

# **Отчёт по лабораторной работе №8**

**Планировщики событий**

Шаханеоядж Хаоладар

# **Содержание**

<b>1 Цель работы</b>	<b>5</b>
<b>2 Выполнение</b>	<b>6</b>
2.1 Планирование задач с помощью cron и at . . . . .	6
<b>3 Контрольные вопросы</b>	<b>11</b>
<b>4 Заключение</b>	<b>13</b>

# **Список иллюстраций**

2.1 Проверка статуса службы crond . . . . .	6
2.2 Содержимое файла /etc/crontab . . . . .	7
2.3 Создание нового задания cron через crontab -e . . . . .	7
2.4 Проверка расписания и результатов выполнения задания . . . . .	8
2.5 Изменённая запись crontab . . . . .	8
2.6 Создание сценария eachhour в /etc/cron.hourly . . . . .	9
2.7 Создание задания eachhour в /etc/cron.d . . . . .	9
2.8 Проверка статуса службы atd . . . . .	10

# **Список таблиц**

# **1 Цель работы**

Получение навыков работы с планировщиками событий cron и at.

## 2 Выполнение

### 2.1 Планирование задач с помощью cron и at

1. В терминале был получен доступ администратора с помощью команды **su -**.

После этого проверен статус службы **crond**, отвечающей за планирование периодических заданий.

```
haoladar@haoladar:~$ su
Password:
root@haoladar:/home/haoladar#
root@haoladar:/home/haoladar# systemctl status crond
● crond.service - Command Scheduler
   Loaded: loaded (/usr/lib/systemd/system/crond.service; enabled; preset: enabled)
   Active: active (running) since Wed 2025-10-08 15:04:25 MSK; 15min ago
     Invocation: eae2c9b7975144a796e43c1542f4dd3a
      Main PID: 1202 (crond)
        Tasks: 1 (limit: 24779)
       Memory: 1M (peak: 1.1M)
          CPU: 6ms
         CGroup: /system.slice/crond.service
                   └─1202 /usr/sbin/crond -n

Oct 08 15:04:25 haoladar.localdomain systemd[1]: Started crond.service - Command Scheduler.
Oct 08 15:04:25 haoladar.localdomain crond[1202]: (CRON) STARTUP (1.7.0)
Oct 08 15:04:25 haoladar.localdomain crond[1202]: (CRON) INFO (Syslog will be used instead of sendmail.)
Oct 08 15:04:25 haoladar.localdomain crond[1202]: (CRON) INFO (RANDOM_DELAY will be scaled with factor 85%)
Oct 08 15:04:25 haoladar.localdomain crond[1202]: (CRON) INFO (running with inotify support)
```

Рис. 2.1: Проверка статуса службы crond

Из вывода видно, что служба **crond.service** находится в состоянии **active (running)** и запущена автоматически при старте системы.

2. Для ознакомления с системным файлом расписаний был просмотрен файл **/etc/crontab**.

Он содержит переменные окружения и пример формата задания в cron.

```

root@haoladar:/home/haoladar# 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@haoladar:/home/haoladar#

```

Рис. 2.2: Содержимое файла /etc/crontab

Каждое задание в этом файле имеет шесть полей: минута, час, день месяца, месяц, день недели и пользователь, от имени которого выполняется команда.

Такой формат определяет расписание и пользователя, от имени которого выполняется задача.

3. С помощью команды **crontab -e** было открыто расписание текущего пользователя и добавлена строка

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

Эта запись означает запуск команды **logger** каждую минуту.

```

haoladar@haoladar:/home/haoladar - crontab -e

*/1 * * * * logger This message is written from root cron
~
~
~
~
~
~
~
```

Рис. 2.3: Создание нового задания cron через crontab -e

4. После сохранения изменений список заданий был проверен с помощью **crontab -l**.

В нём отобразилась добавленная запись.

```
root@haoladar:/home/haoladar# crontab -l
no crontab for root
root@haoladar:/home/haoladar# crontab -e
no crontab for root - using an empty one
crontab: installing new crontab
root@haoladar:/home/haoladar# crontab -l
*/1 * * * * logger This message is written from root cron
root@haoladar:/home/haoladar# grep written /var/log/messages
root@haoladar:/home/haoladar# grep written /var/log/messages
Oct  8 15:22:01 haoladar root[4341]: This message is written from root cron
Oct  8 15:23:01 haoladar root[4468]: This message is written from root cron
Oct  8 15:24:01 haoladar root[4583]: This message is written from root cron
root@haoladar:/home/haoladar#
```

Рис. 2.4: Проверка расписания и результатов выполнения задания

В журнале **/var/log/messages** появились записи, подтверждающие выполнение задачи каждую минуту:

*This message is written from root cron.*

5. Затем расписание было изменено на

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

Эта запись означает выполнение задания каждый час, с понедельника по пятницу.



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

Рис. 2.5: Изменённая запись crontab

6. Для автоматического выполнения скрипта каждый час был создан сценарий **/etc/cron.hourly/eachhour**.

haoladar@haoladar:/etc/cron.hourly – mcedit eachhour  
/etc/cron.hourly

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

Рис. 2.6: Создание сценария eachhour в /etc/cron.hourly

Его содержимое:

```
#!/bin/sh
logger This message is written at $(date)
```

После создания скрипта был сделан исполняемым с помощью команды **chmod +x eachhour**.

7. Также был создан файл задания **/etc/cron.d/eachhour** со следующим содержимым:

haoladar@haoladar:/etc/cron.d – mcedit eachhour  
/etc/cron.d

```
eachhour      [-M--] 63 L:[ 1+ 0 1/ 1] *(63 / 63b) <EOF>
11 * * * * root logger This message is written from /etc/cron.d
```

Рис. 2.7: Создание задания eachhour в /etc/cron.d

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

Эта строка означает, что каждые сутки в 11-й минуте каждого часа команда **logger** будет запускаться от имени пользователя **root**.

8. Для выполнения одноразового задания использовался планировщик **at**.

Сначала была проверена работа службы **atd**.

```
root@haoladar:/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 15:04:25 MSK; 25min ago
    Invocation: a1fac4918b8b4635988189b8ce936381
      Docs: man:atd(8)
   Main PID: 1199 (atd)
     Tasks: 1 (limit: 24779)
    Memory: 320K (peak: 1.1M)
      CPU: 2ms
     CGroup: /system.slice/atd.service
             └─1199 /usr/sbin/atd -f

Oct 08 15:04:25 haoladar.localdomain systemd[1]: Started atd.service - Deferred execution scheduler.
Oct 08 15:04:25 haoladar.localdomain (atd)[1199]: atd.service: Referenced but unset environment variable ev>
root@haoladar:/etc/cron.d# at 15:31
warning: commands will be executed using /bin/sh
at Wed Oct  8 15:31:00 2025
at> logger message from at
at> <EOT>
job 1 at Wed Oct  8 15:31:00 2025
root@haoladar:/etc/cron.d# atq
1      Wed Oct  8 15:31:00 2025 a root
root@haoladar:/etc/cron.d# grep 'from at' /var/log/messages
Oct  8 15:31:00 haoladar root[5835]: message from at
root@haoladar:/etc/cron.d#
```

Рис. 2.8: Проверка статуса службы atd

Она также находится в состоянии **active (running)**.

Далее было запланировано задание на определённое время с помощью **at 15:31** и команды **logger message from at**.

После чего выполнена проверка очереди заданий **atq** и просмотр журнала сообщений **grep 'from at' /var/log/messages**.

В результате в журнале появилась запись: *message from at*, что подтверждает успешное выполнение задания в назначенное время.

### 3 Контрольные вопросы

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

Задания cron можно запускать по дням месяца или неделям. Так как напрямую интервал «раз в 2 недели» не поддерживается, используется комбинация проверки дня недели и числа месяца.

Пример: **0 0/14 \* команда** – выполняет задание каждые 14 дней в полночь.

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** – запуск каждый месяц 1-го и 15-го числа в 2:00.

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

**/2 \* \* \* \* команда** – выполнение каждые 2 минуты независимо от даты и времени суток.

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

**0 0 19 9 \* команда** – запуск ежегодно 19 сентября в 00:00.

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

**0 0 \* 9 4 команда** – запуск каждого четверга (день недели 4) в сентябре (месяц 9) в 00:00.

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

**crontab -u alice -e** — открыть и отредактировать расписание пользователя **alice**.

Пример записи: **0 8 \* \* \*** **/home/alice/backup.sh** — выполнение скрипта резервного копирования каждый день в 8:00 от имени alice.

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

В файл **/etc/cron.deny** нужно добавить имя пользователя **bob**.

Пример:

## 4 Заключение

В ходе работы были освоены основные принципы планирования заданий в Linux с использованием утилит **cron** и **at**.