

# NFS HOWTO

---

Nicolai Langfeldt jan1@linpro.no, перевод Alex Ott ott@phtd.tpu.edu.ru v1.0, 1 Октября 1999

Как установить настроить клиент и сервер NFS

## Содержание

|           |   |           |
|-----------|---|-----------|
| <b>1</b>  | <b>Преамбула</b>  | <b>1</b>  |
| <b>2</b>  | <b>README.first</b>                                       | <b>2</b>  |
| <b>3</b>  | <b>Установка сервера NFS</b>                              | <b>3</b>  |
| <b>4</b>  | <b>Настройка клиента NFS</b>                              | <b>5</b>  |
| <b>5</b>  | <b>NFS через медленные линии</b>                          | <b>7</b>  |
| <b>6</b>  | <b>Безопасность и NFS</b>                                 | <b>9</b>  |
| <b>7</b>  | <b>Контрольный список разрешения проблем монтирования</b> | <b>12</b> |
| <b>8</b>  | <b>Часто Задаваемые Вопросы (FAQ)</b>                     | <b>13</b> |
| <b>9</b>  | <b>Экспортирование файловых систем</b>                    | <b>15</b> |
| <b>10</b> | <b>NFS в Linux 2.2</b>                                    | <b>16</b> |
| <b>11</b> | <b>NFS-сервер на дискете</b>                              | <b>18</b> |
| <b>12</b> | <b>PC-NFS</b>   | <b>20</b> |

**Примечание переводчика:** Шлите мне любые комментарии и замечания, даже небольшие.

## 1 Преамбула

### 1.1 Официальная часть

Авторские права Nicolai Langfeldt и Ron Peters, 1997-1999. Не вносить изменения без изменения авторских прав, распространяется свободно, но с сохранением этого параграф. Раздел FAQ основывается на NFS FAQ, собранном Alan Cox. Раздел по разрешению проблем основан на списке проверки (checklist) проблем монтирования, собранном IBM Corporation. Раздел nfs-server-on-a-floppy был написан Ron Peters.

### 1.2 Отказ от ответственности

Ни Nicolai Langfeldt, ни Ron Peters, ни их работодатели, и никто иной не отвечают за то, что может случиться, если вы будете следовать инструкциям приведенным в этом документе. Если вы в любом случае будете следовать данным инструкциям, то желаем удачи!

### 1.3 Обратная связь

Это документ никогда не будет не завершен, пожалуйста присылайте мне сообщения о ваших успехах и проблемах — это сделает этот документ лучше. Пожалуйста посылайте деньги, комментарии и/или вопросы на адрес [janl@math.uio.no](mailto:janl@math.uio.no), или [rpeters@hevanet.com](mailto:rpeters@hevanet.com), если они относятся к разделу сервер NFS на гибком диске. Если вы посылаете сообщение по электронной почте, пожалуйста *будьте уверены*, что ваш обратный адрес правилен и работает, я получаю довольно много сообщений по электронной почте и определение вашего адреса может задать мне лишнюю работу. Пожалуйста.

### 1.4 Остальное

Если вы хотите перевести этот HOWTO пожалуйста дайте мне знать, чтобы я мог отслеживать на каких языках я опубликован :-).

Проклятия и благодарности Olaf Kirch, который заставил меня написать этот документ и кто сделал много рекомендаций для него :-)

### 1.5 Посвящение

Этот документ посвящен Anne Line Norheim Langfeldt. Хотя она вероятно никогда не прочитает его, поскольку она не относится к девушкам, интересующимся этим вопросом. - Nicolai

## 2 README.first

NFS — Network File System (Сетевая файловая система) имеет три важных свойства:

- Она делает возможным разделение файлов по сети.
- Она в большинстве работает достаточно хорошо.
- Ее открытие может быть риском для безопасности, что хорошо понимается кракерами и может быть легко взломано чтобы получить доступ (чтение, запись, удаление) ко всем вашим файлам.

В этом документе я расскажу о первых двух вопросах. Пожалуйста, убедитесь, что вы читали раздел о безопасности в этом документе и вы сможете уменьшить уязвимость своей системы от проблем безопасности. Пассажи о секретности будут довольно техническими и будут требовать некоторого знания о работе IP сетей и владение используемыми терминами. Если вы не знаете эти термины, то вы должны вернуться назад и просмотреть относящиеся к сети HOWTO или взять книгу о сетевом администрировании TCP/IP, чтобы ознакомить себя с TCP/IP. В любом случае это хорошая идея, если вы занимаетесь администрированием машин с UNIX/Linux. Очень хорошая книга о данном предмете — это книга *TCP/IP Network Administration*, написанная Craig Hunt и опубликованная O'Reilly & Associates, Inc. И после того, как вы прочтаете и поймете ее у вас будет больше шансов при поиске работы, вы просто не сможете быть слабым ;-)

В этом документе есть два раздела чтобы помочь вам найти неисправности при работе с NFS, эти разделы называются *Список проверки проблем монтирования* и *FAQs*. Пожалуйста посмотрите их, если что-то работает не так, как требуется.

Базовым сервером для Linux 2.0 nfsd является <ftp.mathematik.th-darmstadt.de:/pub/linux/okir>, в том случае, если вы хотите/должны пересобрать его сами.

Для информации о NFS в Linux 2.2 пожалуйста смотрите [10](#) (раздел Linux 2.2).

## 3 Установка сервера NFS

### 3.1 Непременные условия

До того, как вы продолжите читать этот документ вам будет необходимо успешно выполнять операцию telnet между машинами, которые вы будете использовать как сервер и клиент. Если что-то не работает, вам нужно прочитать NET-3 HOWTO и правильно настроить работу сети.

### 3.2 Первый шаг

До того, как мы сможем сделать что-нибудь нам необходимо настроить сервер NFS. Если вы являетесь частью сети факультета или университета, то у вас вероятно есть несколько настроенных серверов NFS. Конечно, если они позволят вам получить доступ к ним и если вы читаете этот документ чтобы получить доступ к одному из них, то вам можно не читать это раздел и вы можете просто пропустить его до раздела 4 (Установка клиента NFS)

Если вас нужно настроить не-Linux машину как сервер, то вам нужно прочитать руководство по нужной системе, чтобы определить как разрешить работу сервера NFS и экспортировать файловую систему через NFS. Описание того, как это сделать на разных платформах вынесено в отдельный раздел. После того, как вы определили все, что вам нужно вы можете продолжать чтение следующего раздела этого документа. Или читайте дальше этот раздел, поскольку для некоторых вещей, о которых я буду говорить не важно какой тип машины вы используете как сервер.

Если вы торопитесь, то пожалуйста посмотрите 10 (раздел Linux 2.2) до того, как вы продолжите читать это.

То, о чем вы читали, потребует от вас настройки нескольких программ.

### 3.3 Portmapper

Portmapper на Linux называется либо `portmap` либо `rpc.portmap`. Справочная страница на моей системе говорит, что это "Преобразователь номеров портов DARPA в вызовы соответствующих программ RPC". Это первая дыра в безопасности, которую вы откроете читая этот документ. Описание того, как закрыть одну из таких дыр находится в 6 (разделе по безопасности), который я советую вам обязательно прочитать.

Запустите portmapper. Он называется либо `portmap`, либо `rpc.portmap` и должен находиться в директории `/usr/sbin` (на некоторых машинах он называется `rpcbind`). Вы можете запустить его сейчас вручную, но он должен запускаться при каждом запуске вашей машины, так что вам необходимо создать/отредактировать rc-скрипты. Содержание ваших rc-скриптов объясняется более подробно в справочной странице `init`. Они обычно находятся в директориях `/etc/rc.d`, `/etc/init.d` или `/etc/rc.d/init.d`. Если там есть скрипт, названный `inet`, то его мы и будем редактировать. Но то, что в нем необходимо написать или что необходимо сделать еще, находится вне области рассмотрения этого документа. Запустите `portmap`, и проверьте, что он запущен с помощью команды `ps aux` и затем `rpcinfo -p`. Это сделано? Хорошо.

Одна вещь. Удаленный доступ к вашей программе portmapper определяется содержимым ваших файлов `/etc/hosts.allow` и `/etc/hosts.deny`. Если `rpcinfo -p` не работает, но ваш portmapper запущен, то пожалуйста проверьте указанные файлы. Смотрите 6 (раздел о безопасности) для детального описания этих файлов.

### 3.4 Mountd и nfsd

Следующие программы, которые нам нужно запустить далее — это `mountd` и `nfsd`. Но сначала мы отредактируем другой файл. Это файл `/etc/exports`. Допустим я хочу, чтобы файловая система `/mn/eris/local`, которая находится на машине `eris` была доступна для машины названной `apollo`. Тогда я должен поместить в файл `/etc/exports` на машине `eris` следующие строки:

---

```
/mn/eris/local  apollon(rw)
```

---

Вышеприведенные строки дают машине `apollon` право на чтение/запись в каталог `/mn/eris/local`. Вместо `rw` мы можем сказать `ro`, что означает доступ только для чтения (если вы ничего не поменяете, то по умолчанию будет доступ только для чтения). Существуют другие опции, которые вы можете задать здесь, и я позже рассмотрю некоторые из них, относящиеся к проблеме безопасности. Они все перечислены в справочной странице `exports`, которую вы должны прочитать по крайней мере раз в жизни. Существуют также лучшие способы, чем перечисление всех машин в файле `exports`. Вы например можете использовать сетевые группы, если у вас используется система NIS (или NYS) (NIS также известен как YP), и всегда использовать шаблоны (wild cards) доменов и подсетей IP как списки машин, которым разрешено что-то монтировать. Но вы должны учитывать, кто может получить доступ к серверу неавторизованным способом, если вы используете такую всеобъемлющую авторизацию.

**Замечание: Этот файл `exports` не имеет такой же синтаксис, который используют другие системы Unix.** В этом документе есть отдельный раздел о файлах `exports` других Unix-систем.

Сейчас мы готовы к запуску программ `mountd` (она также может называться `rpc.mountd`) и `nfsd` (который может назван `rpc.nfsd`). Обе эти программы читают данные из файла `exports`.

Если вы отредактировали файл `/etc/exports`, то вы должны быть уверены, что `nfsd` и `mountd` знают о том, что файл изменен. Традиционный способ сделать это — это запустить программу `exportfs`. Во многих дистрибутивах Linux программа `exportfs` отсутствует. Если это так, то вы можете создать такой скрипт на вашей машине:

---

```
#!/bin/sh
killall -HUP /usr/sbin/rpc.mountd
killall -HUP /usr/sbin/rpc.nfsd
echo re-exported file systems
```

---

Сохраните его в файле, скажем `/usr/sbin/exportfs`, и не забудьте выполнить над ним команду `chmod a+rx`. Сейчас, после того как, вы изменили ваш файл `exports`, вы должны запустить программу `exportfs`, имея права администратора.

Теперь вы должны проверить, что `mountd` и `nfsd` запущены правильно. Сначала это делается с помощью команды `rpcinfo -p`. Вывод программы должен показать что-то похожее на следующее:

---

| program | vers | proto | port |            |
|---------|------|-------|------|------------|
| 100000  | 2    | tcp   | 111  | portmapper |
| 100000  | 2    | udp   | 111  | portmapper |
| 100005  | 1    | udp   | 745  | mountd     |
| 100005  | 1    | tcp   | 747  | mountd     |
| 100003  | 2    | udp   | 2049 | nfs        |
| 100003  | 2    | tcp   | 2049 | nfs        |

---

Как вы видите `portmapper` анонсировал свои сервисы, и что `mountd` и `nfsd` запущены.

Если вы получили сообщение `rpcinfo: can't contact portmapper: RPC: Remote system error - Connection refused, RPC_PROG_NOT_REGISTERED` или что-то подобное вместо этого, то значит `portmapper` не запущен. Или у вас может быть что-то записано в файлах `/etc/hosts.{allow,deny}` что запрещает программе `portmapper` отвечать, пожалуйста посмотрите 6 (раздел о безопасности) для подробного описания этих файлов. Если вы получили сообщение `No remote programs registered.`, то либо `portmapper` не хочет говорить с вами, либо что-то не в порядке. Завершите выполнение `nfsd`, `mountd` и `portmapper` и попытайтесь выполнить заново стартовую последовательность.

После проверки того, что portmapper объявил сервисы, вы также можете выполнить проверку работы с помощью команды ps. Portmapper будет продолжать объявлять свои сервисы даже после того как программы расширяющие его возможности завершили работу. Так что проверка с помощью ps может быть необходимой, если вам кажется, что что-то не работает.

Конечно вам будет нужно исправить ваши системные rc-файлы для запуска mountd и nfsd при загрузке. Очень вероятно, что эти скрипты уже существуют на вашей машине, и вам будет нужно только раскомментировать нужный раздел или активизировать скрипт на нужном уровне запуска. Справочные страницы, которые вы должны уже изучить: portmap, mountd, nfsd и exports.

Если вы сделали все как я сказал, то вы должны были установить все необходимое для работы сервера NFS.

## 4 Настройка клиента NFS

Первым делом вам нужно ядро с поддержкой файловой системы NFS, либо вкомпилированной в ядро, либо доступной как модуль. Это настраивается до компиляции ядра. Если вы никогда не компилировали ядро, то вам может быть нужно прочитать *Kernel HOWTO* и выяснить как это делается. Если вы используете хороший дистрибутив (такой как RedHat) и вы никогда не экспериментировали с ядром или модулями (и таким образом разрушали его ;-), то вероятно, что поддержка nfs уже есть в ядре.

Теперь вы можете, в командной строке администратора, ввести соответствующую команду монтирования и файловая система появится у вас. Продолжая пример из предыдущего раздела мы хотим смонтировать /mn/eris/local с машины eris. Это делается с помощью такой команды:

```
mount -o rsize=1024,wsizе=1024 eris:/mn/eris/local /mnt
```

(Мы еще вернемся к опциям rsize и wsize). Файловая система сейчас доступна в /mnt и вы можете перейти туда и выполнить в ней команду ls, и посмотреть на индивидуальные файлы. Вы заметите, что эта операция выполняется не так быстро как над локальной файловой системой, но более удобно чем ftp. Если вместо монтирования файловой системы команда mount выдаст сообщение об ошибке mount: eris:/mn/eris/local failed, reason given by server: Permission denied, то файл exports является неправильным или вы забыли запустить exportfs после редактирования файла exports. Если команда сообщит mount clntudp\_create: RPC: Program not registered это означает, что nfsd или mountd не запущены на сервере. Или у вас проблема с hosts.{allow,deny}, о которой упомянуто выше.

Чтобы прекратить пользоваться файловой системы вы можете выполнить:

```
umount /mnt
```

Чтобы выполнялось автоматическое монтирование файловой системы nfs при загрузке, вам необходимо отредактировать файл /etc/fstab как обычно это делается. Для нашего примера требуется такая строка:

| # device            | mountpoint | fs-type | options               | dump | fsckorder |
|---------------------|------------|---------|-----------------------|------|-----------|
| ...                 |            |         |                       |      |           |
| eris:/mn/eris/local | /mnt       | nfs     | rsize=1024,wsizе=1024 | 0    | 0         |
| ...                 |            |         |                       |      |           |

Это почти все, что необходимо. Читайте пожалуйста дальше.

## 4.1 Опции монтирования

Здесь перечислены некоторые опции, которые вы должны рассмотреть сразу, добавляя их в файл настроек. Они управляют способом, которым клиент NFS обрабатывает прекращение работы сервера или отключение сети. Одно из свойств NFS в том, что он может изящно обрабатывать эти неполадки, если вы правильно установите клиента. Существует два различающихся режима обработки ошибок:

### soft

NFS клиент будет сообщать об ошибке программе, которая пытается получить доступ к файлу расположенному на файловой системе, смонтированной через NFS. Некоторые программы довольно хорошо обрабатывают такого рода ошибки, но большинство программ не делают это. Я не рекомендую использование этой опции, она может привести к появлению испорченных файлов и потерянных данных. Вы особенно не должны использовать эту опцию для дисков, используемых для почты, если ваша почта что-то значит для вас.

### hard

Программа осуществляющая доступ к файлу на смонтированной по NFS файловой системе просто приостановит выполнение при разрыве связи с сервером. Процесс не может быть прерван или убит до тех пор, пока вы явно не укажете опцию `intr`. Когда сервер NFS будет запущен заново, то программа продолжит безмятежно продолжать работу с прерванного места. Это скорее всего то, что вам нужно. Я рекомендую использовать опции `hard, intr` на всех файловых системах смонтированных через NFS.

Продолжая предыдущий пример, теперь в нашем файле `fstab` запись будет выглядеть так:

---

```
# device      mountpoint    fs-type    options                                dump fsckorder
...
eris:/mn/eris/local /mnt      nfs        rsize=1024, wsize=1024, hard, intr 0 0
...
```

---

## 4.2 Оптимизация NFS

Обычно, если не заданы опции `rsize` и `wsize`, то NFS будет читать и писать блоками по 4096 или по 8192 байтов. Некоторые комбинации ядер Linux и сетевых карт не могут обрабатывать такие большие блоки, и это может быть не оптимально. Так что нам нужно поэкспериментировать и найти значения `rsize` и `wsize`, которые работают так быстро, насколько это возможно. Вы можете протестировать скорость передачи при заданных опциях при помощи нескольких простых команд. Выполнив вышеприведенную команду монтирования и получив доступ с правом записи на диск, вы можете выполнить тестирование производительности последовательной записи:

---

```
time dd if=/dev/zero of=/mnt/testfile bs=16k count=4096
```

---

Эта команда создает 64Мб файл, заполненный нулевыми значениями (этот файл должен быть достаточно большим, настолько большим, чтобы кэширование не сыграло значительную роль в производительности, используйте больший размер файла, если у вас достаточно много памяти). Прodelайте эту операцию несколько раз (5-10?) и усредните полученные результаты. Полученная величина — это время 'прохода', т.е. величина наиболее интересующая нас в этом эксперименте. Затем вы можете измерить производительность чтения, прочитав файл обратно на свою машину:

---

```
time dd if=/mnt/testfile of=/dev/null bs=16k
```

---

выполните эту операцию несколько раз и усредните результат. Затем отмонтируйте файловую систему и примонтируйте ее заново, с увеличенными значениями `rsize` и `wsize`. Вероятно они должны быть кратными 1024, и не больше чем 16384 байтов, поскольку это максимальный размер блока данных в NFS версии 2. Прямо после монтирования с увеличенными значениями перейдите в смонтированную файловую систему и выполните команду подобную `ls`, исследуйте файловую систему, чтобы убедиться, что все в норме. Если значения `rsize/wsize` слишком большие, то симптомы *очень* необычные и не на 100% очевидные. Типичный симптом выражается в неполном списке файлов при выполнении команды `'ls'`, и отсутствие сообщений об ошибках. Или чтение файлов загадочно срывается без сообщения об ошибке. После того, как вы установите, что заданные значения `rsize/wsize` работают, вы можете далее продолжать тестировать производительность. Различные серверные платформы вероятно имеют различные оптимальные размеры блоков. SunOS и Solaris по общему мнению, работают довольно быстрее при размере блока равном 4096 байт, чем при других значениях.

Новые ядра Linux (с версии 1.3) выполняют предваряющее чтение для значений `rsize` больших или равных размеру страницы машины. На процессорах Intel размер страницы равен 4096 байтам. Предваряющее чтение *значительно* увеличивает производительность NFS при чтении. Так что на машинах с процессором Intel вы можете захотеть использовать значение `rsize` равное 4096 байтам. Помните, что вам нужно отредактировать `/etc/fstab` для использования найденных значений `rsize/wsize`.

Прием для увеличения производительности NFS при записи заключается в запрещении синхронной записи на сервер. Спецификация NFS требует, чтобы запросы NFS на запись не считались законченными до записи данных на носитель (обычно диск). Это ограничивает производительность записи, а асинхронная запись значительно увеличит скорость записи по NFS. Демон `nfsd` для Linux никогда не делает синхронную запись, поскольку реализация файловой системы Linux сама не дает сделать это, но серверах работающих на отличных от Linux системах вы можете увеличить производительность этим способом, поместив в ваш файл `exports`:

---

```
/dir    -async,access=linuxbox
```

---

или что-то подобное. Пожалуйста посмотрите справочную страницу `exports` на данной машине. Также запомните, что это увеличивает риск потери данных.

## 5 NFS через медленные линии

Медленные линии включают в себя модемы, ISDN и другие соединения на дальние расстояния. Этот раздел базируется на знании об используемых протоколах, а не на настоящих экспериментах. Пожалуйста дайте мне знать, если вы попробуете сделать это :-)

Первая вещь которую вы должны помнить, что NFS — медленный протокол. Использование NFS в большинстве своем подобно использованию протокола `kermit` для переноса файлов. Это — *медлено*. Почти все быстрее чем NFS. FTP быстрее. HTTP быстрее. `rcp` быстрее. `ssh` быстрее. Вы все еще хотите попробовать его в работе? Ок.

Параметры по умолчанию для NFS установлены для довольно быстрых линий с малым временем запаздывания. Если вы будете использовать эти настройки для линий с высоким временем запаздывания, то это приведет к выдаче сообщений об ошибках, прерыванию операций, система может притворяться, что файлы короче, чем они есть на самом деле и странно работать в других случаях. Первое, что вам необходимо сделать — это *не* использовать опцию монтирования `soft`. Это вызовет возвращение программному обеспечению сигналов об ошибках при таймаутах. В основном обычное программное обеспечение не слишком хорошо обрабатывает такие ошибки. Это хороший способ получить странные сбои. Вместо этого используйте опцию монтирования `hard`. Когда активна опция `hard`, то таймауты вызывают бесконечные попытки возобновления вместо прерывания работы ваших программ. Это то, что вам действительно нужно.

Следующая вещь, которую нужно сделать — это поэкспериментировать с опциями монтирования `timeo` и `retrans`. Они описаны в справочной странице `nfs(5)`, здесь приводится выдержка из нее:

---

|                        |   |
|------------------------|---|
| <code>timeo=n</code>   | Величина в десятых долях секунды до отправки первой ретрансляции после таймаута RPC. По умолчанию эта величина равна 7 десятых секунды. После первого таймаута, время таймаута удваивается после каждого таймаута, пока не будет достигнута величина максимального таймаута равна 60 секундам, или произойдет достаточно ретрансляции, вызвав главный таймаут. Затем если файловая система смонтирована с опцией <code>hard</code> , то каждый новый таймаут каскадно запускается с начальным значением в два раза больше, чем при предыдущем каскаде, кроме того удваиваясь на каждой ретрансляции. Максимальный таймаут всегда равен 60 секундам. Наилучшая общая производительность может быть достигнута увеличением таймаута при монтировании на загруженной сети, к медленному серверу, или сквозь несколько маршрутизаторов. |
| <code>retrans=n</code> | Эта величина задает количество неосновных таймаутов и ретрансляций, которые должны произойти до возникновения главного таймаута. По умолчанию эта величина равна 3. Когда возникает главный таймаут, то файловые операции либо прерываются или на консоли печатается сообщение <code>"server not responding"</code> .   |

---

Другими словами: Если запрос не будет передан за таймаут равный 0.7 секунды (700ms), то клиент NFS повторит запрос и увеличит таймаут в два раза, до 1.4 секунды. Если ответ не придет в течении 1.4 секунды, то запрос повторится снова и таймаут будет увеличен до 2.8 секунды. Скорость линии может быть измерена с помощью команды `ping` с размером пакета равным значению, установленному опциями `rsize/wsize`.

---

```
$ ping -s 8192 lugulbanda
PING lugulbanda.uio.no (129.240.222.99): 8192 data bytes
8200 bytes from 129.240.222.99: icmp_seq=0 ttl=64 time=15.2 ms
8200 bytes from 129.240.222.99: icmp_seq=1 ttl=64 time=15.9 ms
8200 bytes from 129.240.222.99: icmp_seq=2 ttl=64 time=14.9 ms
8200 bytes from 129.240.222.99: icmp_seq=3 ttl=64 time=14.9 ms
8200 bytes from 129.240.222.99: icmp_seq=4 ttl=64 time=15.0 ms

--- lugulbanda.uio.no ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 14.9/15.1/15.9 ms
```

---

Здесь время показывает как долго пакет программы `ping` идет туда и обратно к машине `lugulbanda`. 15ms это довольно быстро. При работе через модем со скоростью 28.000 бод вы можете ожидать где-то 4000-5000ms, и если линия нагружена еще кем-то, то время будет даже выше, может быть раза в два. Когда это время высоко, мы говорим что это 'высокое запаздывание'. В общем для



больших пакетов и для более загруженных линий запаздывание будет увеличиваться. Увеличьте `timeo` соответственно вашей линии и загрузке. И поскольку запаздывание увеличивается когда вы используете линию для других вещей: даже если вы хотите использовать FTP и NFS в одно и тоже время, то вы должны попытаться измерить `timeo ping` во время использования FTP для передачи файлов.

## 6 Безопасность и NFS

Я ни коим образом не являюсь экспертом в области компьютерной безопасности. Но у меня есть *маленький* совет для сознающих проблему безопасность. Но будьте предупреждены: этот список ни в коем случае не является полным списком проблем относящихся к NFS, и если вы думаете, что вы обезопасились один раз прочитав и выполнив, все что я даю здесь, то я хочу предупредить вас. Этот раздел не должен беспокоить вас, если вы находитесь в *закрытой* сети, где вы доверяете всем пользователям, и никто из тех кому вы не доверяете ни может получить доступ к машинам в сети. Например, не должно быть dial-соединения в сеть, и не должно быть никакого способа подключиться к сети, в которой вы есть люди, которым вы не доверяете. Вы думаете я параноик? Я не параноик. Это базовый совет по безопасности. *Безопасность* требует наличия тщательного и знающего администратора, который знает где найти информацию о текущих и потенциальных проблемах безопасности.

Основная проблема NFS в том, что клиент, если не задано, будет доверять серверу и наоборот. Это может быть плохо. Это значит, что если запись администратора сервера NFS взломана, то также легко может быть взломана запись администратора клиентской машины. И наоборот. Существует набор полицейских стратегий для этого, мы к ним еще вернемся.

Что вам необходимо прочитать — это консультационные материалы CERT относящиеся к NFS. Большинство текстов приведенных ниже, связаны с советами, написанными в выпусках CERT. Смотрите [ftp.cert.org/01-README](http://ftp.cert.org/01-README) для обновленного списка консультативных материалов CERT. Здесь приведены некоторые относящиеся к NFS консультативные материалы:

---

|  |          |
|--|----------|
| CA-91:21.SunOS.NFS.Jumbo.and.fsirand   | 12/06/91 |
| Уязвимость в отношении сетевой файловой системы (NFS) Sun Microsystems, Inc. (Sun) и программы fsirand. Эта уязвимость возможна в версиях SunOS 4.1.1, 4.1, and 4.0.3 на всех архитектурах. Заплаты (Patches) доступны для SunOS 4.1.1. Также доступна начальная заплатка для SunOS 4.1 NFS. Sun будет обеспечить полные заплатки для SunOS 4.1 и SunOS 4.0.3 позже. |          |
| CA-94:15.NFS.Vulnerabilities   | 12/19/94 |
| Этот консультационный материал обеспечивает измерение безопасности для охраны против некоторых дыр в безопасности в сетевой файловой системе (NFS). Этот материал выпущен в связи с увеличением случаев взлома машин, используя утилиты для взлома через уязвимые точки.   |          |
| CA-96.08.pcnfsd  | 04/18/96 |
| Этот материал описывает проблемы с безопасностью в программе pcnfsd (также известной как rpc.pcnfsd). Заплата для исправления ошибки прилагается.  |          |

---

### 6.1 Безопасность клиента

На клиентской стороне мы можем решить, что мы не хотим слишком сильно доверять серверу. Это делается несколькими способами, используя опции монтирования. Например, мы можем запретить

выполнение программ с установленным битом `suid` в файловой системе NFS, это делается опцией монтирования `nosuid`. Это хорошая идея и вы должны рассмотреть ее, используя смонтированные через NFS файловые системы. Это означает, что администратор сервера не сможет сделать программы с установленным `suid`-администратора на файловой системе, затем войти на машину клиента как обычный пользователь и используя программу с `suid`-администратора приобрести также права администратора на машине клиента. Мы также можем запретить выполнение файлов на смонтированной файловой системе с помощью опции `noexec`. Но она применяется реже по сравнению с опцией `nosuid`, поскольку файловая система может содержать по крайней мере *некоторые* скрипты, или программы, которые необходимо выполнять. Вы можете ввести эти опции в колонке опций вместе с опциями `rsize` и `wsize`, разделяя их запятыми.

## 6.2 Безопасность сервера: `nfsd`

На стороне сервера мы можем решить, что мы не хотим доверять администратору клиента. Мы можем сделать это указав опцию `root_squash` в файле `exports`:

---

```
/mn/eris/local apollon(rw,root_squash)
```

---

Теперь, если пользователь с UID 0 на стороне клиента попытается получить доступ (чтение, запись, удаление), то файловый сервер выполнит подстановку UID пользователя 'nobody' на сервере. Это означает, что администратор клиента не сможет получить доступ или изменять файлы, которые может изменять или иметь доступ к которым может только администратор сервера. Это хорошо и вы должны использовать опцию `root_squash` на всех экспортируемых файловых системах. Вы скажете, что "Администратор клиента все равно может выполнять команду 'su', чтобы зайти как любой другой пользователь и получить доступ и изменить любые пользовательские файлы". На это есть ответ: "Да есть такой способ, и это работает в Unix и NFS. Это имеет одно важное заключение: Все важные файлы и программы должны иметь владельцем пользователя `root`, а не пользователя `bin` или другого пользователя не-администратора, поскольку только администратор клиента не может получить доступ как администратор сервера. С справочной странице NFSd есть несколько других подобных опций, так что вы можете решить, что вы (не) доверяете кому-либо со стороны клиента. У вас также имеются опции для осечения любых диапазонов UID и GID. Это описывается в справочной странице Linux NFSd.

Опция `root_squash` является установленной по умолчанию для NFSd в Linux, для передачи администраторских полномочий для доступа к файловой системе используйте опцию `no_root_squash`. Другая важная вещь, которую необходимо сделать, это проверить, что `nfsd` проверяет, все ли запросы приходят с привелигированного порта. Если он принимает запросы с любого старого порта на клиенте, то пользователь без специальных привилегий может запустить программу, которую легко получить по Internet. Он умеет "говорить" на языке протокола `nfs` и будет притворяться, что пользователь является любым пользователем, которым он хочет быть. NFSd на Linux делает эту проверку по умолчанию, но для других операционных систем вы должны разрешить эту проверку сами. Это должно быть описано в справочной странице `nfsd` для вашей операционной системы. Другая вещь. Никогда не экспортируйте файловую систему для машины с именем 'localhost' или 127.0.0.1. Доверяйте мне.

## 6.3 Безопасность сервера: `portmapper`

Основа `portmapper`, в соединении с `nfsd` имеет проблему в проектировании, которая делает возможной получить файлы с серверов NFS без каких-либо привилегий. К счастью `portmapper` используемый в большинстве дистрибутивов Linux использует относительную безопасность против такой атаки, и может быть сделано более безопасной настройкой списка доступа в двух файлах. Не все дистрибутивы Linux обладают равными возможностями. Некоторые дистрибутивы, которые кажутся современными, не включают в свой состав безопасный вариант программы `portmapper`,

даже сейчас, через много лет после того как стало известно о ее уязвимостях. По крайней мере один дистрибутив даже содержит справочную страницу более безопасного portmapper, но на самом деле его portmapper *не является* безопасным. Самым легким способом проверить является ли ваш portmapper хорошим или нет, является запуск программы strings(1) и просмотр ее вывода на наличие файлов, /etc/hosts.deny и /etc/hosts.allow. Предполагая, что ваш portmapper находится /usr/sbin/portmap, вы можете проверить это выполнив следующую команду: strings /usr/sbin/portmap | grep hosts. на моей машине это вышло со следующими результатами:

---

```
/etc/hosts.allow
/etc/hosts.deny
@(#) hosts_ctl.c 1.4 94/12/28 17:42:27
@(#) hosts_access.c 1.20 96/02/11 17:01:27
```

---

Сначала мы отредактируем файл /etc/hosts.deny. Он должен содержать строку

---

```
portmap: ALL
```

---

которая запретит доступ *всем*. Поскольку это закрыло доступ всем, запустите rpcinfo -p просто для проверки того, что ваш portmapper читает этот файл и выполняет данные инструкции. Команда rpcinfo не должна ничего вывести или должна выдать сообщение об ошибке. Перезапуск portmapper *не является* необходимым.

Закрытие portmap для всех может быть слишком кардинальным, поэтому мы снова откроем доступ, изменив файл /etc/hosts.allow. Но сначала нам надо определить, что мы туда поместим. В этом файле перечисляются все машины, которые могут получить доступ к вашему portmapper. Среди множества работающих под Linux систем только некоторым машинам нужен полный доступ для любой работы. Portmapper обслуживает nfsd, mountd, ypbind/ypserv, rpcnfsd, и 'r' сервисы, такие как rpartime и rusers. Из них только nfsd, mountd, ypbind/ypserv и возможно rpcnfsd имеют какое-либо важное значение. Всем машинам, которым необходим доступ к сервисам на вашей машине должно быть разрешено делать это. Скажем адрес машины равен 129.240.223.254 и она находится в подсети 129.240.223.0, и ей нужен доступ к сервисам на вашей машине (эти термины введены HOWTO по сетям, вернитесь к нему и освежите свои знания, если это необходимо). Для этого мы напишем в файле hosts.allow

---

```
portmap: 129.240.223.0/255.255.255.0
```

---

Это тоже самое, что и сетевой адрес, который вы даете командой route и маска подсети, которую вы передаете команде ifconfig. Для устройства eth0 на этой машине ifconfig должен показывать

---

```
...
eth0      Link encap:10Mbps Ethernet  HWaddr 00:60:8C:96:D5:56
          inet addr:129.240.223.254  Bcast:129.240.223.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:360315 errors:0 dropped:0 overruns:0
          TX packets:179274 errors:0 dropped:0 overruns:0
          Interrupt:10 Base address:0x320
...
```

---

а программа netstat -rn должна показывать

---

```
Kernel routing table
Destination      Gateway          Genmask          Flags Metric Ref Use     Iface
...
129.240.223.0    0.0.0.0          255.255.255.0   U        0      0    174412 eth0
...
```

(Сетевой адрес находится в первой колонке).

Файлы `hosts.deny` и `hosts.allow` описаны в справочных страницах с теми же именами.

**ВАЖНО:** Не помещайте в этих файлах *ничего*, кроме *IP HOMEPOB* в строках для настройки `portmap`. Поиск имен машин может вызвать активность `portmap`, которая вызовет поиск имен машин, которое вызовет `portmap`, которое вызовет...

Вышеприведенные вещи должны вызвать переключение вашего сервера. Остающаяся проблема в том, что кто-то взламывает администратора (или загрузит MS-DOS) на машине, которой доверяют и использует эти привилегии для посылки запросов на безопасный порт, как любой пользователь, которым он захочет быть.

## 6.4 NFS и firewall

Очень хорошая идея защитить порты `nfs` и `portmap` с помощью `firewall` на вашем маршрутизаторе. `Nfsd` работает на порту 2049, используя оба протокола — `udp` и `tcp`. `Portmapper` работает на порту 111, `tcp` и `udp`, а `mountd` работает на портах 745 и 747, `tcp` и `udp`. По умолчанию. Вы должны проверить номера используемых портов, используя команду `rpcinfo -p`.

Если вы хотите использовать NFS сквозь `firewall`, то есть опции для новых версий `NFSd` и `mountd`, для того, чтобы заставить их использовать нестандартные порты, которые могут быть открыты в `firewall`.

## 6.5 Резюме

Если вы используете `hosts.allow/deny`, `root_squash`, `nosuid` и привилегированные порты в программном обеспечении `portmapper/nfs`, то вы можете избежать известных ошибок в `nfs` и можете чувствовать себя почти в безопасности. Но все равно: когда взломщик имеет доступ к вашей сети, то он/она может добавить странные команды в ваш файл `.forward` или почтовый ящик, когда `/home` или `/var/spool/mail` смонтирован через NFS. По той же причине, вы никогда не должны осуществлять доступ к вашим личным ключам PGP через `nfs`. Или по крайней мере вы должны знать какой риск существует. И знать о нем хотя бы немного.

NFS и `portmapper` создают комплексную систему и поэтому не полностью невероятно, что новые ошибки будут найдены, либо в основе проекта, либо в реализации, которую мы используем. Также могут быть известные дыры, которые кто-нибудь использует. Но такова жизнь. Чтобы быть в курсе таких вещей, вы должны как минимум читать группы новостей `comp.os.linux.announce` и `comp.security.announce`.

## 7 Контрольный список разрешения проблем монтирования

Этот раздел основан на контрольном списке проблем монтирования, этот документ написан в IBM Corp. Я благодарен им за то, что они сделали его доступным для использования в этом документе. Если у вас есть проблема с монтированием файловой системы через NFS, то пожалуйста проверьте этот список, до того как вы пошлете сообщение об ошибке. Каждый пункт описывает конкретную проблему и ее решение.

1. Команда `mount` продолжает сообщать `RPC: Program not registered`

Запущен ли `portmapper`?

**Исправление:** Запустите его.

Запущен ли `mountd`?

**Исправление:** Запустите его.

Запущен ли `nfsd`?

**Исправление:** Запустите его.

Не запрещено ли `portmapper` отвечать на ваши запросы файлом `/etc/hosts.deny`?

**Исправление:** Либо удалите правило из файла `hosts.deny` либо добавьте правило в файл `hosts.allow`, так что `portmapper` будет разрешено общаться с вами.

2. Файловая система не экспортируется, или не экспортируется при запросе клиента.

**Исправление:** Экспортируйте ее

3. Система разрешения имен не выдает соответствия со списком машин в файле `exports`.

Например: список экспортируемых ресурсов задает экспортирование `johnmad`, но имя `johnmad` разрешается как `johnmad.austin.ibm.com` и монтирование запрещается.

**Исправление:** Экспортируйте ресурс для обеих форм имени машины.

Это также случается, если клиент имеет 2 интерфейса с разными именами для каждого из них и файловая система экспортируется только для одного указанного имени.

**Исправление:** Экспортируйте оба интерфейса.

Это также может произойти, если сервер не может выполнить функции `lookuphostbyname` или `lookuphostbyaddr` (это библиотечные функции) на клиенте. Убедитесь, что клиент может выполнять команды `host <name>; host <ip_addr>;` и обе они указывают на одну и ту же машину.

**Исправление:** наладьте систему разрешения имен.

4. Файловая система была смонтирована, после того как NFS был запущен (на том сервере). В таком случае сервер экспортирует саму точку монтирования, а не смонтированную файловую систему.

**Исправление:** Завершите `NFSd` и затем перезапустите его.

**Замечание:** Клиенты, которые уже были примонтированы к точке монтирования файловой системы будут иметь проблемы с доступом к ней после перезапуска сервера.

5. Дата наобум изменяется на одной или обеих машинах (это может спутать `make`).

**Исправление:** Установите правильную дату.

Автор HOWTO рекомендует использовать NTP для синхронизации часов. Поскольку существуют экспортные ограничения на NTP в US, то вы можете получить NTP для Debian, Red Hat или Slackware с <ftp://ftp.hacktic.nl/pub/replay/pub/linux> или с сервера-зеркала.

6. Сервер не допускает монтирования от пользователя, который находится больше чем в 8 группах.

**Исправление:** уменьшите количество групп в которые входит пользователь или смонтируйте под именем другого пользователя.

## 8 Часто Задаваемые Вопросы (FAQ)

Этот раздел часто задаваемых вопросов (FAQ). Большая часть его основана на старом FAQ, написанном Alan Cox.

Если у вас есть проблемы с монтированием файловой системы, то пожалуйста посмотрите, не описана ли она в разделе "Список проверок при монтировании".

1. Я получаю сообщения об ошибках "stale nfs handle (устарелый дескриптор nfs)" при использовании Linux как сервера nfs. Это вызывается ошибкой в одной из старых версий nfsd. Это исправлено в nfs-server2.2beta16 и более поздних.

2. Когда я пытаюсь примонтировать файловую систему я получаю сообщение

```
can't register with portmap: system error on send
(не могу зарегистрироваться с помощью portmap: системная ошибка при посылке)
```

Вы вероятно используете систему Caldera. Это ошибка в скриптах rc. Пожалуйста свяжитесь с Caldera для получения исправления.

3. Почему я не могу выполнить файл после копирования его на NFS сервер? Причина в том, что nfsd кэширует дескрипторы открытых файлов для улучшения производительности (помните, что он запущен в пространстве пользователей). Пока nfsd держит файл открытым (как в этом случае, после записи в него), то ядро не позволит вам выполнять его. Nfsds новее чем версии выпуска весны 95 держат файлы открытыми в течении нескольких секунд, более старые могут держать файл открытым в течении нескольких дней.

4. Мои файлы на NFS все считаются с правом только на чтение

По умолчанию сервер NFS для Linux выдается все как только для чтения. Перечитайте разделы "Mountd и nfsd" и "Экспортирование файловых систем" данного документа, а также справочные страницы по "exports" и nfsd. Вам необходимо изменить файл /etc/exports.

5. Я монтирую файловую систему с сервера nfs под linux и пока работает команда ls я не могу читать или записывать файлы.

На старых версиях Linux вы должны монтировать сервер NFS с опциями rsize=1024, wsize=1024.

6. Я монтирую файловую систему с сервера NFS под Linux с размером блока между 3500-4000 и это регулярно роняет машину с Linux

Обычно не делайте так. Это не случается с ядрами версий 2.0 и 2.2. Также нет проблемы с ядрами серии 2.1.

7. Может Linux выполнять NFS по TCP Нет

8. Я получаю странные ошибки при монтировании машины с машины под Linux. Убедитесь, что ваш пользователь находится в 8 или меньшем количестве групп. Старые сервера требуют этого.

9. Когда я перезагружаю свою машину она иногда вешается при попытке отмонтироваться от зависшего сервера NFS. **Не** отмонтируйтесь от серверов NFS при перезагрузке или выключении, просто проигнорируйте это, ничто не повредится, если вы не отмонтируетесь от него. Команда будет выглядеть следующим образом `umount -avt nonfs`.

10. Клиент NFS для Linux работает очень медленно при записи на системы Sun и BSD.

Обычно NFS записывает в синхронном режиме (вы можете запретить это, если вы считаете, что вы не рискуете потерять данные). Хуже всего то, что ядра произошедшие от BSD не могут работать с маленькими блоками. Таким образом когда вы пишете 4К данных с машины под Linux в 1К пакетах, то BSD выполняет это следующим образом

```
прочитать страницу размером 4К
изменить 1К
записать страницу размером 4К обратно на диск
прочитать страницу размером 4К
изменить 1К
записать страницу размером 4К обратно на диск
и т.д...
```

11. Когда я подключаю много клиентов к Linux NFS серверу, его производительность неожиданно падает.

Протокол NFS использует фрагментированные UDP-пакеты. В ядре имеется предел на то, как много фрагментов или неполных пакетов прибывает до того, как оно начнет отбрасывать пакеты. В ядрах серии 2.2 это настраивается во время работы через файловую систему `/proc`: `/proc/sys/net/ipv4/ipfrag_high_thresh` и `ipfrag_low_thresh`. В ядрах серии 2.0 эти константы определяются во время компиляции и определены в файле `.../linux/net/ipv4/ip_fragment.c`, `IPFRAG_HIGH_THRESH` и `IPFRAG_LOW_THRESH`. Эти параметры означают, что когда потребление памяти не собранными фрагментами пакетов UDP достигает значения "ipfrag\_high\_thresh" в байтах (по умолчанию 256К в ядрах 2.2.3 и 2.0.36) оно уменьшится до значения "ipfrag\_low\_tresh". Это делается отбрасыванием фрагментов. Это будет выглядеть как почти полная потеря пакетов и если будет достигнута верхняя граница, то производительность вашего сервера сильно уменьшится.

256К достаточно для обслуживания до 30 клиентов. Если у вас 60 клиентов, то увеличьте это значение в 2 раза. И также увеличьте значение нижней границы.

12. Я использую Linux 2.2 (или более поздний) с `knfsd` и я не могу заставить мои машины с AIX, IRIX, Solaris, DEC-Unix, ... монтироваться к нему.

`Knfsd` объявляет, что реализует NFS версии 3. Это не так. Существует опция, которая запрещает ему анонсировать это. Используйте ее. Или вы можете поместить параметр "vers=2" в список опций монтирования на клиенте.

13. Моя машина с AIX 4 не может произвести монтирование моего NFS сервера под Linux. Она сообщает

```
mount: 1831-011 access denied for server:/dir
mount: 1831-008 giving up on:
server:/dir
The file access permissions do not allow the specified action.
```

или что-то подобное.

AIX 4.2 использует зарезервированные порты (<1024) для NFS. AIX 4.2.1 и 4.3 не ограничены резервированными портами. Также AIX 4.2.1 и 4.3 пытаются произвести монтирование используя версию NFS3, затем NFS/TCP, и только потом NFS/UDP.

Добавление строк

---

```
nfsd -o nfs_use_reserved_ports=1
```

---

в конец файла `rc.tcpip` заставит его использовать резервированные порты. (Этот совет был послан Brian Goraka).

## 9 Экспортирование файловых систем

Способ экспортирования файловых систем с помощью NFS не является полностью совместимым между платформами. В этом случае отличаются Linux и Solaris 2. Этот раздел поверхностно перечисляет способы как выполнить эту операцию на большинстве систем. Если ваша система не была перечислена здесь, то посмотрите справочные страницы по вашей операционной системе. Ключевые слова следующие: `nfsd`, system administration tool (утилиты системного администрирования), `rc scripts`, `boot scripts`, `boot sequence`, `/etc/exports`, `exportfs`. Я буду использовать один пример для всего раздела: как экспортировать файловую систему `/mn/eris/local` для машины `arollon` с правами на чтение/запись.

## 9.1 IRIX, HP-UX, Digital-UNIX, Ultrix, SunOS 4 (Solaris 1), AIX

Эти операционные системы используют традиционный формат Sun для экспортирования. В файле `/etc/exports` напишите:

```
/mn/eris/local -rw=apollon
```

Полная документация находится в справочной странице `exports`. После редактирования файла запустите `exportfs -av` для экспортирования файловых систем.

Точный синтаксис команды `exportfs` различается. В некоторых операционных системах вы можете обнаружить, что введенные строки читаются как:

```
/mn/eris/local apollon
```

или даже вот так:

```
/mn/eris/local rw=apollon
```

Я рекомендую быть формальным. Вы рискуете, если следующая версия `exportfs` будет другая и тогда все внезапно перестанет работать.

## 9.2 Solaris 2

Sun полностью переизобрел колесо при разработке Solaris 2. Так что он полностью отличается от других операционных систем. То, что вам нужно сделать — это отредактировать файл `/etc/dfs/dfstab`. В нем вы должны поместить команды организации доступа так, как это описано в справочной странице `share(1M)`. Примерно вот такие строки:

```
share -o rw=apollon -d "Eris Local" /mn/eris/local
```

После редактирования запустите программу `shareall` для экспортирования файловой системы.

## 10 NFS в Linux 2.2

Как я написал, Linux 2.2.12 является текущей версией ядра и для использования NFS в нем может быть нужно будет выполнить немного рутинной работы. Или не нужно.

Каким будет статус NFS в Linux 2.4 я не знаю.

Большим новшеством в Linux 2.2 является поддержка демона `nfs` в ядре, который называется `knfsd`. Этот способ реализации `nfsd` имеет некоторые преимущества, главный из которых — скорость работы. Машина с Linux 2.2 и `knfsd` является солидным сервером `nfs`. Вы все равно можете использовать старый `nfsd` с Linux 2.2, и также существует несколько преимуществ, в основном простота. Если вы используете исходные тексты ядра или двоичные пакеты, созданные кем-то типа RedHat (6.0 и поздние), SuSE (6.1 или более поздние) или каким-то другим профессиональным системным интегратором, то скорее всего они будут поставляться с полной интеграцией "knfsd" в ядро и вы не должны будете беспокоиться. В большинстве случаев. До тех пор пока вы не заходите скомпилировать ядро сами. Если вы используете обычное ядро Linux 2.2 (по крайней мере 2.2.12), то `knfsd` не будет работать.

Для того, чтобы самому заставить это работать вам необходимо взять пакет `knfsd` сделанный Н.Ж. Lus. Этот пакет является набором заплаток и необходимых утилит для ядер серии 2.2, которые Lu сопровождает в свое свободное время. Вы можете взять его с вашего локального зеркала сервера



ядра, или с основного сервера, по адресу <ftp.kernel.org/pub/linux/devel/gcc/>. **Он не предназначен для всеобщего использования.** Если этот пакет вам непонятен, то не пытайтесь использовать его сами. Подождите пока пакеты с ядром не выпустит ваш любимый системный интегратор (например, Red Hat, SuSE или ...).

Также, пожалуйста не посылайте мне вопросы об этом пакете, я не могу помочь вам. У меня нет ни одного сервера с работающим knfsd. Если вы нашли ошибку или пропуск в документации, то пожалуйста напишите мне, я исправлю этот документ и выпущу его снова.

Все еще читаете? Ок. H.J.Lu посылает сообщения о новых версиях своего пакета в список рассылки linux-kernel. Другие сообщения, относящиеся к NFS в версии ядра 2.2 также посылаются туда. Читайте их.

Существует одна интересная вещь, которую необходимо сказать о пакете knfsd. Он анонсирует, что использует NFS версии 3. Однако он не поддерживает эту версию. Существует опция, которую вы можете использовать для того, чтобы запретить пакету анонсирование NFS3, или на клиентах необходимо указать опцию "vers=2" среди других опций монтирования.

## 10.1 Клиент

Клиент очень прост. Для того, чтобы получить правильное блокирование вам необходимо иметь statd (из пакета knfsd) скомпилированным, установленным и запущенным из загрузочных скриптов. Сделайте это. Для работы statd необходим каталог /var/lib/nfs, иначе он просто прекратит работу без каких либо сообщений об ошибках, так что необходимо создать каталог до запуска программы.

Когда statd уже запущен вы можете использовать программу testlk (в каталоге tools/locktest) для тестирования того, что блокировка файлов работает на файловых системах смонтированных по NFS. Она должна работать нормально. Если программа сообщает *No locks available (Блокировки не доступны)*, то statd не работает.

В действительности вы также можете полностью избежать блокировок (заметьте, что я не рекомендую вам делать это), используя параметр "nolock" в списке опций монтирования.

Насколько я знаю — это все, что необходимо для работы клиентов.

Если у вас Sparc или Alpha NFS-сервер, то вы обнаружите, что nfs-клиент в Linux 2.2 совершенное не работает. Скорость передачи данных на и с сервера настолько плоха, что это трудно представить. Это хуже даже чем под Linux 2.0. Намного хуже. Но для этой проблемы существует исправление. Серия ядер 2.2 Alan Cox (которые являются более экспериментальными, чем нормальные ядра 2.2, сопровождаемые Linus) включают заплатку, которая позволяет Linux 2.2 увеличить производительность при работе с серверами Alpha и Sparc. Если вы хотите использовать ядра исправленные Alan Cox, то вы должны читать список рассылки linux-kernel, и вы должны узнать где можно найти необходимые заплатки. Основным сервером для этой заплатки является сервер <<http://www.uio.no/~trondmy/src/>>, в случае, если вы хотите попробовать применить его к нормальному ядру серии 2.2. Эта заплатка скорее всего не будет входит в состав Linux 2.4, поскольку требует сделать слишком много изменений в течении текущего цикла разработки. Ждите появления Linux 2.5.

trondmy также имеет заплатки для того, чтобы Linux использовал NFS версии 3, они также позволят вам использовать tcp как транспортный механизм, вместо UDP. NFSv3 очень хорош для сетей с большим количеством переходов, а также для сетей где потери пакетов не равны нулю или где время ожидание очень высокое.

Вы должны читать список рассылки linux-kernel, если вы собираетесь использовать эти заплатки, поскольку время от времени в них находят какую-нибудь неприятные ошибки. Ошибки, которые портят ваши файлы. Так что пожалуйста будьте **осторожны**.

## 10.2 Сервер

Демон nfs-сервера в Linux 2.2 и более поздних называется "knfsd". Он сложен в установке. Вы можете настроить его сами или поставить то, что вам предлагают SuSE, Red Hat и другие в виде

пакетов ядра серии 2.2. Извините. Хотя вы все равно можете использовать старый `nfsd` в Linux 2.2. Он медленен, но легок в установке.

## 11 NFS-сервер на дискете

Этот раздел был написан Ron Peters, [rpeters@hevanet.com](mailto:rpeters@hevanet.com). Он объясняет как настроить NFS-сервер при загрузке с дискеты. Сначала это было придумано для обеспечения доступа по NFS к `cdrom` на другой машине без Linux/UNIX для установки Linux на машину на которой нет `cdrom`.

### 11.1 Введение

Этот документ был написан для тех, кто столкнется с той же проблемой, с которой я столкнулся недавно. Я устанавливал сервер с Linux на машину на которой не было `cdrom` и не было возможности добавления его, за исключением внешних приводов. Сейчас все меньше и меньше вероятность того, что вы будете устанавливать такие машины, так что этот документ перестанет быть ценным для вас. Однако я оценил его по достоинству, когда я попробовал настроить свою машину.

Поскольку на моей машине не было `cdrom`, я попробовал найти NFS-сервер для Win95 и организовать доступ к `cdrom`, достаточный для установки машины. Я нашел две программы (я не упоминаю их названий, хотя одна из них распространялась свободно, а вторая имела 14-дневную ограниченную лицензию), но одна программы не работала, а вторая не смогла обработать правильно имена Linux, так что я не смог выполнить установку.

Затем я попробовал загрузить мою машину с Win95 используя набор `boot/root` дисков и затем использовать дискеты из поставки для настройки NFS-сервера.

Это было замечательно просто и процедура вероятно была более легкой, чем чтение этого введения, но я верю, что помещение описания всей процедуры в одно место будет лучшим решением.

### 11.2 Ожидания

Этот документ был написан, используя набор `boot/root` дисков от одного из дистрибутивов Slackware. Я использовал ядро версии 2.0.34 для `boot/root` дисков, но программы для NFS-сервера были взяты от сервера из версии 2.0.30. Я всегда использую методику установки от Slackware, не потому что она легче или лучше или хуже, просто я чувствую себя удобно при его использовании, и у меня нет времени пробовать другие методы.

Я не верю, что будет много проблем с использованием других версий ОС. Я хочу рекомендовать вам использовать текущие версии. Поскольку вероятно, что они будут использоваться для установки, то будут использоваться и наборы `boot/root` дисков от них.

### 11.3 Требования

- Система умеющая работать с сетью и загрузочный диск. Система, которая будет работать как NFS-сервер должна иметь сетевую карту, и она должна определяться в процессе загрузки. Больше информации вы найдете в Networking HOWTO.
- Дополнительная дискета, которая содержит `rpc.portmap`, `rpc.mountd` и `rpc.nfsd`. Эти файлы легко можно найти используя `ftpsearch`.
- Носитель (предполагается `cd`) со Slackware (или другим дистрибутивом).

### 11.4 Установка сервера

#### 11.4.1 Загрузка временного NFS-сервера

Загрузите систему, которая будет работать как NFS-сервер с загрузочной дискеты и убедитесь, что сетевая карта обнаружена. Также необходимо, чтобы CDROM также был обнаружен ядром. Я

буду использовать eth0 в качестве примера имени сетевой карты.

#### 11.4.2 Смонтируйте дисковод и cdrom

После загрузки системы boot/root дискеты вам больше не нужны. Система полностью находится в ОЗУ.

Замените дискету с корневой файловой системой на другую. Смонтируйте ее:

```
mount /dev/fd0 /floppy
```

Предполагается что на дискете находится файловая система ext2. Я могу представить, что диск поддержки может иметь формат дискет DOS, но я не пробовал это. Я В этом случае вы должны выполнить `mount -t msdos ...etc`. Вероятно это надо поместить в раздел todo.

Смонтируйте cdrom:

```
mount -t iso9660 /dev/hdc /cdrom
```

Я указал Устройства, которые я использовал сам. Они могут быть разными в зависимости от приложений. Точки монтирования /floppy и /cdrom существуют на корневой дискете, так что они могут быть использованы. Если их нет, то создайте их или используйте другие точки монтирования.

#### 11.4.3 Настройка сети на временном сервере

Здесь мы опишем как настроить сеть для работы временного NFS-сервера. Необходимо запустить несколько команд. Нужно только немного информации для настройки (приведенные значения являются примерами):

```
IPADDR:172.16.5.100 #Адрес временного сервера
```

```
NETMASK:255.255.255.0 #Маска сети
```

```
BROADCAST:172.16.5.255 #Последний номер (255) отличается от IPADDR.
```

```
ETHNETWORK:172.16.5.0 #еще раз, полностью отличное от IPADDR значение.
```

```
GATEWAY:172.16.5.251 #необходим только, если вы используете шлюз. Вы скорее всего знаете этот адрес.
```

Выполните следующие команды. Вставьте в них вышеприведенные значения:

```
ifconfig eth0 inet IPADDR arp netmask NETMASK broadcast BROADCAST
```

```
route add -net ETHNETWORK netmask NETMASK eth0
```

Используйте следующую команду только если вы используете шлюз и вам нужно работать через него:

```
route add default gw GATEWAY netmask 0.0.0.0 eth0
```

Если все идет нормально, то ваша сеть заработает и вы сможете увидеть другие машины

#### 11.4.4 Настройка доступа по NFS.

Определите каталог к которому вы хотите предоставить доступ через NFS. В моем случае я использовал каталог /cdrom/slakware. Поместите этот каталог в файл /etc/exports:

```
echo "/cdrom/slakware"> /etc/exports
```

#### 11.4.5 Запуск сервера NFS

Перейдите в каталог /floppy/usr/sbin и запустите:

```
./rpc.portmap
```

```
./rpc.mountd
```

```
./rpc.nfsd
```

### 11.4.6 Готово, запускайте процесс установки

Описанная процедура должна предоставить вам доступ к каталогу `"/cdrom/slakware"`. После того, как все сделано вы можете загрузить машину на которую вы хотите установить систему и запустить процесс установки.

Когда вы перейдете к выбору носителя дистрибутива, выберите NFS-сервер. Вас спросят об адресе сервера. Задайте тот адрес, который вы использовали при настройке сервера. Вас также спросят о том, какой каталог необходимо монтировать. Укажите нужный каталог (тот, который вы указали в файле `/etc/exports` на NFS-сервере).

Система сможет подключиться к NFS-серверу. Посмотрите есть ли сообщения об ошибках. Все должно выполниться нормально и вы сможете продолжить процесс установки.

## 11.5 Разрешение проблем

### 11.5.1 Еще ничего нет

У меня еще нет никаких сообщений о проблемах. Возможно когда люди будут использовать эту процедуру, то появится больше разных полезных советов и приемов.

## 11.6 Что сделать

### 11.6.1 DOS Disk.

Проверить возможность использования дискеты в формате DOS как диска поддержки.

### 11.6.2 Команды `grc`.

Проверить порядок запуска команд `grc.*` и все ли команды надо запускать или только некоторые.

## 12 PC-NFS

Вы не должны запускать PC-NFS. Вы должны запустить `samba`.

`Samba` намного лучше, чем PC-NFS и работает с Windows 3 для Рабочих групп, а также более поздними версиями Windows. Она более быстрая и безопасная. Так что используйте ее.