Отчёт по лабораторной работе №8

Планировщики событий

Анастасия Мазуркевич

Содержание

# 1 Цель работы

Получение навыков работы с планировщиками событий cron и at.

# 2 Ход выполнения

## 2.1 Планирование задач с помощью cron

После входа в систему были получены права администратора с помощью команды su.  
Затем была проверена работа службы планировщика заданий cron. Команда systemctl status crond -l показала, что служба активна (active (running)) и включена в автозагрузку (enabled).

Далее было просмотрено содержимое конфигурационного файла /etc/crontab.  
В нём заданы переменные окружения и приведён пример синтаксиса задания, где каждая позиция соответствует минутам, часам, дню месяца, месяцу и дню недели.

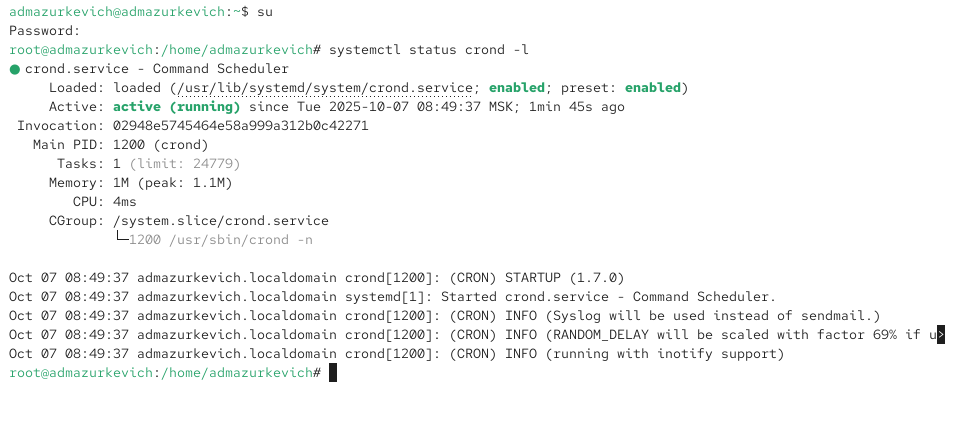


Рис. 1: Просмотр содержимого /etc/crontab

Проверка списка заданий командой crontab -l показала отсутствие записей, так как расписание ещё не было настроено.  
Редактирование расписания началось через crontab -e, после чего добавлена строка:

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

Эта запись означает выполнение команды каждую минуту. Символ \*/1 указывает на выполнение через один интервал, а остальные звёздочки — на любые значения часов, дней и месяцев. Команда logger создаёт сообщение в системном журнале. После сохранения файла задание было установлено успешно.

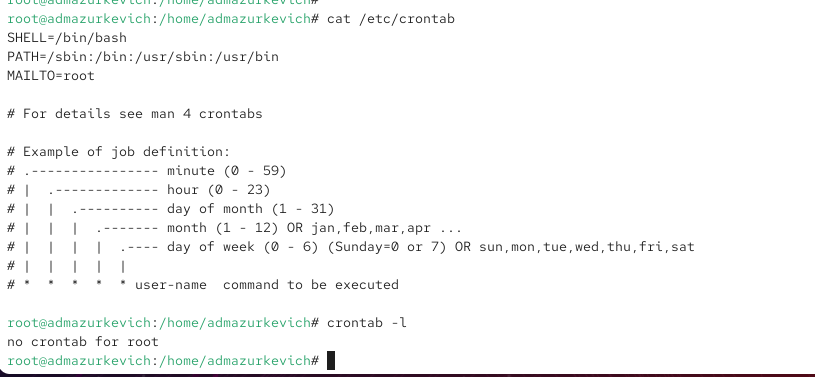


Рис. 2: Создание и проверка задания в crontab

Через несколько минут в системном журнале /var/log/messages появились сообщения This message is written from root cron, подтверждающие выполнение задачи по расписанию.

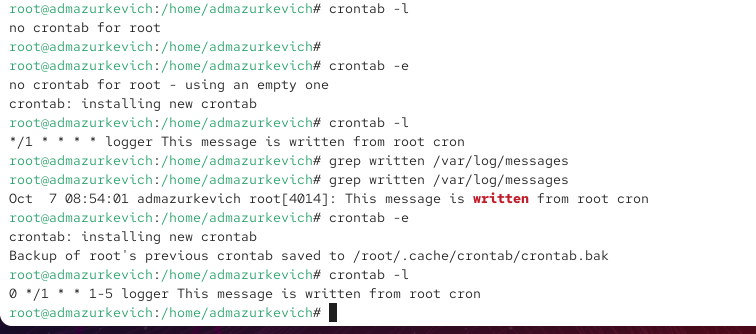


Рис. 3: Появление записей cron в журнале

Затем расписание было изменено на следующую строку:

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

Данное выражение означает выполнение команды каждый час в нулевую минуту, с понедельника по пятницу. После сохранения изменений crontab -l подтвердил наличие нового расписания.

В каталоге /etc/cron.hourly был создан сценарий eachhour с содержимым, которое формирует сообщение с текущей датой.  
Файл был сделан исполняемым при помощи chmod +x eachhour.

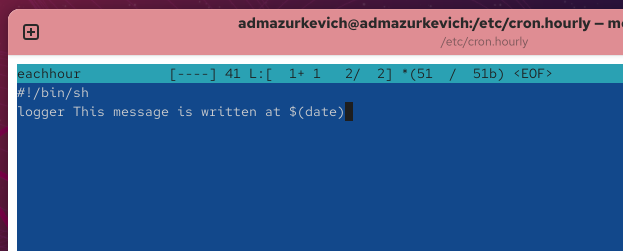


Рис. 4: Сценарий eachhour в каталоге /etc/cron.hourly

Далее в каталоге /etc/cron.d был создан файл eachhour, содержащий строку:

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

Эта запись задаёт выполнение команды на 11-й минуте каждого часа от имени пользователя root.

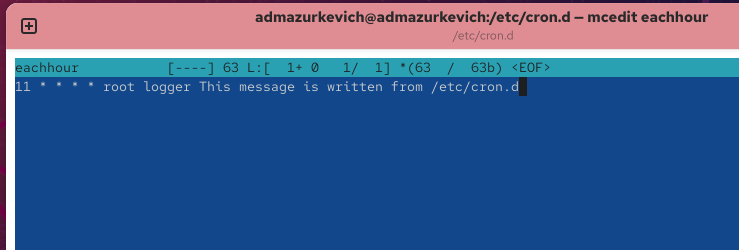


Рис. 5: Создание файла расписания в /etc/cron.d

Проверка системного журнала через некоторое время подтвердила корректное выполнение заданий, созданных как через crontab, так и с помощью сценариев из каталогов cron.hourly и cron.d.  
В логах отобразились сообщения с указанием времени их выполнения.

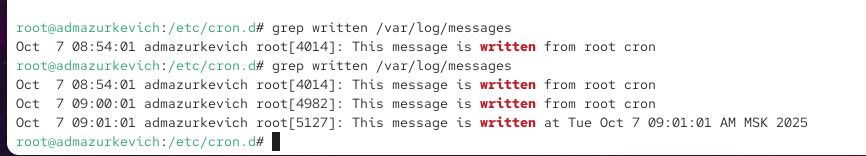


Рис. 6: Проверка выполнения заданий cron

## 2.2 Планирование заданий с помощью at

Для одноразового выполнения команд была проверена служба atd. Команда systemctl status atd показала, что служба активна (active (running)) и включена в автозагрузку (enabled).

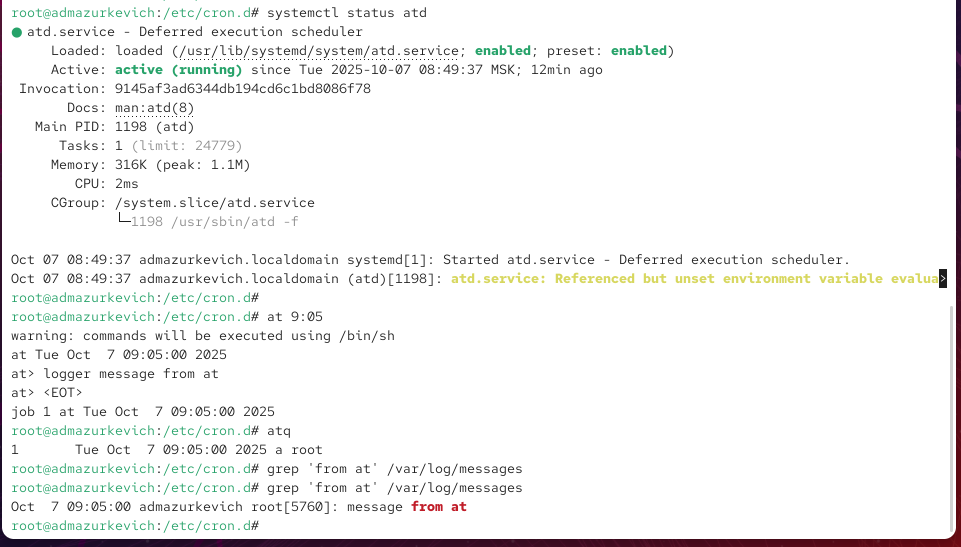


Рис. 7: Проверка состояния службы atd

Было запланировано выполнение команды logger message from at на 09:05. После завершения ввода с помощью сочетания Ctrl+D задание добавилось в очередь, что подтвердилось командой atq.  
В назначенное время в системном журнале /var/log/messages появилось сообщение message from at, подтверждающее успешное выполнение задания.

# 3 Контрольные вопросы

**1. Как настроить задание cron, чтобы оно выполнялось раз в 2 недели?**  
Задания с периодом раз в две недели можно задать через комбинирование дня недели и номера недели.  
Например:  
0 0 \*/14 \* \* команда  
Эта запись означает выполнение задания каждые 14 дней в полночь.  
Однако cron не поддерживает точный интервал «раз в две недели» — для надёжности используют внешние проверки (например, через скрипт, который запускается ежедневно, но выполняет действие только если прошло 14 дней).

**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  
Расшифровка:  
- 0 — минута (ровно в начале часа);  
- 2 — час (в 2:00 ночи);  
- 1,15 — дни месяца;  
- \* \* — каждый месяц и любой день недели.

**3. Как настроить задание cron, чтобы оно выполнялось каждые 2 минуты каждый день?**  
\*/2 \* \* \* \* команда  
Задание будет выполняться через каждые две минуты, независимо от даты и времени суток.

**4. Как настроить задание cron, чтобы оно выполнялось 19 сентября ежегодно?**  
0 0 19 9 \* команда  
Расшифровка:  
- 0 0 — запуск в полночь;  
- 19 — день месяца;  
- 9 — сентябрь;  
- \* — любой день недели.  
Таким образом, команда выполнится один раз в год — 19 сентября.

**5. Как настроить задание cron, чтобы оно выполнялось каждый четверг сентября ежегодно?**  
0 0 \* 9 4 команда  
Расшифровка:  
- 0 0 — запуск в полночь;  
- \* — любой день месяца;  
- 9 — сентябрь;  
- 4 — четверг (нумерация дней недели от 0 до 6, где 0 или 7 — воскресенье).

**6. Какая команда позволяет вам назначить задание cron для пользователя alice? Приведите подтверждающий пример.**  
Для создания расписания от имени конкретного пользователя используется параметр -u.  
Пример:  
crontab -u alice -e  
Команда откроет файл расписания для пользователя **alice** в редакторе, где можно добавить, например:  
0 8 \* \* \* logger "Задание от имени пользователя alice".

**7. Как указать, что пользователю bob никогда не разрешено назначать задания через cron? Приведите подтверждающий пример.**  
Для ограничения доступа создаётся файл /etc/cron.deny.  
Если добавить туда имя пользователя bob, он не сможет использовать команду crontab.  
Пример содержимого файла:  
bob  
После этого при попытке выполнить crontab -e пользователь получит сообщение об отказе в доступе.

**8. Вам нужно убедиться, что задание выполняется каждый день, даже если сервер во время выполнения временно недоступен. Как это сделать?**  
Следует использовать каталог **/etc/cron.daily** или утилиту **anacron**, которая выполняет пропущенные задания при следующем запуске системы.  
Anacron гарантирует выполнение ежедневных, еженедельных и ежемесячных задач даже при отключении сервера в момент их планового запуска.

**9. Какая команда позволяет узнать, запланированы ли какие-либо задания на выполнение планировщиком atd?**  
Для просмотра очереди заданий используется команда:  
atq  
Она показывает список всех заданий, поставленных на выполнение с помощью **at**, включая их идентификаторы и время запуска.

# 4 Заключение

В ходе выполнения работы была изучена система планирования заданий в Linux.  
С помощью cron удалось создать периодические задания, выполняющиеся по расписанию, а служба atd позволила настроить одноразовое выполнение команды в указанное время.  
Все задания успешно отработали, что подтверждено записями в системном журнале.