**1. Основні функції планувальника завдань в ОС (Kochubey S)**

Планувальник завдань (Task Scheduler) забезпечує автоматизацію виконання завдань у визначений час або за заданих умов. Основні функції:

* Планування виконання завдань на основі часових інтервалів (щодня, щотижня, щомісяця).
* Запуск задач за подіями (вхід користувача, завантаження системи, зміна файлів).
* Повторення завдань через визначені інтервали.
* Моніторинг та логування виконаних задач.

**Порівняння планувальників Windows та Linux:**

|  |  |  |
| --- | --- | --- |
| **Функція** | **Windows Task Scheduler** | **Linux Cron** |
| **Інтерфейс** | Графічний інтерфейс та CLI | CLI |
| **Точність планування** | Можливість планування задач з точністю до секунди | Мінімальна точність — 1 хвилина |
| **Гнучкість умов запуску** | Широкий набір умов (події, стан системи) | Обмежений набір умов (час, дата) |
| **Логування** | Автоматичне логування усіх завдань | Потрібно налаштовувати окремо через Syslog |
| **Інтеграція з системою** | Тісна інтеграція з Windows API | Легко інтегрується з іншими інструментами Linux |

**2. Основні принципи роботи з Cron в Linux**

**Cron** — це демон, який запускає задачі за розкладом, описаним у спеціальних файлах (crontab).

Кожен рядок описує задачу і складається з таких полів:

\* \* \* \* \* /шлях/до/команди

| | | | |

| | | | +— День тижня (0-7, де 0 або 7 — неділя)

| | | +——— Місяць (1-12)

| | +————— День місяця (1-31)

| +——————— Година (0-23)

+————————— Хвилина (0-59)

**Приклад:**

30 8 \* \* 1 /home/user/backup.sh

Це завдання запускає скрипт backup.sh кожен понеділок о 8:30 ранку.

**Налаштування Cron:**

**Відкрити crontab для редагування:**

crontab -e

**Перегляд запланованих задач:**

crontab -l

Альтернативи Cron

1. **Anacron**

* Призначений для завдань, які мають виконуватися з регулярними інтервалами, навіть якщо система була вимкнена.
* Підходить для ноутбуків або серверів з нерегулярним робочим часом.

1. **systemd timers**

* Частина системи systemd. Таймери можуть бути налаштовані для запуску завдань з гнучкими умовами (запуск після завантаження системи, затримка, періодичні інтервали).
* Відрізняється інтеграцією з системними службами.

**Приклад налаштування таймера:**

sudo systemctl enable mytask.timer

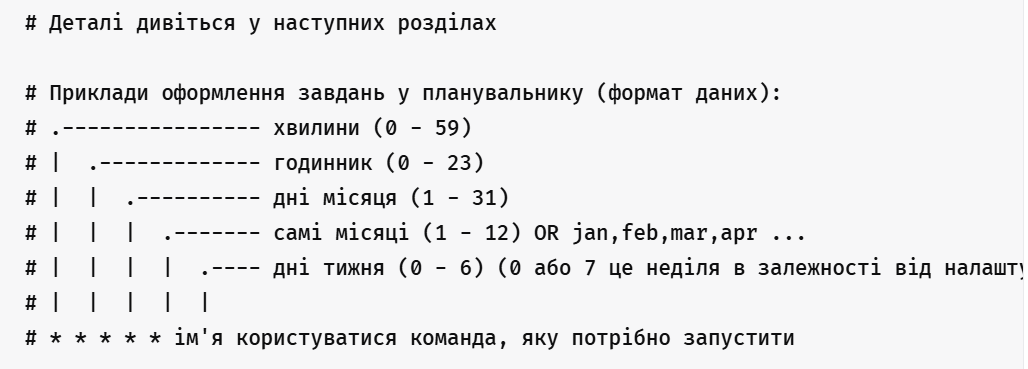
1. **at**

* Використовується для одноразового виконання завдань у визначений час.
* Команда:

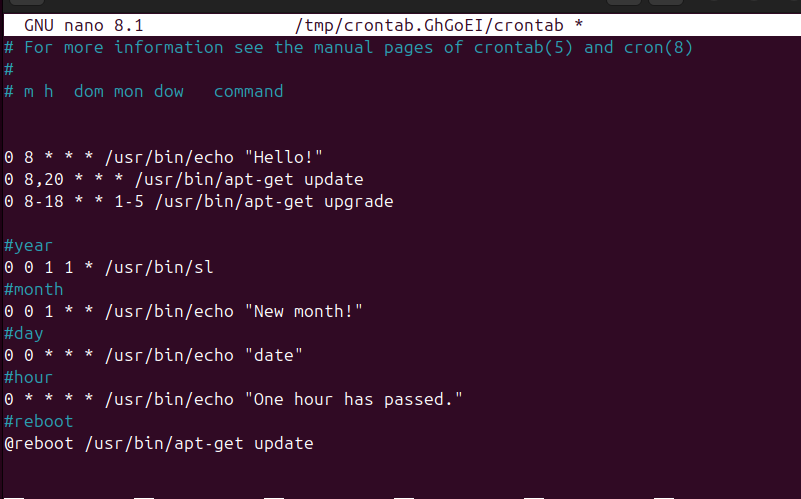
echo "команда" | at 14:00

2) (Sarapyn)

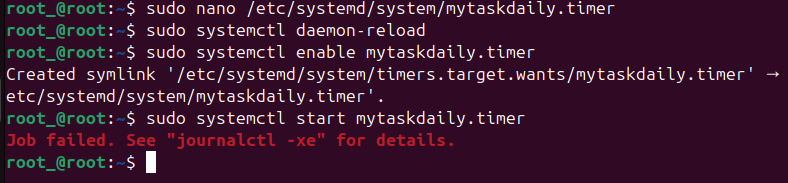
First of all I learn reference of this program:



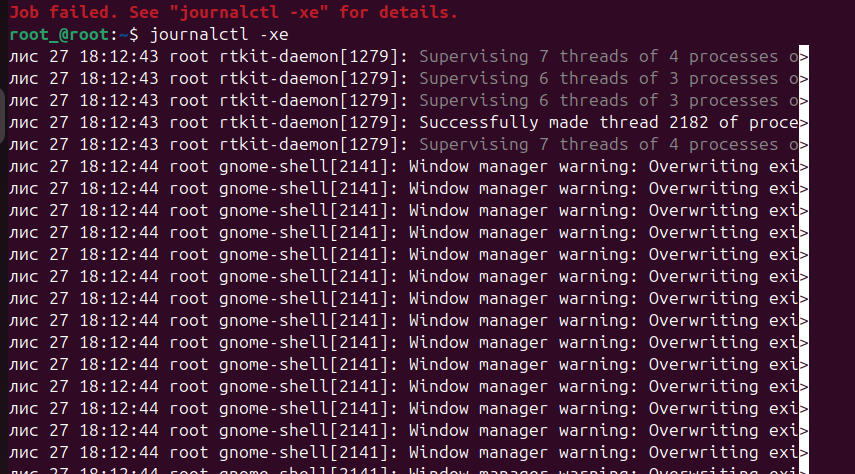
Second I set up crontab to automate recurring tasks daily, hourly, once a month, etc. Tasks for the weekend. Commands that run after a system reboot. My sample shows an example of simple commands:



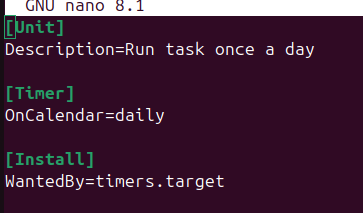
3) I used systemd to create the task timer and try to start it but I catch an unknown error:



And I tried to figurate out but I didn’t get it :(



But as I understood the file should has next content to run the task once a day:



**Висновок: (Koshkin I)**

**Automating tasks with schedulers is an integral part of modern system administration. Tools such as Cron and systemd timers provide powerful capabilities for creating complex automation scenarios. They can greatly simplify routine operations, increase system reliability, and optimize resource utilization.**

**The right choice of scheduler depends on your specific needs and the complexity of your tasks. For simple scenarios, Cron is sufficient, while for more complex and integrated systems, it is better to use systemd timers.**