



Linux Notları

Alihan Özen



İçerik

1	Temel Bilgiler	3
2	Temel Komutlar	3
3	Dosyalar İle İlgili İşlemler	4
4	Processler	6
5	Grup İşlemleri	8
6	Shell Kolaylıkları	10
7	Environment Variables (Ortam Değişkenleri)	10
7.1	Ortam Değişkenlerini Oturum Boyunca Kullanabilme	10
8	VI (Visual Editory) Temel Kavramlar	12
9	Sistemin Donanımını Alma	13
10	Disk	13
11	Formatlama	14
12	Kota	15
13	Arşivleme	15
14	Sıkıştırma	16
15	Zamanlı İşler	18
16	Paket Yönetimi	19
17	Sistem Dosyaları ve Çekirdek . . .	21
18	Shutdown	24
19	Network	24
20	Servisler	26
21	Güvenlik	26
22	ACL	26
23	LOG	27



24 Script Komutları	27
25 Yazıcı	28

1 Temel Bilgiler

- **Console** : ekran-klavye-mouse (doğrudan bağlı olanlar)
- **Ctrl Alt F1** ile açılır,şifre girilirse shell açılır
- Kabuk komut vermemizi sağlayan ekrandır, **Ctrl d** ile logout yapılır.
- **Terminal**: uzaktan bağlantı/kabuğa bağlanmamızı sağlayan program
- En çok kullanılan kabuk programları : **bash - zsh - tsch**
- **\$** yetkili kullanıcı işareti
- **bin**: temel kullanıcı komutları
- **sbin**: temel sistem yönetimi komutları
- **lib**: kitaplıklar
- **usr**: program kurduğumuzda dosyalar altındaki bin'e gider.
- **usr/share**: veri dosyaları
- **usr/local**: manuel kurulumda dosyaları içindeki bin-lib-sbin'e koymak mantıklı.
- **root/opt**: opsiyonel olarak gelen paketler
- **var**: değişebilen yerler/e-mailler,log dosyaları,printer dosyaları...
- **tmp**: dosya oluşturulabilir ve herkes görebilir
- **home**: her kullanıcının kendi dosyalarının olduğu yerler
- **etc**: bütün conf dosyaları / araçların ayar dosyaları burada yer alır
- **orn/passwd**: "kullanıcı adı/ev klasörü neresi /hangi kabuğu kullanıyor" yazar

2 Temel Komutlar

- **cd** (change directory) = **cd ~**
 - sadece **cd** yazarsak home directory yönlendirir.
- **cd -** : bir önceki klasör
- **cd ..** : bir üst klasör
- **pwd** : bulunduğumuz klasör



- `ls` (list) = `ls`.
- `ls -l` : bilgileri gösterir
- `ls -a` : tüm dosyalar(gizliler de dahil)
- `ls -R(recursive)` : altta ne varsa gösterir
- teminalde `dcrool` var ama konsolda yok
- `ls -lS(boya göre büyükten küçüğe) -lt(zamana göre yeniden eskiye)`
 - sonuna `r` (reverse) yazınca tersini yapar
- `man` (komut adı) : komut ile ilgili bilgi almak için
 - `man` sayfasından "`q`" ile çıkılır,"`b`" ile bir sayfa yukarı çıkılır.
 - `/s` yazarsak içinde `s` yazan yerleri gösterir
- `man`'da bazı komutlar olmayabilir, o zaman "`info`" komutu kullanılır.
- `ls -al` olur ama `ls -all -long` olmaz.
- **whereis** : bulunduğu dizini gösterir
- **which** : kısa bilgi verir /which sıralamada hangisini kullanacağını gösterir
- **whatis** : kısa açıklama yazar
- **whoami** : kim olduğunu söyler

3 Dosyalar İle İlgili İşlemler

- `ls`
- `cp` (copy)
 - `cp` (dosya adı) (yeni dosya ismi)
 - `cp` (dosya adı) *tmp* : *tmp* dizinine aynı isimle kopyalar
 - `cp` dosya dosya *tmp* : iki dosyayı yazılan dizine atar ancak iki dosya olduğu için klasör varsa yapar
 - `cp` dosya *tmp* yeni ad : dizine yeni ad ile kopyalar
 - `cp -i` : işlem sırasında sana soru sorar/bazı makineler sormadığı için
 - `cp -R` : recursive
- `rm` : silmek
 - `rm -i` : sana sorar silmeden önce
 - `rm -f` : force



- rm -rf : sormadan siler
- dosya oluşturma ve silme
 - mkdir __ : oluşturma
 - rmdir __ : silmek
- mkdir -p (parents)
 - mkdir 1/2/3 1 ve 2 olmadığı için 3 oluşmaz
 - mkdir -p ile olur.
- rmdir -p :en alttan yukarı doğru siler/içi boş olanı silmez
- tree __ : şema halinde gösterir
- mv : taşıma / isim değiştirme (işlem yaptığımızı siler.)
- cat : dosyanın içine bakmak /contanate : peş peşe ekleme
 - cat 1 2 : sıra sıra gösterir
 - cat -n dosya : satır numaralarımıda yazar
 - more'da kullanılabilir cat yerine
 - less ile more aynı şekilde kullanılır
- head -5 (5 satırı göster) / sondan da gösterebilir
 - tail -5 (son 5 satır)
- file __ : dosya tipini gösterir
 - file "__" ya da /_____
- wc ls -l __ (dosya satır/sözlük/byte)
- wc -l : kaç satır
- find __ : altındakilerin listesi
 - find ____/ -name file.png : kaç tane file.png dosyası var
 - find ____/ - name .doctrees -type f(file) ya da -d (directory)
- du dosya/ : içindekilerin kb cinsinden değeri(disk usage)
 - du -s* : içindekileri kısa özet yapar
 - du -sm : megabyte cinsinden gösterir
 - find ____/ size +1M : 1'mb dan büyük dosyalar
- pipe (boru hattı) "|"



- çıktıyı bir yere göndermek için "-" kullanılır
 - cat dos1- dos2 (aynı isimli dosya varsa ezer,onun için – kullanılır)
- tail -f (follow)
- ln (link komutu)
 - ln a.txt b.txt : b.txt a.txt'nin linki / birini silince diğeri gitmez(hard link)
 - hard diskte yer tutmaz hard link ve soft link
 - soft link : ln -s a.txt b.txt biri değişince diğerede etkilenir (link mavi gözüktür.)
 - kaynak dosya silinince diğerede gider

*usr/share/doc : ubuntu'nun paketlerini kurunca dökümantasyonları buraya kurar.

- klasörlerde hardlink yapılamaz
- xviewer : resim dosyalarını görmek için program
- convert x.png x.jpg (uzantı değiştirme)
- wget (internet adresi) : o dosyayı indirir
- lynx(resim çalışmayan bir browser) : açar biz istersek indirir.
- wget ___ / mirror : ne var ne yok indirir
- curl : dosyayı ekrana yazar,indirmez (f-x.pdf yaparsak kaydeder/ -o dersek sadece ekrana basmaz,indirir)
 - -o x.pdf dersek git o dosyaya yaz demek.(curl ve wget için de geçerli)

4 Processler

- ps : çalışmakta olan programlar ile ilgili bilgi verir.
 - PID: her prosesin sistemdeki özel numarası
 - TTY:hangi konsol/terminalde çalıştığını gösterir
 - ps a : userin bütün terminallerdeki prosesleri
 - ps ax: terminal harici çalışan kullanıcılar ("?" olarak gözüktürler.)
 - aux : hangi kullanıcıda olduğunu söyler
- | more : sayfa sayfa göster
- top : sistemi en çok yoran komutlar (açılan listede R:çalışan program S:uyku modunda bekleyen) (ag:printerde birşeyin boşalmasını bekliyor)
- anne prosesin yaratılması : "fork" anne prosesin çalıştırılması : "exec"



- PPID:anne prosesin kodu
- pstree : proses şeması
- deamon : sistemde otomatik olarak çalışan prosesler(örnek:e-mail alan process/login printer sistem saati) genel konvansiyonel deamon proseslerinin sonunda d harfi vardır.
- komutu arka planda çalıştırmak için sonuna &(ampersant) koyarız.
- (komut) - /dev/null (çalışsın ama görmek istemiyorum)
- jobs : hangi komutlar çalışıyor onu görürüz
- fg : ön plana alma / fg %1 / %2
- ctrl z ile durdurulur ama sonlandırılmaz / sonra fg dersek arka plana atarız.
- her prosesin 3 iletişim kanalı vardır
 1. stdin (standart input)
 2. stdout (standart çıkış/yani ekran)
 3. stderr (error / bunların numarası da 0 -1 -2 tur sırasıyla)
- örnek = ls -lR | -files.txt 1 - x.txt bu şekilde errorlar ekrana gelir 2 - yazarsak ekrana yazmaz / < yaparsam inputu yapar aynı dosyaya yazıcaksak -&1
- proses 1'i öldürmek için kullanılan komut "kill"
- kill "numara" sinyal ile öldürme (numara için "ps ax")
- kill -9 : öldürme sinyali kill -9 7232 dersek öldürür "sigkill" ya da "sigint (interrupt)/ctrl c" durdurma komutları
- hangup : işini bitir kontrollü kapat
- ps ax'te s ve r'ler iyidir/d'ler zombi-ölmemiş proseslerdir. annesi ile ilgili sorun vardır/ölemez
- process bazen runaway kaca proses olabilir/makina donar ve mahfolur
- killall -9 yes : bütün bu kalıba uyan şeyleri öldür
- skill (super kill) : belirtilen kullanıcıyı sonlandır skill -u ____
- proseslerin öncelikleri "nice values"

-meşgul eden proses örneğin "nice ls -lR" dersek önceliği düşer.

- 40 öncelik seviyesi vardır (-20den +19a kadar/ne kadar düşükse o kadar az öncelikli çalışır)
- sadece yetkili user seviye yükseltebilir
- nice sadece komutu çalıştırmadan yapılabilir
- örneğin : nice yazmayı unutursak,komutu arka plana attık



- ps ax ile pid'i aldım
- renice process numarasını yazdım
- renice -10 (numara) en fazla 20 düşürülebilir
- bir komut çalışırken terminali kapatırsak bağlantı kopabilir ve zombiler oluşabilir
 - anne terminali kapattığımız için geri kalanlar kopar ve durdurmak gerekir
 - başına "nohup" yazarsak annesi ile bağlantısı kopar,terminal kapansada ölmez
 - nohup kullanmak zombiye dönüşmesinden daha iyi
- cat /etc/passwd

user adı;x(parola);kullanıcının user id;kullanıcıların grup id;(gecos kısmı)userin full name,tel no(zorunlu değil);userin home directory;userin kullandığı kabuk örnek: mint;x;999;999;live session user,,;/home/mint;/bin/bas

- chfn : change full name
- chsh : shell değiştirilebilir
- passwd ile şifre değiştirilebilir
- getent passwd : makinada ldap ya da radius login olabilecek kullanıcıları gösterir
- /etc/shadow : makinaya login olmamış kullanıcıları ve hashlenmiş şifreleri gösterir
- sudo ile root yetkileri veririz çünkü root normalde login olamaz
- /etc/sudoers : sudo yetkisi olabilecek kullanıcılar
- hangi makinalar : host alias spec hangi userlar : user alias spec hangi komutlar : cmnd alias spec

5 Grup İşlemleri

- her local kullanıcı bir gruba dahil olmalıdır.(/etc/groups altında yazarlar)
- groupadd : yeni grup oluşturma
- sudo groupadd -g (grup numarası) (users grubun adı)
- lokal gruplar iç içe değildir,ldap ile yapılabilir
- groups : kullanıcı hangi gruplarda (önce hangi gruplar)
- adduser (kullanıcı adı) (grup adı) :gruba kullanıcı ekleme
- sudo /etc/groups dosyasını açıp buradan eklemek daha iyi
- groupdel (grup adı) : silme
- user oluşturucam ama kafasına göre grup açmasın diyorum
 - useradd ile kullanıcı oluşturdum



- Kullanıcı Değiştirme -makina yeni local user eklenmelidir: /etc/group /etc/pass /etc/shadows /etc/home dir
- buralara yazması lazım, home ve shell hariç diğerlerine yazmış bu yüzden *d ile home dir. -s ile shell verilir.
- -m : home dir yaratmak için kullanılır
- ls /etc/skel adlı directory'den yeni yaratılan kullanıcıya default kopyalanacak dosyaları atar
- group pass shadows home dir yarattı
 - etc/
 - grup dosyasına yaz
 - pass dosyasına yaz
 - shadow dosyasına yaz
 - home directory yaratmak
 - etc/skel (standart kullanacağı şeyler)
- sudo userdel -r == (dosyaları ve useri siler)
- sudo useradd -g users -d /home/_____ -m -s /bin/bash
- su : başka bir kullanıcıya geçme
- ctrl d : eski usera geri dönülür #Directory İzinler
- r : dosya listesi alma
- w : dosya oluşturma
- x : dosyaya geçebilme
- chmod : dosya/dizin yetkisi değiştirme komutu chmod g+w == (gruba write yetkisi) ya da u+r , g-w ...
- u+x o(others)-r
- chmod a+e __.txt , go-w __.txt (a : all users)
- chmod a-x * hepsinden kaldırma(dizine geçmeyi engeller)
- a+X * sadece çalıştırma izni olan directorylere verir
- r:4 w:2 x:1 hiç bir yetki :0 chmod 444 deneme.txt
- umask : yeni yaratılan dosyalarda hangi izinle kaldırılacak(oturum bazında yapılan bir ayar)
- chown : sahip değiştir chown (sahibi) (başka user) deneme.txt
- chgrp ile kendi gruplarım arasında değiştirebilirim



- chgrp (grubun adı) deneme txt
- w : hangi kullanıcı nereden girmiş
- who : login user (:0 yazarsa yanında terminalden girdiğini gösterir)

6 Shell Kolaylıkları

(bash shell)

- alias : var olan komuta yeni isim verme alias ll="ls -l"
- unalias : yaptığımız komutu silme / unalias -ll
- alias rm="rm -i" /mv/cp : bazı dağıtımlarda var önemli bir kullanım

7 Environment Variables (Ortam Değişkenleri)

- bazı durumlarda prosesler bu değişkenlere göre çalışabilir
- echo : ortam değişkeninin durumu
- değişkenler \$ ile başlar/genellikle tümü büyük harftir
- echo \$HOME ya da echo\$PWD = home/mint
- echo \$PATH : normal dosyalardan bahsederken gidilecek yolu gösterir
- echo PS1 : tilde in nasıl gözükeceğini yazar (renk kodu sonra dolar vs...)
- set : bashyeki tüm değişkenlerin değeri sonuna -more dersek parça parça gösterir
- LANG=tr_TR __ : hangi dili yazarsan o dil ile çalışır
- LANG=en_US / UTF-8 / ISO8859-9
- LC_MONETARY : para sembolü olarak ne kullanacağını belirler
- LC_COLLATE : sıralama için türkçe kurallarını kullan / mysql db bu şekilde programın kurallarını değiştirebilirim örneğin program i'yi unicode'a göre I yapıp z'nin arkasına atabilir.

7.1 Ortam Değişkenlerini Oturum Boyunca Kullanabilme

- export __ : __ neydi ?
- exportun başına \$ konmaz örn : export ____="\$" dersek sadece \$ gözüktür örn : export ____="\$" yapınca cd /home/mint'sin der.
- !/bin/bash : shabang
- export PATH =\$PATH:\$HOME/.local/bin : path değıştime normalde kullanıcılar usr/local/bin diye yazar



- export PATH=\$PATH : bulunduğun klasörü pathe eklemek(pek önerilmez)
- l.='ls -d .*' : çok kullanılır
 - neredeyse tüm programlar conf dosyalarına . ile başlayan isimler verir.(kalabalık etmemesi için)
 - örn: .gitconfig de email name gibi conf dosyaları var.

-.program rc gibi kullanır genelde .bashrc gibi bash ilk çalışınca default şeyleri okur sonra userin home directorysindeki conf dosyasındakileri okur ve çalıştırır

- ubuntuda .bashaliases'e yazıyoruz.
- alias htu xxx yazarsak dosya çalışır ama htu olamaz
 - source.bash_aliases sonra tekrar htu dersek görür
 - ya da yeniden bash çalıştırsak olur
- . demek source yerine geçer/source bash komutudur
- more etc/bash/bashrc de sisteminkiler var sistem yukardakilere bakar sonra home directory dekinde bakar orada home altındaki bashaliases'e source ettiği için oradakileri çalıştırır.
- ls .config/ : çoğu program buraya koyuyor
- program kurulunca ya da çalışınca conf dosyası değilse .local/share altına koyuyor
- find / -type f name -name "x*" uzun komut çalıştırdık
 - sonra durdurduk | more dersek sayfa sayfa çalıştırır
 - * !! dersek son komutu tekrar çalıştır demek
- !\$: son komutun son sözcüğü
- rm !\$: komutu düzenle sonra sil
- history : eski komutlar
- which __ ile numarasını bulurum örn:1277 !1277 yazarsam çalıştırır
- !__ : hatırlaymadığın komutun başını hatırlıyorsan bulabilirsin
- more .inputrc
 - w yazıp basınca history içinde gezinirim
- komutun başına bir boşluk koyarsam history'e girmez
- ortam değişkeni unset HOSTFILE dersem oturum sonunda silinir historyler



8 VI (Visual Editory) Temel Kavramlar

- vim = vi (eski adı) vim paketi kurmak lazım
- vi (edit etmek istediğim dosya) örn: vi bash_aliases
- edit moda geçmek için a harfi (append) vi a
- satır üstünde a'ya basınca sağında yazma izni verir
- esc ile komut moduna geri dönülür
- i harfi (insert) soluna yazmaya izin verir
- komut modunda x bulunduğu harfi silme
- r üstüne yazma/değiştir
- u (undo) değişikliği geri al
- yeni satır için enter
- komut modunda tilde ye basınca o harfi büyük yapar
- . son komutu tekrarla(neyin üstündeysem)
- 2 kere dd yaparsam satırı siler ya da edit mode ile backspace ya da delete yaparım
- yy (yang yang) : kopyala / p (paste)
- n bir sonraki dosyaya geçme
- dosyanın sonuna ile G ile giderim
- satır no verip G yaparsam o satırın başına gider (1G en baş)
- cut-paste = dd-p
- / aramak için moredaki gibi
- n tuşuna basarak bulurum
- :wn sonraki dosyaya geç
- :q yazarsak sorar :wq yaparsak sormadan
- :q! kaydetmeden çıkma
- :w (dosya adı.txt) editlediğimizi save as gibi yapar ve kaydeder ona geçer otomatik,eski dosya eski hali ile kalır
- shift ZZ = :wq
- ctrl z ile vi durur çalıştırmak için fg ile tekrar öne gelir
- ctrl q ile çıkılır



- ctrl s ile kilitlenir (scrool lock)
- reset yazarsan termianlde ekran düzelir(ekranda saçmalıklar olduğunda)

9 Sistemin Donanımını Alma

- cat /etc/iss (issue) : hangi os versiyonu
- uname -a : daha ayrıntılı bilgi için uname -r : çekirdek numarası
- free : ram miktarı free -h (human) : insan dilinde gösterir
- swap: takas yeri gibi,sistem kullanmadığı şeyleri tutuyor (belleğin 2 katı boyutunda)
- proc : root'un altında / sistemi gözetlemek ve yönetmek için kullanılan bir directory ls ile listelenir
- proc altında her proses için bir klasör var bu da onun numarası ls (numarası) : proses ile ilgili bilgiler cat (numarası) / cmdline : nasıl çalışır usb-devices : hangi usbler bağlı lsusb : daha iyi görürüz usbleri lshw : tüm aygıtları gösterir
- linuxta her aygıtın bir "device node" su vardır,her aygıta bir isim verilir, /dev in altında ls dersek tüm aygıtlar için düğümler oluştuğunu görürüz.
- 2 tür aygıt var temelde 1-karakter aygıt (blok aygıt harici şeyler) 2-blok aygıtları(üstünde dosya tutulabilecek şeyler/ disk ram)
- b : blok aygıt l : soft aygıt c : karakter aygıt d : klasör
- : dosya
- aygıtlar diskte yer tutmaz ve dosya değildir
- aygıt için ya herkese izin vericem ya da bir gruptakilere
- mknod : yeni nod oluşturmak (örn : devicenode)
- console giriş çıkış yaptık ls -l tty* yazdım tty onun aygıtı grafik ekranlar tty den sonraki numaralı olanlardır
- unix te her aygıt için /dev/node adları ve kısaltmaları var (örn : tty console / lp line printerlar)

10 Disk

- ön kısaltma eki : SD (eskiden HD : bu ide'ler için sonra sc çıkmış ama kalmamış sonra ssd ama ssd yede sd deniyor artık)
- sda : 1.disk sdb:2.disk sda1: 1.diskin 1.partition flash disk takınca devicenode'u kendi yaratır
- df : diskler ne kadar dolu en sonda mounted on yazar (bir disk partition ile klasör arasındaki bağlantı)
- mount : nereye bağlı olduğu hakkında detaylı bilgi



- vfat ext4 dosya sistemi tipleridir
- ext4 linuxun standart dosya sistemi tipi
- jfs zfs vfat raiser ntfs bunlarda var
- mounting işlemi : 2 türlü olur
 1. otomatik (örneğin flash taktık,df ile bakınca node'unu yaratıp mount ettiğini görürüm,dosya sistemi ile çalışır,node'un yaratılmasını çekirdek yaptı,dosyaların açılmasını lxd(xfc hangi dağıtım yüklüyse))
 2. otomatik istemiyorsam settings-removable drives
 - mount /dev/sdc1 /mnt yazınca mnt'ye bağlar (mnt varsa)
 - mount edilecek klasörün boş olması lazım / sonra ls -mnt ile içeri görürüm
 - sudo umount yazıp klasör ya da aygıt adı verilince unmount eder
 - mount u -t (dosya tipini anlamazsa kendimiz söyleriz)
 - more /etc/fstab file system table'a girip yazarsak açılışta otomatik başlar
 - ! bir diske önce partition açılmalı sonra format sonra mount olmalı boş disk için
 - partitioning fdisk ile yapılır(parametre olarak DISK verilir)
 - oluşturduk diyelim "t" ile tipi değişir
 - sadece 4 primaryt pertition yaratılabilir eğer dördü geçeceksem birini extended yaratıp logical partition oluşturmalyım
 - sistem diskini logical diske koyamayız,primary tipli bir diske koymalıyız
 - extended partition : doğrudan formatlanamaz

11 Formatlama

- sdc1 sdc2 filan var
 - sudo mkfs make file system
 - örn : mkfs -vfat ya da -ext4 denilebilir (.ext4 olabilir)
 - sonra partition'u veriyoruz
- sonra mount etmeliyiz (swap formatlama mkswap ile)
- swap mount edilemez , /etc/fstab'a yazdık ama restart etmek istemiyoruz ,o anda swapon swap off komutları var
- swapon -a : tüm swaplar etkinleşir = swapon /dev/sdb2
- swap alanını yapabilirsen partition yap / file olarak performans zararı var



12 Kota

- kullanıcılara ve gruplara tanımlanabilir
- 3 kavram vardır 1-hard limit : kullanıcın geçemeyeceği kota 2-soft limit : sınırı geçebilir ,katı sınıra kadar gider ve bu kullanıcıya uyarı gider 3-grace period (izin verme süresi) : kullanıcıyı uyardıktan istediğimizi belirlediğimiz süre sonunda yapıyoruz
- bir sistemde kota kullanıcaksan more /etc/fstab dosyasında opsiyonlar var örn : defaults gibi
 - /dev/sdc1 aygıtını /disk2'ye bağlamak istiyorum (tip ext4)
 - opsiyon vereceğim : defaultları kullan usrquota,grpquota 0 0
 - sonra "sudo quotacheck -avug (a:all v:verbosa)" ne yaptığını söyle
 - örneğin user sonra grup dersek ne yapar : /dev/sdc1 in bağlı olduğu yerde disk2 altında herşeyi sayıyor
 - sonra bu dizinde klasörler oluşturup içine çıkan sonucu yazar
 - sudo edquota __ , editör çıkar burada ayarları elle giriyorsun
 - sonra aktif olur
 - (repquota : rapor vermeye yarar)

13 Arşivleme

- unixede zip kullanılabilir
- *.pdf dersem hepsini zipler
- unzip -t __.zip : içini görmek için kullanılır,bozulup bozulmadığını... -t demezsek dizini açar sadece
- *.gif : sadece gifleri çıkartırsam
- zip -r : klasörü istersek zip 2 iş yapar 1-birden fazla dosyayı tek hale getirme 2-sıkıştırma
- tar : sadece arşivleme yapar
- tar'ın yanına komut verilmeli c:create v:verbose(yaptığın işi söyle)
 - ekranı script kullanırken v yazılmaz
 - genellikle cvf kullanılır
 - tar cvf f (oluşturacağım arşivin adı.tar) *.pdf (sonra ekranda gösterir)
 - test tvf __.tar : sağlam mı diye bakar
 - xvf : açmak için (x:extract)



14 Sıkıştırma

- gzip ile olur / gzip __.jpg
- eskisini siler ,silmemesi için gzip -k(keep) __.jpeg
- gunzip : açmak için
- gzip -d : açmak için (bu da gzip'i siler, silmemesi için yine keep)
- gzipin 1den 9a kadar seviyesi vardır
- en çok sıkıştıran 9 ya da default
- gzip -9 __.tar yaparsam eskisini siler __.tar.gz yapar
- zcat : sıkıştırılmış dosyayı okuma
 - zmore ya da zless kullanılabilir
- tar cvzf : bu dosyayı gzip'e ver
- örn : tar cvzf __.tar.gz *.pdf : ismini tar.gz yapıp içindekileri pdf olarak gösterir (yaygın bir kullanım)
- xvzf ile de aynen açılır
- gzipten sonra bzip2 (.bz2) çıktı
 - açmak için bzip2 -d
 - tar cvjf __.tar.bz2 *.pdf : tar'da bz2 kullanmak için
- bzip2 ve bzmore da kullanılabilir
- yakın zamanda xz çıktı
 - tar cvjf __.tar.gz *.pdf : daha iyi sıkıştırır
- tar'da v yi kaldırırsak ekrana basmaz , zipde -q yazarım

not : kullanıcın destktoptaki dosyalarını başka userlar görebilir

#PARÇALAMA

- split ile olur , split __.tar
- split -b (byte cinsinden) mesela 4G dersem 4G'a kadar demiş olurum
 - dosyayı yazıyorum prefix olarakta myxxx.pdfs dersem oluşturur
 - myxxxaa.pdf , myxxxbb.pdf , myxxxcc.pdf yapar
- içindekileri görmek için benim birleştirmem lazım
 - cat myxxxaa myxxxbb yazarsam ekrana yazar



- diff myxxx pdfs.tar ile bakarız
 - -n ile kaç bölçegimize söyleriz -t ile satırlardan yapar (txt ise mantıklı)
 - dd (data duplicate) : copy gibi
 - dd ile parametre kullanmak zorunlu / dd if input file yani of output file (bu aygıt okumak içinde kullanılır)
 - örn : bir flash'ın imajını alacağız
 - dd if=/dev/sdb1 of=flash.img
 - dd if=/dev/sdb of=/dev/sdc 2.flash diski yapmış olurum, 2. 1.nin görüntüsü olur
 - dd bs (block size) : farklı aygıtların değişik çalıştığı block boy yapar
 - count verebiliriz dd if=_____ of=_____ bs=1024 count=2000 (en baştan başla 1k lık şeyler halinde 2 blok kopyala demek
 - drescue /dev/cdroom mycd.iso : imajını çıkartır takıldığı yerde tekrar tekrar dener (aslında kpopy aygıtlar üzerinde çalışabilir)
 - fsck : dosya sistemi sağlam mı (chkdsk gibi)
 - mount edilmemiş dosyalarda yapılmalı
 - fsck -y : sormadan problemleri düzelt
 - sudo cp -a : dataların aynısını başka yerde aynı özellikler yazma okuma ile kopyalama (-ar r:recursive)
 - örneğin root home'un komple backup'ını alıyor
 - internetten birşey indirdik,tar dosyayı yaratırken -no-same-owner ile açarsak hakkını root'a verir
 - chown -R root .root dersek tek komutla sahibini değiştirebiliriz (hem kullanıcıyı hem grupları)
 - klasör backup alcam ama tar ile istemiyorum
 - cp -ar /dev/disk1 ile kopyalanır ama problem olabilir
 - rsyncs : senkronize
 - rsyncs -a (a:recursive)
 - rsyncs -av (v:verbosa)
 - rsync -av -delete *disk1 disk2*
 - dosya sistemi ntfs fat32 olursa -a düzgün çalışmaz o zaman t kullanmalıyız rsync -rvt -delete *disk1 disk2*
- yalnız t ile kopyalarken tarihlerde saniyelik farklar olabilir rsync -rvt -modify-window=1 -delete *disk1 disk2* yaparız o zaman (flash disk ile linux arasında kopyalama işleminde kullanılabilir)
- -size-only == örn : mp3 sadece boyuna bak diyebiliriz



- rsync hedef klasör olarak başka bir makinadaki yeride verebilir

örn : rsync -av --delete disk1 uuu:ssh.itu.tr:klasöradı/publicxx/ klasörüne kopyala demek secure shell ile (-avz ile zipleyip gönderir)

- --dry-run --delete : yapacağı işi simule edip çalıştırmıyor(silme işi açısından)
- rsync backp daha geniş özelliklere sahip bir program

15 Zamanlı İşler

- echo ls -l at 18:45 (belirttiğimiz saatte çalışır ama sonucu bilemeyiz.)
- echo "ls -l - files " at 18:45 çalıştırır ama çalıştığı yerde directory yok files nerde oluşur bilmiyoruz bu yüzden oluşturulacak path'ı vermemiz lazım
- shell dosyaları sh ile çalışır
 - cat islerim.sh
 - #!/bin/bash
 - ls -l - ~/file
 - gzip -9 ~/file

bunu shell'e vermek için at-f islerim.sh 19:00 yaparız -birden çok iş için echo "ls -l - ~/files && shut-down -h" | at at'i tek iş için kullanın,birden çok komut için shell script yazın ve buna at verin

- cron : çok işler için
 - crontab -e
 - ilk : dakika
 - ikinci : saat
 - üçüncü : ayın kaçınıcı günü
 - dördüncü : hangi ay
 - beşinci : haftanın kaçınıcı günü
 - sonrada komut
- örneğin tar için bir işlem yapıcaksak tar başında - olmaz
- * * * : hergün aynı saatte yap
- / kullanırsak (/5 * * *) her 5 dk'da bir yap
- crontab -l : userin işleri
- atq : kuyruktaki işler / atrm : işleri silme



- cron : tüm işleri iptal etme, tek iş iptali için edit ile girilip öyle iptal edilir
- örn : sudo crontab -u aaa -e : aaa isimli userinkiler gelir
- cron arkaplanda çalışan bir servis / deamon .

16 Paket Yönetimi

- rpm : redhat'ta var ,sadece al şu dosyayı kur
- dnf : store ile haberleşmeyi bilen programın adı
- ubuntu aslında debian adlı bir dağıtımın üstüne kuruludur,debianın paket yöneticisinin adı dpkg (rpm'in karşılığıdır) , daha üst düzeyinin adı apt-get
- artık apt kullanılıyor
- paket kurmayıp arıyacaksam sudo'ya gerek yoktur
- ilk iş veri tabanı güncel olmalı "sudo apt update"
- /etc/ap : repoların olduğu yer (örn : sources.list bunun altında darepositorylerin linki yazıyor)
- http'ler var https yayınlarda var (imzalı ve sürekli kontrol ediliyor)
- sudo apt upgrade : güncellenmesi gerekenler günceller
- sudo apt dist -upgrade : işletim sistemi dağıtımını upgrade eder
- debian'ın rolling edition'u var aslında,serverlarda debian kullanılmalı
- sudo apt install __
- apt install -no-install-recommends : nodeps dependencyleri kurma
- apt search : paketler arası arama yapma (kurulu olmasa bile)

(solunda harfler var i:installed p:present/available)

- apt show : paket ile ilgili bilgi
- sudo apt remove __ (conf dosyalarını silmez,sadece paketi siler)
- confları silmek için purge gerek (öngereksinimi yine silmez ama)
- sudo apt autoremove : yanına paket yazma (temizlik yapar)

ihtiyaç olmayan öngereksinimleri siler (kendin kurduysan silmez)

- sudo apt install __ : uygulama varsa yeni sürümünü kurar
- apt download __ : paketi indir ama kurma
- apt clean : indexleme cache yapar , yarıda kalan download varsa temizler
- apt contains : bu paket nereden geldi, kim kurdu



- sistemde kurulu olan birşeyi başka sistemede kurmak istersen apt contains /usr/bin/chromium-browser apt contains /usr/bin/convert
- sudo add-apt-repository ppa:webup... gibi formatı ppa'dır / server makinalarda bu iş yapılmaz genelde / client'te belki olur
- sonra sudo apt update ile buradaki paketler neler,refresh etmem lazım
- add-apt-key : imza eklemek için
- aptitude : apt gibi daha çok şey yüklüyor
- paket dosyalarının uzantısı debian'da deb'dir
- örn x.deb paketini manuel indirdiyem "sudo dpkg -i x.deb" ile kurabilirim
- sudo apt-get -f install : paket adı vermeden yazarsam eksik kalmış kurulumların neye ihtiyacı varsa onları kurar
- dpkg : tarlanmış,ziplenmiş dosyayı açar,o şekilde kullanma
- echo "xxx hold" | sudo dpkg --set-selections : uygulama otomatik update yapamasın
- dpkg -r : paket remove ederiz = apt remove / ama dpkg dependencyleri siler başka şey çalışmayacaksa silmez
- dpkg'yı sadece çekirdek ile ilgili şeylerde kullanabiliriz
- dpkg -l : sistemdeki tüm paketler ile ilgili bilgi verir
- dpkg -L __ : paket içindeki dosyalar neler = apt content __
- apt held : hold durumundakiler
- flatpak : yeni dosya sistemi kolay kurulum için

#Bir paketi kendim derleyip çalıştırıcım (örn : python)

- indirdim,(python.tar.xz)
- tar xvJF ile açıp,ls ile içine girdim
- configure adlı dosya ile derlenir __ ./configure --prefix yazarım (yazmazsam prefix/usr/local altına atar ama ~/.local de diyebilirim (bu işlemi sadece makedir diye bir klasör oluşturdu)
- make : makefile'i okuyup yapacaklarına karar veriyor,derlemeye başlar
- make install : kurar (usr/local'e kuracaksan sudo gerek)

#Sistemde kullanılan default uygulamalar

- örneğin java kullanacağım,sistemden gelen java openjdk ama ben oracle javasını kurdum
- çalıştırdım sisteminki geldi ama oracle'nki gelmiyor



- birden çoksa system environment'e eklenir ama tüm kullanıcıları ekliyeceksek (dosyaya zarar verebilir)
- daha kolayı var : sudo update-alternatives --config x-www-browser dedim buradan hangisinin çalışacağını seçebilirim
- update-alternatives --install java :dersem yeni gelen java da bu seçenekler arasına girsin ve öncelik verilirse o kullanılsın
- sls /usr/share/applications : desktop dosyaları var oradan silersen o uygulama desktoptan kalkar

17 Sistem Dosyaları ve Çekirdek ...

- çekirdek aslında bir dosya,en tepeye gidersek vmlinuz dosyası
- vmlinuz.old : sistemde birden çok çekirdek olabilir mesela güncelledik çalışmadı ,bunla eskiye dönülür
- linux çekirdeği modüller bir çekirdektir çok küçük bir ana çekirdek var vmlinuz driverler,dosya sistemleri ... ise modüldür
- ls lib/modules : modüller burda,çekirdek için olan klasör ve dizinler örneğin içinde kerne/drives/net/wireless var içinde ko uzantılı şeyler var (kernel object dosyaları deniyor/driverler var)
- /etc/modprobe.d : dağıtım kurulduğunda bunların listesini çıkartır
- blacklist.conf : bazı modüllerin sistemin açılmasını engeller/durdurmak için kara liste
- modprobe : bir modülü çekirdeğe ekler (netten indirdiğin bir driver)
- insmod ____.ko : direk ekler (insmod dependencylere bakmaz,denemk için yapılır,makina restart edince gider,modprobede gider)
- kalıcılık için conf dosyasına yazılmalı
- kaldırmak için rmmod
- bir modül yüklerken parametre verebilirsin insmod xyz.ko x=5
- birşey kurarken önce vmlinuz okunur sonra gerekli modülleri yükler ve çalıştırır
- ramdisk : temel ve gerekli modüller buraya atılmalı en tepeye gidip initrd.img dosyasıdır
- bootloader : vmlinuz'u yükler,öncekine dönmeyi çalıştırma
- makina açılınca önce bios açılır sonra bios bootloaderi çalıştırır,o da hangi os içindeyse o os'u başlatır
- eskiden lilo vardı ama artık kullanılmıyor
- artık grub var : menü veriyor hangi os'u çalıştırmak istiyorsun diye
- grub linuxtan önce çalışır yani grub br bootloader yazılımı



- /boot/grub : açılış dosyaları var grub.cfg : grub ile ilgili şeyler yazar
- makina başlarken e'ye basınca grub ile ilgili menü çıkar
- grub'ta değişiklik yaparsam grub-install ile bios'un baktığı yere koyup tekrar yüklemek lazım
- more /etc/grub.d : birşey ekleme kısmı
- makina single user mode ile açtık fsck /dev/sda1 yapabiliriz bu modda sadece boot'u mount eder
- chroot : mesela makina recovery ya da flashtan açtım ,root var etc filan yok,bir paket sağlam mı bakamam,gerçek disk değil (bu bir sistem live cd ile iş yapmak için)
- mesela mount dev/sda2 /mnt : diski mnt'ye mount et chroot /mnt dersem root olarak mnt'yi görür
- dpkg apt dersem hep gerçek sistemdekini yap demek (jailing ya da jaled demek(hapse sokma))
- istediğin servisi root olarak gösterilen yerde hapis edebilirsin daha üstünü göremez
- ama hapis bölgesine herhangi bir kitaplık kullanılıyorsa konması lazım
- recovery mode : güvenli mod gibi
- netten aldık tar.gz açtık,içindeki make'i açtık,make file'i okuyup içindekileri nasıl derleyeceğine bakar sonucunda bir tane xxx.ko dosyası oluşturur

sisteme kurmak için modprobe ile yaptık .ko yu lib/modules/4.4.xxx'e koy dedi , kopyaladık

- install aslında copy ama izin verebiliyoruz sahibi ve grubu ile ilgili

install -o root -g root -m 755 ile kopyaladık(elde ve chmod ile de kopyalanır)

- modülü koyduk ama os farkında değil
- depmod -a : tüm modüllerin listesi
- hem os yeni modülleri buldu hem de bağılıklarını çıkartır
- sinbra modprobe 88xx filan deyince yükledi
- kalıcı olması için:
- etc/modules-load.d altında modules.conf var bu da etc altında modules linkiymiş.vi ile aç bu etc modules dosyasını yeni satıra en alta 88xx i yazıcaz
- 88x2be yazınca sistem bu ko'yu lib/modules altında bulacak aslında çünkü depmode yazınca buldurtmuştuk
- ama çekirdek update sudo apt upgrade deyip linux img güncellenirse yani yeni güncellemelerde bunu tekrar yapmak gerekecek

eğer driver da yoksa tabi

- sistemdeki servisler nasıl başlatılır



2 yöntem var

1. sudo service network-manager stop / start /restart istediğini yap

cache temizliği vr recycle reload var linuxta ubuntu hala kullanıyor

- fedora systemctl ile yapıyor(ubuntuda destekliyor)
- sudo systemctl start ntp
- ls /etc/systemd : bazı servisler var

-açılıştaki hangileri çalışıp çalışmayacak:

- her servisin /etc/init/d altında scriptleri var
- unixte runleveller var

0 - kapalı demek 1 - single user mod 2 - kullanmıyoruz 3 - normal seviye 5 - grafik ekran 6 - makineyi reboot et

- init 3 : 3 numaraya geç
- sistemin defaultunun runleveli ya 3tür ya 5tir
- eskiden /etc/inittab'da yazıyormuş
- more /etc/init/rc-sysinit.conf da default runleveller yazar (içini açınca çekirdeğe parametreyi cmdline ile verebilirsin der)

cd /boot/grub/grub.cfg den bakarız.

- /etc/default : mesela more keyboard klavye dili filan locale sistm dili filan burada yazar buradan karar veririz ona
- runlevel 2 text mod , 3 network açar
- belli bir runlevel a geçmek demek bazı servisleri durdur / başlat demek

peki hangileri durdur / başlata nasıl karar verir :

- /etc altında ls ile bakarsak rc0.d 1.d fila diye her çalışma düzeyi için bir klasör var
- read me den bakabiliriz
- init3 ya da runlevel 3 e geçince readme de servisleri kendi çalıştıracak normalde alfabetik yapar
- yalnız bu servislerin çalışmasını sağlıyor bunlar hep durması için k koymak lazım başına
- k:kill demek
- s yerine k yazmamız lazım
- bir servisin dğrudan 5.seviyede çalışmasını istiyorum : rc5/d ye gidip servis adı filan yazabiliriz
- asıl yöntem: sudo systemctl (servis adı) enable (sayı) : enable eder start istersem yazmam lazım sayı yazmazsam tüm runlevellerde çalışır



- rc.local (dosya) : o makinarya özel olarak yapmak istediğim şeyleri yazdığım dosya
- rc (runtime command) : rc olan herşey sistem açılışına otomatik çalışan dosyalar demek
- fazla özel conf yapmamak gerekir

18 Shutdown

- **shutdown -h now:** hemen kapat
- sadece now ile single user moda düşer
- alt-ctrl-delete graceful yapmak processlere kendini toparlayacak şey bırakmaz ama diski düzgün unmount eder kapar **reboot** ya da **shutdown -r** gibi yapmaz direk kapar
- grafik ekran donarsa:
 - alt-ctrl ile konsola girmeyi dene , top ya da process ile sorunu bul ve öldür
 - olmazsa secureshell ile login olamayı dene
 - alt-ctrl-delete le prosesleri öldür ama diski unmount eder
 - olmazda düğme ile kapat
 - makina çalışır grafik ekran kilitlenirse ctrl-alt-backspace yapın 5.runlevelden atar tekrar başlatır

19 Network

- **host {{site adı}}:** makinanın ip adresi
- **host {{ip adresi}}:** makina adı
- **/etc/hostname:** makinanın adı yazar
- etc\$ more hostname yazınca o dosyaya bakar
- **0.0.0.0 xxx.com** yazınca erişilmesini istenmeyen siteler bağlanamaz(etc\$ more hosts dosyasına)
- unixte servisin adı **named** paketin adı **bind** en çok kullanılan dns server
- local makinada name server etc'nin altında resolve.conf içinde yazar
- etc\$ more resolv.conf'ta yazar(dhcp yoksa)
- etc\$ cd bind/ : dns server kısmı için
- **named.conf:** ilgili dosyalara buradan bakılır
- **ipconfig -a:** (a:all)
- **ipconfig {{aygıt adı}}:** o aygıt ile ilgili bilgi



- ip set etmek: `sudo ifconfig (aygıt adı) (ip adresi)` : linuxta bu komut hem ip bulma hemde ip set etme için kullanılır, cihazın yanına ip yazmazsak ip sorgulanır, yazarsak o karta ip adresi set etmiş oluruz
 - makinaya ip vermek yetmez ama netmask'ta verilir
 - `sudo ifconfig (makina adı) 192.168.1.11 netmask 255.255.255.0 up`
 - netmaskta uzun yazmak yerine `/24` gibi bilgilerde yazılabilir
 - `sudo ifup (aygıt adı)` : up olur down olur (böyle yazınca ip kaybolur her seferinde tekrar yazmak gerekir)
 - `source /etc/network/interfaces.d/*` dosyasına tanımlanan lazım/ashında dağıtımı kurarken ip soruyor o yüzden bu dosyaya pek dokunmayız
 - `route` : kendi ip adrsimi tanımlayabilirim
 - bunu set temek yerine `netstat -nr` kullanılır
 - değiştirmek için : `sudo route add -net 160.75.27.0 netmask 255.255.255.0` ise bunu `eth1`'e yolla demek
 - `sudo route add default gw (ip)` : tim paketler için
 - bunlar sunucularda çalışır, clientlar için NetworkManager diye bir paket var (bir deamon), sistem tray zaten varsa onu kullanırız
 - `ipcalc` : bir tool ,hesaplama yapar
 - `ethtule` : birşey diagnos edeceksen ya da bakacaksan kullanılır
 - `sudo tcpdump` : bizdeki wireshark gibi
 - kullanılan paket : `iptables`
 - input/output/forward ayarları vardır
 - `iptables -L` ie bakabiliriz
 - `sudo iptables -A INPUT -i eth0 -d 10.1.1.1 -p tcp --dport 443 -j accept` gelen hiç bir pakete izin verme, 10.1.1.1 den ve 443'ten gelen isteklere izin verme
- (-d : karşı tarafın adresi -p : tcp /udp protokolü -j : yapılacak politika)
- `echo iptables -A FORWARD -d 165.21.1.2 -p tcp --dport 22 -j ACCEPT` -A : append kural ekle hangi chain'e forward yönlendirme kuralı -d : destonatoin bu paket bu ipdeki makinaya gidiyorsa protokolü buysa izin ver
 - `more /et/seervices` : hangi servis hangi portta çalışacak
 - `telnet` : servisle konuşma / artık `secureshell` kullanıyoruz
 - port yerine servis adını yazarakta bağlanabilirim



20 Servisler

NOT : NIS(eski adı : yellow pages) : bazı sistem dosyaları bu şekilde paylaştırılabilir udp kullanıldığı için güvenlik yok

- ssh -X : makinanın grafik olarak çizmek istediği herşeyi benim makinama aktar demek (rdp istiyorsan vnc kullanmalısın)
- ssh -L : port forwarding yapabilirsin
- ssh -L 8000:localhost:9000 uzaktaki makinanın portunu bu makinanın portuna yönlendir
- bir http proxy kullanmak iyi add blocking engellemek için en çok kullanılan 2 paket var privoxy privacy içi yapılan proxy polipo iyi bir cachingm proxy'dir
- NFS : dosya paylaşımı için / udp üstünden çalışıyormuş , biraz paket kaybı olabilir
- samba : home directory olarak tavsiye edilmez

21 Güvenlik

- PAM(pluggable authenticaion modules)
- /etc/pam.d : paroladan sonra ne kadar beklet,girmeden mesaj göster gibi birçok şey bu dosyada mevcut
- pam no login : root hariç kimse giremesin
- etc/security - limits.conf : bir kullanıcını aynı anda kaç dosya açabilir
- selinux : ekstra koruma paketi
- ubuntu apparmor : koruma paketi

22 ACL

- kullanabilmek için dosya sistemi desteklemelidir
 - ext3-4/reiserfs/xfs/jfs/btrfs : destekliyor
 - network sistem için nfsv4 popüler
- makina kurulunca "mount -o acl" ile acl açılır
- getfacl xx2.ppt : kimin ne yetkisi var
 - setfacl : değiştirmek için
- zfs : iyi bir dosya sistemi
- reelase : (19 mart 2018) iyi bir proje
 - network kullanılacağı için kerberos kullanmamız gerekir



23 LOG

- (more) lastlog : kim en son ne zaman girmiş
- dmesg : sistem mesajlarını göster
 - sistem açılışından itibaren olan şeyleri gösterir
- /var/log : syslog ve errorlog burada
- rsyslog.conf : sistem loglarının konfigürasyonu
- logrotate : bir paket/servis , güncel olarak yazar syslog'y
 - 2 gün öncekileri otomatik sıkıştırarak

24 Script Komutları

- **cut -d:** üst üstelere bak
- **awk -F:** üst üstelere bak/yaz (daha gelişmiş) '{print \$1 " "\$6}' /etc/passwd
- **wk -F:** aynısı,koşul koyabiliriz '\$3.100 100 den büyük olanları demek
- **sed "s/var/opt:** var'ları opt yap (/ yerine @ kullanılabilir)
- **g:** aynı satırda fazla varsa hepsini değiştirir yazmazsak sadece ilkini
- **find:** -exec rm dersek sonun arattığımız dosyaları sil
 - find __ / -name ".css -exec rm
- **grep:** dosyanın içinde arama yapma
 - grep hep text bulur,binary ise bulamaz
 - string /bin/ls | grep usage : stringleri çıkartır binarylerden
 - grep -i: büyük-küçük ayırma yapma
 - grep -r
 - grep -v : tersi
 - grep -l : sadece dosya adını göster
 - zgrep : sıkıştırılmış dosyalar için (bzgrep,bz2grep,zgrep...)
- **regexe:** daha karmaşık aramalar için
- cat /etc/passd | cut -d: -f1 | grep "m\$" : sonu m ile biten satırlar
- ^*m\$* : m ile başlayıp m ile biten (egrep + ile kullanırsak arada en az bir harf olmalı)
- ^m : m ile başlayan satırlar



- `for f in *.png; do echo $f; done` : tüm pngleri ekrana dök
- `for f in *.png; do echo convert $f ${f/.png}.jpg; done`
 - echo silinirse orijinalin üstüne yazmadan yeni jpgler oluşturdu

25 Yazıcı

- **lp q:** yazıcı kuyruğuna bakınız
- **lpq:** tüm printerler
- **lpr:** hangi yazıcıya göndereceğini yazarsın
- **lpr {{işin numarası}}:** kuyruktan çıkar
- printer servisinin adı cups:
 - **sudo service cups restart:** o servisi restart eder
- `cd /etc/cups` : ayarları burda
- Türkçe karakter olayı için (pdf değilse)
 - `enscript` : TR bilmiyor
 - `a2ps` : TR biliyor ama utf, 8859 filan bilmiyor
 - `paps` : en uygunu
 - `paps (dosya adı) | lpr`
- Encoding etme:

```
1 iconv -f iso8859-9 -t utf-8
```
- **unix2dos/dos2unix:** satır sonu ifadeleri yok eder