

**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ**

**Факультет физико-математических и естественных наук**

**Кафедра прикладной информатики и теории вероятностей**

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ № 13**

**«Настройка NFS»**

*Дисциплина: Администрирование сетевых подсистем*

Студент: Карташова А.С.

Группа: НФИбд-03-18

**МОСКВА**

2020 г.

## Оглавление

Цель работы .....	2
Задачи.....	2
Ход работы .....	3
Настройка сервера NFSv4.....	3
Монтирование NFS на клиенте .....	7
Подключение каталогов к дереву NFS .....	9
Подключение каталогов для работы пользователей.....	11
Внесение изменений в настройки внутреннего окружения виртуальных машин .....	13
Заключение.....	15
Контрольные вопросы.....	15

## Цель работы

Приобретение навыков настройки сервера NFS для удалённого доступа к ресурсам

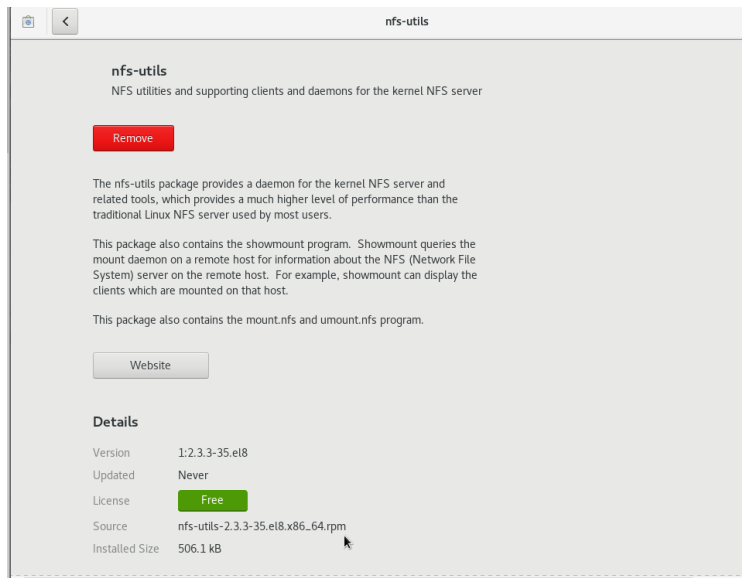
## Задачи

1. Установить и настроить сервер NFSv4
2. Подмонтировать удалённый ресурс на клиенте.
3. Подключить каталог с контентом веб-сервера к дереву NFS.
4. Подключить каталог для удалённой работы вашего пользователя к дереву NFS
5. Написать скрипт для Vagrant, фиксирующий действия по настройке NFS во внутреннем окружении виртуальной машины server. Соответствующим образом внести изменения в Vagrantfile.

## Ход работы

### Настройка сервера NFSv4

На сервере установите необходимое программное обеспечение nfs-utils, скачав необходимый пакет nfs-utils-2.3.3-35.el8.x86\_64



Проверим еще раз установлен ли пакет

```
[root@server.askartashova.net Downloads]# sudo rpm -qa | grep nfs-utils  
nfs-utils-2.3.3-35.el8.x86_64
```

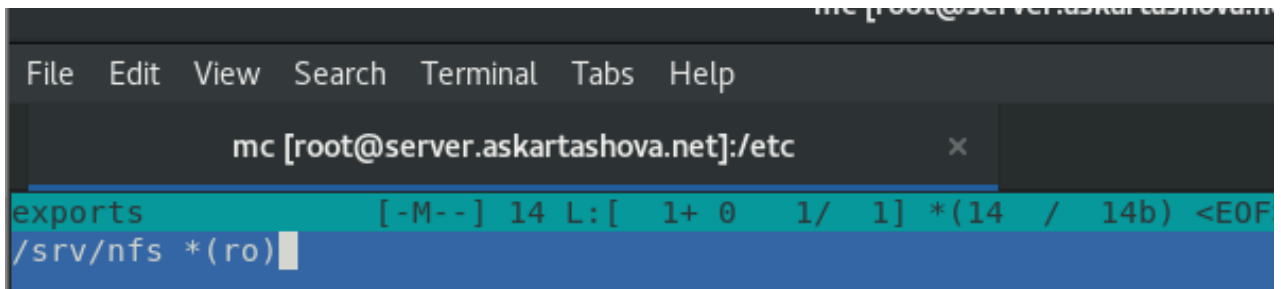
На сервере создадим каталог, который предполагается сделать доступным всем пользователям сети (корень дерева NFS):

**Команда:** `mkdir -p /srv/nfs`

```
[root@server.askartashova.net Downloads]# mkdir -p /srv/nfs
```

В файле `/etc/exports` пропишем подключаемый через NFS общий каталог с доступом только на чтение:

*/srv/nfs \*(ro)*



```
mc [root@server.askartashova.net]:/etc
exports [-M--] 14 L:[ 1+ 0 1/ 1] *(14 / 14b) <EOF
/srv/nfs *(ro)
```

Для общего каталога задайте контекст безопасности NFS:

**Команда:** *semanage fcontext -a -t nfs\_t "/srv/nfs(/.\*)?"*

```
[root@server.askartashova.net etc]# semanage fcontext -a -t nfs_t "/srv/nfs(/.*)?"
```

Применим изменённую настройку SELinux к файловой системе:

**Команда:** *restorecon -vR /srv/nfs*

```
[root@server.askartashova.net etc]# restorecon -vR /srv/nfs
Relabeled /srv/nfs from unconfined_u:object_r:var_t:s0 to unconfined_u:object_r:nfs_t:s0
[root@server.askartashova.net etc]#
```

Запустим сервер NFS:

**Команды:** *systemctl start nfs-server.service*

*systemctl enable nfs-server.service*

```
[root@server.askartashova.net etc]# systemctl start nfs-server.service
[root@server.askartashova.net etc]# systemctl enable nfs-server.service
Created symlink /etc/systemd/system/multi-user.target.wants/nfs-server.service → /usr/lib/systemd/system/nfs-server.service.
```

Настройте межсетевой экран для работы сервера NFS:

**Команды:** *firewall-cmd --add-service=nfs*

*firewall-cmd --add-service=nfs --permanent*

*firewall-cmd --reload*

```
[root@server.askartashova.net etc]# firewall-cmd --add-service=nfs
success
[root@server.askartashova.net etc]# firewall-cmd --add-service=nfs --permanent
success
[root@server.askartashova.net etc]# firewall-cmd --reload
success
```

На клиенте установим необходимое для работы NFS программное обеспечение:

**Команда:** `dnf -y install nfs-utils`

```
[root@client.askartashova.net ~]# dnf -y install nfs-utils
Last metadata expiration check: 3:15:32 ago on Sat 26 Dec 2020 03:07:27 PM MSK.
Package nfs-utils-1:2.3.3-31.el8.x86_64 is already installed.
Dependencies resolved.
=====
Package                Architecture  Version              Repository           Size
=====
Upgrading:
nfs-utils              x86_64       1:2.3.3-35.el8      BaseOS               494 k
Transaction Summary
=====
Upgrade 1 Package

Total download size: 494 k
Downloading Packages:
nfs-utils-2.3.3-35.el8.x86_64.rpm      1.0 MB/s | 494 kB      00:00
-----
Total                                  204 kB/s | 494 kB      00:02
Running transaction check
```

На клиенте попробуем посмотреть имеющиеся подмонтированные удалённые ресурсы

**Команда:** `showmount -e server.user.net`

```
[root@client.askartashova.net ~]# showmount -e server.askartashova.net
mnt_create: RPC: Unable to receive
[root@client.askartashova.net ~]#
```

Команда `showmount` выводит список всех клиентов, которые выполняли удаленное монтирование файловой системы на `host`-машине. Мы видим, что клиент не смог получить список экспортируемых файловых систем

Попробуем на сервере остановить сервис межсетевого экрана:

**Команда:** `systemctl stop firewalld.service`

```
[root@server.askartashova.net etc]# systemctl stop firewalld.service
[root@server.askartashova.net etc]#
```

Затем на клиенте вновь попробуем подключиться к удалённо смонтированному ресурсу:

**Команда:** `showmount -e server.user.net`

```
[root@client.askartashova.net ~]# showmount -e server.askartashova.net
Export list for server.askartashova.net:
/srv/nfs *
```

Теперь мы видим каталог, подключенный нами через NFS

На сервере запустим сервис межсетевого экрана

**Команда:** `systemctl start firewalld`

```
[root@server.askartashova.net etc]# systemctl start firewalld
```

На сервере посмотрим, какие службы задействованы при удалённом монтировании:

**Команда:** `lsof | grep TCP`

`lsof | grep UDP`

```
[root@server.askartashova.net etc]# lsof | grep TCP
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1001/gvfs
Output information may be incomplete.
systemd      1      root    211u    IPv4        17636      0t0      TCP *:sunrpc (LISTEN)
systemd      1      root    213u    IPv6        17638      0t0      TCP *:sunrpc (LISTEN)
rpcbind     675      rpc      4u     IPv4        17636      0t0      TCP *:sunrpc (LISTEN)
rpcbind     675      rpc      6u     IPv6        17638      0t0      TCP *:sunrpc (LISTEN)
cupsd       884      root     9u     IPv6       25388      0t0      TCP localhost:ipp (LISTEN)
cupsd       884      root    10u    IPv4       25389      0t0      TCP localhost:ipp (LISTEN)
sshd        888      root     5u     IPv4       24681      0t0      TCP *:down (LISTEN)
sshd        888      root     7u     IPv6       24683      0t0      TCP *:down (LISTEN)
```

```
rpc.statd 44315      rpcuser 10u     IPv4       185499     0t0      TCP *:39259 (LISTEN)
rpc.statd 44315      rpcuser 12u     IPv6       185505     0t0      TCP *:37749 (LISTEN)
rpc.mount 44320      root     8u     IPv4       186264     0t0      TCP *:mountd (LISTEN)
rpc.mount 44320      root    10u     IPv6       186270     0t0      TCP *:mountd (LISTEN)
```

```
[root@server.askartashova.net etc]# lsof | grep UDP
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1001/gvfs
Output information may be incomplete.
systemd      1      root    212u    IPv4        17637      0t0      UDP *:sunrpc
systemd      1      root    214u    IPv6        17639      0t0      UDP *:sunrpc
rpcbind     675      rpc      5u     IPv4        17637      0t0      UDP *:sunrpc
rpcbind     675      rpc      7u     IPv6        17639      0t0      UDP *:sunrpc
```

```
rpc.statd 44315      rpcuser 5u     IPv4
rpc.statd 44315      rpcuser 9u     IPv4
rpc.statd 44315      rpcuser 11u    IPv6
rpc.mount 44320      root    7u     IPv4
rpc.mount 44320      root    9u     IPv6
```

Добавим службы rpc-bind и mountd в настройки межсетевого экрана на сервере:

**Команда:** `firewall-cmd --get-services`

`firewall-cmd --add-service=mountd --add-service=rpc-bind`

`firewall-cmd --add-service=mountd --add-service=rpc-bind --permanent`

`firewall-cmd --reload`

```

[re-mount 1320 1000 50 1170 100207 000 000 1000000]
[root@server.askartashova.net etc]# firewall-cmd --get-services
RH-Satellite-6 amanda-client amanda-k5-client amqp amqps apcupsd audit bacula bacula-client bb bgp bitcoin bitco
in-rpc bitcoin-testnet bitcoin-testnet-rpc bittorrent-lsd ceph ceph-mon cfengine cockpit condor-collector ctdb d
hcp dhcpv6 dhcpv6-client distcc dns dns-over-tls docker-registry docker-swarm dropbox-lansync elasticsearch etcd
-client etcd-server finger freeipa-4 freeipa-ldap freeipa-ldaps freeipa-replication freeipa-trust ftp ganglia-cl
ient ganglia-master git grafana gre high-availability http https imap imaps ipp ipp-client ipsec irc ircs iscsi-
target isns jenkins kadmin kdeconnect kerberos kibana klogin kpasswd kprop kshell kube-apiserver ldap ldaps libv
irt libvirt-tls lightning-network llmnr managesieve matrix mdns memcache minidlna mongodb mosh mountd mqtt mqtt-
tls ms-wbt mssql murmur mysql nfs nfs3 nmea-0183 nrpe ntp nut openvpn ovirt-imageio ovirt-storageconsole ovirt-v
mconsole plex pmcd pmproxy pmwebapi pmwebapis pop3 pop3s postgresql privoxy prometheus proxy-dhcp ptp pulseaudio
puppetmaster quassel radius rdp redis redis-sentinel rpc-bind rsh rsyncd rtsp salt-master samba samba-client sa
mba-dc sane sip sips slp smtp smtp-submission smtps snmp snmptrap spideroak-lansync spotify-sync squid ssdp ssh
ssh-custom steam-streaming svdrp svn syncthing syncthing-gui synergy syslog syslog-tls telnet tentacle tftp tftp
-client tile38 tinc tor-socks transmission-client upnp-client vdsm vnc-server wbem-http wbem-https wsman wsmans
xdmcp xmpp-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-server
[root@server.askartashova.net etc]# firewall-cmd --add-service=mountd --add-service=rpc-bind
success
[root@server.askartashova.net etc]# firewall-cmd --add-service=mountd --add-service=rpc-bind
Warning: ALREADY_ENABLED: 'mountd' already in 'public'
Warning: ALREADY_ENABLED: 'rpc-bind' already in 'public'
success
[root@server.askartashova.net etc]# firewall-cmd --add-service=mountd --add-service=rpc-bind --permanent
success
[root@server.askartashova.net etc]# firewall-cmd --reload
success

```

На клиенте проверим подключение удалённого ресурса:

**Команда:** *showmount -e server.user.net*

Удаленный ресурс успешно подключен

```

[root@client.askartashova.net ~]# showmount -e server.askartashova.net
Export list for server.askartashova.net:
/srv/nfs *
[root@client.askartashova.net ~]#

```

## Монтирование NFS на клиенте

На клиенте создайте каталог, в который будет монтироваться удалённый ресурс, и подмонтируйте дерево NFS:

**Команды:** *mkdir -p /mnt/nfs*

*mount server.askartashova.net:/srv/nfs /mnt/nfs*

```

/srv/nfs *
[root@client.askartashova.net ~]# mkdir -p /mnt/nfs
[root@client.askartashova.net ~]# mount server.user.net:/srv/nfs /mnt/nfs
mount.nfs: Failed to resolve server server.user.net: Name or service not known
[root@client.askartashova.net ~]# mount server.askartashova.net:/srv/nfs /mnt/nfs

```

Проверьте, что общий ресурс NFS подключён правильно:

**Команда:** *mount*

```
[root@client.askartashova.net ~]# mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=474400k,nr_inodes=118600,mode=755)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,mode=755)
tmpfs on /sys/fs/cgroup type tmpfs (ro,nosuid,nodev,noexec,seclabel,mode=755)
cgroup on /sys/fs/cgroup/systemd type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,xattr,release_agent=/usr/lib/systemd/systemd-cgroups-agent,name=systemd)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime,seclabel)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
cgroup on /sys/fs/cgroup/cpu,cpuacct type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,cpu,cpuacct)
cgroup on /sys/fs/cgroup/net_cls,net_prio type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,net_cls,net_prio)
cgroup on /sys/fs/cgroup/freezer type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,freezer)
cgroup on /sys/fs/cgroup/rdma type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,rdma)
cgroup on /sys/fs/cgroup/blkio type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,blkio)
cgroup on /sys/fs/cgroup/pids type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,pids)
cgroup on /sys/fs/cgroup/hugetlb type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,hugetlb)
cgroup on /sys/fs/cgroup/perf_event type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,perf_event)
cgroup on /sys/fs/cgroup/memory type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,memory)
cgroup on /sys/fs/cgroup/devices type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,devices)
cgroup on /sys/fs/cgroup/cpuset type cgroup (rw,nosuid,nodev,noexec,relatime,seclabel,cpuset)
configfs on /sys/kernel/config type configfs (rw,relatime)
/dev/sdal on / type xfs (rw,relatime,seclabel,attr2,inode64,quota)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,relatime)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=40,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe_ino=17704)
debugfs on /sys/kernel/debug type debugfs (rw,relatime,seclabel)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,seclabel,pagesize=2M)
mqueue on /dev/mqueue type mqueue (rw,relatime,seclabel)
fusectl on /sys/fs/fuse/connections type fusectl (rw,relatime)
/vagrant on /vagrant type vboxsf (rw,nodev,relatime,ioccharset=utf8,uid=1000,gid=1000)
sunrpc on /var/lib/nfs/rpc pipefs type rpc_pipefs (rw,relatime)
vagrant on /vagrant type vboxsf (rw,nodev,relatime,ioccharset=utf8,uid=1000,gid=1000)
tmpfs on /run/user/42 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=100284k,mode=700,uid=42,gid=42)
tmpfs on /run/user/1001 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=100284k,mode=700,uid=1001,gid=1001)
gvfsd-fuse on /run/user/1001/gvfs type fuse.gvfsd-fuse (rw,nosuid,nodev,relatime,user_id=1001,group_id=1001)
server.askartashova.net:/srv/nfs on /mnt/nfs type nfs4 (rw,relatime,vers=4.2,rsize=131072,wsiz=131072,namlen=255,hard,proto=tcp,timeo=600,retra
ns=2,sec=sys,clientaddr=192.168.1.30,local_lock=none,addr=192.168.1.1)
```

Мы видим список всех точек монтирования. в первой строке сообщается, что в качестве корневой ФС выступает файловая система sysfs с параметрами монтирования: доступ на чтение и запись (rw). sysfs, udev, devpts — это стандартные виртуальные ФС

На клиенте в конце файла /etc/fstab добавьте следующую запись:

**Команда:** `server.user.net:/srv/nfs /mnt/nfs nfs _netdev 0 0`

```
GNU nano 2.9.8 /etc/fstab

#
# /etc/fstab
# Created by anaconda on Sun Nov 15 23:04:32 2020
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=3a0f6e9b-14ee-45ec-a7c0-710c860358f6 / xfs defaults 0 0
/swapfile none swap defaults 0 0
#VAGRANT-BEGIN
# The contents below are automatically generated by Vagrant. Do not modify.
/vagrant /vagrant vboxsf uid=1000,gid=1000,nofail 0 0
#VAGRANT-END
server.askartashova.net:/srv/nfs /mnt/nfs nfs _netdev 0 0
```

Синтаксис строки fstab:

устройство точка\_монтирования файловая\_система опции резерв{0,1}  
 проверка{0,1,2}



На клиенте проверьте наличие автоматического монтирования удалённых ресурсов при запуске операционной системы:

**Команда:** `systemctl status remote-fs.target`

```
[root@client.askartashova.net etc]# systemctl status remote-fs.target
● remote-fs.target - Remote File Systems
   Loaded: loaded (/usr/lib/systemd/system/remote-fs.target; enabled; vendor preset: enabled)
   Active: active since Sat 2020-12-26 12:57:52 MSK; 6h ago
     Docs: man:systemd.special(7)

Dec 26 12:57:52 client.askartashova.net systemd[1]: Reached target Remote File Systems.
[root@client.askartashova.net etc]#
```

Перезапустим клиента и убедитесь, что удалённый ресурс подключается автоматически.

```
[root@client.askartashova.net ~]# systemctl status remote-fs.target
● remote-fs.target - Remote File Systems
   Loaded: loaded (/usr/lib/systemd/system/remote-fs.target; enabled; vendor preset: enabl
   Active: active since Sat 2020-12-26 19:21:28 MSK; 3s ago
     Docs: man:systemd.special(7)

Dec 26 19:21:28 client.askartashova.net systemd[1]: Reached target Remote File Systems.
```

```
[root@client.askartashova.net ~]# showmount -e server.askartashova.net
Export list for server.askartashova.net:
/srv/nfs *
```

### Подключение каталогов к дереву NFS

На сервере создайте общий каталог, в который затем будет подмонтирован каталог с контентом веб-сервера:

**Команда:** `mkdir -p /srv/nfs/www`

Подмонтируйте каталог web-сервера:

**Команда:** `mount -o bind /var/www/ /srv/nfs/www/`

```
[root@server.askartashova.net etc]# mkdir -p /srv/nfs/www
[root@server.askartashova.net etc]# mount -o bind /var/www/ /srv/nfs/www/
```

На сервере проверьте, что отображается в каталоге /srv/nfs.

```
[root@server.askartashova.net etc]# ls /srv/nfs
ww  www
```

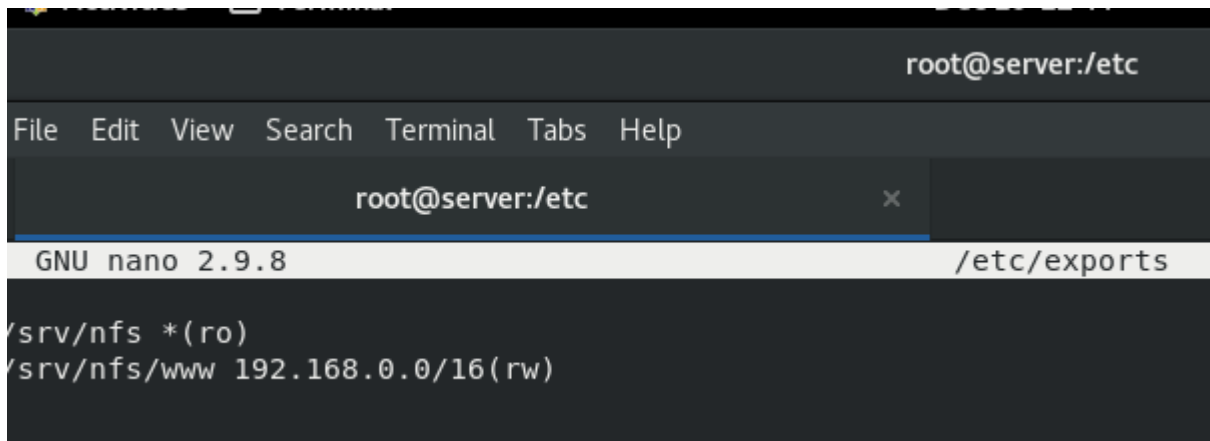
На клиенте посмотрим, что отображается в каталоге /mnt/nfs.

```
[root@client.askartashova.net ~]# ls /mnt/nfs
ww  www
[root@client.askartashova.net ~]#
```

Видим, что на сервере и клиенте в монтированных каталогах лежат одинаковые файлы

На сервере в файле `/etc/exports` добавьте экспорт каталога веб-сервера с удалённого ресурса:

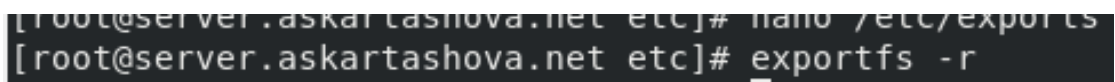
**Команда:** `/srv/nfs/www 192.168.0.0/16(rw)`



```
root@server:/etc
File Edit View Search Terminal Tabs Help
root@server:/etc
GNU nano 2.9.8 /etc/exports
/srv/nfs *(ro)
/srv/nfs/www 192.168.0.0/16(rw)
```

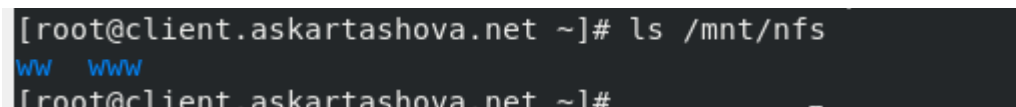
Экспортируйте все каталоги, упомянутые в файле `/etc/exports`:

*exportfs -r*



```
[root@server.askartashova.net etc]# nano /etc/exports
[root@server.askartashova.net etc]# exportfs -r
```

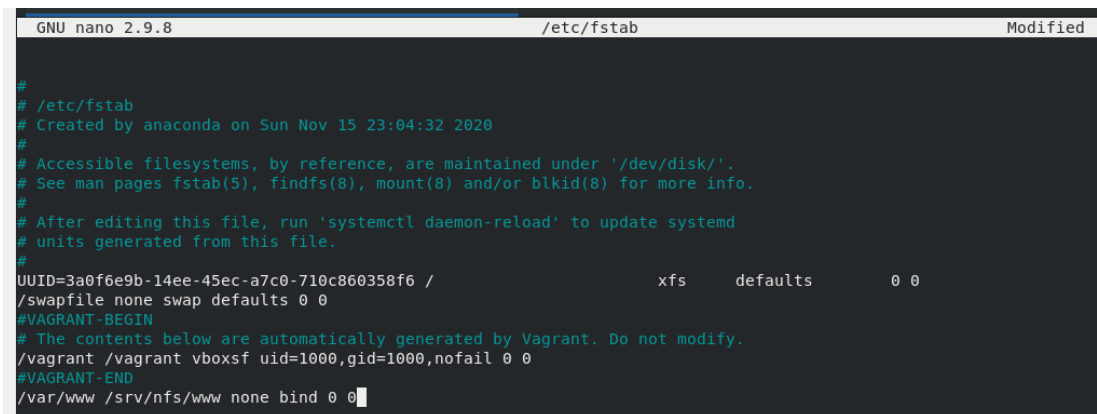
Проверьте на клиенте каталог `/mnt/nfs`.



```
[root@client.askartashova.net ~]# ls /mnt/nfs
www
[root@client.askartashova.net ~]#
```

На сервере в конце файла `/etc/fstab` добавьте следующую запись:

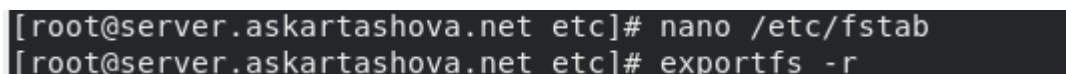
*/var/www/srv/nfs/www none bind 0 0*



```
GNU nano 2.9.8 /etc/fstab Modified
#
# /etc/fstab
# Created by anaconda on Sun Nov 15 23:04:32 2020
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=3a0f6e9b-14ee-45ec-a7c0-710c860358f6 / xfs defaults 0 0
/swapfile none swap defaults 0 0
#VAGRANT-BEGIN
# The contents below are automatically generated by Vagrant. Do not modify.
/vagrant /vagrant vboxsf uid=1000,gid=1000,nofail 0 0
#VAGRANT-END
/var/www/srv/nfs/www none bind 0 0
```

Повторно экспортируйте каталоги, указанные в файле `/etc/exports`:

*exportfs -r*



```
[root@server.askartashova.net etc]# nano /etc/fstab
[root@server.askartashova.net etc]# exportfs -r
```

На клиенте проверьте каталог `/mnt/nfs`.

```
[root@client.askartashova.net ~]# ls /mnt/nfs  
ww  www
```

### ***Подключение каталогов для работы пользователей***

На сервере под пользователем user в его домашнем каталоге создайте каталог common с полными правами доступа только для этого пользователя, а в нём файл askartashova@server.txt :

**Команды:** *mkdir -p -m 700 ~/common*

*cd ~/common*

*touch askartashova @server.txt*

```
[askartashova@server.askartashova.net ~]$ mkdir -p -m 700 ~/common  
[askartashova@server.askartashova.net ~]$ cd ~/common  
[askartashova@server.askartashova.net common]$ touch askartashova@server.txt
```

На сервере создайте общий каталог для работы пользователя user по сети :

**Команда:** *mkdir -p /srv/nfs/home/ askartashova*

```
[askartashova@server.askartashova.net common]$ mkdir -p /srv/nfs/home/askartashova  
mkdir: cannot create directory '/srv/nfs/home': Permission denied  
[askartashova@server.askartashova.net common]$ sudo -i  
[sudo] password for askartashova:  
[root@server.askartashova.net ~]# ^C  
[root@server.askartashova.net ~]# mkdir -p /srv/nfs/home/askartashova
```

Подмонтируйте каталог common пользователя user в NFS:

**Команда:** *mount -o bind /home/ askartashova /common/srv/nfs/home/user*

```
[root@server.askartashova.net ~]# mount -o bind /home/askartashova/common /srv/nfs/home/askartashova  
  
[root@server.askartashova.net askartashova]# ls -l /srv/nfs/home/askartashova  
total 0  
-rw-rw-r--. 1 askartashova askartashova 0 Dec 26 22:23 askartashova@server.txt
```

На этот каталог установлены права на чтение и запись, но без выполнения (rw-).

Подключим каталог пользователя в файле /etc/exports, прописав в нём

*/srv/nfs/home/ askartashova 192.168.0.0/16(rw)*

```
root@server:/srv/nfs/home/askartashova
File Edit View Search Terminal Tabs Help
root@server:/srv/nfs/home/askartashova x root@server:~
GNU nano 2.9.8 /etc/exports

/srv/nfs *(ro)
/srv/nfs/www 192.168.0.0/16(rw)
/srv/nfs/home/askartashova 192.168.0.0/16(rw)
```

Внесем изменения в файл /etc/fstab :

*/home/askartashova/common/srv/nfs/home/user none bind 0 0*

```
root@server:/srv/nfs/home/askartashova
File Edit View Search Terminal Tabs Help
root@server:/srv/nfs/home/askartashova x root@server:~ x
GNU nano 2.9.8 /etc/fstab Modified

#
# /etc/fstab
# Created by anaconda on Sun Nov 15 23:04:32 2020
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=3a0f6e9b-14ee-45ec-a7c0-710c860358f6 / xfs defaults 0 0
/swapfile none swap defaults 0 0
#VAGRANT-BEGIN
# The contents below are automatically generated by Vagrant. Do not modify.
/vagrant /vagrant vboxsf uid=1000,gid=1000,nofail 0 0
#VAGRANT-END
/var/www /srv/nfs/www none bind 0 0
/home/askartashova/common /srv/nfs/home/askartashova none bind 0 0
```

Повторно экспортируем каталоги:

*exportfs -r*

```
[root@server.askartashova.net askartashova]# exportfs -r
[root@server.askartashova.net askartashova]#
```

На клиенте проверьте каталог /mnt/nfs.

```
[root@client.askartashova.net ~]# ls /mnt/nfs
home ww www
```

На клиенте под пользователем user перейдем в каталог

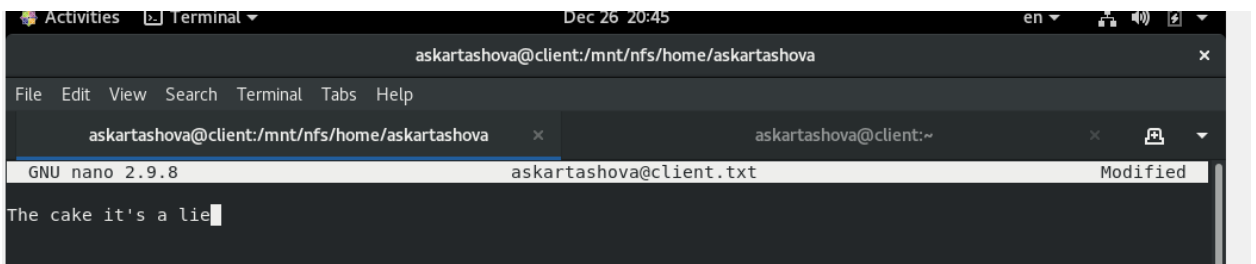
*/mnt/nfs/home/askartashova*

и создадим в нём файл askartashova @client.txt и внесем в него какие-либо изменения:

**Команда:** `cd /mnt/nfs/home/askartashova`

`touch askartashova@client.txt`

```
[askartashova@client.askartashova.net ~]$ cd /mnt/nfs/home/askartashova
[askartashova@client.askartashova.net askartashova]$ touch askartashova@client.txt
```



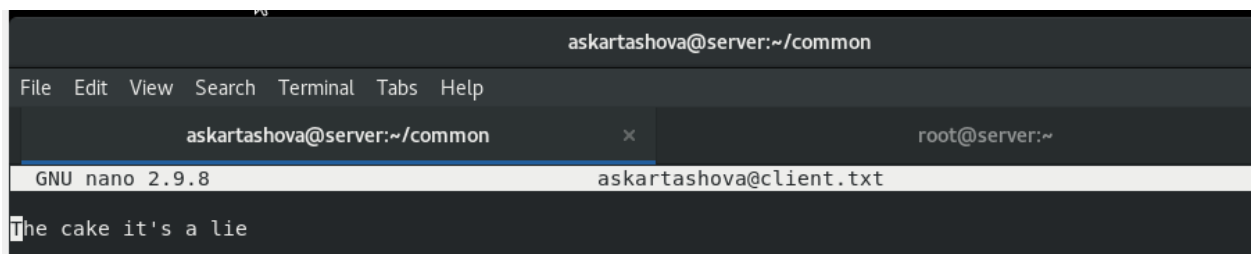
Попробуйте это проделать под пользователем root.

```
[root@client.askartashova.net ~]#
[root@client.askartashova.net ~]# cd /mnt/nfs/home/askartashova
-bash: cd: /mnt/nfs/home/askartashova: Permission denied
[root@client.askartashova.net ~]#
```

Под пользователем root перейти в каталог не удалось

На сервере посмотрим, появились ли изменения каталоге пользователя */home/askartashova/common*.

```
logout
[askartashova@server.askartashova.net common]$ cd /home/askartashova/common
[askartashova@server.askartashova.net common]$ ls
askartashova@client.txt askartashova@server.txt
[askartashova@server.askartashova.net common]$ nano askartashova@client.txt
[askartashova@server.askartashova.net common]$
```



Мы видим файл, созданный на сервере.

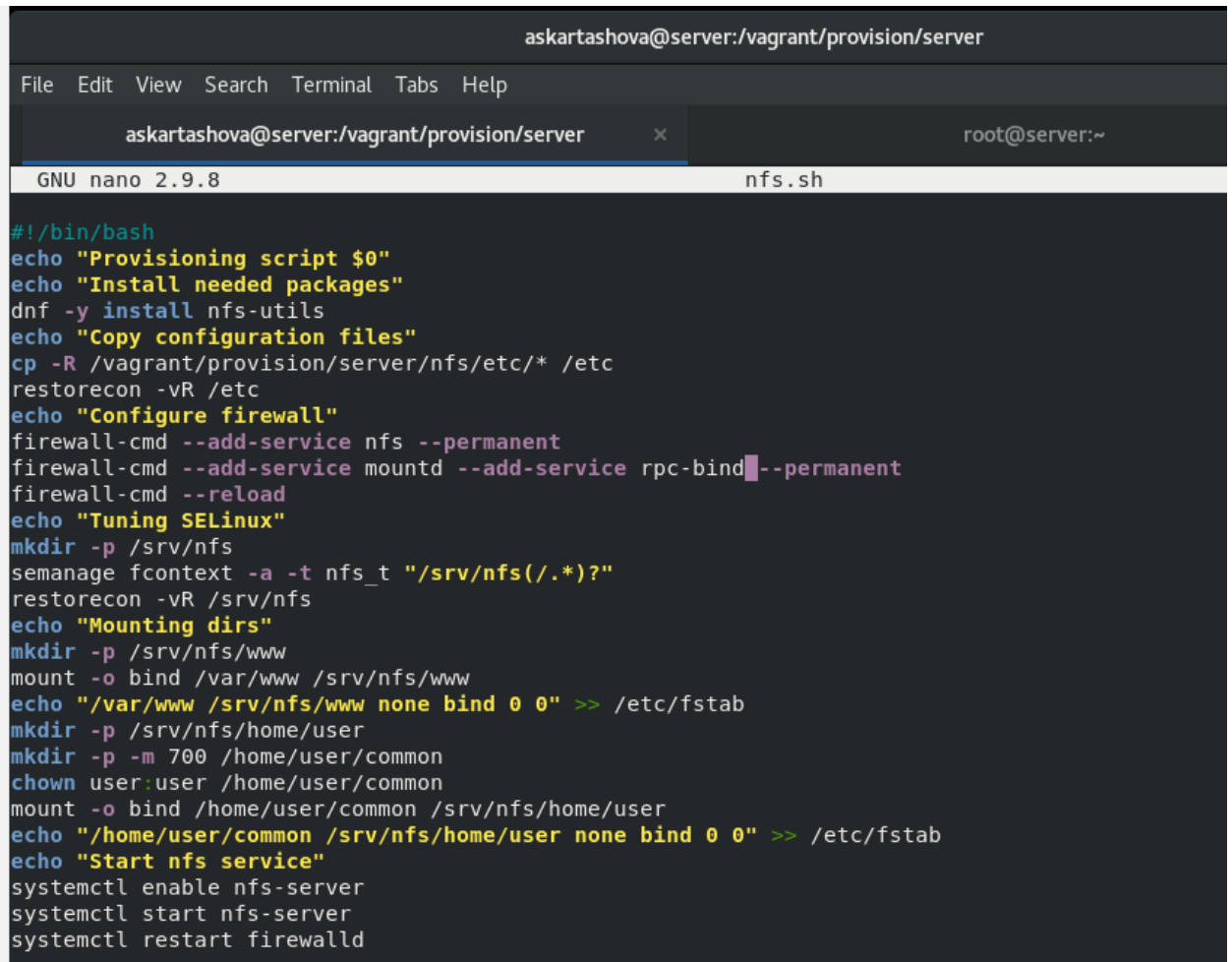
## ***Внесение изменений в настройки внутреннего окружения виртуальных машин***

На виртуальной машине server перейдем в каталог для внесения изменений в настройки внутреннего окружения */vagrant/provision/server/*, создайте в нём каталог *fs*, в который поместим в соответствующие подкаталоги конфигурационные файлы:

```
[askartashova@server.askartashova.net common]$ cd /vagrant/provision/server
[askartashova@server.askartashova.net server]$ mkdir -p /vagrant/provision/server/nfs/etc
[askartashova@server.askartashova.net server]$ cp -R /etc/exports /vagrant/provision/server/nfs/etc/
```

В каталоге `/vagrant/provision/server` создайте исполняемый файл `nfs.sh`.  
Открыв его на редактирование, пропишем в нём следующий скрипт,  
повторяющий наши действия настройки сервера

```
[askartashova@server.askartashova.net server]$ cd /vagrant/provision/serve
[askartashova@server.askartashova.net server]$ touch nfs.sh
[askartashova@server.askartashova.net server]$ chmod +x nfs.sh
[askartashova@server.askartashova.net server]$ nano nfs.sh
```



The screenshot shows a terminal window titled `askartashova@server:/vagrant/provision/server`. The window has a menu bar with `File`, `Edit`, `View`, `Search`, `Terminal`, `Tabs`, and `Help`. Below the menu bar, there are two tabs: `askartashova@server:/vagrant/provision/server` and `root@server:~`. The active tab is `askartashova@server:/vagrant/provision/server`. The terminal shows the `GNU nano 2.9.8` editor editing the `nfs.sh` file. The script content is as follows:

```
#!/bin/bash
echo "Provisioning script $0"
echo "Install needed packages"
dnf -y install nfs-utils
echo "Copy configuration files"
cp -R /vagrant/provision/server/nfs/etc/* /etc
restorecon -vR /etc
echo "Configure firewall"
firewall-cmd --add-service nfs --permanent
firewall-cmd --add-service mountd --add-service rpc-bind --permanent
firewall-cmd --reload
echo "Tuning SELinux"
mkdir -p /srv/nfs
semanage fcontext -a -t nfs_t "/srv/nfs(/.*)?"
restorecon -vR /srv/nfs
echo "Mounting dirs"
mkdir -p /srv/nfs/www
mount -o bind /var/www /srv/nfs/www
echo "/var/www /srv/nfs/www none bind 0 0" >> /etc/fstab
mkdir -p /srv/nfs/home/user
mkdir -p -m 700 /home/user/common
chown user:user /home/user/common
mount -o bind /home/user/common /srv/nfs/home/user
echo "/home/user/common /srv/nfs/home/user none bind 0 0" >> /etc/fstab
echo "Start nfs service"
systemctl enable nfs-server
systemctl start nfs-server
systemctl restart firewalld
```

На виртуальной машине `client` перейдем в каталог для внесения изменений в  
настройки внутреннего окружения `/vagrant/provision/client/`.

В каталоге `/vagrant/provision/client` создайте исполняемый файл `nfs.sh`.  
Открыв его на редактирование, пропишем в нём следующий скрипт,  
повторяющий наши действия по подмонтированию сервера.

```
[root@client.askartashova.net ~]# cd /vagrant/provision/client
[root@client.askartashova.net client]# cd /vagrant/provision/client
[root@client.askartashova.net client]# touch nfs.sh
[root@client.askartashova.net client]# chmod +x nfs.sh
[root@client.askartashova.net client]# nano nfs.sh
```

```
GNU nano 2.9.8                                nfs.sh

#!/bin/bash
echo "Provisioning script $0"
echo "Install needed packages"
dnf -y install nfs-utils
echo "Mounting dirs"
mkdir -p /mnt/nfs
mount server.user.net:/srv/nfs /mnt/nfs
echo "server.user.net:/srv/nfs /mnt/nfs nfs _netdev 0 0" >> /etc/fstab
restorecon -vR /etc
```

Для отработки созданных скриптов во время загрузки виртуальных машин server и client в конфигурационном файле Vagrantfile необходимо добавить в соответствующих разделах конфигураций для сервера и клиента:

```
server.vm.provision "server nfs",
  type: "shell",
  preserve_order: true,
  path: "provision/server/nfs.sh"
client.vm.provision "client nfs",
  type: "shell",
  preserve_order: true,
  path: "provision/client/nfs.sh"
```

```
server.vm.provision "server nfs",
  type: "shell",
  preserve_order: true,
  path: "provision/server/nfs.sh"
```

```
client.vm.provision "client nfs",
  type: "shell",
  preserve_order: true,
  path: "provision/client/nfs.sh"
```

## Заключение

Мы приобрели навыки по установке и настройке сервера NFS для удалённого доступа к ресурсам

## Контрольные вопросы

1. Как называется файл конфигурации, содержащий общие ресурсы NFS?

Файлы конфигурации для NFS-сервера:

`/etc/nfs.conf` — основной файл конфигурации для демонов и инструментов NFS.

`/etc/nfsmount.conf` — файл конфигурации монтирования NFS.

2. Какие порты должны быть открыты в брандмауэре, чтобы обеспечить полный доступ к серверу NFS?

Порты `tcp/udp: 111,2049`

3. Какую опцию следует использовать в `/etc/fstab`, чтобы убедиться, что общие ресурсы NFS могут быть установлены автоматически при перезагрузке?

Опцию `auto` - автоматически монтировать при загрузке, действие по умолчанию;