

Лабораторная работа № 13

Тема: Сетевые сервисы на Linux. NFS.

Цель работы: Создать и настроить NFS-папку.

Необходимое оборудование и программное обеспечение: Виртуальные машины под управлением Linux (CentOS, Ubuntu или др.).

Сервисы хранения данных. NFS

NFS (Network File System) - это протокол для удаленного доступа к файловой системе через сеть. Он позволяет пользователям или компьютерам в сети монтировать удаленные файловые системы так, как если бы они были локальными.

Основные концепции и компоненты NFS:

1. **Сервер NFS:** Это компьютер или устройство, на котором работает служба NFS и которое предоставляет доступ к своим локальным файлам для удаленных клиентов.
2. **Клиент NFS:** Это компьютер или устройство, которое монтирует удаленные файловые системы с сервера NFS и получает к ним доступ.
3. **Экспорт NFS:** Это процесс на сервере NFS, при котором определенные локальные файловые системы или каталоги сделаны доступными для удаленных клиентов NFS.
4. **Монтирование NFS:** Это процесс на клиенте NFS, при котором удаленные файловые системы экспортируются сервером NFS и присоединяются (монтируются) к локальной файловой системе клиента.

Преимущества NFS включают простоту в настройке и использовании, высокую производительность и возможность централизованного хранения данных. NFS широко используется в Unix-подобных операционных системах, таких как Linux и FreeBSD, а также в средах виртуализации и кластеризации для обмена данными между серверами и хранения общих файлов.

nfs-utils - это набор утилит и служб, предназначенных для работы с протоколом Network File System (NFS) в Linux. Этот пакет предоставляет инструменты для настройки, управления и мониторинга NFS-серверов и клиентов.

Включает в себя следующие основные компоненты:

1. **nfsd:** Это демон NFS-сервера, который обслуживает запросы клиентов NFS и управляет доступом к файлам на сервере. Он обычно запускается на NFS-сервере.
2. **rpc.nfsd:** Это служба, которая обеспечивает поддержку протокола RPC для демона nfsd.
3. **rpc.mountd:** Этот демон обеспечивает поддержку протокола RPC для монтирования удаленных файловых систем клиентов NFS.
4. **rpc.lockd и rpc.statd:** Эти демоны обеспечивают поддержку блокировок файлов и управление состоянием для NFS.
5. **showmount:** Это утилита командной строки, которая отображает список экспортированных файловых систем на сервере NFS.
6. **mount.nfs:** Это утилита командной строки, которая используется для монтирования удаленных файловых систем NFS на клиенте.
7. **umount.nfs:** Это утилита командной строки, используемая для отмонтирования удаленных файловых систем NFS на клиенте.
8. **exportfs:** Это утилита командной строки для управления файлом **/etc/exports**, который определяет, какие локальные файловые системы экспортируются для доступа через NFS.

nfs-utils позволяют администраторам управлять настройками NFS-сервера и клиента, а также монтировать и отмонтировать удаленные файловые системы NFS. Этот пакет является ключевым

компонентом в средах, где используется NFS для обмена данными и централизованного хранения файлов.

Пример настройки NFS.

Сервер. (r3)

Установка nfs-utils:

```
[root@r3 ~]# yum install -y nfs-utils
Загружены модули: fastestmirror
Loading mirror speeds from cached hostfile
 * base: mirror.ps.kz
 * extras: mirror.ps.kz
 * updates: mirror.ps.kz

quota-nls.noarch 1:4.01-19.el7
tcp_wrappers.x86_64 0:7.6-77.el7

Выполнено!
[root@r3 ~]#
```

Запустим сервис:

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

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

```
[root@r3 ~]# systemctl start nfs-server
[root@r3 ~]# systemctl enable nfs-server
Created symlink from /etc/systemd/system/multi-user.target.wants/nfs-
server.service to /usr/lib/systemd/system/nfs-server.service.
[root@r3 ~]# systemctl status nfs-server
● nfs-server.service - NFS server and services
   Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; enable
   reset: disabled)
   Active: active (exited) since Пн 2024-03-18 18:26:28 +05; 19s ago
 Main PID: 1695 (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/nfs-server.service

map 18 18:26:28 r3 systemd[1]: Starting NFS server and services...
map 18 18:26:28 r3 systemd[1]: Started NFS server and services.
[root@r3 ~]#
```

Создадим директорию, в ней файл с текстом:

```
[root@r3 ~]# mkdir /test-nfs
```

```
GNU nano 2.3.1                               Файл: /nfs-test/test-file.txt

nfs-server
```

Отредактируем конфигурационный файл `/etc/exports`:

`nano /etc/exports`

```
GNU nano 2.3.1                                Файл: /etc/exports
nfs-test 2.2.2.2 (rw,sync,no_root_squash)
```

Файл `/etc/exports` является конфигурационным файлом для настройки экспорта файловых систем через протокол NFS (Network File System) в системах Linux. В этом файле определяются пути к файловым системам, которые разрешено экспортировать для доступа клиентов NFS, а также параметры доступа к этим ресурсам.

Каждая строка в файле `/etc/exports` обычно содержит следующие элементы:

1. **Путь к экспортируемой файловой системе:** Это путь к каталогу или файловой системе, которую вы хотите сделать доступной для клиентов NFS. Например, `/home` или `/srv/nfs`.
2. **Опции экспорта:** Это опции, определяющие параметры доступа к экспортируемой файловой системе, такие как разрешения на чтение, запись, а также права доступа. Опции указываются в круглых скобках и разделяются запятыми. Например, `(rw,sync,no_root_squash)`.

Для примера, вот как может выглядеть содержимое файла `/etc/exports`:

```
/home 192.168.1.0/24(rw,sync,no_root_squash)
```

```
/srv/nfs 192.168.1.100(rw) 192.168.1.101(ro)
```

- `/home` экспортируется для всех клиентов в сети `192.168.1.0/24` с правами чтения и записи (`rw`), синхронизацией записей (`sync`) и без подмены `root` (`no_root_squash`).
- `/srv/nfs` экспортируется для клиента с IP-адресом `192.168.1.100` с правами чтения и записи (`rw`) и для клиента с IP-адресом `192.168.1.101` только для чтения (`ro`).

Опции (`rw`, `sync`, `no_root_squash`) в файле `/etc/exports` определяют параметры доступа к экспортируемой файловой системе через NFS. Давайте рассмотрим каждую из этих опций:

1. **rw:** Это сокращение от "read-write" и указывает, что клиенты NFS имеют права на чтение и запись файлов на экспортированной файловой системе. То есть клиенты могут как читать файлы с удаленного хоста, так и записывать новые файлы или изменять существующие файлы на этом хосте.
2. **ro:** в файле `/etc/exports` для NFS определяет, что экспортируемая файловая система доступна только для чтения (read-only) для клиентов NFS.
3. **sync:** Эта опция указывает на то, что NFS-сервер должен синхронизировать изменения файловой системы с клиентами, прежде чем ответить на запросы

обработки файлов. Это означает, что сервер подтверждает запись только после того, как данные записаны на диск. Это может повлиять на производительность, так как операции NFS будут замедлены из-за ожидания записи на диск.

4. **no_root_squash**: Эта опция отключает механизм "root squashing", который по умолчанию предотвращает клиентам с правами root на удаленном хосте выполнение привилегированных операций на файловой системе, экспортированной через NFS. При использовании этой опции клиент с правами root сохраняет свои привилегии и может выполнять операции на экспортированной файловой системе от имени root.
5. * вместо ip-адреса – папка доступна всем

После внесения изменений в файл `/etc/exports`, необходимо перезапустить службу NFS, чтобы изменения вступили в силу:

```
[root@r3 ~]# systemctl restart nfs-server  
[root@r3 ~]#
```

Или `exportfs -a`

Добавим правило в фаерволл и перезапустим его:

```
firewall-cmd --permanent --zone=public --add-service=nfs
```

```
firewall-cmd --reload
```

```
# firewall-cmd --list-all
```

```
[root@r3 ~]# systemctl restart nfs-server  
[root@r3 ~]# firewall-cmd --add-port=2049/tcp --permanent  
success  
[root@r3 ~]# systemctl reload firewalld  
[root@r3 ~]#
```

Настройка на стороне клиента (r1)

Устанавливаем nfs-utils:

```
[root@r1 ~]# yum install -y nfs-utils  
Загружены модули: fastestmirror  
Loading mirror speeds from cached hostfile  
* base: mirror.ps.kz  
* extras: mirror.ps.kz  
* updates: mirror.ps.kz
```

Создадим директорию:

```
[root@r2 ~]# mkdir /nfs
```

Смонтируем в нее nfs-папку. Создадим файл и заполним его:

```
[root@r2 ~]# mount -t nfs 3.3.3.3:/nfs-test /nfs  
[root@r2 ~]# nano /nfs/test.txt
```

```
root@r2:~  
GNU nano 2.3.1          файл: /nfs/test.txt  
hello. test nfs share
```

Прочитаем файл, созданный на сервере:

```
[root@r2 ~]# cat /nfs/test-file.txt  
nfs-server
```

Теперь прочитаем файл, созданный на клиенте, на сервере:

```
[root@r3 ~]# cat /nfs-test/test.txt  
hello. test nfs share  
[root@r3 ~]#
```

Чтобы NFS папка автоматически монтировалась при каждой загрузке клиентской машины, добавьте запись в файл **/etc/fstab**:

`sudo nano /etc/fstab`

Добавьте строку следующего формата:

`3.3.3:/nfs-test /nfs nfs defaults 0 0`