

# Лабораторная работа №13 (Настройка NFS)

---

Шубина София Антоновна

6 ноября 2025

Российский университет дружбы народов

Приобрести навыки настройки сервера NFS для удалённого доступа к ресурсам.

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 скрипты для Vagrant, фиксирующие действия по установке и настройке сервера NFSv4 во внутреннем окружении виртуальных машин `server` и `client`. Соответствующим образом внести изменения в `Vagrantfile`.

# Выполнение лабораторной работы

---

## Настройка сервера NFSv4

---

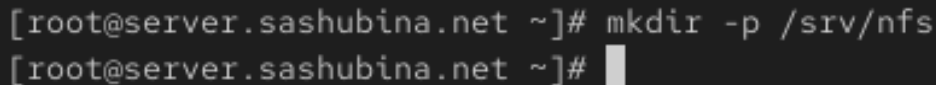
# На сервере установим необходимое программное обеспечение: `dnf -y install nfs-utils`

```
[root@server.sashubina.net ~]# dnf -y install nfs-utils
Extra Packages for Enterprise Linux 10 - x86_64                26 kB/s | 40 kB    00:01
Extra Packages for Enterprise Linux 10 - x86_64                156 kB/s | 4.8 MB    00:31
Rocky Linux 10 - BaseOS                                         10 kB/s | 4.3 kB    00:00
Rocky Linux 10 - AppStream                                       17 kB/s | 4.3 kB    00:00
Rocky Linux 10 - CRB                                             16 kB/s | 4.3 kB    00:00
Rocky Linux 10 - Extras                                          12 kB/s | 3.1 kB    00:00
Dependencies resolved.
=====
Package                Architecture      Version              Repository           Size
=====
Installing:
nfs-utils                x86_64            1:2.8.2-3.el10      baseos               473 k
Installing dependencies:
gssproxy                  x86_64            0.9.2-10.el10       baseos               111 k
libev                     x86_64            4.33-14.el10         baseos                52 k
libnfsidmap               x86_64            1:2.8.2-3.el10       baseos                61 k
libverto-libev            x86_64            0.3.2-10.el10        baseos                13 k
rpcbind                   x86_64            1.2.7-3.el10         baseos                57 k
sssd-nfs-idmap            x86_64            2.10.2-3.el10_0.2    baseos                36 k
Transaction Summary
=====
Install 7 Packages

Total download size: 804 k
Installed size: 2.0 M
Downloading Packages:
(1/7): gssproxy-0.9.2-10.el10.x86_64.rpm                       1.1 MB/s | 111 kB    00:00
(2/7): libnfsidmap-2.8.2-3.el10.x86_64.rpm                     480 kB/s | 61 kB    00:00
(3/7): libverto-libev-0.3.2-10.el10.x86_64.rpm                 168 kB/s | 13 kB    00:00
(4/7): libev-4.33-14.el10.x86_64.rpm                           282 kB/s | 52 kB    00:00
(5/7): sssd-nfs-idmap-2.10.2-3.el10_0.2.x86_64.rpm             1.2 MB/s | 36 kB    00:00
(6/7): nfs-utils-2.8.2-3.el10.x86_64.rpm                       3.6 MB/s | 473 kB    00:00
(7/7): rpcbind-1.2.7-3.el10.x86_64.rpm                         166 kB/s | 57 kB    00:00
=====
```

Рис. 1: Установка пакетов

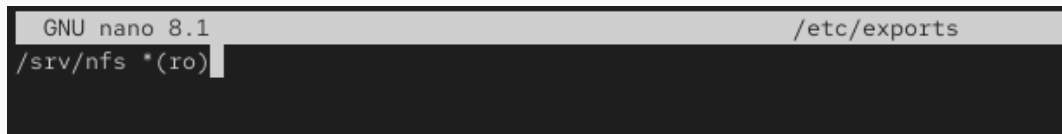
На сервере создадим каталог, который предполагается сделать доступным всем пользователям сети (корень дерева NFS): `mkdir -p /srv/nfs`



```
[root@server.sashubina.net ~]# mkdir -p /srv/nfs  
[root@server.sashubina.net ~]#
```

Рис. 2: создание каталога

**В файле `/etc/exports` пропишем подключаемый через NFS общий каталог с доступом только на чтение: `/srv/nfs *(ro)`**



```
GNU nano 8.1 /etc/exports
/srv/nfs *(ro)
```

**Рис. 3:** Редактирование файла



Для общего каталога зададим контекст безопасности NFS: `semanage fcontext -a -t nfs_t "/srv/nfs(/.*)?"` Применим изменённую настройку SELinux к файловой системе: `restorecon -vR /srv/nfs`

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

**Рис. 4:** зададим контекст безопасности и применим изменённую настройку

# Запустим сервер NFS Настроим межсетевой экран для работы сервера NFS

```
[root@server.sashubina.net ~]# systemctl start nfs-server.service
[root@server.sashubina.net ~]# systemctl enable nfs-server.service
Created symlink '/etc/systemd/system/multi-user.target.wants/nfs-server.service' → '/usr/lib/systemd/system/nfs-server.service'.
[root@server.sashubina.net ~]# firewall-cmd --add-service=nfs
success
[root@server.sashubina.net ~]# firewall-cmd --add-service=nfs --permanent
success
[root@server.sashubina.net ~]# firewall-cmd --reload
success
[root@server.sashubina.net ~]#
```

Рис. 5: Настройка межсетевого экрана

# На клиенте установим необходимое для работы NFS программное обеспечение: `dnf -y install nfs-utils`

```
[sashubina@client.sashubina.net ~]$ sudo -i
[sudo] password for sashubina:
[root@client.sashubina.net ~]# dnf -y install nfs-utils
```

Extra Packages for Enterprise Linux 10	2.6 kB/s	48 kB	00:15
Extra Packages for Enterprise Linux 10	8.8 MB/s	4.8 MB	00:00
Rocky Linux 10 - BaseOS	7.4 kB/s	4.3 kB	00:00
Rocky Linux 10 - AppStream	11 kB/s	4.3 kB	00:00
Rocky Linux 10 - CRB	12 kB/s	4.3 kB	00:00
Rocky Linux 10 - Extras	8.8 kB/s	3.1 kB	00:00

Dependencies resolved.

Package	Arch	Version	Repository	Size
Installing:				
nfs-utils	x86_64	1:2.8.2-3.el10	baseos	473 k
Installing dependencies:				
gssproxy	x86_64	0.9.2-10.el10	baseos	111 k
libev	x86_64	4.33-14.el10	baseos	52 k
libnfsidmap	x86_64	1:2.8.2-3.el10	baseos	61 k
libverto-libev	x86_64	0.3.2-10.el10	baseos	13 k
rpcbind	x86_64	1.2.7-3.el10	baseos	57 k
sssd-nfs-idmap	x86_64	2.10.2-3.el10_0.2	baseos	36 k

Transaction Summary

Install 7 Packages

Total download size: 804 k  
Installed size: 2.0 M  
Downloading Packages:

(1/7): gssproxy-0.9.2-10.el10.x86_64.rpm	1.4 MB/s	111 kB	00:00
(2/7): libverto-libev-0.3.2-10.el10.x86_64.rpm	627 kB/s	13 kB	00:00
(3/7): libev-4.33-14.el10.x86_64.rpm	455 kB/s	52 kB	00:00
(4/7): libnfsidmap-2.8.2-3.el10.x86_64.rpm	500 kB/s	61 kB	00:00
(5/7): nfs-utils-2.8.2-3.el10.x86_64.rpm	8.1 MB/s	473 kB	00:00
(6/7): rpcbind-1.2.7-3.el10.x86_64.rpm	973 kB/s	57 kB	00:00
(7/7): sssd-nfs-idmap-2.10.2-3.el10_0.2.x86_64.rpm	436 kB/s	36 kB	00:00

Рис. 6: Установка пакетов

На клиенте попробуем посмотреть имеющиеся подмонтированные удалённые ресурсы: `showmount -e server.sashubina.net`

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

**Рис. 7:** Просмотр подмонтированных удаленных ресурсов

Попробуем на сервере остановить сервис межсетевого экрана: `systemctl stop firewalld.service`

```
[root@server.sashubina.net ~]# systemctl stop firewalld.service  
[root@server.sashubina.net ~]#
```

**Рис. 8:** остановка сервиса межсетевого экрана

Затем на клиенте вновь попробуем подключиться к удалённо смонтированному ресурсу: `showmount -e server.sashubina.net`

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

**Рис. 9:** Подключение к удаленно смонтированному ресурсу

На сервере запустим сервис межсетевого экрана `systemctl start firewalld`

```
[root@server.sashubina.net ~]# systemctl start firewalld  
[root@server.sashubina.net ~]#
```

**Рис. 10:** запустим сервис межсетевого экрана

# На сервере посмотрим, какие службы задействованы при удалённом монтировании:

```
[root@server.sashubina.net ~]# lsof | grep TCP
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1001/gvfs
Output information may be incomplete.
lsof: WARNING: can't stat() fuse.portal file system /run/user/1001/doc
Output information may be incomplete.
systemd 1 root 420u IPv6 8854 0t0 TCP *:websock (LISTEN)
systemd 1 root 432u IPv4 42421 0t0 TCP *:sunrpc (LISTEN)
systemd 1 root 440u IPv6 42435 0t0 TCP *:sunrpc (LISTEN)
cupsd 1237 root 7u IPv6 11445 0t0 TCP localhost:ipp (LISTEN)
cupsd 1237 root 8u IPv4 11446 0t0 TCP localhost:ipp (LISTEN)
sshd 1253 root 7u IPv4 11463 0t0 TCP *:down (LISTEN)
sshd 1253 root 8u IPv6 11465 0t0 TCP *:down (LISTEN)
sshd 1253 root 9u IPv4 11467 0t0 TCP *:ssh (LISTEN)
sshd 1253 root 10u IPv6 11469 0t0 TCP *:ssh (LISTEN)
named 1341 named 27u IPv4 10793 0t0 TCP localhost:domain (LISTEN)
named 1341 named 29u IPv4 10794 0t0 TCP localhost:domain (LISTEN)
named 1341 named 33u IPv6 10797 0t0 TCP localhost:domain (LISTEN)
named 1341 named 34u IPv6 10798 0t0 TCP localhost:domain (LISTEN)
named 1341 named 35u IPv4 11530 0t0 TCP localhost:rndc (LISTEN)
named 1341 named 36u IPv4 11531 0t0 TCP localhost:rndc (LISTEN)
named 1341 named 37u IPv6 11533 0t0 TCP localhost:rndc (LISTEN)
named 1341 named 38u IPv6 11534 0t0 TCP localhost:rndc (LISTEN)
named 1341 named 41u IPv4 11744 0t0 TCP server.sashubina.net:domain (LISTEN)
named 1341 named 42u IPv4 11745 0t0 TCP server.sashubina.net:domain (LISTEN)
named 1341 named 45u IPv4 24297 0t0 TCP dhcp.sashubina.net:domain (LISTEN)
named 1341 named 46u IPv4 24298 0t0 TCP dhcp.sashubina.net:domain (LISTEN)
named 1341 1343 isc-net-0 named 27u IPv4 10793 0t0 TCP localhost:domain (LISTEN)
named 1341 1343 isc-net-0 named 29u IPv4 10794 0t0 TCP localhost:domain (LISTEN)
named 1341 1343 isc-net-0 named 33u IPv6 10797 0t0 TCP localhost:domain (LISTEN)
named 1341 1343 isc-net-0 named 34u IPv6 10798 0t0 TCP localhost:domain (LISTEN)
named 1341 1343 isc-net-0 named 35u IPv4 11530 0t0 TCP localhost:rndc (LISTEN)
named 1341 1343 isc-net-0 named 36u IPv4 11531 0t0 TCP localhost:rndc (LISTEN)
named 1341 1343 isc-net-0 named 37u IPv6 11533 0t0 TCP localhost:rndc (LISTEN)
named 1341 1343 isc-net-0 named 38u IPv6 11534 0t0 TCP localhost:rndc (LISTEN)
named 1341 1343 isc-net-0 named 41u IPv4 11744 0t0 TCP server.sashubina.net:domain (LISTEN)
named 1341 1343 isc-net-0 named 42u IPv4 11745 0t0 TCP server.sashubina.net:domain (LISTEN)
named 1341 1343 isc-net-0 named 45u IPv4 24297 0t0 TCP dhcp.sashubina.net:domain (LISTEN)
named 1341 1343 isc-net-0 named 46u IPv4 24298 0t0 TCP dhcp.sashubina.net:domain (LISTEN)
```

Рис. 11: Задействованные службы при удаленном монтировании по протоколу TCP

```
[root@server.sashubina.net ~]# lsof | grep UDP
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1001/gvfs
Output information may be incomplete.
lsof: WARNING: can't stat() fuse.portal file system /run/user/1001/doc
Output information may be incomplete.
systemd 1 root 430u IPv4 42428 0t0 UDP *:sunrpc
systemd 1 root 440u IPv6 42442 0t0 UDP *:sunrpc
avahi-daemon 885 avahi 12u IPv4 7796 0t0 UDP *:mdns
avahi-daemon 885 avahi 13u IPv6 7796 0t0 UDP *:mdns
```



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

```
[root@server.sashubina.net ~]# firewall-cmd --get-services
0-AD RH-Satellite-6 RH-Satellite-6-capsule afp alvr amanda-client amanda-k5-client amqp amqps anno-1602 anno-1800 apcupsd aseqnet audit ausweisapp2 bacula ba
cula-client bareos-director bareos-filedaemon bareos-storage bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitcoin-testnet-rpc bittorrent-lsd ceph ceph-exporter
ceph-mon cfengine checkmk-agent civilization-iv civilization-v cockpit collectd condor-collector cratedb ctdb dds dds-multicast dds-unicast dhcp dhcpv6 dhcp
v6-client distcc dns dns-over-qtcp dns-over-tls docker-registry docker-swarm dropbox-lansync elasticsearch etcd-client etcd-server factorio finger foreman fo
reman-proxy freeipa-4 freeipa-ldap freeipa-ldaps freeipa-replication freeipa-trust ftp galera ganglia-client ganglia-master git gpsd grafana gre high-availab
ility http http3 https ident imap imaps iperf2 iperf3 ipfs ipp ipp-client ipsec irc ircs iscsi-target isns jenkins kadmin kdeconnect kerberos kibana klogin k
passwd kprop kshell kube-api kube-apiserver kube-control-plane kube-control-plane-secure kube-controller-manager kube-controller-manager-secure kube-nodeport
-services kube-scheduler kube-scheduler-secure kube-worker kubelet kubelet-readonly kubelet-worker ldap ldaps libvirt libvirt-tls lightning-network llmnr llm
nr-client llmnr-tcp llmnr-udp managesieve matrix mdns memcache minecraft minidlna mndp mongodb mosh mountd mpd mqtt mqtt-tls ms-wbt mssql murmur mysql nbd ne
bula need-for-speed-most-wanted netbios-ns netdata-dashboard nfs nfs3 nmea-0183 nrpe ntp nut opentelemetry openvpn ovirt-imageio ovirt-storageconsole ovirt-v
mconsole plex pmcd pmproxy pmwebapi pmwebapis pop3 pop3s postgresql privoxy prometheus prometheus-node-exporter proxy-dhcp ps2link ps3netsrv ptp pulseaudio p
uppetmaster quassel radius radsec rdp redis redis-sentinel rootd rpc-bind rquotad rsh rsyncd rtsp salt-master samba samba-client samba-dc sane settlers-histo
ry-collection sip sips slimevr slp smtp smtp-submission smtps snmp snmptls snmptls-trap snmptrap spideroak-lansync spotify-sync squid ssdp ssh ssh-custom sta
tsrv steam-lan-transfer steam-streaming stellaris stronghold-crusader stun stuns submission supertuxkart svdrp svn syncthing syncthing-gui syncthing-relay sy
nergy syscomlan syslog syslog-tls telnet tentacle terraria tftp tile38 tinc tor-socks transmission-client turn turns upnp-client vdsd vnc-server vrrp warpina
tor wdem-http wdem-https wireguard ws-discovery ws-discovery-client ws-discovery-host ws-discovery-tcp ws-discovery-udp wsdd wsdd-http wsmn wsmans xdmcp xmp
p-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-java-gateway zabbix-server zabbix-trapper zabbix-web-service zero-k zerotier

[root@server.sashubina.net ~]# firewall-cmd --add-service=mountd --add-service=rpc-bind
success

[root@server.sashubina.net ~]# firewall-cmd --add-service=mountd --add-service=rpc-bind --permanent
success

[root@server.sashubina.net ~]# firewall-cmd --reload
success
```

Рис. 13: Настройка межсетевого экрана на сервере

На клиенте проверим подключение удалённого ресурса: `showmount -e server.sashubina.net`

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

**Рис. 14:** Подключение к удаленно смонтированному ресурсу

# Монтирование NFS на клиенте

---

**На клиенте создадим каталог, в который будет монтироваться удалённый ресурс, и подмонтируем дерево NFS:**

```
[root@client.sashubina.net ~]# mkdir -p /mnt/nfs  
[root@client.sashubina.net ~]# mount server.sashubina.net:/srv/nfs /mnt/  
nfs  
[root@client.sashubina.net ~]#
```

**Рис. 15:** создание каталога

# Проверим, что общий ресурс NFS подключён правильно: mount

```
[root@client.sashubina.net ~]# mount
/dev/mapper/rl_vbox-root on / type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=4096k,nr_inodes=211279,mode=755,inode64)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=000)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,seclabel,nfsdelegate,memory_recursiveprot)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime,seclabel)
efivarfs on /sys/firmware/efi/efivars type efivarfs (rw,nosuid,nodev,noexec,relatime)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,size=345568k,nr_inodes=819200,mode=755,inode64)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,nosuid,noexec,relatime)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=36,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe_ino=5657)
nqueue on /dev/nqueue type nqueue (rw,nosuid,nodev,noexec,relatime,seclabel)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,nosuid,nodev,relatime,seclabel,pagesize=2M)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime,seclabel)
tracfs on /sys/kernel/tracing type tracfs (rw,nosuid,nodev,noexec,relatime,seclabel)
tmpfs on /run/credentials/systemd-journald.service type tmpfs (ro,nosuid,nodev,noexec,relatime,nosymfollow,seclabel,size=1024k,nr_inodes=1024,mode=700,inode64,nosnap)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
/dev/sda2 on /boot type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
/dev/sda1 on /boot/efi type vfat (rw,relatime,fmask=0077,dmask=0077,codepage=437,iocharset=ascii,shortname=winnt,errors=remount-ro)
/dev/mapper/rl_vbox-home on /home type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
vagrant on /vagrant type vboxsf (rw,nodev,relatime,iocharset=utf8,uid=1000,gid=1000)
vagrant on /vagrant type vboxsf (rw,nodev,relatime,iocharset=utf8,uid=1000,gid=1000,_netdev)
tmpfs on /run/user/1001 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=172776k,nr_inodes=43194,mode=700,uid=1001,gid=1001,inode64)
gvfsd-fuse on /run/user/1001/gvfs type fuse.gvfsd-fuse (rw,nosuid,nodev,relatime,user_id=1001,group_id=1001)
portal on /run/user/1001/doc type fuse.portal (rw,nosuid,nodev,relatime,user_id=1001,group_id=1001)
tmpfs on /run/user/0 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=172776k,nr_inodes=43194,mode=700,inode64)
server.sashubina.net:/srv/nfs on /mnt/nfs type nfs4 (rw,relatime,vers=4.2,rsize=262144,wsize=262144,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,clientaddr=192.168.1.30,local_lock=none,addr=192.168.1.1)
[root@client.sashubina.net ~]#
```

Рис. 16: Монтирование NFS на клиенте

На клиенте в конце файла `/etc/fstab` добавим следующую запись:  
`server.sashubina.net:/srv/nfs /mnt/nfs nfs _netdev 0 0`

```
GNU nano 8.1 /etc/fstab Modified
#
# /etc/fstab
# Created by anaconda on Mon Sep  8 16:08:06 2025
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more
#
# After editing this file, run 'systemctl daemon-reload' to update syst
# units generated from this file.
#
UUID=3b63f118-3ab0-4ef9-924b-0f58bc98b6f7 / xfs
UUID=c2609db3-859d-41c9-a1a4-f66785f24f62 /boot xfs
UUID=F61C-8765 /boot/efi vfat umask=0077,shor
UUID=e05cd838-7f05-42f5-bdf8-498e29d736d5 /home xfs
UUID=10d4a7c4-f6ca-46ac-ae28-80797fcce7c4 none swap
#VAGRANT-BEGIN
# The contents below are automatically generated by Vagrant. Do not mod
vagrant /vagrant vboxsf uid=1000,gid=1000,_netdev 0 0
#VAGRANT-END
server.sashubina.net:/srv/nfs /mnt/nfs nfs _netdev 0 0
```

Рис. 17: Редактирование файла

На клиенте проверим наличие автоматического монтирования удалённых ресурсов при запуске операционной системы: `systemctl status remote-fs.target`

```
[root@client.sashubina.net ~]# systemctl status remote-fs.target
● remote-fs.target - Remote File Systems
   Loaded: loaded (/usr/lib/systemd/system/remote-fs.target; enabled;>
   Active: active since Wed 2025-11-05 18:56:14 UTC; 29min ago
   Invocation: 2d12462253f2442d920d0208d0248f4d
   Docs: man:systemd.special(7)

Nov 05 18:56:14 client.sashubina.net systemd[1]: Reached target remote->
lines 1-7/7 (END)
```

**Рис. 18:** Проверка наличия автоматического монтирования удалённых ресурсов

## Перезапустим клиент и убедимся, что удалённый ресурс подключается автоматически.

```
[root@client.sashubina.net ~]# systemctl status remote-fs.target
● remote-fs.target - Remote File Systems
   Loaded: loaded (/usr/lib/systemd/system/remote-fs.target; enabled;
   Active: active since Wed 2025-11-05 19:27:13 UTC; 19min ago
   Invocation: 7fc04932032446deaa78023d83dff520
   Docs: man:systemd.special(7)

Nov 05 19:27:13 client.sashubina.net systemd[1]: Reached target remote-fs.target

[1]+  Stopped                  systemctl status remote-fs.target
[root@client.sashubina.net ~]# mount | grep nfs
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw,relatime)
server.sashubina.net:/srv/nfs on /mnt/nfs type nfs4 (rw,relatime,vers=4.
2,rsiz=262144,wsiz=262144,namlen=255,hard,proto=tcp,timeo=600,retrans=
2,sec=sys,clientaddr=192.168.1.30,local_lock=none,addr=192.168.1.1,_netc
...)
```

Рис. 19: Проверка



# Подключение каталогов к дереву NFS

---

На сервере создадим общий каталог, в который затем будет подмонтирован каталог с контентом веб-сервера: `mkdir -p /srv/nfs/www` Подмонтируем каталог web-сервера `mount -o bind /var/www/ /srv/nfs/www/`

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

**Рис. 20:** создание и подмонтирование каталога

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

```
[root@server.sashubina.net srv]# ls -la /srv/nfs
total 0
drwxr-xr-x. 3 root    root    17 Nov  5 19:49 .
drwxr-xr-x. 3 root    root    17 Nov  5 19:03 ..
drwxr-xr-x. 4 apache  apache 33 Sep 23 17:40 www
[root@server.sashubina.net srv]#
```

Рис. 21: Содержимое каталога

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

```
[root@client.sashubina.net ~]# ls -la /mnt/nfs
total 0
drwxr-xr-x. 3 root root 17 Nov  5 19:49 .
drwxr-xr-x. 3 root root 17 Nov  5 19:18 ..
drwxr-xr-x. 2 root root  6 Nov  5 19:49 www
[root@client.sashubina.net ~]#
```

Рис. 22: Содержимое каталога

На сервере в файле `/etc/exports` добавим экспорт каталога веб-сервера с удалённого ресурса: `/srv/nfs/www 192.168.0.0/16(rw)`



```
GNU nano 8.1 /etc/exports
/srv/nfs *(ro)
/srv/nfs/www 192.168.0.0/16(rw)
```

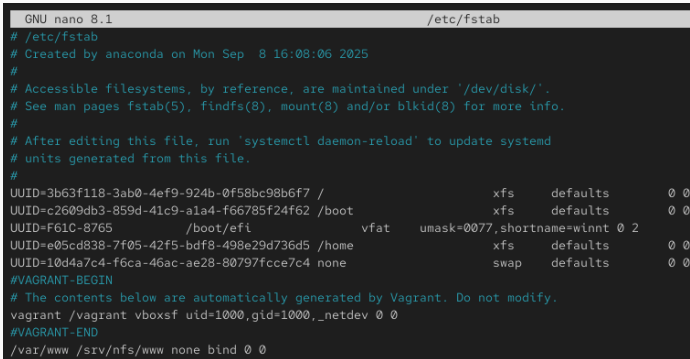
Рис. 23: Редактирование файла

Экспортируем все каталоги, упомянутые в файле `/etc/exports`: `exportfs -r` Проверим на клиенте каталог `/mnt/nfs`.

```
[root@client.sashubina.net ~]# ls -la /mnt/nfs
total 0
drwxr-xr-x. 3 root root 17 Nov  5 19:49 .
drwxr-xr-x. 3 root root 17 Nov  5 19:18 ..
drwxr-xr-x. 4  48   48 33 Sep 23 17:40 www
```

Рис. 24: Содержимое каталога

На сервере в конце файла `/etc/fstab` добавим следующую запись: `/var/www /srv/nfs/www none bind 0 0`



```
GNU nano 8.1 /etc/fstab
# /etc/fstab
# Created by anaconda on Mon Sep  8 16:08:06 2025
#
# 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=3b63f118-3ab0-4ef9-924b-0f58bc98b6f7 / xfs defaults 0 0
UUID=c2609db3-859d-41c9-a1a4-f66785f24f62 /boot xfs defaults 0 0
UUID=F61C-8765 /boot/efi vfat umask=0077,shortname=winnt 0 2
UUID=e05cd838-7f05-42f5-bdf8-498e29d736d5 /home xfs defaults 0 0
UUID=10d4a7c4-f6ca-46ac-ae28-80797fcce7c4 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,_netdev 0 0
#VAGRANT-END
/var/www /srv/nfs/www none bind 0 0
```

Рис. 25: Редактирование файла

Повторно экспортируем каталоги, указанные в файле `/etc/exports`:  
`exportfs -r`

A terminal window with a dark background and light gray text. The prompt is [root@server.sashubina.net ~]# and the command being entered is exportfs -r.

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

**Рис. 26:** экспорт закалов



## На клиенте проверим каталог /mnt/nfs. Перед этим подмонтируем каталог

```
[root@client.sashubina.net ~]# mount -t nfs -o vers=3 server.sashubina.net:/srv/nfs/www /mnt/nfs
Created symlink '/run/systemd/system/remote-fs.target.wants/rpc-statd.service' → '/usr/lib/systemd/system/rpc-statd.service'.
[root@client.sashubina.net ~]# ls -la /mnt/nfs
total 0
drwxr-xr-x. 4 48 48 33 Sep 23 17:40 .
drwxr-xr-x. 3 root root 17 Nov 5 19:18 ..
drwxr-xr-x. 2 48 48 6 Jul 15 00:00 cgi-bin
drwxr-xr-x. 5 48 48 100 Oct 27 17:07 html
[root@client.sashubina.net ~]#
```

Рис. 27: Содержимое каталога

## **Подключение каталогов для работы пользователей**

---

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

```
[root@server.sashubina.net ~]# mkdir -p -m 700 ~/common
[root@server.sashubina.net ~]# cd ~/common
[root@server.sashubina.net common]# touch sashubina@server.txt
[root@server.sashubina.net common]#
```

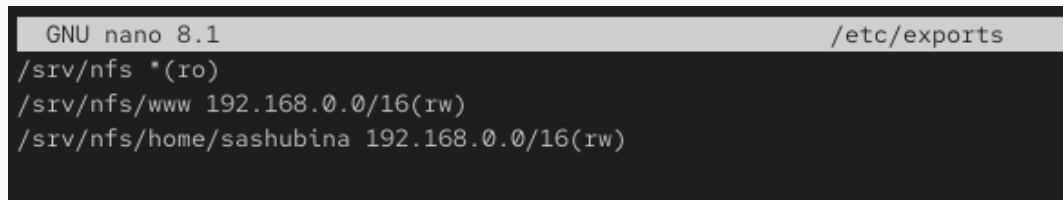
Рис. 28: Создание каталога

На сервере создадим общий каталог для работы пользователя sashubina по сети: `mkdir -p /srv/nfs/home/user` Подмонтируем каталог common пользователя sashubina в NFS `mount -o bind /home/user/common /srv/nfs/home/user`

```
[root@server.sashubina.net common]# mkdir -p /srv/nfs/home/sashubina
[root@server.sashubina.net common]# mount -o bind /home/sashubina/common /srv/nfs/home/sashubina
mount: /srv/nfs/home/sashubina: special device /home/sashubina/common does not exist.
      dmesg(1) may have more information after failed mount system call.
mount: (hint) your fstab has been modified, but systemd still uses
      the old version; use 'systemctl daemon-reload' to reload.
[root@server.sashubina.net common]# systemctl daemon-reload
[root@server.sashubina.net common]#
```

**Рис. 29:** Подключение каталогов для работы пользователей

Подключим каталог пользователя в файле `/etc/exports`, прописав в нём (вместо `user` укажите свой логин):`/srv/nfs/home/user 192.168.0.0/16(rw)`

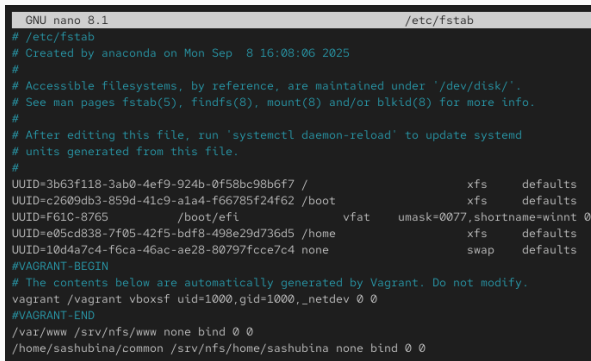


```
GNU nano 8.1 /etc/exports
/srv/nfs *(ro)
/srv/nfs/www 192.168.0.0/16(rw)
/srv/nfs/home/sashubina 192.168.0.0/16(rw)
```

Рис. 30: Редактирование файла

Внесем изменения в файл `/etc/fstab` (вместо `user` укажите свой логин):

```
/home/sashubina/common /srv/nfs/home/sashubina none bind 0 0
```



```
GNU nano 8.1 /etc/fstab
# /etc/fstab
# Created by anaconda on Mon Sep  8 16:08:06 2025
#
# 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=3b63f118-3ab0-4ef9-924b-0f58bc98b6f7 / xfs defaults
UUID=c2609db3-859d-41c9-a1a4-f66785f24f62 /boot xfs defaults
UUID=F61C-8765 /boot/efi vfat umask=0077,shortname=winnt 0
UUID=e05cd838-7f05-42f5-bdf8-498e29d736d5 /home xfs defaults
UUID=10d4a7c4-f6ca-46ac-ae28-80797fcce7c4 none swap defaults
#VAGRANT-BEGIN
# The contents below are automatically generated by Vagrant. Do not modify.
vagrant /vagrant vboxsf uid=1000,gid=1000,_netdev 0 0
#VAGRANT-END
/var/www /srv/nfs/www none bind 0 0
/home/sashubina/common /srv/nfs/home/sashubina none bind 0 0
```

Рис. 31: Редактирование файла

## Повторно экспортируем каталоги: `exportfs -r`, а также я поменяла права доступа

```
[root@server.sashubina.net common]# chown -R sashubina:sashubina /home/sashubina/common/  
[root@server.sashubina.net common]# chown -R sashubina:sashubina /srv/nfs/home/sashubina/
```

Рис. 32: права доступа

## На клиенте проверим каталог /mnt/nfs.

```
[sashubina@client.sashubina.net ~]$ ls -la /mnt/nfs
total 0
drwx-----. 2 sashubina sashubina 34 Nov  5 20:45 .
drwxr-xr-x. 3 root      root      17 Nov  5 19:18 ..
-rw-r--r--. 1 sashubina sashubina  0 Nov  5 20:47 sashubina@server.txt
[sashubina@client.sashubina.net ~]$
```

**Рис. 33:** Проверка содержимого каталога

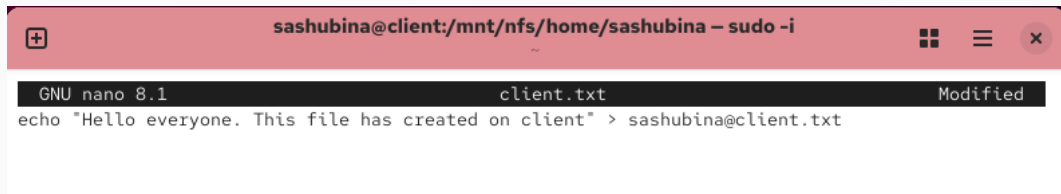


На клиенте под пользователем `user` перейдем в каталог `/mnt/nfs/home/sashubina` и попробуем создать в нём файл `sashubina@client.txt` и внести в него какие-либо изменения

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

Рис. 34: Создание файла

## Добавила описание



```
sashubina@client:/mnt/nfs/home/sashubina - sudo -i
```

```
GNU nano 8.1 client.txt Modified
echo "Hello everyone. This file has created on client" > sashubina@client.txt
```

Рис. 35: Добавила описание

**Попробуем проделать это под пользователем root.**

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

**Рис. 36:** попробуем через root

На сервере посмотрите, появились ли изменения в каталоге пользователя `/home/user/common`.

```
[root@server.sashubina.net ~]# ls -la /home/sashubina/common
total 8
drwx-----.  2 sashubina sashubina   80 Nov  5 21:06 .
drwx-----. 17 sashubina sashubina 4096 Nov  5 20:44 ..
-rw-r--r--.   1 sashubina sashubina   78 Nov  5 21:06 client.txt
-rw-r--r--.   1 sashubina sashubina    0 Nov  5 21:03 sashubina@client.txt
-rw-r--r--.   1 sashubina sashubina    0 Nov  5 20:47 sashubina@server.txt
[root@server.sashubina.net ~]#
```

**Рис. 37:** изменения в каталоге

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

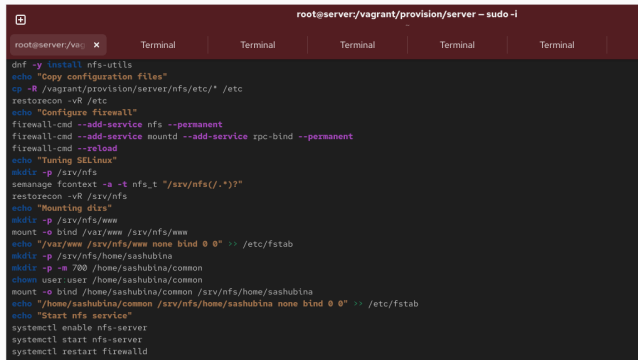
---

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

```
[root@server.sashubina.net ~]# cd /vagrant/provision/server
[root@server.sashubina.net server]# mkdir -p /vagrant/provision/server/nfs/etc
[root@server.sashubina.net server]# cp -R /etc/exports /vagrant/provision/server/nfs/etc/
[root@server.sashubina.net server]# cd /vagrant/provision/server
[root@server.sashubina.net server]# touch nfs.sh
[root@server.sashubina.net server]# chmod +x nfs.sh
[root@server.sashubina.net server]#
```

Рис. 38: создание файла

# Открыв его на редактирование, пропишем в нём следующий скрипт:



```
root@server:/vagrant/provision/server -- sudo -i
root@server/vag x Terminal Terminal Terminal Terminal Terminal
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/sashubina
mkdir -p -m 700 /home/sashubina/common
chown user:user /home/sashubina/common
mount -o bind /home/sashubina/common /srv/nfs/home/sashubina
echo "/home/sashubina/common /srv/nfs/home/sashubina none bind 0 0" >> /etc/fstab
echo "Start nfs service"
systemctl enable nfs-server
systemctl start nfs-server
systemctl restart firewalld
```

Рис. 39: Редактирование файла

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

```
[root@client.sashubina.net client]# cd /vagrant/provision/client
[root@client.sashubina.net client]# touch nfs.sh
[root@client.sashubina.net client]# chmod +x nfs.sh
[root@client.sashubina.net client]# █
```

Рис. 40: создание файла



Открыв его на редактирование, пропишем в нём следующий скрипт:

```
GNU nano 8.1                                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.sashubina.net:/srv/nfs /mnt/nfs
echo "server.sashubina.net:/srv/nfs /mnt/nfs nfs _netdev 0 0" >> /etc/fstab
restorecon -vR /etc
```

Рис. 41: Редактирование файла

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"
```

Рис. 42: Редактирование файла

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

Рис. 43: Редактирование файла

В процессе выполнения данной лабораторной работы я приобрела навыки настройки сервера NFS для удалённого доступа к ресурсам.