

Лабораторная работа № 13

Настройка NFS

Демидова Екатерина Алексеевна

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
3.1	Настройка сервера NFSv4	6
3.2	Монтирование NFS на клиенте	9
4	Подключение каталогов к дереву NFS	12
5	Подключение каталогов для работы пользователей	15
5.1	Внесение изменений в настройки внутреннего окружения вирту- альных машины	17
6	Контрольные вопросы	20
7	Выводы	21

Список иллюстраций

3.1	Подключение через NFS каталога только на чтение	6
3.2	Запуск NFS-сервера	7
3.3	Просмотр подмонтированных удалённых ресурсов на клиенте . .	7
3.4	Просмотр текущего системного времени на сервере	8
3.5	Просмотр задействованных при удалённом монтировании служб	8
3.6	Просмотр задействованных при удалённом монтировании служб	8
3.7	Добавление служб rpc-bind и mountd в настройки межсетевого экрана	9
3.8	Проверка подключения удалённого ресурса на клиенте	9
3.9	Проверка правильности подключения общего ресурса NFS	10
3.10	Добавление записи в файл /etc/fstab на клиенте	10
3.11	Проверка наличия автоматического монтирования удалённых ре- сурсов при запуске ОС	11
3.12	Проверка автоматического подключения удалённого ресурса . . .	11
4.1	Проверка содержимого /srv/nfs	12
4.2	Проверка содержимого /mnt/nfs	12
4.3	Добавление в файл /etc/exports экспорт каталога веб-сервера . . .	13
4.4	Проверка содержимого /mnt/nfs	13
4.5	Добавление записи в файл /etc/fstab	13
4.6	Проверка содержимого /mnt/nfs	14
5.1	Проверка прав доступа на каталог	15
5.2	Подключение каталога пользователя в файле /etc/exports	16
5.3	Добавление записи в файл /etc/fstab	16
5.4	Проверка содержимого /mnt/nfs	16
5.5	Проверка содержимого /mnt/nfs	17
5.6	Скрипта файла /vagrant/provision/server/nfs.sh	18
5.7	Скрипта файла /vagrant/provision/client/nfs.sh	18

1 Цель работы

Приобретение практических навыков настройки сервера NFS для удалённого доступа к ресурсам.

2 Задание

1. Установите и настройте сервер NFSv4.
2. Подмонтируйте удалённый ресурс на клиенте.
3. Подключите каталог с контентом веб-сервера к дереву NFS.
4. Подключите каталог для удалённой работы вашего пользователя к дереву NFS.
5. Напишите скрипты для Vagrant, фиксирующие действия по установке и настройке сервера NFSv4 во внутреннем окружении виртуальных машин server и client. Соответствующим образом внесите изменения в Vagrantfile.

3 Выполнение лабораторной работы

3.1 Настройка сервера NFSv4

Загрузим нашу операционную систему и перейдем в рабочий каталог с проектом:

```
cd /var/tmp/eademidova/vagrant
```

Затем запустим виртуальную машину server:

```
make server-up
```

На сервере установим необходимое программное обеспечение:

```
dnf -y install nfs-utils
```

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

```
mkdir -p /srv/nfs
```

В файле /etc/exports пропишем подключаемый через NFS общий каталог с доступом только на чтение(рис. 3.1):

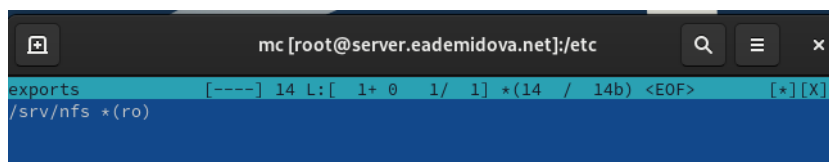


Рис. 3.1: Подключение через NFS каталога только на чтение

Для общего каталога зададим контекст безопасности NFS и применим изменённую настройку SELinux к файловой системе, затем запустим сервер NFS и настроим межсетевой экран для работы сервера NFS(рис. 3.2):

```
[root@server.eademidova.net ~]# mkdir -p /srv/nfs
[root@server.eademidova.net ~]# cd /etc/
[root@server.eademidova.net etc]# mc
[root@server.eademidova.net etc]# semanage fcontext -a -t nfs_t "/srv/nfs(/.*)?"
[root@server.eademidova.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.eademidova.net etc]# systemctl start nfs-server.service
[root@server.eademidova.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.
[root@server.eademidova.net etc]# firewall-cmd --add-service=nfs
success
[root@server.eademidova.net etc]# firewall-cmd --add-service=nfs --permanent
success
[root@server.eademidova.net etc]# firewall-cmd --reload
success
[root@server.eademidova.net etc]#
```

Рис. 3.2: Запуск NFS-сервера

На клиенте установим необходимое для работы NFS программное обеспечение:

```
dnf -y install nfs-utils
```

Попробуем посмотреть имеющиеся подмонтированные удалённые ресурсы(3.3):

```
eademidova@client:~
[eademidova@client.eademidova.net ~]$ showmount -e server.eademidova.net
clnt_create: RPC: Unable to receive
[eademidova@client.eademidova.net ~]$
```

Рис. 3.3: Просмотр подмонтированных удалённых ресурсов на клиенте

Увидим, что протокол удалённого вызова процедур (RPC) не может получить доступ к запрашиваемым ресурсам.

На сервере остановим сервис межсетевого экрана с помощью команды `systemctl stop firewalld.service` и на клиенте вновь попробуем подключиться к удалённо смонтированному ресурсу(3.4):

```

eademidova@client:~
[eademidova@client.eademidova.net ~]$ showmount -e server.eademidova.net
clnt_create: RPC: Unable to receive
[eademidova@client.eademidova.net ~]$ showmount -e server.eademidova.net
Export list for server.eademidova.net:
/srv/nfs *
[eademidova@client.eademidova.net ~]$

```

Рис. 3.4: Просмотр текущего системного времени на сервере

Теперь подключение к удаленно смонтированному ресурсу успешно произведено.

На сервере запустим сервис межсетевого экрана и посмотрим, какие службы задействованы при удалённом монтировании(3.5, 3.6):

```

demidova.net:bootpc->_gateway:bootps
rpcbind 43438      rpc      5u      IPv4      38555      0t0      UDP *:sunrpc
rpcbind 43438      rpc      7u      IPv6      38573      0t0      UDP *:sunrpc
rpc.statd 43440     rpcuser  5u      IPv4      78806      0t0      UDP localhost
:792
rpc.statd 43440     rpcuser  8u      IPv4      78818      0t0      UDP *:55641
rpc.statd 43440     rpcuser  10u     IPv6      78828      0t0      UDP *:55868
rpc.mount 43444     root    4u      IPv4      79725      0t0      UDP *:mountd
rpc.mount 43444     root    6u      IPv6      79731      0t0      UDP *:mountd
[root@server.eademidova.net etc]#

```

Рис. 3.5: Просмотр задействованных при удалённом монтировании служб

```

TCP
rpcbind 43438      rpc      4u      IPv4      38546      0t0      TCP *:sunrpc
rpcbind 43438      rpc      6u      IPv6      38564      0t0      TCP *:sunrpc
rpc.statd 43440     rpcuser  9u      IPv4      78824      0t0      TCP *:53869 (
rpc.statd 43440     rpcuser  11u     IPv6      78832      0t0      TCP *:50533 (
rpc.mount 43444     root    5u      IPv4      79728      0t0      TCP *:mountd
rpc.mount 43444     root    7u      IPv6      79734      0t0      TCP *:mountd
[root@server.eademidova.net etc]# ss -l -n | grep UDP

```

Рис. 3.6: Просмотр задействованных при удалённом монтировании служб

Затем добавим службы rpc-bind и mountd в настройки межсетевого экрана на сервере(3.7):


```
[root@server.eademidova.net etc]# firewall-cmd --get-services
RH-Satellite-6 RH-Satellite-6-capsule afp amanda-client amanda-k5-client amqp amqps apcupsd audit ausweisapp2 b
acula bacula-client bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitcoin-testnet-rpc bittorrent-lsd ceph ceph-mon
cfengine checkmk-agent cockpit collectd condor-collector cratedb ctdb dhcp dhcpv6 dhcpv6-client distcc dns dns
-over-tls docker-registry docker-swarm dropbox-lansync elasticsearch etcd-client etcd-server finger foreman for
eman-proxy freeipa-4 freeipa-ldap freeipa-ldaps freeipa-replication freeipa-trust ftp galera ganglia-client gan
glia-master git gpsd grafana gre high-availability http http3 https ident imap imaps ipfs ipp ipp-client ipsec
irc ircs iscsi-target isns jellyfin jenkins kadmin kdeconnect kerberos kibana klogin kpasswd 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 kubele
t-worker ldap ldaps libvirt libvirt-tls lightning-network llmnr llmnr-tcp llmnr-udp managesieve matrix mdns mem
cache minidlna mongodb mosh mountd mqtt mqtt-tls ms-wbt mssql murmur mysql nbd netbios-ns netdata-dashboard nfs
nfs3 nmea-0183 nrpe ntp nut openvpn ovirt-imageio ovirt-storageconsole ovirt-vmconsole plex pmcd pmproxy pmweb
api pmwebapi pop3 pop3s postgresql privoxy prometheus prometheus-node-exporter proxy-dhcp ps3netsrv ptp pulsea
udio puppetmaster quassel radius rdp redis redis-sentinel rpc-bind rquotad rsh rsyncd rtsp salt-master samba sa
mba-client samba-dc sane sip sips sip smtp smtp-submission smtps snmp snmptls snmptls-trap snmptrap spideroak-l
ansync spotify-sync squid ssdp ssh ssh-custom steam-streaming svdrp svn syncthing syncthing-gui synergy syslog
syslog-tls telnet tentacle tftp tile38 tinc tor-socks transmission-client upnp-client vdsim vnc-server wbem-http
wbem-https wireguard ws-discovery ws-discovery-client ws-discovery-tcp ws-discovery-udp wsman wsmans xdmcp xmp
p-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zabbix-server zerotier
[root@server.eademidova.net etc]# firewall-cmd --add-service=mountd --add-service=rpc-bind
success
[root@server.eademidova.net etc]# firewall-cmd --add-service=mountd --add-service=rpc-bind --permanent
success
[root@server.eademidova.net etc]# firewall-cmd --reload
success
[root@server.eademidova.net etc]#
```

Рис. 3.7: Добавление служб rpc-bind и mountd в настройки межсетевого экрана

Теперь на клиенте проверим подключение удалённого ресурса(3.8):

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

Рис. 3.8: Проверка подключения удалённого ресурса на клиенте

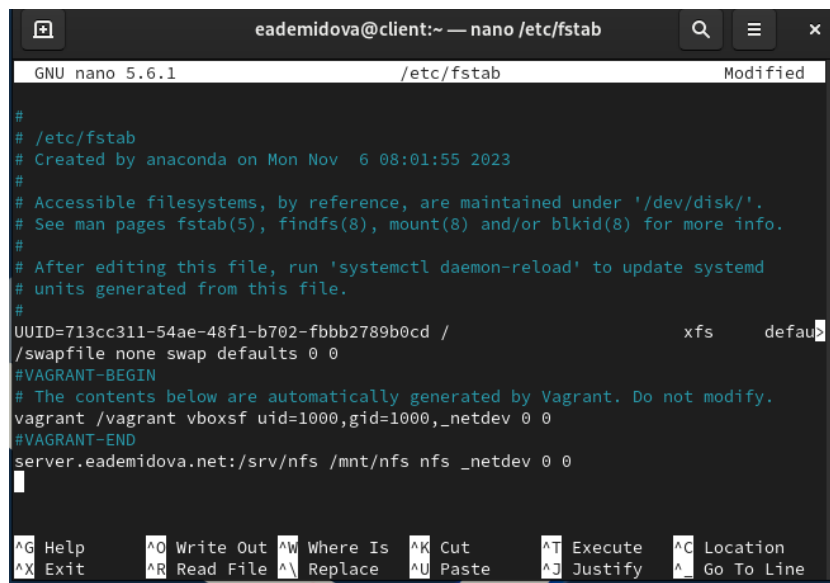
3.2 Монтирование NFS на клиенте

На клиенте создадим каталог, в который будет монтироваться удалённый ре-сурс, и подмонтируем дерево NFS. Затем проверим, что общий ресурс NFS под-ключён правильно(3.9):

```
[root@client.eademidova.net ~]# mkdir -p /mnt/nfs
[root@client.eademidova.net ~]# mount server.eademidova.net:/srv/nfs /mnt/nfs
[root@client.eademidova.net ~]# mount
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=4096k,nr_inodes=114493,mode=755,inode64)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
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)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,size=194916k,nr_inodes=819200,mode=755,inode64)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,seclabel,nsdelegate,memory_recursiveprot)
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)
/dev/sda1 on / type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,nosuid,noexec,relatime)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=29,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,
e_ino=17484)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime,seclabel)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime,seclabel)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,seclabel,pagesize=2M)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime,seclabel)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
none on /run/credentials/systemd-sysctl.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
none on /run/credentials/systemd-tmpfiles-setup-dev.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
none on /run/credentials/systemd-tmpfiles-setup.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
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=97456k,nr_inodes=24364,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)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw,relatime)
server.eademidova.net:/srv/nfs on /mnt/nfs type nfs4 (rw,relatime,vers=4.2,rsize=131072,wsiz=131072,namlen=255,hard,rotoctcp,timeo=600,retrans=2,sec=sys,clientaddr=192.168.1.30,local_lock=none,addr=192.168.1.1)
[root@client.eademidova.net ~]#
```

Рис. 3.9: Проверка правильности подключения общего ресурса NFS

На клиенте в конце файла `/etc/fstab` добавьте следующую запись(3.10):



```
eademidova@client:~ — nano /etc/fstab
GNU nano 5.6.1 /etc/fstab Modified
#
# /etc/fstab
# Created by anaconda on Mon Nov 6 08:01:55 2023
#
# 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=713cc311-54ae-48f1-b702-fbbb2789b0cd / 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,_netdev 0 0
#VAGRANT-END
server.eademidova.net:/srv/nfs /mnt/nfs nfs _netdev 0 0
^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line
```

Рис. 3.10: Добавление записи в файл `/etc/fstab` на клиенте

Мы подключаем каталоги сервера для автоматического монтирования при загрузке через `nfs`, для этого указываются каталоги и `_netdev` – опция, указывающая, что данный ресурс является сетевым(то есть монтироваться каталоги будут

после загрузки сетевой подсистемы).

На клиенте проверьте наличие автоматического монтирования удалённых ресурсов при запуске операционной системы(рис. 3.11):

```
[root@client.eademidova.net ~]# nano /etc/fstab
[root@client.eademidova.net ~]# systemctl status remote-fs.target
● remote-fs.target - Remote File Systems
   Loaded: loaded (/usr/lib/systemd/system/remote-fs.target; enabled; preset:
   Active: active since Sat 2023-12-16 12:08:01 UTC; 44min ago
   Until: Sat 2023-12-16 12:08:01 UTC; 44min ago
   Docs: man:systemd.special(7)

Dec 16 12:08:01 client.eademidova.net systemd[1]: Reached target Remote File Sy
...skipping...
● remote-fs.target - Remote File Systems
   Loaded: loaded (/usr/lib/systemd/system/remote-fs.target; enabled; preset:
   Active: active since Sat 2023-12-16 12:08:01 UTC; 44min ago
   Until: Sat 2023-12-16 12:08:01 UTC; 44min ago
   Docs: man:systemd.special(7)

Dec 16 12:08:01 client.eademidova.net systemd[1]: Reached target Remote File Sy
```

Рис. 3.11: Проверка наличия автоматического монтирования удалённых ресурсов при запуске ОС

Перезапустим клиента и убедимся, что удалённый ресурс подключается автоматически(рис. 3.12):

```
[eademidova@client.eademidova.net ~]$ mount
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=4096k,nr_inodes=114493,mode=755,inode64)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
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)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,size=194916k,nr_inodes=819200,mode=755,inode64)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,seclabel,nsdelegate,memory_recursiveprot)
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)
/dev/sda1 on / type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,nosuid,noexec,relatime)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=29,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,
e_ino=17486)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,seclabel,pagesize=2M)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime,seclabel)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime,seclabel)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime,seclabel)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
none on /run/credentials/systemd-tmpfiles-setup-dev.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
none on /run/credentials/systemd-sysctl.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
none 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)
server.eademidova.net:/srv/nfs on /mnt/nfs type nfs4 (rw,relatime,vers=4.2,rsize=131072,wsz=131072,namlen=255,hard,roto=tcp,timeo=600,retrans=2,sec=sys,clientaddr=192.168.1.30,local_lock=none,addr=192.168.1.1,_netdev)
vagrant on /vagrant type vboxsf (rw,nodev,relatime,ioccharset=utf8,uid=1000,gid=1000,_netdev)
tmpfs on /run/user/1001 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=97456k,nr_inodes=24364,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)
[eademidova@client.eademidova.net ~]$
```

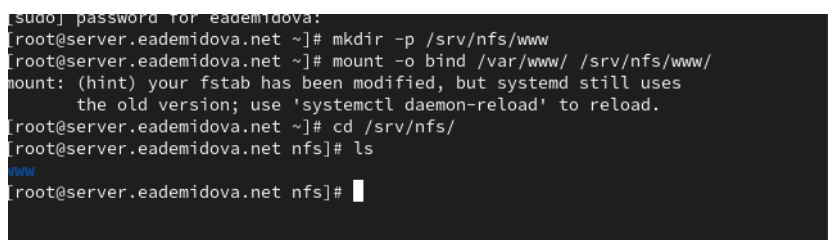
Рис. 3.12: Проверка автоматического подключения удалённого ресурса

4 Подключение каталогов к дереву NFS

На сервере создадим общий каталог, в который затем будет подмонтирован каталог с контентом веб-сервера и подмонтируем каталог web-сервера:

```
mkdir -p /srv/nfs/www  
mount -o bind /var/www/ /srv/nfs/www/
```

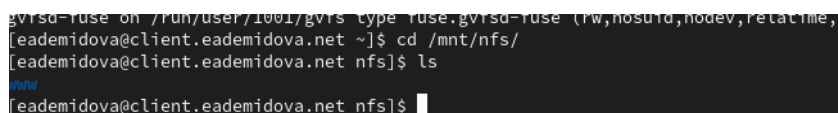
На сервере проверим, что отображается в каталоге /srv/nfs(4.1)



```
sudo] password for eademidova:  
[root@server.eademidova.net ~]# mkdir -p /srv/nfs/www  
[root@server.eademidova.net ~]# mount -o bind /var/www/ /srv/nfs/www/  
mount: (hint) your fstab has been modified, but systemd still uses  
the old version; use 'systemctl daemon-reload' to reload.  
[root@server.eademidova.net ~]# cd /srv/nfs/  
[root@server.eademidova.net nfs]# ls  
www  
[root@server.eademidova.net nfs]#
```

Рис. 4.1: Проверка содержимого /srv/nfs

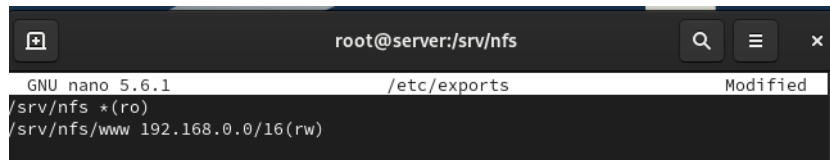
На сервере проверим, что отображается в каталоге /mnt/nfs(4.2)



```
gvfsd-fuse on /run/user/1001/gvfs type fuse.gvfsd-fuse (rw,nosuid,nodev,relatime,  
[eadeimidova@client.eademidova.net ~]$ cd /mnt/nfs/  
[eadeimidova@client.eademidova.net nfs]$ ls  
www  
[eadeimidova@client.eademidova.net nfs]$
```

Рис. 4.2: Проверка содержимого /mnt/nfs

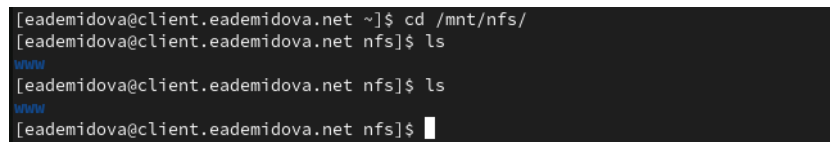
На сервере в файле /etc/exports добавьте экспорт каталога веб-сервера с удалённого ресурса(4.3)



```
root@server:/srv/nfs
GNU nano 5.6.1 /etc/exports Modified
/srv/nfs *(ro)
/srv/nfs/www 192.168.0.0/16(rw)
```

Рис. 4.3: Добавление в файл /etc/exports экспорт каталога веб-сервера

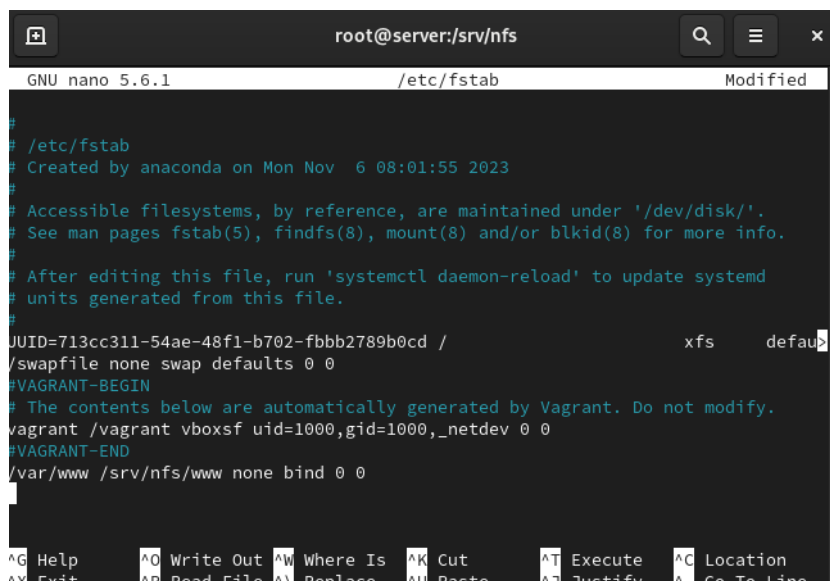
Экспортируем все каталоги, упомянутые в файле /etc/exports командой `exportfs -r`. Проверим на клиенте каталог /mnt/nfs(4.4)



```
[eademidova@client.eademidova.net ~]$ cd /mnt/nfs/
[eademidova@client.eademidova.net nfs]$ ls
www
[eademidova@client.eademidova.net nfs]$ ls
www
[eademidova@client.eademidova.net nfs]$
```

Рис. 4.4: Проверка содержимого /mnt/nfs

На сервере в конце файла /etc/fstab добавим следующую запись(4.5)



```
root@server:/srv/nfs
GNU nano 5.6.1 /etc/fstab Modified
#
# /etc/fstab
# Created by anaconda on Mon Nov  6 08:01:55 2023
#
# 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=713cc311-54ae-48f1-b702-fbbb2789b0cd / 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,_netdev 0 0
#VAGRANT-END
/var/www /srv/nfs/www none bind 0 0
```

Рис. 4.5: Добавление записи в файл /etc/fstab

Повторно экспортируем каталоги, указанные в файле /etc/exports и на клиенте проверим каталог /mnt/nfs(4.6)

```
www
[eademidova@client.eademidova.net nfs]$ cd /mnt/nfs/
[eademidova@client.eademidova.net nfs]$ ls
www
[eademidova@client.eademidova.net nfs]$ █
```

Рис. 4.6: Проверка содержимого /mnt/nfs

5 Подключение каталогов для работы пользователей

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

```
mkdir -p -m 700 ~/common
cd ~/common
touch eademidova@server.txt
```

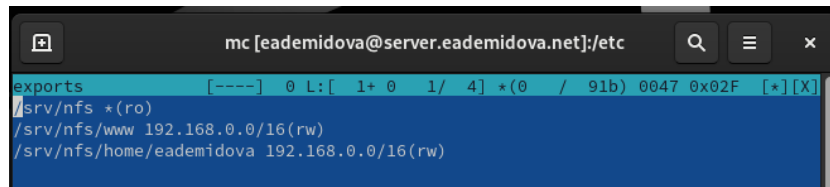
На сервере создадим общий каталог для работы пользователя eademidova по сети, подмонтируем каталог common пользователя eademidova в NFS. Затем посмотрим какие права доступа установлены на этот каталог(5.1):



```
root@server:/srv/nfs/home
[eademidova@server.eademidova.net ~]$ mkdir -p -m 700 ~/common
[eademidova@server.eademidova.net ~]$ cd ~/common
[eademidova@server.eademidova.net common]$ touch eademidova@server.txt
[eademidova@server.eademidova.net common]$ mkdir -p /srv/nfs/home/eademidova
mkdir: cannot create directory '/srv/nfs/home': Permission denied
[eademidova@server.eademidova.net common]$ sudo -i
[root@server.eademidova.net ~]# mount -o bind /home/eademidova/common /srv/nfs/home/eademidova
mount: (hint) your fstab has been modified, but systemd still uses
the old version; use 'systemctl daemon-reload' to reload.
[root@server.eademidova.net ~]# cd /srv/nfs/home/
[root@server.eademidova.net home]# ls -l
total 0
drwx-----. 2 eademidova eademidova 35 Dec 16 16:22 eademidova
[root@server.eademidova.net home]#
```

Рис. 5.1: Проверка прав доступа на каталог

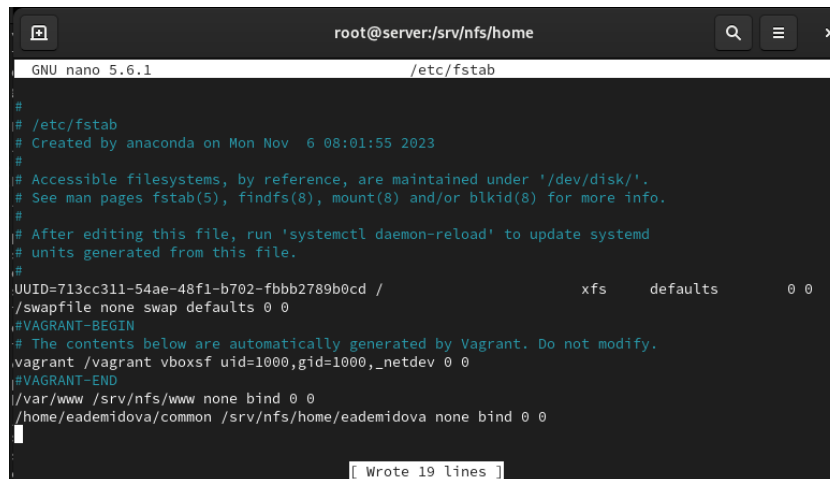
Подключим каталог пользователя в файле /etc/exports, прописав в нём(5.2):



```
mc [eademidova@server.eademidova.net]:/etc
exports  [----]  0 L: [ 1+ 0 1/ 4] *(0 / 91b) 0047 0x02F [*][X]
/srv/nfs *(ro)
/srv/nfs/www 192.168.0.0/16(rw)
/srv/nfs/home/eademidova 192.168.0.0/16(rw)
```

Рис. 5.2: Подключение каталога пользователя в файле /etc/exports

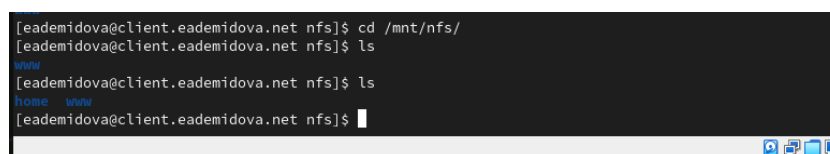
Внесем изменения в файл /etc/fstab(5.3):



```
root@server:/srv/nfs/home
GNU nano 5.6.1 /etc/fstab
#
# /etc/fstab
# Created by anaconda on Mon Nov  6 08:01:55 2023
#
# 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=713cc311-54ae-48f1-b702-fbbb2789b0cd /                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,_netdev 0 0
#VAGRANT-END
/var/www /srv/nfs/www none bind 0 0
/home/eademidova/common /srv/nfs/home/eademidova none bind 0 0
```

Рис. 5.3: Добавление записи в файл /etc/fstab

Затем повторно экспортируем каталоги и проверим на клиенте каталог mnt/nfs(5.4):



```
[eademidova@client.eademidova.net nfs]$ cd /mnt/nfs/
[eademidova@client.eademidova.net nfs]$ ls
www
[eademidova@client.eademidova.net nfs]$ ls
home www
[eademidova@client.eademidova.net nfs]$
```

Рис. 5.4: Проверка содержимого /mnt/nfs

На клиенте попробуем создать файл в каталоге /mnt/nfs/home/eademidova под пользователями eademidova и root (5.5):


```

[root@client.eademidova.net eademidova]# cd /mnt/nfs/
[root@client.eademidova.net nfs]# ls
home  www
[root@client.eademidova.net nfs]# cd home/eademidova/
[root@client.eademidova.net eademidova]# touch eademidova@client.txt
touch: cannot touch 'eademidova@client.txt': Read-only file system
[root@client.eademidova.net eademidova]# exit
logout
[eademidova@client.eademidova.net eademidova]$ touch eademidova@client.txt
touch: cannot touch 'eademidova@client.txt': Read-only file system
[eademidova@client.eademidova.net eademidova]$

```

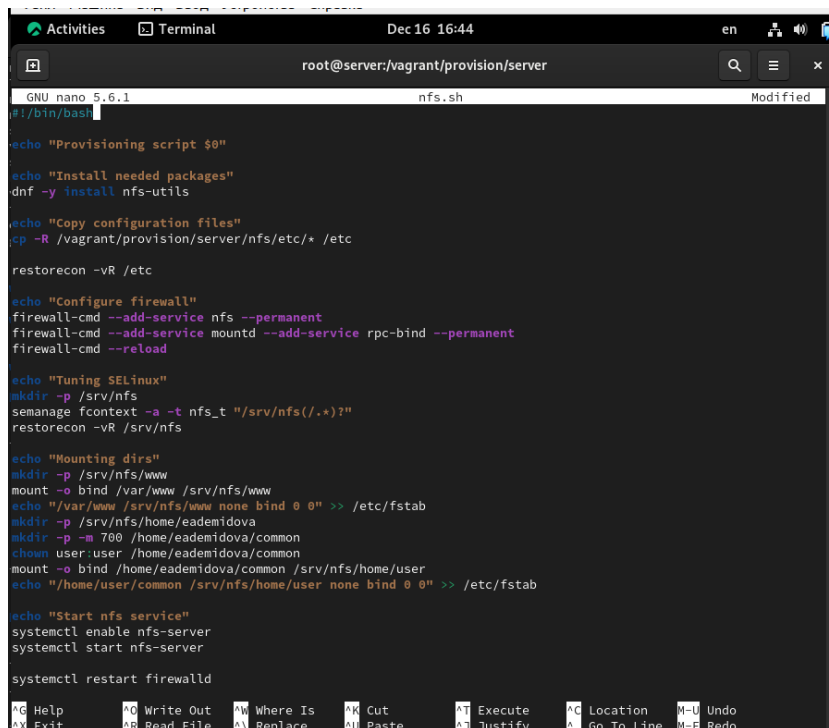
Рис. 5.5: Проверка содержимого /mnt/nfs

Прав на создание файла нет.

5.1 Внесение изменений в настройки внутреннего окружения виртуальных машины

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.

В каталоге `/vagrant/provision/server` создадим исполняемый файл `nfs.sh` и внесем скрипт(5.6):



```
GNU nano 5.6.1 nfs.sh
#!/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/eademidova
mkdir -p -m 700 /home/eademidova/common
chown user:user /home/eademidova/common
mount -o bind /home/eademidova/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
```

Рис. 5.6: Скрипта файла /vagrant/provision/server/nfs.sh

В каталоге /vagrant/provision/client создадим исполняемый файл nfs.sh и внесем скрипт(5.7):



```
GNU nano 5.6.1 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.eademidova.net:/srv/nfs /mnt/nfs
echo "server.eademidova.net:/srv/nfs /mnt/nfs nfs _netdev 0 0" >> /etc/fstab
restorecon -vR /etc
```

Рис. 5.7: Скрипта файла /vagrant/provision/client/nfs.sh

Затем для отработки созданных скриптов 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"
```

6 Контрольные вопросы

1. Как называется файл конфигурации, содержащий общие ресурсы NFS?
2. Какие порты должны быть открыты в брандмауэре, чтобы обеспечить полный доступ к серверу NFS?
3. Какую опцию следует использовать в `/etc/fstab`, чтобы убедиться, что общие ресурсы NFS могут быть установлены автоматически при перезагрузке?
4. Файл конфигурации, содержащий общие ресурсы NFS, называется `/etc/exports`.
5. Для полного доступа к серверу NFS следует открыть TCP и UDP порты 2049 в брандмауэре.
6. Для автоматической установки общих ресурсов NFS при перезагрузке следует использовать опцию `"auto"` в `/etc/fstab`.

7 Выводы

В результате выполнения данной работы были приобретены практические навыки настройки сервера NFS для удалённого доступа к ресурсам.