Лабораторная работа №8

Отчет

Устинова Виктория Вадимовна

Содержание

# 1 Цель работы

Получение навыков работы с планировщиками событий cron и at

# 2 Задание

1. Выполните задания по планированию задач с помощью crond (см. раздел 8.4.1).
2. Выполните задания по планированию задач с помощью atd (см. раздел 8.4.2)

# 3 Выполнение лабораторной работы

Посмотрите статус демона crond:(рис. 1).

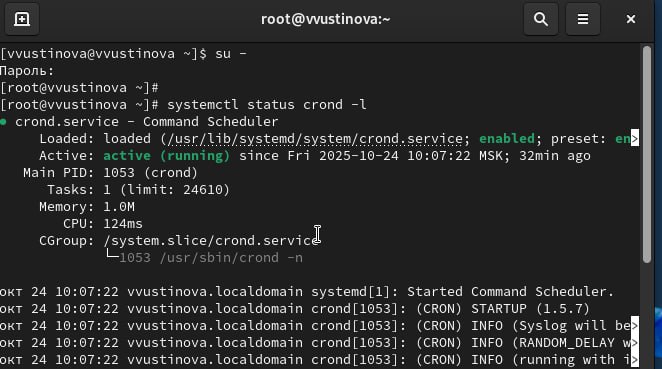


Рис. 1: Получаем полномочия администратора и смотрим статус

Посмотрите содержимое файла конфигурации /etc/crontab, посмотрите список заданий в расписании(рис. 2).

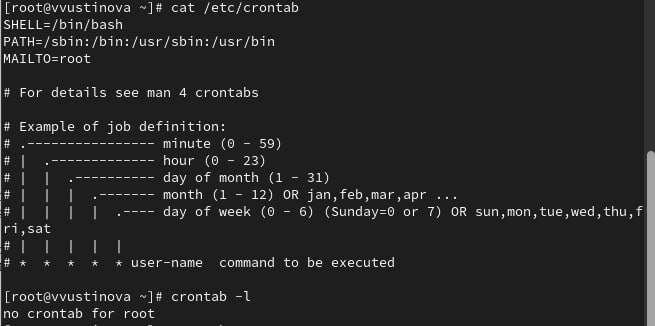


Рис. 2: Смотрим файл и у нас не отображаются задания потому что расписание еще не задано

Откройте файл расписания на редактирование и добавьте следующую строку в файл расписания (рис. 3).

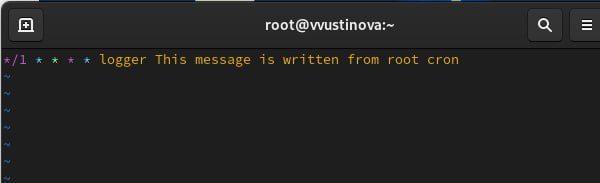


Рис. 3: Записываем строку в файл

Посмотрите список заданий в расписании(рис. 4).

В расписании появилась запись о запланированном событии.

Рис. 4: В расписании появилась запись о запланированном событии.

Не выключая систему, через некоторое время (2–3 минуты) просмотрите журнал системных событий(рис. 5).

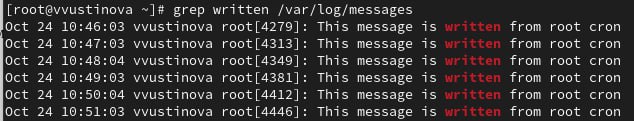


Рис. 5: ждем 6 минут и возвращаемся, у нас отображается 6 строк

Измените запись в расписании crontab на следующую: 0 \*/1 \*\*1-5 logger This message is written from root cron и посмотрите список заданий в расписании:(рис. 6).

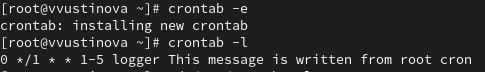


Рис. 6: Добавляем через vi новую запись(замена) и проверяем(Каждый будний день в начале каждого часа записывает сообщение в системный журнал.)

Перейдите в каталог /etc/cron.hourly и создайте в нём файл сценария с именем eachhour,откройте файл для редактирования и пропишите в нём строку, делайте файл сценария eachhour исполняемым: (рис. 7).

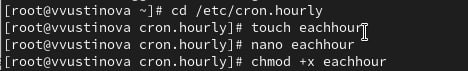


Рис. 7: Переходим в др.каталог, создаем файл и в nano добавляем стрчоку, сделали его исполняемым

Теперь перейдите в каталог /etc/crond.d и создайте в нём файл с расписанием eachhour, Откройте этот файл для редактирования и поместите в него строчку(рис. 8).

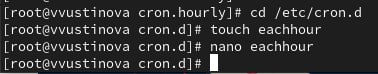


Рис. 8: Переходим в др.каталог, создаем файл и в nano добавляем строчку( Каждый час на 11-й минуте записывает сообщение в системный журнал, от имени пользователя root)

Не выключая систему, через некоторое время (2–3 часа) просмотрите журнал системных событий, По журналу определите, был ли осуществлён запуск сценария eachhour в соответствии с заданным расписанием.(рис. 9).

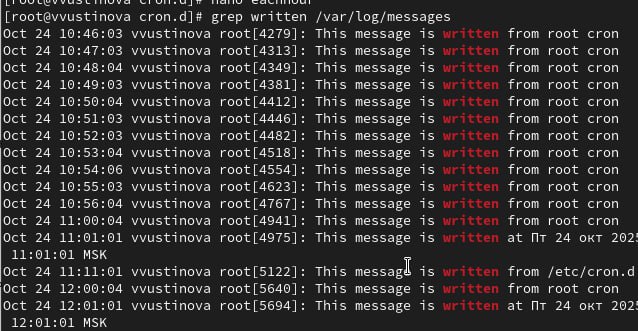


Рис. 9: Ждем 2,5 часа и проверяем, все осуществлено с заданием

Проверьте, что служба atd загружена и включена(рис. 10).

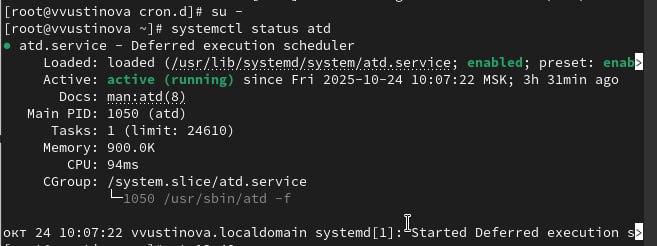


Рис. 10: Получаем полномочия администратора и смотрим статус

Задайте выполнение команды logger message from at в 9:30 (или замените на любое другое время, когда вы работаете над этим упражнением и введите там logger message from at, убедитесь, что задание действительно запланировано:(рис. 11).

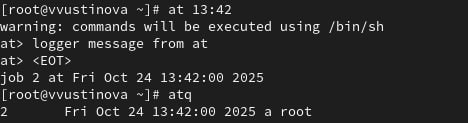


Рис. 11: действие запланировано

С помощью команды grep ‘from at’ /var/log/messages посмотрите, появилось ли соответствующее сообщение в лог-файле в указанное вами время(рис. 12).

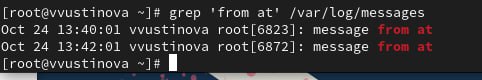


Рис. 12: Да, все появилось

# 4 Выводы

Мы успешно получили навыкы работы с планировщиками событий cron и at

# 5 Ответы на контрольные вопросы

1. Раз в 2 недели: Cron не имеет прямой поддержки. Используйте скрипт, проверяющий дату или параметр date в cron.
2. 1-е и 15-е в 2 ночи: 0 2 1,15 команда
3. Каждые 2 минуты: /2 \* команда
4. 19 сентября ежегодно: 0 0 19 9 \* команда
5. Четверг сентября ежегодно: 0 0 \* 9 4 команда
6. Назначить cron пользователю alice: sudo crontab -u alice -e (может потребоваться sudo)
7. Запретить bob использовать cron: Добавьте bob в /etc/cron.deny.
8. Выполнять задание даже при пропуске: Используйте anacron.
9. Посмотреть запланированные atd задания: atq