
```

Sıkıştırma yöntemleri için direk gzip ve bzip2 komutları da kullanılabilir.

gzip test.tar

bzip2 test.tar

tar ile arşivden çıkarma

Bir arşiv dosyasını dışarı aktarmak için -x argümanı kullanılır. -C argümanı dosyaların belirleyeceğiniz bir dizine çıkarılmasını sağlar.

tar -xvf /dosyalarim/test.tar -C /testyedek/

```
root@ubuntu:/# tar -xvf /dosyalarim/test.tar -C /testyedek/
test/
test/testfile1.txt
test/testfile2.txt
```

Sıkıştırılmış dosyalar için -x argümanı kullanılır.

```
root@ubuntu:/# tar -xvf /dosyalarim/test2.tar.gz -C /testyedek/
test/
test/testfile1.txt
test/testfile2.txt
```

```
root@ubuntu:/# tar -xvf /dosyalarim/test3.tar.bz2 -C /testyedek/
test/
test/testfile1.txt
test/testfile2.txt
```

Arşiv dosyasını açmadan içerisindeki dosyaları görmek için -t argümanı kullanılır.

tar -tvf /dosyalarim/test3.tar.bz2

```
root@ubuntu:/# tar -tvf /dosyalarim/test3.tar.bz2
drwxr-xr-x root/root      0 2021-02-19 19:48 test/
-rw-r--r-- root/root 38888896 2021-02-19 19:48 test/testfile1.txt
-rw-r--r-- root/root 123888897 2021-02-19 19:48 test/testfile2.txt
```

Arşiv içerisindeki belirli bir dosya veya dizini dışarı çıkarmak için;

tar -xvf /dosyalarim/test.tar test/testfile1.txt -C /testyedek/

```
root@ubuntu:/# tar -xvf /dosyalarim/test.tar test/testfile1.txt -C /testyedek/
test/testfile1.txt
```

gzip ile sıkıştırılmış arşiv dosyasından bir dosya veya dizin çıkarmak için -z argümanı kullanılır.

tar -zxvf /dosyalarim/test2.tar.gz test/testfile1.txt -C /testyedek/

```
root@ubuntu:/# tar -zxvf /dosyalarim/test2.tar.gz test/testfile1.txt -C /testyedek/
test/testfile1.txt
```

bz2 ile sıkıştırılmış arşiv dosyasından bir dosya veya dizin çıkarmak için -j argümanı kullanılır.

tar -jxvf /dosyalarim/test3.tar.bz2 test/testfile2.txt -C /testyedek/

```
root@ubuntu:/# tar -jxvf /dosyalarim/test3.tar.bz2 test/testfile2.txt -C /testyedek/  
test/testfile2.txt
```

Birden fazla dosyayı çıkarmak için dosya isimlerini arka arkaya ekleyebilirsiniz.

Belirli bir uzantıya sahip dosyaları çıkarmak için wildcards argümanını kullanabilirsiniz.

```
tar -xvf /dosyalarim/test4.tar --wildcards '*.php' -C /testyedek/
```

```
root@ubuntu:~# tar -xvf /dosyalarim/test4.tar --wildcards '*.*html' -C /testyedek/  
test/index.html  
root@ubuntu:~# tar -xvf /dosyalarim/test4.tar --wildcards '*.*php' -C /testyedek/  
test/index.php  
test/info.php  
test/about.php
```

Arşive Dosya Ekleme

Varolan bir arşiv dosyasına dosya eklemek için -r argümanı kullanılır.

```
tar -rvf /dosyalarim/test.tar /test/deneme.txt
```

```
root@ubuntu:~# tar -rvf /dosyalarim/test.tar /test/deneme.txt  
tar: Removing leading '/' from member names  
/test/deneme.txt  
tar: Removing leading '//' from hard link targets  
root@ubuntu:~# tar -tvf /dosyalarim/test.tar  
drwxr-xr-x root/root 0 2021-02-19 19:48 test/  
-rw-r--r-- root/root 38888896 2021-02-19 19:48 test/testfile1.txt  
-rw-r--r-- root/root 123888897 2021-02-19 19:48 test/testfile2.txt  
-rw-r--r-- root/root 0 2021-02-19 20:44 test/deneme.txt
```

Arşivi Uzak bir Sunucuya Gönderme - scp

Scp komutu ile bu işlemi gerçekleştirmeden önce dosyaları göndereceğimiz sunucudaki dizinin izinlerini gözden geçirmeliyiz. Yedeklerimizi, sunucumuza bağlı ikincil bir diskin mount olduğu dosyalarım klasörüne göndereceğiz. Klasöre dikkat edersek sahibi root'tur. ozgur kullanıcı ile dosya gönderirsek permission denied hatası alarak işlemimiz yarınl kalır. Bu nedenle bu dosyanın izinlerini düzeltceğiz.

```
drwxr-xr-x 2 root root 4.0K Feb 20 13:25 dosyalarim
```

İlk olarak ozgur kullanıcısına root grubuna dahil edeceğiz.

```
usermod -G root ozgur
```

```
root@ubuntu:/# usermod -G root ozgur  
root@ubuntu:/# id ozgur  
uid=1000(ozgur) gid=1000(ozgur) groups=1000(ozgur),0(root)
```

ozgur kullanıcısını root grubuna atadıktan sonra grup kullanıcılarına yazma yetkisi vermemiz gerekiyor.

```
chmod 770 /dosyalarim/
```

```

root@ubuntu:/# chmod 770 /dosyalarim/
root@ubuntu:/# ls -lah
total 2.5G
drwxr-xr-x  22 root root 4.0K Feb 20 13:32 .
drwxr-xr-x  22 root root 4.0K Feb 20 13:32 ..
lrwxrwxrwx  1 root root    7 Jul 31 2020 bin -> usr/bin
drwxr-xr-x  4 root root 4.0K Feb 14 12:40 boot
drwxr-xr-x  2 root root 4.0K Feb 14 12:25 cdrom
drwxr-xr-x 18 root root 4.1K Feb 20 12:36 dev
-rw-r--r--  1 root root 311M Feb 19 21:19 --diff
drwxrwx---  2 root root 4.0K Feb 20 13:25 dosyalarim

```

tar -cvjf test.tar.bz2 /test && scp test.tar.bz2 ozgur@192.168.1.39:/dosyalarim/ komutu iki bölümden oluşuyor. İlk bölümde dosyaları sıkıştırıyor, ikinci bölümde scp komutu dosyaları uzak sunucuya gönderiyoruz.

```

root@ubuntu:/# tar -cvjf test.tar.bz2 /test && scp test.tar.bz2 ozgur@192.168.1.39:/dosyalarim/
tar: Removing leading `/' from member names
/test/
/test/testfile1.txt
/test/testfile2.txt
ozgur@192.168.1.39's password:
test.tar.bz2                                              100% 2437KB  81.3MB/s   00:00
root@ubuntu:/#

```

Bu işlemi bir cronjob görevi belirli periyodlarla yapılmasını sağlayabilirsiniz. Bu işlem için görevleri zamanlama bölümüne gidin.

scp ile karşısındaki sunucuda login olmadan kendi tarafımıza dosya kopyalayabiliriz.

scp ozgur@192.168.1.39:/home/ozgur/scptest.txt .

root@ubuntu:~# scp ozgur@192.168.1.39:/home/ozgur/scptest.txt . ozgur@192.168.1.39's password: scptest.txt root@ubuntu:~#	100% 11 14.0KB/s 00:00	root@client:/home/ozgur# cd root@client:~# ls /home/ozgur/ scptest.txt test.tar.gz root@client:~#
--	------------------------	--

dump Kullanımı

Dosya veya dizinden ziyade depolama aygıtlarını yedeklemek için dump aracı kullanılır. Yani bir diski diğer diske kopyalamak için yapılandırılır. ext2/3/4 dosya sistemindeki dosyaları yedeklenmesi için kullanılır. Dosyaları, güvenli bir şekilde saklama için disk, tape veya usb bellek gibi bir depolama ortamına kopyalar.

Dump aracı varsayılan olarak Ubuntu'da yüklü değildir. apt install dump komutuyla kısa süre içerisinde aracı sunucunuza kurabilirsiniz.

dump aracı 0-9 arası seviyelerden (level) oluşur. Bu seviyeler sayesinde aldığınız ilk full yedekten sonra incremental yedekler alabilirsiniz.

Test ortamımızdaki sunucuda sdc1 ve sdd1 isimli iki diskimiz var. Bu disklere yedekler ve dosyalarım isimli klasörler bağlanmış durumdadır.

sdc	8:32	0	3G	0	disk
└─sdc1	8:33	0	3G	0	part /yedekler
sdd	8:48	0	1G	0	disk
└─sdd1	8:49	0	1023M	0	part /dosyalarim
sr0	11:0	1	1024M	0	rom

Amacımız sdd1'i sdc1'e yedeklemektir. 1 adet full, 5 adet incremental ve son olarak 1 adet differential yedek alacağız.

İlk olarak bir full yedek alacağız. 0 sayısı ilk yedeğimizi yani full yedeğimizi tanımlar. **-u** argümanı alınan dump'ın kaydını **/etc/dumpdates** dosyasına yazar. **f** argümanı yedekleme yapılacak yeri belirtmemizi sağlar.

dump 0uf /yedekler/full /dosyalarim

```
root@ubuntu:/# dump 0uf /yedekler/full /dosyalarim
DUMP: Date of this level 0 dump: Sat Feb 20 14:59:15 2021
DUMP: Dumping /dev/sdd1 (/dosyalarim) to /yedekler/full
DUMP: Label: none
DUMP: Writing 10 Kilobyte records
DUMP: mapping (Pass I) [regular files]
DUMP: mapping (Pass II) [directories]
DUMP: estimated 159347 blocks.
DUMP: Volume 1 started with block 1 at: Sat Feb 20 14:59:18 2021
DUMP: dumping (Pass III) [directories]
DUMP: dumping (Pass IV) [regular files]
DUMP: Closing /yedekler/full
DUMP: Volume 1 completed at: Sat Feb 20 14:59:23 2021
DUMP: Volume 1 159330 blocks (155.60MB)
DUMP: Volume 1 took 0:00:05
DUMP: Volume 1 transfer rate: 31866 kB/s
DUMP: 159330 blocks (155.60MB) on 1 volume(s)
DUMP: finished in 5 seconds, throughput 31866 kBytes/sec
DUMP: Date of this level 0 dump: Sat Feb 20 14:59:15 2021
DUMP: Date this dump completed: Sat Feb 20 14:59:23 2021
DUMP: Average transfer rate: 31866 kB/s
DUMP: DUMP IS DONE
```

Aldığımız yedeği kontrol edelim.

```
root@ubuntu:/# ls -lah /yedekler/full
-rw-r--r-- 1 root root 156M Feb 20 14:59 /yedekler/full
```

dump -W komutuyla aldığımız dump'ların listesi görüntülenir.

```
root@ubuntu:/# dump -W | grep sdd1
/dev/sdd1 (/dosyalarim) Last dump: Level 0, Date Sat Feb 20 14:59:15 2021
```

Incremental yedek için seviyeyi 2'den başlatıyoruz. 1'e differential yedeğimizi alacağız.

dump 2uf /yedekler/inc1 /dosyalarim

```
root@ubuntu:/dosyalarim# dump 2uf /yedekler/inc1 /dosyalarim
DUMP: Date of this level 2 dump: Sat Feb 20 15:40:28 2021
DUMP: Date of last level 0 dump: Sat Feb 20 15:19:48 2021
DUMP: Dumping /dev/sdd1 (/dosyalarim) to /yedekler/inc1
```

Sırayla diğer incrementalları da alıyoruz.

dump 3uf /yedekler/inc2 /dosyalarim

dump 4uf /yedekler/inc3 /dosyalarim

Arada dosyalarım bölümü içerisinde touch komutuyla birkaç dosya ekleyelim.

dump 5uf /yedekler/inc4 /dosyalarim

dump 6uf /yedekler/inc5 /dosyalarim

Incremental yedeğimizi 1nci seviye olarak belirliyoruz. Böylece Full + Incremental'ları kapsayan bir yedek almış olacağız.

dump 1uf /yedekler/diff /dosyalarim

```
root@ubuntu:/dosyalarim# dump 1uf /yedekler/diff /dosyalarim
DUMP: Date of this level 1 dump: Sat Feb 20 15:41:49 2021
DUMP: Date of last level 0 dump: Sat Feb 20 15:19:48 2021
DUMP: Dumping /dev/sdd1 (/dosyalarim) to /yedekler/diff
```

Differential yedek dosyasına dikkat edildiğinde incremental ve full yedekle birlikte alındığı görülüyor. Elbette her gün full da alabilirsiniz. Ancak depolama maliyeti artacaktır.

```
root@ubuntu:# ls -ltr /yedekler/
total 479476
drwx----- 2 root root 16384 Feb 20 14:42 lost+found
-rw-r--r-- 1 root root 163522560 Feb 20 15:19 full
-rw-r--r-- 1 root root 163645440 Feb 20 15:40 inc1
-rw-r--r-- 1 root root 30720 Feb 20 15:40 inc2
-rw-r--r-- 1 root root 30720 Feb 20 15:40 inc3
-rw-r--r-- 1 root root 40960 Feb 20 15:41 inc4
-rw-r--r-- 1 root root 30720 Feb 20 15:41 inc5
-rw-r--r-- 1 root root 163655680 Feb 20 15:41 diff
```

restore -tf /yedekler/full komutuyla yedekleri dönmeden içerisindeki dosyaları görüntüleyebiliriz.

```
root@ubuntu:/dosyalarim# restore -tf /yedekler/full
Dump date: Sat Feb 20 15:19:48 2021
Dumped from: the epoch
Level 0 dump of /dosyalarim on ubuntu:/dev/sdd1
Label: none
      2 .
      11 ./lost+found
      12 ./test
      13 ./test/testfile1.txt
      14 ./test/testfile2.txt
      15 ./test/index.html
      16 ./test/index.php
      17 ./test/about.php
      18 ./test/info.php
      19 ./test/deneme.txt
      20 ./test/denemeler.html
      21 ./test/farkdosyasi.html
      22 ./testfile.txt
      23 ./testfile1.txt
      24 ./testfile2.txt
```

restore komutuyla yedeklerimizden dönebiliyoruz. dosyalarım klasörünün içini varsayıyalım kaza ile sildik. Dosyaları çıkaracağımıza dizine girerek **restore -rf /yedekler/full** komutunu kullanarak dosyalarımızı geri getiriyoruz.

```
root@ubuntu:~# rm -rf /dosyalarim/*
root@ubuntu:~# ls -ltr /dosyalarim/
total 0
root@ubuntu:~# cd /dosyalarim/
root@ubuntu:/dosyalarim# restore -rf /yedekler/full
root@ubuntu:/dosyalarim# ls -ltr
total 484
drwxr-xr-x 2 root root 4096 Feb 19 21:28 test
drwx----- 2 root root 4096 Feb 20 14:49 lost+found
-rw-r--r-- 1 root root 20518 Feb 20 15:04 testfile.txt
-rw-r--r-- 1 root root 20518 Feb 20 15:04 testfile1.txt
-rw-r--r-- 1 root root 314844 Feb 20 15:04 testfile2.txt
-rw----- 1 root root 119224 Feb 20 15:55 restoresymtable
root@ubuntu:/dosyalarim#
```

Differential yedekten dönmek istersek diff isimli yedeği çağrırlıyoruz.

```
root@ubuntu:/dosyalarim# restore -rf /yedekler/diff
root@ubuntu:/dosyalarim# ls -ltr
total 484
drwxr-xr-x 2 root root 4096 Feb 19 21:28 test
drwx----- 2 root root 4096 Feb 20 14:49 lost+found
-rw-r--r-- 1 root root 20518 Feb 20 15:04 testfile.txt
-rw-r--r-- 1 root root 20518 Feb 20 15:04 testfile1.txt
-rw-r--r-- 1 root root 314844 Feb 20 15:04 testfile2.txt
-rw-r--r-- 1 root root 0 Feb 20 15:41 testler7.txt
-rw-r--r-- 1 root root 0 Feb 20 15:41 testler6.txt
-rw-r--r-- 1 root root 0 Feb 20 15:41 testler5.txt
-rw-r--r-- 1 root root 0 Feb 20 15:41 testler4.txt
-rw-r--r-- 1 root root 0 Feb 20 15:41 testler3.txt
-rw-r--r-- 1 root root 0 Feb 20 15:41 testler2.txt
-rw-r--r-- 1 root root 0 Feb 20 15:41 testler1.txt
-rw----- 1 root root 119864 Feb 20 15:59 restoresymtable
root@ubuntu:/dosyalarim#
```

Rsync

Rsync (Remote Sync), dosya ve dizinleri uzak sunucuya veya lokal bir diske kopyalamak ve birbirleriyle senkronize etmek için kullanılır. Bu şekilde dosya ve dizinlerinizin yedekleri farklı bir sunucuya, network storage (NAS) gibi cihazlara gönderebilirsiniz.

Bir dosyayı rsync ile senkronize etmek için;

```
rsync -avh /tmp/dosyalarim/veriler.txt /yedeklerim/
```

komutu kullanılır. -a argümanı dosyaların recursively olarak kopyalanmasına izin verir. -v verbose ile yapılan işlemleri ekrana yazdırıyor. -h ile ekran çıktılarının okunabilir bir şekilde olmasını sağlıyoruz.

```
root@client:~# rsync -avh /tmp/dosyalarim/veriler.txt /yedeklerim/
sending incremental file list
veriler.txt

sent 111 bytes  received 35 bytes  292.00 bytes/sec
total size is 14  speedup is 0.10
```

Bir dizini rsync ile senkronize etmek istersek kaynak olarak dosya yerine dizini gösteriyoruz.

```

root@client:~# rsync -avh /tmp/dosyalarim/ /yedeklerim/
sending incremental file list
./

sent 77 bytes received 19 bytes 192.00 bytes/sec
total size is 14 speedup is 0.15

```

Dizin içerisinde başka dosyalar ekledikten sonra tekrar rsync komutunu çalıştırıldığımızda sadece yeni eklenenler ve değişiklikler senkronize edilir. ls komutıyla verilerimizi senkronize ettiğimiz dosyayı görüntüleyelim. Ardından yenidosya.txt isimli dosyaya bir satır ekleyelim. Tekrar rsync komutunu çalıştırıldığımızda sadece değişiklik yaptığımız dosyanın senkronize olduğunu görüyoruz.

```

root@client:~# ls -ltr /yedeklerim/
total 8
-rw-r--r-- 1 root root 14 Feb 28 21:42 veriler.txt
-rw-r--r-- 1 root root 11 Feb 28 21:43 yenidosya.txt
root@client:~# echo 'Yeni Dosyaya yeni satır' >> /tmp/dosyalarim/yenidosya.txt
root@client:~# rsync -avh /tmp/dosyalarim/ /yedeklerim/
sending incremental file list
yenidosya.txt

sent 181 bytes received 35 bytes 432.00 bytes/sec
total size is 50 speedup is 0.23

```

Uzak bir sunucuya göndermek istersek hedef olarak uzak sunucu adres ve verileri senkronize edeceğimiz dosya yolunun bilgilerini belirtiyoruz.

rsync -avh /tmp/dosyalarim/ ozgur@192.168.1.42:/clientyedekler/

Dilerseniz hedefte bir dosya ismi belirterek doğrulama yaptığınız kullanıcının home dizininde bir dizin oluşturup verileri o dizine de senkronize edebilirsiniz.

```

root@client:~# rsync -avh /tmp/dosyalarim/ ozgur@192.168.1.42:clientyedekler
ozgur@192.168.1.42's password:
sending incremental file list
created directory clientyedekler
./
veriler.txt
yenidosya.txt

sent 241 bytes received 94 bytes 134.00 bytes/sec
total size is 50 speedup is 0.15

root@ubuntu:~# cd /home/ozgur/clientyedekler/
root@ubuntu:/home/ozgur/clientyedekler# ls -lah
total 16K
drwxr-xr-x  2 ozgur root  4.0K Feb 28 21:43 .
drwxr-xr-x. 6 ozgur ozgur 4.0K Feb 28 22:03 ..
-rw-r--r--  1 ozgur root     14 Feb 28 21:42 veriler.txt
-rw-r--r--  1 ozgur root    36 Feb 28 21:46 yenidosya.txt

```

21 - LOGLARI ANLAMAK

İşletim sistemleri üzerinde farklı amaçlarla kullanmak üzere bir çok farklı log dosyası vardır. Bir problem hakkında araştırma yapmak için en doğru hareket ilgili log dosyasına bakmaktadır. Loglar problemin neden ve nereden kaynaklandığını açık bir şekilde ortaya koyacaktır. Linux sistemlerde loglar eğer özel bir yapılandırma yapılmadıysa /var/log dizini altında toplanır. Sisteme yüklediğiniz uygulamalar kendi dizinlerinde logları tutacağı gibi /var/log dizini altına da log dosyası oluşturabilirler. Bazı log dosyaları teknik bilgi içermekle birlikte bir uzman

tarafından incelendiğinde anlaşılırken çoğu log dosyası ne aradığın bilen birisi için yeterince anlaşılırlıdır. Genel olarak bulunan log dosyaları aşağıdaki tabloda verilmiştir.

Log Yolu (Facilities)	Tanımı
/var/log/auth.log	Kullanıcı oturum açma ve kimlik doğrulama işlemleri dahil olmak üzere sistem yetkilendirme bilgilerini içerir.
/var/log/kern.log	Kernel tarafından kaydedilen loglardır.
/var/log/syslog	Sistemin genel log dosyasıdır. Genel problemlerde hatayı aramak için ilk bu dosyadan başlamakta faydalıdır.
/var/log/faillog	Oturum açma limit logları görüntülenir. Ayrıca faillog komutu olarak da kullanılır. <pre>root@ubuntu:~# faillog -a Login Failures Maximum Latest On root 0 0 01/01/70 00:00:00 +0000 daemon 0 0 01/01/70 00:00:00 +0000</pre>
/var/log/lastlog	Sunucuya en son login olanlar hakkında log bilgisi verir. lastlog komutu ile kullanılabilir. <pre>root@ubuntu:~# lastlog Username Port From Latest root * * **Never logged in** daemon * * **Never logged in** bin * * **Never logged in**</pre>
/var/log/ufw.log	Firewalllarındaki loglardır.
/var/log/wtmp	Oturum açan kullanıcılar hakkında bilgileri içerir. Daha önceki konularda kullandığımız who komutunu kullanır.
/var/log/dmesg.log	Kernel'in ön yükleme işlemi sırasında algıladığı donanımlar hakkında bilgi içerir.
/var/log/dpkg.log	Sisteme yüklenen veya kaldırılan paketler hakkında bilgi içerir.
/var/log/deamon.log	Arkaplanda çalışan programların loglarıdır.
/var/log/bootstrap.log	Boot sırasında olaylar hakkında bilgi içerir.

Log Dosyalarını Okumak

Loglar içerikleri metin dosyaları gibi olduğundan cat ve tail komutlarını kullanarak içlerine bakabilirsiniz. Loglardan en iyi şekilde faydalananın için cat ve tail için kullanılan argümanları iyi bilmeli ve aradığınız bilgiyi bulmayı kolaylaştıracak grep, less, more gibi yardımcı komutlardan faydalı olabilirsiniz.

cat /var/log/syslog komutuyla syslog dosyasındaki en son oluşan kaydın olduğu bölüm görüntülenir.

cat /var/log/syslog |less komutuyla en baştan itibaren loglar görüntülenir ve dosya içinde arama yapabilirsiniz.

```
root@ubuntu:~# cat /var/log/syslog |less
Feb 20 12:24:15 ubuntu rsyslogd: [origin software="rsyslogd" swVersion="8.2001.0" x-pid="806" x-info="http://www.rsyslog.com"] rsyslogd was HUPed
Feb 20 12:24:15 ubuntu systemd[1]: logrotate.service: Succeeded.
Feb 20 12:24:15 ubuntu systemd[1]: Finished Rotate log files.
```

cat /var/log/syslog |grep anahtar_kelime komutuyla direkt aradığınız bilgiyi bulmaya yakınlaşırız. Elbette grep ile alınan çıktı uzunsa less komutu ile de birlikte çalıştırılabilir.

cat /var/log/auth.log |less |grep fail

```
root@ubuntu:~# cat /var/log/auth.log |grep fail
Feb 16 11:48:54 ubuntu sshd[2910]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=
ssh ruser= rhost=192.168.1.36 user=ozugur
Feb 20 13:23:16 ubuntu sshd[3677]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=
ssh ruser= rhost=192.168.1.42 user=root
```

tail -f /var/log/apache2/access.log komutu logları takip etmenin en çok kullanılan yoludur. tail komutu ile loglar canlı olarak takip edilir. Ctrl+C ile bash satırına dönülür.

```
root@ubuntu:~# tail -f /var/log/apache2/access.log
192.168.1.37 - - [20/Feb/2021:21:58:49 +0000] "GET / HTTP/1.1" 200 3477 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.182 Safari/537.36"
192.168.1.37 - - [20/Feb/2021:21:58:49 +0000] "GET /icons/ubuntu-logo.png HTTP/1.1" 200 3623 "http://192.168.1.39/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.182 Safari/537.36"
192.168.1.37 - - [20/Feb/2021:21:58:49 +0000] "GET /favicon.ico HTTP/1.1" 404 490 "http://192.168.1.39/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.182 Safari/537.36"
```

Logları Anlamak

Bir sistem yöneticisinin logları takip etmesi en önemli konulardan bir tanesidir. Loglara bakarak olası bir donanım arızasını önceden görüp önlem alabilir yada yetkisiz kişilerin sürekli sisteminize girmeye çalıştığını görerek sunucuda ve networkde güvenlik sıkıştırmaları yapabilirsiniz.

Her log dosyası farklı bilgi satır ve sutunları içerir. Kimisi belirli bir formatta çıktı verirken kimisi daha karmaşık çıktılar verebilir.

syslog ve auth.log loglarını inceleyelim.

cat /var/log/syslog çıktısından 1 satır 4 bölümünden oluşur. İlk bölüm zaman, ikinci satır sunucunun ismi, üçüncü bölüm logun gerçekleştiği servis ve son bölüm olan olaydır.

İlk satırında kernel tarafından bir log düşürülmüş. Olayı incelediğimizde UFW, eth0 interface'ne 192.168.1.37 IP'li tarafından 22nci porta yapılan isteği engellemiş.

İkinci satırda, systemd tarafından bir log düşürülmüş. UID'si 1000 olan bir kullanıcı oluşturulduğu bildiriliyor.

Üçüncü satırda ise yine systemd tarafından UID'si 1000 olan kullanıcı bir directory çalışması başlatıldığını bildirmiştir.

```
Feb 20 12:25:56 ubuntu kernel: [ 125.768976] [UFW BLOCK] IN=eth0 OUT= MAC=00:15:5d:02:b5:12:44:8a:5b:8b:99:5d:08:00 SRC=192.168.1.37 DST=192.168.1.39 LEN=52 TOS=0x00 PREC=0x00 TTL=128 ID=2382 DF PROTO=TCP SPT=55392 DPT=22 WINDOW=64240 RES=0x00 SYN URGP=0
Feb 20 12:26:28 ubuntu systemd[1]: Created slice User Slice of UID 1000.
Feb 20 12:26:28 ubuntu systemd[1]: Starting User Runtime Directory /run/user/1000...
```

tail -f /var/log/auth.log çıktısı da syslog gibi 4 bölümünden oluşuyor.

İlk satır 192.168.1.37 IP'sinden fake_user isimli birinin bağlanmaya çalıştığını bildiriyor.

İkinci satırda fake_user bir parola girmiştir ve sunucuya göndermiş sunucuda böyle bir kullanıcı yok diye bildirimde bulunuyor.

Üçüncü satırda 192.168.1.37 IP'sinden yapılan başarısız kimlik doğrulama için log tutuluyor.

Son satırda ise olayın tamamının kaydı oluşturuluyor.

```
Feb 20 22:23:44 ubuntu sshd[1460]: Invalid user fake_user from 192.168.1.37 port 64019
Feb 20 22:23:47 ubuntu sshd[1460]: pam_unix(sshd:auth): check pass; user unknown
Feb 20 22:23:47 ubuntu sshd[1460]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.1.37
Feb 20 22:23:49 ubuntu sshd[1460]: Failed password for invalid user fake_user from 192.168.1.37 port 64019
ssh2
```

journald ve journalctl

journald gelişmiş bir log yönetim sistemidir. Kernel, boot işlemleri ve servislerden logları toplar ve binary olarak /run/log/journal altında saklar. journald'nin topladığı loglara journalctl

komutu ile sorgulanabilir. Journald logları kalıcı değildir. Sistem yeniden başlatırken logları rsyslogd'ye aktarır. rsyslogd de logları ilgili log dosyalarına dağıtır.

journald, logları belirlenen dosyalara yazabilir, uzaktan loglar yönetilebilir ve log sunucularına log gönderimi sağlayabilir.

journalctl komutuyla sunucunun en son başladığı zamandan itibaren loglar görüntülenir. journalctl çıktısı less komutu yazılmış gibi ilk kayıttan itibaren bir ekran çıktısı verir ve kalveye tuşları içinde gezinme ve / ve ? işaretiley birlikte kullanacağınız anahtar kelime ile arama yapılabilir.

```
Feb 14 13:33:10 ubuntu kernel: e820: update [mem 0x00000000-0x0000ffff] usable ==> reserved
Feb 14 13:33:10 ubuntu kernel: e820: remove [mem 0x000a0000-0x000fffff] usable
Feb 14 13:33:10 ubuntu kernel: AGP: No AGP bridge found
Feb 14 13:33:10 ubuntu kernel: last_pfn = 0x7fff0 max_arch_pfn = 0x400000000
Feb 14 13:33:10 ubuntu kernel: MTRR default type: uncachable
Feb 14 13:33:10 ubuntu kernel: MTRR fixed ranges enabled:
Feb 14 13:33:10 ubuntu kernel: 00000-FFFFF uncachable
Feb 14 13:33:10 ubuntu kernel: MTRR variable ranges enabled:
lines 1-41
```

-f argümanı en log satırlarını gösterir.

```
root@ubuntu:/run/log/journal# journalctl -f
-- Logs begin at Sun 2021-02-14 13:33:10 UTC. --
Feb 21 12:17:01 ubuntu CRON[2603]: pam_unix(cron:session): session closed for user root
Feb 21 12:19:27 ubuntu systemd[1]: Starting Message of the Day...
Feb 21 12:19:28 ubuntu 50-motd-news[2649]: * Introducing self-healing high availability clusters in
MicroK8s.
Feb 21 12:19:28 ubuntu 50-motd-news[2649]: Simple, hardened, Kubernetes for production, from Rasp
berryPi to DC.
Feb 21 12:19:28 ubuntu 50-motd-news[2649]: https://microk8s.io/high-availability
Feb 21 12:19:28 ubuntu systemd[1]: motd-news.service: Succeeded.
Feb 21 12:19:28 ubuntu systemd[1]: Finished Message of the Day.
```

journalctl yazıp iki kere tab tuşuna bastığınızda filtreleme seçenekleri görüntülenir.

```
root@ubuntu:/run/log/journal# journalctl
_AUDIT_FIELD_APPARMOR= _DISK_KEEP_FREE= _SEAT_ID=
_AUDIT_FIELD_CAPABILITY= _DISK_KEEP_FREE_PRETTY= _SELINUX_CONTEXT=
_AUDIT_FIELD_CAPNAME= _EXE= _SESSION_ID=
_AUDIT_FIELD_INFO= _EXIT_CODE= _SOURCE_MONOTONIC_TIMESTAMP=
_AUDIT_FIELD_NAME= _EXIT_STATUS= _SOURCE_REALTIME_TIMESTAMP=
_AUDIT_FIELD_OPERATION= _GID= _STREAM_ID=
_AUDIT_FIELD_PROFILE= _HOSTNAME= _SYSLOG_FACILITY=
_AUDIT_ID= _INTERFACE= _SYSLOG_IDENTIFIER=
_AUDIT_LOGINUID= _INVOCATION_ID= _SYSLOG_PID=
_AUDIT_SESSION= _JOB_ID= _SYSLOG_RAW=
_AUDIT_TYPE= _JOB_RESULT= _SYSLOG_TIMESTAMP=
```

journalctl _UID=1000 -n 10 komutuyla UID'si 1000 olan kullanıcının, -n ile, ilk 10 kaydını ekrana getiriyoruz.

Ne aradığınızı biliyorsanız parametrelere göre aramak aradığınız şeyi bulmanızda çok yardımcı olacaktır.

```
root@ubuntu:/run/log/journal# journalctl _UID=1000 -n 10
-- Logs begin at Sun 2021-02-14 13:33:10 UTC, end at Sun 2021-02-21 12:25:01 UTC. --
Feb 21 09:29:38 ubuntu systemd[1186]: Listening on GnuPG cryptographic agent and passphrase cache.
Feb 21 09:29:38 ubuntu systemd[1186]: Listening on debconf communication socket.
Feb 21 09:29:38 ubuntu systemd[1186]: Listening on REST API socket for snapd user session agent.
Feb 21 09:29:38 ubuntu systemd[1186]: Listening on D-Bus User Message Bus Socket.
Feb 21 09:29:38 ubuntu systemd[1186]: Reached target Sockets.
Feb 21 09:29:38 ubuntu systemd[1186]: Reached target Basic System.
Feb 21 09:29:38 ubuntu systemd[1186]: Reached target Main User Target.
Feb 21 09:29:38 ubuntu systemd[1186]: Startup finished in 170ms.
Feb 21 09:29:40 ubuntu sudo[1331]: pam_unix(sudo:auth): Couldn't open /etc/securetty: No such file
Feb 21 09:29:41 ubuntu sudo[1331]: pam_unix(sudo:auth): Couldn't open /etc/securetty: No such file
```

journalctl -p err, warning, info vb. log seviyelerini belirttiğiniz komutla direk belirttiğiniz log seviyesine göre loglar gösterilecektir.

journalctl -p info -n 10

```
root@ubuntu:/run/log/journal# journalctl -p info -n 10
-- Logs begin at Sun 2021-02-14 13:33:10 UTC, end at Sun 2021-02-21 12:35:01 UTC. --
Feb 21 12:19:28 ubuntu 50-motd-news[2649]:      Simple, hardened, Kubernetes for production, from Ras...
Feb 21 12:19:28 ubuntu 50-motd-news[2649]:      https://microk8s.io/high-availability
Feb 21 12:19:28 ubuntu systemd[1]: motd-news.service: Succeeded.
Feb 21 12:19:28 ubuntu systemd[1]: Finished Message of the Day.
Feb 21 12:25:01 ubuntu CRON[2672]: pam_unix(cron:session): session opened for user root by (uid=0)
Feb 21 12:25:01 ubuntu CRON[2673]: (root) CMD (command -v debian-sal > /dev/null && debian-sal 1 1)
```

journalctl ile --since ve --until argümanlarını da kullanabilirsiniz.

journalctl --until 15:00:00 -n 10

```
root@ubuntu:/run/log/journal# journalctl --until 15:00:00 -n 10
-- Logs begin at Sun 2021-02-14 13:33:10 UTC, end at Sun 2021-02-21 12:35:01 UTC. --
Feb 21 12:19:28 ubuntu 50-motd-news[2649]:      Simple, hardened, Kubernetes for production, from Ras...
Feb 21 12:19:28 ubuntu 50-motd-news[2649]:      https://microk8s.io/high-availability
Feb 21 12:19:28 ubuntu systemd[1]: motd-news.service: Succeeded.
Feb 21 12:19:28 ubuntu systemd[1]: Finished Message of the Day.
Feb 21 12:25:01 ubuntu CRON[2672]: pam_unix(cron:session): session opened for user root by (uid=0)
```

journalctl --since 11:00:00 -n 10 -p info

```
root@ubuntu:/run/log/journal# journalctl --since 11:00:00 -n 10 -p info
-- Logs begin at Sun 2021-02-14 13:33:10 UTC, end at Sun 2021-02-21 12:45:01 UTC. --
Feb 21 11:05:01 ubuntu CRON[2058]: pam_unix(cron:session): session opened for user root by (uid=0)
Feb 21 11:05:01 ubuntu CRON[2059]: (root) CMD (command -v debian-sal > /dev/null && debian-sal 1 1)
Feb 21 11:05:01 ubuntu CRON[2058]: pam_unix(cron:session): session closed for user root
Feb 21 11:15:01 ubuntu CRON[2216]: pam_unix(cron:session): session opened for user root by (uid=0)
Feb 21 11:15:01 ubuntu CRON[2217]: (root) CMD (command -v debian-sal > /dev/null && debian-sal 1 1)
Feb 21 11:15:01 ubuntu CRON[2216]: pam_unix(cron:session): session closed for user root
Feb 21 11:17:01 ubuntu CRON[2220]: pam_unix(cron:session): session opened for user root by (uid=0)
```

journalctl -o verbose komutuyla loglar, parametrelerle birlikte PID, ID, bağlı olan kullanıcı ve gruplar, mesajlar gibi bir çok ayrıntıyla görüntülenir.

Rsyslogd

rsyslogd, log mesajlarını yönetmek için kullanılan yardımcı bir uygulamadır. Hem lokal hem de uzaktan log toplama sistemleri ile entegre olunması için destek sağlar. rsyslogd'yi yapılandırmak için /etc/rsyslog.conf ve /etc/rsyslogd altındaki dosyalar kullanılır.

Rsyslog alınan her log mesajı için, bu mesajın nasıl işleneceğini belirlemek üzere /etc/rsyslog.conf yapılandırma dosyasına bakar. Yapılandırma dosyası içindeki logla eşleşen kuralı bulur ve kuraldaki ifadeye göre logu işler. Logla eşleşen bir kural yoksa log ıskartaya çıkarılır. rsyslog.conf içerisinde modules, global directives ve rules bölümleri vardır. Uzaktan log sistemleri entegre olunabilmesi için port bilgilerinin bulunduğu satırlar önemlidir. Varsayılan olarak 514 portu rsyslog'un logları aldığı ve gönderdiği porttur. Log toplama sisteminize göre port bilgileri değişiklik gösterebilir. Modüller logların işlenmesinde esneklik sağlayarak rsyslog'u daha fonksiyonel bir hale getirir.

```
#####
#### MODULES #####
#####

module(load="imuxsock") # provides support for local system logging
#module(load="immark")  # provides --MARK-- message capability

# provides UDP syslog reception
#module(load="imudp")
#input(type="imudp" port="514")

# provides TCP syslog reception
#module(load="imtcp")
#input(type="imtcp" port="514")

# provides kernel logging support and enable non-kernel klog messages
module(load="imklog" permitnonkernelfacility="on")
```

/etc/rsyslog.d/50-default.conf içinde rsyslog'un log toplayacağı kaynaklar ve dosyalarının nerede bulacağını belirleyen konfigürasyon dosyasıdır. Sol taraf selector diye tabir edilen hangi logun alınacağı sağ taraf ise logun nereye gönderileceği veya ne yapılacağını belirleyen kurallar vardır. Dosya yollarının başındaki – işaretçi logun hemen kaydedilmeden bellekte durmasını sağlayarak loglara erişimi hızlandırır. – işaretçi koymadığınız dosyalara loglar direk yazılır.

İlk satırda auth ve authpriv loglarının tüm seviyelerdeki mesajlarını /var/log/auth.log altında bulunacağını belirtir. * işaretçi tüm log seviyelerini tanımlar.

İkinci satırda noktalı virgül ile ayrılmış iki selektör var. *.* tüm logları ifade ederken auth,authpriv.none ifadesi auth,authpriv loglarını hariç tutulmasını sağlıyor. Yani auth, authpriv logları hariç tüm logları syslog içerisinde tanımlıdır anlamına geliyor.

Log'lar farklı öncelik değerlerine göre de kayıt altına alınabilir. Bir öncelik kullanıldığında altında bulunan öncelikler de dahil tüm loglarda kayıt edilir. Örneğin debug en düşük öncelik seviyesindedir. Eğer debug olarak seçim yaparsanız debug'dan başlayıp emergency'e kadar tüm loglar kayıt altına alınacaktır. Eğer warning seçiminde bulunursanız emergency'e kadar loglar toplanır.

```

# Some "catch-all" log files.

#*.=debug;\ 
#      auth,authpriv.none;\ 
#      news.none;mail.none      -/var/log/debug
#*.=info;*.=notice;*.=warn;\ 
#      auth,authpriv.none;\ 
#      cron,daemon.none;\ 
#      mail,news.none          -/var/log/messages

#
# Emergencies are sent to everybody logged in.
#
*.emerg                           :omusrmsg:*

#
# I like to have messages displayed on the console, but only on a virtual
# console I usually leave idle.
#
#daemon,mail.*;\ 
#      news.=crit;news.=err;news.=notice;\ 
#      *.=debug;*.=info;\ 
#      *.=notice;*.=warn       /dev/tty8

```

Öncelik	Kullanım
debug	Hata ayıklama olarak adlandırılan mesajlar bir sistem hakkındaki bütün olan olayların bilgilendirmesini yapar. Bu mesajlara bir hata olduğu zaman ve diğer öncelliklerde bu log görülmüyorsa debug loglarına bakılır.
info	Servisler hakkındaki bilgilendirici mesajlardır.
notice	Bir sorun olabileceğine dair bilgilendirici mesajlardır.
warning/warn	Gerçek bir hatanın habercisidir. Anormal bir durum olduğunu bildirir.
error/err	Kritik olmayan bir hatayı bildirir.
critical/crit	Kritik bir hatayı bildirir.
alert	Servisin hizmetinin sona ermek üzere olduğunu bildirir.
emerg/panic	Servis hizmetinin sona erdiğini bildirir.

Uygulama Loglarının Rsyslog ile Toplanması

Örnek çalışmamızda Apache2 uygulamamızın **/etc/apache2/apache2.conf** isimli yapılandırma dosyasına giriyoruz. **ErrorLog syslog:local1** satırını ekleyip log seviyesini de warn dan debug'a çeviriyoruz. local0-7 aracılığıyla rsyslog içinde olmayan hizmetleri rsyslog'da yapılandırmak için kullanıyoruz.

nano /etc/apache2/apache2.conf

```
GNU nano 4.8                               /etc/apache2/apache2.conf
#
ErrorLog ${APACHE_LOG_DIR}/error.log
ErrorLog syslog:local1
#
# LogLevel: Control the severity of messages logged to
# Available values: trace8, ..., trace1, debug, info,
# error, crit, alert, emerg.
# It is also possible to configure the log level for
# "LogLevel info ssl:warn"
#
LogLevel debug
```

Ardından **systemctl restart apache2** komutuyla servisi restart ediyoruz.

Ardından nano /etc/rsyslog.d/50-default.conf komutuyla rsyslog yapılandırma dosyamıza girip **local1=debug -/var/log/apache.log** satırını ekliyoruz. Bu satırla apache2 içinde debug loglarını local1'e gönderdiğimiz logları alıp apache.log isimli dosyaya yazılacağına talimatını verdik. Dosyayı kapatıp **apache2** ve **rsyslog** servislerini restart ediyoruz.

```
GNU nano 4.8                               /etc/rsyslog.d/50-default.conf
# First some standard log files. Log by facility.
#
auth,authpriv.*                      /var/log/auth.log
*.*/auth,authpriv.none                -/var/log/syslog
#cron.*
#daemon.*
kern.*                                -/var/log/kern.log
#lpr.*
mail.*                                 -/var/log/mail.log
#user.*
local1=debug                           -/var/log/apache.log
```

/var/log/ dizinine gittiğimizde **apache.log** dosyamız oluşmuş.

```
-rw-r----- 1 syslog      adm   2020 Feb 21 11:26 apache.log
```

Dosya içeriğine baktığımızda logların yazılmaya başlandığını görüyoruz. Dilerseniz log seviyesini değiştirerek warning'den itibaren önemli logları da toplayabilirsiniz.

```
root@ubuntu:~# cat /var/log/apache.log
Feb 21 11:26:34 ubuntu rsyslogd: [origin software="rsyslogd" swVersion="8.2001.0" x-pid="1793" x-inf
o="https://www.rsyslog.com"] exiting on signal 15.
Feb 21 11:26:34 ubuntu systemd[1]: Stopping System Logging Service...
Feb 21 11:26:34 ubuntu systemd[1]: rsyslog.service: Succeeded.
Feb 21 11:26:34 ubuntu systemd[1]: Stopped System Logging Service.
```

Log Döngüsünü Ayarlama – logrotate

Logrotate, sunucunuzdaki logları dosyalara sürekli olarak diskinizin dolmasını önler. Belirli bir eşiğe ulaşan log dosyası kapatılarak yeni bir günlük dosya oluşturulur. Bu işlemi de crond servisi tarafından başlatılarak yapar. Varsayılan olarak 4 adet log dosyası tutulur. Haftalık olarak eski dosyalar 1'den 4'e kadar numaralandırılır. Aynı zamanda dosyaların sıkıştırılma işlemleri de gerçekleştirilebilir.

Logrotate'in varsayılan yapılandırmaya göre hareket eder. Hizmet verdiğiniz servise göre logları tutma süreniz daha fazla olması gerekiyorsa logrotate yapılandırma dosyaları ile bunu sağlayabilirsiniz.

Logrotate yapılandırma dosyası /etc dizini altında bulunan logrotate.conf dosyasıdır. Genel yapılandırma bu dosyadan yapılırken belirli bir uygulama için yapılandırma /etc/logrotate.d dizini altındaki dosyalarla yapılır.

```
GNU nano 4.8                                     logrotate.conf
# see "man logrotate" for details
# rotate log files weekly
weekly

# use the adm group by default, since this is the owning group
# of /var/log/syslog.
su root adm

# keep 4 weeks worth of backlogs
rotate 4

# create new (empty) log files after rotating old ones
create

# use date as a suffix of the rotated file
#dateext

# uncomment this if you want your log files compressed
#compress

# packages drop log rotation information into this directory
include /etc/logrotate.d
```

```
root@ubuntu:/etc/logrotate.d# ls
alternatives  apport  bootlog  dpkg      ubuntu-adantage-tools  unattended-upgrades
apache2       apt      btmp     rsyslog   ufw                  wtmp
```

rsyslog için oluşturulan logrotate dosyasını incelersek iki bölümden oluştuğu görülür. İlk olarak syslog dosyasının günlük olarak 7 kopyasının tutulması belirtilmiş. / gün boyunca eski log dosyası kapatılıp yeni dosya açılacak ve 8nci olan silinecek.

İkinci kısımda rsyslog'un diğer kontrol ettiği loglar için yazılmış bir scrip var. Burda da haftalık olarak 4 kopya ve kopyaların sıkıştırılması istenmiş.

```

root@ubuntu:/etc/logrotate.d# cat rsyslog
/var/log/syslog
{
    rotate 7
    daily
    missingok
    notifempty
    delaycompress
    compress
    postrotate
        /usr/lib/rsyslog/rsyslog-rotate
    endscript
}

/var/log/mail.info
/var/log/mail.warn
/var/log/mail.err
/var/log/mail.log
/var/log/daemon.log
/var/log/kern.log
/var/log/auth.log
/var/log/user.log
/var/log/lpr.log
/var/log/cron.log
/var/log/debug
/var/log/messages
{
    rotate 4
    weekly
    missingok
    notifempty
    compress
    delaycompress
    sharedscripts
    postrotate
        /usr/lib/rsyslog/rsyslog-rotate
    endscript
}

```

21 - GÖREVLERİ ZAMANLAMA

Linux sisteminde, bazı görevlerin düzenli olarak otomatikleştirilmesi gereklidir. Her gün tekrar eden görevlerle başa çıkmak için her görevde özgür özel çözümler bulmak bir seçenek olabilir ancak her bir süreç ile ayrı ayrı ilgilenmek çokta verimli olmaz. Bu nedenle Linux'ta işlemleri belirli zamanlarda otomatik olarak çalıştırılmak için Cron servisi kullanılır. Cron, bir işi belirli zamanlarda otomatik olarak çalıştırmayı sağlayan en etkili servislerden biridir.

Cron Servisini Yönetme

Cron servisi sunucu başladığında otomatik olarak başlatılır. Bunun nedeni Linux sisteminde kendi içindeki log temizliği, bakım gibi görevleri otomatikleştirmek için Cron'u kullanmasıdır. Cron servisi çok fazla yönetilmeye ihtiyaç duymaz. Kendisine tanımlanan görevleri her dakika kontrol eder. Eğer görev varsa yerine getirir.

systemctl status cron -l komutuyla servisin durumu görüntülenir.

```

root@ubuntu:~# systemctl status cron -l
● cron.service - Regular background program processing daemon
   Loaded: loaded (/lib/systemd/system/cron.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2021-02-21 09:22:46 UTC; 4h 7min ago
     Docs: man:cron(8)
 Main PID: 816 (cron)
    Tasks: 1 (limit: 2204)
   Memory: 2.1M
      CPU: 0.000 CPU(s) since start
     CGroup: /system.slice/cron.service
             └─816 /usr/sbin/cron -f

Feb 21 12:55:01 ubuntu CRON[2787]: pam_unix(cron:session): session closed for user root
Feb 21 13:05:01 ubuntu CRON[2792]: pam_unix(cron:session): session opened for user root by (uid=0)

```

Cron yapılandırma dosyası crontab /etc dizini altında yer alır. Ayrıca saatlik, günlük, haftalık ve aylık olarak zamanlanmış görevler yine /etc dizini altındaki cron.hourly, cron.daily, cron.weekly ve cron.monthly dizinleri altında bulunur.

```
root@ubuntu:/etc/cron.daily# ls
apache2 apt-compat dpkg man-db sysstat
apport bsdmainutils logrotate popularity-contest update-notifier-common
```

Cron Zamanlamasını Anlama

Hizmetleri, cron kullanarak planlarken, hizmetlerin tam olarak ne zaman başlatılması gerektiğini belirtmek için bir zaman dizesi kullanılır.

Zaman dizesi sırasıyla dakika, saat, ayın günü, ay ve haftanın günüden oluşur.

```
# Example of job definition:
# ----- minute (0 - 59)
# | ----- hour (0 - 23)
# | | ----- day of month (1 - 31)
# | | | ----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | ----- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | |
# * * * * * user-name command to be executed
17 * * * * root cd / && run-parts --report /etc/cron.hourly
```

Alan	Alibilecegi Değer
Minute	0-59
Hour	0-23
Day of month	1-31
Month	1-12 veya İngilizce ay isimlerinin kısaltmaları Jan,feb,mar,apr,may,jun,jul, aug,sep,oct,nov,dec
Day of week	0-6 (Günler Pazar'dan başlayarak, Pazar günü 0 veya 7 olabilir) sun,mon,tue,wed,thu,fri,sat

Crontab Değeri	Anlamı
17 * * * *	Her 17 dakikada bir bu görev çalışsin.
*/3 * * * *	Her 3 dakikada bir çalışsin
47 6 * * 7	Pazar günleri saat 6:47'de çalışsin.
* */4 12 * *	Her ayın 12nci günü 4 saate bir çalıştır.
* 5 1,3,5 3 *	Mart ayının 1,3 ve 5 nci günlerinde saat 5'de çalıştır.

Cron Configurasyon Dosyasını Düzenleme

Cron, /etc dizini altındaki crontab dosyasını yapılandırılarak veya crontab -e komutuyla login olunan kullanıcıya ait bir cron dosyası oluşturulularak kullanılır. Aradaki fark /etc dizini altındaki crontab dosyası sistem tarafından kullanılır. Bu nedenle bir kullanıcı crontab içerisinde eklediği görevleri görmek için crontab -l komutuyla görevleri listelemek istediğinizde herhangi bir görev göremez. Kendi oluşturduğunuz görevleri görebilmek için crontab -e komutuyla açılan dosyayı düzenlemelisiniz.

nano /etc/crontab

```
17 * * * * root cd / && run-parts --report /etc/cron.hourly
25 6 * * * root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
```

İlk satırda her saatin 17nci dakikasında root olarak cd / ile root dizinine gir ve /etc/cron.hourly dizin altındaki scriptleri çalıştır.

Satırlara bakıldığından satır sonlarında /etc/cron.hourly, daily gibi ifadeler gösterilen klasör altında tanımlanmış uygulamalar için olan scriptlerin yolunu gösterir. Daily klasörünü gördiğimizde çeşitli servisler için hazırlanmış scriptler bulunmaktadır. Daily ilgili olan satırda, hergün 6:25'de daily dizini altındaki scriptlerin çalıştırılması tanımlanmış.

```
root@ubuntu:/etc/cron.daily# ls
apache2 apt-compat dpkg man-db sysstat
apport bsdmainutils logrotate popularity-contest update-notifier-common
root@ubuntu:/etc/cron.daily# cat apache2
#!/bin/sh

# run htcachecclean if set to 'cron' mode

set -e
set -u

type htcachecclean > /dev/null 2>&1 || exit 0
[ -e /etc/default/apache-htcachecclean ] || exit 0

# edit /etc/default/apache-htcachecclean to change this
HTCACHECLEAN_MODE=daemon
HTCACHECLEAN_RUN=auto
HTCACHECLEAN_SIZE=300M
HTCACHECLEAN_PATH=/var/cache/apache2/mod_cache_disk
HTCACHECLEAN_OPTIONS=""

. /etc/default/apache-htcachecclean

[ "$HTCACHECLEAN_MODE" = "cron" ] || exit 0

htcachecclean ${HTCACHECLEAN_OPTIONS} \
    -p${HTCACHECLEAN_PATH} \
    -l${HTCACHECLEAN_SIZE}
```

Örnek Bir Görevi Zamanlamak

crontab -e komutuyla açılan dosya içerisinde basit bir görev yazalım. Her dakikada bir görevler dizini altındaki test.txt dosyasına Linux içeriğini ekle dedik.

crontab -e

```
*/1 * * * *      root      echo Linux >> /gorevler/test.txt
```

crontab -l komutuyla oluşturduğumuz görevi listeliyoruz. Dikkat ederseniz ana crontab dizin altındaki görevler burada listelenmedi. Burada sadece kullanıcının oluşturduğu görevler listelenir.

```
root@ubuntu:~# crontab -l
*/1 * * * *      root      echo Linux >> /gorevler/test.txt
```

5 dakika sonra kontrol ettiğimizde her dakika için bir Linux yazıldığını görüyoruz.

```
root@ubuntu:/gorevler# cat test.txt
Linux
Linux
Linux
Linux
Linux
```

Cron ile yedekleme, bir servisi restart etme gibi görevleri yerine getirebilirsiniz.