



ОНЛАЙН-ОБРАЗОВАНИЕ

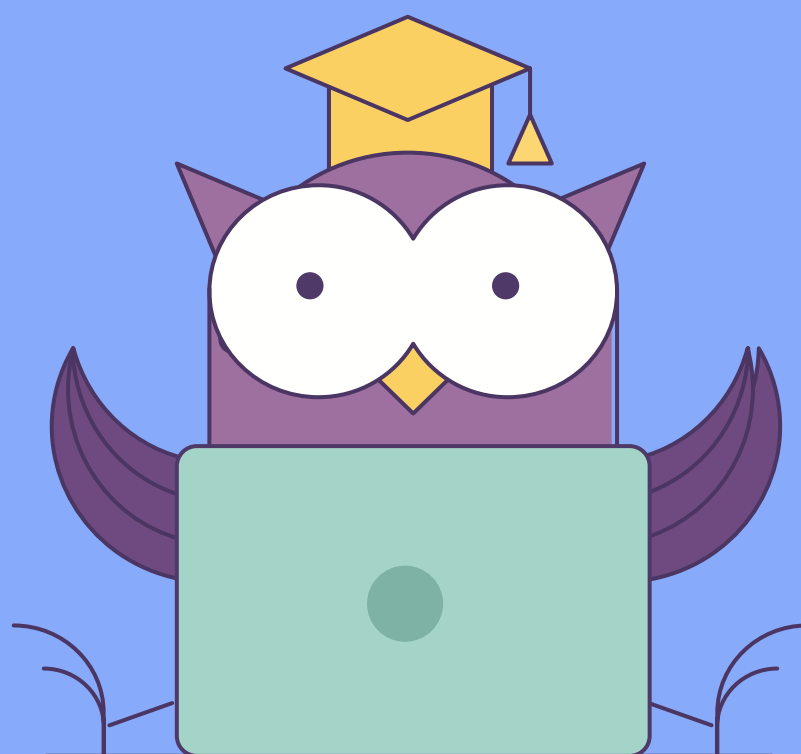


NFS, FUSE

Курс «Администратор Linux»



Меня хорошо слышно && видно?



Напишите в чат, если есть проблемы!

Ставьте ☐ + если все хорошо
Ставьте ☐ - если есть проблемы

- NFS
- FUSE
- SSHFS

NFS

- Клиент-серверный протокол доступа к файловым ресурсам
- SUN Microsystem, первая версия 1984 год - только внутренние цели.
- NFS v2 - 1989
- NFS v3 - 1995
- NFS v4 - 2000

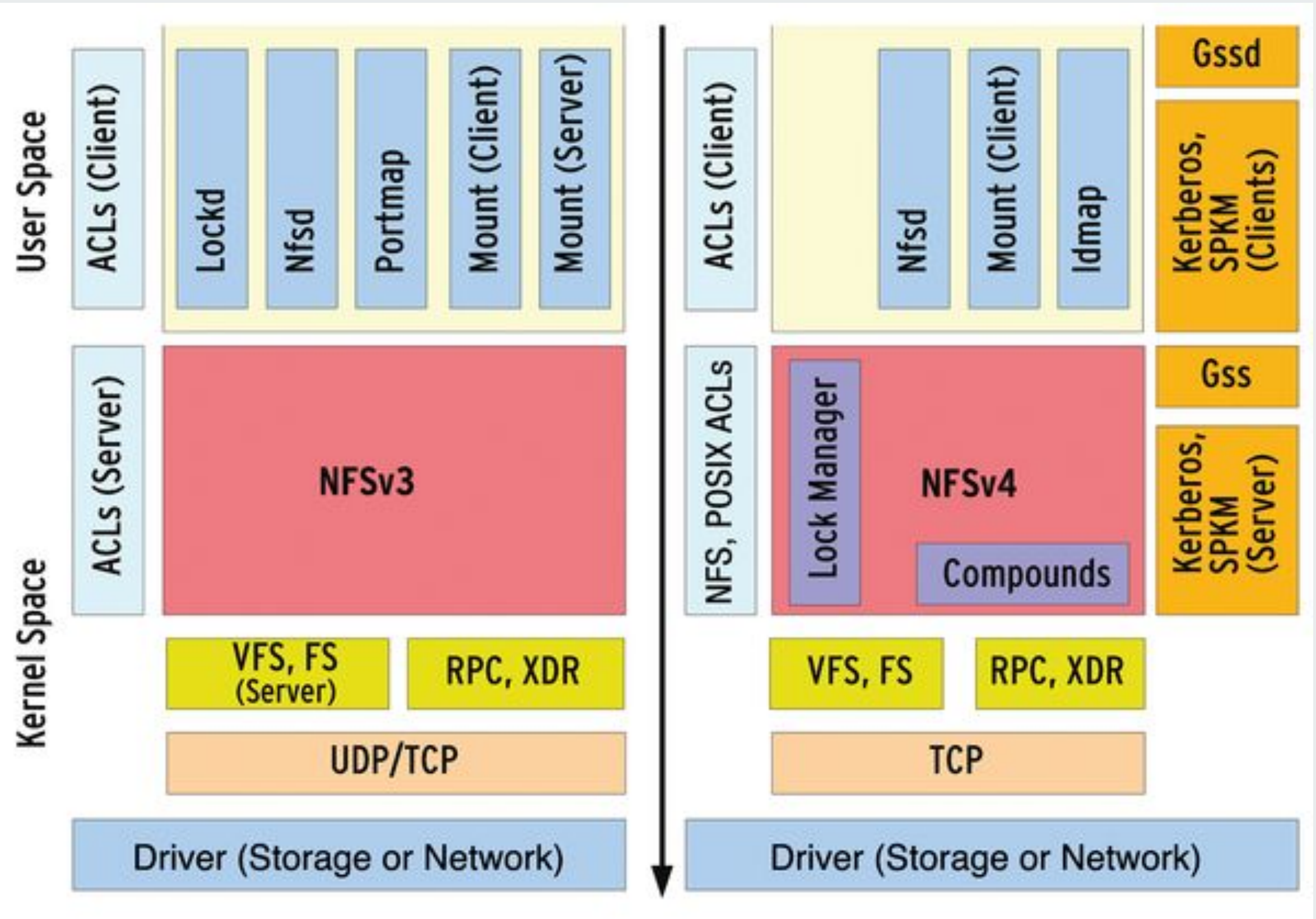
- Документы
- Изображения (фотографии)
- Видео файлы
- Архивы
- ...

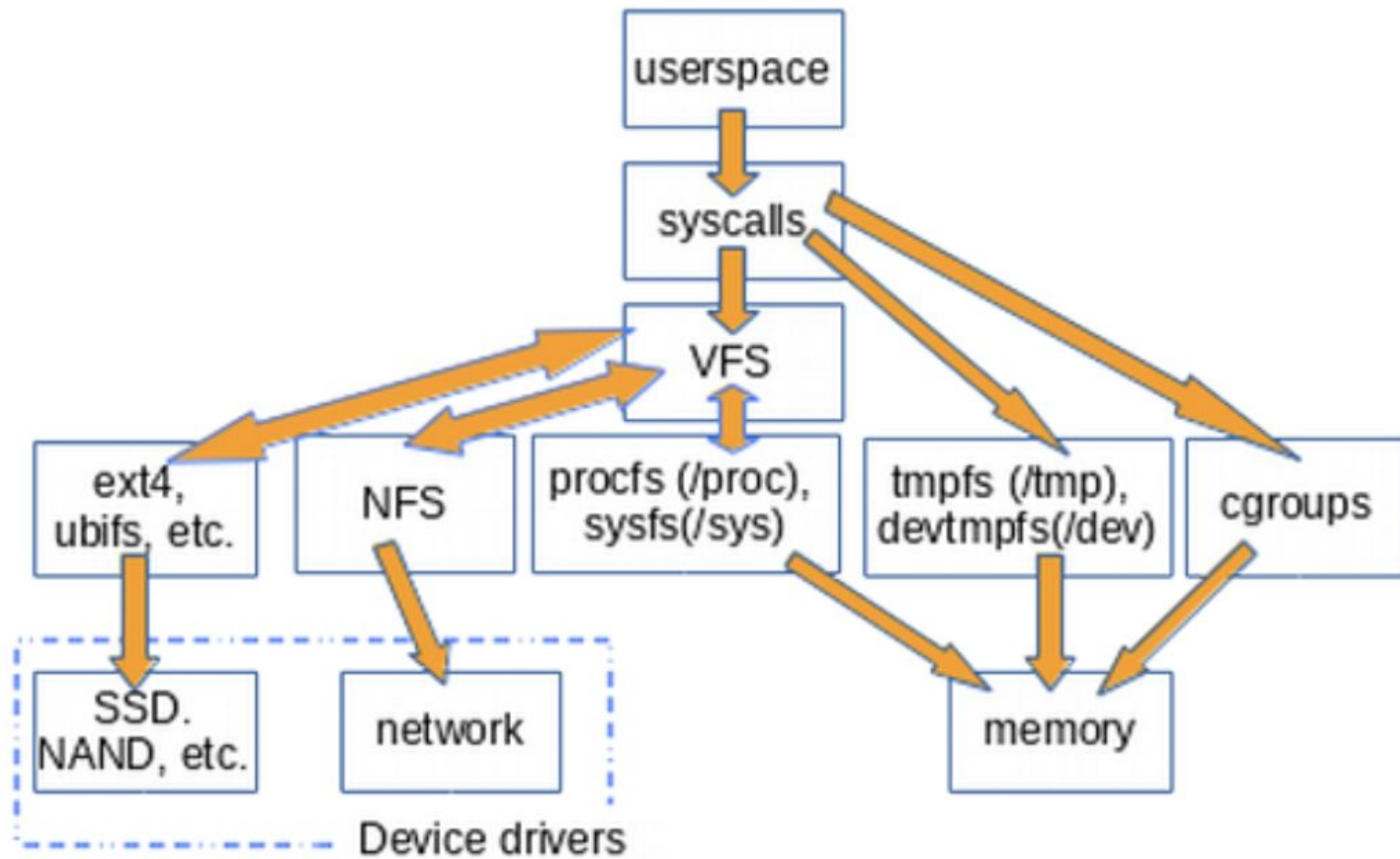
Что мы хотим от файлового хранилища?

- Взаимодействие с хранилищем как с обычным диском (монтирование)
- Доступность из разных мест
- Предоставление публичного/закрытого доступа
- Предоставление прав на чтение/запись/добавление/удаление
- Наличие/отсутствие/необходимость локальной копии
- Мультиредактирование, синхронизация
- Редактирование файлов/документов в вебе

- Асинхронная запись
- Устойчивость к ошибкам
- Работа с большими файлами (>2G)
- Stateless
- Построен над RPC (Remote Procedure Call)
- Блокировки через NLM (lockd)
- Работает поверх TCP и UDP (port 2049)

- Блокировки и управление общим доступом реализованы внутри основного демона NFS
- Не требует rpcbind, lockd, rpc.statd
- Поддержка локального кеша
- Только TCP (port 2049) -> Statefull
- Поддержка ACL (Access Control Lists)
- Расширенная аутентификация (Kerberos)





VFS — это «слой оболочки» между системными вызовами и реализациями определенных **file_operations**, таких как ext4 и procfs.

Функции **file_operations** могут взаимодействовать либо с драйверами устройств, либо с устройствами доступа к памяти.

tmpfs, devtmpfs и cgroups не используют **file_operations**, а напрямую обращаются к памяти.

Существование VFS обеспечивает возможность переиспользовать код, так как основные методы, связанные с файловыми системами, не должны быть повторно реализованы каждым типом файловой системы.

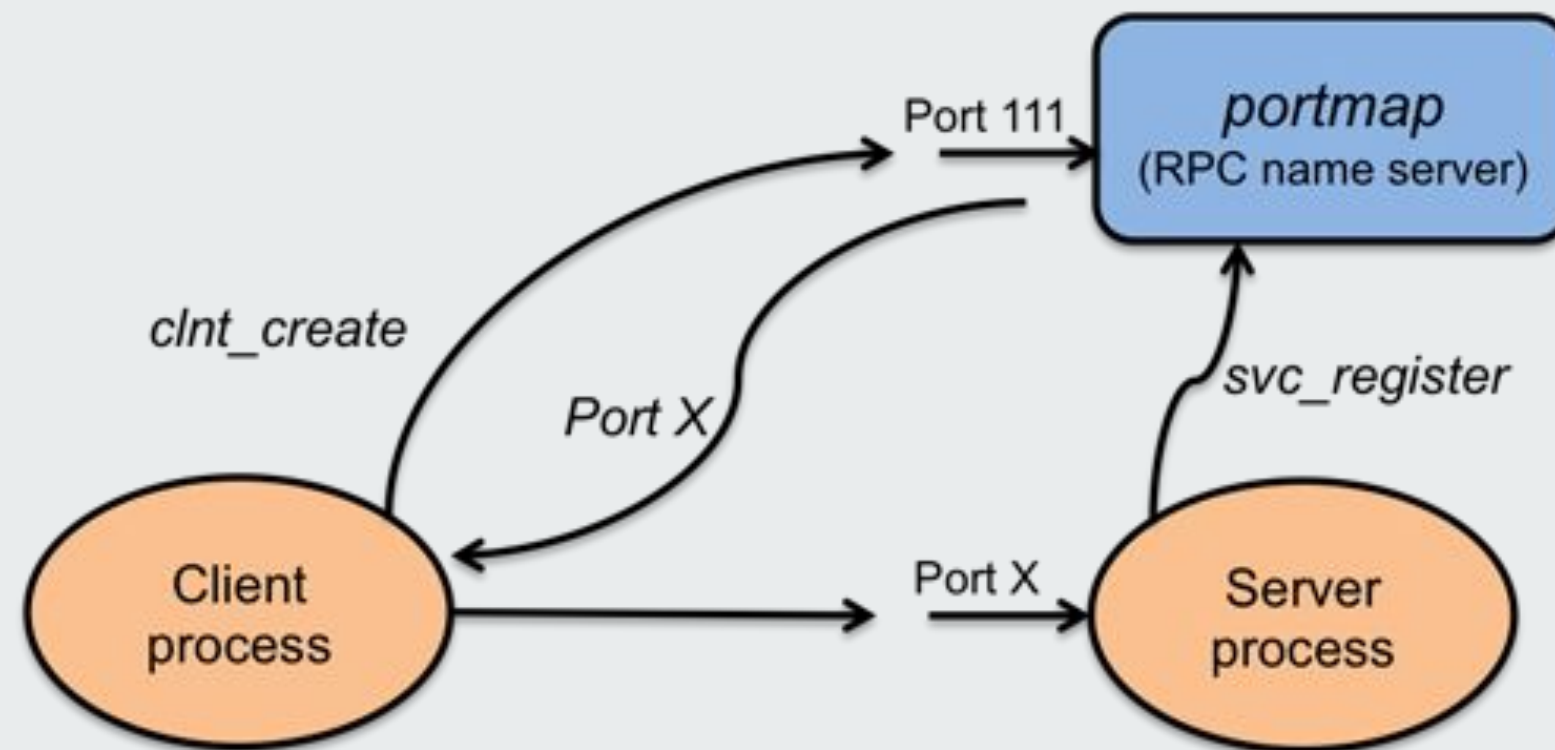
- **rpc.nfsd** - Основной демон сервера
- **rpc.mountd** - обрабатывает запросы клиентов на монтирование каталогов.
- **rpc.statd** - (Network Status Monitor - NSM). Корректно снимает блокировку после сбоя/перезагрузки. Для уведомления о сбое использует программу /usr/sbin/sm-notify.
- **rpc.lockd** - (NFS lock manager (NLM)) обрабатывает запросы на блокировку файлов. (В современных ядрах не нужен)
- **rpc.idmapd** - Для NFSv4 на сервере преобразует локальные **uid/gid** пользователей в формат вида **имя@домен**, а на клиенте обратно
- **rpc.rquotad** - данные о квотах пользователя на сервере

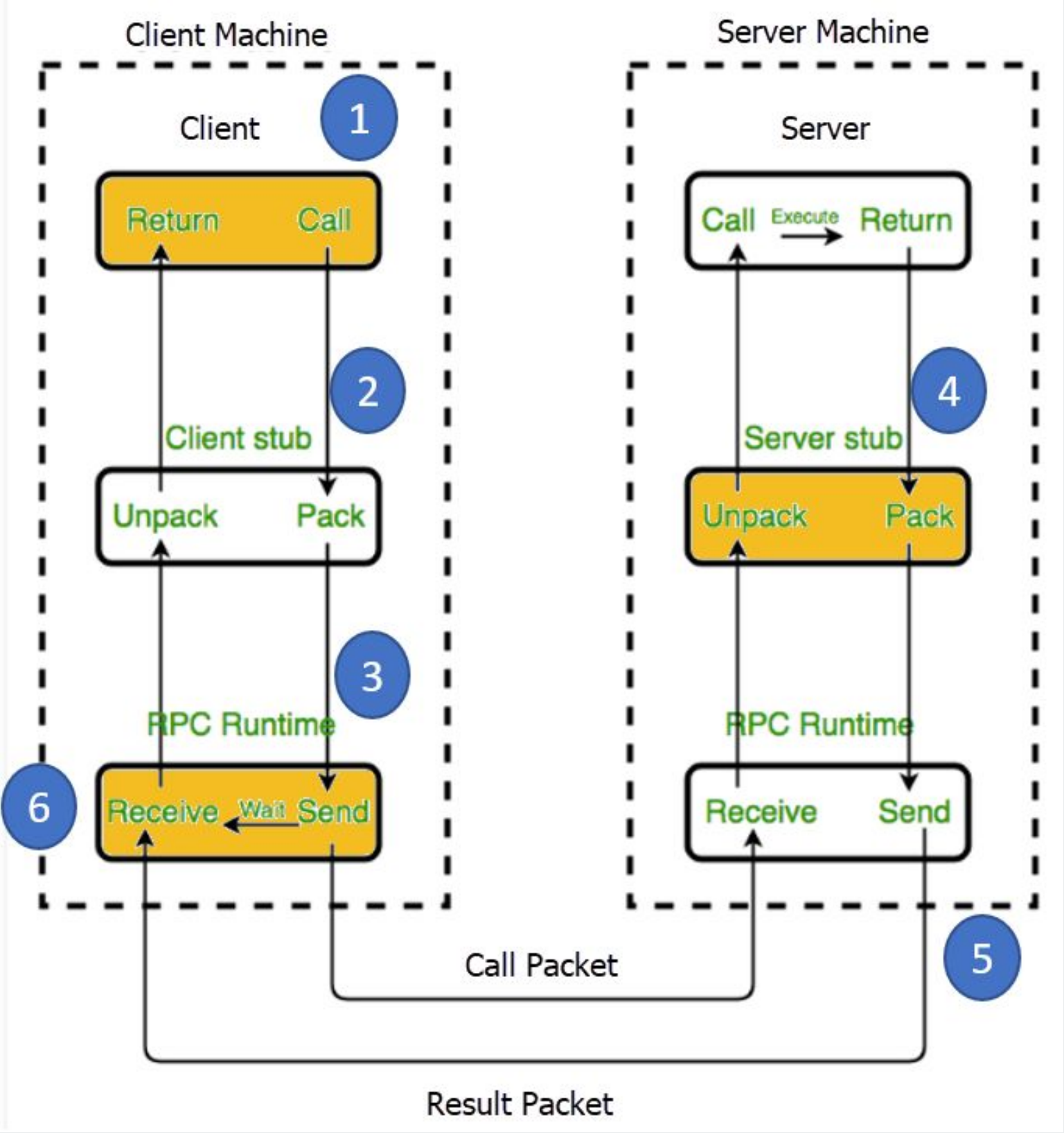
NFSv4 - авторизация

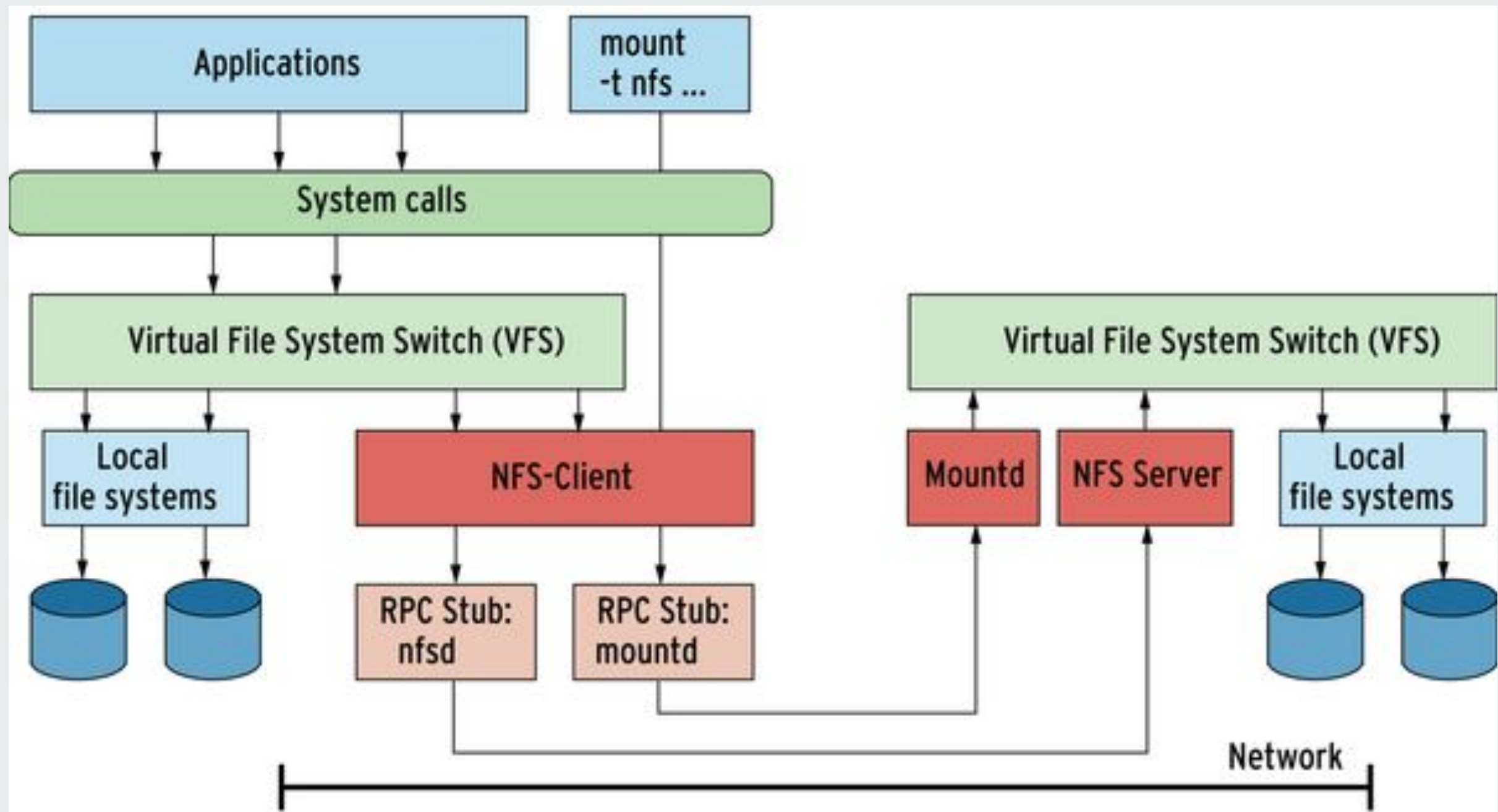
- **rpc.gssd** - обеспечивает методы аутентификации через GSS-API (Kerberos-аутентификация). Работает на клиенте и сервере.
- **rpc.svcgssd** - обеспечивает проверку подлинности клиента на стороне сервера

portmap - rpcbind

- клиент обращается к portmap (port 111) для получения порта сервиса RPC
- затем обращается на этот порт к конкретному сервису через функции заглушки (stub)
- *rpcinfo -p*







- **/etc/exports** - список директорий для экспорта
- **/var/lib/nfs/xtab** - список каталогов, примонтированных удаленными клиентами.
- **/var/lib/nfs/etab** - список каталогов, которые могут быть смонтированы удаленными системами с указанием всех параметров экспортированных каталогов.
- **/var/lib/nfs/rmtab** - rpc.mount регистрируют все успешные запросы на монтирование, при размонтировании записи удаляются
- **/proc/fs/nfsd** -
 - **exports** - список активных экспортированных иерархий и клиентов, которым их экспортировали, а также параметры.
 - **threads** - содержит число потоков (также можно изменять)
 - с помощью **filehandle** можно получить указатель на файл
- **/proc/net/rpc** - содержит "сырую" (raw) статистику, которую можно получить с помощью `nfsstat`, а также различные кеши.
- **/var/run/portmap_mapping** - информация о зарегистрированных в RPC сервисах

- `yum install nfs-utils -y`
- `/etc/exports`
 - `/another/exported/directory 192.168.0.3(rw,async)`
 - `/path/to/nfs *(ro,sync,wdelay,root_squash)`
- `exportfs /path/to/nfs`
- `exportfs -u /path/to/nfs`

- **sec=(krb5,krb5i,krbp)** - какой протокол защиты использовать
- **secure/insecure** - запросы с портов (<1024)
- **ro/rw** - read only, read write
- **root_squash/no_root_squash** - автоподмена владельца файла с root на анонимного пользователя
- **all_squash** - автоподмена на анонима для всех файлов
- **anonuid=UID** и **anongid=GID** - Явно задает UID/GID для анонимного пользователя.
- **map_static=/etc/file_maps_users** - Задает файл, в котором можно задать сопоставление удаленных UID/GID - локальным UID/GID.

uid 0-50 1002

gid 0-50 1002

- **sync/async** - sync (async) - указывает, что сервер должен отвечать на запросы только после записи на диск изменений, выполненных этими запросами. Опция async указывает серверу не ждать записи информации на диск, что повышает производительность, но понижает надежность, т.к. в случае обрыва соединения или отказа оборудования возможна потеря данных;

- **wdelay/no-wdelay** - задержка при записи (если sync)
- **subtree_check/no_subtree_check** - проверка на соответствие файла подкаталогу
- **insecure_locks** - позволять неавторизованные блокировки
- **hide/nohide** - видимость вложенных фс
- **crossmnt/nocrossmnt** - видимость вложенных фс (для родительской фс)
- **pnfs** - опция параллельного сервера
- **noaccess** – запрещает доступ к указанной директории.

- **nfsstat**
 - статистика NFS и RPC
- **showmount**
 - информация о смонтированных директориях
- **exportfs**
 - управление экспортированными каталогами
- **nfs4_setfacl**
- **nfs4_editfacl**

- **nosuid** - запрещает исполнять setuid программы из смонтированного каталога.
- **noexec** - запрещает исполнять программы из смонтированного каталога.
- **tcp/udp**
- **nodev** (no device - не устройство) - запрещает использовать в качестве устройств символьные и блочные специальные файлы.
- **lock (nolock)** - управление блокировкой
- **rsize=n** (read block size - размер блока чтения) - Количество байтов, читаемых за один раз с NFS-сервера. Стандартно - 4096.
- **wsizе=n** (write block size - размер блока записи) - Количество байтов, записываемых за один раз на NFS-сервер. Стандартно - 4096.
- **nfsvers=(3|4)**

- **ac (noac)** (*attribute cache* - кэширование атрибутов) - Разрешает кэширование атрибутов (по-умолчанию). Хотя опция **noac** замедляет работу сервера, она позволяет избежать устаревания атрибутов, когда несколько клиентов активно записывают информацию в общую иерархию.
- **actimeo=n** (*attribute cache timeout* - таймаут кэширования атрибутов)

- **noatime**: запрет обновления access time атрибута
- **nodiratime**: аналогично для директории
- **rsize=32768, wsize=32768**
-

- **fg (bg)** (*foreground/background*) - где производить попытки монтирования отказавшей NFS
- **hard (soft)** - выводит на консоль сообщение "server not responding" при достижении таймаута и продолжает попытки монтирования. При заданной опции **soft** - при таймауте сообщает вызвавшей операцию программе об ошибке ввода/вывода. (опцию **soft** советуют не использовать)
- **nointr (intr)** (*no interrupt* - не прерывать) - Не разрешает сигналам прерывать файловые операции в жестко смонтированной иерархии каталогов при достижении большого таймаута. **intr** - разрешает прерывание.
- рекомендуемая комбинация: **hard, intr**

- **retrans=n** (*retransmission value* - значение повторной передачи) - После n малых таймаутов NFS генерирует большой таймаут (по-умолчанию 3). Большой таймаут прекращает выполнение операций или выводит на консоль сообщение "server not responding", в зависимости от указания опции hard/soft.
- **retry=n** (*retry value* - значение повторно попытки) - Количество минут повторений службы NFS операций монтирования, прежде чем сдаться (по-умолчанию 10000).
- **timeo=n** (*timeout value* - значение таймаута) - Количество десятых долей секунды ожидания службой NFS до повторной передачи в случае RPC или малого таймаута (по-умолчанию 7). Это значение увеличивается при каждом таймауте до максимального значения 60 секунд или до наступления большого таймаута.
 - подбирать с помощью `ping -s <wsize>`

- `/etc/fstab`
 - возможная проблема: сеть грузится после монтирования
- вариант решения: `noauto,x-systemd.automount`
 - `sudo systemctl restart remote-fs.target`
 - фактическое монтирование происходит при первом обращении
- другой вариант: `systemd .mount unit`

[Unit]

Description = Mount NFS Share

Requires=network-online.service

After=network-online.service

[Mount]

What=yourserver.com:/srv/backups

Where=/mnt/backups

Type=nfs

Options=defaults

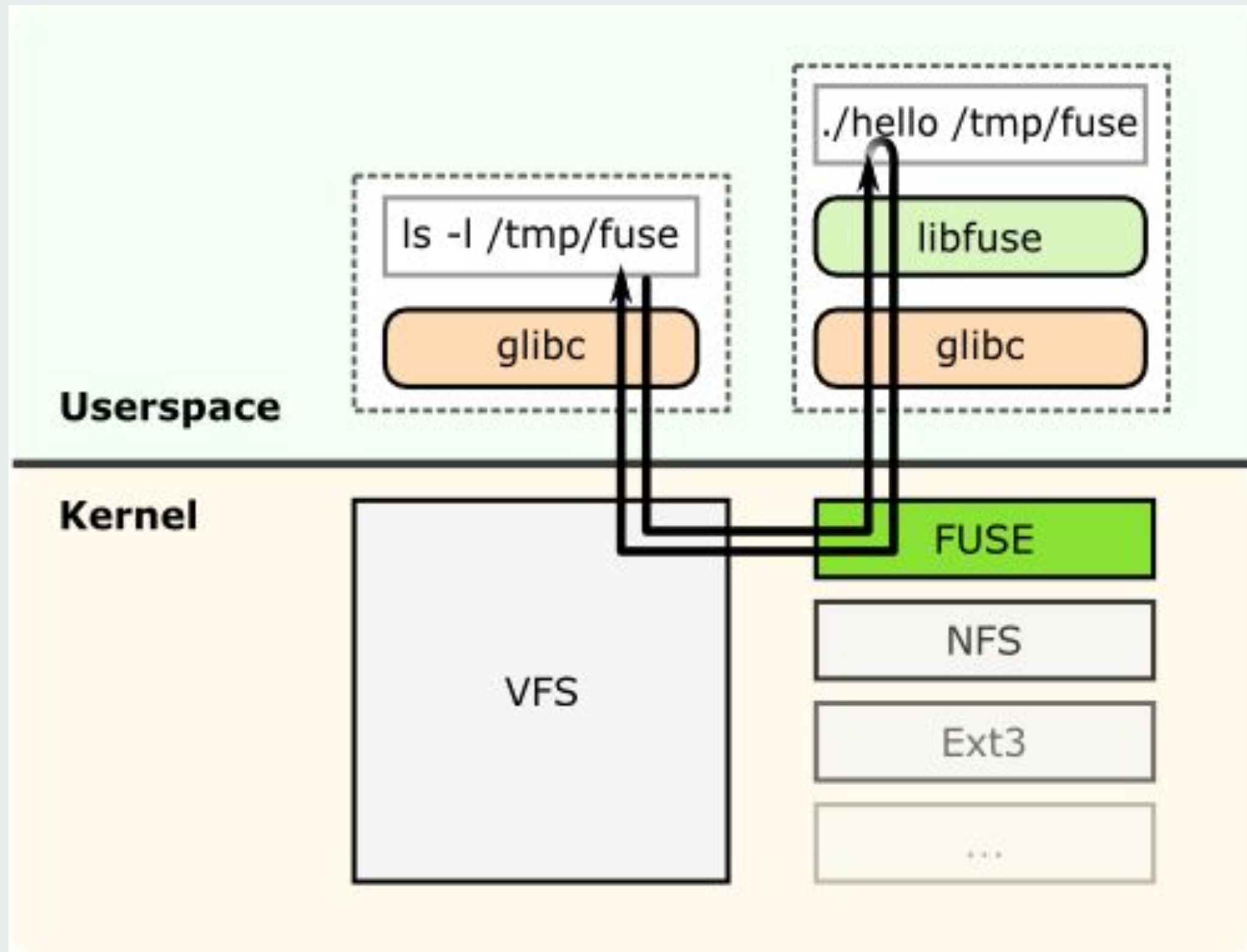
[Install]

WantedBy=multi-user.target

SSHFS

FUSE - модуль для ядер Linux/Unix. Запуск кода файловой системы в пользовательском пространстве, что в свою очередь позволяет монтировать эти ФС непривилегированным пользователям.

Используется, например, в - SSHFS, NTFS-3G, GlusterFS, ZFS.



- `sudo yum install sshfs`
- `sshfs your.server.name:/path /mountpoint`
- `fusermount -u /mountpoint`
- `/etc/fstab`
 - `sshfs#user@host:/ /mnt/host fuse defaults 0 0`

Ваши вопросы?

**Заполните, пожалуйста,
опрос в ЛК о занятии**

**Спасибо
за внимание!**

До встречи в Slack и на вебинаре

