

ASTRA LINUX GUIDE

PS

PS

ps (processes status — статус процессов) — это встроенная утилита Unix/Linux для просмотра информации, касающейся выбора запущенных процессов в системе: она считывает эту информацию из виртуальных файлов в файловой системе /proc. Это одна из важных утилит для системного администрирования, особенно в рамках мониторинга процессов, чтобы помочь вам понять, что происходит в системе Linux.

Программа ps имеет множество опций для манипулирования выходными данными, однако вы найдёте небольшое количество из них практически полезными для ежедневного использования.

Утилита ps выводит снимок процессов на вашем компьютере Linux. Вы сможете найти процессы по имени, пользователю или даже терминалу с таким количеством деталей, сколько вам нужно.

Управление процессами в Linux

Сердцем всех Linux и Unix-подобных операционных систем является ядро. Среди его многочисленных обязанностей — распределение системных ресурсов, таких как оперативная память и процессорное время. Они должны выполняться в режиме реального времени, чтобы все запущенные процессы получали свою справедливую долю в соответствии с приоритетом каждой задачи.

Иногда задачи могут блокироваться, или зацикливаться, или перестать отвечать по другим причинам. Или они могут продолжать работать, но сожрать слишком много процессорного времени или оперативной памяти, или вести себя каким-то похожим антисоциальным образом. Иногда задачи должны быть убиты для сохранения стабильной работы системы. Разумеется, первых шаг заключается в идентификации проблемного процесса.

Но, возможно, у вас вообще нет проблем с задачами или производительностью. Возможно, вам просто любопытно, какие процессы выполняются на вашем компьютере, и вы хотели бы заглянуть под капот операционной системы Linux. Команда ps удовлетворяет обоим этим требованиям. Она даёт вам снимок того, что происходит внутри вашего компьютера «прямо сейчас».

ps достаточно гибка, чтобы предоставить вам именно ту информацию, которая вам нужна, именно в том формате, который вам нравится. На самом деле, у ps очень много опций. Опции, описанные здесь, будут соответствовать большинству обычных потребностей. Если вы хотите изучить команду ps ещё глубже, то знакомство с командой ps в этой статье и примеры использования ps облегчат вам восприятие справочной страницы.

Программа для показа процессов в Linux

Самый простой способ использовать ps — запустить её без параметров:

ps

ps покажет список процессов в данном терминале.

В выводе присутствует четыре столбца:

- PID: идентификационный номер процесса.
- TTY: имя консоли, на которой пользователь выполнил вход.
- TIME: количество времени центрального процессора, которое потребил процесс.
- CMD: имя команды, которая запустила процесс

Как увидеть все процессы в Linux

Добавление опции -e (выбрать все процессы) сделает так, что ps перечислит процессы, которые были запущены всеми пользователями, а не только пользователем, который запускает команду ps. Поскольку это будет длинный список, то вы можете добавить команду less.

```
ps -e | less
```

Теперь записей о процессах намного больше, но мы видим те же четыре столбца, что и раньше. Вопросительный знак (?) в столбце TTY означает, что процесс запускался не из окна терминала.

Отображение иерархии процессов (дерево процессов в Linux)

Если вы видите, какие процессы запустили другие процессы, то иногда это может помочь выяснить проблему или определить конкретный процесс. Для этого мы используем опцию -H.

```
ps -eH | less
```

Отступы указывают, какие процессы являются родителями каких других процессов.

Чтобы добавить немного ясности, мы можем попросить ps добавить несколько линий ASCII и нарисовать иерархию в виде дерева. Это можно сделать опцией --forest.

```
ps -e --forest | less
```

Это позволит проще отслеживать, какие процессы являются родителями других процессов.

Как напечатать дерево определённого процесса

Вы можете получить дерево процессов только нужной вам программы следующим образом (замените sshd на интересующий вас процесс):

```
ps -f --forest -C sshd
```

Или так:

```
ps -ef --forest | grep -v grep | grep sshd
```

Фильтрация вывода ps по определённым строкам (по имени команды, например)

Вы можете направить вывод из ps через grep и найти нужные записи о процессах по любым строкам. Здесь мы ищем записи, соответствующие поисковому запросу «firefox»:

```
ps -e | grep firefox
```

В этом случае выходные данные представляют собой одну запись для интересующего нас процесса. Конечно, если бы мы запустили несколько экземпляров Firefox, в списке было бы более одного элемента.

Больше столбцов в выводе ps

Чтобы добавить дополнительные столбцы к выводу, используйте параметр -f (полный формат).

```
ps -ef | less
```

Дополнительный набор столбцов включён в вывод ps. Добавлены следующие новые столбцы:

- UID: идентификатор пользователя владельца этого процесса.
- PPID: идентификатор родительского процесса.
- С: Количество детей, которые есть у процесса.
- STIME: Время начала. Время, когда процесс был запущен.

Используя опцию -F (дополнительный полный формат), мы можем получить ещё больше столбцов:

```
ps -eF | less
```

Если у вас маленькое окно терминала, то столбцы, которые мы получаем в этот раз, требуют прокрутки экрана в сторону, чтобы показать их все. Нажатие клавиши «Стрелка вправо» смещает дисплей влево.

Теперь добавились следующие столбцы:

- SZ: размер страниц ОЗУ образа процесса.
- RSS: резидентный размер набора. Это не подкаченная физическая память, используемая процессом.
- PSR: процессор, которому назначен процесс.

Нужно ли указывать дефис перед опциями ps

В некоторых примерах вы можете увидеть использование ps с опциями без дефиса или с длинными вариантами написания опций в стиле GNU. Для совместимости, ps поддерживает все три формата. Опции без дефиса — это стиль BSD и значение опций с дефисом и без может быть различным!

Пример показа процессов в формате BSD:

```
ps au
```

ИЛИ

```
ps axu
```

В этой команде значение опций следующее:

- u — ориентированный на пользователя формат
- a — убирает ограничение «только свои процессы»
- x — убирает ограничение «только процессы с терминалом»

Проще говоря, если использовать вместе а и х, то будут показаны все процессы.

Нужно быть аккуратным, и не забывать ставить дефис если вы используете опции UNIX, поскольку в случае неопределённости ps будет пытаться трактовать в разных вариантах. В этой инструкции кроме рассмотренного примера везде используются опции UNIX.

Поиск процессов по идентификатору процесса

Как только вы нашли идентификатор процесса для интересующего вас процесса, вы можете использовать его с командой ps, чтобы вывести подробную информацию об этом процессе. Для этого используйте спользуйте параметр -r после которого укажите число — идентификатор процесса:

```
ps -r 3403
```

Можно указывать более чем один идентификатор процесса, перечислив их через запятую или через пробел.

Поиск процессов по имени команды

Опция -C КОМАНДА позволяет вам искать процесс, используя имя команды. То есть имя команды, которая запустила процесс. Это несколько отличается от строки команды, которая может включать имена путей и параметры или опции.

```
ps -F -C soffice.bin
```

Процессов может быть несколько если запущено множество экземпляров данной команды:

```
ps -F -C bash
```

Как увидеть потоки процесса

Чтобы вывести все потоки процесса, используйте флаг -H. Опция -L приведёт к показу столбца LWP (light weight process — процесс с малым весом), а также столбца NLWP (number of light weight process — число процессов с малым весом).

```
ps -fL -C httpd
```

Как увидеть процессы определённого пользователя

Чтобы увидеть процессы, принадлежащие конкретному пользователю, используйте опцию `-u` СПИСОК ПОЛЬЗОВАТЕЛЕЙ:

```
ps -u mial
```

Отображаются процессы, принадлежащие учётной записи пользователя `mial`.

Как вывести все процессы запущенные пользователем root

Это частный случай показа процессов определённого пользователя.

Команда ниже позволяет вам просматривать каждый процесс, работающий с привилегиями пользователя `root` (действительный и эффективный идентификатор) в формате пользователя.

```
ps -U root -u root
```

Просмотр групповых процессов

Если вы хотите перечислить все процессы, принадлежащие определённой группе (реальный идентификатор группы (RGID) или имя), введите:

```
ps -fG www-data
```

ИЛИ

```
ps -fG 33
```

Чтобы вывести список всех процессов, принадлежащих эффективному имени группы (или сеанса), введите.

```
ps -fg www-data
```

Листинг процессов по терминалам

Чтобы увидеть процессы, связанные с TTY, используйте опцию `-t` УКАЖИТЕ TTY. При использовании без номера TTY опция `-t` сообщает о процессах, связанных с текущим окном терминала.

```
ps -t 1
```

Все перечисленные процессы связаны с `pts/1`.

Выбор столбцов для отображения

С опцией `-o` ФОРМАТ вы можете выбрать, какие столбцы вы хотите включить в вывод `ps`. Столбцы нужно указывать по имени. В руководстве по `ps`:

```
man ps
```

Вы найдёте длинный список имён столбцов в разделе STANDARD FORMAT SPECIFIERS.

В следующем примере мы выводим потребление процессом времени центрального процессора (pcpu), потребление процессором памяти (rmem) и запустившая его команда вместе с опциями (args):

```
ps -e -o pcru,rmem,args | less
```

Обратите внимание, что опция -o не добавляет столбцы в стандартным, а выводит только запрошенные поля.

Сортировка вывода по столбцам

Вы можете отсортировать вывод, используя опцию --sort. Давайте отсортируем вывод по столбцу CPU:

```
ps -e -o pcru,rmem,args --sort -pcru | less
```

Дефис «-» означает сортировку от большего к меньшему.

Чтобы увидеть десять самых ресурсоемких процессов, передайте вывод через команду head:

```
ps -e -o pcru,args,args --sort -pcru | head -10
```

Мы получаем отсортированный, усечённый список.

Если мы добавим больше столбцов для вывода, мы сможем отсортировать по большему количеству столбцов.

Без дефиса или со знаком «+» сортировка выполняется от меньшего к большему.

Добавим в сортировку столбец rmem:

```
ps -e -o pcru,rmem,args --sort -pcru,rmem
```

Сортировка по-прежнему выполняется по значению pcru, но если для каких-то записей эти значения одинаковые, то выполняется сортировка по rmem для этих значений.

Давайте сделаем вывод результатов немного более полезным и добавим столбец идентификатора процесса (pid), чтобы мы могли видеть номер процесса каждого процесса в нашем листинге.

```
ps -e -o pid,pcru,rmem,args --sort -pcru,rmem | head -10
```

Теперь мы можем идентифицировать процессы.

Все возможные поля ps

Чтобы увидеть все возможные для вывода поля ps выполните такую команду:

```
ps L
```

Эти поля вы можете применять с опцией -o.

Примеры настраиваемого вывода ps

Команда ниже позволяет вам увидеть PID, PPID, имя пользователя и команду процесса.

```
ps -eo pid,ppid,user,cmd
```

Ниже приведён ещё один пример пользовательского формата вывода, показывающий группу файловой системы, значение nice, время начала и истекшее время процесса.

```
ps -p 1154 -o pid,ppid,fgroup,ni,lstart,etime
```

Как найти имя процесса по PID

Чтобы найти имя процесса, используя его PID.

```
ps -p 1154 -o comm=
```

Показать родительский и дочерний процессы

Чтобы выбрать конкретный процесс по его имени, используйте флаг -C, это также отобразит все его дочерние процессы.

```
ps -C sshd
```

Чтобы найти все PID всех экземпляров процесса, что полезно при написании сценариев, которые должны считывать PID из выходных данных (из стандартного входа).

```
ps -C httpd -o pid=
```

Как выключить процесс по идентификатору процесса

Мы рассмотрели ряд способов идентификации процессов, включая имя, команду, пользователя и терминал. Мы также рассмотрели способы идентификации процессов по их динамическим атрибутам, таким как использование процессора и памяти.

Так или иначе, мы можем определить процессы, которые работают. Зная их идентификатор процесса, мы можем (если нужно) остановить любой из этих процессов с помощью команды kill. Если бы мы хотели убить процесс 898, мы бы использовали этот формат:

```
sudo kill 898
```

Как выключить процесс зная его имя

Команда pkill позволяет вам убивать процессы по имени. Убедитесь, что вы определили правильный процесс! Эта команда завершит процесс top.

```
sudo pkill top
```

Как остановить несколько процессов по имени

Если у вас запущено несколько копий процесса, или процесс породил несколько дочерних процессов (как это может сделать Google Chrome), как вы можете выключить их? Это так же просто. Мы используем команду killall.

У нас запущено два экземпляра top:

```
ps -e | grep top
```

Мы можем завершить их обоих с помощью этой команды:

```
sudo killall top
```

Отсутствие ответа означает отсутствие проблем, т. е. оба эти процессы были остановлены.

Прежде чем убить процесс

Убедитесь, что это тот, который вам нужен, и убедитесь, что это не вызовет никаких проблем. В частности, стоит проверить с помощью параметров -H и --forest, чтобы убедиться, что в нем нет важных дочерних процессов, о которых вы забыли.

Устранение неполадок производительности системы Linux

Если ваша система не работает должным образом, например, если она необычно медленная, вы можете выполнить некоторые неполадки системы следующим образом.

Чтобы найти все процессы, потребляющие больше всего памяти и ЦПУ в Linux:

```
ps -eo pid,ppid,cmd,%mem,%cpu --sort=-%mem | head
```

ИЛИ

```
ps -eo pid,ppid,cmd,%mem,%cpu --sort=-%cpu | head
```

Вывод информации о безопасности

Показать контекст безопасности (специально для SELinux) можно следующим образом:

```
ps -eM ИЛИ ps --context
```

С помощью этой команды вы также можете отобразить информацию о безопасности в определённом пользователем формате:

```
ps -eo euser,ruser,suser,fuser,f,comm,label
```

Выполните мониторинг процессов в режиме реального времени с помощью утилиты watch

Наконец, поскольку ps отображает статическую информацию, вы можете использовать утилиту watch для непрерывного обновления информации на экране и мониторинга процессов в режиме реального времени с повторяющимся выводом. В этом примере информация будет одновляться через каждую секунду. Укажите свою собственную команду ps для соответствия вашей цели.

```
watch -n 1 'ps -eo pid,ppid,cmd,%mem,%cpu --sort=-%mem | head'
```