

Лабораторная работа №8

Планировщики событий

Комягин Андрей Николаевич

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Планирование задач с помощью cron	6
2.2	Планирование заданий с помощью at	9
3	Контрольные вопросы	10
4	Вывод	12
	Список литературы	13

Список иллюстраций

2.1	статус crond задания в расписании	6
2.2	файл расписания, журнал событий	7
2.3	crontab	7
2.4	eachhour	8
2.5	служба atd	9

Список таблиц

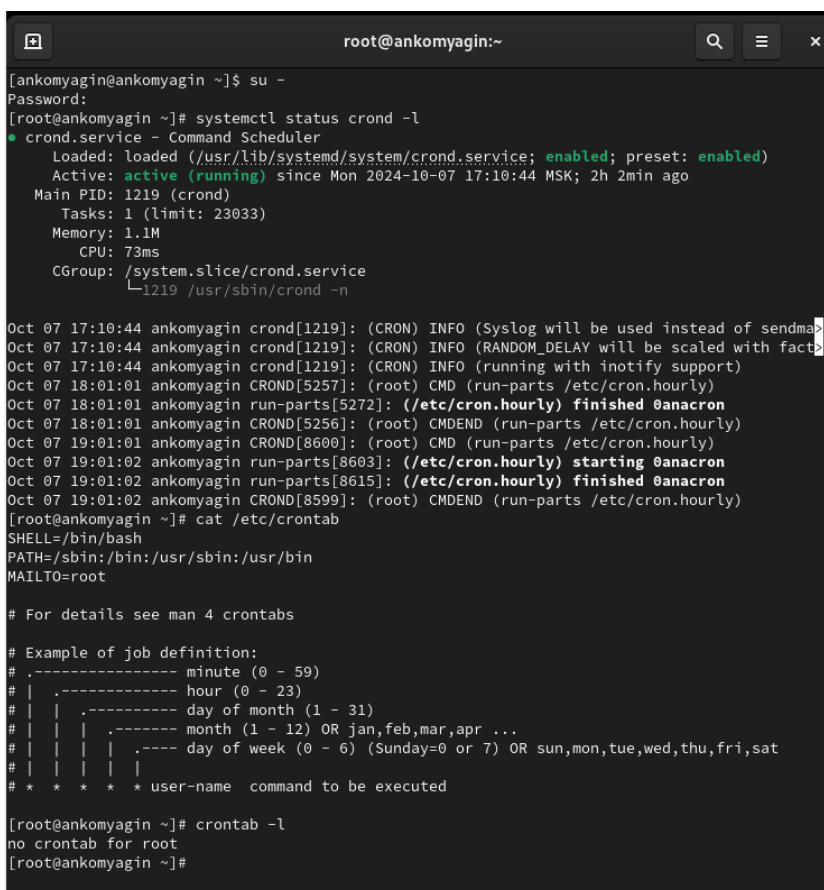
1 Цель работы

Получение навыков работы с планировщиками событий cron и at.

2 Выполнение лабораторной работы

2.1 Планирование задач с помощью cron

Посмотрив статус демона crond, посмотрим содержимое файла конфигурации /etc/crontab, осмотрим список заданий в расписании (рис. 2.1).

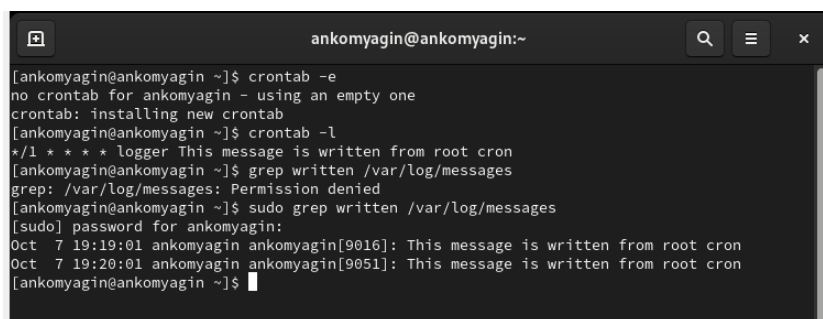


```
root@ankomyagin:~  
[ankomyagin@ankomyagin ~]$ su -  
Password:  
[root@ankomyagin ~]# systemctl status crond -l  
● crond.service - Command Scheduler  
   Loaded: loaded (/usr/lib/systemd/system/crond.service; enabled; preset: enabled)  
   Active: active (running) since Mon 2024-10-07 17:10:44 MSK; 2h 2min ago  
     Main PID: 1219 (crond)  
       Tasks: 1 (limit: 23033)  
      Memory: 1.1M  
         CPU: 73ms  
    CGroup: /system.slice/crond.service  
            └─1219 /usr/sbin/crond -n  
  
Oct 07 17:10:44 ankomyagin crond[1219]: (CRON) INFO (Syslog will be used instead of sendmail)  
Oct 07 17:10:44 ankomyagin crond[1219]: (CRON) INFO (RANDOM_DELAY will be scaled with factor  
Oct 07 17:10:44 ankomyagin crond[1219]: (CRON) INFO (running with inotify support)  
Oct 07 18:01:01 ankomyagin CROND[5257]: (root) CMD (run-parts /etc/cron.hourly)  
Oct 07 18:01:01 ankomyagin run-parts[5272]: (/etc/cron.hourly) finished 0anacron  
Oct 07 18:01:01 ankomyagin CROND[5256]: (root) CMDEND (run-parts /etc/cron.hourly)  
Oct 07 19:01:01 ankomyagin CROND[8600]: (root) CMD (run-parts /etc/cron.hourly)  
Oct 07 19:01:02 ankomyagin run-parts[8603]: (/etc/cron.hourly) starting 0anacron  
Oct 07 19:01:02 ankomyagin run-parts[8615]: (/etc/cron.hourly) finished 0anacron  
Oct 07 19:01:02 ankomyagin CROND[8599]: (root) CMDEND (run-parts /etc/cron.hourly)  
[root@ankomyagin ~]# 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@ankomyagin ~]# crontab -l  
no crontab for root  
[root@ankomyagin ~]#
```

Рис. 2.1: статус crond задания в расписании

Добавим строку в файл расписания. Посмотрим список заданий в расписании.

Просмотрим журнал системных событий(рис. 2.2).



```
ankomyagin@ankomyagin:~$ crontab -e
no crontab for ankomyagin - using an empty one
crontab: installing new crontab
ankomyagin@ankomyagin ~]$ crontab -l
*/1 * * * * logger This message is written from root cron
ankomyagin@ankomyagin ~]$ grep written /var/log/messages
grep: /var/log/messages: Permission denied
ankomyagin@ankomyagin ~]$ sudo grep written /var/log/messages
[sudo] password for ankomyagin:
Oct  7 19:19:01 ankomyagin ankomyagin[9016]: This message is written from root cron
Oct  7 19:20:01 ankomyagin ankomyagin[9051]: This message is written from root cron
ankomyagin@ankomyagin ~]$
```

Рис. 2.2: файл расписания, журнал событий

“*/1 * * * * logger This message is written from root cron”

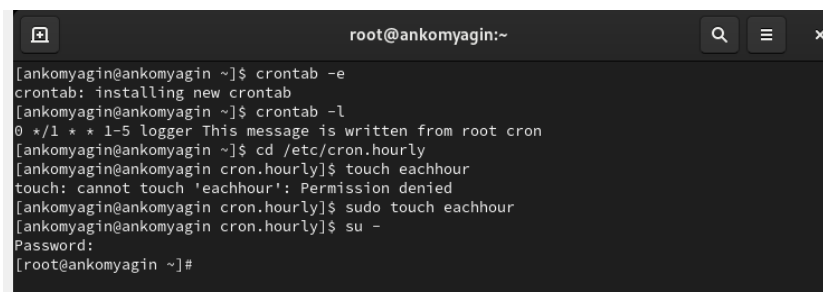
Синтаксис: (по порядку звездочек)

- /1 — это поле минут. Оно означает “каждую минуту”. Знак * означает “каждое значение”, а /1 указывает, что это происходит каждые 1 минуту.
- — — поле часов. Здесь также указано “каждый час”.
- — — поле дней месяца. “Каждый день месяца”.
- — — поле месяцев. “Каждый месяц”.
- — — поле дней недели. “Каждый день недели”.

Эта запись означает, что команда **logger This message is written from root cron** будет выполняться каждую минуту.

Изменим запись в расписании crontab на следующую:

0 /1 * 1-5 logger This message is written from root cron (рис. 2.3).



```
root@ankomyagin:~$ crontab -e
crontab: installing new crontab
ankomyagin@ankomyagin ~]$ crontab -l
0 */1 * * 1-5 logger This message is written from root cron
ankomyagin@ankomyagin ~]$ cd /etc/cron.hourly
ankomyagin@ankomyagin cron.hourly]$ touch eachhour
touch: cannot touch 'eachhour': Permission denied
ankomyagin@ankomyagin cron.hourly]$ sudo touch eachhour
ankomyagin@ankomyagin cron.hourly]$ su -
Password:
[root@ankomyagin ~]#
```

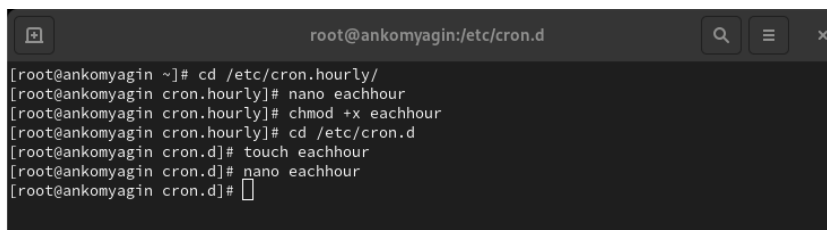
Рис. 2.3: crontab

Перейдём в каталог `/etc/cron.hourly` и создадим в нём файл сценария с именем **eachhour**. Откроем файл `eachhour` и пропишем в нём скрипт. Сделаем файл сценария **eachhour** исполняемым. Перейдём в каталог `/etc/cron.d` и создадим в нём файл с расписанием **eachhour** (рис. 2.4).

Синтаксис:

- 0 — это поле минут. Команда будет выполняться в начале часа (0 минут).
- */1 — поле часов. Это означает “каждый час”.
- — — поле дней месяца. “Каждый день месяца”.
- — — поле месяцев. “Каждый месяц”.
- 1-5 — поле дней недели. Это означает, что команда будет выполняться только с понедельника по пятницу.

Эта запись означает, что команда будет выполняться каждый час в начале часа (0 минут) только с понедельника по пятницу.



```
root@ankomyagin:/etc/cron.d
[root@ankomyagin ~]# cd /etc/cron.hourly/
[root@ankomyagin cron.hourly]# nano eachhour
[root@ankomyagin cron.hourly]# chmod +x eachhour
[root@ankomyagin cron.hourly]# cd /etc/cron.d
[root@ankomyagin cron.d]# touch eachhour
[root@ankomyagin cron.d]# nano eachhour
[root@ankomyagin cron.d]#
```

Рис. 2.4: eachhour

Откроем этот файл для редактирования и поместим в него следующее содержимое:

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

Синтаксис:

- 11 — это поле минут. Команда будет выполняться в 11-й минуте каждого часа.

- — — поле часов. “Каждый час”.
- — — поле дней месяца. “Каждый день месяца”.
- — — поле месяцев. “Каждый месяц”.
- — — поле дней недели. “Каждый день недели”.
- root — это поле пользователя, указывающее, что команда будет выполняться от имени пользователя root.

Эта запись означает, что команда **logger This message is written from /etc/cron.d** будет выполняться каждый час в 11-й минуте от имени пользователя root.

2.2 Планирование заданий с помощью at

Проверим, что служба atd загружена и включена. Зададим выполнение команды `logger message from at` в 19:29. Убедимся, что задание действительно запланировано (рис. 2.5).

```
[root@ankomyagin 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 Mon 2024-10-07 17:10:44 MSK; 2h 16min ago
     Docs: man:atd(8)
    Main PID: 1216 (atd)
      Tasks: 1 (limit: 23033)
    Memory: 296.0K
       CPU: 3ms
    CGroup: /system.slice/atd.service
            └─1216 /usr/sbin/atd -f

Oct 07 17:10:44 ankomyagin systemd[1]: Started Deferred execution scheduler.
[root@ankomyagin cron.d]# at 19:29
warning: commands will be executed using /bin/sh
at> logger message from at
at> <EOT>
job 2 at Mon Oct 7 19:29:00 2024
[root@ankomyagin cron.d]# atq
2      Mon Oct 7 19:29:00 2024 a root
[root@ankomyagin cron.d]# grep 'from at' /var/log/messages
Oct 7 19:29:00 ankomyagin root[9366]: message from at
[root@ankomyagin cron.d]#
```

Рис. 2.5: служба atd

3 Контрольные вопросы

1. Настройка задания cron для выполнения раз в 2 недели

Чтобы задать выполнение задания раз в 2 недели, можно использовать следующие записи:

0 0 * * 0,14 command

Эта запись будет выполнять команду в полночь (00:00) каждое воскресенье и 14-го числа каждого месяца. Однако, если вам нужно строгое выполнение каждые 14 дней, это потребует более сложной логики, так как cron не поддерживает это напрямую.

2. Настройка задания cron для выполнения 1-го и 15-го числа каждого месяца в 2 часа ночи

0 2 1,15 * * command

Эта запись будет выполнять команду в 2:00 ночи 1-го и 15-го числа каждого месяца.

3. Настройка задания cron для выполнения каждые 2 минуты каждый день

***/2 * * * * command**

Эта запись будет выполнять команду каждые 2 минуты каждый день.

4. Настройка задания cron для выполнения 19 сентября ежегодно

0 0 19 9 * command

Эта запись будет выполнять команду в полночь (00:00) 19 сентября каждого года.

5. Настройка задания cron для выполнения каждый четверг сентября ежегодно

0 0 * 9 4 command

Эта запись будет выполнять команду в полночь (00:00) каждую пятницу (4 - четверг) в сентябре.

6. Команда для назначения задания cron для пользователя alice Вы можете использовать команду crontab с параметром -u:

sudo crontab -u alice -e

Это откроет crontab для пользователя alice, где вы можете добавить свои задания.

7. Запрет назначения заданий через cron для пользователя bob

Для этого можно отредактировать файл /etc/cron.d/cron.allow и /etc/cron.d/cron.deny. Если вы добавите пользователя bob в файл cron.deny, он не сможет назначать задания через cron.

echo "bob" » /etc/cron.d/cron.deny

8. Убедиться, что задание выполняется каждый день, даже если сервер временно недоступен

Вы можете использовать systemd или другой менеджер задач, который поддерживает повторные попытки выполнения. Кроме того, можно добавить логику в сам скрипт, чтобы проверять, был ли он выполнен в предыдущие дни, например, записывать дату последнего выполнения в файл и проверять его наличие.

9. Команда для проверки запланированных заданий на выполнение с помощью atd

atq

Эта команда покажет список всех заданий, запланированных с помощью at.

4 Вывод

В ходе выполнения лабораторной работы я получил навыки работы с планировщиками событий cron и at.

Список литературы

Туис, курс Администрирование операционных систем