Лабараторная работа №8

Отчет

Славинский Владислав Вадимович

Содержание

# 1 Цель работы

Получить навыки работы с планировщиками событий cron и at.

# 2 Выполнение лабораторной работы

Запустим терминал и получим полномочия администратора. (рис. 1)

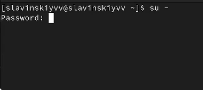


Рис. 1: Переход в режим суперпользователя

Посмотрим статус демона crond: systemctl status crond -l. Как видим, служба активна.(рис. 2)

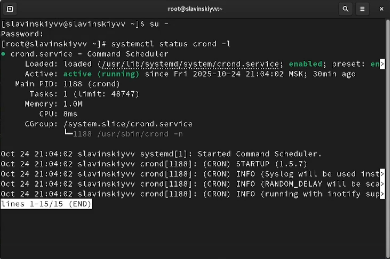


Рис. 2: Проверка статуса crond

Посмотрим содержимое файла конфигурации /etc/crontab: cat /etc/crontab. В файле нам показаны несколько переменных окрежения такие как SHELL, PATH, MAILTO, и приведены примеры расписаний для задач, которые выполняется ежечасно, ежедневно, еженедельно, ежемесячно. (рис. 3)

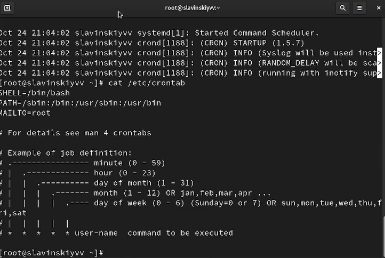


Рис. 3: Содержимое файла конфигурации

Посмотрим список заданий в расписании: crontab -l. И замечаем, что у нас пусто, так как расписание еще не задано. (рис. 4)

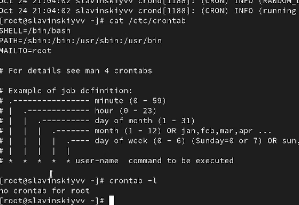


Рис. 4: Список заданий

Откроем файл расписания на редактирование (crontab -e) и введем в нем слудующую команду: */1*  \* \* \* logger This message is written from root cron. Синтаксис записи в crontab у нас таков: минута, час, день месяца, месяц, день недели. В нашем случае */1 - это выполнение каждую минуту, а*  просто каждый, т.е. каждый час, каждый день и т.д. (рис. 5)

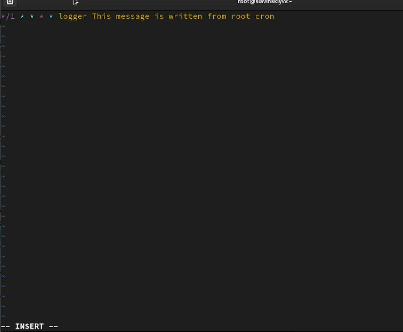


Рис. 5: Редактирование файла расписания

Посмотрим список заданий в расписании: crontab -l. В расписании у нас появилась запись о запланированном событии. (рис. 6)

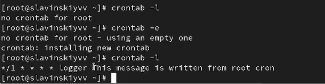


Рис. 6: Проверка списка заданий в расписании

Теперь оставим нашу систему на 2-3 минуты и потом посмотрим журнал системных событий: grep written /var/log/messages. И как видим, у нас повторяются записи в журнале с интервалом в 1 минуту. (рис. 7)

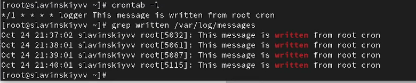


Рис. 7: Журнал системных событий

Изменим запись в расписании crontab на следующую: 0 */1*  \* 1-5 logger This message is written from root cron. В этом же случае при 0 будет просиходить выполнение в нулевю минуту каждого часа, \*/1 - это выполнение каждый час, а 1-5 это выполнение с понедельника по пятницу.(рис. 8)

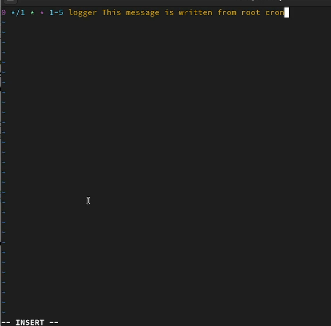


Рис. 8: Новая запись

Посмотрим список заданий в расписании: crontab -l.(рис. 9)

Просмотр списка заданий в расписании

Рис. 9: Просмотр списка заданий в расписании

Перейдем в каталог /etc/cron.hourly и создадим в нём файл сценария с именем eachhour: cd /etc/cron.hourly, touch eachhour. (рис. 10)

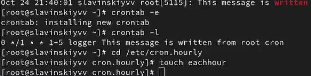


Рис. 10: Создание сценария eachhour

Откроем eachhour для редактирования и пропишем в нем: #!/bin/sh, logger This message is written at $(date). (рис. 11)

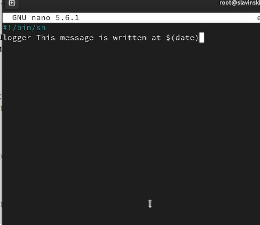


Рис. 11: Редактирование eachhour

Сделаем файл сценария eachhour исполняемым: chmod +x eachhour. (рис. 12)

Становление файла eachour испольняемым

Рис. 12: Становление файла eachour испольняемым

Перейдем в каталог /etc/crond.d и создадим в нём файл с расписанием eachhour: cd /etc/cron.d, touch eachhour. (рис. 13)

Создание файла с расписанием в каталоге /etc/crond.d

Рис. 13: Создание файла с расписанием в каталоге /etc/crond.d

Откроем файл для редактировани и пропишем следующее: 11 \* \* \* \* root logger This message is written from /etc/cron.d. Здесь же у нас будет выполняться в 11-ую минуту каждого часа и будет выполняться каждый час, каждый день, каждый месяц и каждый день недели.(рис. 14)

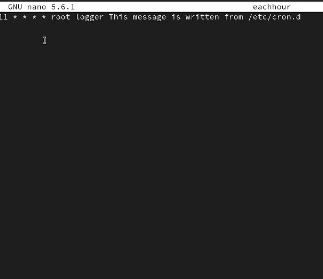


Рис. 14: Редактирование eachhour в cron.d

Теперь оставим нашу систему на 2 часа и потом посмотрим в журнал системных событий: grep written /var/log/messages. И видим, что запуск сценария eachhour был осуществлен, так как мы видим, что у нас есть записи, сгенерированные сценарием eachhour: is written at Fri Oct 24 10:01:01 PM MSK 2025. (рис. 15)

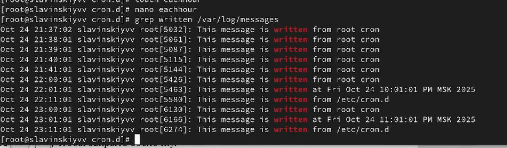


Рис. 15: Журнал системных событий

Проверим, что служба atd загружена и включена: systemctl status atd. Как видим, все работает.(рис. 16)

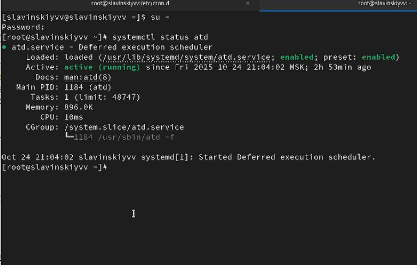


Рис. 16: Проверка службы atd

Зададим выполнение команды logger message from at в 23:59. Далее введем logger message from at и используем ctrl+d для закрытия оболочки.(рис. 17)

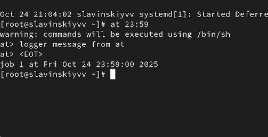


Рис. 17: Настройка выполнения команды

Проверяем, что задание запланировано: atq. Видим, что оно запланировано.(рис. 18)

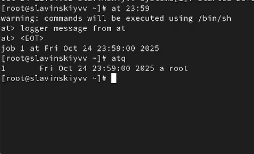


Рис. 18: Проверка запланировано ли задание

И в 23:59 вводим команду grep ‘from at’ /var/log/messages, чтобы проверить, появилось ли сообщение в нужное нам время. И как видим, у нас все работает.(рис. 19)

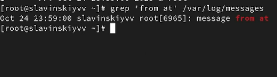


Рис. 19: Проверка сообщения

# 3 Выводы

В ходе выполнения лабораторной работы были получены навыки работы с планировщиками событий cron и at.

# 4 Ответы на контрольные вопросы

1. 0 0 1,15 \* \* выполнение 1-ого и 15-го числа каждый месяц
2. 0 2 1,15 \* \* нулевая минута, 2:00 ночи, 1,15 числа месяца, любой месяц, любой день недели
3. */2*  \* \* \* каждые две минуты
4. 0 0 19 9 \* в 00:00, 19-ое число, сентабрь, любой день недели
5. 0 0 \* 9 4 в 00:00, каждый день месяца, сентябрь, четверг
6. crontab -u alice -e, 0 9 \* \* 1-5 /home/scripts/daily\_report.sh, crontab -u alice -l (рис. 20)

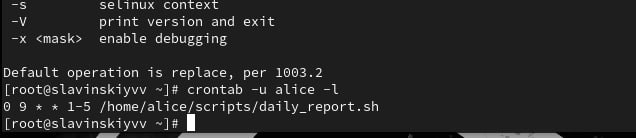


Рис. 20: Подтверждающий пример