

Ma'ruza: Fayl tizimlari. Linux va Windows fayl tizimlari. Linuxda fayllar bilan ishlash

1. Kirish

Fayl tizimi operatsion tizimning eng muhim quyi tizimlaridan biridir.

Har qanday zamonaviy OT quyidagilarni bajara olishi kerak:

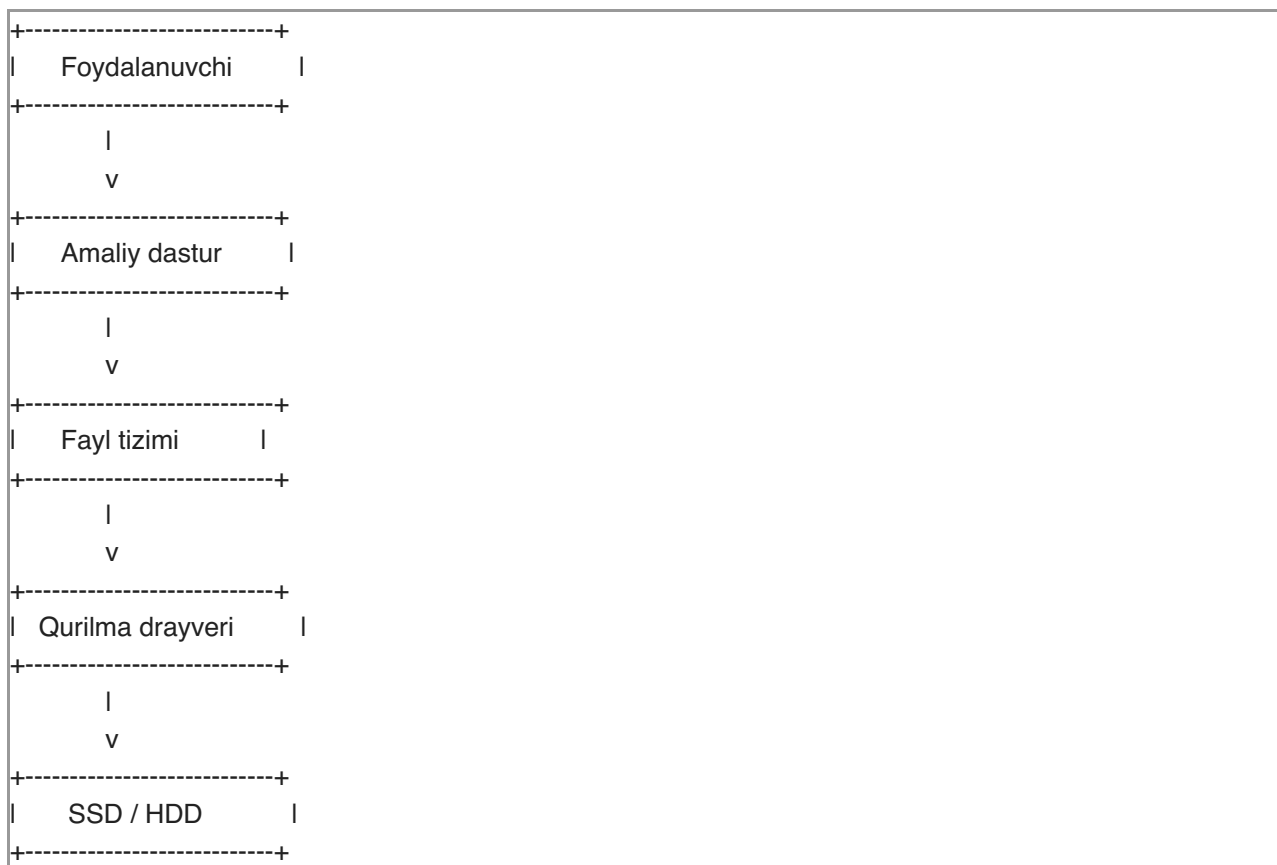
- ma'lumotlarni saqlash;
- fayllarni tashkil qilish;
- tashuvchilarda fayllarni topish;
- fayllarni yaratish va o'chirish;
- tarkibni o'zgartirish;
- kirish huquqlarini nazorat qilish;
- bo'sh joyni boshqarish;
- ma'lumotlar butunligini ta'minlash.

Foydalanuvchi odatda fayllar bilan o'ziga tushunarli kataloglar, nomlar va kengaytmalar orqali ishlaydi:

Documents/ report.docx presentation.pdf photo.jpg
--

Biroq jismoniy jihatdan fayl ma'lumotlar tashuvchisida joylashgan ma'lumotlar to'plamidir.

Ilova va jismoniy qurilma o'rtasida bir necha darajalar mavjud:



Aynan fayl tizimi ma'lumotlarni jismoniy saqlashning ko'plab murakkabliklarini yashiradi.

2. Fayl nima?

Fayl — bu ma'lumotlarni o'z ichiga olgan yoki ma'lum bir tizim resursini ifodalovchi fayl tizimining nomlangan ob'ekti.

Fayl o'z ichiga quyidagilarni olishi mumkin:

- matn;
- tasvir;
- audio;
- video;
- dastur;
- ma'lumotlar bazasi;
- konfiguratsiya;
- arxiv;
- har qanday boshqa ma'lumotlar to'plami.

Masalan:

```
report.pdf
photo.jpg
server.py
database.db
config.json
```

3. Fayl tizimi nima?

Fayl tizimi (File System) — bu tashuvchida fayllar va kataloglarni tashkil etish, saqlash va boshqarish mexanizmi.

U quyidagilarni belgilaydi:

- fayllar qanday saqlanadi;
- metama'lumotlar qayerda joylashgan;
- bo'sh bloklar qanday aniqlanadi;
- kataloglar qanday tashkil etiladi;
- fayllarni qidirish qanday bajariladi;
- kirish huquqlari qanday o'rnatiladi;
- nosozliklardan so'ng tuzilma qanday tiklanadi.

Fayl tizimi operatsion tizim umumiy arxitekturasining bir qismidir.

4. Tashuvchi va fayl tizimi — bir xil narsa emas

Quyidagilarni farqlash kerak:

Tashuvchi — jismoniy saqlash qurilmasi.

Masalan:

- HDD;
- SSD;
- NVMe SSD;
- USB Flash.

Fayl tizimi — bu tashuvchida ma'lumotlarni tashkil etish usuli.

Masalan, bitta SSD ni quyidagilarga formatlash mumkin:

NTFS

yoki:

ext4

yoki:

exFAT

Shunday qilib:

SSD

|
+--- Partition 1 → NTFS
|
+--- Partition 2 → ext4

5. Fayl tizimining asosiy elementlari

Fayl tizimi odatda bir nechta asosiy tushunchalar bilan ishlaydi:

- fayl;
- katalog;
- yo'l;
- blok;
- metama'lumotlar;
- kirish huquqlari;
- bo'sh joy;
- fayl identifikatori;
- jurnallashtirish.

6. Fayl va uning metama'lumotlari

Fayl tizimi fayl tarkibidan tashqari fayl haqidagi ma'lumotlarni ham saqlaydi.

Bunday ma'lumotlar **metama'lumotlar** deb ataladi.

Metama'lumotlarga quyidagilar kirishi mumkin:

- fayl nomi;
- o'lchami;
- turi;
- egasi;
- kirish huquqlari;
- yaratilish vaqti;
- o'zgartirilish vaqti;
- oxirgi marta murojaat qilingan vaqt;
- ma'lumotlarning joylashuvi;
- atributlar.

Masalan:

```
report.txt
O'lchami: 25 KB
Egasi: user
Huquqlar: rw-r--r--
O'zgartirilgan: 25.08.2026
```

7. Kataloglar

Katalog (directory) — boshqa fayllar va kataloglarni tashkil qilish uchun mo'ljallangan fayl tizimi ob'ekti.

Masalan:

```
/home/user/  
├── Documents/  
├── Downloads/  
├── Pictures/  
└── Projects/
```

Kataloglar ierarxik tuzilmani yaratishga imkon beradi.

8. Fayl tizimi daraxti

Fayl tizimi odatda daraxt ko'rinishida taqdim etiladi.

Masalan:

```
/  
├── bin/  
├── etc/  
├── home/  
│   ├── user1/  
│   └── user2/  
├── tmp/  
├── usr/  
└── var/
```

Linuxda eng yuqori daraja quyidagi belgi bilan ifodalanadi:

```
/
```

U **ildiz katalog (root directory)** deb ataladi.

9. Mutlaq va nisbiy yo'l

Yo'llarning ikkita asosiy turi mavjud.

Mutlaq yo'l

Ob'ektning joylashuvini to'liq tavsiflaydi.

Masalan:

```
/home/user/Documents/report.txt
```

Bunday yo'l fayl tizimining ildizidan boshlanadi.

Nisbiy yo'l

Joriy katalogga nisbatan hisoblanadi.

Masalan, agar joriy katalog quyidagicha bo'lsa:

```
/home/user
```

unda:

```
Documents/report.txt
```

nisbiy yo'l hisoblanadi.

10. Linuxning maxsus kataloglari

Linuxda maxsus belgilashlar qo'llaniladi.

•

Joriy katalog:

```
.
```

••

Ota katalog:

```
..
```

~

Joriy foydalanuvchining uy katalogi:

```
~
```

Masalan:

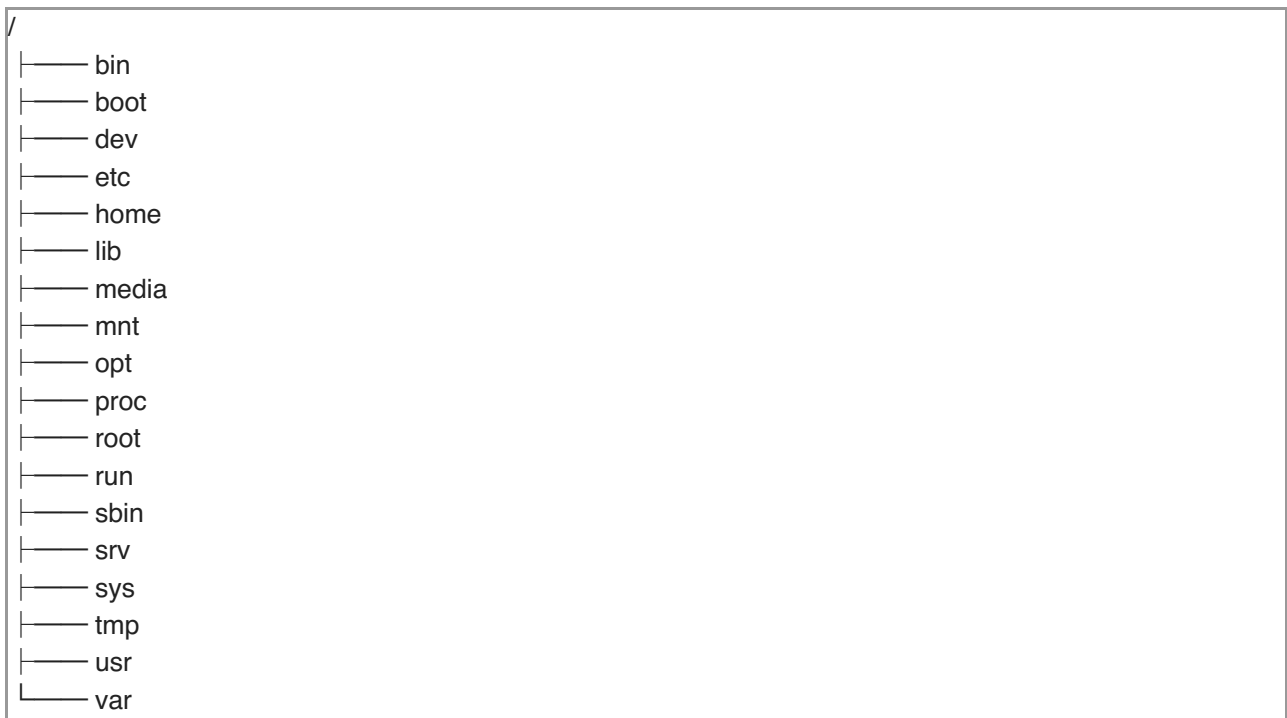
```
cd ~
```

uy katalogiga o'tadi.

11. Linuxning mantiqiy tuzilishi

Linux yagona kataloglar ierarxiasidan foydalanadi.

Odatdagi tuzilma:



Endi asosiy kataloglarning vazifasini ko'rib chiqamiz.

12. / katalogi

Ildiz katalogi:

```
/
```

butun Linux fayl ierarxiyasining boshi hisoblanadi.

Boshqa barcha kataloglar uning ichida joylashgan.

13. /home katalogi

Oddiy foydalanuvchilarning uy kataloglarini o'z ichiga oladi.

Masalan:

```
/home/ahmad
/home/student
/home/admin
```

Ichkarida quyidagilar bo'lishi mumkin:

```
Documents/
Downloads/
Pictures/
Projects/
```

14. /root katalogi

/root — root (super) foydalanuvchining uy katalogi.

Adashtirmaslik kerak:

/root

bilan:

/

/ — butun fayl tizimining ildizi.

/root — superfoydalanuvchining uy katalogi.

15. /etc katalogi

Tizim va turli xizmatlarning konfiguratsiya fayllarini o'z ichiga oladi.

Masalan:

/etc/hosts
/etc/passwd
/etc/fstab

Server dasturlarining konfiguratsiyalari ham ko'pincha /etc ichida joylashgan bo'ladi.

16. /var katalogi

/var ichida tizim ishlashi davomida o'zgaradigan ma'lumotlar saqlanadi.

Masalan:

- jurnallar (logs);
- navbatlar;
- kesh;
- ba'zi xizmatlarning vaqtinchalik ma'lumotlari.

Odatdagi katalog:

/var/log

tizim va ilova jurnallarini o'z ichiga oladi.

17. /tmp katalogi

Vaqtinchalik fayllar uchun mo'ljallangan.

Masalan:

```
/tmp
```

ilovalar tomonidan vaqtinchalik ma'lumotlarni saqlash uchun ishlatilishi mumkin.

Muhim doimiy ma'lumotlarni /tmp da saqlamaslik kerak.

18. /usr katalogi

Foydalanuvchi dasturlarining ko'p qismini, kutubxonalarni va tizimning umumiy resurslarini o'z ichiga oladi.

Masalan:

```
/usr/bin  
/usr/lib  
/usr/share
```

Zamonaviy Linux tizimlarida dasturlarning muhim qismi aynan /usr da joylashgan bo'ladi.

19. /bin katalogi

An'anaviy ravishda tizimning asosiy ishlashi uchun zarur bo'lgan asosiy bajariluvchi dasturlarni o'z ichiga oladi.

Zamonaviy distributivlarda ba'zi kataloglar, shu jumladan /bin, /usr ichidagi kataloglarga simvolik havolalar (symbolic links) bo'lishi mumkin.

Foydalanuvchi uchun bu shuni anglatadiki, kataloglarning jismoniy tashkil etilishi tarixiy sxemadan farq qilishi mumkin.

20. /dev katalogi

/dev maxsus qurilma fayllarini o'z ichiga oladi.

Masalan:

```
/dev/sda  
/dev/nvme0n1  
/dev/null
```

Bu fayl interfeysi orqali qurilmalarni taqdim etishning fundamental Unix-konsepsiyasi bilan bog'liq.

21. /proc katalogi

/proc — bu jarayonlar va yadro holati haqida ma'lumot beruvchi virtual fayl tizimi.

Masalan:

```
/proc/cpuinfo  
/proc/meminfo  
/proc/uptime
```

Quyidagi buyruqni bajarish mumkin:

```
cat /proc/cpuinfo
```

va protsessor haqida ma'lumot olish mumkin.

/proc dagi ma'lumotlar odatda diskdagi oddiy fayllar emasligini tushunish muhim.

Ular yadro tomonidan dinamik ravishda taqdim etiladi.

22. /sys katalogi

/sys — yana bir virtual fayl tizimi.

U quyidagilar haqida ma'lumot beradi:

- qurilmalar;
- drayverlar;
- yadro;
- apparat komponentlari.

Masalan:

```
/sys/class/  
/sys/devices/
```

23. /media katalogi

Odatda olinadigan (removable) qurilmalarni avtomatik ulash uchun ishlatiladi.

Masalan:

```
/media/user/USB
```

24. /mnt katalogi

An'anaviy ravishda fayl tizimlarini vaqtincha qo'lda ulash joyi sifatida ishlatiladi.

Masalan:

```
sudo mount /dev/sdb1 /mnt
```

25. Mount nima?

Linuxda fayl tizimi uning tarkibidan foydalanish uchun ma'lum bir katalogga **ulanishi (mounted)** kerak.

Masalan:

```
/dev/sdb1
|
| mount
v
/mnt/data
```

Shundan so'ng, bo'lim (partition) tarkibidan quyidagi manzil orqali foydalanish mumkin bo'ladi:

```
/mnt/data
```

26. Fayl tizimini ulash

Ulangan fayl tizimlarini ko'rish:

```
mount
```

yoki:

```
findmnt
```

Disk maydonini tahlil qilish uchun:

```
df -h
```

Misol:

Filesystem	Size	Used	Avail	Use%
/dev/nvme0n1p2	100G	50G	45G	53%

27. Linux fayl tizimlari

Linux ko'plab fayl tizimlarini qo'llab-quvvatlaydi.

Eng mashhurlari:

- ext2;
- ext3;
- ext4;
- XFS;
- Btrfs;
- ZFS;
- F2FS;
- tmpfs;
- NFS;

- CIFS/SMB.

Asosiylarini ko'rib chiqamiz.

28. Ext2

ext2 (Second Extended File System) — tarixiy Linux fayl tizimi.

U ilk Linux fayl tizimlarining davomi sifatida paydo bo'ldi.

ext2 ning asosiy kamchiligi:

- to'liq jurnallashtirishning yo'qligi.

Bugungi kunda ext2 ext4 ga qaraganda ancha kam ishlatiladi.

29. Ext3

ext3 ga jurnallashtirish qo'shilgan.

Jurnallashtirish fayl tizimining nosozliklarga chidamliligini oshiradi.

Kompyuter to'satdan o'chgandan so'ng, fayl tizimi o'zining moslashuvchan holatini tiklashi osonroq bo'ladi.

30. Ext4

ext4 — Linuxning eng keng tarqalgan fayl tizimlaridan biri.

Uning xususiyatlari:

- jurnallashtirish;
- katta fayllar va fayl tizimlarini qo'llab-quvvatlash;
- extents;
- yaxshilangan unumdorlik;
- ishonchlilik;
- ba'zi stsenariylarda ext2/ext3 bilan orqaga moslik.

Misol:

df -T

quyidagini ko'rsatishi mumkin:

Filesystem	Type	...
/dev/sda2	ext4	...

31. Journallashtirish nima?

Jurnallashtirish (journaling) — fayl tizimi ma'lum o'zgarishlar jurnalini olib boradigan mexanizm.

Soddalashtirilgan ko'rinishda:



Agar kompyuter to'satdan o'chib qolsa, jurnal tugallanmagan operatsiyalarni aniqlashga va mos holatni tiklashga yordam beradi.

Jurnallashtirish ishonchlilikni oshiradi, ammo foydalanuvchi ma'lumotlari yo'qolishidan to'liq himoya qila olmaydi.

Zaxira nusxalarini yaratish (backup) baribir zarur.

32. XFS

XFS — bu yuqori unumdorlikdagi jurnallashtiriladigan fayl tizimi, u Linuxda, ayniqsa server tizimlarida keng qo'llaniladi.

U quyidagilar uchun juda mos keladi:

- katta fayllar;
- katta hajmdagi ma'lumotlar;
- server yuklamalari;
- saqlash tizimlari.

XFS korporativ Linux infratuzilmasida keng tarqalgan.

33. Btrfs

Btrfs (B-tree File System) — kengaytirilgan imkoniyatlar to'plamiga ega zamonaviy Linux fayl tizimi.

U quyidagi mexanizmlarni taqdim etadi:

- snapshots (suratlar);
- subvolumes (kichik tomlar);
- checksums (nazorat summalari);
- copy-on-write;
- bo'sh joyni boshqarish;
- RAID bilan ichki ishlashning ba'zi imkoniyatlari.

Btrfs fayl tizimi suratlari (snapshots) va ma'lumotlarni boshqarish uchun qo'shimcha mexanizmlarni talab qiladigan tizimlar uchun qiziq.

34. ZFS

ZFS — dastlab Sun Microsystems tomonidan ishlab chiqilgan fayl tizimi va ma'lumotlarni saqlashni boshqarish tizimi.

U quyidagi xususiyatlari bilan mashhur:

- copy-on-write;
- nazorat summalari;
- snapshots;
- saqlash pullari;
- diskarni boshqarish;
- ma'lumotlarning yuqori darajadagi butunligi.

Linuxda ZFS alohida yechim sifatida mavjud, ammo standart Linux yadrosi fayl tizimi hisoblanmaydi.

35. tmpfs

tmpfs — odatda ma'lumotlarni tezkor xotirada (RAM) saqlaydigan va kerak bo'lganda swap'dan foydalanishi mumkin bo'lgan fayl tizimi.

Masalan:

/tmp

ba'zi tizimlarda tmpfs sifatida ulangan bo'lishi mumkin.

Afzalligi — yuqori tezlik.

Kamchiligi — ma'lumotlar doimiy saqlash uchun mo'ljallanmagan.

36. F2FS

F2FS (Flash-Friendly File System) flesh-xotira (flash) xususiyatlarini hisobga olgan holda ishlab chiqilgan.

U quyidagilarda ishlatilishi mumkin:

- SSD;
 - eMMC;
 - SD-kartalar;
 - boshqa flesh-qurilmalar.
-

37. Linux tarmog'i fayl tizimlari

Linux masofaviy (tarmoq) fayl tizimlari bilan ishlash qobiliyatiga ega.

Masalan:

NFS

Network File System.

Ko'pincha Linux/Unix tarmoqlari uchun ishlatiladi.

SMB/CIFS

Windows tarmoq resurslari bilan aloqa qilish uchun ishlatiladi.

Masalan:

\\server\shared

Windows tomonidan Linux'ga SMB/CIFS orqali ulanish mumkin.

38. Windows fayl tizimlari

Windows bir nechta fayl tizimlarini qo'llab-quvvatlaydi.

Eng mashhurlari:

- FAT12;
 - FAT16;
 - FAT32;
 - exFAT;
 - NTFS;
 - ReFS.
-

39. FAT

FAT (File Allocation Table) — Microsoft'ning tarixiy fayl tizimlari oilasi.

Ketma-ketlik:

FAT12 → FAT16 → FAT32

FAT dastlabki DOS va Windows versiyalarida keng qo'llanilgan.

FAT32 ning muhim kamchiliklaridan biri — bitta faylning maksimal o'lchami cheklanganligidir (4 GB), shuning uchun zamonaviy katta fayllar uchun FAT32 ko'pincha noqulay.

40. exFAT

exFAT Microsoft tomonidan flesh-xotiralar va olinadigan (removable) qurilmalar uchun ishlab chiqilgan.

Afzalliklari:

- katta fayllarni qo'llab-quvvatlaydi;
- flesh-qurilmalar uchun mos keladi;

- tuzilishi bo'yicha NTFS dan soddaroq;
- turli xil OT va qurilmalar tomonidan keng qo'llab-quvvatlanadi.

exFAT ko'pincha quyidagilar uchun ishlatiladi:

- USB-fleshkalar;
- SD-kartalar;
- tashqi disklar.

41. NTFS

NTFS (New Technology File System) — Windows NT oilasining asosiy fayl tizimi va zamonaviy Windows kompyuterlarida keng qo'llaniladigan fayl tizimi.

NTFS quyidagilarni taqdim etadi:

- kirish huquqlari (ruxsatlar);
- jurnallashtirish;
- katta fayllar;
- katta bo'limlar (partitions);
- siqish;
- Windows ning tegishli mexanizmlari orqali shifrlash;
- havolalar;
- kvotalar;
- metama'lumotlarning rivojlangan tizimi.

42. NTFS ning asosiy elementlari

NTFS ning eng muhim tarkibiy qismlaridan biri bu **MFT — Master File Table**.

MFT fayllar va kataloglar haqidagi yozuvlarni o'z ichiga oladi.

Soddalashtirilgan ko'rinishda:

```
NTFS
|
+-- MFT
| |
| +-- file1
| +-- file2
| +-- directory
|
+-- Data
|
+-- Metadata
|
+-- Journal
```

MFT NTFS tuzilishining markaziy elementlaridan biridir.

43. ReFS

ReFS (Resilient File System) — chidamlilik va yirik ma'lumotlarni saqlash tizimlariga mo'ljallangan Microsoft fayl tizimi.

U asosan ma'lum server va ixtisoslashtirilgan stsenariylarda qo'llaniladi.

Xususiyatlari quyidagilar bilan bog'liq:

- ma'lumotlar butunligi;
- katta saqlash hajmlari;
- ayrim turdagi shikastlanishlarga chidamlilik;
- ma'lumotlarni boshqarishning zamonaviy mexanizmlari.

ReFS barcha stsenariylarda NTFS ni to'liq almashtirish uchun mo'ljallanmagan.

44. Linux va Windows fayl tizimlarini solishtirish

Xususiyat	Linux	Windows
Asosiy FT	ext4, XFS, Btrfs	NTFS, exFAT, FAT32, ReFS
Ildiz (Root)	/	Odatda harf bilan disk, masalan C:
Uy kataloglari	/home	C:\Users
FT ni ulash	mount	Mount point / volume
Huquqlar (Ruxsatlar)	Unix permissions, ACL	NTFS ACL
Jurnallashtirish	ext4, XFS va boshq.	NTFS
Buyruqlar qatori	Bash va boshq.	PowerShell, cmd
Qurilmalarni ifodalash	/dev/...	Windows disklari/tomlari
Fayl kengaytmasi	Majburiy emas	Odatda foydalaniladi

45. Muhim farq: / va C:\

Linuxda barcha fayl tizimlari odatda bitta yagona ierarxiyaga birlashtiriladi.

Masalan:

```
/
├── home
├── var
├── etc
└── mnt
```

Windowsda foydalanuvchi odatda harflar bilan tomlarni (disklarni) ko'radi:

```
C:\
D:\
E:\
```

Masalan:

```
C:\Users\Ahmad\Documents
```

Bu fayl tizimini ifodalashdagi fundamental farqdir.

46. Registrni farqlash (Katta-kichik harflar)

Linuxda fayl nomlari odatda registrga sezgir (case-sensitive) bo'ladi.

Masalan:

```
file.txt
File.txt
FILE.txt
```

bular uchta xil nom bo'lishi mumkin.

Windowsda fayl tizimining registrga munosabati farq qiladi: standart NTFS stsenariylari odatda nomlarni moslashtirishda registrga sezgir emas (case-insensitive), garchi NTFS nomlarning registrini saqlasa ham va ma'lum Windows konfiguratsiyalarida case-sensitive kataloglar mexanizmlari mavjud bo'lsa ham.

Dasturchilar uchun bu juda muhim farqdir.

47. Fayl kengaytmalari

Linuxda kengaytma fayl nomining bir qismi bo'lib, fayl tizimi darajasida fayl turini belgilamaydi.

Masalan:

```
program.py
```

shunchaki oddiy fayl bo'lishi mumkin.

Uning tarkibi va huquqlari uning qanday ishlatilishini belgilaydi.

Windowsda kengaytmalar an'anaviy ravishda fayl qaysi dastur yordamida ochilishini aniqlashda katta rol o'ynaydi.

Masalan:

```
.txt
.pdf
.jpg
.exe
```

48. Linuxda kirish huquqlari

Linux egasi (owner), guruhi (group) va qolgan barcha foydalanuvchilar (others) modelidan foydalanadi.

Masalan:

```
-rwxr-xr--
```

Keling, tahlil qilamiz:

```
- rwx r-x r--  
| | |  
| | +--- others (boshqalar)  
| +----- group (guruh)  
+----- owner (egasi)
```

Bu yerda:

- r — read (o'qish);
- w — write (yozish/o'zgartirish);
- x — execute (bajarish/ishga tushirish).

49. ls -l buyrug'i

Bajarib ko'ramiz:

```
ls -l
```

Masalan:

```
-rw-r--r-- 1 user user 1250 Aug 25 09:10 report.txt
```

Quyidagilarni ko'rish mumkin:

```
-rw-r--r--
```

huquqlar (ruxsatlar);

```
user
```

egasi;

```
user
```

guruhi;

```
1250
```

o'lchami;

```
report.txt
```

fayl nomi.

50. Huquqlarni o'zgartirish — chmod

Buyruq:

```
chmod
```

kirish huquqlarini o'zgartiradi.

Masalan:

```
chmod +x script.sh
```

bajarish (execute) huquqini qo'shadi.

Raqamli yozuvdan ham foydalanish mumkin:

```
chmod 755 script.sh
```

Bu yerda:

```
7 = rwx
```

```
5 = r-x
```

```
5 = r-x
```

Yani:

```
Owner (Egasi) → rwx
```

```
Group (Guruhi) → r-x
```

```
Others (Boshqalar) → r-x
```

51. Fayl egasi — chown

Buyruq:

```
chown
```

egasini o'zgartiradi.

Masalan:

```
sudo chown user:user file.txt
```

Fayl egasi va guruhi o'zgaradi.

52. Linux ob'ektlarining turlari

Unix/Linux tizimlarida faqat oddiy fayllar emas, balki boshqa turlar ham mavjud.

Uchrashish mumkin bo'lgan turlar:

- oddiy fayl (regular file);
- katalog (directory);
- simvolik havola (symbolic link);
- qattiq havola (hard link);
- simvolli qurilma (character device);
- blokli qurilma (block device);

- FIFO (pipe);
- Unix socket.

Turini tekshirish uchun:

53. Simvolik havolalar

Simvolik havola (Symlink) — boshqa yo'lga ishorani o'z ichiga olgan maxsus ob'ekt (Windows'dagi yorliqlarga o'xshash).

Yaratish:

Endi:

Tekshirish:

54. Qattiq havolalar

Hard link (Qattiq havola) — bir xil fayl tizimi ob'ekti uchun qo'shimcha nom.

Yaratish:

Simvolik havoladan farqli o'laroq, qattiq havola odatda asl fayl yo'lini o'z ichiga olmaydi.

Ikkala nom ham (buni qo'llab-quvvatlaydigan fayl tizimlarida) bitta inode ga murojaat qiladi.

55. Inode

Unix-ga o'xshash fayl tizimlarida (masalan, ext4) **inode** muhim rol o'ynaydi.

Inode ob'ektning metama'lumotlarini saqlaydi:

- turi;
- kirish huquqlari;
- egasi;
- o'lchami;
- vaqt belgilari;
- havolalar soni;
- ma'lumotlarning joylashuvi haqida ma'lumot.

Fayl nomi va inode — bir xil narsa emas.

Soddalashtirilgan ko'rinishda:

```
Fayl nomi
|
v
inode
|
v
Ma'lumotlar bloklari
```

56. Nima uchun inode muhim?

Ko'rib chiqamiz:

```
file1.txt
file2.txt
```

Ular qattiq havola orqali bitta inode ning ikkita nomi bo'lishi mumkin:

```
file1.txt ----+
      |
      v
      inode
      |
      v
Ma'lumotlar bloklari
      ^
      |
file2.txt ----+
```

Agar bitta nom o'chirilsa ham, kamida bitta qattiq havola mavjud bo'lganda ma'lumotlar boshqa nom orqali ochiq bo'ladi.

57. Linuxda fayllar bilan ishlash

Endi asosiy buyruqlarni ko'rib chiqamiz.

Joriy katalogni ko'rsatish

```
pwd
```

Tarkibni ko'rsatish

```
ls
```

Batafsil:

```
ls -la
```

Katalogga o'tish

```
cd Documents
```

Bir daraja yuqoriga o'tish

```
cd ..
```

58. Katalog yaratish

Foydalaniladi:

```
mkdir projects
```

Ichma-ich tuzilma yaratish:

```
mkdir -p projects/python/app
```

-p opsiyasi mavjud bo'lmagan ota-onalar kataloglarini ham yaratadi.

59. Fayl yaratish

Buyruq:

```
touch file.txt
```

agar u yo'q bo'lsa, bo'sh fayl yaratadi, yoki mavjud bo'lsa uning vaqt belgilarini yangilaydi.

Shuningdek quyidagidan foydalanish mumkin:

```
echo "Salom" > file.txt
```

60. Faylni o'qish

Eng oddiy variant:

```
cat file.txt
```

Katta fayllarni ko'rish uchun:

```
less file.txt
```

Dastlabki qatorlar:

```
head file.txt
```

Oxirgi qatorlar:

```
tail file.txt
```

61. Jurnalni real vaqtda ko'rish

Juda foydali buyruq:

```
tail -f application.log
```

U faylga qo'shilayotgan yangi qatorlarni ko'rsatishda davom etadi.

Bu ko'pincha server loglarini (jurnallarini) tahlil qilishda ishlatiladi.

62. Fayllarni nusxalash

Buyruq:

```
cp source.txt destination.txt
```

Katalogni nusxalash:

```
cp -r source_dir destination_dir
```

Masalan:

```
cp report.pdf ~/Documents/
```

63. Ko'chirish va qayta nomlash

Buyruq:

```
mv old.txt new.txt
```

faylni qayta nomlaydi.

Ko'chirish mumkin:

```
mv report.pdf ~/Documents/
```

Katalogni ham ko'chirish mumkin:

```
mv project ~/Documents/
```

64. Fayllarni o'chirish

Faylni o'chirish:

```
rm file.txt
```

Bir nechtasini o'chirish:

```
rm file1.txt file2.txt
```

Tarkibi bilan katalogni o'chirish:

```
rm -r old_directory
```

Bu operatsiyaga ehtiyotkorlik bilan yondashish kerak.

Linuxda rm buyrug'i odatda faylni savatga (recycle bin) ko'chirmaydi, balki to'g'ridan-to'g'ri fayl tizimi ob'ektiga havolani o'chiradi.

65. Interaktiv o'chirish

Qo'shimcha tekshirish uchun:

```
rm -i file.txt
```

Shell sizdan tasdiqlashni so'raydi.

Muhim kataloglar bilan ishlaganda avval tekshirib ko'rish foydali:

```
pwd  
ls
```

buyruq to'g'ri joyda bajarilayotganiga ishonch hosil qilish uchun.

66. Fayllarni qidirish

find

```
find /home/user -name "*.txt"
```

Kataloglarni topish:

```
find . -type d
```

Oddiy fayllarni topish:

```
find . -type f
```

Katta fayllarni topish:

```
find . -type f -size +500M
```

67. Fayllar ichidagi matnni qidirish

Buyruq:

```
grep
```

Masalan:

```
grep "error" application.log
```

Rekursiv qidiruv bilan:

```
grep -r "database" .
```

Qator raqamlari bilan:

```
grep -n "error" application.log
```

68. Fayllar o'lchami

O'lchamni bilish:

```
du -h file.txt
```

Katalog o'lchami:

```
du -sh project/
```

Joriy katalogdagi barcha fayllar o'lchami:

```
du -h .
```

69. Diskdagi bo'sh joy

Foydalaniladi:

```
df -h
```

Masalan:

Filesystem	Size	Used	Avail	Use%
/dev/nvme0n1p2	100G	61G	34G	65%

du dan farqli o'laroq, df buyrug'i umuman fayl tizimidan foydalanishni (bo'sh joyni) ko'rsatadi.

70. du va df o'rtasidagi farq

df fayl tizimining bloklari (metama'lumotlarni hisobga olgan holda) darajasida band bo'lgan joyni ko'rsatadi.

du esa daraxtni aylanib chiqadi va ob'ektlarning o'lchamlarini qo'shib hisoblaydi.

Ba'zan ular o'rtasida farq bo'lishi mumkin (masalan, agar fayl o'chirilgan bo'lsa-yu, lekin biron bir jarayon uni hali ham ochiq ushlab turgan bo'lsa, df uni band qilingan deb ko'rsatadi, du esa topa olmaydi).