

Отчет по лабораторной работе №8

Простейший вариант

Лупупа Чилеше

1 Цель работы

Получение навыков работы с планировщиками событий cron и at.

2 Выполнение лабораторной работы

1. Запустите терминал и получите полномочия администратора:

su -

```
[root@chileshelupupa ~]# systemctl status crond -l
● crond.service - Command Scheduler
   Loaded: loaded (/usr/lib/systemd/system/crond.service; enabled; preset: enabled)
   Active: active (running) since Sat 2024-10-26 17:27:43 MSK; 18min ago
     Main PID: 1101 (crond)
       Tasks: 1 (limit: 24674)
      Memory: 1.0M
         CPU: 11ms
    CGroup: /system.slice/crond.service
            └─1101 /usr/sbin/crond -n
```

2. Посмотрите статус демона crond:

systemctl status crond -l



3. Посмотрите содержимое файла конфигурации

/etc/crontab:

cat /etc/crontab

```
[root@chilesheLupupa ~]# crontab -l
no crontab for root
[root@chilesheLupupa ~]#
```

4. Посмотрите список заданий в расписании:

`crontab -l`

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

Ничего не отобразится, так как расписание ещё не задано.

5. Откройте файл расписания на редактирование:

`crontab -e`

Команда запустит интерфейс редактора (по умолчанию используется vi). Добавьте 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`

```
[root@chilesheLupupa ~]# crontab -l
*/1 * * * * logger This message is written from root cron
[root@chilesheLupupa ~]#
```

Закройте сеанс редактирования vi и сохраните изменения, используя команду vi: Esc : w q . В отчёте поясните синтаксис записи в crontab.

6. Посмотрите список заданий в расписании:

`crontab -l`

```
[root@chilesheLupupa ~]# crontab -l
*/1 * * * * logger This message is written from root cron
[root@chilesheLupupa ~]#
```

В расписании должна появиться запись о запланированном событии.

7. Не выключая систему, через некоторое время (2–3 минуты) просмотрите журнал системных событий:

`grep written /var/log/messages`

```
[root@chilesheLupupa ~]# grep written /var/log/messages
Oct 26 17:51:01 chilesheLupupa root[3437]: This message is written from root cron
Oct 26 17:52:02 chilesheLupupa root[3515]: This message is written from root cron
[root@chilesheLupupa ~]#
```

В отчёте отразите результат.

8. Измените запись в расписании crontab на следующую:

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

```
[root@chileshelupupa ~]# crontab -l
0 */1 * * 1-5 logger This message is written from root cron
[root@chileshelupupa ~]#
```

В отчёте поясните синтаксис записи в crontab.

9. Посмотрите список заданий в расписании:

```
crontab -l
```

```
[root@chileshelupupa ~]# crontab -l
0 */1 * * 1-5 logger This message is written from root cron
[root@chileshelupupa ~]#
```

10. Перейдите в каталог /etc/cron.hourly и создайте в нём файл сценария с именем eachhour:

```
cd /etc/cron.hourly
```

```
touch eachhour
```

```
[root@chileshelupupa ~]# cd /etc/cron.hourly
[root@chileshelupupa cron.hourly]# touch eachhour
[root@chileshelupupa cron.hourly]#
```

11. Откройте файл eachhour для редактирования и пропишите в нём следующий скрипт (запись сообщения в системный журнал):

```
#!/bin/sh
```

```
logger This message is written at $(date)
```

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

12. Сделайте файл сценария eachhour исполняемым:

```
chmod +x eachhour
```

```
[root@chileshelupupa cron.hourly]# chmod +x eachhour
[root@chileshelupupa cron.hourly]#
```

13. Теперь перейдите в каталог `/etc/cron.d` и создайте в нём файл с расписанием `eachhour`:

```
cd /etc/cron.d touch eachhour
```

Откройте этот файл для редактирования и поместите в него следующее содержимое:

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

```
GNU nano 5.6.1                                     eachhour
11 * * * * root logger This message is written from /etc/cron.d
```

Сохраните изменения. В отчёте поясните синтаксис этой записи.

14. Не выключая систему, через некоторое время (2–3 часа) просмотрите журнал системных событий: `grep written /var/log/messages` По журналу определите, был ли осуществлён запуск сценария `eachhour` в соответствии с заданным расписанием.

```
[root@chilshelupupa cron.d]# grep written /var/log/messages
Oct 26 17:51:01 chilshelupupa root[3437]: This message is written from root cron
Oct 26 17:52:02 chilshelupupa root[3515]: This message is written from root cron
Oct 26 17:53:01 chilshelupupa root[3598]: This message is written from root cron
Oct 26 18:01:01 chilshelupupa root[3699]: This message is written at Sat Oct 26 06:01:01 PM MSK 2024
Oct 26 18:11:01 chilshelupupa root[3732]: This message is written from /etc/cron.d
[root@chilshelupupa cron.d]#
```

Планирование заданий с помощью `at`

1. Запустите терминал и получите полномочия администратора:

```
su -
```

2. Проверьте, что служба `atd` загружена и включена:

```
systemctl status atd
```

3. Задайте выполнение команды `logger message from at` в 9:30 (или замените на любое другое время, когда вы работаете над этим упражнением). Для этого введите `at 9:30` Затем введите

```
logger message from at
```

Используйте `Ctrl + d`, чтобы закрыть оболочку.

4. Убедитесь, что задание действительно запланировано:

```
atq
```

С помощью команды `grep 'from at' /var/log/messages` посмотрите, появилось ли соответствующее сообщение в лог-файле в указанное вами время.