

# Swap (Русский)



Эта страница нуждается в сопровождающем



Статья не гарантирует актуальность информации. Помогите русскоязычному сообществу поддержкой подобных страниц. См. [Команда переводчиков ArchWiki](#)

**Состояние перевода:** На этой странице представлен перевод статьи [Swap](#). Дата последней синхронизации: 15 июня 2018. Вы можете [помочь](#) синхронизировать перевод, если в английской версии произошли [изменения](#).

## Related articles

- [fstab \(Русский\)](#)
- [Power management/Suspend and hibernate \(Русский\)](#)
- [Zswap](#)
- [Swap on video ram](#)
- [ZFS \(Русский\)#Swap volume](#)
- [Dm-crypt/Swap encryption](#)

Эта страница даёт ознакомление с [пространством подкачки и подкачкой страниц](#) в GNU/Linux. Охватывает создание, активацию файлов и разделов подкачки.

Взято из [All about Linux swap space](#):

Linux делит свою физическую RAM (оперативную память) на кусочки памяти, называемые страницами. Подкачка (swapping) это процесс, когда страницы памяти копируются на предварительно сконфигурированное пространство на жестком диске, называемое пространством подкачки, чтобы освободить эту страницу из памяти. Суммарный размер оперативной памяти и пространства подкачки это количество доступной виртуальной памяти.

Поддержка подкачки обеспечивается ядром Linux и утилитами в пользовательском пространстве из [util-linux](#) пакета.

## Contents

[hide]

- 1Пространство подкачки
- 2Раздел подкачки
  - 2.1Активация используя systemd
  - 2.2Отключение подкачки
- 3Файл подкачки
  - 3.1Вручную
    - 3.1.1Создание файла подкачки
    - 3.1.2Удаление файла подкачки
  - 3.2Автоматически
    - 3.2.1systemd-swap
- 4Подкачка с USB устройства
- 5Шифрование подкачки
- 6Производительность
  - 6.1Swappiness
  - 6.2VFS cache pressure
  - 6.3Приоритет
  - 6.4Использование zswap или zram
  - 6.5Чередование

## Пространство подкачки

Пространство подкачки может быть разделом диска или файлом. Пользователи могут создать пространство подкачки во время установки или позднее в любое желаемое время. Пространство подкачки может быть использовано для двух целей, расширить виртуальную память за пределы установленной оперативной памяти (RAM), а также для сохранения данных при гибернации (suspend-to-disk).

Иногда стоит включать Swap в зависимости от установленной оперативной памяти и количества требований для запуска желаемых программ. Если количество оперативной памяти меньше требуемого, тогда стоит включить подкачку. Это позволяет избежать [состояния нехватки памяти \(OOM\)](#), при котором механизм ядра Linux, OOM Killer, будет автоматически пытаться освободить память, убивая процессы. Чтобы увеличить количество виртуальной памяти до требуемого уровня, добавьте необходимую разницу как пространство подкачки. Например, если программа требует 7,5 GB памяти для запуска, а у вас установлено 4 GB оперативной памяти, добавьте разницу 3,5 GB как подкачку. В будущем добавляйте больше пространства к подкачке, учитывая требования. Это вопрос личных предпочтений если вы считаете, что программы должны быть убиты, вместо включения подкачки. Самый большой недостаток в подкачке это снижение производительности, см. раздел [#Производительность](#)

Для проверки статуса подкачки, используйте:

```
$ swapon --show
```

Или

```
$ free -h
```

*free* также покажет недостаток памяти, который может быть исправлен включением или увеличением подкачки.

**Примечание:** Преимущества в производительности между файлом и разделом подкачки нет, оба варианта обрабатываются одинаково.

## Раздел подкачки

Раздел подкачки может быть создан различными GNU/Linux [утилитами разметки](#). Разделы подкачки обычно обозначаются как тип `82`. Хотя есть возможность использовать разные типы как подкачку, рекомендуется использовать тип `82`, в большинстве случаев [systemd](#), будет автоматически определять его и монтировать (см. ниже)

Для установки раздела как область Linux подкачки, можно использовать `mkswap`. Например:

```
# mkswap /dev/sdxy
```

**Важно:** Все данные на указанном разделе будут утеряны.

Для подключения устройства как подкачку:

```
# swapon /dev/sdxy
```

Чтобы подключить этот раздел подкачки при загрузке, добавьте запись в [fstab](#):

```
UUID=<UUID> none swap defaults 0 0
```

где <UUID> может быть получен из команды:

```
lsblk -no UUID /dev/sdxy
```

**Совет:** Предпочтительно использовать UUID и LABEL, чем имена устройств полученные от ядра (*/dev/sd\**), т.к. порядок устройств может измениться в будущем. Смотри [fstab](#).

**Примечание:**

- Эта fstab запись необязательна если раздел подкачки находится на устройстве, использующий GPT разметку, см. следующий подраздел.
- Если используется SSD с поддержкой [TRIM](#), учтите использование `defaults,discard` в [fstab](#) строке подключения подкачки. Если в ручную активировать подкачку с помощью *swapon*, используйте параметр `-d/--discard`, который делает тоже самое. Подробности смотри в [swapon \(8\)](#).

**Важно:** Включение discard в RAID установке с использованием mdadm, приведёт к блокировке системы при загрузке и во время выполнения, если использовать *swapon*.

## Активация используя systemd

Активация разделов подкачки в [systemd](#) базируется на двух различных механизмах. Оба исполняются в `/usr/lib/systemd/system-generators`. Генераторы запускаются при старте системы и создают нативные systemd юниты для монтирования. Первый `systemd-fstab-generator`, читает fstab, чтобы генерировать юниты, включая юнит для подкачки. Второй `systemd-gpt-auto-generator`, осматривает корневой диск, чтобы генерировать юниты. Это операция проходит только на GPT дисках и может идентифицировать разделы подкачки по их типу коду 82.

## Отключение подкачки

Чтобы деактивировать определенное пространство подкачки:

```
# swapoff /dev/sdxy
```

Также можно использовать `-a` ключ, чтобы деактивировать все пространства подкачки.

С тех пор, как systemd управляет подкачкой, она вновь будет активирована при старте системы, для долговременного отключения автоматической активации найденных пространств подкачки, выполните `systemctl --type swap`, чтобы найти связанные со `.swap` юниты и замаскируйте ([systemctl mask юнит](#)) их.

## Файл подкачки

Как альтернатива к созданию целого раздела, файл подкачки даёт возможность менять свой размер на лету, а также его гораздо легче полностью удалить. Это может быть особенно важно, если дисковое пространство ограничено (например, небольшие SSD)

**Важно:** [Btrfs](#) не поддерживает файлы подкачки. Несоблюдение этого предупреждения, может стать результатом разрушения файловой системы. Пока что, файл подкачки может быть использован в Btrfs, если смонтирован через `loop`-устройство, тогда будет сильно уменьшена производительность подкачки.

## Вручную

### Создание файла подкачки

Использовать под суперпользователем `fallocate`, чтобы создать файл подкачки размером на свой выбор (M = [Mebibytes](#), G = [Gibibytes](#)). Например создание 512 MiB файла подкачки:

```
# fallocate -l 512M /swapfile
```

**Примечание:** `fallocate` может вызвать проблемы с некоторыми файловыми системами, такими как: [F2FS](#) или [XFS.11](#). Как вариант, используйте `dd`, наиболее надёжный, но медленный.

```
# dd if=/dev/zero of=/swapfile bs=1M count=512
```

Установите права доступа (всеми читаемый файл подкачки это огромная локальная уязвимость)

```
# chmod 600 /swapfile
```

После создания файла нужного размера, форматируйте его в подкачку:

```
# mkswap /swapfile
```

Активируйте файл подкачки:

```
# swapon /swapfile
```

В завершении, отредактируйте [fstab](#), добавив запись для файла подкачки:

```
/etc/fstab  
  
/swapfile none swap defaults 0 0
```

**Примечание:** Не забудьте добавить `discard`, если у вас SSD с поддержкой TRIM

```
# swapon --discard /swapfile
```

```
/etc/fstab  
  
/swapfile none swap defaults,discard 0 0
```

### Удаление файла подкачки

Чтобы удалить файл подкачки, сначала нужно отключить подкачку, а затем файл может быть удален:

```
# swapoff -a  
# rm -f /swapfile
```

В завершении, удалите соответствующую запись из `/etc/fstab`.

## Автоматически

### systemd-swap

Установить `systemd-swap` пакет. Установить `swapfc_enabled=1` в *Swap File* *Chunked* разделе файла `/etc/systemd/swap.conf`. Start/enable `systemd-swap` сервис. Посетить страницу авторов на GitHub для получения подробностей и установить рекомендуемую конфигурацию.

**Примечание:** Если журнал продолжает показывать следующее предупреждение `systemd-swap[...]: WARN: swapFC: ENOSPC` и файл подкачки не создаётся, вы должны установить параметр `swapfc_force_preallocated=1` в `/etc/systemd/swap.conf`.

## Подкачка с USB устройства

Благодаря модульности, предлагаемой Linux, мы можем иметь множество разделов подкачки на различных устройствах. Если у вас полностью заполнен жесткий диск, то можно использовать USB устройство как временный раздел подкачки. Однако, этот метод имеет серьёзные недостатки:

- USB устройство медленнее чем жесткий диск
- Flash память имеет ограниченное количество циклов записи. Использование его как раздела подкачки, может быстро убить его.

Чтобы добавить USB устройство как подкачку, сначала необходимо разметить USB флешку для подкачки как описано в секции #Раздел подкачки.

Далее откройте `/etc/fstab` и добавьте

```
pri=0
```

в опции монтирования *первоначальной* записи подкачки, таким образом USB подкачка будет иметь приоритет записи над старым разделом.

Данная инструкция будет работать и для других устройств хранения, таких как SD карты и т.д.

## Шифрование подкачки

Смотри: [dm-crypt/Swap encryption](#).

## Производительность

Операции подкачки как правило существенно медленнее чем непосредственный доступ к RAM. Отключение подкачки полностью для повышения производительности, иногда может привести к ухудшению, поскольку это уменьшает доступную память для VFS кеша, вызывая более частые и дорогостоящие операции ввода/вывода.

Значения подкачки можно настроить, чтобы помочь производительности:

### Swappiness

*Swappiness* [sysctl](#) параметр представляющий частоту использования пространства подкачки. *Swappiness* может иметь значение от 0 до 100, значение по умолчанию = 60. Низкое значение заставляет ядро избегать подкачки, высокое значение позволяет ядру использовать подкачку *наперёд*. Использование низкого значения на достаточном количестве памяти, улучшает отзывчивость на многих системах.

Чтобы проверить текущее значение *swappiness*:

```
$ cat /sys/fs/cgroup/memory/memory.swappiness
```

или

```
$ cat /proc/sys/vm/swappiness
```

**Примечание:** Т.к. `/proc` менее организован и сохраняется только для обратной совместимости, вместо него предпочтительнее использовать `/sys`.

Чтобы временно установить значение `swappiness`:

```
# sysctl vm.swappiness=10
```

Чтобы постоянно установить значение `swappiness`, отредактируйте (создайте) конфигурационный файл `sysctl`

```
/etc/sysctl.d/99-sysctl.conf
```

```
vm.swappiness=10
```

Чтобы проверить и больше узнать, почему оно так работает, посмотрите [эту статью](#).

## VFS cache pressure

Другой `sysctl` параметр, который действует на производительность подкачки это `vm.vfs_cache_pressure`, он контролирует склонность ядра к применению памяти, которая используется для кэширования VFS caches, напротив кэширования страниц и подкачки. Увеличение этого значения увеличивает коэффициент с которым VFS caches применяется [\[2\]](#). Для подробной информации смотри [документацию ядра Linux](#).

## Приоритет

Если у вас больше одного файла или раздела подкачки, вы должны учитывать присвоение приоритетного значения (от 0 до 32767) для каждой области подкачки. Система будет использовать области подкачки с высоким приоритетом, перед использованием областей с низким приоритетом. Например, если у вас быстрый диск (`/dev/sda`) и медленный (`/dev/sdb`), назначьте высокий приоритет для подкачки расположенной на быстром устройстве. Приоритет может быть назначен в [fstab](#) как `pri` параметр:

```
/dev/sda1 none swap defaults,pri=100 0 0  
/dev/sdb2 none swap defaults,pri=10 0 0
```

Или как параметр в `swapon` `--priority`

```
# swapon --priority 100 /dev/sda1
```

Если две или более областей будут иметь одинаковый приоритет и он будет самым высоким из доступным приоритетов, то страницы будут распределяться по кругу между областями.

## Использование `zswap` или `zram`

[Zswap](#) это особенность ядра Linux, обеспечивающая сжатие обратного кэша для страниц подкачки. Она увеличивает производительность и уменьшает операции ввода/вывода. [ZRAM](#) создаёт виртуальный сжатый файл подкачки в памяти, как альтернатива файлу подкачки на диске.

## Чередование

Нет необходимости использовать [RAID](#) для повышения производительности подкачки. Ядро самостоятельно может чередовать подкачку на нескольких устройствах, если вы присвоите им одинаковый приоритет в `/etc/fstab`. Для подробной информации смотри [The Software-RAID HOWTO](#).

[Categories:](#)

- [Русский](#)
- [File systems \(Русский\)](#)