Отчет по лабораторной работе №8

Планировщики событий

Сидорова Арина Валерьевна

Содержание

# 1 Цель работы

Получение навыков работы с планировщиками событий cron и at.

# 2 Выполнение лабораторной работы

## 2.1 Планирование задач с помощью cron

Запустим терминал и получим полномочия администратора: su -

Посмотрим статус демона crond: systemctl status crond -l

Посмотрим содержимое файла конфигурации /etc/crontab: cat /etc/crontab

Посмотрим список заданий в расписании: crontab -l Ничего не отобразится, так как расписание ещё не задано. (рис. 1)

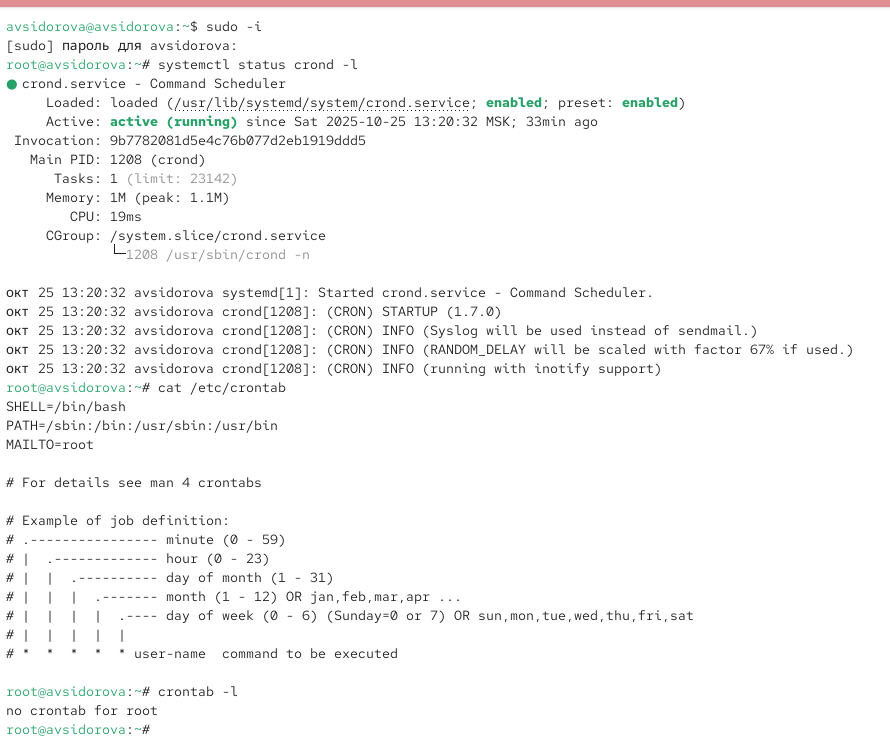


Рис. 1: crontab -l

Откроем файл расписания на редактирование: crontab -e Команда запустит интерфейс редактора (по умолчанию используется vi). Добавим следующую строку в файл расписания (запись сообщения в системный журнал), используя Ins для перехода в vi в режим ввода: */1*  \* \* \* logger This message is written from root cron Закроем сеанс редактирования vi и сохраним изменения, используя команду vi (рис. 2)

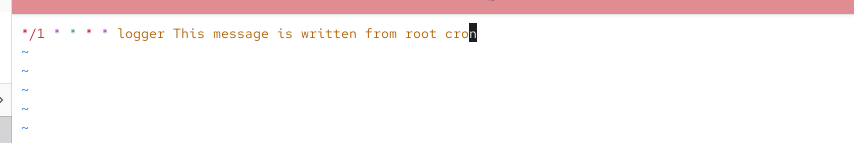


Рис. 2: */1*  \* \* \* logger This message is written from root cron

Посмотрим список заданий в расписании: crontab -l В расписании должна появиться запись о запланированном событии. (рис. 3)

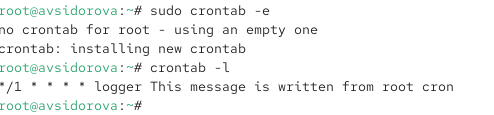


Рис. 3: crontab -l

Не выключая систему, через некоторое время (2–3 минуты) просмотрим журнал системных событий: grep written /var/log/messages В отчёте отразим результат. (рис. 4)

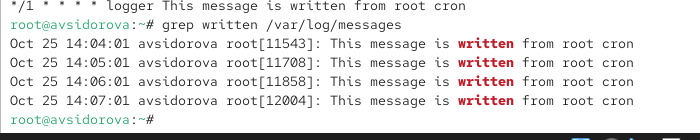


Рис. 4: Результат

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

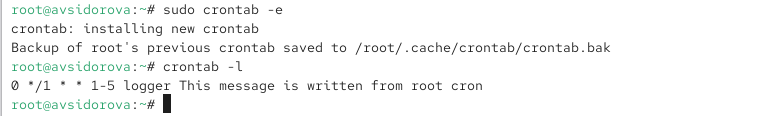


Рис. 5: 0 */1*  \* 1-5 logger This message is written from root cron

Перейдем в каталог /etc/cron.hourly и создайте в нём файл сценария с именем eachhour: cd /etc/cron.hourly touch eachhour (рис. 6)

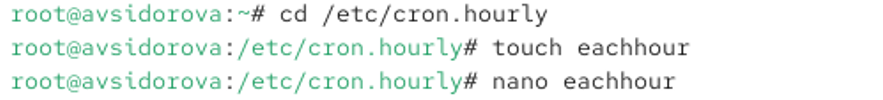


Рис. 6: eachhour

Откроем файл eachhour для редактирования и пропишем в нём следующий скрипт (запись сообщения в системный журнал): #!/bin/sh logger This message is written at $(date) (рис. 7)

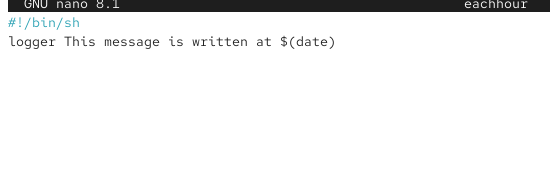


Рис. 7: Прописываем сценарий

Сделаем файл сценария eachhour исполняемым: chmod +x eachhour

Теперь перейдем в каталог /etc/crond.d и создадим в нём файл с расписанием eachhour: cd /etc/cron.d touch eachhour (рис. 8)

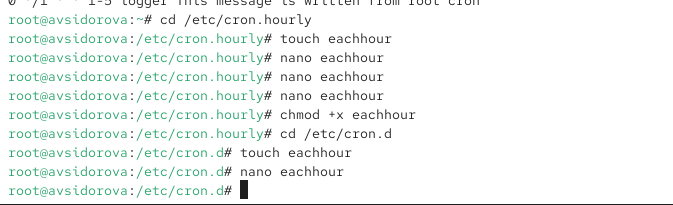


Рис. 8: Создаем файл с расписанием

Откроем этот файл для редактирования и поместим в него следующее содержимое: 11 \* \* \* \* root logger This message is written from /etc/cron.d (рис. 9)

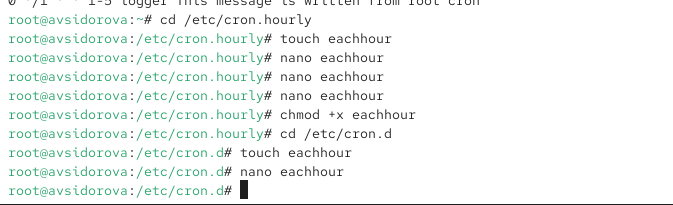


Рис. 9: Вписываем содержимое

## 2.2 Планирование заданий с помощью at

Запустим терминал и получим полномочия администратора: su -

Проверим, что служба atd загружена и включена: systemctl status atd (рис. 10)

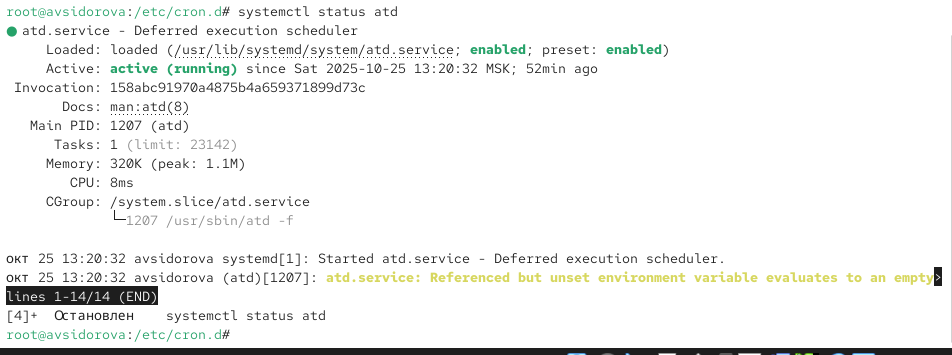


Рис. 10: Проверим, что служба atd загружена и включена

Зададим выполнение команды logger message from at в 9:30 (или замените на любое другое время, когда вы работаете над этим упражнением). Для этого введите at 9:30 Затем введем logger message from at Используем Ctrl + d , чтобы закрыть оболочку.

Убедимся, что задание действительно запланировано: atq С помощью команды grep ‘from at’ /var/log/messages посмотрим, появилось ли соответствующее сообщение в лог-файле в указанное нами время. (рис. 11)

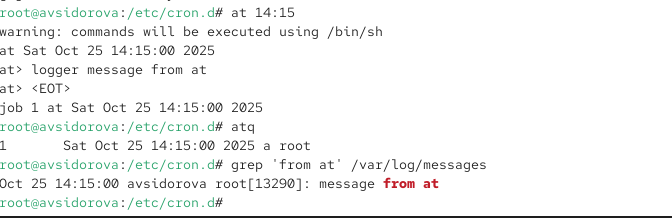


Рис. 11: Устанавливаем сообщение на 14:15 и проверяем исполняемость

# 3 Ответы на контрольные вопросы

1. 0 0 */14*  \* команда или через каталог cron.weekly со скриптом
2. 0 2 1,15 \* \* команда
3. */2*  \* \* \* команда
4. 0 0 19 9 \* команда
5. 0 0 \* 9 4 команда
6. crontab -u alice -e
7. Добавить пользователя bob в файл /etc/cron.deny
8. Использовать anacron или настроить повторение задания с запасом времени
9. atq или at -l

# 4 Выводы

Получили навыки работы с планировщиками событий cron и at.