

## Семинар #8: Процессы

### Программа ps

Команда	Действие
ps	Показывает процессы текущего терминала.
ps -e	Показать все процессы в системе.
ps -f	Показывает расширенную информацию: uid, pid, ppid, c, stime, tty, time, cmd.
ps -ef	Показать все процессы в расширенном формате.
ps -u alice,bob	Показать процессы пользователей alice и bob.
ps -p 100,200	Показать процессы, со значениями PID равными 100 или 200.
ps -o pid,user,cmd	Вывод в пользовательском формате. Для каждого процесса покажет идентификатор процесса ( <b>pid</b> ), имя пользователя ( <b>user</b> ) и команду которой был запущен процесс ( <b>cmd</b> ).
ps -o pid=,user=,cmd=	То же, но не выведет первую строку с описанием столбцов.
ps -C top	Все процессы с именем программы <b>top</b> .
ps --sort time	Отсортировать по ключевому слову <b>time</b> .
ps --sort -time	Сортировка в обратном порядке по ключевому слову <b>time</b> .
ps aux	Все процессы с кл. словами: user, pid, pcru, rmem, vsz, rss, tty, stat, start, time, cmd

### Распространённые ключевые слова для ps -o и ps --sort

Ключевое слово	Расшифровка
pid	Идентификатор процесса (Process ID).
ppid	Идентификатор родительского процесса (Parent Process ID).
user	Имя пользователя, запустившего процесс.
uid	Идентификатор пользователя.
comm	Имя команды (без аргументов).
cmd	Полная командная строка (включая аргументы).
tty	Управляющий терминал.
time	Совокупное время ЦП (CPU time).
etime	Прошедшее время с момента запуска процесса (Elapsed time).
c	Использование ЦП (в минутах).
pcru	Процент использования ЦП.
rmem	Процент использования физической памяти.
vsz	Размер виртуальной памяти в КиБ (Virtual Size).
rss	Размер резидентного набора в КиБ (Resident Set Size).
stat	Текущее состояние процесса (например, R - выполняется, S - спит).
ni	Значение nice (приоритет).
priority	Приоритет ядра.
start	Время или дата запуска процесса.

Больше опций и ключевых слов можно найти в `man ps`.

# Сигналы

## Распространённые сигналы

Номер	Название	Действие по умолчанию	Примечание
1	SIGHUP	Завершение	Посыпается оболочке и процессам в <code>jobs</code> при закрытии терминала.
2	SIGINT	Завершение	Посыпается передним процессам, при нажатии <code>Ctrl+C</code> . Используется для корректного завершения процесса.
3	SIGQUIT	Создание дампа ядра и завершение	Посыпается, когда пользователь нажимает <code>Ctrl+\`</code> . Корректно завершает процесс. Дополнительно создает файл дампа памяти, полезный для отладки
9	SIGKILL	Завершение без возможности перехвата	Принудительное завершение, не дающее программе возможности сохранить данные или корректно освободить ресурсы. Не может быть перехвачен процессом.
15	SIGTERM	Завершение	Стандартный запрос на корректное завершение процесса. Используется по умолчанию, если не указать сигнал в <code>kill</code>
20	SIGTSTP	Приостановка	Посыпается, когда пользователь нажимает <code>Ctrl+Z</code> . Приостанавливает процесс, который затем может быть возобновлен командами <code>fg</code> или <code>bg</code> .
19	SIGSTOP	Приостановка без возможности перехвата	Принудительно приостанавливает выполнение процесса. Не может быть проигнорирован или перехвачен процессом
18	SIGCONT	Возобновление	Возобновляет выполнение процесса, который был остановлен. сигналами SIGSTOP или SIGTSTP. Используются командами <code>fg</code> и <code>bg</code> для возобновления работы процесса.

## Разновидности команды trap

- `trap 'команда' сигнал`

Перехватывает сигнал и выполняет команду. Например, если мы напишем в скрипте:

```
trap 'echo Hello' TERM
```

то, если процессу этого скрипта послать сигнал SIGTERM, вместо завершения будет выполнена команда `echo Hello`. Данное поведение не наследуется дочерними процессами. Если послать сигнал дочернему процессу, то он завершится, даже если у родительского процесса переопределён обработчик сигнала.

- `trap '' сигнал`

Блокирует поступающий сигнал. Например, если мы напишем в скрипте:

```
trap '' TERM
```

то, если процессу этого скрипта послать сигнал SIGTERM, он не будет реагировать на данный сигнал. Данное поведение наследуется дочерними процессами. Если послать сигнал дочернему процессу, то он не завершится, потому что он унаследует это блокирование сигнала у родительского процесса.

- `trap 'команда' EXIT`

Выполняет команду при выходе из скрипта. Команда выполнится при любом завершении скрипта кроме:

- Завершении программы с помощью сигнала SIGKILL.
- Аварийном завершении программы.

- `trap 'команда' ERR`

Выполняет указанную команду при возникновении ошибки в любой команде скрипта. То есть если какая-нибудь команда в скрипте завершится с ненулевым кодом возврата. За исключением команд, используемых в `if`, `while` и операторах `&&` и `||`.



## Основные файлы в /proc/[pid]

Файл	Содержимое
<code>cmdline</code>	Команда и аргументы, использованные для запуска процесса (разделены нулевыми байтами).
<code>status</code>	Обзор состояния процесса: состояние, PID, PPID, UID/GID, потребление памяти и другое.
<code>stat</code>	Однострочный набор числовых данных о процессе, предназначенный для парсинга.
<code>environ</code>	Переменные окружения процесса (разделены нулевыми байтами).
<code>limits</code>	Текущие ограничения ресурсов, установленные для процесса.
<code>maps</code>	Карты памяти процесса, показывающие, что отображено в адресное пространство процесса.
<code>smaps</code>	Усовершенствованная версия <code>maps</code> , предоставляющая более детальную информацию.
<code>exe</code>	Символическая ссылка на оригинальный исполняемый файл программы.
<code>cwd</code>	Символическая ссылка на текущий рабочий каталог процесса.
<code>fd</code>	Поддиректория с символическими ссылками на открытые файловые дескрипторы процесса.
<code>io</code>	Статистика ввода/вывода процесса (количество байт, считанных и записанных).
<code>statm</code>	Краткая статистика использования памяти в страницах.
<code>cgroup</code>	Информация о группе контроля (cgroups), к которой принадлежит процесс.
<code>net</code>	Поддиректория с информацией о сетевых соединениях процесса
<code>oom_score</code>	Оценка для OOM killer. Чем выше, тем вероятнее процесс будет убит при нехватке памяти.
<code>pagetable</code>	Предоставляет информацию о том, как виртуальные страницы отображаются на физические.
<code>fdinfo</code>	Подробная информация о каждом открытом файловом дескрипторе.
<code>syscall</code>	Номер и аргументы последнего системного вызова, выполненного процессом.



## Директория /proc. Информация о (почти) каждом файле.

### Файлы в /proc

Файл	Содержимое
<code>cmdline</code>	Аргументы командной строки, переданные ядру при загрузке.
<code>version</code>	Версия ядра Linux, компилятор и дата сборки.
<code>kallsyms</code>	Таблица символов ядра.
<code>kmsg</code>	Буфер сообщений ядра (то, что выводит команда dmesg).
<code>modules</code>	Список загруженных в ядро модулей (драйверов).
<code>loadavg</code>	Средняя загрузка системы (load average) за 1, 5 и 15 минут, а также информация о запущенных/общем количестве процессов.
<code>stat</code>	Общая статистика системы
<code>uptime</code>	Время работы системы (uptime) и время простоя (idle) в секундах.
<code>filesystems</code>	Список файловых систем, которые в данный момент поддерживаются ядром.
<code>sys</code>	Директория, содержащая параметры ядра, которые можно просматривать и изменять "на лету".
<code>sysvipc</code>	Информация о механизмах межпроцессного взаимодействия.
<code>keys</code>	Информация о ключах аутентификации, используемых ядром.
<code>cputime</code>	Содержит подробную информацию о каждом процессоре.
<code>schedstat</code>	Статистика планировщика задач (scheduler) для каждого CPU.
<code>meminfo</code>	Показывает общую, свободную, доступную и используемую память, а также информацию о буферах, кэше, страницах и подкачке.
<code>vmstat</code>	Статистика виртуальной памяти
<code>swaps</code>	Информация об активных разделах или файлах подкачки.
<code>buddyinfo</code>	Информация о фрагментации памяти.
<code>slabinfo</code>	Информация о кэшах SLAB/SLUB.
<code>zoneinfo</code>	Подробная информация о зонах памяти
<code>kpage*</code>	Низкоуровневая информация о страницах физической памяти.
<code>pressure</code>	Показывает, насколько система замедляется из-за нехватки ресурсов.
<code>vmallocinfo</code>	Информация об областях памяти, выделенных с помощью <code>vmalloc</code>
<code>mtrr</code>	Конфигурация того, как CPU кэширует различные области памяти.
<code>devices</code>	Список зарегистрированных символьных и блочных устройств
<code>interrupts</code>	Статистика по используемым линиям прерываний.
<code>irq</code>	Директория для управления прерываниями
<code>ioports</code>	Зарегистрированные порты ввода-вывода
<code>iomem</code>	Карта физической памяти (I/O memory map), показывающая, какие диапазоны адресов используются какими устройствами
<code>dma</code>	Используемые каналы прямого доступа к памяти.
<code>acpi</code>	Информация об управлении питанием, батареи, температуре.
<code>asound</code>	Директория для звуковой подсистемы.
<code>bus</code>	Директория с информацией о шинах.
<code>driver</code>	Директория с информацией о драйверах.
<code>fb</code>	Список доступных фреймбуферов (устройства вывода видео).
<code>misc</code>	Различные драйверы и устройства.
<code>tty</code>	Директория с информацией о терминалах.

<code>consoles</code>	Список зарегистрированных консолей.
<code>timer_list</code>	Список всех ожидающих таймеров в ядре.
<code>softirqs</code>	Статистика по "мягким" прерываниям
<code>diskstats</code>	Статистика ввода-вывода для всех блочных устройств.
<code>partitions</code>	Таблица разделов дисков, известная ядру.
<code>mounts</code>	Список всех смонтированных файловых систем.
<code>locks</code>	Файлы, заблокированные ядром
<code>fs</code>	Директория с информацией о состоянии файловых систем (квоты, inode).
<code>net/tcp</code>	Список активных TCP-соединений.
<code>net/udp</code>	Список активных UDP-сокетов.
<code>net/dev</code>	Статистика по сетевым интерфейсам (принятые/отправленные пакеты, ошибки).
<code>net/arp</code>	ARP-таблица ядра.
<code>net/route</code>	Таблица маршрутизации ядра.
<code>self</code>	Символическая ссылка на директорию процесса, который в данный момент читает этот файл.
<code>thread-self</code>	Аналогично self, но указывает на директорию конкретного потока
<code>kcore</code>	Виртуальный файл, представляющий всю физическую память системы в формате ELF core.
<code>cgroups</code>	Информация о контрольных группах, используемых для ограничения ресурсов

## Файлы в /proc/[pid]

Файл	Содержимое
<code>cmdline</code>	Команда и аргументы, использованные для запуска процесса (разделены нулевыми байтами).
<code>environ</code>	Переменные окружения процесса (разделены нулевыми байтами).
<code>status</code>	Обзор состояния процесса: состояние, PID, PPID, UID/GID, потребление памяти и другое.
<code>stat</code>	Однострочный набор числовых данных о процессе, предназначенный для парсинга.
<code>statm</code>	Краткая информация об использовании памяти (в страницах).
<code>auxv</code>	Информация, переданная ядром в пространство пользователя при запуске.
<code>loginuid</code>	ID пользователя, который вошел в систему, инициировавший сессию этого процесса.
<code>sessionid</code>	ID сессии процесса
<code>arch_status</code>	Информация о состоянии, специфичная для архитектуры процессора.
<code>personality</code>	Домен выполнения.
<code>cwd</code>	Ссылка на текущую рабочую директорию процесса.
<code>exe</code>	Ссылка на исполняемый файл, который был запущен.
<code>root</code>	Ссылка на корневую директорию (/) с точки зрения процесса. Для процессов в chroot или контейнерах она будет отличаться от реального корня.
<code>fd</code>	Директория, содержащая файловые дескрипторы, открытые процессом.
<code>fdinfo</code>	Директория с дополнительной информацией о каждом файловом дескрипторе.
<code>maps</code>	Карта памяти процесса. Показывает, какие области памяти куда отображены.
<code>smaps</code>	Подробная карта памяти.
<code>smaps_rollup</code>	Суммарная информация из <code>smaps</code> .
<code>pagemap</code>	Низкоуровневая информация о том, как виртуальные страницы памяти процесса отображаются на физическую память.
<code>mem</code>	Файл, представляющий всю виртуальную память процесса.

<code>clear_refs</code>	Файл для сброса счетчиков "обращения" к страницам памяти.
<code>numa_maps</code>	Информация о политике использования памяти на NUMA-системах.
<code>coredump_filter</code>	Битовая маска, определяющая, какие части памяти процесса будут включены в coredump.
<code>ksm_*</code>	Статистика по KSM (Kernel Same-page Merging).
<code>task</code>	Содержит поддиректории для каждого потока внутри этого процесса.
<code>sched</code>	Информация о политике планировщика для этого процесса
<code>schedstat</code>	Детальная статистика планировщика.
<code>stack</code>	Текущий стек вызовов ядра для этого процесса/потока.
<code>wchan</code>	Если процесс спит (Sleeping), этот файл показывает, в какой функции ядра он "ждет".
<code>syscall</code>	Информация о системном вызове, который в данный момент выполняется процессом.
<code>timerslack_ns</code>	"Допуск" для срабатывания таймеров в наносекундах.
<code>timens_offsets</code>	Смещения времени для временных пространств имен.
<code>patch_state</code>	Статус ядра по применению "живых" патчей (live patching) к этому процессу.
<code>limits</code>	Ограничения на ресурсы, установленные для процесса.
<code>oom_score</code>	Оценка для OOM killer. Чем выше, тем вероятнее процесс будет убит при нехватке памяти.
<code>oom_score_adj</code>	Файлы для корректировки <code>oom_score</code> .
<code>cgroup</code>	Контрольные группы, к которым принадлежит процесс.
<code>cpuset</code>	Какие CPU и узлы памяти разрешено использовать процессу.
<code>autogroup</code>	Информация о группе автоматической балансировки планировщика.
<code>cpu_resctrl_groups</code>	Группы контроля ресурсов CPU.
<code>ns</code>	Директория со ссылками на Namespaces, в которых "живет" процесс.
<code>uid_map</code>	Карты для трансляции ID пользователей в Namespaces.
<code>gid_map</code>	Карты для трансляции ID пользователей в Namespaces.
<code>projid_map</code>	Карты для трансляции ID проектов в Namespaces.
<code>setgroups</code>	Используется в User Namespaces, чтобы разрешить или запретить процессу изменять свой список групп.
<code>attr</code>	Атрибуты безопасности.
<code>io</code>	Статистика по операциям ввода-вывода, выполненным процессом (прочитано/записано байт).
<code>mounts</code>	Список файловых систем, смонтированных с точки зрения этого процесса.
<code>mountinfo</code>	Более подробная информация о монтировании.
<code>mountstats</code>	Статистика по точкам монтирования.
<code>net</code>	Директория с сетевой статистикой, специфичной для сетевого пространства имен процесса.