

Лабораторная работа 8

1132232887

Накова Амина Михайловна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	10
	Список литературы	11

Список иллюстраций

2.1 шаг 1	6
2.2 шаг 2	7
2.3 шаг 3	7
2.4 шаг 4	8
2.5 шаг 5	8
2.6 шаг 6	8
2.7 шаг 7	8
2.8 шаг 8	8
2.9 шаг 9	9

Список таблиц

1 Цель работы

Целью данной работы является получение навыков работы с планировщиками событий cron и at.

2 Выполнение лабораторной работы

Планирование задач с помощью cron: Мониторинг журнала системных событий в реальном времени: Запустим терминал и получим полномочия администратора: `su -`. Просмотрим статус демона `crond`: `systemctl status crond -l` и содержимое файла конфигурации `/etc/crontab`: `cat /etc/crontab`

```
[nakova@localhost ~]$ su nakova
Пароль:
[nakova@localhost ~]$ systemctl status crond -l
● crond.service - Command Scheduler
   Loaded: loaded (/usr/lib/systemd/system/crond.service; enabled; preset: en
   Active: active (running) since Mon 2025-02-17 03:17:34 MSK; 7min ago
     Main PID: 919 (crond)
       Tasks: 1 (limit: 23024)
      Memory: 1.0M
         CPU: 41ms
    CGroup: /system.slice/crond.service
            └─919 /usr/sbin/crond -n

фев 17 03:17:34 localhost.localdomain systemd[1]: Started Command Scheduler.
фев 17 03:17:34 localhost.localdomain crond[919]: (CRON) STARTUP (1.5.7)
фев 17 03:17:34 localhost.localdomain crond[919]: (CRON) INFO (Syslog will be u
фев 17 03:17:34 localhost.localdomain crond[919]: (CRON) INFO (RANDOM_DELAY wil
фев 17 03:17:34 localhost.localdomain crond[919]: (CRON) INFO (running with ino
lines 1-15/15 (END)
```

Рис. 2.1: шаг 1

Теперь посмотрим список заданий в расписании: `crontab -l`. Ничего не отобразилось, так как расписание ещё не задано. Далее откроем файл расписания на редактирование: `crontab -e`

```

фев 17 03:17:34 localhost.localdomain crond[919]: (CRON) INFO (Syslog will be up
фев 17 03:17:34 localhost.localdomain crond[919]: (CRON) INFO (RANDOM_DELAY will
фев 17 03:17:34 localhost.localdomain crond[919]: (CRON) INFO (running with ino
[nakova@localhost ~]$ cat /etc/crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root

# For details see man 4 crontabs

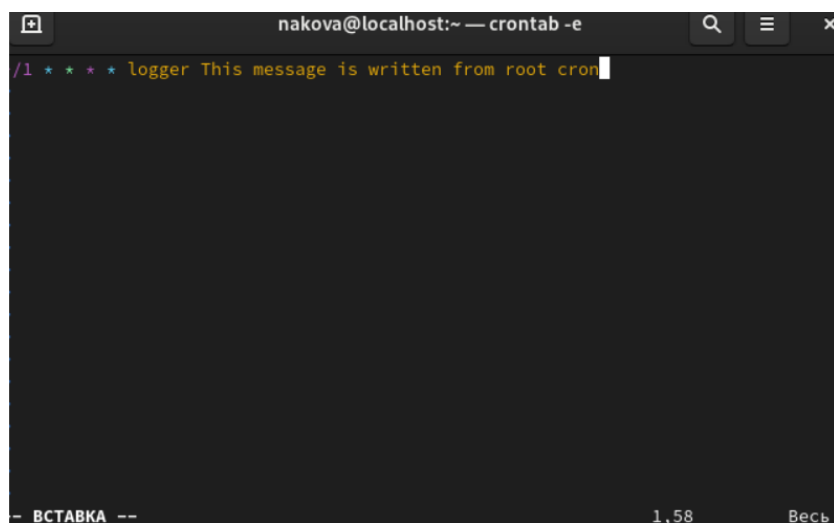
# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,f
ri,sat
# | | | | |
# * * * * * user-name  command to be executed

[nakova@localhost ~]$

```

Рис. 2.2: шаг 2

Предыдущая команда запустила 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: `Esc : wq`



```

nakova@localhost:~ — crontab -e
/1 * * * * logger This message is written from root cron

```

Рис. 2.3: шаг 3

Посмотрим список заданий в расписании: `crontab -l` (в расписании появилась запись о запланированном событии). Не выключая систему, через некоторое время (2–3 минуты) посмотрим журнал системных событий: `grep written`

```
/var/log/messages
```

```
crontab: installing new crontab
[nakova@localhost ~]$ crontab -l
*/1 * * * * logger This message is written from root cron
[nakova@localhost ~]$
```

Рис. 2.4: шаг 4

Вернёмся в текстовый редактор vi и изменим запись в расписании crontab на следующую: 0/1 * 1-5 logger This message is written from root cron

```
0 */1 * * 1-5 logger This message is written from root cron
```

Рис. 2.5: шаг 5

Теперь посмотрим список заданий в расписании: `crontab -l`

```
0 */1 * * 1-5 logger This message is written from root cron
```

Рис. 2.6: шаг 6

Перейдём в каталог `/etc/cron.hourly` и создадим в нём файл сценария с именем `eachhour` (Рис. 1.7): `cd /etc/cron.hourly touch eachhour` Далее откроем файл `eachhour` для редактирования и пропишем в нём следующий скрипт (запись сообщения в системный журнал) (Рис. 1.8): `#!/bin/sh logger This message is written at $(date)`

```
touch eachhour
mccedit eachhour
```

Рис. 2.7: шаг 7

шаг 8

Рис. 2.8: шаг 8

Сделаем файл сценария eachhour исполняемым: `chmod +x eachhour` Теперь перейдём в каталог `/etc/crond.d` и создадим в нём файл с расписанием eachhour (Рис. 1.9): `cd /etc/cron.d touch eachhour` Откроем этот файл для редактирования и поместим в него следующее содержимое: `11 * * * * root logger This message is written from /etc/cron.d` Сохраним изменения

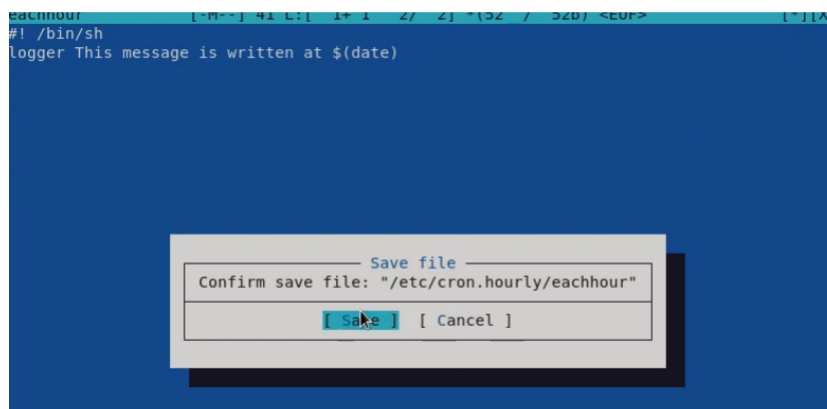


Рис. 2.9: шаг 9

Ответы на контрольные вопросы: 1. Как настроить задание cron, чтобы оно выполнялось раз в 2 недели? `00 00 1,15 * * logger task` 2. Как настроить задание cron, чтобы оно выполнялось 1-го и 15-го числа каждого месяца в 2 часа ночи? `00 02 1,15 * * logger task` 3. Как настроить задание cron, чтобы оно выполнялось каждые 2 минуты каждый день? `/2 * * * logger task` 4. Как настроить задание cron, чтобы оно выполнялось 19 сентября ежегодно? `* * 19 9 logger task` 5. Как настроить задание cron, чтобы оно выполнялось каждый четверг сентября ежегодно? `* * * 4 logger task` 6. Какая команда позволяет вам назначить задание cron для пользователя alice? Приведите подтверждающий пример. `* * * * alice logger task` 7. Как указать, что пользователю bob никогда не разрешено назначать задания через cron? Приведите подтверждающий пример. записать его в `/etc/cron.deny` 8. Вам нужно убедиться, что задание выполняется каждый день, даже если сервер во время выполнения временно недоступен. Как это сделать? Найти задание в логах `grep cron /var/log/messages` 9. Какая команда позволяет узнать, запланированы ли какие-либо задания на выполнение планировщиком atd? `atq`

3 Выводы

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

Список литературы