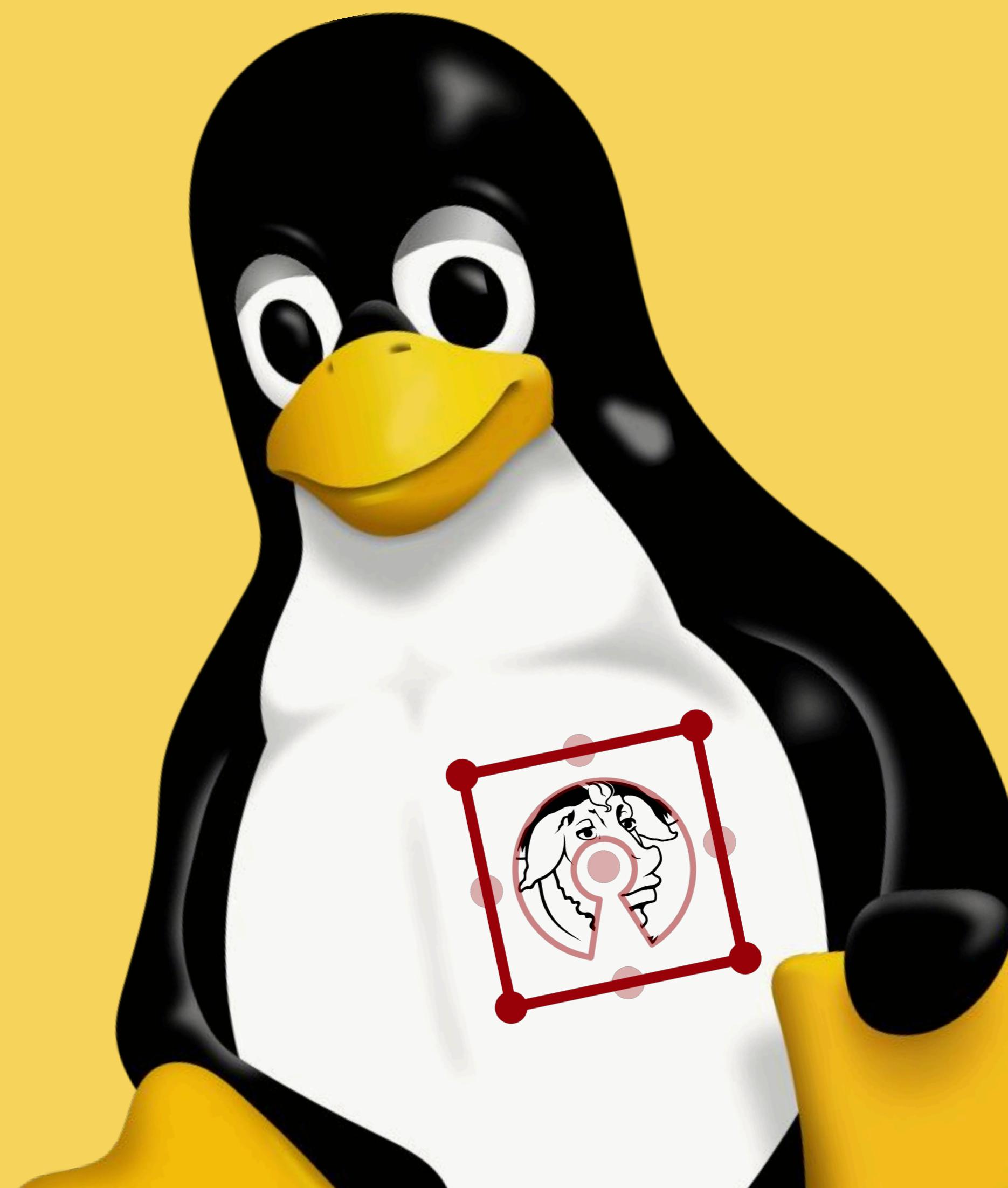


LINUX 101

DOKÜMANTASYON

TAMAMI

Burak Kılınç





LINUX NEDİR?

İşletim sistemi nedir?

İşletim sistemi, bilgisayar donanımı ile kullanıcı arasında bir köprü görevi gören yazılımdır. Bilgisayarın donanım kaynaklarını yönetir ve kullanıcıların bu kaynakları verimli bir şekilde kullanmasını sağlar. İşletim sistemleri, kullanıcıların uygulama programlarını çalıştırmasına, dosya ve dizinleri yönetmesine, ağ bağlantılarını kontrol etmesine ve daha birçok işlemi gerçekleştirmesine olanak tanır.

Linux Nedir?

Linux, Unix benzeri, açık kaynaklı ve özgür bir işletim sistemi çekirdeğidir. İlk olarak 1991 yılında Linus Torvalds tarafından geliştirilmeye başlanmıştır ve zamanla dünya genelinde birçok geliştirici tarafından katkı sağlanarak geliştirilmiştir. Linux, özellikle sunucular, süper bilgisayarlar, gömülü sistemler ve masaüstü bilgisayarlarda yaygın olarak kullanılır.

Linux Tarihçesi ve Gelişimi

Linux, 1991 yılında Linus Torvalds'ın Helsinki Üniversitesi'nde bilgisayar bilimi öğrencisiyken başlattığı bir proje olarak ortaya çıktı. Torvalds, Minix adlı bir eğitim amaçlı işletim sisteminden esinlenerek kendi çekirdek yazılımını geliştirdi. İlk Linux sürümü 0.01, 1991 yılında yayınlandı. Zamanla, dünya genelinde birçok geliştirici Linux çekirdeğine katkı sağladı ve Linux, güçlü, esnek ve güvenilir bir işletim sistemi çekirdeği haline geldi.

Dağıtım Nedir?

Linux dağıtımı (distro), belirli bir işlevsellik ve kullanıcı deneyimi sağlamak üzere özelleştirilmiş, Linux çekirdeği üzerine inşa edilmiş bir işletim sistemi paketidir. Dağıtımlar, çekirdek Linux bileşenlerinin yanı sıra çeşitli uygulama yazılımları, araçlar ve paket yönetim sistemleri içerir. Her dağıtım, belirli kullanıcı ihtiyaçlarına ve kullanım senaryolarına göre yapılandırılmıştır.

**DAĞITIMLARA NEDEN İHTİYACIMIZ VAR?
SONRAKİ SAYFA →**



LINUX DAĞITIMLARI

Neden ihtiyacımız var?

Linux dağıtımlarına, kullanıcılarına özgürlük, esneklik, güvenlik, yüksek performans ve kararlılık sundukları için ihtiyaç duyuyoruz. Unutmayın Linux, tek başına bir işletim sistemi değildir. GNU Projesi'nin sağladığı araçlar ve yazılımlar ile birleşerek tam bir işletim sistemi haline gelir. Bu kombinasyon, kullanıcıların sistemlerini ihtiyaçlarına göre özelleştirmelerine olanak tanır ve geniş topluluk desteği ile sürekli gelişim sağlar.

Neleri barındırır?

Linux dağıtımları, kullanıcılarına tam bir işletim sistemi deneyimi sunmak için çeşitli bileşenler içerir. Bu bileşenler arasında Linux çekirdeği (kernel), GNU araçları ve yazılımları, paket yöneticisi, masaüstü ortamları, uygulamalar ve güvenlik araçları bulunur. Çekirdek, donanım ile yazılım arasındaki iletişimini sağlar. GNU araçları, kabuk ve sistem araçlarını içerir. Paket yöneticileri yazılımları yükleyip güncellemeye yarar. Masaüstü ortamları kullanıcı arayüzü sağlar. Uygulamalar, günlük ihtiyaçları karşılar. Güvenlik araçları ise sistemi korur. Bu bileşenler, bir Linux dağıtımının kullanıcılarına eksiksiz bir işletim sistemi sunmasını sağlar. Linux dağıtımları amacına göre farklı paketleri barındırabilir veya barındırmayabilir.

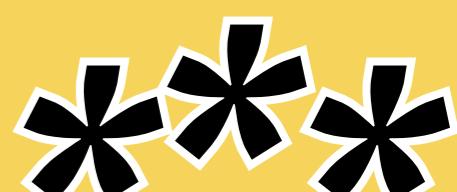
Yaygın dağıtımlar

Yaygın Linux dağıtımları arasında Ubuntu, kullanıcı dostu arayüzü ve geniş yazılım desteği ile öne çıkar. Fedora, en yeni teknolojileri ve güçlü güvenlik özelliklerini sunar. Debian, kararlılığı ve geniş paket arşiviyle bilinir. Arch Linux, ileri düzey kullanıcılar için özelleştirilebilir yapısıyla dikkat çeker. openSUSE, güçlü güvenlik özellikleri ve profesyonel kullanıcılar için uygun yapısıyla öne çıkar. Linux Mint, kullanıcı dostu arayüzü ve Ubuntu tabanlı yapısıyla popülerdir.

ŞİMDİ Temel LINUX Komutları!!!
SONRAKİ SAYFA →



Temel LINUX Komutları



- **ls** komutu, bir dizinin içeriğini listelemek için kullanılır.

```
linux101@oyak:~$ ls
Belgeler Masaüstü İndirilenler Müzik Resimler
```

- **cd** komutu, mevcut çalışma dizinini değiştirmek için kullanılır.

```
linux101@oyak:~$ cd belgeler
linux101@oyak:~/belgeler$ cd ..
linux101@oyak:~$
```

- **pwd** komutu, mevcut çalışma dizinini görüntüler.

```
linux101@oyak:~$ pwd
/home/linux101
```

- **mkdir** komutu, yeni bir dizin oluşturmak için kullanılır.

```
linux101@oyak:~$ mkdir yeni_dizin
linux101@oyak:~$ ls
Belgeler Masaüstü İndirilenler Müzik Resimler yeni_dizin
```

- **rm** komutu, dosya veya dizin silmek için kullanılır.

```
linux101@oyak:~$ rm dosya.txt
linux101@oyak:~$ rm -r eski_dizin
```



Temel LINUX Komutları

- cp komutu, dosya veya dizin kopyalamak için kullanılır.

```
linux101@oyak:~$ cp dosya.txt yeni_dosya.txt  
linux101@oyak:~$ cp -r belgeler yedek_bolgeler
```

- mv komutu, taşımak veya yeniden adlandırmak için kullanılır.

```
linux101@oyak:~$ mv dosya.txt yeni_ad.txt  
linux101@oyak:~$ mv eski_dizin yeni_dizin
```

- touch komutu, yeni bir boş dosya oluşturmak için kullanılır.

```
linux101@oyak:~$ touch yeni_dosya.txt  
linux101@oyak:~$ ls  
Belgeler Masaüstü İndirilenler Müzik Resimler yeni_dosya.txt
```

- find komutu, dosya ve dizinleri aramak için kullanılır.

```
linux101@oyak:~$ find . -name "dosya.txt"  
. ./belgeler/dosya.txt
```

- man komutu, diğer komutlar hakkında detaylı bilgi sağlar.

```
linux101@oyak:~$ man ls  
[ls komutunun kullanım kılavuzu]
```



LINUX DOSYA SİSTEMİ

```
linux101@oyak:~$ tree -L 1 /
```

```
/  
├── bin      → Temel komut dosyaları.  
├── boot    → Önyükleyici dosyaları.  
├── dev      → Donanım aygıt dosyaları.  
├── etc      → Sistem yapılandırma dosyaları.  
├── home    → Kullanıcı ev dizinleri.  
├── lib      → Paylaşılan kütüphaneler.  
├── mnt      → Geçici bağlama noktası.  
├── opt      → Opsiyonel yazılım paketleri.  
├── proc    → Sistem ve süreç bilgisi.  
├── root     → Root kullanıcısının ev dizini.  
├── run      → Çalışma zamanı durumu dosyaları.  
├── sbin     → Sistem yönetim komutları.  
├── srv      → Sunucu hizmet verileri.  
├── sys      → Sistem donanım bilgisi.  
└── tmp      → Geçici dosyalar.  
├── usr      → Kullanıcı uygulama verileri.  
└── var      → Değişken veri dosyaları.
```



LINUX DOSYA SİSTEMİ

1. / (Root) :

Tüm dosya sistemi hiyerarşisinin kök dizini.

- Her dosya ve dizin kök dizinden başlar.
- Yalnızca root kullanıcısının bu dizin altında yazma hakkı vardır.
- /root, root kullanıcısının ana dizinidir ve / ile aynı değildir.

2. /bin :

Temel komutların ikili dosyaları; örneğin cat, ls, cp.

- İkili çalıştırılabilir dosyalar içerir.
- Yaygın Linux komutları bu dizin altındadır.
- Tüm sistem kullanıcıları tarafından kullanılan komutlar burada bulunur, örneğin ps, ls, ping, grep, cp.

3. /boot :

Önyükleyici (boot manager) dosyaları, örneğin çekirdekler, initrd.

- Çekirdek initrd, vmlinuz, grub dosyaları /boot altında bulunur.
- Örnek: initrd.img-2.6.32-24-generic, vmlinuz-2.6.32-24-generic.

4. /dev :

Gerekli aygıt dosyaları, örneğin /dev/null

- Bunlar terminal cihazlarını, usb veya sisteme bağlı herhangi bir cihazı içerir.
- Örnek: /dev/tty1, /dev/usbmon0.

5. /etc :

Ana bilgisayara özgü sistem genelinde yapılandırma dosyaları.

- Tüm programlar tarafından gereksinim duyulan yapılandırma dosyalarını içerir.
- Ayrıca, bireysel programları başlatmak/durdurmak için kullanılan başlangıç ve kapanış betikleri de içerir.
- Örnek: /etc/resolv.conf, /etc/logrotate.conf.

6. /home :

Kullanıcıların ana dizinleri, kaydedilmiş dosyalar, kişisel ayarlar vb.

- Tüm kullanıcıların kişisel dosyalarını saklayabilecekleri ana dizinler. Örnek: /home/linux101, /home/burak.



LINUX DOSYA SİSTEMİ

7. /lib :

/bin/ ve /sbin/ içindeki ikili dosyalar için gerekli kütüphaneler.

- Kütüphane dosya adları ya ld* ya da lib*.so.* şeklindedir.
- Örnek: ld-2.11.1.so, libncurses.so.5.7.

8. /media :

Cıkarılabilir medya için bağlantı noktaları, örneğin CD-ROM'lar (FHS-2.3'te ortaya çıkmıştır).

- Cıkarılabilir aygıtlar için geçici bağlantı dizini.
- Örnekler: CD-ROM için /media/cdrom; disket sürücü için /media/floppy; CD yazıcı için /media/cdrecorder.

9. /mnt :

Geçici olarak monte edilmiş dosya sistemleri.

- Sistem yöneticilerinin dosya sistemlerini monte edebileceği geçici bağlantı dizini.

10. /opt :

Opsiyonel uygulama yazılımı paketleri

- Bireysel sağlayıcılarından gelen ek uygulamaları içerir.
- Ek uygulamalar /opt/ veya /opt/ alt dizininde kurulmalıdır.

11. /sbin :

Gerekli sistem ikili dosyaları, örneğin fsck, init, route.

- /bin gibi, /sbin de ikili çalıştırılabilir dosyalar içerir.
- Bu dizin altında bulunan Linux komutları, genellikle sistem yöneticileri tarafından sistem bakım amaçlarıyla kullanılır.
- Örnek: iptables, reboot, fdisk, ifconfig, swapon.

12. /srv :

Sistemin sunduğu hizmetlerle ilgili verileri depolamak için kullanılır. Bu klasörde, genellikle web sunucuları, FTP sunucuları ve diğer servislerle ilgili veriler bulunur.

- Sunucuya özgü hizmetlerle ilgili verileri içerir.
- Örnek: /srv/cvs, CVS ile ilgili verileri içerir.



LINUX DOSYA SİSTEMİ

13. /tmp :

Geçici dosyalar. Genellikle sistem yeniden başlatıldığında korunmaz ve ciddi boyut sınırlamaları olabilir.

- Sistem ve kullanıcılar tarafından oluşturulan geçici dosyaları içeren dizin.
- Bu dizin altındaki dosyalar sistem yeniden başlatıldığında silinir.

14. /usr :

Salt okunur kullanıcı verileri için ikincil hiyerarşi, çoğu kullanıcı yardımcı programı ve uygulamasını içerir.

- Kullanıcı programları için ikili dosyalar, kütüphaneler, dokümantasyon ve kaynak kod içerir.
- /usr/bin, kullanıcı programları için ikili dosyalar içerir (örneğin: at, awk, cc, less, scp).
- /usr/sbin, sistem yöneticileri için ikili dosyalar içerir (örneğin: atd, cron, sshd, useradd, userdel).
- /usr/lib, /usr/bin ve /usr/sbin için kütüphaneler içerir.
- /usr/local, kaynak koddan derlenen programları içerir.
- /usr/src, Linux çekirdek kaynaklarını ve dokümantasyonu barındırır.

15. /proc:

İşlem ve çekirdek bilgilerini dosya olarak sağlayan sanal dosya sistemi.

- Genellikle sistem tarafından otomatik olarak oluşturulur ve doldurulur.
- Sistem işlemleri hakkında bilgi içerir.
- Bu, çalışan işlemler hakkında bilgi içeren sahte bir dosya sistemidir. Örneğin: /proc/{pid} dizini, belirli pid'ye sahip işlem hakkında bilgi içerir.
- Bu, sistem kaynakları hakkında metin bilgisi içeren sanal bir dosya sistemidir.
- Örneğin: /proc/uptime.

```
linux101@oyak:~$ echo $oyaktan_mesaj
```

Linux Dosya Hiyerarşi Yapısı (FHS), Unix benzeri işletim sistemlerinde dizin yapısını ve içeriğini tanımlar. Linux Vakfı tarafından yönetilen FHS, tüm dosyaların ve dizinlerin kök dizin (/) altında görünmesini sağlar. (fiziksel veya sanal depolama konumlarından bağımsız olarak.) Bu hiyerarşideki birçok dizin çeşitli Unix sistemlerinde ortak olsa da, belirli kullanımları değişimdir ve burada verilen açıklamalar Linux'a özeldir.



DOSYA SİSTEMİ YOLLARI

X X X X X

Mutlak Yol

Kök (root) dizininden başlayarak dosya veya dizine giden tam yoldur.

- Özellikler:
 - Her zaman / ile başlar.
 - Dosya sisteminin en üst seviyesinden başlar ve hedef dosya veya dizine kadar olan tüm dizinleri içerir.

Örnekler:

```
linux101@oyak:~$ cd /home/linux101/Documents
linux101@oyak:~/Documents$ ls /usr/bin
```

Bağlı Yol

Geçerli dizinden başlayarak dosya veya dizine giden yoldur.

- Özellikler:
 - Geçerli çalışma dizinine göre belirlenir.
 - .: geçerli dizini, ..: bir üst dizini ifade eder.

Örnekler:

```
linux101@oyak:~/Documents$ cd .. /Downloads
linux101@oyak:~/Documents$ ls ./scripts
```

Wildcard

Dosya veya dizin adı desenleri ile eşleşmek için kullanılan özel karakterlerdir.

- Özellikler:
 - * : Sıfır veya daha fazla karakterle eşleşir.
 - ? : Tek bir karakterle eşleşir.
 - [] : Belirtilen karakterlerden biriyle eşleşir.

Örnekler:

```
linux101@oyak:~$ ls *.txt      # Tüm .txt dosyalarını listeler
linux101@oyak:~$ ls file?.txt  # file1.txt, fileA.txt gibi dosyaları listeler
linux101@oyak:~$ ls file[0-9].txt # file1.txt, file2.txt gibi dosyaları listeler
```



GÜNÜN SONU

`linux101@oyak:~$ echo $buraktan_mesaj`

Bu dokümantasyon, ÖYAK Linux 101 eğitimiminin ilk gününün bir özeti niteliğindedir. Eksiklikler ve fazlalıklar olabilir. Tasarım dili olarak mükemmel bir hizalama yapamayacağımdan korktuğum için aykırı bir dil tercih ettim; bu sayede herhangi bir hizalama yapmama gerek kalmadı. Buraya kadar okuduysanız gerçekten bir şeyler öğrenmek istiyor olabilirsiniz. Lütfen bizimle alacağınız 4 günlük kursla yetinmeyin. Devamını getirmek sizin elinizde. Linux'u sadece bir işletim sistemi olarak görmeyin. Linux aynı zamanda bir kariyer yolу, hatta kariyerinizin kendisi olabilir. Bunların hiçbiri değilse bile kullanması çok zevkli, size özel bir işletim sistemi olur. Lütfen dokümantasyona sadece göz gezdirmeyin, İNCELEYİN!

HAP BİLGİ

POSIX (Portable Operating System Interface), Unix benzeri işletim sistemleri arasında taşınabilirliği ve uyumluluğu sağlamak amacıyla geliştirilmiş bir standartlar setidir. 1980'lerin ortasında IEEE tarafından geliştirilen POSIX, dosya sistemlerinin temel yapısını, temel Unix komutlarını, shell yapısını ve API standartlarını belirler. Linux ve diğer Unix benzeri işletim sistemleri, POSIX standartlarına uyum sağlayarak geliştirilmiştir, bu da uyumluluğu ve taşınabilirliği artırır.

POSIX'in Temel Bileşenleri

- Dosya Sistemleri:** Dosya sisteminin temel yapısını belirler.
- Komutlar ve Shell:** Temel Unix komutlarını ve shell yapısını tanımlar.
- Programlama ve API'ler:** Sistem çağrıları ve API'ler için standartlar belirler.

LINUX 101

Eğitim Hedefleri:

- İşletim sistemi kavramını ve Linux'un ne olduğunu anlamak
- Linux dağıtımlarını ve dağıtım seçiminin önemini öğrenmek
- Linux dosya sistemi ve dizini yapısını kavramak
- Temel Linux komutlarını öğrenmek ve uygulayabilmek
- Paket yöneticisi sistemini ve önemini kavramak

Yukarıdaki maddeler ilk gün hedeflerini belirtir.



XXX

**ÖZGÜR
YAZILIM
KAVRAMI**

**LINUX
DOĞUYOR**



linux101@oyak:~\$ cat icindekiler.txt

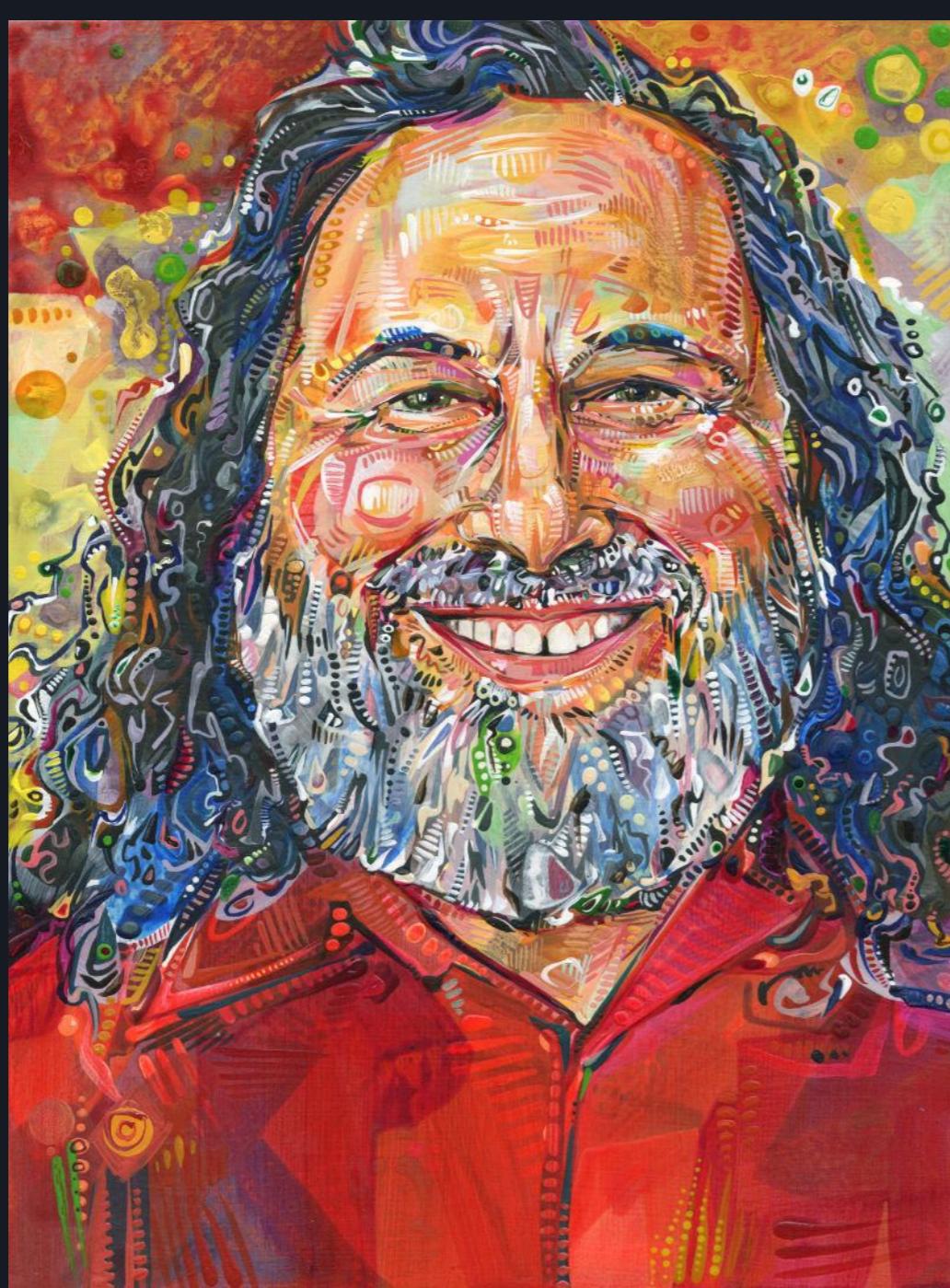
- 1. Özgür Yazılım Kavramının Doğuşu
- 2. Richard Stallman ve Özgür Yazılım Hareketi
- 3. GNU/Linux'un Doğuşu
- 4. Özgür ve Açık Kaynak Lisansları
- 5. Açık Kaynak Hareketi
- 6. Linux'un Yükselişi
- 7. Günümüzde Linux ve Özgür Yazılım

Özgür Yazılım'ın Doğuşu

Özgür yazılım kavramı, 1970'lerin sonlarından 1980'lerin başlarına uzanan bir süreçte, bilgisayar yazılımlarının giderek kapalı ve ticari ürünlere dönüşmesine bir tepki olarak doğmuştur. Bilgisayarların ilk yıllarda yazılımlar genellikle donanımın bir parçası olarak görülmüyordu, kaynak koduyla birlikte paylaşılıyordu ve kullanıcılar tarafından serbestçe değiştirilebiliyordu. Ancak 1970'lerin ortalarından itibaren yazılım şirketleri, rekabet avantajını korumak amacıyla kaynak kodlarını gizlemeye, kullanıcı özgürlüklerini kısıtlayan lisans anlaşmaları uygulamaya ve yazılımları bağımsız ticari ürünler olarak satmaya başladılar. Bu dönüşüm, özellikle AT&T'nin UNIX'i ticarileştirmesiyle hız kazandı ve kullanıcıların yazılımları inceleme, değiştirme ve yeniden dağıtmayı yeteneklerini büyük ölçüde kısıtladı. Yazılım dünyasındaki bu kapalılık eğilimi, sonunda Richard Stallman gibi programcılardan özgür yazılım hareketini başlatması için temel motivasyonu oluşturdu.



Richard Stallman



Richard Stallman, özgür yazılım hareketinin kurucusu ve GNU Projesi'nin yaratıcısıdır. 1983'te başlattığı mücadele ile yazılım dünyasında devrim yaratan Stallman, kullanıcı özgürlüklerini savunmak için hayatını adadı. Özgür Yazılım Vakfı'ni kuran ve GPL lisansını geliştiren bu idealist programcı, teknoloji dünyasında "özgürlük savaşçısı" olarak anılır. Entelektüel, kararlı ve uzlaşmaz kişiliğiyle bilinen Stallman, günümüzde kullandığımız birçok özgür yazılımın fikir babasıdır.

Bu içerik özgür yazılım felsefesine odaklanmaktadır, hiçbir kişinin tartışmalı görüş ve davranışlarını desteklememektedir. Özgür yazılım fikirlerini desteklerken, toplulukta ayrımcı ya da zararlı görüşleri onaylamadığımızı belirtmek isteriz.

Printer Olayı

1980'lerde MIT'de çalışan Richard Stallman, laboratuvarlarına gelen yeni Xerox yazıcısının sık sık sıkışmasıyla karşılaştı. Eski yazıcılarda bu tür sorunları kaynak kodu değiştirerek çözebilen Stallman, yeni yazıcısının koduna erişemedi. Bir meslektaşının Xerox'tan kodu aldığı öğrense de, kişi imzaladığı gizlilik anlaşması nedeniyle paylaşmayı reddetti. Bu olay, Stallman'ı özgür yazılım hareketini başlatmaya iten kivircım oldu. Stallman insanların yazılımı özgürce kullanma, inceleme ve paylaşma hakkından mahrum bırakıldığını gördü.

Dört Özgürlük

Richard Stallman, özgür yazılım kavramını tanımlarken, "özgür" kelimesinin ücretsiz değil özgürlük anlamına geldiğini vurguladı ("free as in freedom, not free as in beer"). 1983 yılında, özgür yazılımın dört temel özgürlüğünü tanımladı. Bu özgürlüklerin sağlanması, yazılım kullanıcılarının özgür olabilmesi için temel şartlardı. Stallman'a göre, bu özgürlüklere sahip olmayan yazılım kullanıcıları, yazılımın kontrolü altındadır.



Dört Temel Özgürlük

```
linux101@oyak:~$ cat dort-ozgurluk.txt
```

Özgürlük 0: Programı herhangi bir amaç için çalışma özgürlüğü

Özgürlük 1: Programın nasıl çalıştığını inceleme ve kendi ihtiyaçlarınıza uyarlama özgürlüğü (kaynak koda erişim bunun bir ön koşuludur)

Özgürlük 2: Programın kopyalarını dağıtma özgürlüğü

Özgürlük 3: Programı geliştirme ve geliştirilmiş versiyonlarını dağıtma özgürlüğü (kaynak koda erişim bunun bir ön koşuludur)

GNU Projesi

Stallman, özgür yazılım felsefesini hayatı geçirmek için 1983 yılında GNU Projesi'ni başlattı. GNU, "GNU's Not UNIX" ifadesinin özyinelemeli kısaltmasıdır ve UNIX ile uyumlu ancak tamamen özgür bir işletim sistemi geliştirmeyi amaçlıyordu.

Projenin ilk duyurusu, 27 Eylül 1983 tarihinde net.unix-wizards ve net.usoft haber gruplarında yapıldı. Stallman şöyle yazmıştır:

"GNU, özgür olacak ve GNU değerlerine sadık kaldığınız sürece GNU'yu yeniden dağıtip değiştirebileceksiniz. GNU UNIX değil, çünkü GNU, UNIX'e benzer olacak ama UNIX değil."

GNU projesi kapsamında, bir işletim sisteminin ihtiyaç duyduğu temel bileşenler geliştirilmeye başlandı:

- Editör (Emacs)
- Derleyici (GCC - GNU Compiler Collection)
- Kabuk (Bash - Bourne Again SHell)
- ve diğer temel araçlar

Özgür Yazılım Vakfı

Stallman, GNU Projesi'ni desteklemek için 1985 yılında Özgür Yazılım Vakfı'nı (Free Software Foundation - FSF) kurdu. FSF, özgür yazılımın geliştirilmesi, dağıtılması ve desteklenmesi için kurulmuş kar amacı gütmeyen bir kuruluştur. Vakıf, özgür yazılım hareketinin kurumsal temsilini oluşturarak, bağış toplama, yasal destek sağlama ve topluluk oluşturma görevlerini üstlendi.

FSF'nin kuruluşundan bu yana temel amaçları şunlardır:

- Özgür yazılım felsefesini yaymak
- GNU sisteminin geliştirilmesini desteklemek
- Özgür yazılımların yasal korunmasını sağlamak
- Yazılım patentleri gibi özgür yazılıma tehdit oluşturan konularda mücadele etmek



GNU/Linux' un doğusu

1984 yılından itibaren, GNU Projesi özgür bir işletim sistemi oluşturmak için gereken bileşenleri geliştirmeye başladı. Projenin hedefi, UNIX sistemlerle uyumlu çalışabilecek tamamen özgür bir işletim sistemi oluşturmaktı.

1990'a gelindiğinde, GNU Projesi işletim sisteminin birçok bileşenini tamamlamıştı:

- GNU Derleyici Koleksiyonu (GCC)
- GNU Emacs metin editörü
- GNU C Kütüphanesi (glibc)
- GNU Core Utilities (temel sistem komutları)
- GNU Bash kabuğu

Ancak, işletim sisteminin en önemli parçası olan çekirdek (kernel) henüz tamamlanmamıştı. GNU Projesi, Hurd adı verilen bir çekirdek geliştirmeye başlamıştı, ancak bu çalışma beklenenden daha karmaşık olmuş ve tamamlanması gecikiyordu.

Linus Torvalds



Linus Torvalds, Linux çekirdeğinin yaratıcısı ve açık kaynak dünyasının önde gelen figürüdür. 1991'de bir Üniversite öğrencisiyken başlattığı "hobi projesi" ile bilişim dünyasında çığır açan Torvalds, pragmatik yaklaşımı ile teoriden çok pratiğe önem verdi. Linux çekirdeğinin baş geliştiricisi olarak görevini sürdürün bu Finlandiyalı programcı, teknoloji dünyasında "pragmatik devrimci" olarak tanınır. Dobra, zeki ve sonuç odaklı kişiliğiyle bilinen Torvalds, günümüzde internetin, akıllı telefonların ve süper bilgisayarların arkasındaki temel teknolojinin mimarıdır.



Linus ve Linux

1991 yılında, Helsinki Üniversitesi'nde bilgisayar bilimleri öğrencisi olan Linus Torvalds, kendi bilgisayarında çalışabilen basit bir çekirdek geliştirmeye başladı. Torvalds'ın asıl amacı, MINIX (eğitim amaçlı geliştirilen küçük bir UNIX benzeri sistem) üzerinde çalışırken karşılaştığı kısıtlamaları aşmaktı.

25 Ağustos 1991 tarihinde, Torvalds comp.os.minix haber grubuna şu mesajı gönderdi:

"Minix kullanan herkese merhaba! 386/486 bilgisayarlar için (ücretsiz) bir işletim sistemi geliştirdiyorum. Sadece bir hobi projesi, GNU kadar büyük ve profesyonel olmayacak. Nisan ayından beri üzerinde çalışıyorum ve hazır olmaya başlıyorum. Sistem biraz Minix'e benziyor, ancak tamamen Minix kodundan bağımsız ve çok iş parçacıklı bir dosya sistemine sahip. Şu an bash ve gcc'yi sisteme uyarladım ve işler yolunda gidiyor. Birkaç ay içinde kullanılabılır bir şey elde edeceğimi düşünüyorum. Hangi özellikleri istediğinizde dair önerilerinizi bekliyorum, ancak hepsini uygulayacağımı söyleyemem. :-)

Linus"

İsimlendirme ve Duyuru

Torvalds başlangıçta projesine "Freax" adını vermek istemişti (free + freak + UNIX kelimelerinin birleşimi), çünkü kendi adını kullanmanın fazla kibirli görüneceğini düşünüyordu. Ancak FTP sunucusunu yöneten arkadaşı Ari Lemmke, Torvalds'a danışmadan dosya için "Linux" dizini açtı. Başta bu isme karşı çıksa da, Torvalds sonunda bu ismi kabul etti ve proje zamanla Linux olarak tanındı.

Torvalds, çekirdeğin ilk sürümünü (0.01) 17 Eylül 1991'de yayınladı. Bu sürüm henüz kullanışlı bir işletim sistemi değildi, ancak temel dosya sistemine erişim sağlıyor ve terminal işlemlerini destekliyordu.

Linux çekirdeği başlangıçta kendi lisansı altında dağıtılmıyordu, ancak bu lisans yeniden dağıtım hakkını sınırlıyordu. 1992 yılında, Torvalds çekirdeği GPL (GNU Genel Kamu Lisansı) altında yeniden lisansladı. Bu karar, Linux'un özgür yazılım hareketinin önemli bir parçası haline gelmesinde büyük rol oynadı.



GNU/Linux İşbirliği

Linux çekirdeği, GNU Projesi'nin geliştirdiği yazılımlarla bir araya geldiğinde, işlevsel bir işletim sistemi ortaya çıktı. GNU araçları (derleyici, kabuk, temel komutlar vs.) ve Linux çekirdeği birleştirilerek, tam bir işletim sistemi oluşturuldu.

Bu nedenle, Özgür Yazılım Vakfı ve Richard Stallman, bu işletim sisteminin adının "GNU/Linux" olarak kullanılmasını savunmaktadır. Çünkü sistem, GNU projesi tarafından geliştirilen birçok temel bileşen ve Linux çekirdeğinin birleşiminden oluşmaktadır. Ancak pratikte çoğu kullanıcı ve dağıtım, sistemi sadece "Linux" olarak adlandırmaktadır.

Bu işbirliği, özgür yazılım dünyasında önemli bir dönüm noktasıdır. GNU Projesi'nin hedeflediği tamamen özgür işletim sistemi, Linux çekirdeği sayesinde gerçeğe dönüşmüştür.

Özgür Lisanslar

GNU Genel Kamu Lisansı (GPL), Richard Stallman tarafından 1989 yılında özgür yazılımları korumak amacıyla oluşturuldu. GPL, yazılım özgürlüklerini koruma altına alan ve "copyleft" ilkesini ortaya koyan devrim niteliğinde bir lisansdır.

Copyleft kavramı, telif hakkı (copyright) kavramının karşıtı olarak düşünülebilir. Copyleft şu ilkeyi benimser: Bir yazılımın özgür olarak kalabilmesi için, o yazılımdan türetilen tüm çalışmalar da aynı özgürlüklerle lisanslanmalıdır. Yani, GPL ile lisanslanan bir yazılımı değiştiren herkes, değiştirilmiş versiyonu da GPL ile lisanslamak zorundadır. Bu, özgür yazılımların "kapalı" sistemlere dönüşmesini engeller.

GPL'nin temel özellikleri:

- Çalıştırma özgürlüğü: Herhangi bir amaç için yazılımı çalışma hakkı
- İnceleme ve değiştirme özgürlüğü: Kaynak koduna erişim ve değişiklik yapma hakkı
- Yeniden dağıtım özgürlüğü: Yazılımın kopyalarını dağıtabilme hakkı
- İyileştirmeleri paylaşma zorunluluğu: Yapılan değişiklıkların de aynı lisansla dağıtılması zorunluluğu



Diğer Özgür Yazılım Lisansları

GPL dışında, özgür ve açık kaynak yazılımlarda kullanılan çeşitli lisanslar vardır. Bunlardan bazıları:

1. MIT Lisansı: Minimal kısıtlamalar içeren, kullanıcıya neredeyse tam özgürlük tanıyan bir lisans. Yazılımı her türlü projede (ticari projeler dahil) kullanma, değiştirme ve dağıtma özgürlüğü verir. Tek şart, lisans metninin ve telif hakkı bildiriminin korunmasıdır.
2. BSD Lisansları: Özgün BSD lisansı ve türevleri (2-Clause, 3-Clause BSD), MIT lisansına benzer şekilde minimum kısıtlamalarla gelir. Yazılımin kapalı kaynaklı versiyonlarını oluşturmaya izin verir.
3. Apache Lisansı: Açık kaynak projelerinde yaygın olarak kullanılan bir lisansdır. Patent hakları konusunda belirgin hükümler içerir ve yazılımların ticari kullanımına izin verir.
4. LGPL (GNU Lesser General Public License): GPL'nin daha az kısıtlayıcı bir versiyonudur. LGPL ile lisanslanan bir kütüphaneyi kullanan programların kaynak kodunu açmak zorunda olmaması, bu lisansı özellikle kütüphaneler için uygun kılar.

Özgür Yazılım vs Açık Kaynak

Özgür yazılım ve açık kaynak terimleri, çoğunlukla aynı yazılımları ifade etseler de, farklı felsefi yaklaşımları temsil ederler:

Özgür Yazılım Yaklaşımı:

- Etik ve felsefi temellere dayanır
- Kullanıcı özgürlüğünü en üst değer olarak görür
- "Özgürlük bir toplumsal değerdir" anlayışını benimser
- Richard Stallman ve FSF tarafından temsil edilir

Açık Kaynak Yaklaşımı:

- Pragmatik ve teknik avantajlara odaklanır
- Geliştirme modelinin verimliliğini vurgular
- İş dünyasına ve ticari gerçeklere daha uyumludur
- Eric Raymond ve OSI tarafından temsil edilir

Richard Stallman bu iki yaklaşım arasındaki farkı şöyle açıklar: "Açık kaynak, bir geliştirme metodolojisidir; özgür yazılım ise bir sosyal harekettir."



Topluluk Odaklı Geliştirme Modeli

Açık kaynak yazılım geliştirme modeli, geleneksel kapalı kaynak modelinden birçok açıdan farklıdır. Eric Raymond, bu farkı "The Cathedral and the Bazaar" (Katedral ve Çarşı) makalesinde şu şekilde açıklar:

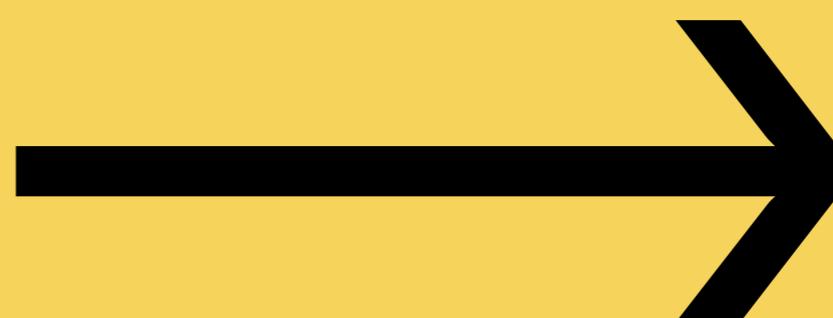
- Katedral Modeli: Geleneksel yazılım geliştirme, az sayıda geliştiricinin kapalı kapılar ardında çalıştığı hiyerarşik bir yapıya sahiptir.
- Çarşı Modeli: Açık kaynak geliştirme, çok sayıda geliştiricinin açık bir ortamda işbirliği yaptığı, daha organik ve dağıtık bir yapıya sahiptir.

Açık kaynak modelin temel ilkeleri şunlardır:

1. Erken ve Sık Sürüm Yayınlama: "Release early, release often" prensibi
2. Hataları Hızlı Tespit Etme: "Given enough eyeballs, all bugs are shallow" (Linus'un Yasası)
3. Kullanıcıların Geliştirici Olması: Kullanıcılar aynı zamanda potansiyel geliştiricilerdir
4. Meritokrasi: Katkılarının değeri, katkı sağlayanın ünvanından değil, katkıının kalitesinden gelir
5. Şeffaflık: Tüm geliştirme süreci açık ve izlenebilirdir

LINUX

YÜKSELIYOR





İlk Linux Dağıtımları

Linux çekirdeği yayınlandıktan sonra, farklı gruplar bu çekirdeği GNU araçlarıyla birleştirerek çeşitli Linux dağıtımları oluşturdu. MCC Interim Linux ve SLS (1992), Slackware ve Debian (1993), Red Hat ve SUSE (1994) gibi ilk önemli dağıtımlar, Linux'un hızla yayılmasını sağladı.

Patrick Volkerding'in 1993'te başlattığı Slackware, günümüzde hala aktif olarak geliştirilen en eski dağıtım olarak istikrar ve sadeliği temsil ediyor. Debian, topluluk odaklı yapısı ve sıkı özgür yazılım ilkeleriyle binlerce paketin uyumlu çalışmasını sağlarken, Red Hat kurumsal dünyada Linux'u standart haline getirdi ve açık kaynak iş modelinin öncüsü oldu. openSUSE (eski SUSE) ise özellikle Avrupa'da güçlü varlığı ve YaST gibi yönetim araçlarıyla teknik olmayan kullanıcılarla Linux'u ulaştırdı. Bu dört ana dağıtım, Linux ekosisteminin temelini oluşturarak işletim sisteminin yaygınlaşmasında kritik rol oynadı.

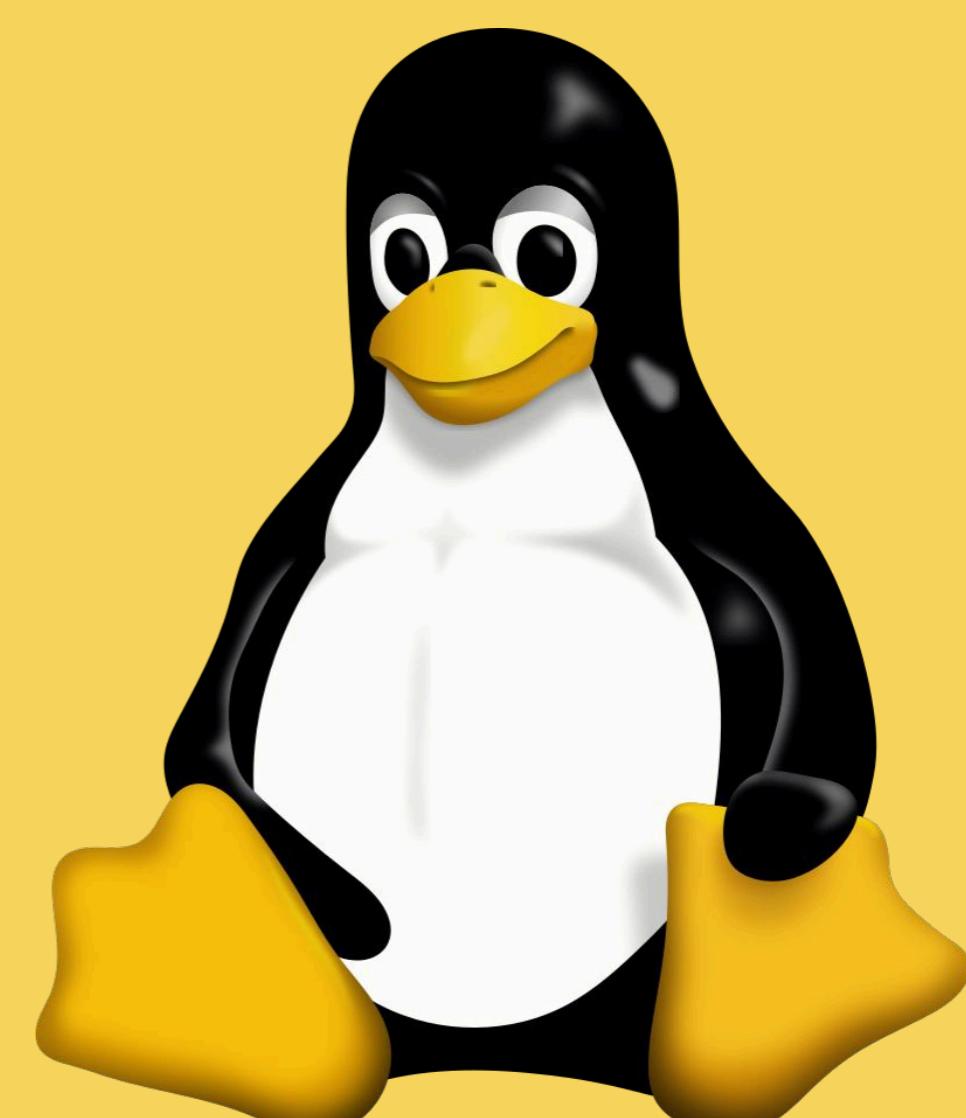
Linux' un Maskotu

Linux' un maskotu olan Tux, bir penguen karakteridir. Tux adı, bazı kaynaklara göre "(T)orvalds (U)ni(X)" kelimelerinin kısaltması olarak, bazı kaynaklara göre ise "tuxedo" (smokin) kelimesinden gelir - penguenlerin smokin giymiş gibi görünen siyah-beyaz deseninden dolayı.

Tux'un hikayesi 1996 yılında başladı. Linus Torvalds'ın anlatımına göre, bir ziyaret ettiği Avustralya'da bir penguen tarafından ısırlıma deneyimi yaşadı (ya da bu hikayeyi esprili bir şekilde uydurdu). Bu olaydan esinlenerek, Linux için bir penguen maskotu fikrini destekledi.

Tux'un orijinal çizimi, Larry Ewing tarafından GIMP programı kullanılarak oluşturuldu ve özgür bir lisans altında yayınlandı. Bu sevimli ve dostane penguen, zamanla Linux' un resmi olmayan sembolü haline geldi ve birçok Linux dağıtımı tarafından benimsendi.

Tux, sadece bir maskot değil, aynı zamanda özgür yazılım kültürünün de bir sembolü olarak kabul edilir. Resmi ve kurumsal logoların aksine, eğlenceli, açık ve topluluk odaklı bir imaj sunar.





Günümüzde Linux

Günümüzde Linux, bilişim dünyasının neredeyse her alanında kullanılmaktadır:

Sunucu Sistemleri:

- Web sunucuları: Dünya üzerindeki web sitelerinin büyük çoğunluğu Linux tabanlı sunucularda çalışmaktadır
- Bulut altyapısı: AWS, Google Cloud, Azure gibi bulut hizmetleri Linux üzerinde çalışır
- Veritabanı sunucuları: MySQL, PostgreSQL, MongoDB gibi veritabanları genellikle Linux sistemlerde kullanılır
- Kurumsal uygulamalar: ERP, CRM gibi kurumsal sistemlerin önemli bir kısmı Linux platformunu tercih eder

Süper Bilgisayarlar:

- Dünyanın en güçlü süper bilgisayarlarının neredeyse tamamı Linux kullanmaktadır

Masaüstü Sistemleri:

- Ubuntu, Fedora, Linux Mint, Elementary OS gibi kullanıcı dostu dağıtımlar
- Grafik tasarım, video düzenleme ve müzik produksiyonu için özelleştirilmiş dağıtımlar (Ubuntu Studio vb.)
- Eğitim kurumları ve kamu kurumlarında kullanım
- Geliştiriciler arasında yaygın kullanım (özellikle web ve açık kaynak geliştirme)

Mobil ve Gömülü Sistemler:

- Android: Dünyanın en yaygın mobil işletim sistemi, Linux çekirdeği üzerine kuruludur
- IoT cihazları: Akıllı ev aletleri, sensörler, ağ cihazları
- Araç bilgi-eğlence sistemleri
- Endüstriyel kontrol sistemleri ve robotlar
- Akıllı TV'ler, medya oynatıcıları ve diğer tüketici elektroniği

Linux'un bu kadar farklı alanlarda kullanılabilmesi, modüler yapısı, ölçeklenebilirliği ve özelleştirilebilirliği sayesindedir. İhtiyaca göre minimal bir sistemden, tam teşekküllü bir masaüstü veya sunucu sisteme kadar farklı konfigürasyonlarda kullanılabilir.



Bizde katkı sağlayalım

Özgür yazılım ekosistemi topluluk katkılarıyla gelişir:

Teknik Katkılar:

- Kod geliştirme (hata düzeltme, yeni özellikler, optimizasyon)
- Dokümantasyon yazımı ve güncellemesi
- Test etme ve hata raporlama

Yerelleştirme ve Tasarım:

- Arayüz ve dokümantasyon çevirileri
- Kullanıcı arayüzü ve deneyimi iyileştirmeleri

Topluluk Desteği:

- Forumlarda kullanıcılaraya yardım ve eğitim içerikleri hazırlama
- Etkinlikler düzenleme veya katılma

Finansal Destek:

- Bağış yapma veya destekleyici ürünleri satın alma

Katkıda Bulunmaya Başlamak İçin:

- İlgiilenilen projeyi seçip dokümantasyonunu okuyun
- Topluluk kanallarına katılın ve basit görevlerle başlayın

SONUÇ

Linux ve özgür yazılım hareketi, bilişim dünyasında devrim niteliğinde bir değişimi temsil eder. Kapalı ve tekelci bir yazılım dünyasından, açık, işbirliğine dayalı ve özgür bir ekosisteme geçiş sa glamıştır.

Bu ekosistemin temelinde yatan felsefe, kullanıcıların bilgisayarları üzerinde tam kontrol sahibi olması, yazılımları inceleyebilmesi, değiştirebilmesi ve paylaşabilmesi prensibine dayanır. Bu prensipler, sadece daha iyi yazılımlar üretilmesini sağlamakla kalmaz, aynı zamanda daha adil, şeffaf ve katılımcı bir dijital dünya yaratılmasına da katkıda bulunur.

Linux, bu felsefenin en başarılı uygulama örneklerinden biridir. Bir hobi projesi olarak başlayan Linux, bugün dünyanın en yaygın işletim sistemlerinden biri haline gelmiş, sunuculardan süper bilgisayarlara, akıllı telefonlardan arabalar ve buzdolaplarına kadar sayısız cihazda çalışır duruma gelmiştir.

Özgür yazılım ve Linux'un hikayesi, bireysel tutkuların ve topluluk işbirliğinin, kurumsal kaynaklarla desteklendiğinde neleri başarabileceğinin ilham verici bir örneğidir.



xxx

DAHA

DERiNE

INME

VAKTİ



```
linux101@oyak:~$ cat konu_basliklari.txt
```

- Wildcards (Joker karakterler)
- Commandline Challenge
- init sistemi: sistemin başlatılması sırasında ilk çalıştırılan süreç
- systemd: servislerin yönetimi
- Servis Logları: journalctl kullanımı
- Daemon: arka planda çalışan işlemler
- Zamanlanmış Görevler: Cronjobs
- Dosya ve Dizin İzinleri
- chmod komutu
- chown ve chgrp
- Özel İzinler: SetUID, SetGID, Sticky Bit
- Kullanıcı Oluşturma ve Silme
- /etc/shadow, /etc/passwd
- Sudoers Yetkileri
- Kullanıcı Bilgilerini Listeleme



Wildcards

Wildcard (joker karakter), Linux terminalinde dosya ve dizin adlarını belirli bir desenle eşleştirmek için kullanılan karakterlerdir.

Dosya adlarını tek tek yazmak yerine, joker karakterlerle esnek aramalar ve işlemler yapılabilir.

Temel Karakterler

```
linux101@oyak:~$ ls *.txt  
linux101@oyak:~$ ls oyak?.txt  
linux101@oyak:~$ ls oyak[123].txt  
linux101@oyak:~$ ls oyak[^1].txt  
linux101@oyak:~$ touch oyak{1,2,3}.txt  
linux101@oyak:~$ touch oyak{1..10}.txt
```

cmdchallenge

```
linux101@oyak:~$ cat cmdchallenge.txt
```

CMD Challenge, komut satırında belirli görevleri çözmeye yönelik yapılan bir yarışma veya alıştırmadır. Katılımcılar, dosya yönetimi, metin işleme, sistem bilgisi, ağ komutları ve basit scripting gibi konularda görevler çözerek komut satırında yeteneklerini test ederler. Bu tür meydan okumalar, verimlilik, otomasyon ve derinlemesine sistem bilgisi kazandırır.

```
linux101@oyak:~$ xdg-open https://cmdchallenge.com/
```



init sistemi

Sistemin başlatılması sırasında ilk çalıştırılan işlemidir (PID 1).

- Çekirdek başlatıldıktan sonra devreye girer ve donanımı, dosya sistemlerini hazırlar.
- Servisleri ve arka plan işlemlerini başlatır.
- Kullanıcı oturumu açılmaya hazır hale geldiğinde terminali sağlar.
- Sistemin kapatılması veya yeniden başlatılması süreçlerini yönetir

systemd nedir?

Systemd, modern Linux dağıtımlarında kullanılan bir init sistemidir.

Örnekler:

```
linux101@oyak:~$ systemctl start <hizmet>
linux101@oyak:~$ systemctl stop <hizmet>
```

Daemon nedir?

Bir daemon, sürekli çalışan ve belirli görevleri yerine getiren veya belirli olayların gerçekleşmesini bekleyen bir arka plan işlemidir.

- Arka planda çalışır
- Belirli görevleri yerine getirir
- Olayları bekler
- Doğrudan bir kullanıcı arayüzüne sahip değildir

Sistem logları

Journalctl, systemd tarafından oluşturulan günlükleri görüntülemek için kullanılan bir komut satırı aracıdır

Örnekler:

```
linux101@oyak:~$ journalctl -xe # Son günlük girişlerini gösterir.
linux101@oyak:~$ journalctl -k # Çekirdek günlüklerini gösterir.
```



Servis Oluşturma:

İlk Systemd servisimizi oluşturalım!

```
linux101@oyak:~$ nano /home/$USER/createtimestamp.sh  
#!/bin/bash  
# timestamp dosyasi  
echo "Timestamp: $(date)" > $HOME/timestamp.txt
```

`#!/bin/bash` = Shebang, bir betik dosyasının ilk satırında yer alan ve hangi yorumlayıcı veya programın o dosyayı çalıştırması gerektiğini belirten özel bir işaretettir.

İlk önce üzerinde gösterebileceğimiz bir örnek oluşturması açısından kendi kullanıcı dizinimizde bir bash scripti oluşturuyoruz. Bu bash scripti her çalıştırıldığında `timestamp.txt` dosyasına geçerli tarihi ve saatı yazacak.

```
linux101@oyak:~$ sudo nano /etc/systemd/system/timestamp.service  
  
[Unit]  
Description=Timestamp Oluşturma Servisi  
  
[Service]  
ExecStart=/bin/bash /home/kullanici/createtimestamp.sh  
Restart=always  
User=bughra  
Group=bughra  
  
[Install]  
WantedBy=multi-user.target
```

- `sudo systemctl daemon-reload`
- `sudo systemctl start timestamp.service`



linux101@oyak:~\$ **man crontab**

CRONTAB(5)

Sistem Yönetimi

CRONTAB(5)

AD

crontab - periyodik görevleri zamanlar

AÇIKLAMA

crontab, komutları periyodik olarak çalıştırmak için kullanılır. Kullanıcılar kendi crontab dosyalarını tanımlayarak belirli zamanlarda komutların çalışmasını sağlarlar.

SÖZDİZİMİ

* * * * * komut
| | | | |
| | | | | haftanın günü (0-7)
| | | | |
| | | | | ay (1-12)
| | | | |
| | | | | ayın günü (1-31)
| | | | |
| | | | | saat (0-23)
| | | | |
| | | | | dakika (0-59)

DOSYALAR

/etc/crontab

Sistem genelindeki crontab dosyası.

/var/spool/cron/crontabs

Kullanıcı crontab dosyalarının bulunduğu dizin.



Cronjob Oluşturma:

```
linux101@oyak:~$ crontab -e  
*/10 * * * * /bin/bash /home/kullanici/createtimestamp.sh
```

*/10: Her 10 dakikada bir çalıştırılacak.
*: Her saat, her gün, her ay, her hafta.
/bin/bash: Scripti çalıştıracak olan yorumlayıcı.
/home/kullanici/createtimestamp.sh: Çalıştırılacak olan scriptin yolu.

- <https://crontab.guru/>

Dosya İzinleri

```
linux101@oyak:~$ ls -l  
drwxr-xr-x. 1 bughra users 46 Feb 6 23:03 bughra
```

Linux'ta dosya ve dizin izinleri:

- r (read): Okuma izni = 4
- w (write): Yazma izni = 2
- x (execute): Çalıştırma izni = 1

```
drwxr-xr-x. 1 bughra users 46 Feb 6 23:03 bughra  
dosya türü: (d)irectory (l)ink (b)lock  
sahibinin yetkileri  
grubunun yetkileri  
diğer kullanıcıların yetkileri
```



chmod, chown, chgrp

```
linux101@oyak:~$ chmod 664 slayt.pdf
-rw-rw-r--. 1 bughra users 623 Feb 6 23:03 slayt.pdf
linux101@oyak:~$ chown bughra:users oyak_gizli
drwxr-xr-x. 1 bughra users 46 Feb 6 23:03 oyak_gizli
linux101@oyak:~$ chgrp users slayt.pdf
```

- chmod: Dosyaların izinlerini yönetir.
- chown: Dizinlerin ve dosyaların sahipliklerini yönetir.
- chgrp: Dizinlerin ve dosyaların ait olduğu grup aidiyetini yönetir.

/etc/passwd

```
linux101@oyak:~$ head -n 10 /etc/passwd
root:x:0:0:Super User:/root:/bin/bash
bin:x:1:1:bin:/bin:/usr/sbin/nologin
daemon:x:2:2:daemon:/sbin:/usr/sbin/nologin
adm:x:3:4:adm:/var/adm:/usr/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/usr/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/usr/sbin/nologin
operator:x:11:0:operator:/root:/usr/sbin/nologin
```



/etc/shadow

```
linux101@oyak:~$ head -n 10 /etc/shadow
root:$y$j9T$Sn7.l9zwy3VX0vbgSX1JQ0$iHNq4Q9CPnTxm/ :: 0:99999:7 :::
bin:*:20084:0:99999:7 :::
daemon:*:20084:0:99999:7 :::
adm:*:20084:0:99999:7 :::
lp:*:20084:0:99999:7 :::
sync:*:20084:0:99999:7 :::
shutdown:*:20084:0:99999:7 :::
halt:*:20084:0:99999:7 :::
mail:*:20084:0:99999:7 :::
operator:*:20084:0:99999:7 :::
```

/etc/sudoers

```
linux101@oyak:~$ echo $sudoers_nedir
```

sudoers, Linux'ta sudo yetkilerini yöneten ve hangi kullanıcıların veya grupların hangi komutları root ayrıcalıklarıyla çalıştırabileceğini belirten bir yapılandırma dosyasıdır.

sudo: Superuser D0. Kullanıcıların geçici olarak root yetkisiyle komut çalıştırmasını sağlar.

wheel: Root yetkisi verebilen özel grup. %wheel grubu sudoers dosyasında sudo yetkisine sahiptir.

NOPASSWD: Sudo komutlarını şifresiz çalıştırma izin verir. sudoers dosyasında belirtilerek kullanılır.



SetUID, SetGID, StickyBit

```
linux101@oyak:~$ cat ozel_izinler.txt
```

setuid bitini bir dosyaya eklediğinizde, o dosya çalıştırıldığında dosyanın sahibi kimse, o kişi gibi çalışır.

setgid bitini bir dosyaya eklediğinizde, o dosya çalıştırıldığında dosyanın grubunun yetkileri kullanılır.

Sticky bit, genellikle paylaşımlı dizinlerde (örneğin, /tmp) kullanılır. Bu izin, dizin içindeki yalnızca dosya sahibinin veya root'un dosyasını silmesini sağlar.

```
linux101@oyak:~$ chmod u+s setuid.sh
```

```
linux101@oyak:~$ chmod g+s setgid.sh
```

```
linux101@oyak:~$ chmod +t klasor
```

Kullanıcı Yönetimi

```
linux101@oyak:~$ useradd -m burak
linux101@oyak:~$ passwd burak
linux101@oyak:~$ usermod -aG wheel burak
linux101@oyak:~$ groupadd oyakusers
linux101@oyak:~$ groupdel oyakusers
```

- useradd: Yeni kullanıcı ekler.
- userdel: Kullanıcı siler.
- usermod: Kullanıcıyı gruba ekler.
- groupadd: Grup oluşturur.
- groupdel: Grup siler.



OKUDUĞUNUZ için TEŞEKKÜRLER

```
linux101@oyak:~$ getent group wheel | awk -F: '{print $4}' | tr ',' '\n'
```

- Burak Kılınç
- Emir Buğra Erdoğan

```
linux101@oyak:~$ cat LICENSE
```

Copyright 2025 Özgür Yazılım ve Açık Kaynak Topluluğu

Permission is hereby granted, free of charge, to any person obtaining a copy of this slide and associated documentation files (the “Slide”), to deal in the Slide without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Slide, and to permit persons to whom the Slide is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Slide.

THE SLIDE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SLIDE OR THE USE OR OTHER DEALINGS IN THE SLIDE.