



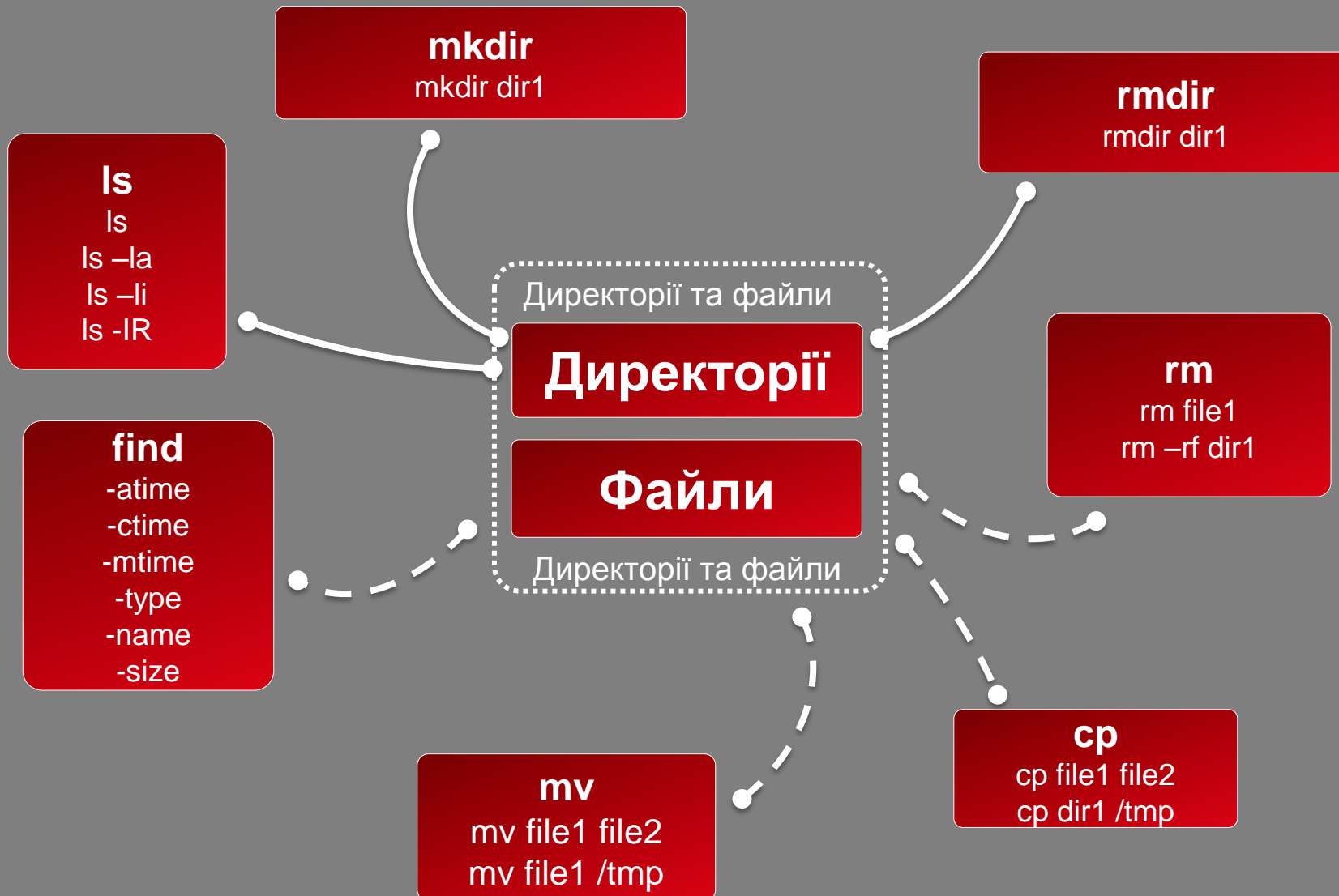
Основи управління файлами та каталогами

Основи управління файлами та каталогами

- mkdir
- rmdir
- rm
- mv
- cp
- ls
- find



Управління файлами



Ключі команди rm

-f --force	ігнорувати неіснуючі файли, виключити підтвердження
-i --interactive	завжди запитувати перед видаленням
-r R --recursive	видаляти каталоги та їх вміст рекурсивно
-v --verbose	виводити звіт про виконані дії

Ключі команди mv

-f --force	не перепитувати перед перезаписом
-i --interactive	перепитувати перед записом
-v --verbose	виводити звіт про виконані дії



Ключі команди `cp`

-d	зберігати зв'язки
-i --interactive	завжди запитувати перед перезаписом
-p	збереження режиму доступу власника і тимчасових міток
-r R --recursive	копіювати вміст рекурсивно
-v --verbose	виводити звіт про виконані дії

Ключі команди ls

-a	не ігнорувати файли, що починаються з точки
-i	вивести індексний дескриптор файлу
-l	довгий формат виведення
-r --reverse	обернений порядок при сортуванні
-R --recursive	рекурсивно виводити підкаталоги
-t	відсортувати за часом зміни
-d	вивести тільки каталоги
-s	відсортувати за розміром
-h	виводити розмір у форматі K, M, G - байт



Ключі команди find

Аргументи ключа type:

b	блоковий пристрій (пристрій зберігання)
c	символьний пристрій
d	каталог
p	іменований канал
f	звичайний файл
l	символьне посилання

Ключі команди find

-atime	час останнього звернення до файлу
-ctime	час останньої зміни дескриптора файлу
-mtime	час останньої модифікації даних файлу
-size n[c]	пошук за розміром файлу, за замовчуванням в блоках. Для пошуку в байтах - с та +/- (більше/менше) зазначеного розміру
-name	пошук в імені файлу
-path	пошук в шляху файлу

Ключі команди find

-iname	шукати файл, що задовольняє пошуковому терму, ігнорувати регістр
-perm	шукати файли з зазначеними правами доступу
-type n	шукати файли зазначеного типу, наприклад: l - для символічних посилань
-user n	шукати файл, що належить користувачеві
-xdev	шукати файли тільки на пристрої знаходження шляху пошуку

Основи управління файлами та каталогами

- touch
- file
- tar
- cpio
- dd



Управління файлами

touch

зміна часу
модифікації;
створення файлів

file

File NAME_OF_FILE

Архівація

tar

```
tar cvzf TARFILE.tar.gz
```

cpio

```
ls | cpio -ov > NEW.cpio
```

dd

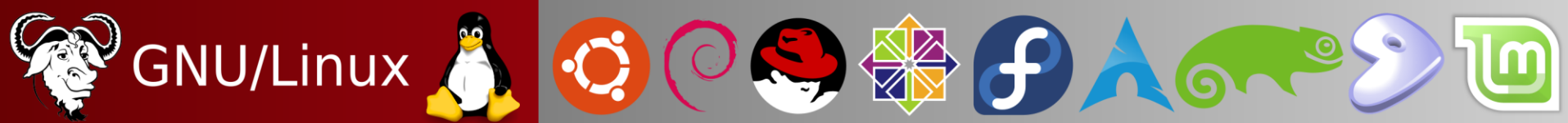
```
dd if=/dev/hda1  
of=/home/paul/boot.img
```

Директорії

Файли

Розділи

Розділи



Ключі команди touch

-a	змінювати час доступу
-m	змінювати час модифікації
-t	замість поточного часу використовувати timestamp у форматі YYMMDDhhmm (зміна mtime)
-d	замість поточного часу використовувати різні формати дати і часу, аналогічні для такого параметра команди data
-r	установка часу за подібністю

Ключі команди tar

-c	створити архів
-f	вказати ім'я архіву
-j	стискати та розпаковувати файли у форматі bzip2
-r	додати файли в існуючий архів
-t	вивести список всіх файлів, що зберігаються в архіві
-u	оновити існуючий архів
-z	стискати та розпаковувати файли у форматі gzip

Ключі команди сріо

-i	режим сору-in для копіювання файлів з архіву
-o	режим сору-out для створення архіву
-p	режим сору-pass; не створює архів, лише копіює файли з одного каталогу в інший

Основи управління файлами та каталогами

- gzip
- gunzip
- bzip2
- bunzip2
- xz
- unxz
- file-globbing



Архівація
та підстановки в назвах файлів



Ключі команд gzip та gunzip

-d	розпакування (застосується тільки з командою gzip)
-f	примусове стиснення/розпакування, навіть якщо файл має багато зв'язків або навіть якщо файл вже існує
-r	рекурсивне стиснення/розпакування
-v	режим виводу додаткової інформації



Ключі команд bzip2 та bunzip2

-d	розпакування (застосується тільки з командою bzip2)
-f	примусове стиснення/розпакування, навіть якщо файл має багато зв'язків або навіть якщо файл вже існує
-r	рекурсивне стиснення/розпакування
-v	режим виводу додаткової інформації
-z	стиснення (застосується тільки з командою bunzip2)



Підстановлювальні символи

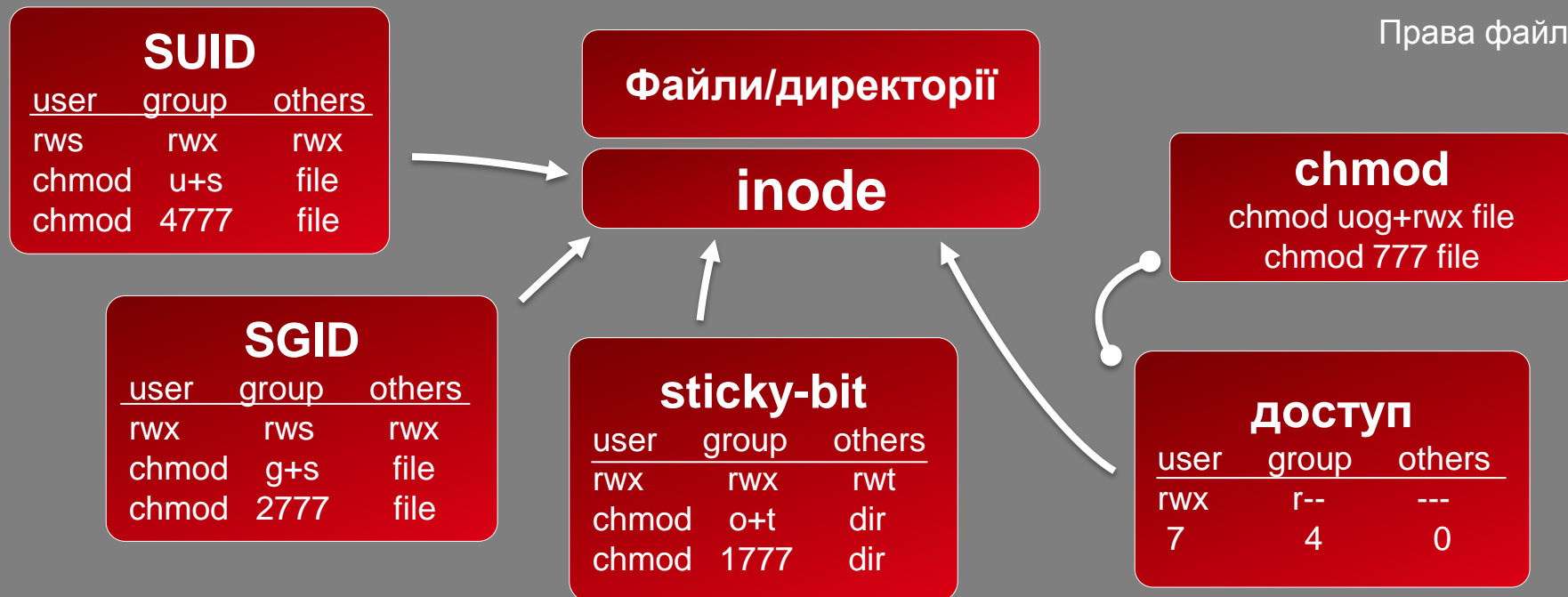
*	один та більше символів. <code>*ab</code> відповідає <code>ab</code> , <code>lab</code> , <code>aaab</code> , <code>1ab</code> и т. д.
?	тільки один символ. <code>?ab</code> відповідає <code>cab</code> и <code>1ab</code>
[at]	будь-який символ із зазначених у дужках. <code>[at]ab</code> відповідає <code>aab</code> и <code>tab</code>
[!c-z]	будь-який символ, крім тих, що вказані в дужках. <code>[!c-z]</code> відповідає <code>aab</code> и <code>bab</code>
{0, 1, 9}	містить символи 0, 1 и 9. <code>f{0, 1, 9}</code> відповідає <code>f0</code> , <code>f1</code> и <code>f9</code>

Управління правами доступу до файлів

Управління правами доступу до файлів

- Права доступу та механізм володіння в Linux
- SetUserID (SUID)
- SetGroupID (GUID)
- sticky bit

Права файлів



Права файлів

SUID	SGID	sticky	user	group	others
S	S	t	rwX	rwX	rwX
4	2	1	421	421	421

Read =4

Write =2

eXecute =1



Типи файлів в Linux

-	звичайний файл
d	каталог
l	символьне посилання
b	блоковий пристрій
c	символьний пристрій
p	іменований канал
s	сокет

Права доступу до файлів

Символьне уявлення	Вісімкове значення	Двійкове значення	Права на файл
rwX	7	111	повні права
rw-	6	110	права на читання та запис
r-X	5	101	права на читання та виконання
r---	4	100	права на читання
-wX	3	011	права на запис та виконання
-w-	2	010	права на запис
--X	1	001	права на виконання
---	0	000	відсутність прав



Права доступу до файлів

- В Linux для файлів і каталогів визначено три суб'єкти, яким вказуються права доступу:
 - користувач-власник (u);
 - група-власник (g);
 - інші користувачі (o).
- Суб'єкт «інші користувачі» включає в себе всі облікові записи, які не є власником файлу і не входять у групу-власника



Права доступу до файлів

- Три види прав доступу в Linux, які застосовуються до різних класів:
 - читання (r) дозволяє прочитати вміст файлу;
 - запис (w) дозволяє модифікувати файл (для каталогу дозволяє змінювати файли і підкаталоги);
 - виконання (x) дозволяє запустити файл на виконання.
- Права доступу можуть бути змінені за допомогою команди `chmod`. Є 2 варіанти визначення прав – вісімковий та символічний.
- Звичайні права доступу можуть розширюватися додаванням прапорів: SUID, GUID та sticky bit

Управління правами доступу до файлів

- umask
- whoami
- groups
- id
- chmod
- chown
- chgrp
- newgrp

Налаштування Umask





Жорсткі та символьні посилання

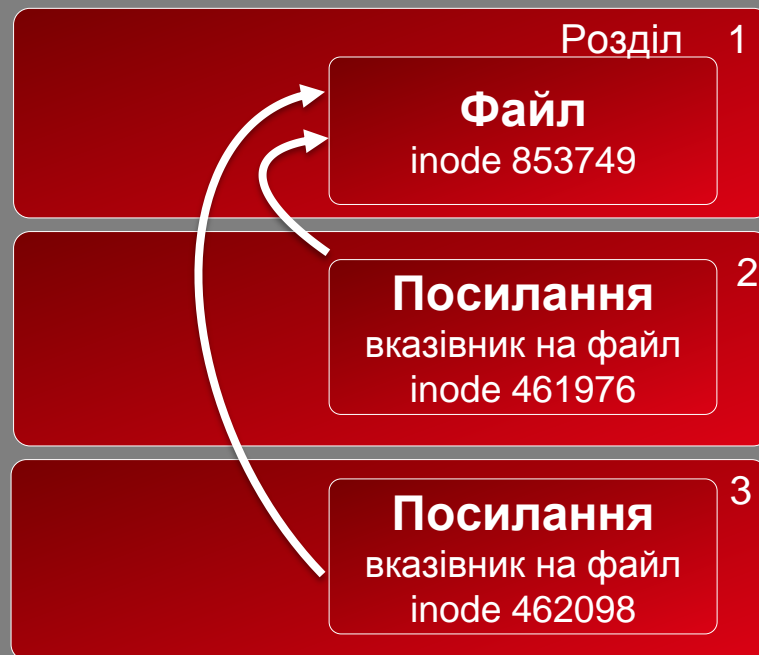
Жорсткі та символні посилання

- Жорсткі посилання
- Символьні посилання (Symlinks, Softlinks)
- In
- Is
- update-alternatives



Жорсткі посилання

Символьні посилання



Жорсткі посилання	
вказівник:	одна inode
видалення:	посилання, що залишені працюють
редагування:	відобразиться на всіх посиланнях
створення на різних розділах:	НІ
команда:	<i>ls -li</i>

Символьні посилання	
вказівник:	у посилань інша inode
видалення:	посилання не працюють
редагування:	відобразиться на всіх посиланнях
створення на різних розділах:	ТАК
команда:	<i>ls -l</i>



Жорсткі та символні посилання

- Для уникнення багаторазового копіювання одного і того ж файлу, якщо він необхідний в декількох місцях, Linux використовує посилання
- Linux використовує два різних типи посилань - жорсткі та символні
- Жорсткі посилання - це вказівники на один дескриптор файлу
- Для створення посилання використовується команда `ln`



Жорсткі та символні посилання

- Символьне посилання (symlink, softlink) - це вказівник на файл за його ім'ям
- Якщо вихідний файл, на який вказує посилання, видалити, то всі символні посилання стануть пошкодженими
- Існує 2 важливі обмеження щодо жорстких посилань. По-перше, вони повинні знаходитись в одній файловій системі з файлом, на який посилаються. По-друге, вони можуть вказувати тільки на файли (але не на каталоги)



Жорсткі та символні посилання

- Для створення символного посилання необхідно використовувати опцію `-s` в команді `ln`
- Символні посилання працюють і з файлами, і з каталогами, а також вони можуть вказувати на файл, розташований в іншій файловій системі

Пошук файлів в Linux



/
коренева файлова
система
повна ієрархія

Стандарт ієрархії файлової системи

/bin	Основні командні виконувані файли
/dev	Основні пристрої
/etc	Вузло-залежні системні файли конфігурації
/lib	Бібліотеки, необхідні для файлів в /bin та /sbin
/sbin	Необхідні системні виконувані файли (root)
/root	Домашня директорія користувача root
/usr	Містить більшість призначених для користувача утиліт і файлів
/var	Файли змінних, файли журналів
/home	Містить домашні директорії користувачів
/media	Точки монтування для портативних носіїв
/mnt	Тимчасово змонтовані файли
/opt	Опціональні пакети для додатків
/srv	Дані для публічних серверних служб
/tmp	Тимчасові файли
/proc	Ядро віртуальної файлової системи і статус процесів
/boot	Файли завантажувача системи



Стандарт ієрархії файлової системи

- Стандарт ієрархії файлової системи (Filesystem Hierarchy Standard, FHS) в Linux описує призначення каталогів в дереві файлової системи
- В Linux з єдиним розділом на жорсткому диску, всі каталоги розташовуються в кореневій файловій системі, яка називається просто «слеш» (/)
- Якщо коренева файлова система знаходиться на окремому розділі (іншому диску), то в неї необхідно включити наступні каталоги: /bin, /dev, /etc, /lib, /sbin, /root

Пошук системних файлів і їх розміщення в файловій системі

- find
- locate
- updatedb
- /etc/updatedb.conf
- which
- type
- whereis





Пошук файлів в Linux

- Команда `find` шукає файли в зазначеному дереві каталогів на підставі зазначеного пошукового виразу
- Команда `locate` шукає файли в БД, що оновлюються щоденними процесами
- БД `mlocate.db`, що використовується командою `locate`, зберігається в каталозі `/var/lib/mlocate`
- Для примусового поновлення БД розміщення використовується команда `updatedb`



Пошук файлів в Linux

- `/etc/update.conf` - конфігураційний файл команди `updatedb`
- `which` відображає повний шлях до бінарного (виконуваного) файлу в призначеній для користувача змінній `PATH`
- Команда `type` визначає існуючі псевдоніми, які можуть вплинути на її виконання
- Команда `whereis` відображає розміщення бінарних файлів, сирцевих кодів та інструкції, що відносяться до файлу