

Отчёт по лабораторной работе №8

Планировщики событий

Щемелев Илья Владимирович

Содержание

1	Цель работы	5
2	Ход выполнения	6
2.1	Планирование задач с помощью stop	6
2.2	Планирование заданий с помощью at	11
3	Контрольные вопросы	13
4	Заключение	16

Список иллюстраций

2.1	Статус службы crond	6
2.2	Содержимое файла /etc/crontab	7
2.3	Редактирование crontab	8
2.4	Список заданий crontab	8
2.5	Сообщения cron в журнале	9
2.6	Сценарий eachhour	10
2.7	Файл расписания в /etc/cron.d	10
2.8	Статус службы atd	11

Список таблиц

1 Цель работы

Получение навыков работы с планировщиками событий cron и at.

2 Ход выполнения

2.1 Планирование задач с помощью cron

1. Для выполнения операций администрирования был запущен терминал и получены полномочия суперпользователя с помощью команды `su -`.
Это обеспечило доступ к системным службам и файлам конфигурации, необходимым для настройки планировщика задач.
2. Выполнена проверка состояния демона планировщика задач `crond`.
В результате установлено, что служба `crond.service` загружена, включена в автозапуск и находится в состоянии `active (running)`, что подтверждает корректную работу планировщика задач.

```
root@ivschemelev:/home/ivschemelev#  
root@ivschemelev:/home/ivschemelev# systemctl status crond -l  
● crond.service - Command Scheduler  
   Loaded: loaded (/usr/lib/systemd/system/crond.service; enabled; preset: enabled)  
   Active: active (running) since Fri 2026-01-16 12:18:21 MSK; 1min 30s ago  
 Invocation: aa4fbee0016347e58540913b38217223  
    Main PID: 1429 (crond)  
       Tasks: 1 (limit: 23034)  
      Memory: 1M (peak: 1.3M)  
         CPU: 3ms  
    CGroup: /system.slice/crond.service  
            └─1429 /usr/sbin/crond -n  
  
Jan 16 12:18:21 ivschemelev.localdomain systemd[1]: Started crond.service - Command Scheduler.  
Jan 16 12:18:21 ivschemelev.localdomain crond[1429]: (CRON) STARTUP (1.7.0)  
Jan 16 12:18:21 ivschemelev.localdomain crond[1429]: (CRON) INFO (Syslog will be used instead of sendm>  
Jan 16 12:18:21 ivschemelev.localdomain crond[1429]: (CRON) INFO (RANDOM_DELAY will be scaled with fac>  
Jan 16 12:18:21 ivschemelev.localdomain crond[1429]: (CRON) INFO (running with inotify support)
```

Рис. 2.1: Статус службы `crond`

3. Просмотрено содержимое системного файла конфигурации `/etc/crontab`.

Файл содержит переменные окружения SHELL, PATH, MAILTO и комментарии с описанием синтаксиса заданий cron. Пользовательские задания в данном файле отсутствуют.

```
root@ivschemelev:/home/ivschemelev#  
root@ivschemelev:/home/ivschemelev# 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@ivschemelev:/home/ivschemelev#
```

Рис. 2.2: Содержимое файла /etc/crontab

4. Выполнен просмотр списка заданий текущего пользователя.

Вывод отсутствует, что свидетельствует об отсутствии ранее настроенных заданий в пользовательском расписании.

5. Файл пользовательского расписания открыт для редактирования.

В редакторе добавлена строка:

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

Пояснение синтаксиса записи:

- */1 — выполнение каждую минуту;
- — — каждый час;
- — — каждый день месяца;
- — — каждый месяц;
- — — каждый день недели;

- `logger` — команда записи сообщения в системный журнал.



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

Рис. 2.3: Редактирование crontab

- После сохранения изменений повторно выполнен просмотр списка заданий.

В расписании отображается добавленное задание, что подтверждает успешную установку расписания.

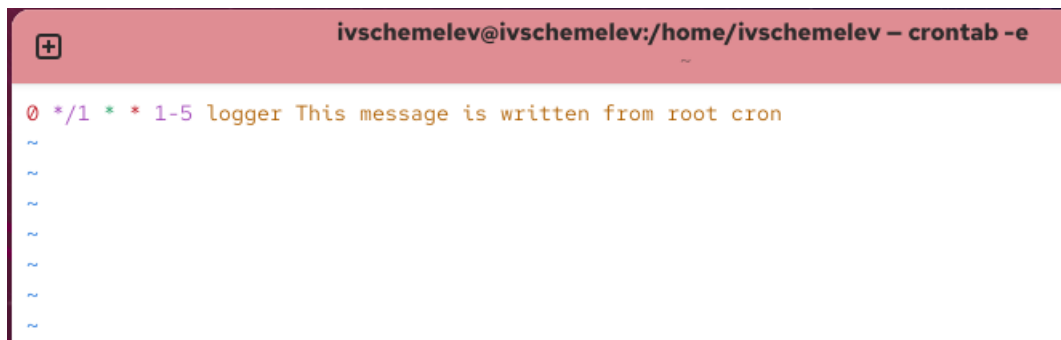
```
root@ivschemelev:/home/ivschemelev#
root@ivschemelev:/home/ivschemelev# crontab -l

root@ivschemelev:/home/ivschemelev# crontab -e
crontab: installing new crontab
Backup of root's previous crontab saved to /root/.cache/crontab/crontab.bak
root@ivschemelev:/home/ivschemelev# crontab -l
*/1 * * * * logger This message is written from root cron
root@ivschemelev:/home/ivschemelev# grep written /var/log/messages
root@ivschemelev:/home/ivschemelev# grep written /var/log/messages
Jan 16 12:23:01 ivschemelev root[4655]: This message is written from root cron
root@ivschemelev:/home/ivschemelev# grep written /var/log/messages
Jan 16 12:23:01 ivschemelev root[4655]: This message is written from root cron
Jan 16 12:24:01 ivschemelev root[4819]: This message is written from root cron
Jan 16 12:25:01 ivschemelev root[4948]: This message is written from root cron
root@ivschemelev:/home/ivschemelev#
```

Рис. 2.4: Список заданий crontab

- Через 2–3 минуты выполнен просмотр системного журнала.

В журнале зафиксированы регулярные сообщения, создаваемые заданием `cron`, что подтверждает его корректное выполнение.

A terminal window with a red title bar. The title bar contains a plus icon on the left and the text 'ivschemelev@ivschemelev:/home/ivschemelev – crontab -e' on the right. The terminal content shows a cron job entry: '0 */1 * * 1-5 logger This message is written from root cron'. Below this entry are several tilde (~) characters, indicating the end of the crontab file.

```
ivschemelev@ivschemelev:/home/ivschemelev – crontab -e
0 */1 * * 1-5 logger This message is written from root cron
~
~
~
~
~
~
```

Рис. 2.5: Сообщения cron в журнале

8. Запись в расписании cron изменена на следующую:

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

Пояснение синтаксиса записи:

- 0 — выполнение в нулевую минуту часа;
- */1 — каждый час;
- — — любой день месяца;
- — — любой месяц;
- 1–5 — с понедельника по пятницу;
- logger — запись сообщения в журнал.

9. Повторно выполнен просмотр списка заданий cron.

В выводе отображается обновлённая запись.

10. Выполнен переход в каталог /etc/cron.hourly и создан файл сценария eachhour.

11. Файл eachhour открыт для редактирования, в него добавлен сценарий записи сообщения в системный журнал с указанием текущих даты и времени.



```
ivschemelev@ivschemelev:/etc/cron.hourly - nano eachhour
/et/cron.hourly
GNU nano 8.1      eachhour
#!/bin/sh
logger This message is written at $(date)
```

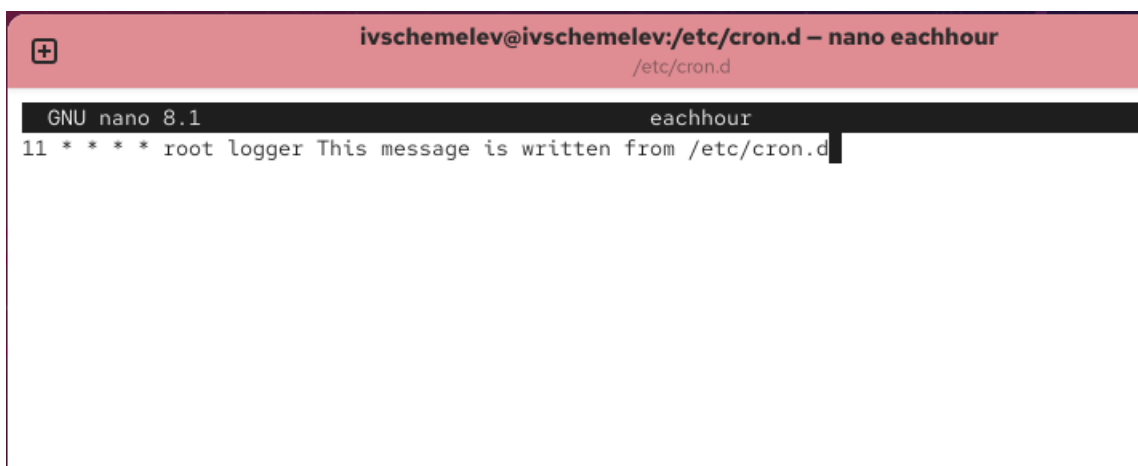
Рис. 2.6: Сценарий eachhour

12. Файлу сценария eachhour выданы права на исполнение, что позволило системе выполнять его по расписанию.

13. В каталоге /etc/cron.d создан файл расписания eachhour, в который добавлена строка:

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

Пояснение синтаксиса записи: - 11 — выполнение на 11-й минуте каждого часа; - * * * — любой час, день месяца и месяц; - * — любой день недели; - root — пользователь, от имени которого выполняется команда; - logger — запись сообщения в системный журнал.



```
ivschemelev@ivschemelev:/etc/cron.d - nano eachhour
/et/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

14. Через установленный промежуток времени выполнен анализ системного журнала.

По наличию соответствующих сообщений подтверждено, что сценарий eachhour запускался в соответствии с заданным расписанием.

2.2 Планирование заданий с помощью at

1. Получены полномочия администратора.

2. Проверено состояние службы отложенного выполнения заданий atd.

Установлено, что служба загружена, включена и находится в состоянии active (running).

```
root@ivschemellev:/etc/cron.d#  
root@ivschemellev:/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 Fri 2026-01-16 12:18:21 MSK; 13min ago  
 Invocation: 6794c5973e0d4d76a04455626b6ed638  
    Docs: man:atd(8)  
 Main PID: 1426 (atd)  
   Tasks: 1 (limit: 23034)  
  Memory: 324K (peak: 1.3M)  
     CPU: 2ms  
  CGroup: /system.slice/atd.service  
          └─1426 /usr/sbin/atd -f  
  
Jan 16 12:18:21 ivschemelev.localdomain systemd[1]: Started atd.service - Deferred execution scheduler.  
Jan 16 12:18:21 ivschemelev.localdomain (atd)[1426]: atd.service: Referenced but unset environment var>  
root@ivschemellev:/etc/cron.d#  
root@ivschemellev:/etc/cron.d# at 12:34  
warning: commands will be executed using /bin/sh  
at Fri Jan 16 12:34:00 2026  
at> logger message from at  
at> <EOT>  
job 1 at Fri Jan 16 12:34:00 2026  
root@ivschemellev:/etc/cron.d# atq  
1      Fri Jan 16 12:34:00 2026 a root  
root@ivschemellev:/etc/cron.d# grep 'from at' /var/log/messages  
root@ivschemellev:/etc/cron.d# grep 'from at' /var/log/messages  
Jan 16 12:34:00 ivschemelev root[6491]: message from at  
root@ivschemellev:/etc/cron.d#
```

Рис. 2.8: Статус службы atd

3. Задано отложенное выполнение команды записи сообщения в системный журнал на указанное время.

После завершения ввода задание было принято системой к исполнению.

4. Выполнена проверка очереди заданий at.

После наступления заданного времени в системном журнале зафиксировано сообщение, подтверждающее успешное выполнение задания.

3 Контрольные вопросы

1. Чтобы задание стоп выполнялось раз в 2 недели, используется указание дня недели и шага в поле недели.

Так как стоп не поддерживает прямое понятие «неделя», обычно выбирается конкретный день недели с шагом 2.

Пример выполнения задания раз в две недели по понедельникам в 3 часа ночи:

0 3 * * 1/2 команда

Здесь: - 0 — минута; - 3 — час; - * — любой день месяца; - * — любой месяц; - 1/2 — каждый второй понедельник.

2. Для выполнения задания 1-го и 15-го числа каждого месяца в 2 часа ночи используется перечисление значений в поле дня месяца.

Пример записи:

0 2 1,15 * * команда

Здесь: - 0 — нулевая минута; - 2 — 2 часа ночи; - 1,15 — 1-е и 15-е число месяца; - * — любой месяц; - * — любой день недели.

3. Для выполнения задания каждые 2 минуты каждый день применяется шаг в поле минут.

Пример записи:

`*/2 * * * * команда`

Здесь: - `*/2` — каждые 2 минуты; - остальные поля со значением `*` означают любое время, день и месяц.

4. Для выполнения задания 19 сентября ежегодно указываются конкретные день и месяц.

Пример записи:

`0 0 19 9 * команда`

Здесь: - `0` — минута; - `0` — час; - `19` — день месяца; - `9` — сентябрь; - `*` — любой день недели.

5. Для выполнения задания каждый четверг сентября ежегодно используется указание месяца и дня недели.

Пример записи:

`0 1 * 9 4 команда`

Здесь: - `0` — минута; - `1` — час; - `*` — любой день месяца; - `9` — сентябрь; - `4` — четверг.

6. Для назначения задания cron конкретному пользователю используется команда переключения на этого пользователя либо параметр `-u` (для администратора).

Пример назначения задания пользователю `alice`:

`crontab -u alice -e`

После выполнения команды открывается файл расписания cron пользователя `alice`, в который можно добавить необходимые задания.

7. Чтобы запретить пользователю bob назначать задания через cron, используется файл `/etc/cron.deny`.

Пример: - открыть файл `/etc/cron.deny`; - добавить в него строку: bob

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

8. Чтобы задание выполнялось каждый день даже в случае временной недоступности сервера, рекомендуется использовать планировщик `anacron`.

Anacron гарантирует выполнение заданий после включения системы, если они были пропущены во время простоя.

Для этого задание следует разместить в конфигурации `anacron` или в каталогах `/etc/cron.daily`, `/etc/cron.weekly`.

9. Для проверки наличия заданий, запланированных планировщиком `atd`, используется команда: `atq`

Команда выводит список всех отложенных заданий. Если список пуст, значит задания `at` не запланированы.

4 Заключение

В ходе лабораторной работы были изучены и практически освоены средства планирования заданий в операционной системе Linux с использованием планировщиков cron и at. Выполнена настройка периодических и отложенных задач, проверена работа соответствующих служб и проанализированы результаты выполнения заданий по системным журналам. Полученные навыки позволяют автоматизировать выполнение административных операций и эффективно управлять задачами в серверной среде.