

FSTAB

Каждая строка в fstab описывает раздел, который нужно примонтировать к определенной точке монтирования. Мы можем указать файловую систему, опции монтирования, а также нужно ли проверять файловую систему на ошибки.

Сначала давайте рассмотрим синтаксис одной строки fstab:

устройство **точка_монтирования** **файловая_система** **опции** **резерв{0,1}** **проверка{0,1,2}**

Теперь подробнее рассмотрим что означает каждый пункт:

- **Устройство** - это раздел диска, который вам нужно примонтировать. Его можно указать в формате файла устройства Linux в каталоге /dev/, например, /dev/sda1 или с помощью уникального идентификатора UUID, тогда формат записи будет таким UUID="XXXX-XXXX-XXXX-XXXX", также возможна запись с помощью метки, например, LABEL=home;
- **Файловая система** указывает в какой файловой системе нужно монтировать это устройство, например, ext4, ext3, ext2, btrfs;
- **Точка монтирования** - куда нужно примонтировать это устройство, например, /home, /boot, /mnt;
- **Опции** - параметры монтирования файловой системы, рассмотрим подробнее ниже;
- **Проверка** - указывает в какой очереди нужно проверять устройство на ошибки, 1 - в первую очередь, 2 - вторую, 0 - не проверять;
- **Резерв** - указывает нужно ли делать резервную копию раздела, может принимать значения только 0 и 1.

В основном с этими пунктами все должно быть понятно, интерес вызывают только опции монтирования fstab. Для разных файловых систем они могут немного отличаться, но есть стандартные. И поскольку чаще всего используются файловые системы семейства ext, то мы будем ориентироваться на них.

Начнем с общих для всех файловых систем опций:

- **sync** - записывать на диск все изменения сразу после того, как они были выполнены, не использовать кэширование для записываемых данных. Может понадобиться для извлечения флешки без размонтирования, но сильно снижает производительность;
- **async** - использовать кэш при записи данных, увеличивает производительность, используется по умолчанию;
- **atime** - сохранять время последнего доступа к файлу;
- **noatime** - не сохранять время последнего доступа, полезно для ssd и флешек;
- **relatime** - обновлять время доступа только при изменении файла, необходимо для работы многих программ;
- **norelatime** - отключить relatime;
- **strictatime** - обновлять время доступа всегда, отключает действие предыдущих опций;
- **auto** - автоматически монтировать при загрузке, действие по умолчанию;
- **noauto** - не монтировать при загрузке;
- **defaults** - использовать опции монтирования fstab по умолчанию - rw,suid,dev,exec,auto,nouser,async;

- **dev** - интерпретировать блочные устройства;
- **nodev** - не интерпретировать блочные устройства;
- **diratime** - аналогично atime только для каталогов;
- **dirnoatime** - аналогично noatime, для каталогов;
- **exec** - разрешить выполнять программы на этом разделе;
- **noexec** - запретить выполнять программы на этом разделе;
- **group** - разрешить другим пользователям кроме root монтировать этот раздел, если их группа совпадает с указанной, обычно используется вместе с noauto;
- **nofail** - не сообщать об ошибках;
- **mand** - разрешить блокирование файловой системы, нужно для некоторых антивирусов;
- **nomand** - запретить блокирование файловой системы;
- **suid** - разрешить выполнение программ с флагом suid от имени другого пользователя;
- **nosuid** - игнорировать флаг suid;
- **owner** - разрешить указанному пользователю монтировать устройство;
- **ro** - монтировать только для чтения;
- **rw** - монтировать для чтения и записи;
- **users** - разрешить монтирование любому пользователю;
- **umask** - установить права доступа к файлам и папкам на этом разделе;
- **uid** - задает владельца каталога, по умолчанию root;
- **gid** - задает группу владельца каталога.

Когда мы рассмотрели всю теорию, настройка fstab не вызовет у вас проблем. Теперь давайте рассмотрим стандартное содержимое файла fstab, а также как выполняется монтирование. Вот так выглядит монтирование корня:

```
/dev/sda2 / ext4 defaults 0 1
```

Здесь в качестве корневой файловой системы будет монтироваться раздел /dev/sda2 с файловой системой ext4 и опциями по умолчанию defaults. Резервная копия не используется, и вообще, этот параметр не читается системой инициализации и сейчас нас не интересует. Для домашнего раздела все будет выглядеть очень похоже:

```
/dev/sda3 /home ext4 defaults,noexec 0 2
```

Только тут мы указали, что его нужно проверить вторым, а в опциях еще добавили, что программы оттуда выполнять нельзя. Более интересен пункт для дополнительного диска, на котором хранятся файлы:

```
/dev/sda4 /media/files/ ext4 noauto,users,rw 0 0
```

Тут мы указываем, что раздел не нужно монтировать во время загрузки, но его могут подключить любые пользователи в режиме для чтения и записи. Ни проверка ни резервная копия не выполняются. Для раздела подкачки файловая система указывается swap, а точка монтирования none:

```
/dev/sda5 none swap defaults 0 0
```

Также вы можете примонтировать немножко оперативной памяти:

```
tmpfs /tmp tmpfs nodev,nosuid,noexec,size=100M 0 0
```

Таким образом, мы примонтировали оперативную память в /tmp и теперь система будет работать немного быстрее. Вот что получилось:

```
/dev/sda2 / ext4 defaults 0 1
/dev/sda3 /home ext4 defaults,noexec 0 2
/dev/sda4 /media/files/ ext4 noauto,users,rw 0 0
/dev/sda5 none swap defaults 0 0
tmpfs /tmp tmpfs nodev,nosuid,noexec,size=100M 0 0
```

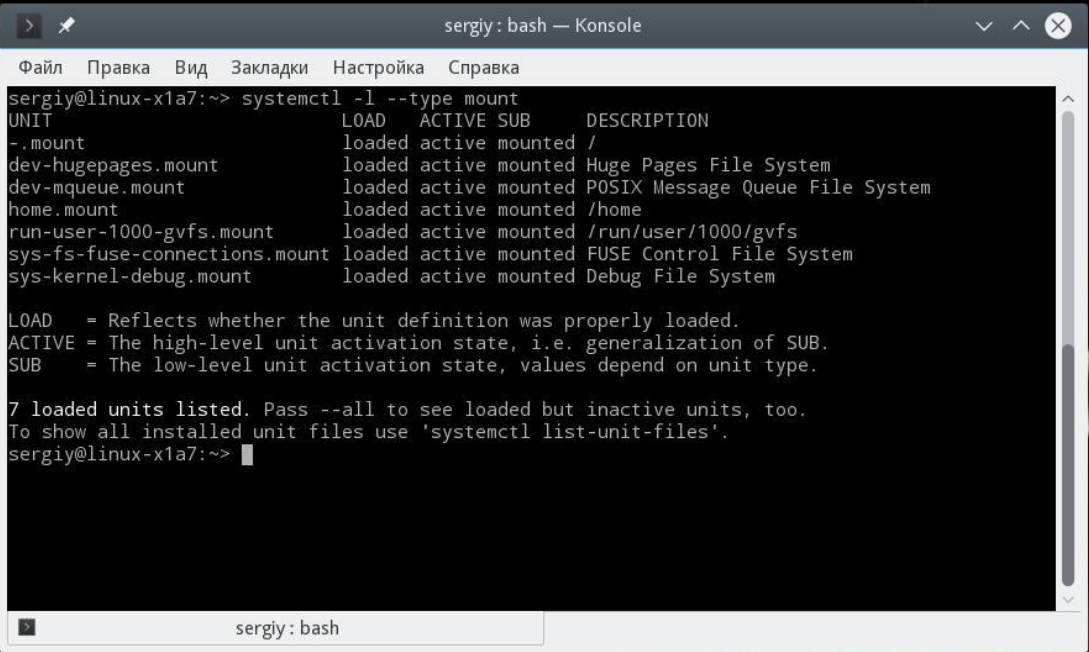
А теперь, как я и обещал рассмотрим как выполняется монтирование с помощью systemd.

АВТОМАТИЧЕСКОЕ МОНТИРОВАНИЕ В SYSTEMD

Система инициализации Systemd анализирует /etc/fstab при загрузке и автоматически генерирует все нужные файлы юнитов на основе описанных там точек монтирования, а уже потом их загружает.

Вы можете посмотреть все созданные в systemd точки монтирования такой командой:

```
systemctl -l --type mount
```



```
sergiy@linux-x1a7:~> systemctl -l --type mount
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
-.mount                            loaded active mounted /
dev-hugepages.mount                loaded active mounted Huge Pages File System
dev-mqueue.mount                   loaded active mounted POSIX Message Queue File System
home.mount                         loaded active mounted /home
run-user-1000-gvfs.mount            loaded active mounted /run/user/1000/gvfs
sys-fs-fuse-connections.mount       loaded active mounted FUSE Control File System
sys-kernel-debug.mount             loaded active mounted Debug File System

LOAD    = Reflects whether the unit definition was properly loaded.
ACTIVE  = The high-level unit activation state, i.e. generalization of SUB.
SUB     = The low-level unit activation state, values depend on unit type.

7 loaded units listed. Pass --all to see loaded but inactive units, too.
To show all installed unit files use 'systemctl list-unit-files'.
sergiy@linux-x1a7:~>
```

Но нам ничего не мешает самим создать такой файл точки монтирования. Это очень просто, давайте рассмотрим синтаксис:

[Unit]

Description=описание

[Mount]

What=адрес_раздела

Where=точка монтирования

Type=файловая система
Options=опции монтирования

[Install]
WantedBy=multi-user.target

Например, для той же домашней папки автоматическое монтирование fstab будет выглядеть вот так, имя обязательно должно состоять из точки монтирования, в которой слеша заменена на дефис:

```
sudo vi /etc/systemd/system/home.mount  
[Unit]  
Description=Mount System Home Directory
```

```
[Mount]  
What=/dev/sda3  
Where=/home  
Type=ext4  
Options=defaults,noexec
```

```
[Install]  
WantedBy=multi-user.target
```

Точно такой же файл может быть создан для любого устройства. Теперь для монтирования достаточно набрать:

```
sudo systemctl start home.mount  
А чтобы добавить эту точку монтирования в автозагрузку выполните:
```

```
sudo systemctl enable home.mount  
Вот и все теперь вы знаете не только как выполняется монтирование fstab, но и автоматическое монтирование в systemd.
```

ВЫВОДЫ

Как видите, редактирование fstab не так уж сложно, как может показаться на первый взгляд. С этим справиться даже новичок в Linux если изучить параметры fstab и синтаксис самого файла. Если у вас остались вопросы, спрашивайте в комментариях!