Work-case 7
Матеріал підготувала Чурюмова Ксенія
1. В ході роботи досить часто виникає завдання планування задач:
- Охарактеризуйте основні функції які може виконувати планувальник завдань в будь-якій ОС. Порівняйте можливості планування завдань в різних ОС на прикладі Windows та Linux.

The task scheduler is an important component of any operating system, and performs the following basic functions:

1. Scheduling tasks: The scheduler allows users to schedule various tasks at a specific time, day, or frequency.

2. Automation of routine tasks: It helps automate routine processes such as backing up files, executing certain programs or scripts.

3. Resource management: The scheduler can control the use of resources, such as processor time, memory, network access, etc., to ensure optimal use of resources.

4. Task monitoring: It enables users to view the status of scheduled tasks and be notified of any errors or problems.

Let's compare the possibilities of scheduling tasks in Windows and Linux:

Windows:

- In Windows, the main tool for scheduling tasks is the Task Scheduler.

- Task Scheduler has an intuitive user interface that allows you to create tasks using a graphical interface.

- In Windows Task Scheduler, you can create tasks at a certain frequency (daily, weekly, monthly, etc.), and also run them at certain events, such as system startup or user login.

- Task Scheduler in Windows supports the execution of programs, scripts, package files, and command lines.

Linux:

- In Linux, one of the main tools for scheduling tasks is "Cron," as well as "systemd" in some modern distributions.

- Cron is a standard mechanism for scheduling tasks in Linux, which works on the basis of configuration files, such as/etc/crontab or other files in the/etc/cron.d/folder.

- Cron allows you to schedule tasks at regular intervals using a special syntax of temporary strings.

- Tasks scheduled in Cron can be Shell scripts, commands, or any other executable files.

So, both Windows and Linux have powerful mechanisms for scheduling tasks, but with different tools and approaches to configuration and management.
- Опишіть основні принципи роботи з планувальником Cron в ОС Linux. Як його налаштовувати? Чи є йому альтернативи (дайте їх характеристику).

The Cron task scheduler is one of the main tools for automating the execution of tasks in the Linux operating system. The basic principles of working with Cron are as follows:

1. Configuration files: Task configuration in Cron is stored in special files such as/etc/crontab or files in the/etc/cron.d/folder. Each user can also have their own file for their tasks, which is usually found in/var/spool/cron/crontabs/.

2. Time string syntax: Cron uses a special syntax to indicate the execution time of tasks. This syntax has five fields indicating minutes, hours, days of the month, months, and days of the week.

Running commands: You can specify a Shell command or script that you want to execute using Cron. This command runs at the specified time, according to the schedule you specify.

4. Frequency of execution: You can set up tasks to perform every hour, every day, every week, every month, or any other frequency you need.

Cron can be configured in several ways:

1. Using the crontab command: Users can use the crontab command to create, view, edit, and delete their tasks. The crontab -e command allows you to edit the user's crontab file in a text editor.

2. Editing system files: Administrators can edit system files such as/etc/crontab or files in the/etc/cron.d/folder to create and customize tasks for all users.

Alternatives to Cron in Linux are the following tools:

1. systemd timers: systemd is a Linux initialization and process management system. It also has the ability to schedule tasks using timers, which provide similar functionality to the Cron scheduler, but have their own syntax and features.

2. Alternative implementations of Cron: There are various alternative implementations of Cron, such as fcron, dcron, mcron, etc., which may have some improvements or additional features compared to the standard Cron. They can use other approaches to task planning or have advanced functionality.

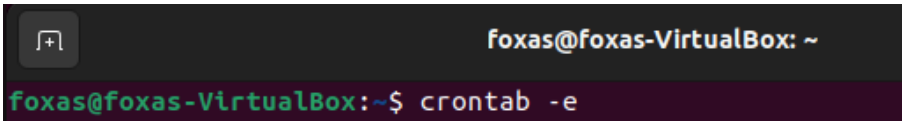
Матеріал підготував Скряга П.В.

2. Для вашої віртуальної машини зі встановленою ОС Linux здійсніть планування обраних вами задач (запуск додатків, вмикання/вимикання машини, очистка каталогів, видалення файлів, резервне копіювання, архівування тощо на ваш вибір) через планувальник Cron:

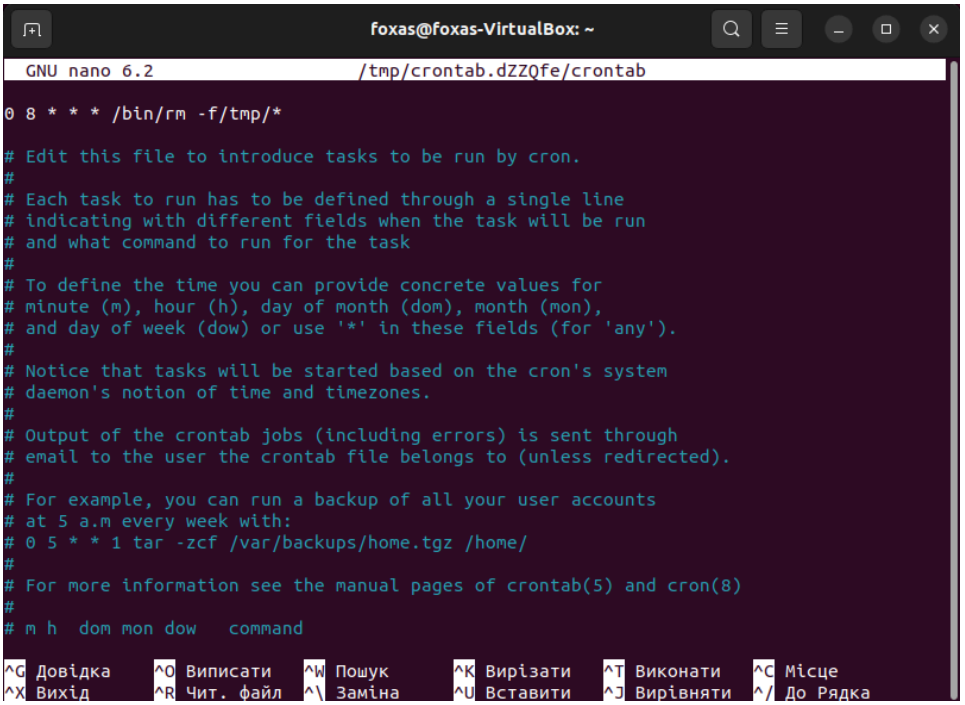
Let's take the task of cleaning the /tmp directory

- Виконання спланованої задачі в чітко визначений Вами час (наприклад о 8 ранку, 18.30 і т.д.).

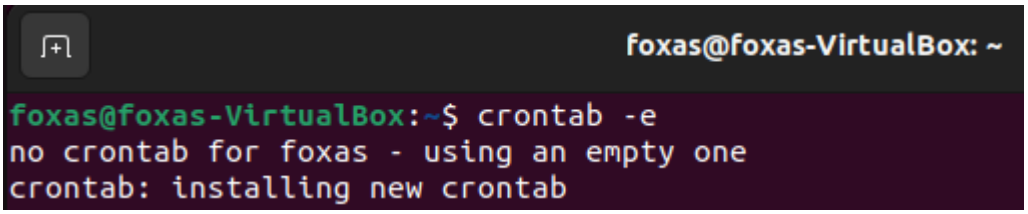
First, let's open the editing of the crontab file



Let's add a line that will call our task at 8 am every day:

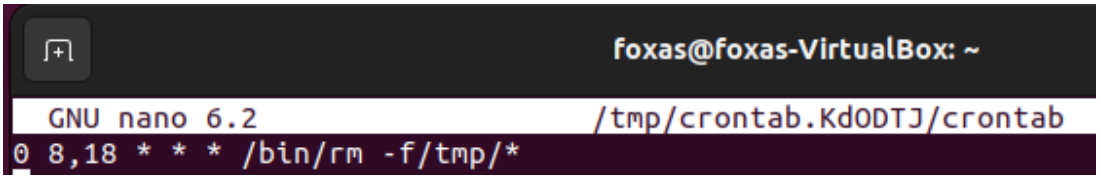


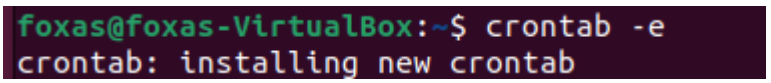
Save the file and get a message as a result



- Виконання однієї й тієї ж задачі двічі в день (час також визначаєте самостійно).

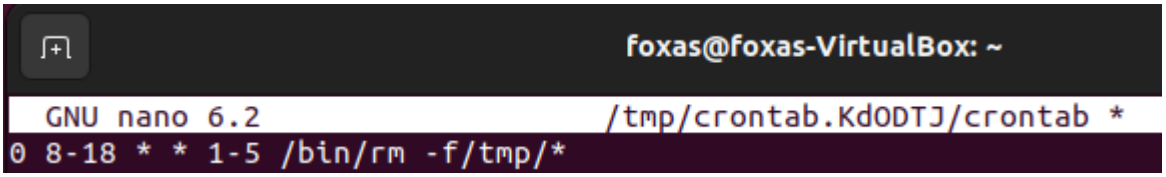
Let's make the same task that will be performed at 8 and 18 hours

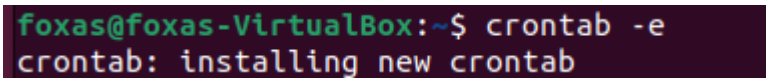




- Виконання однієї й тієї ж задачі тільки в будні (або тільки у вихідні дні) у чітко визначений проміжок часу (наприклад з 8 до 18 години).

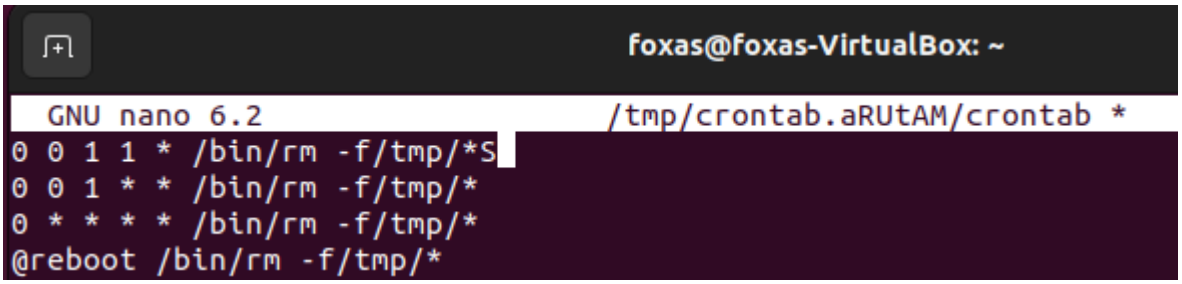
This command will run our task from Monday to Friday between 8am and 6pm.

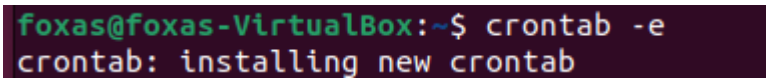




- Виконання задач тільки раз у рік, раз у місяць, раз у день, щогодини, при вмиканні (після перезавантаження).

The first team will launch our task at 12:00 am on January 1. The second team will run our task at 12:00 AM on the first day of each month. The third team will run our task at the beginning of each hour. And the last command will run our task after the system reboots.



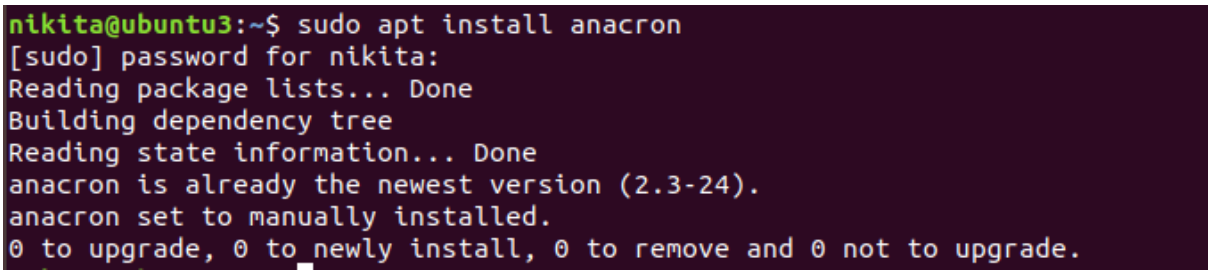


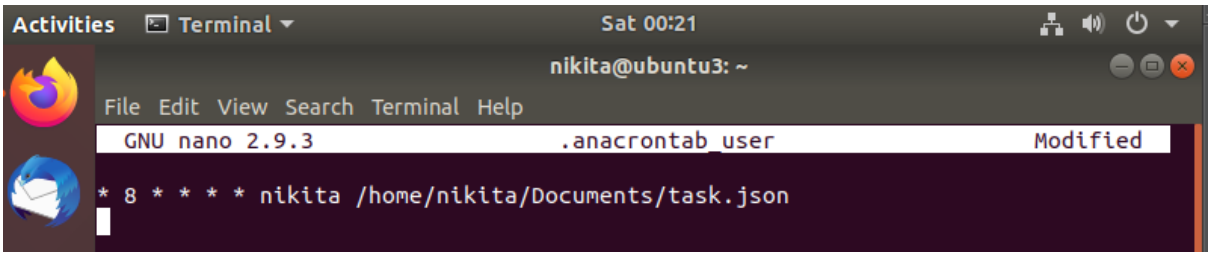
Матеріал підготував Яковенко Нікіта

3. Встановіть альтернативний Cron'у планувальник задач (на Ваш вибір). Виконані у завданні 2 дії продемонструйте через нього.

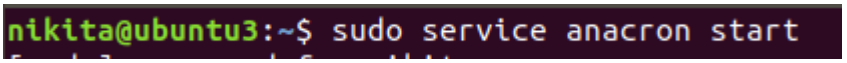
I chose the following alternative to Cron - Anacron

How to install Anacron?

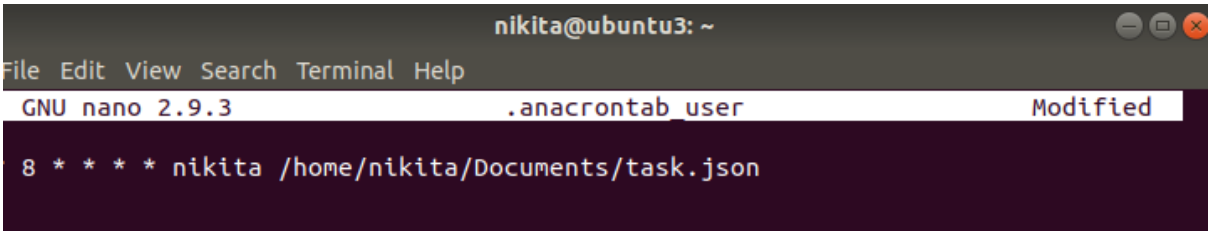


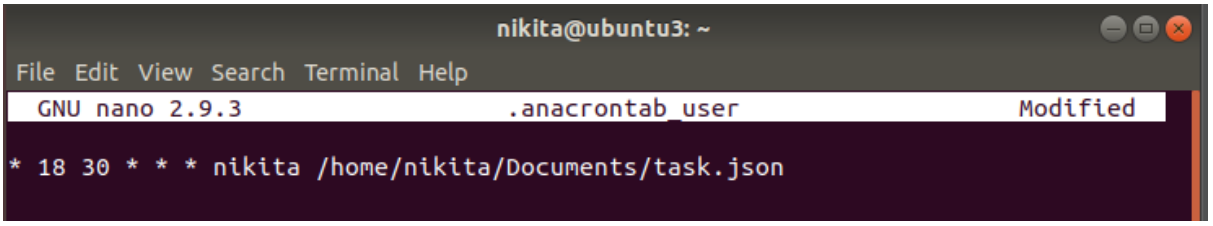


For Anacron to start processing scheduled tasks, you need to specify this :



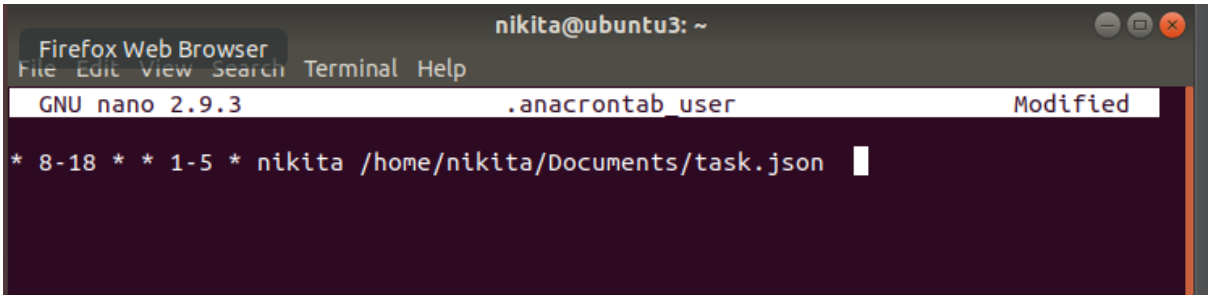
Run script /home/nikita/Documents/task.json at 8:00 AM every day.



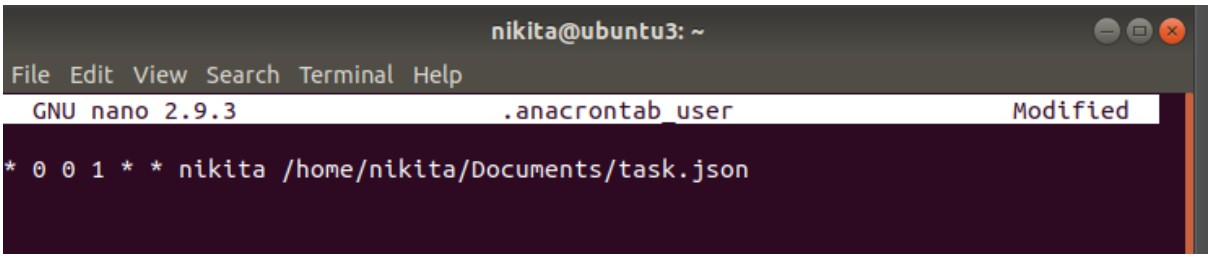


Run script /home/nikita/Documents/task.json at 6:30 PM every day.

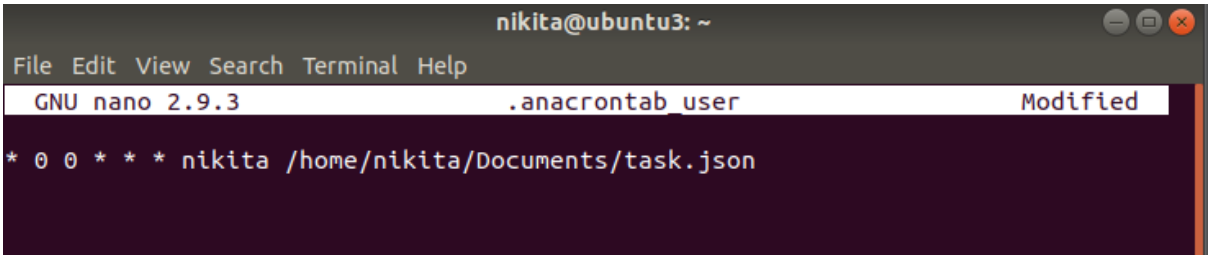
Run script /home/nikita/Documents/task.json every day from 8:00 AM to 6:00 PM on weekdays (Monday to Friday).



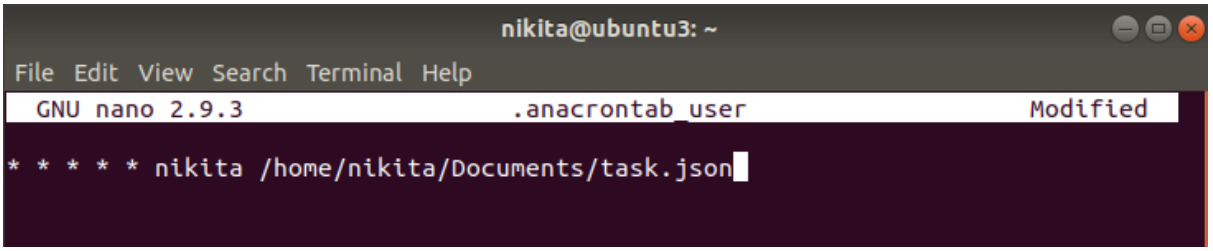
Run script /home/nikita/Documents/task.json on the 1st of each month.



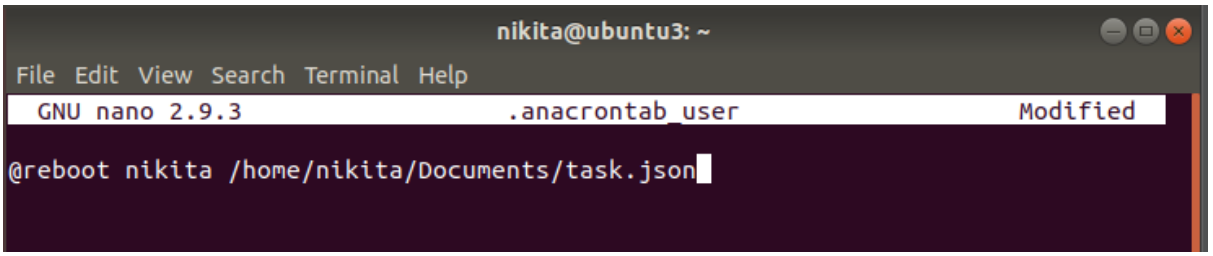
Run script /home/nikita/Documents/task.json at 00:00 every day.



Run script /home/nikita/Documents/task.json every hour.



Run script /home/nikita/Documents/task.json at system startup.



Conclusion

Expanded their knowledge of task schedulers in general and in the Linux operating system.