

Отчёт о лабораторной работе

Лабораторная работа 14

Мошаров Денис Максимович

Содержание

Цель работы

Приобретение навыков настройки доступа групп пользователей к общим ресурсам по протоколу SMB.

Выполнение лабораторной работы

В начале работы мы подключаемся к серверу и выполняем установку необходимых пакетов. С помощью пакетного менеджера dnf мы устанавливаем samba (сам сервер), samba-client (клиентская утилита для проверки) и cifs-utils (утилиты для работы с файловой системой CIFS/SMB) (рис. [-@fig:001]).

```
[dmmosharov@server.dmmosharov.net ~]$ sudo -i
[sudo] password for dmmosharov:
[root@server.dmmosharov.net ~]# dnf -y install samba sam
ba-client cifs-utils
Rocky Linux 10 - BaseOS 1.8 kB/s | 4.3 kB      00:02
Rocky Linux 10 - AppStr 14 kB/s | 4.3 kB      00:00
Rocky Linux 10 - CRB    14 kB/s | 4.3 kB      00:00
Rocky Linux 10 - Extras 11 kB/s | 3.1 kB      00:00
Dependencies resolved.
=====
Package           Arch   Version           Repo              Size
=====
Installing:
cifs-utils         x86_64 7.2-1.el10        baseos            117 k
samba              x86_64 4.22.4-106.el10   baseos            959 k
samba-client       x86_64 4.22.4-106.el10   appstream         770 k
Installing dependencies:
libnetapi          x86_64 4.22.4-106.el10   baseos            144 k
samba-common-tools x86_64 4.22.4-106.el10   baseos            481 k
samba-dcerpc       x86_64 4.22.4-106.el10   baseos            716 k
samba-ldb-ldap-modules
                  x86_64 4.22.4-106.el10   baseos            35 k
samba-libs         x86_64 4.22.4-106.el10   baseos            124 k

Transaction Summary
=====
Install 8 Packages
```

Установка пакетов Samba на сервере

Далее мы переходим к созданию пользователей и групп, которые будут использоваться для доступа к ресурсам. Мы создаем группу `sambagroup` с идентификатором GID 1010 и добавляем в неё нашего текущего пользователя. Также мы создаем директорию `/srv/smbashare`, которая станет общим ресурсом, и открываем файл конфигурации для редактирования (рис. [-@fig:002]).

```
Complete!
[root@server.dmmosharov.net ~]# groupadd -g 1010 sambagroup
[root@server.dmmosharov.net ~]# usermod -aG sambagroup dmmosharov
[root@server.dmmosharov.net ~]# mkdir -p /srv/smbashare
[root@server.dmmosharov.net ~]# nano /etc/samba/smb.conf
```

Создание пользователей, групп и каталога

В открытом файле конфигурации `/etc/samba/smb.conf` мы вносим изменения в раздел `global`, задавая имя рабочей группы `dmmosharov-NET`. В конце файла мы добавляем новую секцию `smbashare`, в которой описываем путь к каталогу, комментарий и список пользователей, имеющих право на запись (рис. [-@fig:003]).

```

[global]
    workgroup = DMMOSHAROV-NET
    security = user

    passdb backend = tdbsam

    printing = cups
    printcap name = cups
    load printers = yes
    cups options = raw

    # Install samba-usershares package for support
    include = /etc/samba/usershares.conf

[homes]
    comment = Home Directories
    valid users = %S, %D%w%S
    browseable = No
    read only = No
    inherit acls = Yes

[printers]
    comment = All Printers
    path = /var/tmp
    printable = Yes
    create mask = 0600
    browseable = No

[print$]
    comment = Printer Drivers
    path = /var/lib/samba/drivers
    # printadmin is a local group
    write list = printadmin root
    force group = printadmin
    create mask = 0664
    directory mask = 0775

[smbashare]
    comment = My Samba Share
    path = /srv/smbashare
    write list = @sambagroup

```

Настройка smb.conf

Мы запускаем утилиту testparm, которая считывает конфиг, подтверждает, что сервисные файлы загружены корректно (Loaded services file OK), и выводит итоговые параметры секций (рис. [-@fig:004]).

```
[root@server.dmmosharov.net ~]# testparm
Load smb config files from /etc/samba/smb.conf
Loaded services file OK.
Weak crypto is allowed by GnuTLS (e.g. NTLM as a compatibility fallback)

Server role: ROLE_STANDALONE

Press enter to see a dump of your service definitions

# Global parameters
[global]
    printcap name = cups
    security = USER
    workgroup = DMMOSHAROV-NET
    idmap config * : backend = tdb
    cups options = raw
    include = /etc/samba/usershares.conf

[homes]
    browseable = No
    comment = Home Directories
    inherit acls = Yes
    read only = No
    valid users = %S %D%w%S

[printers]
    browseable = No
    comment = All Printers
    create mask = 0600
    path = /var/tmp
    printable = Yes

[print$]
    comment = Printer Drivers
    create mask = 0664
    directory mask = 0775
    force group = printadmin
    path = /var/lib/samba/drivers
    write list = printadmin root

[sambashare]
    comment = My Samba Share
    path = /srv/sambashare
    write list = @sambagroup
[root@server.dmmosharov.net ~]#
```

Проверка конфигурации

Убедившись в правильности настроек, мы запускаем службу smb командой `systemctl start`. Затем мы добавляем её в автозагрузку (`enable`), чтобы сервис поднимался после перезагрузки, и проверяем статус, чтобы удостовериться, что служба активна (`active (running)`) (рис. [-@fig:005]).

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

[root@server.dmmosharov.net ~]# systemctl status smb
● smb.service - Samba SMB Daemon
   Loaded: loaded (/usr/lib/systemd/system/smb.servic>
   Active: active (running) since Mon 2026-02-02 12:0>
  Invocation: f90c77d9295e4080a2ba6e817748eded
     Docs: man:smbd(8)
           man:samba(7)
           man:smb.conf(5)
    Main PID: 14754 (smbd)
      Status: "smbd: ready to serve connections..."
        Tasks: 3 (limit: 22836)
       Memory: 13.2M (peak: 13.7M)
          CPU: 58ms
       CGroup: /system.slice/smb.service
               └─14754 /usr/sbin/smbd --foreground --no-p>
                 └─14757 /usr/sbin/smbd --foreground --no-p>
                   └─14758 /usr/sbin/smbd --foreground --no-p>

Feb 02 12:03:33 server.dmmosharov.net systemd[1]: Start>
Feb 02 12:03:33 server.dmmosharov.net systemd[1]: Start>
lines 1-19/19 (END)
```

Запуск службы SMB

Теперь проверим доступность ресурсов локально на сервере. Используя команду `smbclient -L //server`, мы подключаемся (в данном случае анонимно) и видим список доступных ресурсов, среди которых присутствует наш `sambashare`. После этого мы готовимся к настройке файрвола, просматривая расположение XML-файла сервиса (рис. [-@fig:006]).

```
[root@server.dmmosharov.net ~]# smbclient -L //server
Password for [DMMOSHAROV-NET\root]:
Anonymous login successful

      Sharename      Type      Comment
      -----
      print$         Disk      Printer Drivers
      sambashare     Disk      My Samba Share
      IPC$           IPC       IPC Service (Samba 4.2
2.4)
SMB1 disabled -- no workgroup available
[root@server.dmmosharov.net ~]# less /usr/lib/firewalld/
services/samba.xml
```

Проверка доступа через smbclient

Мы открываем файл описания сервиса `/usr/lib/firewalld/services/samba.xml`, чтобы понять, какие порты будут открыты. В файле указано, что сервис предназначен для доступа к файлам и принтерам Windows, и перечислены необходимые порты (TCP 139, 445 и связанные с netbios) (рис. [-@fig:007]).

```
<?xml version="1.0" encoding="utf-8"?>
<service>
  <short>Samba</short>
  <description>This option allows you to access and participate in Windows file and printer sharing networks. You need the samba package installed for this option to be useful.</description>
  <include service="samba-client"/>
  <port protocol="tcp" port="139"/>
  <port protocol="tcp" port="445"/>
</service>
```

Файл конфигурации службы samba.xml

Настраиваем межсетевой экран firewalld: добавляем сервис samba в текущую конфигурацию и в постоянную (–permanent), после чего перезагружаем правила. Также мы меняем группу-владельца каталога /srv/smbashare на sambagroup и устанавливаем полные права для группы. В конце проверяем текущий контекст безопасности SELinux (рис. [-@fig:008]).

```
[root@server.dmmosharov.net ~]# firewall-cmd --add-service=samba
success
[root@server.dmmosharov.net ~]# firewall-cmd --add-service=samba --permanent
success
[root@server.dmmosharov.net ~]# firewall-cmd --reload
success
[root@server.dmmosharov.net ~]# chgrp sambagroup /srv/smbashare
[root@server.dmmosharov.net ~]# chmod g=rwx /srv/smbashare
[root@server.dmmosharov.net ~]# cd /srv
[root@server.dmmosharov.net srv]# ls -Z
unconfined_u:object_r:nfs_t:s0 nfs
unconfined_u:object_r:var_t:s0 sambashare
[root@server.dmmosharov.net srv]#
```

Настройка Firewall и прав доступа

Так как стандартный контекст безопасности не подходит для Samba, мы используем semanage fcontext для назначения типа samba_share_t нашему каталогу. После этого применяем изменения командой restorecon -vR. Повторная проверка ls -Z подтверждает, что контекст успешно изменился на samba_share_t (рис. [-@fig:009]).

```
[root@server.dmmosharov.net srv]# semanage fcontext -a -t samba_share_t "/srv/smbashare(/.*)?"
[root@server.dmmosharov.net srv]# restorecon -vR /srv/smbashare
Relabeled /srv/smbashare from unconfined_u:object_r:var_t:s0 to unconfined_u:object_r:samba_share_t:s0
[root@server.dmmosharov.net srv]# cd /srv
[root@server.dmmosharov.net srv]# ls -Z
unconfined_u:object_r:nfs_t:s0 nfs
unconfined_u:object_r:samba_share_t:s0 sambashare
[root@server.dmmosharov.net srv]#
```

Настройка контекста SELinux

Для корректной работы SELinux с Samba нам также необходимо включить соответствующий переключатель (boolean). Мы выполняем команду setsebool samba_export_all_rw 1 с флагом -P для сохранения настройки. Также проверяем

идентификаторы нашего пользователя, чтобы убедиться, что он состоит в нужной группе (рис. [-@fig:010]).

```
[root@server.dmmosharov.net srv]# setsebool samba_export_all_rw 1
[root@server.dmmosharov.net srv]# setsebool samba_export_all_rw 1 -P
[root@server.dmmosharov.net srv]# id
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@server.dmmosharov.net srv]# logout
[dmmosharov@server.dmmosharov.net ~]$ id
uid=1001(dmmosharov) gid=1001(dmmosharov) groups=1001(dmmosharov),10(wheel) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[dmmosharov@server.dmmosharov.net ~]$
```

Настройка boolean SELinux

Проверяем права на запись в каталог, создавая пустой файл user@server.txt. Затем выполняем важный шаг: добавляем нашего пользователя в базу данных пользователей Samba с помощью команды smbpasswd -a. Мы задаем пароль, который будет использоваться при сетевом подключении к ресурсу (рис. [-@fig:011]).

```
[dmmosharov@server.dmmosharov.net ~]$ cd /srv/sambashare
[dmmosharov@server.dmmosharov.net sambashare]$ touch user@server.txt
touch: cannot touch 'user@server.txt': Permission denied
[dmmosharov@server.dmmosharov.net sambashare]$ sudo touch user@server.txt
[sudo] password for dmmosharov:
[dmmosharov@server.dmmosharov.net sambashare]$ smbpasswd -L -a dmmosharov
smbpasswd -L can only be used by root.
[dmmosharov@server.dmmosharov.net sambashare]$ sudo smbpasswd -L -a dmmosharov
[sudo] password for dmmosharov:
New SMB password:
Retype new SMB password:
Added user dmmosharov.
[dmmosharov@server.dmmosharov.net sambashare]$
```

Создание тестового файла и SMB-пользователя

Настройка сервера завершена, переходим к клиентской машине. Первым делом устанавливаем пакеты, необходимые для подключения к общим ресурсам: samba-client и cifs-utils. Менеджер пакетов разрешает зависимости и успешно устанавливает программное обеспечение (рис. [-@fig:012]).

```
[dmmosharov@client.dmmosharov.net ~]$ sudo dnf -y install samba
-client cifs-utils
[sudo] password for dmmsosharov:
Last metadata expiration check: 0:14:34 ago on Mon 02 Feb 2026
12:22:01 PM UTC.
Dependencies resolved.
=====
Package            Arch      Version              Repository    Size
=====
Installing:
cifs-utils          x86_64    7.2-1.el10           baseos        117 k
samba-client        x86_64    4.22.4-106.el10      appstream     770 k
=====

Transaction Summary
=====
Install 2 Packages

Total download size: 887 k
Installed size: 3.0 M
Downloading Packages:
(1/2): samba-client-4.22.4-106 305 kB/s | 770 kB      00:02
(2/2): cifs-utils-7.2-1.el10.x  22 kB/s | 117 kB      00:05
-----
Total                               81 kB/s | 887 kB      00:10
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      :                                1/1
  Installing     : samba-client-4.22.4-106.el10.x86_64 1/2
  Running scriptlet: samba-client-4.22.4-106.el10.x86_64 1/2
  Installing     : cifs-utils-7.2-1.el10.x86_64        2/2
  Running scriptlet: cifs-utils-7.2-1.el10.x86_64        2/2

Installed:
cifs-utils-7.2-1.el10.x86_64
samba-client-4.22.4-106.el10.x86_64

Complete!
[dmmosharov@client.dmmosharov.net ~]$ less /usr/lib/firewalld/s
ervices/samba-client.xml
```

Установка пакетов на клиенте

Аналогично серверу, мы изучаем конфигурацию файрвола для клиента. Просматриваем файл `/usr/lib/firewalld/services/samba-client.xml`. Это позволяет убедиться, что мы открываем правильные порты (в частности, UDP 137 и 138) для работы клиентской части протокола NetBIOS (рис. [-@fig:013]).

```
<?xml version="1.0" encoding="utf-8"?>
<service>
  <short>Samba Client</short>
  <description>This option allows you to access Windows file an
d printer sharing networks. You need the samba-client package i
nstalled for this option to be useful.</description>
  <include service="netbios-ns"/>
  <port protocol="udp" port="138"/>
</service>
~
```

Просмотр samba-client.xml

Настраиваем межсетевой экран на клиенте, разрешая сервис `samba-client` и перезагружая правила. Затем повторяем настройку окружения: создаем группу `sambagroup` с тем же GID 1010 и добавляем в неё пользователя. Открываем конфиг Samba для правки (рис. [-@fig:014]).

```
[root@client.dmmosharov.net ~]# firewall-cmd --add-service=samba-client
success
[root@client.dmmosharov.net ~]# firewall-cmd --add-service=samba-client --permanent
success
[root@client.dmmosharov.net ~]# firewall-cmd --reload
success
[root@client.dmmosharov.net ~]# groupadd -g 1010 sambagroup
[root@client.dmmosharov.net ~]# usermod -aG sambagroup dmmosharov
[root@client.dmmosharov.net ~]# nano /etc/samba/smb.conf
```

Настройка окружения на клиенте

В файле /etc/samba/smb.conf на клиенте нам нужно привести рабочую группу в соответствие с сервером. Мы меняем значение параметра workgroup на dmmosharov-NET. Остальные параметры можно оставить по умолчанию для базового подключения (рис. [-@fig:015]).

```
GNU nano 8.1 /etc/samba/smb.conf
# See smb.conf.example for a more detailed config file or
# read the smb.conf manpage.
# Run 'testparm' to verify the config is correct after
# you modified it.
#
# Note:
# SMB1 is disabled by default. This means clients without support
# SMB3 are no longer able to connect to smbd (by default).

[global]
    workgroup = DMMOSHAROV-NET
    security = user

    passdb backend = tdbsam

    printing = cups
    printcap name = cups
    load printers = yes
    cups options = raw
```

Настройка рабочей группы на клиенте

Проверяем сетевую доступность ресурсов сервера. Команда smbclient -L //server запрашивает пароль и выводит список ресурсов удаленной машины. Мы видим, что ресурс sambashare доступен, что говорит о правильной настройке сети и Samba на обеих машинах (рис. [-@fig:016]).

```
[dmmosharov@client.dmmosharov.net ~]$ smbclient -L //server -U dmmosharov
Password for [DMMOSHAROV-NET\dmmosharov]:

      Sharename      Type      Comment
      -----
      print$         Disk      Printer Drivers
      sambashare     Disk      My Samba Share
      IPC$           IPC       IPC Service (Samba 4.22.4)
      dmmosharov     Disk      Home Directories
SMB1 disabled -- no workgroup available
[dmmosharov@client.dmmosharov.net ~]$
```

Проверка списка ресурсов сервера с клиента

Для проверки возможности монтирования создаем точку /mnt/samba. Сначала пробуем смонтировать ресурс вручную командой mount, передавая параметры пользователя и прав. После успешного монтирования создаем файл dmmosharov@client.txt в удаленной папке, чтобы подтвердить права на запись, а затем размонтируем ресурс (рис. [-@fig:017]).

```
[dmmosharov@client.dmmosharov.net ~]$ sudo mkdir /mnt/samba
[sudo] password for dmmosharov:
Sorry, try again.
[sudo] password for dmmosharov:
[dmmosharov@client.dmmosharov.net ~]$ mount -o username=dmmosha
rov,user,rw,uid=dmmosharov,gid=sambagroup //server/sambashare /
mnt/samba
This program is not installed setuid root - "user" CIFS mounts
not supported.
mount: (hint) your fstab has been modified, but systemd still u
ses
the old version; use 'systemctl daemon-reload' to reload
.
[dmmosharov@client.dmmosharov.net ~]$ sudo mount -o username=dm
mosharov,user,rw,uid=dmmosharov,gid=sambagroup //server/sambash
are /mnt/samba
Password for dmmosharov@//server/sambashare:
mount: (hint) your fstab has been modified, but systemd still u
ses
the old version; use 'systemctl daemon-reload' to reload
.
[dmmosharov@client.dmmosharov.net ~]$ cd /mnt/samba
[dmmosharov@client.dmmosharov.net samba]$ touch dmmosharov@clie
nt.txt
[dmmosharov@client.dmmosharov.net samba]$ ls
dmmosharov@client.txt  user@server.txt
[dmmosharov@client.dmmosharov.net samba]$ ls
dmmosharov@client.txt  dmmosharov@server.txt
[dmmosharov@client.dmmosharov.net samba]$ cd ..
[dmmosharov@client.dmmosharov.net mnt]$ umount /mnt/samba
umount: /mnt/samba: must be superuser to unmount.
[dmmosharov@client.dmmosharov.net mnt]$ sudo umount /mnt/samba
[dmmosharov@client.dmmosharov.net mnt]$
```

Ручное монтирование и проверка записи

Чтобы настроить автоматическое монтирование и не хранить пароль в открытом виде в fstab, мы создаем файл с учетными данными /etc/samba/smbusers. Устанавливаем на него права доступа 600 (чтение и запись только для root) и открываем его для заполнения (рис. [-@fig:018]).

```
[dmmosharov@client.dmmosharov.net samba]$ sudo touch /etc/samba
/smbusers
[dmmosharov@client.dmmosharov.net samba]$ chmod 600 /etc/samba/
smbusers
chmod: changing permissions of '/etc/samba/smbusers': Operation
not permitted
[dmmosharov@client.dmmosharov.net samba]$ sudo chmod 600 /etc/s
amba/smbusers

[dmmosharov@client.dmmosharov.net samba]$ sudo nano /etc/samba/
smbusers
[dmmosharov@client.dmmosharov.net samba]$
```

Создание файла учетных данных

В файл smbusers мы вписываем две строки: имя пользователя (username) и пароль (password), который мы задали ранее через smbpasswd. Это позволит системе автоматически подставлять эти данные при монтировании ресурса (рис. [-@fig:019]).

```
dmmosharov@client:/etc/samba – sudo nano /etc/samba/sm...
GNU nano 8.1 /etc/samba/smbusers
username=dmmosharov
password=123456
```

Содержимое файла smbusers

Теперь редактируем файл `/etc/fstab`. Мы добавляем строку, описывающую наш сетевой ресурс: указываем источник `//server/smbashare`, точку монтирования, тип файловой системы `cifs`. В опциях указываем путь к файлу с паролем (`credentials=...`), права `rw` и опцию `_netdev`, чтобы монтирование происходило только после поднятия сети (рис. [-@fig:020]).

```
GNU nano 8.1 /etc/fstab

/etc/fstab
Created by anaconda on Tue Nov 25 17:14:02 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=d3e7b147-c1d6-424e-819d-7a2628a132ec / xfs defaults 0 0
UUID=86c00fec-Gd70-4d3e-9d8f-62dca78372f5 /boot xfs defaults 0 0
UUID=3080-501E /boot/efi vfat umask=0077,shortname=winnt 0 2
UUID=e7ad08e1-c163-42a6-8723-3d23736bc2c9 /home xfs defaults 0 0
UUID=84e6e6ba-1981-4517-8c07-547bd3127c94 none swap defaults 0 0
server.dmmosharov.net:/srv/nfs /mnt/nfs nfs _netdev 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/smbashare /mnt/samba cifs user,rw,uid=dmmosharov,gid=sambagroup,credentials=/etc/samba/smbuser,
```

Редктирование fstab

Проверяем корректность записи в `fstab`, выполняя команду `mount -a`, которая монтирует все файловые системы. Ошибок не возникло. Просматриваем содержимое каталога `/mnt/samba` и видим оба файла — созданный на сервере и созданный ранее клиентом (рис. [-@fig:021]).

```
[dmmosharov@client.dmmosharov.net samba]$ sudo nano /etc/fstab
[dmmosharov@client.dmmosharov.net samba]$ sudo mount -a
[dmmosharov@client.dmmosharov.net samba]$ ls samba/
ls: cannot access 'samba/': No such file or directory
[dmmosharov@client.dmmosharov.net samba]$ cd ..
[dmmosharov@client.dmmosharov.net mnt]$ ls samba/
dmmosharov@client.txt dmmosharov@server.txt
[dmmosharov@client.dmmosharov.net mnt]$
```

Проверка автоматического монтирования

Убеждаемся в успешном автоматическом монтировании после перезагрузки. Вывод команды `ls` подтверждает, что в общем ресурсе находятся файлы `dmmosharov@client.txt` и `dmmosharov@server.txt`. Это окончательно подтверждает работоспособность двустороннего обмена данными (рис. [-@fig:022]).

```
Password:
Last login: Mon Feb  2 12:36:15 UTC 2026 on pts/0
[dmmosharov@client.dmmosharov.net ~]$ cd /mnt/samba
[dmmosharov@client.dmmosharov.net samba]$ cd ..
[dmmosharov@client.dmmosharov.net mnt]$ ls samba/
dmmosharov@client.txt dmmosharov@server.txt
[dmmosharov@client.dmmosharov.net mnt]$
```

Список файлов в общем ресурсе

Теперь автоматизируем процесс настройки через `Vagrant`. На сервере переходим в каталог `/vagrant/provision/server`, создаем структуру папок для хранения конфигурационных файлов и копируем туда наш рабочий `smb.conf`. Также создаем файл скрипта `smb.sh` и делаем его исполняемым (рис. [-@fig:023]).

```
[dmmosharov@server.dmmosharov.net sambashare]$ sudo touch dmmosharov@server.txt
[dmmosharov@server.dmmosharov.net sambashare]$ cd /vagrant/provision/server
[dmmosharov@server.dmmosharov.net server]$ mkdir -p /vagrant/provision/server/smb/etc/samba
[dmmosharov@server.dmmosharov.net server]$ cp -R /etc/samba/smb.conf /vagrant/provision/server/smb/etc/samba/
[dmmosharov@server.dmmosharov.net server]$ cd /vagrant/provision/server
[dmmosharov@server.dmmosharov.net server]$ touch smb.sh
[dmmosharov@server.dmmosharov.net server]$ sudo chmod +x smb.sh
[sudo] password for dmmosharov:
[dmmosharov@server.dmmosharov.net server]$ nano smb.sh
```

Подготовка каталогов provision на сервере

В скрипт smb.sh для сервера мы записываем последовательность команд, которую выполняли вручную: установка пакетов, копирование конфига, настройка файрвола, создание пользователей и групп с заданием пароля через переменные, настройка прав доступа к папке и параметров SELinux (рис. [-@fig:024]).

```
GNU nano 8.1 smb.sh
#!/bin/bash
LOGIN=dmmosharov
PASS=123456

echo "Provisioning script $0"
echo "Install needed packages"
dnf -y install samba samba-client cifs-utils
echo "Copy configuration files"
cp -R /vagrant/provision/server/smb/etc/* /etc
chown -R root:root /etc/samba/*
restorecon -vR /etc
echo "Configure firewall"
firewall-cmd --add-service samba --permanent
firewall-cmd --reload
echo "Users and groups"
groupadd -g 1010 sambagroup
usermod -aG sambagroup $LOGIN
echo -ne "$PASS\n$PASS\n" | smbpasswd -L -a -s $LOGIN
echo "Make share dir"
mkdir -p /srv/sambashare
chgrp sambagroup /srv/sambashare
chmod g-rwx /srv/sambashare
echo "Tuning SELinux"
semanage fcontext -a -t samba_share_t "/srv/sambashare(/.*)?"
setsebool samba_export_all_rw 1
setsebool samba_export_all_rw 1 -P
restorecon -vR /srv/sambashare
echo "Start smb service"
systemctl enable smb
systemctl start smb
systemctl restart firewalld
```

Скрипт smb.sh для сервера

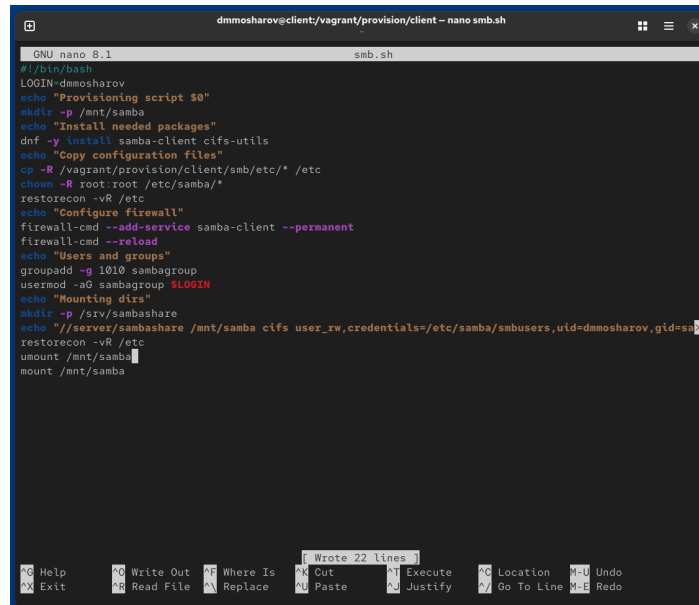
Аналогичные действия выполняем для клиента. В каталоге /vagrant/provision/client создаем структуру папок, копируем туда smb.conf и файл с паролями smbusers. Создаем скрипт smb.sh и даем ему права на выполнение (рис. [-@fig:025]).

```
[dmmosharov@client.dmmosharov.net mnt]$ cd /vagrant/provision/client
[dmmosharov@client.dmmosharov.net client]$ mkdir -p /vagrant/provision/client/smb/etc/samba
[dmmosharov@client.dmmosharov.net client]$ cp -R /etc/samba/smb.conf /vagrant/provision/client/smb/etc/samba/
[dmmosharov@client.dmmosharov.net client]$ cp -R /etc/samba/smbusers /vagrant/provision/client/smb/etc/samba/
cp: cannot open '/etc/samba/smbusers' for reading: Permission denied
[dmmosharov@client.dmmosharov.net client]$ sudo cp -R /etc/samba/smbusers /vagrant/provision/client/smb/etc/samba/
[sudo] password for dmmosharov:
[dmmosharov@client.dmmosharov.net client]$ cd /vagrant/provision/client
[dmmosharov@client.dmmosharov.net client]$ touch smb.sh
[dmmosharov@client.dmmosharov.net client]$ chmod +x smb.sh
[dmmosharov@client.dmmosharov.net client]$ nano smb.sh
```

Подготовка каталогов provision на клиенте

Скрипт smb.sh для клиента содержит команды установки пакетов, копирования

конфигурационных файлов в /etc/samba/ и /etc/, настройки прав доступа к файлу паролей (restorecon), настройки файрвола, создания локальной группы и добавления записи в /etc/fstab для автоматического монтирования (рис. [-@fig:026]).



```
dmmosharov@client:~/vagrant/provision/client - nano smb.sh
GNU nano 8.1 smb.sh
#!/bin/bash
LOGIN=dmmosharov
echo "Provisioning script $0"
mkdir -p /mnt/samba
echo "Install needed packages"
dnf -y install samba-client cifs-utils
echo "Copy configuration files"
cp -R /vagrant/provision/client/smb/etc/* /etc
chown -R root:root /etc/samba/*
restorecon -vR /etc
echo "Configure firewall"
firewall-cmd --add-service=samba-client --permanent
firewall-cmd --reload
echo "Users and groups"
groupadd -g 1010 sambagroup
usermod -s0 sambagroup SLOGIN
echo "Mounting dirs"
mkdir -p /srv/smbashare
echo "server/smbashare /mnt/samba cifs user_rw,credentials=/etc/samba/smbusers,uid=dmmosharov,gid=sambagroup
restorecon -vR /etc
umount /mnt/samba
mount /mnt/samba
```

Скрипт smb.sh для клиента

В завершение лабораторной работы мы редактируем основной Vagrantfile. Добавляем инструкции server.vm.provision и client.vm.provision типа “shell”, указывая пути к нашим скриптам smb.sh. Это обеспечит автоматическую настройку Samba при следующем развертывании виртуальных машин (рис. [-@fig:027]).

```

128     server.vm.provision "SMB server",
129     |         type: "shell",
130     |         preserve_order: true,
131     |         path: "provision/server/smb.sh"
132 end
133
134
135 ## Client configuration
136 config.vm.define "client", autostart: false do |client|
137     client.vm.box = "rockylinux10"
138     client.vm.hostname = 'client'
139
140     client.vm.boot_timeout = 1440
141
142     client.ssh.insert_key = false
143     client.ssh.username = 'vagrant'
144     client.ssh.password = 'vagrant'
145
146     client.vm.network :private_network,
147     |         type: "dhcp",
148     |         virtualbox__intnet: true
149
150     client.vm.provider :virtualbox do |virtualbox|
151         virtualbox.customize ["modifyvm", :id, "--vrde", "on"]
152         virtualbox.customize ["modifyvm", :id, "--vrdeport", "3392"]
153     end
154
155     client.vm.provision "client dummy",
156     |         type: "shell",
157     |         preserve_order: true,
158     |         path: "provision/client/01-dummy.sh"
159
160     client.vm.provision "client routing",
161     |         type: "shell",
162     |         preserve_order: true,
163     |         run: "always",
164     |         path: "provision/client/01-routing.sh"
165     client.vm.provision "client mail",
166     |         type: "shell",
167     |         preserve_order: true,
168     |         path: "provision/client/mail.sh"
169     client.vm.provision "client ntp",
170     |         type: "shell",
171     |         preserve_order: true,
172     |         path: "provision/client/ntp.sh"
173
174     client.vm.provision "client nfs",
175     |         type: "shell",
176     |         preserve_order: true,
177     |         path: "provision/client/nfs.sh"
178
179     client.vm.provision "SMB client",
180     |         type: "shell",
181     |         preserve_order: true,
182     |         path: "provision/client/smb.sh"
183 end
184 end

```

Обновленный Vagrantfile

Выводы

В результате выполнения лабораторной работы были получены навыки настройки и использования Samba