Глава 16

NFS, защищенная с помощью Kerberos

ЦЕЛИ СЕРТИФИКАЦИИ

16.01 Сервер сетевой файловой системы (NFS)
16.02. Тестирование клиента NFS
16.03 NFS с Kerberos
✓ Двухминутная тренировка
Q & A Самопроверка

Linux предназначен для работы в сети. Он позволяет вам обмениваться файлами двумя основными способами: **Samba**, описанная в **главе 15**, и Сетевая файловая система (**Network File System NFS**). RHEL 7 не включает инструменты **GUI** для **NFS**, но это не проблема, поскольку файлы конфигурации **NFS** относительно просты.

Эта глава начинается с описания **NFS**, мощного и универсального способа обмена данными между серверами и рабочими станциями. Установка по умолчанию **RHEL 7** включает в себя **клиент NFS**, который поддерживает соединения с серверами **NFS**.

Сервер **NFS** может ограничивать доступ к клиентам на основе их **имен хостов или IP-адресов**. Кроме того, **NFS** доверяет **UID**, отправленным клиентами, для проверки прав доступа к файлам. Это обеспечивает только базовый уровень безопасности, который может быть недостаточным для некоторых организаций. Но при использовании совместно с **Kerberos NFS** может проверять подлинность доступа к сетевым ресурсам и обеспечивать шифрование данных. В этой главе объясняется, как настроить такую конфигурацию.

Потратьте время на то, чтобы разобраться в файлах конфигурации, связанных со **службой NFS** и **Kerberos**, и попрактиковаться в том, чтобы заставить их работать на компьютере с **Linux**. В некоторых случаях два или три компьютера (например, виртуальные машины **KVM**, описанные в главах 1 и 2) под управлением **Linux** могут помочь вам практиковать уроки этой главы.

Внутри экзаменов

Внутри экзаменов

Как показанные здесь, цели RHCE для **NFS** в основном те же, что и для **Samba**. Конечно, то, что вы делаете для настройки **NFS**, отличается.

- Предоставление сетевых ресурсов определенным клиентам
- Предоставление сетевых ресурсов, подходящих для совместной работы в группах.

Процесс ограничения доступа **NFS** к определенным клиентам является простым. Кроме того, способ настройки групповой совместной работы для общего сетевого ресурса **NFS** основан на методах, которые мы уже обсуждали в **главе 8**.

Интеграция между **NFS** и **Kerberos** является новым требованием для **экзамена RHCE** по **RHEL** 7. Цель состоит в том, чтобы

- Использование **Kerberos** для управления доступом к сетевым ресурсам **NFS**.

Кроме того, вы настроите **firewalld** и **SELinux** для работы с **NFS**.

ЦЕЛЬ СЕРТИФИКАЦИИ 16.01

Сервер сетевой файловой системы (NFS)

NFS является стандартом для обмена файлами с компьютерами **Linux** и **Unix**. Первоначально он был разработан **Sun Microsystems** в середине 1980-х годов. **Linux** поддерживает **NFS** (как клиент и

сервер) в течение многих лет, и **NFS** продолжает пользоваться популярностью в организациях с сетями на основе **Unix** или **Linux**.

Вы можете создать общие ресурсы **NFS**, отредактировав файл конфигурации **/etc/exports** или создав новый файл в каталоге **/etc/exports.d**. Таким образом, вы можете настроить **NFS** для основной работы. Для настройки более сложных конфигураций может быть полезно понять, как работает **NFS** и как она взаимодействует по сети.

Вы можете повысить безопасность NFS несколькими способами, включая следующие:

- Правильно настроенный брандмауэр
- TCP Wrappers
- SELinux
- Kerberos аутентификация и шифрование

Опции NFS для RHEL 7

Хотя **NFS версии 4 (NFSv4)** используется по умолчанию, **RHEL 7** также поддерживает **NFS 3 (NFSv3)**. Различия между **NFSv3 и NFSv4** включают способ взаимодействия клиентов и серверов, максимальные размеры файлов и поддержку списков контроля доступа в стиле **Windows (ACL).**

Если вы используете **NFSv4**, вам не нужно устанавливать связь удаленного вызова процедур (**Remote Procedure Call RPC**) с пакетом **rpcbind**. Тем не менее, **RPC** требуется для **NFSv3**.

В **NFSv3** появилась поддержка 64-битных размеров файлов для обработки файлов размером более 2 ГБ. **NFSv4** расширяет **NFSv3** и обеспечивает несколько улучшений производительности. Он также поддерживает лучшую безопасность благодаря интеграции с **Kerberos**. В то время как **NFSv3** использует отдельный протокол для блокировки файлов, известный как «**NLM» (Network Lock Manager)**, **NFSv4** изначально включает блокировку файлов.

!!!!! On the job !!!!
NFS версии 4.1 поддерживает кластеризованное развертывание через расширение pNFS (параллельный NFS). pNFS позволяет масштабировать NFS, распределяя данные по нескольким серверам и получая эти данные параллельно от клиентов. Для получения дополнительной информации посетите веб-сайты http://www.pnfs.org и https://github.com/nfs-ganesha/nfs-ganesha.
!!!!!!!!!!!!

Базовая установка NFS

Основной группой, связанной с программным обеспечением **NFS**, является группа «Сервер файлов и хранилищ (**File and Storage Server**)». Другими словами, если вы выполните следующую команду, **yum** установит обязательные пакеты из этой группы:

 # yum group install "File and Storage Server"

Однако в эту группу также входят пакеты для поддержки целевых объектов **Samba**, **CIFS** и **iSCSI**. Единственный пакет, необходимый для настройки **NFS**-сервера или клиента, - это **nfs-utils**:

 # yum -y install nfs-utils

Вы можете установить дополнительные пакеты, в том числе следующие:

- **nfs4-acl-tools** Предоставляет утилиты командной строки для получения и редактирования списков доступа в общих папках **NFS**.
- **portreserve** Поддерживает службу **portreserve**, преемника **portmap** для связи **NFS**. Препятствует **NFS** принимать порты, необходимые для других служб.
- **quota** Обеспечивает поддержку квот для общих каталогов **NFS**.
- **rpcbind** Включает поддержку связи **RPC** для различных каналов **NFS**.

Базовая конфигурация сервера NFS

NFS-серверы относительно просты в настройке. Все, что вам нужно сделать, это экспортировать файловую систему и затем смонтировать эту файловую систему с удаленного клиента.

Конечно, это предполагает, что вы открыли правильные порты в брандмауэре и изменили соответствующие параметры **SELinux**. **NFS** контролируется рядом системных сервисных модулей. Он также поставляется с широким спектром команд управления.

Службы NFS

После установки соответствующих пакетов они контролируются несколькими различными **units** обслуживания **systemd**:

- **nfs-server.service** Сервисный блок для **NFS-сервера**; ссылается на **/etc/sysconfig/nfs** для базовой конфигурации.
- **nfs-secure-server.service** Запускает демон **rpc.svcgssd**, который обеспечивает поддержку аутентификации и шифрования **Kerberos** для сервера **NFS**.
- **nfs-secure.service** Запускает демон **rpc.gssd**, который согласовывает аутентификацию и шифрование **Kerberos** между клиентом и сервером **NFS**.
- **nfs-idmap.service** Запускает демон **rpc.idmapd**, который переводит идентификаторы пользователей и групп в имена. Автоматически запускается модулем **nfs-server systemd**.
- **nfs-lock.service** Требуется **NFSv3**. Запускает демон **rpc.statd**, который предоставляет блокировки и статус для файлов, используемых в данный момент.
- **nfs-mountd.service** Запускает демон монтирования **NFS rpc.mountd**. Требуется **NFSv3.**
- **nfs-rquotad.service** Запускает демон **rpc.rquotad**, который предоставляет службы квот файловой системы для общих ресурсов **NFS**. Автоматически запускается модулем **nfs-server systemd**.
- **rpcbind.service** Запускает демон **rpcbind**, который преобразует номера программ **RPC** в адреса. Используется **NFSv3**. Автоматически запускается модулем **nfs-server systemd**.

Чтобы запустить **NFS-сервер**, вам не нужно запоминать все перечисленные **сервисные units**. Учитывая зависимости по умолчанию между сервисными модулями, все, что вам нужно сделать, это запустить следующие команды на компьютере с **NFS-сервером**:

```
# systemctl start nfs-server
# systemctl enable nfs-server
```

Чтобы включить поддержку **Kerberos** для **NFS**, вам также необходимо активировать службы **nfs-secure-server** и **nfs-secure** на сервере и клиентских компьютерах соответственно. Это будет рассмотрено более подробно в следующих разделах.

Команды и файлы управления NFS

NFS включает в себя широкий спектр команд для настройки экспорта, чтобы показать, что доступно, посмотреть, что смонтировано, просмотреть статистику и многое другое. За исключением специализированных команд монтирования, эти команды можно найти в каталоге **/usr/sbin**.

Команды монтирования **NFS - это mount.nfs** и **umount.nfs**. Это две символические ссылки: **mount.nfs4** и **umount.nfs4**. Функционально они работают как обычные команды **mount** и **umount**. Как предложено расширениями, они применяются к файловым системам, используемым совместно с **NFSv4** и другими версиями **NFS**. Как и другие команды **mount.*,** Они имеют функциональные эквиваленты. Например, команда **mount.nfs4** функционально эквивалентна команде **mount -t nfs4**.

Если вы монтируете общий ресурс с помощью команд **mount.nfs** и **mount -t nfs**, **NFS** пытается подключить общий ресурс с помощью **NFSv4** и возвращается к **NFSv3**, если версия 4 не поддерживается сервером.

Пакеты, связанные с **NFS**, содержат значительное количество команд в каталоге **/usr/sbin**. Список команд, показанный здесь, является наиболее часто используемым для настройки и тестирования **NFS**:

- **exportfs** Команда **exportfs** может использоваться для управления каталогами, которые используются совместно и настроены в файле **/etc/exports**.
- **nfsiostat** Команда статистики для скоростей ввода/вывода на основе существующей точки монтирования. Использует информацию из файла **/proc/self/mountstats**.
- **nfsstat** Команда статистики для активности **клиент/сервер** на основе существующей точки монтирования. Использует информацию из файла **/proc/self/mountstats**.

- **showmount** Команда, наиболее тесно связанная с отображением общих каталогов **NFS**, локально и удаленно.

Вы можете использовать связанные с **ACL** команды из **RPM-пакета nfs4-acl-tools**. Вы можете запускать эти команды для файловых систем, смонтированных локально, с помощью опции **acl**, как описано в **главе 6**. Сами команды просты, так как они устанавливают (**nfs4_setfacl**), редактируют (**nfs4_editfacl**) и выводят список (**nfs4_getfacl**) текущих **ACL** указанных файлов. Хотя эти команды выходят за рамки основной работы **NFS**, они кратко обсуждаются здесь и в **главе 4**.

Предположим, что вы смонтировали каталог **/home** с опцией **acl**. Вы поделились этим каталогом через **NFS**. Когда вы применяете команду **nfs4_getfacl** к файлу в этом общем каталоге, вы можете увидеть следующий вывод:

A::OWNER@:rwatTcCy
A::GROUP@:tcy
A::EVERYONE@:tcy

ACL-списки имеют разрешение «Разрешить **Allow (A)**» или «Запретить **Deny (D)**» для владельца файла (**OWNER, GROUP** или **EVERYONE**). Следующие разрешения более детальны, чем обычные разрешения **rwx**. Например, для представления разрешений на запись в **Linux** списки **ACL** включают разрешения как на запись (**w**), так и на добавление (**a**).

Возможно, самый простой способ изменить эти **ACL**-списки - использовать команду **nfs4_setfacl -e *filename***, которая позволяет редактировать текущие разрешения в текстовом редакторе. Например, чтобы редактировать файл **ACL** на общем ресурсе, смонтированном через **NFSv4** из удаленной системы, выполните следующую команду:

$ nfs4_setfacl -e /tmp/michael/filename.txt

Эта команда открывает указанные списки **ACL NFSv4** в текстовом редакторе по умолчанию для пользователя (обычно vi). Когда мы удалили разрешения на добавление для владельца файла, а затем сохранили изменения, это действие фактически удалило разрешения на добавление и запись для файла. Чтобы просмотреть результат, снова запустите команду **nfs4_getfacl**:

D::OWNER@:wa
A::OWNER@:rtTcCy
A::GROUP@:rwatcy
A::EVERYONE@:rtcy

Если вы попробуете команду **ls -l** для того же файла, вы заметите, что у владельца файла больше нет прав на запись.

Настройте NFS для основной работы

Файл конфигурации общего ресурса **NFS**, **/etc/exports**, довольно прост. После настройки вы можете экспортировать каталоги, настроенные в этом файле, с помощью команды **exportfs -a**.

Каждая строка в **/etc/exports** содержит список экспортируемого каталога, хосты, в которые он будет экспортироваться, и параметры, которые применяются к этому экспорту. Хотя вы можете установить несколько условий, вы можете экспортировать определенный каталог только один раз. Возьмите следующие примеры из файла **/etc/exports**:

/pub tester1.example.com(rw,sync) *(ro,sync)
/home *.example.com(rw,async) 172.16.10.0/24(ro)
/tftp nodisk.example.net(rw,no_root_squash,sync)

В этом примере каталог **/pub** экспортируется в клиент **tester1.example.com** с разрешениями на **чтение/запись**. Он также экспортируется всем остальным клиентам с правами только для чтения. Каталог **/home** экспортируется с разрешениями на **чтение/запись** всем клиентам в сеть ***.example.com** и доступна только для чтения клиентам в **подсети 172.16.10.0/24**. Наконец, каталог **/tftp** экспортируется с полными разрешениями на **чтение/запись** (даже для пользователей root) на компьютер **nodisk.example.net**.

Хотя эти параметры довольно просты, файл **/etc/exports** несколько требователен. Пробел в конце строки может привести к синтаксической ошибке. Пробел между именем хоста и условиями в скобках откроет доступ ко всем хостам.

Все эти параметры включают флаг синхронизации. Это требует фиксации операций записи на диск перед возвратом статуса клиенту. До **NFSv4** многие такие опции включали небезопасный флаг, который разрешает доступ через порты выше **1024**. Дополнительные параметры будут обсуждаться в следующих разделах.

Вы также можете разделить конфигурацию **NFS** на несколько файлов с расширением **.exports** в **каталоге /etc/exports.d**. Например, вы можете взять три строки конфигурации в предыдущем файле **/etc/exports** и переместить их в отдельные файлы с именами **pub.exports, home.exports** и **tftp.exports** в каталоге **/etc/exports.d.**

!!!!! Examen watch !!!!!!
Будьте осторожны с файлом /etc/exports. Например, дополнительный пробел после любой запятой в (ro, no_root_squash, sync) означает, что указанный каталог не будет экспортирован.
!!!!!!

Wildcard и Globbing

В файлах конфигурации сети **Linux** вы можете указать группу компьютеров с правильным подстановочным знаком, который в **Linux** также известен как глобализация. Что можно использовать в качестве подстановочного знака, зависит от файла конфигурации. В файле **NFS /etc/exports** используются «**обычные**» символы подстановки: например, ***.example.net** указывает все компьютеры в домене **example.net**. Напротив, в файле **/etc/hosts.deny .example.net** с начальной точкой указывает все компьютеры в этом же домене.

В сетях **IPv4** подстановочные знаки часто указывают неявную маску подсети. Например, **192.168.0.*** Эквивалентен **192.168.0.0/255.255.255.0**, который указывает сеть компьютеров **192.168.0.0** с **IP-адресами** в диапазоне от **192.168.0.1 до 192.168.0.254**. Некоторые службы, включая **NFS**, поддерживают использование нотации **CIDR** (бесклассовой междоменной маршрутизации). В **CIDR**, поскольку маска **255.255.255.0 это 24 бита**, **CIDR** представляет это **числом 24**. При настройке сети в нотации **CIDR** эту сеть можно представить как **192.168.0.0/24.**

Дополнительные параметры NFS-сервера

С **/etc/exports** можно использовать несколько различных параметров. Параметры, описанные в **таблицах 16-1 и 16-2**, делятся на две категории: общие и параметры безопасности.

ТАБЛИЦА 16-1 NFS /etc/exports Общие параметры

Параметр	Описание
async	Операции записи выполняются асинхронно. Обеспечивает лучшую пропускную способность, рискуя потерять данные в случае сбоя сервера **NFS**.
hide	Скрывает файловые системы; если вы экспортируете каталог и подкаталог, такие как **/mnt** и **/mnt/inst**, общие ресурсы в **/mnt/inst** должны быть явно подключены.
mp	Экспортирует каталог, только если он был успешно смонтирован; требует, чтобы точка экспорта также была точкой монтирования на сервере.
ro	Экспортирует том только для чтения.
rw	Экспортирует том чтения-записи.
sync	Передает операции записи на диск, прежде чем ответить клиенту. Активен по умолчанию.

Другие параметры относятся к настройкам безопасности общих каталогов **NFS**. Как показано в **Таблице 16-2**, параметры связаны с пользователем root-администратора, анонимными пользователями и аутентификацией **Kerberos**.

Активировать список экспорта

После того, как вы настроите файл **/etc/exports**, сделайте эти каталоги доступными для клиентов с помощью команды **exportfs -a**. При следующей загрузке **RHEL 7**, если активируются нужные службы, системный модуль **nfs-server** выполняет команду **exportfs -r**, которая повторно экспортирует каталоги, настроенные в **/etc/exports**.

Однако, если вы изменяете, перемещаете или удаляете общие ресурсы **NFS**, сначала вы должны временно отменить экспорт всех каталогов с помощью команды **exportfs -ua**. Вы можете сделать желаемое изменения, а затем экспортируйте общие ресурсы с помощью команды **exportfs -a** или **exportfs -r**. Разница между **-a и -r** незначительна, но важна: тогда как **-a** экспортирует (или не экспортирует в сочетании с **-u**) все каталоги, **-r** повторно экспортирует все каталоги, синхронизируя список общих ресурсов и удаляя те, которые имеют был удален из файла конфигурации **/etc/exports**.

Когда экспорт активен, вы можете просмотреть его состояние с помощью команды **showmount -e servername**. Например, команда **showmount -e server1.example.com** ищет список экспортированных каталогов **NFS** из системы **server1.example.com**. Если эта команда не выполнена, связь может быть заблокирована брандмауэром.

ТАБЛИЦА 16-2 Параметры безопасности NFS /etc/exports

Параметр	Описание
all_squash	Сопоставляет все локальные и удаленные учетные записи анонимному пользователю.
anongid=groupid	Указывает идентификатор группы для учетной записи анонимного пользователя.
anonuid=userid	Указывает идентификатор пользователя для учетной записи анонимного пользователя.
insecure	Поддерживает связь через порт 1024, в первую очередь для NFS версий 2 и 3.
no_root_squash	Рассматривает удаленного пользователя **root** как локального **root**; если этот параметр не задан, по умолчанию пользователь **root** будет сопоставлен с пользователем **nfsnobody**.
sec=value	Задает список параметров безопасности, разделенных двоеточиями. Значением по умолчанию является **sys**, которое предписывает серверу **NFS** полагаться на **UID/GID** для доступа к файлу. Значения, связанные с **Kerberos: krb5, krb5i** и **krb5p**.

Исправлены порты в /etc/sysconfig/nfs

NFSv4 проще в настройке, особенно в отношении брандмауэров. Чтобы включить связь с сервером **NFSv4**, единственный порт, который вам нужно открыть, это **TCP-порт 2049**. Этот порт является частью службы **nfs** в **firewalld**, поэтому вы должны выполнить следующие команды на сервере **NFS**, чтобы разрешить входящие соединения:

```
# firewall-cmd --permanent --add-service=nfs
# firewall-cmd --reload
```

Хотя по умолчанию используется **NFSv4, RHEL 7** также поддерживает **NFSv3**. Поэтому, учитывая общедоступную информацию об экзамене **RHCE**, вам также может понадобиться знать, как обращаться с этой версией **NFS**. **NFSv3** использует динамические номера портов через службу **RPC**, которая прослушивает **UDP-порт 111** и связана со службой **rpc-bind firewalld**. Вам также необходимо предоставить доступ к службе **mountd**, поэтому в целом вам необходимо разрешить следующим службам поддерживать **NFSv3** через **firewalld**:

```
# firewall-cmd --permanent --add-service=nfs --add-service=rpc-bind --add-service=mountd
# firewall-cmd --reload
```

После запуска службы **NFS** с помощью команды **systemctl start nfs-server** в случае успеха вы увидите соответствующие порты в выходных данных команды **rpcinfo**, в которой перечислены все каналы связи, связанные с **RPC**. Следующая команда является более точной, поскольку она изолирует фактические номера портов:

```
# rpcinfo -p
```

Пример вывода показан на **рисунке 16-1**. На первый взгляд линии могут показаться повторяющимися; однако у каждой строки есть цель. Если не запущена другая связанная с **RPC** служба, такая как служба сетевой информации (**NIS**), все показанные здесь линии необходимы для обмена данными по **NFS.** Изучите первую строку, показанную здесь:

Program	**vers**	**proto**	**port**	**service**
100000	**4**	**tcp**	**111**	**portmapper**

РИСУНОК 16-1 Пример вывода rpcinfo -p с портами, связанными с NFS

```
[root@server1 ~]# rpcinfo -p
   program vers proto   port  service
    100000    4   tcp    111  portmapper
    100000    3   tcp    111  portmapper
    100000    2   tcp    111  portmapper
    100000    4   udp    111  portmapper
    100000    3   udp    111  portmapper
    100000    2   udp    111  portmapper
    100024    1   udp  35364  status
    100024    1   tcp  50967  status
    100005    1   udp  20048  mountd
    100005    1   tcp  20048  mountd
    100005    2   udp  20048  mountd
    100005    2   tcp  20048  mountd
    100005    3   udp  20048  mountd
    100005    3   tcp  20048  mountd
    100003    3   tcp   2049  nfs
    100003    4   tcp   2049  nfs
    100227    3   tcp   2049  nfs_acl
    100003    3   udp   2049  nfs
    100003    4   udp   2049  nfs
    100227    3   udp   2049  nfs_acl
    100021    1   udp  41077  nlockmgr
    100021    3   udp  41077  nlockmgr
    100021    4   udp  41077  nlockmgr
    100021    1   tcp  46344  nlockmgr
    100021    3   tcp  46344  nlockmgr
    100021    4   tcp  46344  nlockmgr
    100011    1   udp    875  rquotad
    100011    2   udp    875  rquotad
    100011    1   tcp    875  rquotad
    100011    2   tcp    875  rquotad
[root@server1 ~]#
```

Первая строка представляет произвольный номер программы **RPC**, версию **NFS** и использование **TCP** в качестве протокола связи через порт **111** со службой **portmapper**. Обратите внимание на доступность службы **portmapper** для **NFS** версий **2, 3 и 4,** взаимодействующих по протоколам **TCP** и **UDP**.

Связь через выбранные порты также должна быть разрешена через любой настроенный брандмауэр. Например, на **рисунке 16-2** показана, конфигурация firewalld которая поддерживает удаленный доступ к локальному **NFS-серверу** через протоколы версий **3** и **4.**

Вы можете установить эти правила брандмауэра с помощью графической утилиты **firewall-config**, описанной в **главе 4**.

Заставить NFS работать с SELinux

Конечно, вам нужно настроить больше, чем брандмауэр. **SELinux** является неотъемлемой частью ландшафта безопасности, как в отношении логических параметров, так и файлов. Во-первых, обратите внимание на следующие типы файлов **NFS SELinux**:

- **nfs_t** Связывается с общими папками **NFS**, которые экспортируются только для чтения или для чтения и записи.
- **public_content_ro_t** Связанный с общими ресурсами NFS, которые экспортируются только для чтения.

- **public_content_rw_t** Связан с общими папками **NFS**, которые экспортируются для чтения и записи. Требуется установить логическое значение **nfsd_anon_write.**
- **var_lib_nfs_t** Связан с динамическими файлами в каталоге **/var/lib/nfs**. Файлы в этом каталоге обновляются по мере того, как общие ресурсы экспортируются и монтируются клиентами.
- **nfsd_exec_t** Назначается системным исполняемым файлам, таким как **rpc.mountd** и **rpc.nfsd** в каталоге **/usr/sbin**. Тесно связаны типы файлов **rpcd_exec_t** и **gssd_exec_t** для служб, связанных с **RPC** и связями с серверами **Kerberos**.

РИСУНОК 16-2 Правила брандмауэра для NFS

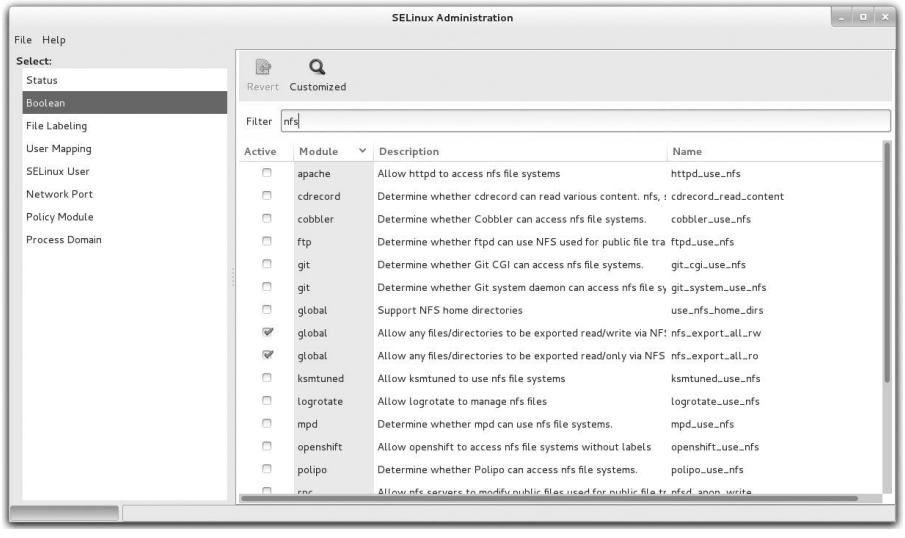
```
[root@server1 ~]# firewall-cmd --permanent --add-service=nfs \
> --add-service=rpc-bind --add-service=mountd
success
[root@server1 ~]# firewall-cmd --reload
success
[root@server1 ~]# firewall-cmd --list-all
public (default, active)
  interfaces: eth0
  sources:
  services: dhcpv6-client mountd nfs rpc-bind ssh
  ports:
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:

[root@server1 ~]#
```

В общем случае вам не нужно присваивать новый тип файла общему каталогу **NFS**. Фактически, типы файлов **SELinux**, которые связаны с общими файлами (**nfs_t, public_content_ro_t** и **public_content_rw_t**), действуют только тогда, когда логические значения **nfs_exports_all_ro** и **nfs_exports_all_rw** отключены. Таким образом, для большинства администраторов эти типы файлов показаны для справки.

Для **SELinux** логические директивы являются наиболее важными. Опции показаны в разделе **Booleans** инструмента администрирования **SELinux** с фильтром **nfs**, как показано на **рисунке 16-3**. На рисунке показана конфигурация по умолчанию; как видите, две опции в глобальном модуле включены по умолчанию.

Следующие директивы связаны с работой **NFS** с **SELinux** в целевом режиме. Хотя большинство из этих вариантов уже перечислены в **главе 10**, их стоит повторить, хотя бы для того, чтобы помочь тем, кто боится **SELinux**. Опции описаны в порядке, показанном на рисунке.

РИСУНОК 16-3. Логические параметры SELinux, связанные с NFS



- **httpd_use_nfs** Поддерживает доступ веб-сервера **Apache** к общим ресурсам **NFS**.

- **cdrecord_read_content** Включает доступ к смонтированным общим ресурсам **NFS** с помощью команды **cdrecord**.
- **cobbler_use_nfs** Позволяет **Cobbler** получить доступ к файловым системам **NFS**.
- **ftpd_use_nfs** Позволяет использовать общие каталоги **NFS** серверами **FTP**.
- **git_cgi_use_nfs** Поддерживает доступ к общим ресурсам **NFS** с помощью службы системы контроля версий **git** в сценариях **CGI**.
- **git_system_use_nfs** Поддерживает доступ к общим ресурсам **NFS** службой системы контроля версий **git**.
- **use_nfs_home_dirs** Включает монтирование **/home** с удаленного **NFS-сервера.**
- **nfs_export_all_rw** Поддерживает доступ на чтение и запись к общим каталогам **NFS**.
- **nfs_export_all_ro** Поддерживает доступ только для чтения к общим каталогам **NFS**.
- **ksmtuned_use_nfs** Позволяет **ksmtuned** получить доступ к общим ресурсам **NFS**.
- **logrotate_use_nfs** Позволяет **logrotate** получить доступ к файлам **NFS**.
- **mpd_use_nfs** Позволяет демону музыкального проигрывателя получать доступ к содержимому из общих папок **NFS**.
- **openshift_use_nfs** Позволяет **OpenShift** получать доступ к файловым системам **NFS**.
- **polipo_use_nfs** Разрешает доступ через **веб-прокси Polipo** к файловым системам, смонтированным в **NFS**.
- **nfsd_anon_write** Позволяет серверам **NFS** изменять публичные файлы. Файлы должны быть помечены с типом **public_content_rw_t**.
- **samba_share_nfs** Позволяет **Samba** экспортировать файловые системы, смонтированные в **NFS**.
- **sanlock_use_nfs** Включает демон менеджера блокировки **SANlock** для доступа к файлам **NFS**.
- **sge_use_nfs** Позволяет **Sun Grid Engine** получать доступ к файлам **NFS**.
- **virt_use_nfs** Включает доступ виртуальных машин к файловым системам, смонтированным в **NFS**.
- **virt_sandbox_use_nfs** Позволяет контейнерам песочницы получать доступ к файловым системам **NFS**.
- **xen_use_nfs** Разрешает доступ гипервизора **Xen** к файловым системам, смонтированным в **NFS**.

Чтобы установить эти директивы, используйте команду **setsebool**. Например, чтобы активировать доступ к файловым системам **NFS** через **FTP-сервер** таким образом, чтобы он выдержал перезагрузку, выполните следующую команду:

 # setsebool -P ftpd_use_nfs 1

Причуды и ограничения NFS

У **NFS** есть свои ограничения. Любой администратор, который контролирует общие каталоги **NFS**, будет разумно принять к сведению эти ограничения.

Без сохранения состояния

NFSv3 - это протокол без сохранения состояния. Другими словами, вам не нужно входить отдельно, чтобы получить доступ к общему каталогу **NFS**. Вместо этого клиент **NFS** обычно связывается с сервером **rpc.mountd**. Демон **rpc.mountd** обрабатывает запросы монтирования. Он проверяет запрос к текущим экспортируемым файловым системам. Если запрос действителен, **rpc.mountd** предоставляет дескриптор файла **NFS** («волшебный файл cookie»), который затем используется для дальнейшего обмена данными между клиентом и сервером для этого общего ресурса.

Протокол без сохранения состояния позволяет клиенту **NFS** ожидать, если когда-либо потребуется перезагрузить сервер **NFS**. Программное обеспечение ждет, и ждет, и ждет. Это может привести к зависанию клиента **NFS**. Клиент может даже перезагрузить или даже выключить и выключить систему.

Это также может привести к проблемам с небезопасными однопользовательскими клиентами. Когда файл открывается через общий ресурс, он может быть «заблокирован» от других пользователей. Когда **NFS-сервер** перезагружается, обработка заблокированного файла может быть затруднена.

Изменения, которые привели к разработке **NFSv4**, представили протокол с отслеживанием состояния, чтобы сделать механизм блокировки более надежным, и должны помочь решить эту проблему.

Root Squash (Запретить доступ с правами root)

По умолчанию для **NFS** задано значение **root_squash**, что не позволяет корневым пользователям на клиенте **NFS** получить **root-доступ** к общему ресурсу на сервере **NFS**. В частности, пользователь **root** на клиенте (с идентификатором пользователя 0) сопоставляется с непривилегированной учетной записью **nfsnobody** (в случае сомнений проверьте локальную учетную запись в файле **/etc/passwd**).

Это поведение можно отключить с помощью опции экспорта сервера **no_root_squash** в **/etc/exports**. Для экспортированных каталогов с параметром **no_root_squash** удаленные корневые пользователи могут использовать свои корневые привилегии в общем каталоге **NFS**. Хотя это может быть полезно, это также риск для безопасности, особенно от хакеров «черной шляпы», которые используют свои собственные системы **Linux**, чтобы воспользоваться этими привилегиями **root**.

NFS зависает

Поскольку NFSv3 работает без сохранения состояния, клиенты **NFS** могут ждать до нескольких минут на сервере. В некоторых случаях клиент **NFS** может ждать неограниченное время, если сервер выходит из строя. Во время ожидания любой процесс, который ищет файл на смонтированном общем ресурсе **NFS**, будет зависать. Как только это происходит, обычно сложно размонтировать поврежденные файловые системы, если вы не передадите опцию «**lazy**» команде **umount (umount -l).** Это может по-прежнему оставлять некоторые процессы в непрерывном спящем состоянии, ожидая ввода-вывода. Вы можете сделать несколько вещей, чтобы уменьшить влияние этой проблемы:

- Будьте внимательны, чтобы обеспечить надежность серверов **NFS** и сети.
- Монтируйте редко используемые экспорты **NFS** только при необходимости. Клиенты **NFS** должны размонтировать эти ресурсы после использования.
- Не используйте асинхронную настройку и настройте общие папки **NFS** с параметром синхронизации (по умолчанию), который должен по крайней мере снизить вероятность потери данных.
- Держите **NFS**-установленные каталоги вне пути поиска пользователей, особенно корневого.
- Держите **NFS**-установленные каталоги вне корневого (/) каталога; вместо этого разделите их на менее часто используемые, если это возможно, на отдельные разделы.

Обратные указатели DNS

Демон **NFS-сервера** проверяет запросы на монтирование. Сначала он просматривает текущий список экспорта, основанный на **/etc/exports**. Затем он ищет IP-адрес клиента, чтобы найти его имя хоста. Это требует обратного просмотра **DNS**.

Затем это имя хоста наконец проверяется по списку экспорта. Если **NFS** не может найти имя хоста, **rpc.mountd** запретит доступ этому клиенту. По соображениям безопасности он также добавляет запись «запрос от неизвестного хоста» в **/var/log/messages**.

Блокировка файлов

Можно настроить несколько клиентов **NFS** для монтирования одного и того же экспортированного каталога с одного и того же сервера. Вполне возможно, что люди на разных компьютерах в конечном итоге пытаются использовать один и тот же общий файл. Это решается службой демона блокировки файлов.

Хотя обязательные блокировки поддерживаются **NFSv4**, в **NFS** исторически существовали серьезные проблемы с блокировками файлов. Если у вас есть приложение, которое зависит от блокировки файлов через **NFS**, тщательно протестируйте его, прежде чем запускать в работу.

Кроме того, вы никогда не должны совместно использовать один и тот же каталог с **NFS** и **Samba** одновременно, поскольку различные механизмы блокировки, используемые этими службами, могут привести к повреждению данных.

Советы по производительности

Вы делаете несколько шагов, чтобы **NFS** работала стабильно и надежно. По мере приобретения опыта работы с **NFS** вы можете отслеживать или даже экспериментировать со следующими факторами:

- Восемь процессов **NFS**, которые используются по умолчанию, как правило, достаточны для хорошей производительности даже при довольно высоких нагрузках. Чтобы увеличить пропускную способность службы, вы можете добавить больше процессов **NFS** через директиву

RPCNFSDCOUNT в файле конфигурации **/etc/sysconfig/nfs**. Просто имейте в виду, что дополнительные процессы потребляют дополнительные системные ресурсы.

- Производительность записи NFS может быть низкой. В приложениях, где потеря данных не является большой проблемой, вы можете попробовать асинхронную опцию. Это делает **NFS** быстрее, потому что сервер немедленно возвращает клиенту состояние операции записи, не дожидаясь записи данных на диск. Однако потеря питания или сетевого подключения может привести к потере данных.
- Поиск имени хоста часто выполняется сервером **NFS**; Вы можете запустить демон переключения имен (**NSCD**) для ускорения поиска.

Директивы безопасности NFS

NFS включает в себя ряд потенциальных проблем безопасности и никогда не должен использоваться в неблагоприятных средах (например, на сервере, напрямую подключенном к Интернету), по крайней мере, без строгих мер предосторожности.

Недостатки и риски

NFS - это простая в использовании, но мощная система обмена файлами. Однако это не без его ограничений. Ниже приведены несколько вопросов безопасности, которые следует иметь в виду:

- Аутентификация **NFS** полагается на хост, чтобы сообщать идентификаторы пользователей и групп. Тем не мение, это может быть угрозой безопасности, если пользователи **root** на других компьютерах получают доступ к общим ресурсам **NFS**. Другими словами, данные, которые доступны через **NFS** любому пользователю, потенциально могут быть доступ к любому другому пользователю. Этот риск устраняется с помощью **NFSv4**, если **Kerberos** используется для аутентификация.
- Конфиденциальность До **NFSv4 NFS** не поддерживала шифрование. **NFSv4** с поддержкой **Kerberos** может обеспечить шифрованную связь.
- Инфраструктура **rpcbind** И клиент, и сервер **NFSv3** зависят от демона **RPC portmap**. Более ранние версии демона исторически имели ряд серьезных дыр в безопасности. По этой причине **RHEL 7** заменил его на службу **rpcbind**.

Советы по безопасности

Если **NFS** необходимо использовать в агрессивной среде или рядом с ней, вы можете уменьшить риски безопасности:

Узнайте больше о безопасности **NFS**. Если возможно, настройте зашифрованные соединения **NFSv4** с помощью **Kerberos**. В противном случае ограничьте **NFS** дружественными внутренними сетями, защищенными хорошим межсетевым экраном.
Экспортируйте как можно меньше данных и, если возможно, экспортируйте файловые системы только для чтения.
За исключением случаев, когда это абсолютно необходимо, не заменяйте параметр **root_squash**. В противном случае хакеры «черной шляпы» на разрешенных клиентах могут предоставить доступ корневого уровня к экспортированным файловым системам.
Используйте соответствующие параметры брандмауэра, чтобы запретить доступ к портам **portmapper** и **nfsd**, кроме как с явно доверенных хостов или сетей. Если вы используете **NFSv4**, достаточно открыть через порт **nfs firewalld** только следующий порт:

2049 TCP nfsd (server)

Варианты безопасности на основе хоста

Чтобы проверить, безопасность на основе хоста в системах **NFS** основана главным образом на системах, которым разрешен доступ к общему ресурсу в файле **/etc/exports**. Конечно, безопасность на основе хоста может также включать ограничения, основанные на правилах брандмауэра.

Параметры для безопасности пользователя

Поскольку монтирование **NFS** должно отражать безопасность, связанную с общей базой данных пользователей, должны применяться стандартные параметры безопасности на основе пользователя. Это включает в себя настройку общей группы, как описано в **главе 8**.

!!!!!! Exam watch !!!!!
Пока существует общая пользовательская база данных, такая как LDAP, разрешения, связанные с общим каталогом группы, переносятся на монтирование, совместно используемое через NFS.
!!!!!!!

УПРАЖНЕНИЕ 16-1

NFS

Для этого упражнения требуется две системы: одна настроена как **NFS-сервер**, другая - как **NFS-клиент**. Затем на сервере **NFS** выполните следующие действия:

1. Настройте группу с именем IT для группы информационных технологий в **/etc/group**.
2. Создайте каталог **/MIS**. Назначьте владение группой **MIS** с помощью команды **chgrp**.
3. Установите бит **SGID** в этом каталоге, чтобы обеспечить владение группой.
4. Убедитесь, что **RPM-пакет nfs-utils** установлен.
5. На сервере запустите и включите службу **NFS** для запуска при загрузке:

   # systemctl start nfs-server
   # systemctl enable nfs-server

6. Обновите файл **/etc/exports**, чтобы разрешить чтение и запись для общего ресурса в локальной сети. Выполните следующую команду, чтобы применить изменения:

   # exportfs -a

7. Убедитесь, что **логические значения SELinux** установлены правильно; в частности, убедитесь, что оба логических значения **nfs_export_all_ro** и **nfs_export_all_rw** включены. Это значение по умолчанию. Вы можете сделать это либо с помощью команды **getsebool**, либо с помощью инструмента управления **SELinux**.
8. Откройте необходимые порты на брандмауэре. Для **NFSv4** требуются следующие команды:

   # firewall-cmd --permanent --add-service=nfs
   # firewall-cmd --reload

   Затем **на клиенте NFS** выполните следующие действия:

9. Убедитесь, что **RPM-пакет nfs-utils установлен**.
10. Создайте каталог для общего ресурса сервера с именем **/mnt/MIS**.
11. Смонтируйте общий каталог **NFS** в **/mnt/MIS**.
12. Перечислите все экспортированные общие ресурсы с сервера и сохраните эти выходные данные в файле **shares.list** в каталоге **/mnt/MIS**.
13. Сделайте это постоянным монтированием в файле **/etc/fstab**. Предположим, что соединение может быть проблематичным, и добавьте соответствующие параметры, такие как мягкое крепление.
14. Запустите команду **mount -a**, чтобы перечитать **/etc/fstab**. Проверьте, правильно ли перемонтирована шара.
15. Проверьте соединение **NFS**. Остановите службу **NFS** на сервере и попробуйте скопировать файл в каталог **/mnt/MIS**. Хотя попытка копирования не удастся, клиент не должен зависать.
16. Перезапустите **службу NFS** на сервере.
17. Отредактируйте **/etc/fstab** снова. На этот раз предположим, что **NFS** надежен, и удалите специальные параметры, добавленные в шаге 13.
18. Теперь выключите сервер и проверьте, что происходит. При подключении к службе подключенный каталог **NFS** на клиенте должен зависать.

19. Клиентский компьютер может заблокироваться. Если это так, вы можете загрузить **rescue target**, как описано **в главе 5**, чтобы избежать проблем с перезагрузкой. Восстановите исходную конфигурацию.

ЦЕЛЬ СЕРТИФИКАЦИИ 16.02

Проверьте клиент NFS

Теперь вы можете смонтировать общий каталог **NFS** с клиентского компьютера. Команды и файлы конфигурации аналогичны тем, которые используются для любой локальной файловой системы. В предыдущем разделе вы настроили сервер NFS. Пока оставайтесь в системе NFS-сервера, поскольку первый клиентский тест можно запустить непосредственно с этого компьютера.

Параметры монтирования NFS

Прежде чем делать что-либо сложное, вы должны проверить список общих каталогов **NFS**. Затем вы можете смонтировать общий том **NFS** из второй системы **Linux**, предположительно из системы **RHEL 7 (или эквивалентной)**. Для этого команда **showmount** отображает доступные общие тома.

Запустите команду **showmount** с параметром **-e**; в сочетании с **именем хоста или IP-адресом** сервера NFS команда отображает список экспорта, возможно, включая ограничения хоста для общего ресурса. Например, при наличии простого общего доступа к каталогам **/mnt** и **/home** на данном **NFS-сервере** команда **showmount -e server1.example.com** обеспечивает следующий результат:

Export list for server1.example.com:
/mnt 192.168.100.0/24
/home 192.168.122.0/24

Если вы не видите список общих каталогов, войдите в систему сервера **NFS**. Повторите команду **showmount**, заменив **localhost** или **127.0.0.1** именем хоста или IP-адресом. Если по-прежнему нет вывода, просмотрите шаги, описанные ранее в этой главе. Убедитесь, что файл **/etc/exports** настроен правильно. Не забудьте экспортировать общие каталоги. Используйте команду

 # systemctl status nfs-server

чтобы убедиться, что службы работают.

Теперь, чтобы монтировать этот каталог локально, вам понадобится пустой локальный каталог. Создайте каталог, такой как **/remotemnt** или другой. Затем вы можете смонтировать общий каталог из системы, такой как **192.168.122.50**, с помощью следующей команды:

 # mount.nfs 192.168.122.50:/share/remotemnt

Эта команда монтирует каталог **NFS / share** с компьютера **по указанному IP-адресу**. При желании вы можете заменить команду **mount -t nfs** на **mount.nfs**. Когда это сработает, вы сможете получить доступ к файлам из каталога **remote/share**, как если бы это был локальный каталог. Если локальное подключение работает, а удаленное - нет, проверьте настройки брандмауэра и убедитесь, что служба работает.

Configure NFS in /etc/fstab

Вы также можете настроить клиент **NFS** для монтирования удаленного каталога **NFS** во время процесса загрузки, как определено в **/etc/fstab**. Например, следующая запись в клиенте **/etc/fstab** монтирует общий ресурс **/homenfs** с компьютера с именем **nfsserv** в локальный каталог **/nfs/home**, используя версию протокола по умолчанию 4:
nfsserv:/homenfs /nfs/home nfs soft, timeo = 100 0 0

Опции **soft** и **timeo** - это две специальные опции монтирования **NFS**. Такие параметры, как показано здесь, также можно использовать для настройки способа монтирования во время процесса загрузки в файле **/etc/fstab**.

Подумайте об использовании опции **soft** при монтировании файловых систем **NFS**. При отказе сервера **NFS** файловая система NFS с мягким монтированием скорее выйдет из строя, а не зависнет. Однако это может привести к повреждению данных в случае временного отключения сети. Используйте эту опцию, только если отзывчивость клиента важнее целостности данных. Кроме того, вы можете использовать опцию **timeo**, чтобы установить интервал ожидания в десятых долях секунды.

Для получения дополнительной информации об этих и связанных параметрах см. Справочную страницу **nfs**, доступную с помощью команды **man nfs**.

Кроме того, автомонтирование может использоваться для динамического монтирования файловых систем **NFS** в соответствии с требованиями клиентского компьютера. Автомонтировщик также может отключить эти удаленные файловые системы после периода бездействия. Для получения дополнительной информации о управляющей службе **autofs** см. **Главу 6**.

!!!!EXAM Watch !!!!!
Специфичные для NFS параметры команды mount, которые также можно использовать в /etc/fstab можно найти на справочной странице nfs.
!!!!!!!

Бездисковые клиенты

NFS поддерживает бездисковых клиентов, которые являются компьютерами, которые не хранят операционную систему локально. Бездисковый клиент может использовать чип флэш-памяти для начала работы. Затем встроенные команды могут смонтировать соответствующий корневой каталог (/), настроить пространство подкачки, установить каталог **/usr** только для чтения и настроить другие общие каталоги, такие как **/home**, в режиме **чтения/записи**. Если ваш компьютер настроен как бездисковый клиент, вам также потребуется доступ к серверам **DHCP** и **TFTP** для загрузки системы с сетевого загрузочного сервера.

Red Hat Enterprise Linux включает в себя функции, которые поддерживают бездисковые клиенты. Хотя они не включены в текущие требования к экзаменам Red Hat или в соответствующие схемы курсов, мы не удивимся, увидев такие требования в будущем.

Текущий статус NFS

Текущее состояние служб **NFS** задокументировано в двух каталогах: **/var/lib/nfs** и **/proc/fs/nfsd**. Если есть проблема с **NFS**, просмотрите некоторые файлы в этих каталогах. Просматривайте эти каталоги по одному. Во-первых, в каталоге **/var/lib/nfs** есть два ключевых файла:

- **etab** Включает полное описание экспортированных каталогов, включая параметры по умолчанию
- **rmtab** Определяет состояние общих каталогов, которые в данный момент смонтированы.

Посмотрите на содержимое каталога **/proc/fs/nfsd**. Поскольку это виртуальный каталог, файлы в дереве каталогов **/proc** имеют нулевой размер. Однако в качестве динамических файлов они могут содержать ценную информацию. Возможно, ключевым параметром для базовой операции является файл **/proc/fs/nfsd/version**. Содержимое этого файла указывает признанные в настоящее время версии **NFS**.

Обычное содержимое этого файла немного загадочно, что говорит о том, что текущий сервер **NFS** может обмениваться данными с использованием **NFSv3, NFSv4** и **NFSv4.1**, но не с **NFSv4.2** и **NFSv2**:

-2 +3 +4 +4.1 -4.2

Если вы установите параметр **RPCNFSDARGS ="- V 4.2"** в файле /**etc/sysconfig/nfs** и перезапустите службу **NFS**, содержимое файла версий изменится на

-2 +3 +4 +4.1 +4.2

Разница тонкая, но важная. Фактически, **NFSv4.2** предоставляет экспериментальную функцию, которая позволяет вам сохранять исходный контекст **SELinux** каждого файла в общем каталоге. Вы можете переключиться на **NFSv4.2**, если вам нужна эта функция.

ЦЕЛЬ СЕРТИФИКАЦИИ 16.03

NFS с Kerberos

В течение нескольких лет **NFS** считался небезопасным протоколом. Одна из причин заключается в том, что **NFS** по умолчанию доверяет **UID** и **GID**, отправленным клиентом. Хакер «черной шляпы», имеющий доступ к общему ресурсу **NFS**, может легко выдать себя за другого пользователя и передать его учетные данные **UID/GID**, поскольку **NFS** основывается на доверии.

Проблемы безопасности **NFSv4** были решены с помощью **Kerberos**, который может обеспечить надежную аутентификацию, целостность и услуги шифрования. Если вам нужна безопасность с **NFS**, защитите экспорт **NFS** с помощью **Kerberos**.

Этот раздел посвящен настройке **NFS** с сервером **Kerberos**. Предполагается, что вы настроили **Kerberos KDC** и что клиенты присоединились к области **Kerberos**, как описано в **главе 12**.

Службы NFS с поддержкой Kerberos

Чтобы настроить простую службу **NFS**, как вы это делали в **упражнении 16-1**, вам нужно активировать системный модуль **nfs-server** на хосте **NFS-сервера**. Если вы хотите интегрировать **NFS** с **Kerberos**, вам нужно включить две дополнительные службы, **nfs-secure-server** и **nfs-secure**, как показано в таблицах **16-3** и **16-4**.

Поэтому самый простой способ настроить все необходимые службы на **NFS-сервере** - использовать следующие команды:

```
# systemctl start nfs-server
# systemctl start nfs-secure-server
# systemctl enable nfs-server
# systemctl enable nfs-secure-server
```

Также важно включить следующий сервисный модуль на всех клиентах **NFS**:

```
# systemctl start nfs-secure
# systemctl enable nfs-secure
```

Как отмечено в **Главе 11**, эти команды запускают указанные сервисные блоки и обеспечивают запуск сервисов при следующей перезагрузке системы.

ТАБЛИЦА 16-3 Служебные модули systemd на сервере NFS с поддержкой Kerberos

systemd Service Unit	Описание
nfs-serve	Основной сервисный блок для **NFS-сервера**. Он активирует другие сервисные модули, такие как **nfs-idmap, nfs-rquotad** и **rpcbind.service**. Он использует **/etc/sysconfig/nfs** для базовой конфигурации.
nfs-secure-server	Предоставляет проверку подлинности и шифрование на основе **Kerberos** для сервера **NFS** с помощью демона **rpc.svcgssd**.

ТАБЛИЦА 16-4 Служебные модули systemd на клиенте NFS с поддержкой Kerberos

systemd Service Unit	Описание
nfs-secure	Предоставляет службы аутентификации и шифрования **Kerberos** клиенту **NFS** через демон **rpc.gssd**

Настройте экспорты NFS с помощью Kerberos

Конфигурация экспорта **NFS** с поддержкой **Kerberos** проста и основана на параметре безопасности **sec** в **/etc/exports**, который мы уже встречали в **Таблице 16-2**.

За параметром **sec** следует разделенный двоеточиями список вариантов безопасности, которые **NFS-сервер** предоставляет своему клиенту. В качестве примера рассмотрим следующую строку из файла **/etc/exports**:

```
/nfs-share *.example.com(rw,sec=sys:krb5:krb5p)
```

Эта конфигурация экспортирует каталог **/nfs-share** через **NFS** клиентам в домене **example.com** с доступом для чтения и записи. Клиенты могут подключить общий ресурс **NFS**, используя один из следующих параметров безопасности: **sys, krb5** или **krb5p**.

Эти параметры показаны в **таблице 16-5**. Из приведенной в таблице информации самый безопасный метод экспорта - **krb5p**, поскольку он обеспечивает аутентификацию **Kerberos**, целостность данных и шифрование. Однако это обходится «дорого», поскольку для шифрования данных требуются **ресурсы ЦП**, что может существенно повлиять на производительность.

Опции безопасности **krb5** и **krb5i** предоставляют услуги аутентификации и целостности и являются хорошим компромиссом между безопасностью и пропускной способностью. Наконец, метод безопасности **sys** соответствует модели доверия **UID/GID NFS**, которая всегда принимается в качестве метода безопасности по умолчанию, если опция безопасности **sec** не указана в **/etc/exports**.

Если вы хотите, чтобы клиенты **NFS** монтировали общий ресурс **NFS**, используя определенный параметр безопасности, включите этот параметр в качестве параметра **sec**. Например, следующая строка в **/etc/exports** гарантирует, что клиенты в домене **example.com** монтируют каталог **nfs-share** с аутентификацией, целостностью и шифрованием **Kerberos**:

nfs-share *.example.com(rw,sec=krb5p)

Не забудьте запустить **exportfs -r** на сервере **NFS**, чтобы применить изменения и обновить список экспортированных каталогов.

ТАБЛИЦА 16-5 Параметры безопасности NFS

Опция безопасности	Описание
sys	Доверяет **UID/GID**, предоставленный клиентами, для определения прав доступа к файлу. Включено по умолчанию, если опция **sec=не указана**.
krb5	Проверяет UID / GID, предоставленный клиентами с использованием аутентификации Kerberos.
krb5i	Имеет тот же эффект, что и опция krb5, но обеспечивает надежную коммуникационную целостность.
krb5p	Имеет тот же эффект, что и опция krb5i, но дополнительно предоставляет услуги шифрования.

Настройте NFS-клиенты с помощью Kerberos

Клиенты **NFS** могут легко смонтировать общий ресурс **NFS** с помощью служб аутентификации, целостности и шифрования **Kerberos**, используя параметр **sec** со значениями, перечисленными в **таблице 16-5**. Для этого включите параметр **sec** либо с командой **mount**, либо в файл **/etc/fstab**.

Например, следующая команда монтирует каталог **nfs-share** с хоста **192.168.122.50** с использованием аутентификации **Kerberos**:

mount -t nfs -o sec=krb5 192.168.122.50:/nfs-share /mnt

Аналогично, следующая строка в **/etc/fstab** указывает системе монтировать каталог **nfs-share** во время процесса загрузки с использованием аутентификации **Kerberos**, шифрования и строгой целостности:

192.168.122.50:/nfs-share /mnt nfs soft,sec=krb5p 0 0

УПРАЖНЕНИЕ 16-2
Подготовьте систему для NFS, защищенной с помощью Kerberos

Чтобы подготовить систему для экспорта общих каталогов через **NFS**, защищенную с помощью **Kerberos**, необходимо выполнить несколько шагов настройки. Мы предполагаем, что вы установили **Kerberos KDC** и настроили **server1.example.com** для аутентификации **Kerberos**, как показано в **упражнении 12-5**.

Затем на **KDC** выполните следующие действия:

1. Создайте участников хоста для сервера **NFS** (**server1.example.com**) и всех клиентов (например, **tester1.example.com**):

   # kadmin.local
   Authenticating as principal root/admin@WAMPLE.COM with password
   kadmin.local: addprinc -randkey host/server1.example.com
   WARNING: no policy specified for host/server1.example.com@EXAMPLE.COM;
   defaulting to no policy
   Principal "host/server1.example.com@EXAMPLE.COM" created.
   kadmin.local: addprinc -randkey host/tester1.example.com
   WARNING: no policy specified for host/tester1.example.com@EXAMPLE.COM;
   defaulting to no policy
   Principal "host/tester1.example.com@EXAMPLE.COM" created.
   kadmin.local:

2. Добавьте участников службы **NFS** для сервера и клиентских компьютеров:

   kadmin.local: addprinc -randkey nfs/server1.example.com
   WARNING: no policy specified for nfs/server1.example.com@EXAMPLE.COM;
   defaulting to no policy
   Principal "nfs/server1.example.com@EXAMPLE.COM" created.
   kadmin.local: addprinc -randkey nfs/tester1.example.com
   WARNING: no policy specified for nfs/tester1.example.com@EXAMPLE.COM;
   defaulting to no policy
   Principal "nfs/tester1.example.com@EXAMPLE.COM" created.
   kadmin.local:

3. Создайте файлы ключей для сервера **NFS** и клиентских компьютеров:

   # kadmin.local: ktadd -k /tmp/server1.keytab nfs/server1.example.com
   [output truncated]
   # kadmin.local: ktadd -k /tmp/tester1.keytab nfs/server1.example.com
   [output truncated]

4. Скопируйте файлы ключей в файл **/etc/krb5.keytab** на удаленных системах:

   # scp /tmp/server1.keytab server1.example.com:/etc/krb5.keytab
   # scp /tmp/tester1.keytab tester1.example.com:/etc/krb5.keytab

5. Скопируйте файл **/etc/krb5.conf** из **KDC** на все серверы и клиенты **NFS**:

   # scp /etc/krb5.conf server1.example.com:/etc/krb5.conf
   # scp /etc/krb5.conf tester1.example.com:/etc/krb5.conf

УПРАЖНЕНИЕ 16-3

Настройка общего ресурса NFS с поддержкой Kerberos

В этом упражнении вы установите сервер **NFS** в системе **RHEL** и экспортируете общий ресурс с аутентификацией и шифрованием **Kerberos**. В этом упражнении предполагается, что вы настроили центр распространения ключей **Kerberos** и настроили виртуальные машины **server1.example.com** и **tester1.example.com**, как описано в **упражнениях 12-5 и 16-2.**

1. Убедитесь, что сервер **NFS** установлен на **server1.example.com**. Самый простой способ с помощью следующей команды:

   # rpm -q nfs-utils

2. Если он еще не установлен, используйте методы, описанные ранее, для установки **RPM-пакета nfs-utils**.
3. Запустите службу **NFS** и ее безопасный компонент, чтобы обеспечить службы аутентификации и шифрования **Kerberos**:

   # systemctl start nfs-server nfs-secure-server

4. Убедитесь, что сервисы автоматически активируются при следующей загрузке системы с помощью следующей команды:

   # systemctl enable nfs-server nfs-secure-server

5. Создайте каталог с именем **nfs-secure**:

   # mkdir /nfs-secure

6. Сконфигурируйте общий ресурс в файле **/etc/exports**, чтобы разрешить чтение и запись всем клиентам с аутентификацией и шифрованием **Kerberos**:

   # echo "/nfs-secure *(rw,sec = krb5p)" >> /etc/exports

7. Примените изменения:

   # exportfs -r

8. Убедитесь, что служба **nfs** включена в зоне брандмауэра по умолчанию:

   # firewall-cmd --list-all

9. Если он не включен, добавьте службу в зону по умолчанию:

   # firewall-cmd --permanent --add-service=nfs
   # firewall-cmd --reload

10. На клиенте **tester1.example.com** убедитесь, что установлен **RPM-пакет nfs-utils.**
11. Запустите службу **nfs-secure** и активируйте ее при загрузке:

    # systemctl start nfs-secure
    # systemctl enable nfs-secure

12. Создайте каталог для общего ресурса сервера с именем **/mnt/nfs**:

    # mkdir /mnt/nfs

13. Добавьте следующую строку в **/etc/fstab**:

192.168.122.50:/nfs-secure /mnt/nfs nfs sec=krb5p 0 0

14. Запустите команду **mount -a**, чтобы смонтировать общий ресурс.

СЦЕНАРИЙ И РЕШЕНИЕ	
У вас возникают проблемы при настройке брандмауэра для **NFS**.	Включите службу **nfs**, запустив **firewall-cmd –add-\ service=nfs**.
Вы хотите запретить доступ на чтение/запись к общим каталогам **NFS**.	Убедитесь, что общие ресурсы настроены с помощью параметра **ro** в **/etc/exports**.

Вам необходимо настроить автоматическое монтирование общего каталога **NFS**.	Настройте общий каталог в **/etc/fstab**.
Вы хотите экспортировать общий ресурс **NFS** с аутентификацией и шифрованием **Kerberos**.	Экспортируйте и смонтируйте общий ресурс с помощью опция **sec=krb5p**. Убедитесь, что в ваших системах настроена проверка подлинности **Kerberos**, как описано в **Приложении А**.
Вам необходимо запустить службы **NFS** для экспорта общего ресурса **NFS** с проверкой подлинности **Kerberos**.	Включите службы **nfs-server** и **nfs-secure-server** на сервере **NFS** и **nfs-secure** на клиентах.

РЕЗЮМЕ СЕРТИФИКАЦИИ

NFS позволяет обмениваться файловыми системами между компьютерами **Linux** и **Unix**. Это эффективный способ обмена файлами между такими системами, который можно защитить с помощью аутентификации и шифрования **Kerberos**.

Хотя **RHEL 7** поддерживает **NFSv4**, он также поддерживает доступ клиентов **NFSv3**. Он контролируется группой системных модулей. Сервисный блок **nfs-server** необходим для запуска демона **NFS**. Аутентификация и шифрование на основе **Kerberos** контролируются демонами **rpcsvcgssd** и **rpcgssd**, которые зависят соответственно от сервисного блока n**fs-secure-server (на сервере) и nfs-secure (на клиенте).** Глобальные параметры для службы **NFS** настраиваются в основном в файле **/etc/sysconfig/nfs**. Связанные команды включают **exportfs** и **showmount**.

В большинстве случаев вы можете настроить базовую конфигурацию **NFS** с помощью простой однострочной директивы в файле **/etc/exports**. После запуска службы **NFS** такой экспорт активируется с помощью команды **exportfs**. Межсетевые экраны должны быть настроены путем включения службы **nfs** в соответствующую зону. Активные порты и службы могут быть подтверждены с помощью команды **rpcinfo -p**.

Как правило, конфигурация **SELinux** по умолчанию поддерживает базовую работу **NFS**. Вы можете настроить безопасность для смонтированных каталогов **NFS**, как если бы смонтированные файловые системы были локальными. Вы также можете автоматизировать монтирование **NFS** в **/etc/fstab** или через автомонтирование. Текущее состояние **NFS** задокументировано в различных файлах в каталогах **/var/lib/nfs** и **/proc/fs/nfsd**.

Пару минут проверки

Вот некоторые из ключевых моментов целей сертификации в главе 16.

Сервер сетевой файловой системы (NFS)

- **NFS** - это стандарт для обмена файлами между компьютерами **Linux** и **Unix**. **RHEL7** поддерживает **NFS версий 3 и 4**; **NFSv4** используется по умолчанию.
- Ключевыми демонами **NFS** являются **rpc.mountd** для запросов на монтирование, **rpc.rquotad** для работы с запросами с определёнными квотами и демон **nfsd**.
- Параметры конфигурации для этих процессов можно найти в файле **/etc/sysconfig/nfs**.
- Общие ресурсы **NFS** настраиваются в **/etc/exports** и активируются с помощью команды **exportfs \ -r**.
- Брандмауэры можно настроить, включив службу **nfs** в соответствующую зону в **firewalld.**
- В большинстве случаев требуемые логические значения для работы **ntfs** в **SELinux** уже активны.
- Чтобы запретить доступ на **чтение/запись** в **SELinux**, отключите логическое значение **nfs_export_all_rw**.
- Когда каталоги **NFS** смонтированы, они должны выглядеть незаметно. Пользовательские разрешения работают так же, как с локальными каталогами.

Проверьте клиент NFS

- Клиенты могут монтировать постоянные общие ресурсы **NFS** через **/etc/fstab**.
- Вы можете просмотреть общие каталоги на клиенте с помощью команды **showmount**.
- Команда **mount** предназначена для монтирования общих каталогов через **NFSv4** и **NFSv3**.

- Если сервер **NFS** дает сбой, он может «повесить» клиента **NFS**. Опции **soft** и **timeo**
- команды **mount** могут помочь предотвратить такие зависания. Однако их использование может привести к подрыву целостности данных в случае сбоя системы.

NFS с Kerberos

По умолчанию **NFS** небезопасен, поскольку доверяет **UID/GID**, отправляемому клиентами.
При интеграции с **Kerberos NFS** может обеспечить строгую аутентификацию (**sec=krb5**),
целостность связи (**sec=krb5i**) и шифрование (**sec=krb5p**).
Чтобы настроить общие папки **NFS** на основе **Kerberos**, укажите соответствующий параметр безопасности через параметр **sec=** на клиентах и сервере **NFS**.
Служба **nfs-secure-server** должна быть запущена на сервере **NFS**, чтобы обеспечить услуги **Kerberos**.
Служба **nfs-secure** должна быть запущена на клиентах **NFS** для поддержки проверки подлинности при монтирование с использованием **Kerberos**.
NFS с Kerberos требует от вас настроить **KDC**, как описано в **главе 12**.

Самопроверка

Следующие вопросы помогут вам оценить ваше понимание материала, представленного в этом глава. Поскольку на экзаменах Red Hat нет вопросов с несколькими вариантами ответов, нет вопросов с несколькими вариантами ответов появляются в этой книге. Эти вопросы исключительно проверяют ваше понимание главы. Это нормально, если у вас есть другой способ выполнения задачи. Получение результатов, а не запоминание пустяков, это то, что рассчитывает на Red Hat экзамены. На многие из этих вопросов может быть более одного ответа.

Сервер сетевой файловой системы (NFS)

1. В файле **/etc/exports** вы хотите экспортировать каталог **/data** только для чтения на все хосты и предоставить разрешение на чтение и запись для **супервизора** имени хоста в домене **example.com**. Какую директиву вы бы ввели в этот файл?

_____

2. После того, как вы настроили **/etc/exports**, какая команда экспортирует эти ресурсы?

_____

3. Какой номер порта связан с **portmapper**?

_____

4. Какой номер порта связан с **NFSv4**?

_____

5. Что такое опция конфигурации **NFS**, которая поддерживает доступ пользователя с правами администратора?

_____

Проверьте клиент NFS

6. У вас проблемы с клиентами **NFS** по разным причинам, включая частые простои
на сервере **NFS** и разрывов сети между клиентами и серверами **NFS**. Какой тип
монтирование может помешать клиентам **NFS** зависать и повторять запросы **NFS** на неопределенный срок?

_____

7. Какая команда может отображать общие каталоги **NFS** из системы **outsider1.example.org**?

_____

NFS с Kerberos

8. Какой сервис следует запустить на клиенте **NFS** для поддержки аутентификации на основе **Kerberos** через демон **rpcgssd**?

_____

9. Какую директиву следует включить для подключения общего ресурса **NFS** с проверкой подлинности **Kerberos** и шифрование?

_____

10. Какую директиву следует добавить в **/etc/exports** для экспорта общего ресурса **NFS** со стандартным доступом к файлам разрешения и опционально с проверкой подлинности **Kerberos**?

_____

ВОПРОСЫ ЛАБОРОТОРНОЙ РАБОТЫ

Некоторые из этих лабораторий включают в себя упражнения по настройке. Вы должны делать эти упражнения только на тестовых машинах. Предполагается, что вы выполняете эти упражнения на виртуальных машинах, таких как KVM. В этой главе также предполагается, что вы можете изменять конфигурацию физической хост-системы для таких виртуальных машин.

Red Hat представляет свои экзамены в электронном виде. По этой причине лабораторные работы в этой главе доступны в подкаталоге глава 16/ на носителе, который сопровождает книгу. Если вы еще не настроили **RHEL 7** в системе, инструкции по установке см. В **главах 1 и 2**.

Ответы для каждой лаборатории следуют за ответами самопроверки для вопросов, которые заполняются.

Лабораторная работа 1

Для этой лабораторной работы вам понадобятся два компьютера с **Linux**: один в качестве **сервера NFS**, а второй в качестве **клиента NFS**. Назовите эти компьютеры **server1.example.com** и **tester1.example.com**. Настройте каталог с именем **/shared**. Добавьте один файл в этот каталог. Начните с настройки правил для совместного использования этого каталога с директивой **no_root_squash**. Для любого настроенного брандмауэра обязательно ограничьте доступ к сети **example.com**. Если вы следовали инструкциям в **Главе 1, это была бы сеть 192.168.122.0/24**.

Перейдите в удаленную систему и убедитесь, что вы можете просматривать **общие каталоги NFS** с сервера. Смонтируйте эту систему в локальном каталоге **/testing**. Вы можете скопировать файлы в этот каталог?

Размонтируйте общий ресурс и удалите директиву **no_root_squash**. Что происходит, когда вы монтируете общий ресурс **NFS** во второй раз с клиента?

Лабораторная работа 2

В этой лабораторной работе могут использоваться те же две системы, которые использовались в лабораторной работе 1. На сервере совместно используйте каталоги **/home** и предоставьте разрешения на запись клиентскому компьютеру. На клиенте настройте каталог **/home** с **сервера NFS**, который будет монтироваться при следующей загрузке клиентского компьютера. Поскольку на клиентском компьютере, вероятно, уже есть каталог **/home**, настройте его в каталоге **/remote.**

Лаборатория 3

В этой лабораторной работе вы будете экспериментировать с различными настройками **SELinux**. Например, если вас просят запретить запись в какие-либо общие каталоги **NFS**, что вы делаете?

Лабораторная работа 4

Эта лабораторная работа расширяет **лабораторную работу 1** для настройки каталога /**shared NFS** с дополнительной аутентификацией **Kerberos**, целостностью связи и шифрованием. Сначала **настройте KDC** и настройте системы **server1 и tester1**, как описано в **упражнениях 12-5 и 16-2**.
Перейдите на удаленный клиент **tester1.example.com** и убедитесь, что вы можете смонтировать общий ресурс **NFS**, используя любой из параметров безопасности **sec=krb5, sec=krb5i и sec=krb5p**.

Затем протестируйте каждый из следующих сценариев. Что происходит, когда вы пытаетесь смонтировать общий ресурс **NFS**?

1. Остановите службу **nfs-secure-server** на **server1.example.com**. Попробуйте смонтировать общий ресурс **NFS**.
2. Запустите службу **nfs-secure-server** на сервере **server1.example.com**. Остановите службу **nfs-secure** на **tester1.example.com**. Попробуйте смонтировать общий ресурс **NFS**.
3. Запустите **службу nfs-secure** на **tester1.example.com**. Переместите файл **/etc/krb5.keytab** в другое место на **tester1.example.com**. Попробуйте смонтировать общий ресурс **NFS**.
4. Восстановите файл **/etc/krb5.keytab** на **tester1.example.com**. Переместите файл **/etc/krb5.conf** в другое место на **tester1.example.com**. Попробуйте смонтировать общий ресурс **NFS**.
5. Восстановите файл **/etc/krb5.conf** на **tester1.example.com**. Удалите службу **nfs** в **firewalld**. Попробуйте смонтировать общий ресурс **NFS**.
6. Добавьте службу **nfs** в **firewalld**. Остановите службы **KDC**. Попробуйте смонтировать общий ресурс **NFS**.

ОТВЕТЫ НА САМОПРОВЕРКУ

Сервер сетевой файловой системы (NFS)

1. Следующая запись в **/etc/exports** экспортирует каталог **/data** только для чтения на все хосты и предоставьте разрешение на чтение и запись для хоста **superv** в домене **example.com**:

/data superv.example.com(rw,sync) (ro,sync)

2. После того, как вы изменили **/etc/exports**, команда **exportfs -a** экспортирует все файловые системы. Да, вы также можете повторно экспортируйте файловые системы с помощью команды **exportfs -r**.
3. Номер порта, связанный с **portmapper**, является портом **UDP 111**.
4. Номер порта, связанный с **SNFv4**, - это порт **TCP 2049**.
5. Параметр конфигурации **NFS**, который поддерживает доступ пользователя с правами администратора, - **no_root_squash**.

Проверьте клиент NFS

6. Мягкий монтаж и тайм-ауты, связанные с параметрами **soft** и **timeo**, могут помочь клиентам избегать зависание и повторение запросов **NFS** на неопределенный срок.
7. Команда, которая может отображать общие каталоги **NFS** из указанной удаленной системы:

showmount -e outsider1.example.org.

NFS с Kerberos

8. Вам следует запустить **сервер nfs-secure**, чтобы обеспечить поддержку аутентификации на основе **Kerberos** на клиенте через демона **rpcgssd**.
9. Директива, которую вы должны включить, чтобы смонтировать общий ресурс **NFS** с аутентификацией и шифрованием **Kerberos** будет **sec=krb5p**.
10. Вы можете экспортировать общий ресурс с помощью опции безопасности **sec=sys:krb5p**.

Ответы Лабораторной работы

Лабораторная работа 1

По завершении этой лабораторной работы вы увидите следующие функции в системе с **NFS-сервером:**

- **RPM-пакет nfs-utils** в списке установленных пакетов.
- **Активная служба NFS**, которая может быть подтверждена в выходных данных команды **systemctl status nfs-server**
- Брандмауэр на основе зоны, поддерживающий доступ к службе **nfs**. Также должен быть ограничен по IP адресу сети.

Кроме того, вы сможете выполнять следующие задачи из клиента **NFS**:

Вы можете выполнить команду **showmount -e server1.example.com**, где **server1.example.com** имя системы **NFS-сервера** (подставьте при необходимости).
Вы можете смонтировать общий каталог как пользователь **root** с помощью **mount -t nfs server1.example .com**:**/shared** командой **/testing**.
При первом подключении общего ресурса вы сможете копировать локальные файлы от имени пользователя **root** в каталог **/testing**.
Во время второго монтирования общего ресурса с действующей директивой **no_root_squash**, например, копирование не должно работать, по крайней мере, из учетной записи пользователя **root**.

Лабораторная работа 2

Эта лабораторная работа - первый шаг к созданию единого **/home** каталога для вашей сети. Как только вы получите это работая над одной комбинацией **клиент/сервер**, вы можете настроить ее на всех клиентах и серверах. Вы можете затем используйте сервер **LDAP** для настройки единой базы данных **Linux/Unix** с именами пользователей и паролями для сети. В качестве альтернативы, совпадающие имена пользователей (с совпадающими номерами **UID** и **GID**) на разных локальных системах также должны работать. На сервере **NFS** выполните следующие действия:

1. Настройте пару пользователей и определите файлы, такие как **user1** и **user1.txt**, в системе использующих в качестве сервера **NFS**.
2. Настройте доступ в общий каталог **/home** в **/etc/exports** на клиенте **server1.example.com**. Вы можете сделать это в указанном файле с помощью следующей команды:

   /home *.example.com(rw,sync)

3. Экспортируйте этот каталог с помощью следующей команды:

   #exportfs –a

4. Убедитесь, что каталог **/home** отображается в списке экспорта. На локальном сервере вы может сделать это с помощью следующей команды:

   # showmount -e server1.example.com

5. Если во время этого процесса возникают проблемы, внимательно проверьте файл **/etc/exports**. Убедитесь, что в **/etc/exports** нет лишних пробелов, даже в конце строки кода. Убедитесь, что служба **NFS** фактически выполняется с помощью команды **systemctl status nfs-server**.
6. Вы также можете проверить брандмауэр и убедиться, что соответствующие службы описаны в с помощью команды **rpcinfo -p**.
7. Не забудьте убедиться, что сервер **NFS** запускается автоматически при следующем запуске системы. Один из способов сделать это с помощью следующей команды:

   # systemctl enable nfs-server

Теперь на клиенте NFS выполните следующие шаги для подключения к общему каталогу **/home:**

1. Убедитесь, что вы видите общий **/home** каталог. Вы можете заменить IP-адрес доменным именем **server1.example.com**:

 # showmount -e server1.example.com

2. Теперь смонтируйте общий ресурс, локально в каталоге **/remote**Ж

   # mount -t nfs server1.example.com:/home /remote

3. Запустите команду монтирования. Если вы видите монтирование **NFS**, все хорошо.
4. Изучите смонтированный каталог **/home**. Найдите файлы ***.txt**, созданные ранее в этой лабораторной работе. Если вы нашли эти файлы, значит вы успешно создали и подключились к общему каталогу **/home.**
5. Чтобы сделать монтирование постоянным, добавьте его в файл **/etc/fstab** на клиенте. Как только вы добавили строку, указанную далее в этом файле, клиент **Linux** автоматически монтирует общий каталог **/home** с **сервера NFS** при следующей загрузке клиента, с опцией **soft** и тайм-аут 100 секунд, который может помочь предотвратить «зависание»:

   server1.example.com:/home /remote nfs soft soft,timeout=100

Лабораторная работа 3

Ссылка на **SELinux** является преднамеренной и должна дать важный совет. Вам может не хватить время для изменения всех общих и настроенных каталогов в файле **/etc/exports** на каждом сервере **NFS**. Один простой способ предотвратить запись в общие каталоги NFS - это деактивировать связанный логический **SELinux**, с помощью следующей команды:

 # setsebool -P nfs_export_all_rw off

После этого вы сможете проверить результат при следующем подключении общего каталога **NFS**.

Лабораторная работа 4

Эта лаборатория является продолжением **Упражнения 16-2** и пытается познакомить вас с некоторыми из общих проблем при настройке общих ресурсов **NFS с Kerberos**.

Экспортируйте общий ресурс с параметром безопасности **sec=sys:krb5:krb5i:krb5p**, чтобы предоставить дополнительный **Kerberos** аутентификация, целостность связи и шифрование. Посмотрите, может ли клиент **tester1.example.com** смонтируйте общий ресурс **NFS**, используя любой из доступных методов безопасности. Воспроизведите сценарии устранения неполадок описано в лаборатории и обратите внимание на сообщения об ошибках, с которыми вы сталкиваетесь.