

## 1. ПОЯСНИТЬ ЕТАПИ ЗАВАНТАЖЕННЯ СИСТЕМИ LINUX.

### Run Levels

Система Linux може працювати в різних режимах . Ці режими контролюються ядром і вони відповідають яким чином ваша система може працювати. Якщо ви розташовуєте ваші програми/процеси на відповідних "run levels", то це значить, що ваш процес буде запускатися на цьому рівні. Нприклад, якщо процес запущений на Run levels 1(Single User Interface), то даний процес не буде доступний на Run Levels 3 - для інших користувачів і т.д. Це різні рівні ОС на якому можезапускатися процес. На кожному рівнізапуску можна створювати ланцюг(дерево) процесів які пов'язані один з одним і які виконуються лише на цьому рівні.

Процес завантаження системи:

- BIOS

- Master Boot Record (MBR)

- LILO or GRUB

- Kernel

- Init

- Run Levels

Запуск BIOS у якому у свою чергу BIOS boot partition (завантажник) ОС.BIOS запускає завантаження ОС, також перевіряє систему чи все ОК, після чого він завантажує завантажник MBR. Сектор MBR, що містить первинний завантажник GRUB чи LILO. GRUB/LILO в свою чергу завантажується в оперативну пам'ять системи та йде процес завантаження ядра ОС. Коли жорсткий диск системи стає активним, починається процес завантаження ядра, тобто ядро LINUX (Shel, бібліотеки тощо).

Після того як всі процеси доступні і ядро завантажене у систему, в першу чергу запускається Init процес або systemd( перший процес) і він є родичем для всіх інших наступних процесів.

Init - Традиційна система ініціалізації, яка запускає всі процеси користувача та сервіси

Systemd - сучасна система ініціалізації та керування системними процесами, яка замінила init у багатьох дистрибутивах.

Вона є більш ефективною та гнучкою. Systemd запускає units, які можуть бути services, mount points, sockets. В цей етап завантажуються всі необхідні служби, daemons та користувацькі процеси.

## 2. ЯК ПЕРЕГЛЯНУТИ СИСТЕМНІ ЖУРНАЛИ?

В операційних системах на базі Linux системні журнали містять важливу інформацію про роботу системи, програм, сервісів та ядра.

- Використання journalctl

Для перегляду конкретного опроцесу варто ввести його назву або його PID

Основні журнали знаходяться в каталозі /var/log/syslog

Для його перегляду можна використовувати команду cat

Найпоширенішим інструментом для перегляду логів є "tail", який відображає останні декілька рядків файлу. Інші корисні інструменти такі як "grep" для пошуку ключових слів, "less" для навігації по великому файлу та "journalctl" для перегляду логів systemd.

Ключові файли логів:

- /var/log/messages - містить загальносистемні повідомлення, включаючи повідомлення ядра, службові повідомлення та інші системні повідомлення

- /var/log/syslog - містить системні повідомлення, створені системним журналом

- /var/log/auth.log - містить журнали, пов'язані з автентифікацією, такі як вхід користувачів і автентифікація системи

- /var/log/dpkg.log - містить журнали, пов'язані з керуванням пакетами

- /var/log/kern.log - містить журнали ядра, включаючи апаратні помилки та проблеми з драйверами

- /var/log/boot.log - містить журнали, створені під час процесу завантаження, включаючи ініціалізацію обладнання та запуск служби.

## 3. -RW----- : ОПИШІТЬ ЦЕЙ ДОЗВІЛ. ЯК ДОДАТИ ПРАПОРЕЦЬ ВИКОНУВАНОГО ФАЙЛУ?

- "-" означає, що це звичайний файл ( не директорія і не спеціальний фал)

- "r" власник може читати файл

- "w" власник може змінювати файл

Далі власник не може виконувати файл, група немає права читати, змінювати та виконувати файл, інші користувачі не мають права

читати, змінювати та виконувати файл.  
Тобто лише власник файлу може читати та змінювати його, але не виконувати.  
Щоб зробити файл виконуваним, необхідно додати право виконання для власника, групи або інших користувачів.  
Це можна зробити за допомогою команди "chmod".  
Додати право на виконання для власника - `chmod u+x filename`  
Додати право на виконання для групи - `chmod g+x filename`  
Додати право на виконання для інших користувачів - `chmod o+x filename`  
Додати право виконання для всіх (власник, група, інші) - `chmod +x filename`

#### 4. ЯКА РІЗНИЦЯ МІЖ ART І DPKG?

`apt` та `dpkg` - це два різних інструменти для управління пакетами в операційних системах на базі Debian, таких як Ubuntu.  
Хоча обидва ці інструменти використовуються для встановлення, видалення та керування програмами, вони мають різні функції та використовуються в різних контекстах.

`dpkg` (Debian Package) є базовою утилітою для управління пакетами в системах Debian. Це низькорівневий інструмент, який дозволяє встановлювати, видаляти, та надавати інформацію про пакети `.deb`.  
- Працює тільки з локальними `.deb` файлами.  
- Не вирішує залежності автоматично. Якщо залежності не задоволені, потрібно вручну встановити їх перед використанням `dpkg`.

`apt` (Advanced Package Tool) - це високорівневий інструмент для управління пакетами, який забезпечує зручний інтерфейс для роботи з пакетами та їх залежностями. `apt` використовує `dpkg` для встановлення та видалення пакетів, але додає функції для автоматичного вирішення залежностей та роботи з репозиторіями.

- Автоматично вирішує залежності.
- Працює з віддаленими репозиторіями, завантажує та встановлює пакети з них.
- Забезпечує зручний інтерфейс для користувачів.

`dpkg` і `apt` обидва служать для управління пакетами в системах на базі Debian, але виконують різні функції.

`dpkg` - це базовий інструмент для роботи з локальними пакетами `.deb`, тоді як `apt` є високорівневим інструментом, який полегшує роботу з пакетами, автоматично вирішує залежності та працює з репозиторіями. Зазвичай, користувачі використовують `apt` для повсякденної роботи з пакетами, тоді як `dpkg` може знадобитися в специфічних випадках.