## Запуск задач по расписанию

### Цель работы

Изучить работу стандартного планировщика задач cron, научиться использовать его для регулярного выполнения административных задач.

### Задания для выполнения

1. Пишем пустой скрипт с расширением .sh и запускаем его. Скрипт - это простой текстовый файл, в первой строке которого содержится шебанг:

#!/usr/bin/bash

1. Делаем так, чтобы скрипт можно было запускать без указания пути
2. Делаем так, чтобы скрипт инициализировал DATE и записывал туда значение времени, затем выводил на экран
3. Делаем так, чтобы вместо вывода на экран создавался файл в директории подобный /tmp/14-05-01.txt
4. Делаем так, чтобы в файл записывалось время создания
5. Используя CRON заставляем скрипт исполняться ежеминутно
6. Через 10 минут проверяем, чтобы в директории было столько же файлов, сколько прошло минут
7. Обнаруживаем ошибку
8. Используя find делаем так, чтобы директория проверялась раз в 10 минут и удаляла файлы, созданные больше, чем 17 минут назад

### Методические указания[[1]](#footnote-0)

Cron используется для планирования задач, которые будут выполняться периодически. Вы можете настроить команды или сценарии, которые будут повторяться в заданное время. Cron — один из самых полезных инструментов в Linux или UNIX. Служба(daemon) cron работает в фоновом режиме и постоянно проверяет файл /etc/crontab и каталог /etc/cron.\*/. Он также проверяет каталог /var/spool/cron/.

Crontab — это команда, используемая для установки, удаления или вывода файла конфигурации cron, используемого для управления демоном cron. Каждый пользователь может иметь свой собственный файл задач crontab, и хотя эти файлы хранятся в /var/spool/cron/crontabs, они не предназначены для прямого редактирования. Вам нужно использовать команду crontab для редактирования или настройки ваших собственных заданий, чтобы планировщик Cron работал корректно.

Существуют различные типы файлов конфигурации:

Системный Сrontab: Обычно используется системными службами и критическими заданиями, требующими привилегий root доступа. Шестое поле (см. ниже описание поля) — это имя пользователя которое по умолчанию присвоено команде. Это дает системе возможность запускать команды от разных пользователей.

Пользовательский crontab: Пользователь может установить свои собственные задания в программе планировщик Cron, используя команду crontab. Шестое поле — это команда для запуска, и все команды будут выполняться от пользователя, который задал их в своем собственном crontab.

Чтобы отредактировать файл crontab, введите следующую команду в командной строке оболочки Unix / Linux:

$ crontab -e

#### Синтаксис crontab (описание полей)

1 2 3 4 5 /path/to/command arg1 arg2

Где цифры обозначают:

* 1: Минуты (0-59)
* 2: Часы (0-23)
* 3: День (0-31)
* 4: Месяц (0-12 [12 == Декабрь])
* 5: День недели (0-7 [7 или 0 == sunday])
* /path/to/command — имя скрипта или команды для выполнения по расписанию

Легко запоминаемый формат:

\* \* \* \* \* Команда, которая будет выполнена

- - - - -

| | | | |

| | | | - День недели (0 - 7) (воскресенье = 0 или 7)

| | | --- Месяц (1 - 12)

| | --- День месяца (1 - 31)

| ---- Час (0 - 23)

----- Минута (0 - 59)

Пример

# выполнять скрипт backupscript каждые 5 минут #

\*/5 \* \* \* \* /root/backupscript.sh

# выполнять скрипт backupscript каждый день в час ночи #

0 1 \* \* \* /root/backupscript.sh

# выполнять скрипт каждый месяц первого числа в три часа ночи пятнадцать минут #

15 3 1 \* \* /root/backupscript.sh

#### Как использовать операторы?

Оператор позволяет указать несколько значений в поле. Существует три оператора:

Звездочка (\*): этот оператор задает все возможные значения для поля. Например, звездочка в поле «Часы» будет эквивалентна каждому часу, а звездочка в поле месяца будет эквивалентна каждому месяцу и т.д.

Запятая (,): Этот оператор задает список значений, например: «1,5,10,15,20,25».

Тире (-): Этот оператор задает диапазон значений, например: «5-15» дней, что эквивалентно набору «5,6,7,8,9 … …, 13,14,15», при использования оператора «Запятая».

Слеш (/): этот оператор задает значение шага, например: «0-23 /» можно использовать в поле часов, чтобы указывать выполнение команды каждый час. Шаги также разрешены после звездочки, поэтому, если вы хотите выполнять что-то каждые два часа, просто используйте «\*/2».

Используйте специальную строку, чтобы сэкономить время

Вместо первых пяти полей вы можете использовать любую из восьми специальных строк. Планировщик Cron позволяет такую настройку. Это не только сэкономит ваше время, но и улучшит читаемость.

Специальная строка Значение

@reboot Запускается только один раз при запуске.

@yearly Запускается раз в год, «0 0 1 1 \*».

@annually Такое же, как у @yearly

@monthly Запускается раз в месяц, «0 0 1 \* \*».

@weekly Выполняется раз в неделю, «0 0 \* \* 0».

@daily Выполняется один раз в день, «0 0 \* \* \*».

@midnight Такое же, как у @daily.

@hourly Запускается один раз в час, «0 \* \* \* \*».

Пример:

# Запуск команды ntpdate каждый час #

@hourly /path/to/ntpdate

### Контрольные вопросы

1. Как можно запретить определенным пользователям запуск заданий планировщика?
2. Как можно просмотреть или удалить расписание текущего пользователя?
3. Что такое демонизация процесса? Зачем оно нужно?
4. Как можно пользоваться специальными папками cron для планирования задач?

### Дополнительные задания

1. После выполнения работы просмотрите системный журнал и найдите в нем (автоматически) записи, соответствующие выполнению вашего скрипта. Подсчитайте количество запусков.
2. Напишите скрипт автоматического архивирования домашней папки. Напишите установочный скрипт, который спрашивает пользователя о периодичности архивации и автоматический добавляет скрипт в расписание.

1. По материалам <https://blog.sedicomm.com/2017/07/24/kak-dobavit-zadanie-v-planirovshhik-cron-v-linux-unix/> [↑](#footnote-ref-0)