# LAB-03. Регистрация событий jornald

## Основы просмотра журнала

1. Чтобы просмотреть журналы, собранные демоном journald, используйте команду journalctl.
2. Когда эта команда используется отдельно, все записи журналов в системе будут выведены в многостраничном списке (обычно занимает меньше страницы), который вы сможете просматривать. Самые старые записи будут отображаться сверху:

journalctl

-- Logs begin at Tue 2015-02-03 21:48:52 UTC, end at Tue 2015-02-03 22:29:38 UTC. --

Feb 03 21:48:52 localhost.localdomain systemd-journal[243]: Runtime journal is using 6.2M (max allowed 49.

Feb 03 21:48:52 localhost.localdomain systemd-journal[243]: Runtime journal is using 6.2M (max allowed 49.

Feb 03 21:48:52 localhost.localdomain systemd-journald[139]: Received SIGTERM from PID 1 (systemd).

Feb 03 21:48:52 localhost.localdomain kernel: audit: type=1404 audit(1423000132.274:2): enforcing=1 old\_en

Feb 03 21:48:52 localhost.localdomain kernel: SELinux: 2048 avtab hash slots, 104131 rules.

Feb 03 21:48:52 localhost.localdomain kernel: SELinux: 2048 avtab hash slots, 104131 rules.

Feb 03 21:48:52 localhost.localdomain kernel: input: ImExPS/2 Generic Explorer Mouse as /devices/platform/

Feb 03 21:48:52 localhost.localdomain kernel: SELinux:  8 users, 102 roles, 4976 types, 294 bools, 1 sens,

Feb 03 21:48:52 localhost.localdomain kernel: SELinux:  83 classes, 104131 rules

. . .

1. Если вы хотите вывести временные метки в формате UTC, вы можете использовать флаг --utc:

journalctl --utc

**Вывод журналов текущей загрузки**

1. Флаг -b — одна из самых простых опций, которой вы часто будете пользоваться. С его помощью вы сможете вывести для просмотра все записи журнала, собранные с момента последней перезагрузки.

journalctl -b

Это поможет вам определять, какая информация важна для текущей среды, и управлять этой информацией.

1. Если на вашем сервере включено хранение журналов предыдущих сеансов загрузки, утилита journalctl предоставит ряд команд, которые помогут работать с сеансами загрузки как с единицами хранения. Чтобы просмотреть сеансы загрузки, о которых известно journald, используйте опцию --list-boots с командой journalctl:

journalctl --list-boots

-2 caf0524a1d394ce0bdbcff75b94444fe Tue 2015-02-03 21:48:52 UTC—Tue 2015-02-03 22:17:00 UTC

-1 13883d180dc0420db0abcb5fa26d6198 Tue 2015-02-03 22:17:03 UTC—Tue 2015-02-03 22:19:08 UTC

 0 bed718b17a73415fade0e4e7f4bea609 Tue 2015-02-03 22:19:12 UTC—Tue 2015-02-03 23:01:01 UTC

1. Например, чтобы просмотреть журнал предыдущей загрузки, используйте относительный указатель -1 с флагом -b:

journalctl -b -1

1. Также вы можете использовать идентификатор сеанса загрузки для получения данных по сеансу загрузки:

journalctl -b caf0524a1d394ce0bdbcff75b94444fe

1. Для фильтрации временных окон можно использовать опции --since и --until, которые ограничивают вывод записями после или до указанного времени соответственно.

Значения времени могут иметь разные форматы. Для абсолютных значений времени нужно использовать следующий формат:

YYYY-MM-DD HH:MM:SS

Например, чтобы просмотреть все записи с 10 января 2015 г. 17:15, мы введем:

journalctl --since "2015-01-10 17:15:00"

1. Если какие-то компоненты вышеописанного формата будут пропущены, будут применены значения по умолчанию. Например, если дата не будет указана, по умолчанию будет использоваться текущая дата. Если компонент времени отсутствует, для замены будет использоваться значение “00:00:00” (полночь). Поле секунд можно опустить, и тогда для него будет по умолчанию использовано значение “00”:

journalctl --since "2015-01-10" --until "2015-01-11 03:00"

1. Чтобы получить данные за вчерашний день, введите:

journalctl --since yesterday

1. Если вы получили отчеты о перебоях в работе службы, которые начались в 9:00 и закончились час назад, вы можете ввести следующую команду:

journalctl --since 09:00 --until "1 hour ago"

1. Чтобы посмотреть все журналы единицы Nginx в нашей системе, мы можем ввести команду:

journalctl -u nginx.service

1. Обычно при этом также используется фильтрация по времени, чтобы вывести строки, которые вас интересуют. Например, чтобы проверить, как служба работает сегодня, можно ввести команду:

journalctl -u nginx.service --since today

Такая фокусировка особенно полезна, если вы используете возможности чередования записей разных единиц в журнальной системе. Например, если ваш процесс Nginx использует единицу PHP-FPM для обработки динамического контента, вы можете объединить их данные в хронологическом порядке, указав обе единицы:

journalctl -u nginx.service -u php-fpm.service --since today

1. Чтобы вывести только эти сообщения, можно добавить к команде флаг -k или --dmesg:

journalctl -k

1. **Отображение последних журналов**

Чтобы вывести указанное количество записей, вы можете использовать опцию -n, которая работает как tail -n.

По умолчанию отображается 10 последних записей:

journalctl -n

1. Вы можете указать желаемое количество записей, задав число после -n:

journalctl -n 20

1. **Наблюдение за журналами**

Для активного наблюдения за журналами по мере их пополнения можно использовать флаг -f. Это будет работать именно так, как можно ожидать, если у вас есть опыт использования tail -f:

journalctl -f

1. Вы можете определить, сколько места занимает журнал на диске, используя флаг --disk-usage:

journalctl --disk-usage

Journals take up 8.0M on disk.

1. Если вы используете опцию --vacuum-size, вы можете сократить журнал, указав размер. При использовании этой опции старые записи будут удаляться, пока занимаемое журнальной системой место на диске не сократится до требуемого размера:

sudo journalctl --vacuum-size=1G

1. Также можно сократить размер журнала, указав время отсечки с помощью --vacuum-time. Любые записи вне этого времени удаляются. Эта опция позволяет сохранить записи, созданные после истечения определенного времени.

Например, чтобы сохранить записи с прошлого года, вы можете ввести:

sudo journalctl --vacuum-time=1years

Для ограничения роста занимаемого журнальной системой объема можно использовать следующие элементы:

* **SystemMaxUse=**: указывает максимальное пространство на диске, которое может использоваться журнальной системой.
* **SystemKeepFree=**: указывает пространство на диске, которое журнальная система должна оставлять свободным при добавлении записей в журналы.
* **SystemMaxFileSize=**: определяет, до какого размера могут увеличиваться большие файлы журнала на диске до ротации.
* **RuntimeMaxUse=**: указывает, сколько места на диске может использоваться для временного хранения (в /run filesystem).
* **RuntimeKeepFree=**: указывает, сколько места на диске следует оставлять на диске для других целей при записи данных во временное хранилище (в файловой системе /run).
* **RuntimeMaxFileSize=**: указывает, сколько места может занимать отдельный файл журнала во временном хранилище (в файловой системе /run) до ротации.

Задавая эти значения, вы можете контролировать, сколько места на вашем сервере использует journald. Помните, что **SystemMaxFileSize** и **RuntimeMaxFileSize** будут применяться к архивным файлам для достижения заданных ограничений. Это важно помнить при интерпретации подсчета файлов после операции вакуумной очистки.