

Отчёт по лабораторной работе №8

Планировщики событий

Эзиз Хатамов

Содержание

1	Цель работы	5
2	Отчёт по выполнению работы	6
2.1	Планирование заданий с помощью cron	6
2.2	Планирование заданий с помощью at	10
3	Контрольные вопросы	12
4	Заключение	15

Список иллюстраций

2.1	Содержимое файла <code>/etc/crontab</code>	7
2.2	Создание задания <code>cron</code> с интервалом в 1 минуту	8
2.3	Проверка выполнения <code>cron</code> -заданий в системном журнале	8
2.4	Редактирование расписания <code>cron</code> на выполнение по рабочим дням	9
2.5	Создание сценария <code>eachhour</code> в <code>/etc/cron.hourly</code>	9
2.6	Создание задания в <code>/etc/cron.d</code>	10
2.7	Проверка выполнения задания <code>at</code> в системном журнале	11

Список таблиц

1 Цель работы

Получение навыков работы с планировщиками событий cron и at.

2 Отчёт по выполнению работы

2.1 Планирование заданий с помощью cron

1. В терминале были получены права суперпользователя с помощью команды **su -**.

Затем был проверен статус службы планировщика заданий **crond**.

Команда **systemctl status crond -l** показала, что служба активна и работает корректно.

2. Далее было просмотрено содержимое конфигурационного файла **/etc/crontab**.

Файл содержит переменные среды, комментарии и пример формата задания.

```

ehatamov@ehatamov:~$ su
Password:
root@ehatamov:/home/ehatamov# systemctl status crond -l
● crond.service - Command Scheduler
   Loaded: loaded (/usr/lib/systemd/system/crond.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-10-08 13:28:01 MSK; 2min 15s ago
  Invocation: 3ed3e08db28e4d58ad7f9cdd28cdfd10
    Main PID: 1220 (crond)
      Tasks: 1 (limit: 24779)
     Memory: 1M (peak: 1.1M)
        CPU: 7ms
    CGroup: /system.slice/crond.service
            └─1220 /usr/sbin/crond -n

Oct 08 13:28:01 ehatamov.localdomain systemd[1]: Started crond.service - Command Scheduler.
Oct 08 13:28:01 ehatamov.localdomain crond[1220]: (CRON) STARTUP (1.7.0)
Oct 08 13:28:01 ehatamov.localdomain crond[1220]: (CRON) INFO (Syslog will be used instead of sendmail.)
Oct 08 13:28:01 ehatamov.localdomain crond[1220]: (CRON) INFO (RANDOM_DELAY will be scaled with factor 72% if used.)
Oct 08 13:28:01 ehatamov.localdomain crond[1220]: (CRON) INFO (running with inotify support)
root@ehatamov:/home/ehatamov# 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,fri,sat
# | | | | |
# * * * * * user-name    command to be executed

root@ehatamov:/home/ehatamov# █

```

Рис. 2.1: Содержимое файла /etc/crontab

3. Проверка текущего расписания пользователя **root** командой **crontab -l** показала отсутствие заданий.
4. Через команду **crontab -e** было добавлено задание, выполняющее запись в системный журнал каждую минуту:

/1 * * * logger This message is written from root cron*.

Синтаксис выражения означает:

- ***/1** — выполнение каждые 1 минуту;
- ***** — каждый час, день, месяц и день недели;
- команда **logger** записывает сообщение в системный журнал.

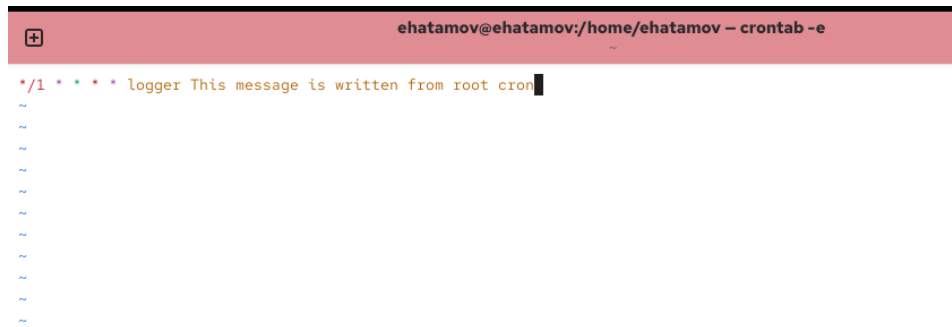


Рис. 2.2: Создание задания cron с интервалом в 1 минуту

5. После сохранения изменений команда **crontab -l** подтвердила наличие записи в расписании.

Через несколько минут проверка системного журнала показала выполнение задачи:

grep written /var/log/messages.

Сообщения с текстом *“This message is written from root cron”* появлялись с интервалом в одну минуту.

```
root@ehatamov:/home/ehatamov# crontab -l
no crontab for root
root@ehatamov:/home/ehatamov# crontab -e
no crontab for root - using an empty one
crontab: installing new crontab
root@ehatamov:/home/ehatamov# crontab -l
*/1 * * * * logger This message is written from root cron
root@ehatamov:/home/ehatamov# grep written /var/log/messages
root@ehatamov:/home/ehatamov# grep written /var/log/messages
Oct 8 13:33:01 ehatamov root[4144]: This message is written from root cron
root@ehatamov:/home/ehatamov# grep written /var/log/messages
Oct 8 13:33:01 ehatamov root[4144]: This message is written from root cron
Oct 8 13:34:01 ehatamov root[4301]: This message is written from root cron
Oct 8 13:35:01 ehatamov root[4439]: This message is written from root cron
root@ehatamov:/home/ehatamov#
```

Рис. 2.3: Проверка выполнения cron-заданий в системном журнале

6. Запись в расписании была изменена на:

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

Здесь указано:

- 0 */1 — запуск каждый час в 0-й минуте;

- 1-5 — выполнение только по рабочим дням (с понедельника по пятницу).

```
ehatamov@ehatamov:/home/ehatamov - crontab -e
* */1 * * 1-5 logger This message is written from root cron
```

Рис. 2.4: Редактирование расписания cron на выполнение по рабочим дням

7. В каталоге **/etc/cron.hourly** был создан файл сценария **eachhour**.

В нём был размещён скрипт, выполняющий запись в системный журнал:

logger This message is written at \$(date).

Скрипт делает запись в системный журнал с указанием текущей даты и времени.

```
ehatamov@ehatamov:/etc/cron.hourly - mc /etc/cron.hourly
eachhour [----] 41 L:[ 1+ 1 2/ 2] *(51 / 51b) <EOF>
#!/bin/sh
logger This message is written at $(date)
```

Рис. 2.5: Создание сценария eachhour в /etc/cron.hourly

8. Для задания, запускаемого из каталога **/etc/cron.d**, был создан файл **eachhour**, содержащий строку:

11 * * * * root logger This message is written from /etc/cron.d.

Синтаксис означает:

- выполнение в **11-й минуте** каждого часа;
- от имени пользователя **root**;
- запуск команды **logger**.

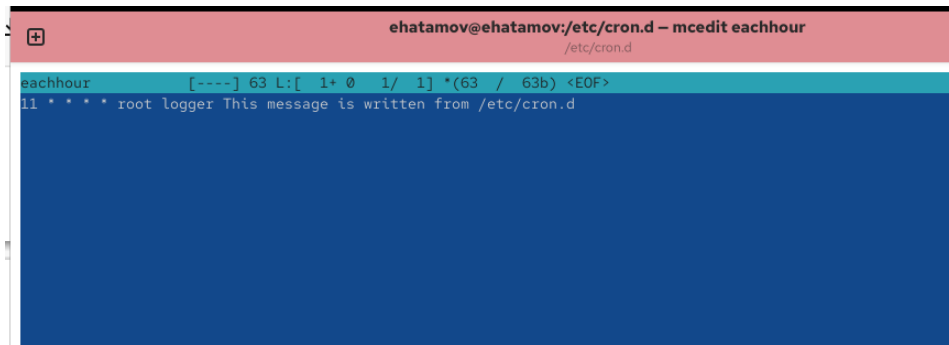


Рис. 2.6: Создание задания в /etc/cron.d

2.2 Планирование заданий с помощью at

1. Проверка службы **atd** показала, что она активна и запущена.
2. Было создано однократное задание с помощью команды **at 13:43**.
В интерактивной оболочке введена команда **logger message from at**.
После закрытия ввода сочетанием **Ctrl + D** задание было добавлено в очередь.
3. Проверка командой **atq** подтвердила наличие задания, запланированного на 13:43.
4. После указанного времени команда **grep 'from at' /var/log/messages** показала запись *"message from at"*, подтверждающую успешное выполнение задания.

```

root@ehatamov:/etc/cron.d# cat eachhour
11 * * * * root logger This message is written from /etc/cron.droot@ehatamov:/etc/cron.d#
root@ehatamov:/etc/cron.d#
root@ehatamov:/etc/cron.d# systemctl status atd
● atd.service - Deferred execution scheduler
   Loaded: loaded (/usr/lib/systemd/system/atd.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-10-08 13:28:01 MSK; 12min ago
  Invocation: a4861525e8594d608a8e763c8b144652
     Docs: man:atd(8)
    Main PID: 1219 (atd)
      Tasks: 1 (limit: 24779)
     Memory: 328K (peak: 1.1M)
        CPU: 4ms
    CGroup: /system.slice/atd.service
            └─1219 /usr/sbin/atd -f

Oct 08 13:28:01 ehatamov.localdomain systemd[1]: Started atd.service - Deferred execution scheduler.
Oct 08 13:28:01 ehatamov.localdomain (atd)[1219]: atd.service: Referenced but unset environment variable evaluates to an empty str
root@ehatamov:/etc/cron.d# at 13:43
warning: commands will be executed using /bin/sh
at Wed Oct 8 13:43:00 2025
at> logger message from at
at> <EOT>
job 1 at Wed Oct 8 13:43:00 2025
root@ehatamov:/etc/cron.d# atq
1          Wed Oct 8 13:43:00 2025 a root
root@ehatamov:/etc/cron.d# atq
root@ehatamov:/etc/cron.d# grep 'from at' /var/log/messages
Oct 8 13:43:00 ehatamov root[6044]: message from at
root@ehatamov:/etc/cron.d#

```

Рис. 2.7: Проверка выполнения задания at в системном журнале

3 Контрольные вопросы

1. Как настроить задание cron, чтобы оно выполнялось раз в 2 недели?

- В cron отсутствует прямой параметр для задания интервала в 2 недели.
- Можно использовать выполнение по дню недели и неделе месяца:
`0 0 */14 * * <команда>` — запуск каждые 14 дней в полночь.
- Альтернатива — создать скрипт, который сам проверяет дату и выполняет действия раз в две недели.

2. Как настроить задание cron, чтобы оно выполнялось 1-го и 15-го числа каждого месяца в 2 часа ночи?

- `0 2 1,15 * * <команда>`
- Здесь 0 — минуты, 2 — час (2:00), 1, 15 — дни месяца, * * — каждый месяц и любой день недели.

3. Как настроить задание cron, чтобы оно выполнялось каждые 2 минуты каждый день?

- `*/2 * * * * <команда>`
- Символ */2 в поле минут задаёт выполнение каждые 2 минуты.

4. Как настроить задание cron, чтобы оно выполнялось 19 сентября ежегодно?

- 0 0 19 9 * <команда>

- Выполнение 19-го числа 9-го месяца (сентября) в полночь каждого года.

5. **Как настроить задание cron, чтобы оно выполнялось каждый четверг сентября ежегодно?**

- 0 0 * 9 4 <команда>

- 0 0 — запуск в полночь, * — каждый день месяца, 9 — сентябрь, 4 — четверг (номер дня недели).

6. **Какая команда позволяет вам назначить задание cron для пользователя alice? Приведите подтверждающий пример.**

- `crontab -u alice -e` — открыть файл расписания пользователя **alice** для редактирования.

- Пример: `crontab -u alice -e` и добавление строки `0 8 * * * echo "Good morning" >> /home/alice/log.txt`.

7. **Как указать, что пользователю bob никогда не разрешено назначать задания через cron? Приведите подтверждающий пример.**

- Для этого используется файл **/etc/cron.deny**.

- В него добавляется имя пользователя:
`echo "bob" >> /etc/cron.deny`

- После этого пользователь **bob** не сможет использовать команду `crontab`.

8. **Вам нужно убедиться, что задание выполняется каждый день, даже если сервер во время выполнения временно недоступен. Как это сделать?**

- Следует использовать **anacron**, который выполняет пропущенные задания после восстановления системы.
- Конфигурация находится в файле **/etc/anacrontab**, где можно задать ежедневное выполнение.

9. **Какая команда позволяет узнать, запланированы ли какие-либо задания на выполнение планировщиком atd?**

- **atq** — отображает очередь заданий, запланированных с помощью **at**.
- В выводе будут показаны идентификаторы заданий, время запуска и пользователь, создавший их.

4 Заключение

В ходе работы были изучены механизмы планирования заданий в Linux с использованием утилит **cron** и **at**.

В результате освоены базовые приёмы автоматизации системных процессов и администрирования в среде Linux.