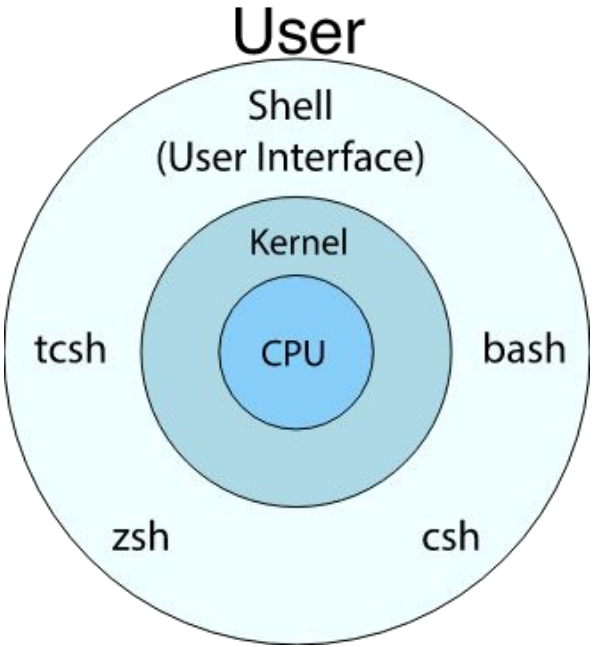
Основи Linux

Чому Linux

- Linux — потужна операційна система.
  - Багато веб-сайтів використовують Linux як операційну систему.
  - Навіть Стів Балмер з Microsoft сказав, що Linux займає 60% ринку серверів у 2008 році.
  - Толерантність до ряду апаратних платформ без спеціальної конфігурації.
- Комп'ютерна експертиза: Розгляд серверної експертизи
  - Експертні проблеми також можуть виникнути на серверних платформах.
- Експертні інструменти на основі хосту часто працюють на платформах Linux.
  - Безкоштовна платформа
  - Гнучкий і надійний
  - Простіший доступ до низькорівневих інтерфейсів
  - Хороші експертні якості.
  - Використання CAINE (a Linux live cd) для судово-експертного аналізу хостів, включаючи The Forensic Toolkit та Autopsy.

Оболонка Linux

- *Оболонка* інтерпретує команди користувача і переводить їх у дії, які може виконати ядро.
- Подібно до DOS, але DOS має лише один набір інтерфейсу, тоді як Linux може вибрати іншу оболонку ядра
    - Оболонка Bourne Again (Bash), оболонка TC (Tcsh), оболонка Z (Zsh)
- Різна оболонка має схожу, але різну функціональність
- *Bash* є типовим для Linux
- Графічний інтерфейс користувача Linux фактично є прикладною програмою, що працює над оболонкою



GRUB

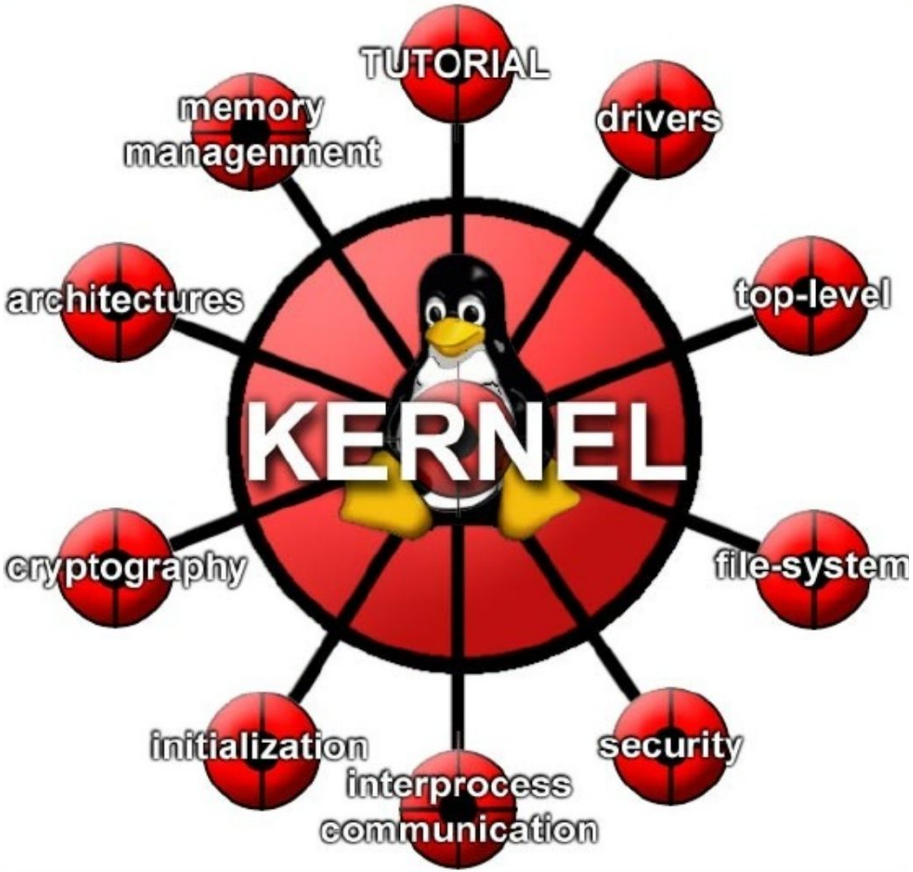
- GNU GRUB (скорочення від GNU GRand Unified Bootloader) — пакет завантажувача з проекту GNU.
- GRUB є еталонною реалізацією Специфікації Multiboot, яка надає користувачеві можливість вибору завантаження однієї з кількох операційних систем, встановлених на комп’ютері, або вибору певної конфігурації ядра, доступної для певних розділів операційної системи.

ЯДРО

- Ядро є основним центром операційної системи комп’ютера, ядром, яке забезпечує базові служби для всіх інших частин операційної системи.
- Ядро можна порівняти з оболонкою, зовнішньою частиною операційної системи, яка взаємодіє з командами користувача.
- Ядро та оболонка — це терміни, які частіше використовуються в операційних системах Unix, ніж у мейнфреймах IBM або системах Microsoft Windows.

ЯДРО

- Як правило, ядро (або будь-який подібний центр операційної системи) містить обробник переривань, який обробляє всі запити або завершені операції вводу/виводу, які конкурують за служби ядра, планувальник, який визначає, які програми спільно використовують час обробки ядра в якому порядку, і супервізор, який фактично надає використання комп’ютера кожному процесу, коли він запланований.
- Ядро також може включати менеджер адресних просторів операційних систем у пам’яті чи сховищі, який ділиться ними між усіма компонентами та іншими користувачами служб ядер. Послуги ядра запитуються іншими частинами операційної системи або прикладними програмами через певний набір програмних інтерфейсів, іноді відомих як системні виклики.


TUTORIAL
memory managenment
drivers
architectures
top-level
KERNEL
cryptography
file-system
initialization
security
interprocess communication

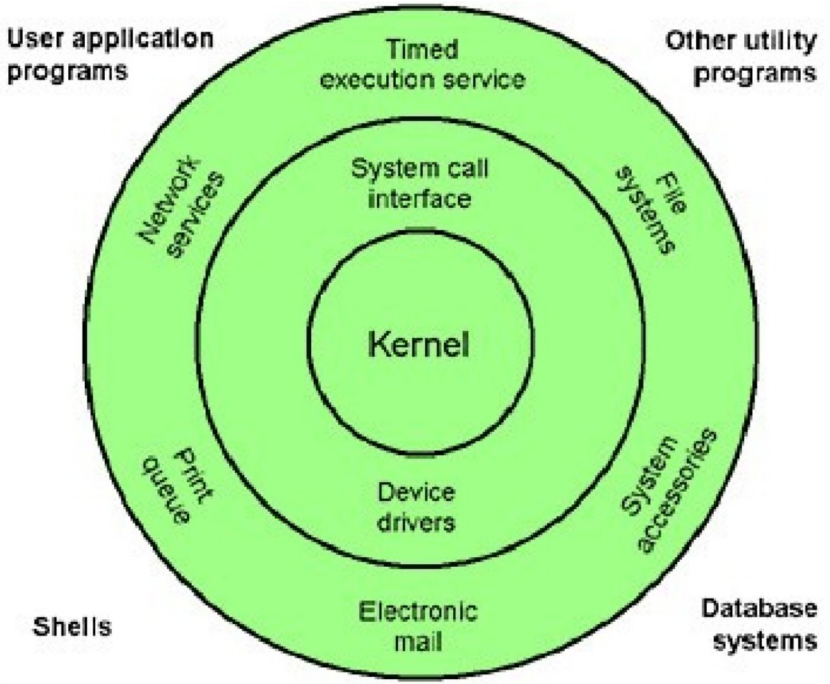
Версії ядра

- **UBUNTU 11.0 running**
- Що таке версія ядра?

Версії ядра

- UBUNTU 11.0 –назва компанії
- TYPE umane –r щоб знайти ядро .
- UBUNTU 11.0 має версію ядра 2.6.38.
- Ми також можемо створити ядро, коли запускаємо машину Linux. (GRUB)

ЯДРО


User application programs
Timed execution service
Other utility programs
Network services
System call interface
File systems
Kernel
Print queue
Device drivers
System accessories
Shells
Electronic mail
Database systems

KDE та GNOME

- KDE та GNOME — це два робочих середовища (набір програмного забезпечення, яке забезпечує певні функції та зовнішній вигляд операційних систем), які працюють в операційних системах, що використовують систему X Window (переважно Unix, Linux, Solaris, FreeBSD і Mac OS X).
- Основною мовою програмування KDE є C++. Основною причиною цього є те, що основна функціональність KDE кодується за допомогою QT, який написаний мовою C++. Для встановлення базової системи KDE потрібно приблизно 210 Мб.

  Основною мовою програмування GNOME є C, оскільки для написання GNOME використовувався набір інструментів GTK+, і він написаний на C. Для встановлення базової системи GNOME потрібно приблизно 180 МБ.

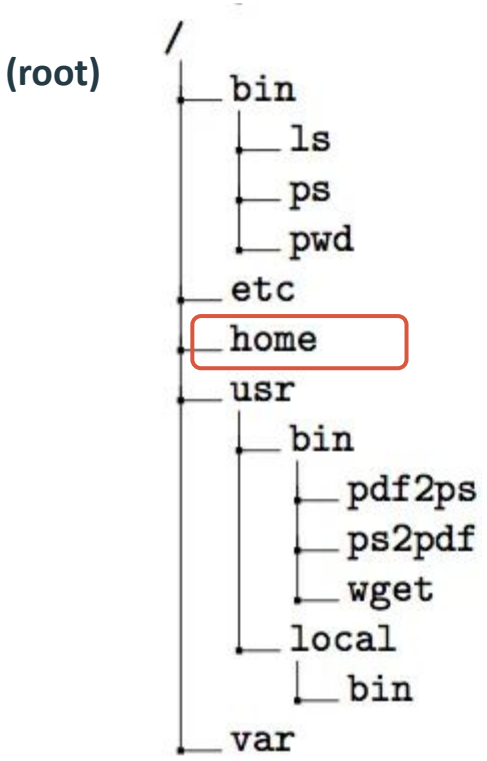
  Після нещодавнього ребрендингу «KDE» фактично стосується цілої колекції програм, включаючи середовище робочого столу, тоді як GNOME стосується лише середовища робочого столу.

Керування файлами

Дерево каталогів

Дерево каталогів

Коли ви входите в ОС Linux, використовуючи своє ім’я користувача, ви автоматично потрапляєте у свій домашній каталог.



- **/bin** Системні двійкові файли, включаючи командну оболонку
- **/boot** Процедури завантаження
- **/dev** Файли пристроїв для всіх ваших периферійних пристроїв
- **/etc** Файли конфігурації системи
- **/home** Каталоги користувачів
- **/lib** Спільні бібліотеки та модулі
- **/lost+found** Файли втраченого кластера, відновлені після перевірки диска

- **/mnt** Змонтовані файлові системи
- **/opt** Додаткове програмне забезпечення
- **/proc** Псевдофайлова система процесів ядра
- **/root** Домашній каталог адміністратора
- **/sbin** Двійкові файли системного адміністрування
- **/usr** Орієнтоване на користувача програмне забезпечення
- **/var** Інші файли: пошта, спулинг і

Каталоги

- /bin : Він містить команди, які користувач може виконувати, наприклад «ls», але які можуть знадобитися під час завантаження.
- /dev : Містить такі пристрої, як миша.
- /home : Тут користувачі зберігають свої файли.
- /tmp : Тимчасове сховище для користувачів і системи
- /var : Системні файли, які можна змінювати.
- /etc : Файли конфігурації системи, які не змінюються
- /lib : Де знаходяться всі системні бібліотеки
- /proc : Файли, які представляють запущену систему (як процеси).
- /sbin : Команди, які потрібні лише адміністратору.
- /usr : Команди, які ніколи не потрібні під час завантаження.

Найбільш важливі підкаталоги всередині кореневого каталогу

- /bin : Важливі команди Linux, доступні пересічному користувачеві.
- /boot : Файли, необхідні для завантаження системи. Не всі дистрибутиви Linux використовують однакову структуру, але деякі з них, зокрема Fedora.
- /dev : Всі драйвери пристроїв. Драйвери пристроїв — це файли, які ваша система Linux використовує для спілкування з вашим обладнанням. Наприклад, у каталозі /dev є файл для вашої конкретної марки та моделі монітора, і всі ваші комп'ютери Linux зв'язуються з монітором через цей файл.
- /etc : Файли конфігурації системи.
- /home : Кожен користувач, окрім root, отримує тут власну папку, названу за її обліковим записом. Отже, користувач, який входить до системи за допомогою linda, має каталог /home/linda, де зберігаються всі його особисті файли.
- /lib : Системні бібліотеки. Бібліотеки — це лише пакети програмного коду, які програми у вашій системі використовують для виконання завдань.

Найбільш важливі підкаталоги всередині кореневого каталогу

- /mnt : Точки кріплення. Коли ви тимчасово завантажуєте вміст CD-ROM або USB-накопичувача, ви зазвичай використовуєте спеціальне ім’я в /mnt. Наприклад, багато дистрибутивів (включаючи Fedora) за замовчуванням постачаються з каталогом /mnt/cdrom, де доступний вміст ваших приводів CD-ROM.
- /root : Домашній каталог користувача root
- /sbin : Основні команди, призначені лише для системного адміністратора.
- /tmp : Тимчасові файли та місце для зберігання. Не кладіть сюди нічого, що ви хочете зберегти. Більшість дистрибутивів Linux (включаючи Fedora) налаштовано на видалення будь-якого файлу, який знаходиться в цьому каталозі більше трьох днів.
- /usr : Програми та дані, якими можна спільно користуватися в багатьох системах і не потребують змін.
- /var : Дані, які постійно змінюються (файли журналу, які містять інформацію про те, що відбувається у вашій системі, дані на шляху до принтера тощо).

Найбільш важливі підкаталоги всередині кореневого каталогу

- /proc : віртуальна файлова система (imp для експертизи).
- /etc/passwd: містить всю інформацію про користувача.
- /etc/shadow: зберігає пароль.

Домашній каталог

- Ви можете побачити, як називається ваш домашній каталог, ввівши
- *pwd* (друк поточного робочого каталогу)

Корисні команди

Command	Description
ls	List dir and files
dir	See directory
Ifconfig	See IP address config
Useradd	Add new user
Reboot	Reboot machine
Su	Switch user
Wget	To download any file for internet
Top	See process activity
Ps	Display processes

Корисні команди

Command	Description
Iptraf	Eal time network statistics
Tcpdump	Detailed network traffic analysis
Cat	Displays contents of the file
vim	Opens text editor
Mount	Mounts dir
Cd	Change dir
Mkdir	Make dir
Rmdir	Remove dir
Pwdshow	Show present working dir

ПІДКАЗКА

- Щойно ви ввійдете в систему, ви побачите підказку. Тут ви можете виконувати свої команди.
- Все в Linux є або файлом, або каталогом.
- Файл, який виконується, стає процесом.
- Процеси також можна розглядати як файли.
- Пристрої, такі як сканери та жорсткі диски, також є файлами.

Файлова система

- Windows використовує літери алфавіту для позначення різних пристроїв і різних розділів жорсткого диска. У Windows вам потрібно знати, на якому томі (C:, D:,...) знаходиться файл, щоб вибрати його, фізичне розташування файлів є частиною його імені.
- У Linux усі каталоги приєднуються до кореневого каталогу, який позначається скісною рискою "/". - корінь.

Файлова система

Наприклад, нижче наведено кілька каталогів другого рівня:

- # - команда оболонки.
- # fdisk -l /*list partitions*/
- /dev/sda1
  - /dev – device
  - /sda1 or /hda1
    - sd – SATA /*SATA- tech to read/write data*/
    - hd – IDE
    - a
    - Sd ..a/b/c/d.....1/2/3.....
      - a – primary master
      - b – primary slave • c – secondary master • d – secondary slave • 1/2/3... – first/second/third partition

Файлова система

- # fdisk /dev/sda /*’m’ for help*/
- (if type ‘l’ він перерахує всі доступні файлові системи з їхніми ідентифікаторами e.g. Windows -7 та Linux -83)
- (‘q’ to quit)
- /dev/sda1 - системний резерв
- /dev/sda2 - is C:
- (windows makes 2 partitions: 100 Mb(from 100 GB) для системного резерву та залишку C: (100GB))

Перенаправлення

- Якщо завершити команду символом «>», її вивід буде відправлено у файл.
- Якщо завершити команду символом «<», її вхідні дані надходять із файлу.

Довідка Linux

- man
- info
- command –help
- Forums

Man -k

- Ви можете шукати команди за ключовими словами
- Наприклад, які команди показують календар?

$ man -k календарний

cal (1) - відображає календар

cal (1p) - роздрукувати календар

difftime (3p) - обчисліть різницю...

Man команди

- Параметр «-k».
  - man –k друк
- Сторінки посібника поділені на 8 розділів:
  - Команди користувача
  - Системні виклики
  - Виклики Libc
  - Пристрої
  - Формати файлів і протоколи
  - Ігри
  - Конвенції, пакети макросів і так далі
  - Системне адміністрування
- Щоб вибрати правильний розділ, додайте номер розділу:
  - man 1 passwd, man 5 passwd

Інформаційна команда

- Програма для читання документації, іноді замінює довідкові сторінки
- Приклад: info ls

Послідовність завантаження Linux

Запуск рівнів

- Вони ініціюють системний виклик, який взаємодіє з ядром.
- рівень виконання 0 : відключення.
- рівень виконання 1 : єдиний інтерфейс користувача.
- рівень виконання 2 : багато користувачів без підтримки NFS (нова файлова система).
- рівень виконання 3 : багато користувачів із підтримкою NFS (нова файлова система).
- рівень виконання 4 GUI.
- рівень виконання 5 перезапуск.

Підсумок послідовності завантаження

- BIOS
- Основний завантажувальний запис (MBR)
- LILO або GRUB
- Ядро
- ініціал
- Рівні виконання

Середовище Linux

Що таке Linux?

- "Linux" - це просто ядро ОС
- решта є додатковим програмним забезпеченням з відкритим кодом
- разом вони є "дистрибутивом Linux"
- [Knoppix, Ubuntu, Redhat, Novell/SuSe]

Великий вибір середовищ GUI та/або командного рядка

- найпопулярнішими є KDE та Gnome
- Unix-like, Mac-like, MS Windows-like, NeXT-like
- передові середовища оболонки
- веб-інтерфейси, інтерфейси GUI
- [KDE, Gnome, Windowmaker, bash, zsh, emacs, mc]

systemd

Systemd

systemd — це менеджер системи та сервісів для Linux. Це типова система ініціалізації для Debian, починаючи з DebianJessie. Systemd сумісний зі сценаріями ініціалізації SysV і LSB. Він може працювати як додаткова заміна sysvinit. Systemd

- Забезпечує агресивні можливості розпаралелювання
- Використовує сокет і активацію D-Bus для запуску служб
- On-demand starting of daemons (запуск демонів за запитом)
- Реалізує логіку керування послугами на основі транзакційних залежностей
- Відстежує процеси за допомогою контрольних груп Linux
- Підтримує знімки та відновлення
- Підтримує точки монтування та автомонтування
- Systemd працює як демон з PID 1.

Systemd

Завдання Systemd організовані як одиниці. Найпоширенішими одиницями є служби (.service), точки монтування (.mount), пристрої (.device), сокети (.socket) або таймери (.timer). Наприклад, запуск демона безпечної оболонки виконується модулем ssh.service. Systemd поміщає кожну службу в спеціальну групу керування (cgroup), названу на честь служби. Сучасні ядра підтримують ізоляцію процесів і розподіл ресурсів на основі контрольних груп. Мішені— це групи одиниць. Мішені викликають відповідні одиниці, які відповідають за різні функції системи. Наприклад, graphical.target викликає всі модулі, необхідні для запуску робочої станції з графічним інтерфейсом користувача. Мішені можуть створюватися поверх інших або залежати від інших цілей. Під час завантаження systemd активує ціль default.target, який є псевдонімом для іншої цілі, наприклад graphical.target. Systemd створює сокети, які використовуються для зв'язку між компонентами системи, і керує ними. Наприклад, спочатку створюється сокет /dev/log, а потім запускається демон syslog. Цей підхід має дві переваги: по-перше, процеси, які взаємодіють із syslog через /dev/log, можуть запускатися паралельно. По-друге, аварійні служби можна перезапустити без втрати з'єднання процесів, які спілкуються через сокети. Ядро буферизуватиме зв'язок під час перезапуску процесу.

systemd Базове використання

Отримання інформації про стан системи

Показати стан системи:

```
$ systemctl status
```

Список невдалих одиниць:

```
$ systemctl --failed
```

Список встановлених файлів пристрою:

```
$ systemctl list-unit-files
```

Управління службами

Відображення всіх активних служб:

```
$ systemctl
```

Миттєво активувати службу "example1":

```
# systemctl start example1
```

Миттєво деактивувати службу "example1":

```
# systemctl stop example1
```

Миттєво перезапустити службу "example1":

```
# systemctl restart example1
```

Показати стан служби "example1"":

```
# systemctl status example1
```

Дозволити запускати "example1" під час завантаження:

```
# systemctl enable example1
```

Вимкнути "example1", щоб не запускатися під час завантаження:

```
# systemctl disable example1
```

Створення або зміна послуг

Одиниці визначаються окремими конфігураційними файлами, які називаються файлами одиниць. Тип пристрою розпізнається за суфіксом імені файлу, .mount у випадку точки монтування. Файли модулів, надані Debian, знаходяться в каталозі /lib/systemd/system. Якщо в каталозі /etc/systemd/system існує файл локальної одиниці з однаковою назвою, він матиме пріоритет, і systemd ігноруватиме файл у каталозі /lib/systemd/system. Деякі модулі створюються системою без файлу модуля у файловій системі.

Системним адміністраторам слід розмістити нові або сильно налаштовані файли модулів у /etc/systemd/system.

Для невеликих налаштувань юніт-файлу системним адміністраторам слід використовувати функцію «випадаючого каталогу». Почніть із визначення канонічної назви служби systemd (наприклад, ssh.service, а не псевдонім, як sshd.service). Для цього прикладу ми використаємо "name.service".

Створення або зміна послуг

- Створіть каталог /etc/systemd/system/name.service.d
- Створюйте файли в цьому каталозі із суфіксом ".conf". Наприклад, /etc/systemd/system/name.service.d/local.conf
- Кожен файл має містити заголовки розділів і параметри розділів, які потрібно замінити, використовуючи той самий формат, що й файли блоків.

Невдалі одиниці

У деяких випадках підрозділи переходять у несправний стан. Команду status можна використовувати, щоб дізнатися деякі деталі:

$ systemctl status <UNITNAME>

Несправні одиниці можна видалити вручну:

 # systemctl reset-failed

Файлова система

Друга розширена файлова система (EXT2)

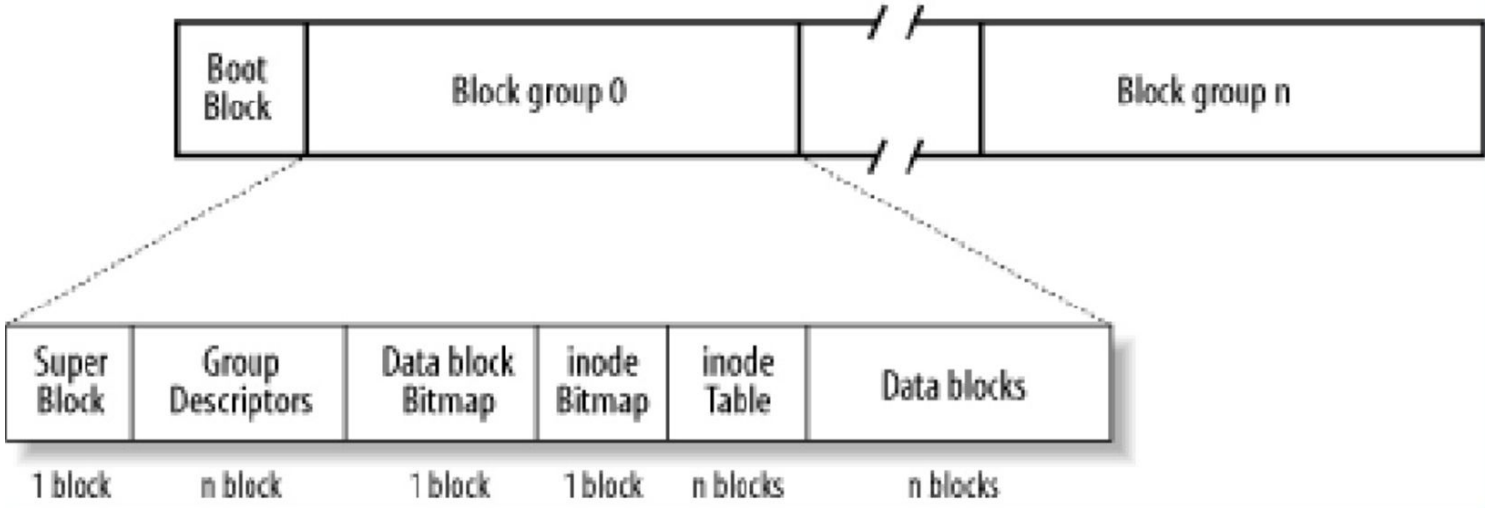
- Зазвичай використовується в Linux.
- Будівельні блоки, відомі як блоки даних (розміром 4 Кб): подібні до файлової системи FAT32.
- Allocation of blocks similar to FAT32.Thus physical sizes and logical sizes in EXT2 differ as they do in FAT32.
- Основні компоненти:
  - Каталоги.
  - Іноди.
  - Блоки даних.

Друга розширена файлова система (EXT2)

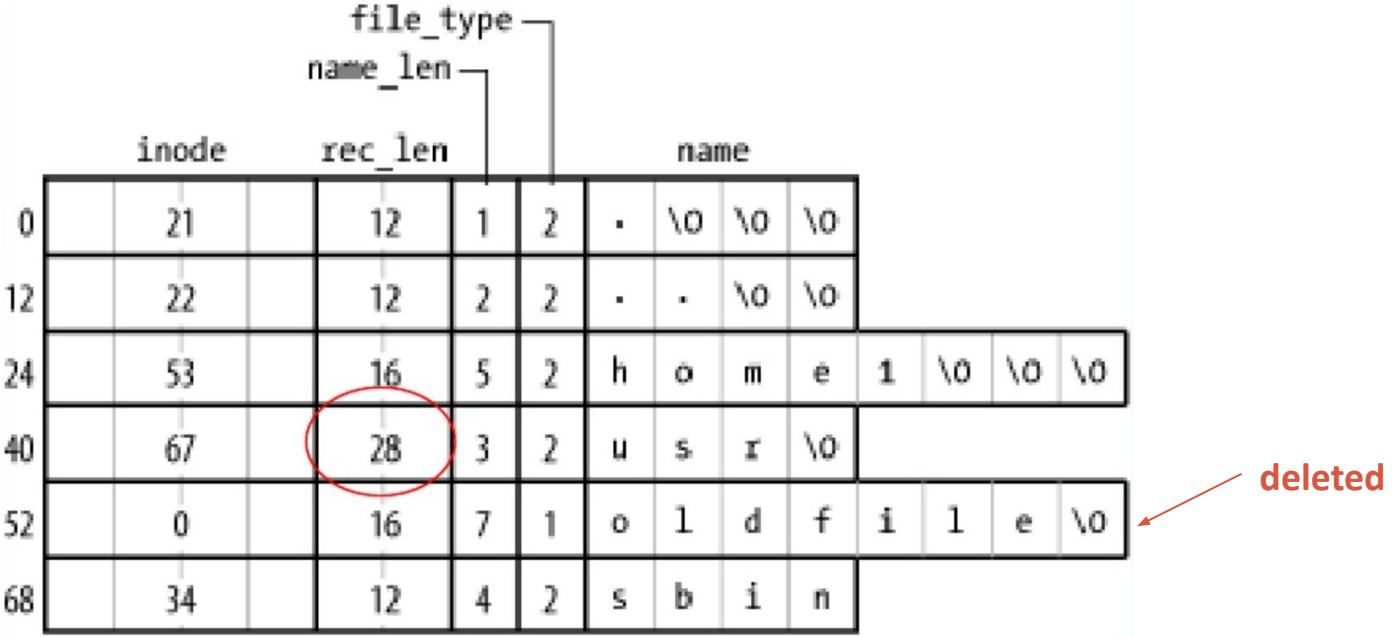
- Каталоги зберігають імена файлів у файлі sys і inode, пов’язані з файлами.
- У EXT2 файли представлені inodes. Inode містить такі атрибути, як тип файлу, розмір, права доступу, мітки часу, адреса блоків даних.
- Inode схожий на запис каталогу у FAT32. Кожен inode має унікальний номер. ідентифікатор.
- Файлова система складається з «груп блоків» — послідовного розташування групи блоків даних. Таким чином, уся файлова система управляється як ряд груп блоків.
- Метадані групи блоків надаються «дескриптором групи блоків». Він містить копію суперблоку (розмір таблиці inode і файл sys), бітове зображення блоку (відстежує розміщення кожного блоку даних), частину таблиці inode і блоків даних.

ext2 характеристики

Складна внутрішня структура для підвищення продуктивності, але структура на диску проста



ext2 Структура каталогу



inode

Існують тисячі і тисячі кожної структури. Кожному файлу та каталогу потрібен inode, і оскільки кожен файл знаходиться в каталозі, кожному файлу також потрібна структура каталогу. Структури каталогу також називаються записами каталогу або «зубцями». Кожен inode має номер inode, який є унікальним у файловій системі. Один і той самий номер inode може з'являтися в кількох файлових системах. Однак ідентифікатор файлової системи та номер inode поєднуються, щоб створити унікальний ідентифікатор, незалежно від того, скільки файлових систем змонтовано у вашій системі Linux.

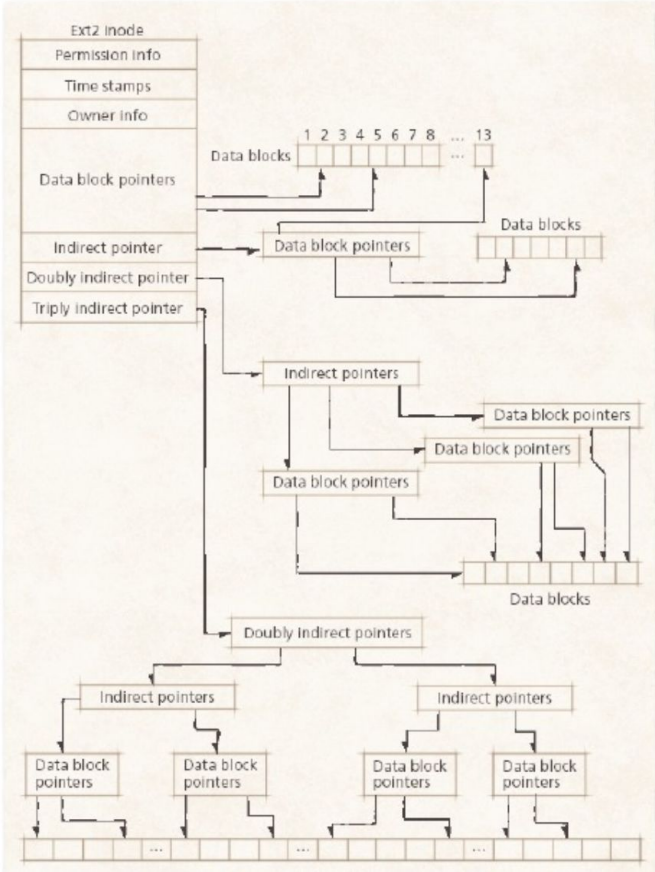
Пам'ятайте, що в Linux ви не монтуєте жорсткий диск або розділ. Ви монтуєте файлову систему, яка знаходиться на розділі, тому легко мати кілька файлових систем, не усвідомлюючи цього. Якщо у вас кілька жорстких дисків або розділів на одному диску, у вас більше однієї файлової системи. Вони можуть бути одного типу (наприклад, усі ext4), але вони все одно будуть різними файловими системами. Усі inodes зберігаються в одній таблиці. Використовуючи номер inode, файлова система легко обчислює зміщення в таблиці inode, у якій знаходиться цей inode. Ви можете зрозуміти, чому «i» в inode означає індекс.

Змінна, яка містить номер inode, оголошується у вихідному коді як 32-розрядне довге ціле число без знаку. Це означає, що номер inode є цілим числом із максимальним розміром 2^32, що обчислюється як 4 294 967 295, що значно перевищує 4 мільярди inode.

Ви можете побачити номер inode каталогу так само легко, як і для файлів. У наступному прикладі ми використаємо ls із параметрами -l (довгий формат), -i (inode) і -d (каталог) і подивимося на каталог:

```
user@Ubuntu18-1:~$ ls -lid Demo/
```

```
3539490 drwxrwxr-x 4 user user 4096 sep 30 11:54 Demo/
```



ext3 характеристики

- Двійкова сумісна з ext2 на диску
- Причина існування: величезні диски == величезна кількість часу для відновлення узгодженості файлової системи після неправильного завершення роботи
- (Треба перевірити inodes тощо)
- Значне вдосконалення порівняно з ext2: журнал, який зберігає інформацію про поточні операції з файлами
- Під час завантаження можна перевірити журнал і швидко відновити узгодженість файлової системи
- Журналювання файлових систем: НЕБЕЗПЕКА!

Видалення файлу: Linux

- видалення файлу *ext2*
  - Налаштуйте довжину попереднього запису каталогу, щоб приховати видалений запис
  - Немає реорганізації, щоб звільнити місце в каталогах
  - «перша підгонка» для нових записів каталогу на основі справжньої довжини імені
  - Inode # запису каталогу очищено
- видалення файлу *ext3*
  - Те саме, що для ext2, але…
  - Inode стирається під час видалення файлу, тому номери блоків втрачаються
  - Основна проблема з антифорензікою!
  - Але inode # запису каталогу не очищено…

Питання та відповіді