

Отчёт по лабораторной работе №8

Планировщики событий

Руслан Алиев

Содержание

1 Цель работы	5
2 Выполнение	6
2.1 Работа с планировщиком cron	6
2.2 Работа с планировщиком at	10
3 Контрольные вопросы	12
4 Заключение	14

Список иллюстраций

2.1 Статус службы crond	6
2.2 Содержимое файла /etc/crontab	7
2.3 Создание задания в crontab	7
2.4 Результат выполнения задания cron	8
2.5 Изменённая запись в crontab	9
2.6 Создание сценария eachhour	9
2.7 Файл расписания в /etc/cron.d	10
2.8 Статус службы atd	11
2.9 Результат выполнения задания at	11

Список таблиц

1 Цель работы

Получение навыков работы с планировщиками событий cron и at.

2 Выполнение

2.1 Работа с планировщиком cron

1. В терминале был выполнен переход в режим суперпользователя с помощью команды **su -**.

Затем с помощью команды **systemctl status crond -l** был проверен статус службы **crond**, отвечающей за выполнение задач по расписанию.

```
raliev@raliev:~$ su
Password:
root@raliev:/home/raliev# cd
root@raliev:~#
root@raliev:~# 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 18:41:09 MSK; 6min ago
     Invocation: 83b13f3ece6b4340a5474444df8dfb4f
      Main PID: 1214 (crond)
        Tasks: 1 (limit: 24779)
       Memory: 1M (peak: 1.1M)
         CPU: 8ms
        CGroup: /system.slice/crond.service
                  └─1214 /usr/sbin/crond -n

Oct 08 18:41:09 raliev.localdomain systemd[1]: Started crond.service - Command Scheduler.
Oct 08 18:41:09 raliev.localdomain crond[1214]: (CRON) STARTUP (1.7.0)
Oct 08 18:41:09 raliev.localdomain crond[1214]: (CRON) INFO (Syslog will be used instead of sendmail..)
Oct 08 18:41:09 raliev.localdomain crond[1214]: (CRON) INFO (RANDOM_DELAY will be scaled with factor 78)
Oct 08 18:41:09 raliev.localdomain crond[1214]: (CRON) INFO (running with inotify support)
root@raliev:~#
```

Рис. 2.1: Статус службы crond

Служба **crond** находится в активном состоянии (*active (running)*), что подтверждает корректную работу планировщика задач.

2. Далее было просмотрено содержимое файла конфигурации **/etc/crontab**, в котором заданы глобальные параметры расписаний.

```

root@raliev:~# 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@raliev:~#

```

Рис. 2.2: Содержимое файла /etc/crontab

В файле определены переменные окружения:

- SHELL=/bin/bash – оболочка для выполнения заданий;
- PATH=/sbin:/bin:/usr/sbin:/usr/bin – путь к исполняемым файлам;
- MAILTO=root – адрес получателя уведомлений.

Также приведён шаблон структуры задания: * * * * * user-name command где пять звёздочек означают соответственно: минуту, час, день месяца, месяц и день недели.

3. Для редактирования личного расписания root использована команда **crontab -e**.

В файл добавлена строка:

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

```

raliev@raliev:~ - crontab -e
/root

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

Рис. 2.3: Создание задания в crontab

Пояснение синтаксиса:

- */1 — выполнение каждую минуту;
 - * — любое значение часов, дней и месяцев;
 - команда logger записывает сообщение в системный журнал /var/log/messages.
4. После сохранения изменений команда **crontab -l** подтвердила наличие записи в расписании.

Через несколько минут проверка с помощью **grep written /var/log/messages** показала, что сообщения действительно создаются каждую минуту.

```
root@raliev:~#  
root@raliev:~# crontab -l  
no crontab for root  
root@raliev:~# crontab -e  
no crontab for root - using an empty one  
crontab: installing new crontab  
root@raliev:~# crontab -l  
*/1 * * * * logger This message is written from root cron  
root@raliev:~# grep written /var/log/messages  
Oct  8 18:50:02 raliev root[4579]: This message is written from root cron  
root@raliev:~# grep written /var/log/messages  
Oct  8 18:50:02 raliev root[4579]: This message is written from root cron  
Oct  8 18:51:01 raliev root[4759]: This message is written from root cron  
Oct  8 18:52:01 raliev root[4898]: This message is written from root cron  
root@raliev:~#
```

Рис. 2.4: Результат выполнения задания cron

5. Затем задание было изменено на:

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



```
raliev@raliev:~ - crontab -e
/root

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

Рис. 2.5: Изменённая запись в crontab

Пояснение синтаксиса:

- 0 */1 — выполнение в начале каждого часа;
 - 1-5 — с понедельника по пятницу;
 - команда `logger` создаёт запись в журнале.
6. Далее был создан сценарий в каталоге **/etc/cron.hourly** с содержимым:

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



```
raliev@raliev:/etc/cron.hourly - nano eachhour
/etc/cron.hourly

GNU nano 8.1                                     eachhour
#!/bin/sh
logger This message is written at $(date)
```

Рис. 2.6: Создание сценария eachhour

Файл был сделан исполняемым командой **chmod +x eachhour**, что позволяет выполнять его каждый час автоматически.

7. В каталоге **/etc/cron.d** создан файл **eachhour** со следующим содержимым:

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



```
raliev@raliev:/etc/cron.d - nano eachhour
/etc/cron.d
GNU nano 8.1
eachhour
11 * * * * root logger This message is written from /etc/cron.d
```

Рис. 2.7: Файл расписания в /etc/cron.d

Пояснение синтаксиса:

- **11 * * * *** — выполнение на 11-й минуте каждого часа;
- **root** — пользователь, от имени которого выполняется команда;
- **logger** записывает сообщение в системный журнал.

2.2 Работа с планировщиком at

1. Для одноразового запуска задач используется служба **atd**.

Её статус был проверен с помощью **systemctl status atd**.

```

root@raliev:/etc/cron.d#
root@raliev:/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 18:41:09 MSK; 14min ago
  Invocation: 605ee8d29c1e4a6bbfbe461c7179576b
    Docs: man:atd(8)
   Main PID: 1213 (atd)
     Tasks: 1 (limit: 24779)
    Memory: 320K (peak: 1.1M)
      CPU: 4ms
     CGroup: /system.slice/atd.service
             └─1213 /usr/sbin/atd -f

Oct 08 18:41:09 raliev.localdomain systemd[1]: Started atd.service - Deferred execution scheduler.
Oct 08 18:41:09 raliev.localdomain (atd)[1213]: atd.service: Referenced but unset environment variable >
root@raliev:/etc/cron.d# at 18:58
warning: commands will be executed using /bin/sh
at Wed Oct  8 18:58:00 2025
at> logger message from at
at> <EOT>
job 1 at Wed Oct  8 18:58:00 2025
root@raliev:/etc/cron.d# atq
1      Wed Oct  8 18:58:00 2025 a root

```

Рис. 2.8: Статус службы atd

Служба находится в активном состоянии (*active (running)*), что позволяет использовать утилиту **at**.

- Командой **at 18:58** было запланировано выполнение задачи **logger message from at** на указанное время.

Проверка списка заданий командой **atq** подтвердила наличие одного активного задания.

- После наступления времени выполнения команда **grep 'from at' /var/log/messages** показала, что сообщение было успешно записано в системный журнал.

```

root@raliev:/etc/cron.d#
root@raliev:/etc/cron.d# grep 'from at' /var/log/messages
root@raliev:/etc/cron.d#
root@raliev:/etc/cron.d# grep 'from at' /var/log/messages
Oct  8 18:58:00 raliev root[6062]: message from at
root@raliev:/etc/cron.d# grep written /var/log/messages
Oct  8 18:50:02 raliev root[4579]: This message is written from root cron
Oct  8 18:51:01 raliev root[4759]: This message is written from root cron
Oct  8 18:52:01 raliev root[4898]: This message is written from root cron
Oct  8 19:00:01 raliev root[6347]: This message is written from root cron
Oct  8 19:01:01 raliev root[6491]: This message is written at Wed Oct 8 07:01:01 PM MSK 2025
root@raliev:/etc/cron.d#

```

Рис. 2.9: Результат выполнения задания at

3 Контрольные вопросы

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

Использовать комбинацию запуска каждое воскресенье с интервалом 14 дней:

0 0/14 * команда

или, если требуется более точный контроль, можно добавить условие через date в скрипте:

0 0 * * 0 [(((date +%U) % 2)) -eq 0] && команда

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

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

/2 * * * команда

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

0 0 19 9 * команда

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

0 0 * 9 4 команда

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

crontab -u alice -e

Пример:

crontab -u alice -e

и добавить строку:

0 9 * * * echo “Доброе утро, Alice” » /home/alice/cron.log

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

Добавить имя пользователя **bob** в файл **/etc/cron.deny**:

echo “bob” » /etc/cron.deny

После этого при попытке выполнить **crontab -e** от имени bob будет выдан отказ в доступе.

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

Использовать планировщик **anacron**, который выполняет пропущенные задания после восстановления работы системы.

Необходимо добавить задание в файл **/etc/anacrontab**.

Пример строки:

1 5 cron.daily nice run-parts /etc/cron.daily

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

atq

4 Заключение

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

Было рассмотрено создание периодических заданий, настройка расписаний различной сложности, а также работа с системными каталогами **/etc/cron.hourly** и **/etc/cron.d**.

Практически подтверждена корректная работа службы **crond**, обеспечивающей выполнение заданий по расписанию, и службы **atd**, отвечающей за одноразовые отложенные задания.