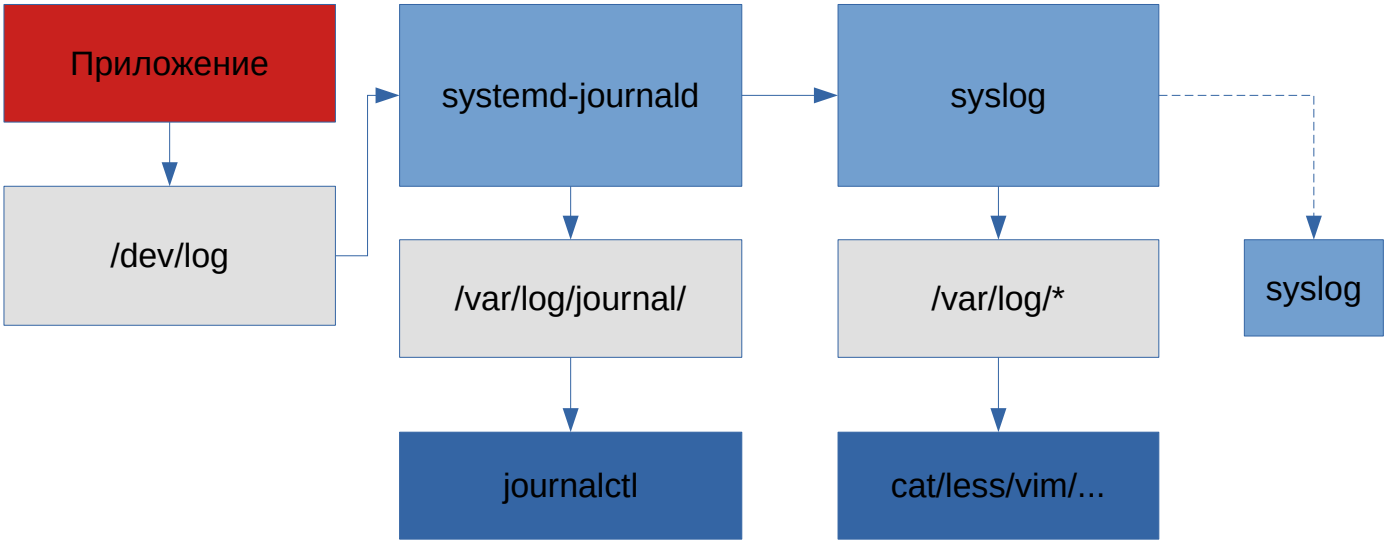
Мониторинг событий

Журналирование в Linux



В процессе своей работы и сама операционная система, и работающие в ней процессы время от времени генерируют специальные уведомления, предназначенные для администратора и технического персонала. Эти уведомления позволяют отследить (а чаще предотвратить) возникновение различных проблемных ситуаций.

Фиксированием этих уведомлений занимается специальный демон syslog и (начиная с версии 7) к нему был добавлен ещё один инструмент - systemd-journald. Разница между ними состоит в том, что syslog фиксирует уведомления в логах, представляющих обычные текстовые файлы, а systemd-journald использует для этих целей специальный журнал, данные в котором хранятся в своём собственном формате и получить доступ к ним можно только используя специальный инструмент - команду journalctl.

Журнальные файлы

Файл	Назначение
/var/log/messages	Общесистемный log-файл. Сюда, как правило попадают уведомления от программ, не имеющих своего отдельного лога.
/var/log/secure	Файл для регистрации событий, связанных с аутентификацией. Сюда попадают уведомления об использовании команд su и sudo, уведомления о неуспешных попытках входа в систему, ssh-сессиях и пр.
/var/log/maillog	Файл для регистрации событий, связанных с работой локального почтового сервера
/var/log/cron	Log-файл планировщика. В него записываются в том числе уведомления о запущенных планировщиком заданиях.
/var/log/boot.log	Log-файл с информацией о загрузке. Порой, уведомления, появляющиеся на экране во время старта системы исчезают слишком быстро. Если не удалось их прочитать, ознакомиться с ними можно уже после того, как система загрузилась.
/var/log/lastlog	Файл для регистрации событий, связанных с аутентификацией пользователей. Для получения информации о дате и времени доступа используется команда lastlog. Без параметров команда генерирует отчет по всем пользователям, можно также указать интересующего пользователя: «lastlog -u root»

Для хранения log-файлов в системе отведена специальная директория /var/log. Некоторые логи лежат непосредственно в ней, для некоторых создаются отдельные поддиректории (обычно так происходит, если у программы есть сразу несколько логов).

Каким образом отчеты о событиях планировщика попадают в один файл, а отчёты о событиях почтовой подсистемы - в другой? Демон syslog-ng в своей работе опирается на конфигурационные файлы, лежащие в директории /etc/syslog-ng/conf.d/ (они должны иметь расширение .conf) и основной конфигурационный файл
/etc/syslog-ng/syslog-ng.conf Директория предназначена для того, чтобы

А) не загромождать основной конфигурационный файл
Б) сразу видеть, какие добавления были сделаны.

В основном конфигурационном файле есть описания источников событий (source), по умолчанию это системные событий, фильтров (filter) и назначений (destination). В директивах log описываются правила, в соответствии с которыми syslog-ng распределяет уведомления по разным файлам. Происходит это так: у каждого события есть два признака: priority (приоритет) и facility. Facility определяет принадлежность уведомления.

Facility и Priority

Priority	Facility
emerg	syslog
crit	auth, authpriv
err	cron
warn	ftp, mail , news, ntp
notice	daemon
info	kern
notice	local0..7

В конфигурационном файле по умолчанию содержатся такие строки:

```
source s_sys {
    system();
    internal();
    # udp(ip(0.0.0.0) port(514));
};

…
destination d_auth { file("/var/log/secure"); };
destination d_mail { file("/var/log/maillog" flush_lines(10)); };
destination d_mlal { usertty("*"); };
…
filter f_auth       { facility(authpriv); };
filter f_mail       { facility(mail); };
filter f_emergency  { level(emerg); };
…
log { source(s_sys); filter(f_auth); destination(d_auth); };
log { source(s_sys); filter(f_mail); destination(d_mail); };
log { source(s_sys); filter(f_emergency); destination(d_mlal); };
```

Вначале указывается источник сообщений — это системные сообщение, полученные через /dev/log, и собственный сообщения самого syslog-ng. Приём сообщений по сети закомментирован.

Далее они определяют, что сообщения от authpriv с любым приоритетом отправляются в файл /var/log/secure, а сообщения от mail и любые сообщения с приоритетом emerg отправляются сразу на консоль пользователя.

Подробная документация по структуре файла syslog-ng.conf содержится в man-странице “man 5 syslog-ng.conf”

Для того, чтобы изменения конфигурационных файлов вступили в силу, необходимо перезапустить syslog-ng.

Для проверки работоспособности syslog-ng (а также возможности отправки сообщений в log-файлы из shell-сценариев) предусмотрена команда logger. Синтаксис её следующий: logger [-p приоритет] “Сообщение”

Например:

```
[demo@localhost ~]$ logger -p err "Message from the some script"
```

Без указания приоритета команда генерирует сообщение с facility=user и priority=notice

Задание 1

Цель упражнения: настроить syslog-ng так, чтобы события с приоритетом crit и выше записывались в файл /var/log/critical.log

1) Сконфигурируйте демон syslog-ng так, чтобы все события с приоритетом crit и выше отправлялись в файл /var/log/critical.log
2) Проверьте работоспособность конфигурации с помощью команды logger

1) Добавьте новый конфигурационный файл или измените существующий:

```
destination d_my { file("/var/log/critical.log"); };
filter f_my { level(crit..emerg); };
log { source(s_sys); filter(f_my); destination(d_my); };
```

2) logger -p crit "Useless critical error"

cat /var/log/critical.log

Использование journalctl

journalctl -p err	Вывести все сообщения с приоритетом err и выше
journalctl --since	Вывести все сообщения с определённой даты
journalctl -o verbose	Вывести все сообщения в подробном виде
journalctl _SYSTEMD_UNIT=	Вывести все сообщения от определённого юнита

systemd-journald хранит уведомления в директории /run/log/journal, причём, не в текстовом виде, а в собственном бинарном формате. Это сделано для увеличения скорости доступа к данным. В связи с этим, принято называть накопленные этой службой данные не логами, а журналом.

Для просмотра журнала systemd используется команда journalctl Она требует прав администратора и по умолчанию показывает события постранично от самых старых к более новым. События, имеющие приоритет warning выделяются жирным шрифтом, а события с приоритетом error и выше окрашиваются красным цветом.

С опцией -n команда выводит на экран 10 последних записей журнала (после n может быть указано количество записей, например, journalctl -n 20), опция --no-pager отменяет постраничный вывод, а опция -f работает также, как и в случае с командой tail: показывает обновления журнала в реальном времени.

Также при отладке может пригодиться способность команды отбирать уведомления с определённым приоритетом. Возможные значения приоритетов приведены выше в таблице 2. Эта способность реализована с помощью опции «p»:

```
[demo@localhost ~]$ sudo journalctl -p err              # уровень err и выше …
[demo@localhost ~]$ sudo journalctl -p crit..alert
```

Иногда возникает необходимость отобрать записи журнала за определённый период. Для указания временных рамок предусмотрены две опции: --since и --until. Формат обеих опций ГГГГ-ММ-ДД ЧЧ:ММ:СС (вместо даты и времени можно пользоваться зарезервированными словами yesterday, today и др.) Например:

```
[demo@localhost ~]$ sudo journalctl  --since today       # за сегодня …
[demo@localhost ~]$ sudo journalctl --since "2018-05-01 21:00"   \
> --until "2018-05-01 21:30"

[demo@localhost ~]$ sudo journalctl --since -30m         # за последние полчаса
```

Для того, чтобы сделать вывод journalctl более детальным, можно использовать параметр "-o verbose". В этом случае запись вида:

Jun 07 03:39:15 localhost.localdomain auditd[661]: Audit daemon is low on disk space for logging

превращается в

Thu 2018-06-07 03:39:15.791871 EDT
[s=8306538df9f941f0a857b87cb73dcca6;i=2ac;b=ca362b5125c24f168e34de3fcb911226;m=ec199e;t=56e08615c6fd1;x=348e329c8453539c]
_UID=0 _GID=0
_SYSTEMD_SLICE=system.slice
_BOOT_ID=ca362b5125c24f168e34de3fcb911226
_MACHINE_ID=d9917077e17e4f3195c6fda0f00cac26
_HOSTNAME=localhost.localdomain
SYSLOG_FACILITY=3
_CAP_EFFECTIVE=1fffffffff
_TRANSPORT=syslog
PRIORITY=1
SYSLOG_IDENTIFIER=auditd
SYSLOG_PID=661
MESSAGE=Audit daemon is low on disk space for logging _PID=661
_COMM=auditd
_EXE=/usr/sbin/auditd
_CMDLINE=/sbin/auditd
_SYSTEMD_CGROUP=/system.slice/auditd.service
_SYSTEMD_UNIT=auditd.service
_SELINUX_CONTEXT=system_u:system_r:auditd_t:s0
_SOURCE_REALTIME_TIMESTAMP=1528357155791871

В результате мы видим значительное количество полей, с помощью которых можно прояснить ситуацию и понять, что за процесс является причиной уведомления. Например:

- _COMM - имя исполняемой команды
- _EXE - путь к исполняемому файлу
- _UID - Идентификатор пользователя, запустившего процесс
- _PID - Идентификатор процесса
- _SYSTEMD_UNIT - Имя юнита systemd, запустившего процесс

Полезным свойством команды journalctl является возможность отбирать сообщения по любому из этих полей, например:

[demo@localhost ~]$ sudo journalctl _SYSTEMD_UNIT=httpd.service

Полный список возможных полей можно посмотреть с помощью команды "man 7 systemd.journal-fields"

Также эти поля можно использовать для определения того, кто генерировал уведомления в системе. Делается это с помощью параметра "-F", например:

[demo@localhost ~]$ sudo journalctl -F _SYSTEMD_UNIT smartd.service chronyd.service auditd.service
[demo@localhost ~]$ sudo journalctl -F _PID 1
294
27
9880

Сохранение журнала при рестарте системы

Обычное место хранения файлов журнала	/run/log/journal
Долговременное хранение файлов журнала	/var/log/journal

Т.к. содержимое директории /run не сохраняется при перезагрузке системы, то и журнал (который лежит в /run/log/journal) начинает писаться с нуля каждый раз, когда система перезагружается. Если возникает необходимость хранить записи журнала, касающиеся не только текущего сеанса, но и предыдущих, то стоить поступить следующим образом:

1. Создать директорию /var/log/journal (если эта директория существует, system-journald считает, что необходимо использовать её вместо /run/log)
   [demo@localhost ~]$ sudo mkdir -m 2775 /var/log/journal
2. Сменить группу, связанную с директорией на systemd-journal
   [demo@localhost ~]$ sudo chown :systemd-journal \
   /var/log/journal
3. Выполнить вызов restart (reload она не поддерживает) для службы systemdjournald
   [demo@localhost ~]$ sudo systemctl restart \ systemd-journald

Посмотреть, какой объем дискового пространства занимает журнал, можно с помощью команды

[demo@localhost ~]$ sudo journalctl --disk-usage

Задание 2

1) Используя команду journalctl, отобразите все записи журнала за текущие сутки
2) Отобразите все записи журнала, касающиеся самого systemd (подсказка: systemd - это процесс с фиксированным PID=1)
3) Отобразите все записи журнала, имеющие приоритет warning и выше и касающиеся процессов, запущенных с привилегиями пользователя root
4) Отобразите все записи журнала за последние полчаса
5) Отобразите последние десять записей, попавших в журнал

1) Используя команду journalctl, отобразите все записи журнала за текущие сутки

```
[demo@localhost ~]$ sudo journalctl  --since today
```

2) Отобразите все записи журнала, касающиеся самого systemd (подсказка: systemd - это процесс с фиксированным PID=1)

```
[demo@localhost ~]$ sudo journalctl  _PID=1
```

3) Отобразите все записи журнала, имеющие приоритет warning и выше и касающиеся процессов, запущенных с привилегиями пользователя root

```
[demo@localhost ~]$ sudo journalctl  -p warning \
_UID=0 --no-pager
```

4) Отобразите все записи журнала за последние полчаса

```
[demo@localhost ~]$ sudo journalctl  --since -30m
```

5) Отобразите последние десять записей, попавших в журнал

```
[demo@localhost ~]$ sudo journalctl  -n
```