

# Администрирование сетевых подсистем

## Лабораторная работа №13

---

Машков И. Е.

13 февраля 2026

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Машков Илья Евгеньевич
- Студент 3-го курса, группа НФИбд-02-23
- Российский университет дружбы народов
- 1132231984@pfur.ru
- <https://github.com/7S7eVe7N7>

Приобретение навыков настройки сервера NFS для удалённого доступа к ресурсам.

# Выполнение лабораторной работы

```
[user@server ~]$ sudo -i
[sudo] password for user:
[root@server ~]# dnf -y install nfs-utils
Last metadata expiration check: 0:20:31 ago on Thu 12 Feb 2026 04:53:07 PM MSK
.
Dependencies resolved.
=====
Package                Arch      Version              Repository           Size
=====
Installing:
nfs-utils                x86_64    1:2.5.4-38.el9      baseos               432 k
Upgrading:
libipa_hbac              x86_64    2.9.7-4.el9_7.1     baseos               33 k
libldb                   x86_64    4.22.4-6.el9        baseos               181 k
libsmbclient             x86_64    4.22.4-6.el9        baseos               74 k
libsss_certmap           x86_64    2.9.7-4.el9_7.1     baseos               88 k
libsss_idmap             x86_64    2.9.7-4.el9_7.1     baseos               39 k
libsss_nss_idmap         x86_64    2.9.7-4.el9_7.1     baseos               43 k
libsss_sudo              x86_64    2.9.7-4.el9_7.1     baseos               33 k
libtalloc                x86_64    2.4.3-1.el9         baseos               33 k
libtdb                   x86_64    1.4.13-1.el9        baseos               53 k
libtevent                x86_64    0.16.2-1.el9        baseos               50 k
libwbclient              x86_64    4.22.4-6.el9        baseos               42 k
samba-client-libs        x86_64    4.22.4-6.el9        baseos               5.3 M
samba-common             noarch    4.22.4-6.el9        baseos               173 k
samba-common-libs        x86_64    4.22.4-6.el9        baseos               104 k
sssd                     x86_64    2.9.7-4.el9_7.1     baseos               25 k
sssd-ad                  x86_64    2.9.7-4.el9_7.1     baseos               217 k
sssd-client              x86_64    2.9.7-4.el9_7.1     baseos               158 k
sssd-common              x86_64    2.9.7-4.el9_7.1     baseos               1.6 M
sssd-common-pac          x86_64    2.9.7-4.el9_7.1     baseos               94 k
sssd-ipa                 x86_64    2.9.7-4.el9_7.1     baseos               286 k
sssd-kcm                 x86_64    2.9.7-4.el9_7.1     baseos               107 k
sssd-krb5                x86_64    2.9.7-4.el9_7.1     baseos               70 k
sssd-krb5-common         x86_64    2.9.7-4.el9_7.1     baseos               92 k
sssd-ldap                x86_64    2.9.7-4.el9_7.1     baseos               159 k
sssd-proxy               x86_64    2.9.7-4.el9_7.1     baseos               70 k
Installing dependencies:
gssproxy                 x86_64    0.8.4-7.el9         baseos               108 k
```

Рис. 1: Установка nfs-utils на сервер

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

Рис. 2: Работа с контекстом безопасности и запуск NFS

```
[root@server ~]# firewall-cmd --add-service=nfs
success
[root@server ~]# firewall-cmd --add-service=nfs --permanent
success
[root@server ~]# firewall-cmd --reload
success
[root@server ~]#
```

Рис. 3: Настройка межсетевого экрана

# Выполнение лабораторной работы

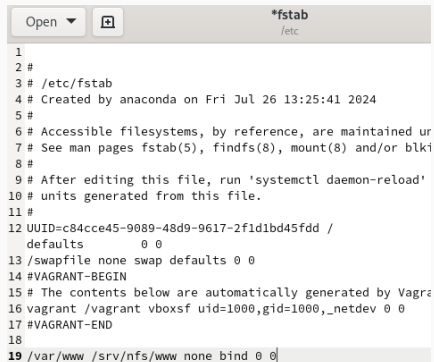
```
[root@server ~]# firewall-cmd --get-services
RH-Satellite-6 RH-Satellite-6-capsule afp amanda-client amanda-k5-client amqp
amqps apcupsd audit ausweisapp2 bacula bacula-client bareos-director bareos-fi
ledaemon bareos-storage bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitcoin-tes
tnet-rpc bittorrent-lsd ceph ceph-exporter ceph-mon cfengine checkmk-agent coc
kpit collectd condor-collector cratedb ctdb dds dds-multicast dds-unicast dhcp
dhcpcv6 dhcpcv6-client distcc dns dns-over-tls docker-registry docker-swarm dro
pbox-lansync elasticsearch etcd-client etcd-server finger foreman foreman-prox
y freeipa-4 freeipa-ldap freeipa-ldaps freeipa-replication freeipa-trust ftp g
alera ganglia-client ganglia-master git gssd grafana gre high-availability htt
p http3 https ident imap imaps ipfs ipp ipp-client ipsec irc ircs iscsi-target
isns jenkins kadmin kdeconnect kerberos kibana klogin kpasswd kprop kshell ku
be-api kube-apiserver kube-control-plane kube-control-plane-secure kube-contro
ller-manager kube-controller-manager-secure kube-nodeport-services kube-schedu
ler kube-scheduler-secure kube-worker kubelet kubelet-readonly kubelet-worker
ldap ldaps libvirt libvirt-tls lightning-network llmnr llmnr-client llmnr-tcp
llmnr-udp managesieve matrix mdns memcache minidlna mongodb mosh mounstd mqtt m
qtt-tls ms-wbt mssql murmur mysql nbd nebula netbios-ns netdata-dashboard nfs
nfs3 nmea-0183 nrpe ntp nut openvpn ovirt-imageio ovirt-storageconsole ovirt-v
mconsole plex pmcd pmproxy pmwebapi pmwebapis pop3 pop3s postgresql privoxy pr
ometheus prometheus-node-exporter proxy-dhcp ps2link ps3netshv ptp pulseaudio
puppetmaster quassel radius rdp redis redis-sentinel rpc-bind rquotad rsh rsyn
cd rtsp salt-master samba samba-client samba-dc sane sip sips slp smtp smtp-su
bmission smtps snmp snmpv3 snmpv3-trap snmptrap spideroak-lansync spotify-sy
nc squid ssdp ssh ssh-custom steam-streaming svdrp svn syncthing syncthing-gui
syncthing-relay synergy syslog syslog-tls telnet tentacle tftp tile38 tinc to
r-socks transmission-client upnp-client vdsim vnc-server warpinator wbem-http w
bem-https wireguard ws-discovery ws-discovery-client ws-discovery-tcp ws-disco
very-udp wsman wsmans xdmcp xmpp-bosh xmpp-client xmpp-local xmpp-server zabbi
x-agent zabbi-server zerotier
[root@server ~]# firewall-cmd --add-service=mounstd --add-service=rpc-bind
success
[root@server ~]# firewall-cmd --add-service=mounstd --add-service=rpc-bind --pe
rmanent
success
[root@server ~]# firewall-cmd --reload
success
[root@server ~]#
```

Рис. 4: Добавление служб rpc-bind и mounstd на сервере



```
[root@client ~]# showmount -e server.user.net  
clnt_create: RPC: Unknown host  
[root@client ~]#
```

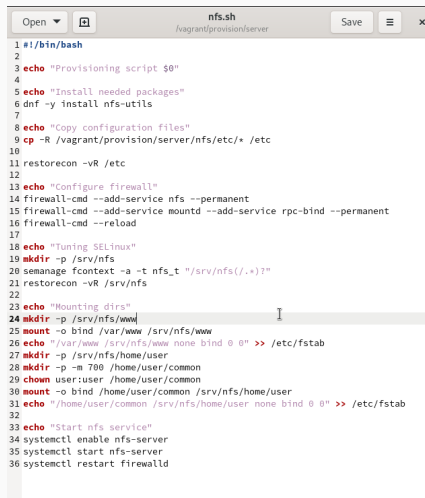
Рис. 5: Неудачная попытка подключения удалённого ресурса



```
1
2 #
3 # /etc/fstab
4 # Created by anaconda on Fri Jul 26 13:25:41 2024
5 #
6 # Accessible filesystems, by reference, are maintained ur
7 # See man pages fstab(5), findfs(8), mount(8) and/or blk
8 #
9 # After editing this file, run 'systemctl daemon-reload'
10 # units generated from this file.
11 #
12 UUID=c84cce45-9089-48d9-9617-2f1d1bd45fdd /
   defaults          0 0
13 /swapfile none swap defaults 0 0
14 #VAGRANT-BEGIN
15 # The contents below are automatically generated by Vagra
16 vagrant /vagrant vboxsf uid=1000,gid=1000,_netdev 0 0
17 #VAGRANT-END
18
19 /var/www /srv/nfs/www none bind 0 0
```

Рис. 6: Добавление записи в fstab

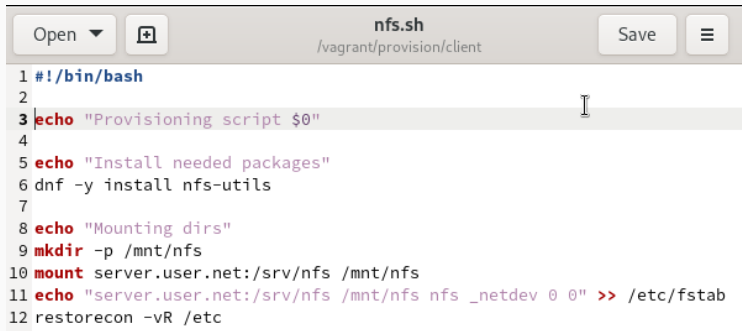
# Выполнение лабораторной работы



The screenshot shows a terminal window titled 'nfs.sh' with the path '/vagrant/provision/server' below it. The window has a standard Linux terminal interface with a title bar, a menu icon, and buttons for 'Open', 'Save', and a close 'x' button. The terminal content is a shell script with 36 lines, numbered 1 through 36. The script performs various tasks including installing packages, copying files, configuring the firewall, tuning SELinux, creating directories, mounting filesystems, and starting the NFS service. The script is written in a mix of red and black text, with red text likely representing comments or echo statements, and black text representing commands. The script ends with a 'systemctl restart firewallld' command.

```
1 #!/bin/bash
2
3 echo "Provisioning script $0"
4
5 echo "Install needed packages"
6 dnf -y install nfs-utils
7
8 echo "Copy configuration files"
9 cp -R /vagrant/provision/server/nfs/etc/* /etc
10
11 restorecon -vR /etc
12
13 echo "Configure firewall"
14 firewall-cmd --add-service nfs --permanent
15 firewall-cmd --add-service mountd --add-service rpc-bind --permanent
16 firewall-cmd --reload
17
18 echo "Tuning SELinux"
19 mkdir -p /srv/nfs
20 semanage fcontext -a -t nfs_t "/srv/nfs(/.*)?"
21 restorecon -vR /srv/nfs
22
23 echo "Mounting dirs"
24 mkdir -p /srv/nfs/www
25 mount -o bind /var/www /srv/nfs/www
26 echo "/var/www /srv/nfs/www none bind 0 0" >> /etc/fstab
27 mkdir -p /srv/nfs/home/user
28 mkdir -p -m 700 /home/user/common
29 chown user:user /home/user/common
30 mount -o bind /home/user/common /srv/nfs/home/user
31 echo "/home/user/common /srv/nfs/home/user none bind 0 0" >> /etc/fstab
32
33 echo "Start nfs service"
34 systemctl enable nfs-server
35 systemctl start nfs-server
36 systemctl restart firewallld
```

Рис. 7: nfs.sh на сервере



```
1 #!/bin/bash
2
3 echo "Provisioning script $0"
4
5 echo "Install needed packages"
6 dnf -y install nfs-utils
7
8 echo "Mounting dirs"
9 mkdir -p /mnt/nfs
10 mount server.user.net:/srv/nfs /mnt/nfs
11 echo "server.user.net:/srv/nfs /mnt/nfs nfs _netdev 0 0" >> /etc/fstab
12 restorecon -vR /etc
```

Рис. 8: nfs.sh на клиенте

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

Рис. 9: Строки в Vagrantfile для сервера

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

Рис. 10: Строки в Vagrantfile для клиента

Во время выполнения данной лабораторной работы я освоил практические навыки настройки сервера NFS для удалённого доступа к ресурсам.