FreeBSD и твердотельные устройства

John Kozubik <john@kozubik.com >

Издание: 43234

Авторские права © 2001, 2009 The FreeBSD Documentation Project

FreeBSD это зарегистрированная торговая марка FreeBSD Foundation.

Многие из обозначений, используемые производителями и продавцами для обозначения своих продуктов, заявляются в качестве торговых марок. Когда такие обозначения появляются в этом документе, и Проекту FreeBSD известно о торговой марке, к обозначению добавляется знак «™» или «®».

Распространение и использование исходных (SGML DocBook) и «скомпилированных» форм (SGML, HTML, PDF, PostScript, RTF и прочих) с модификацией или без оной, разрешены при соблюдении следующих соглашений:

1. Распространяемые копии исходного кода (SGML DocBook) должны сохранять вышеупомянутые объявления copyright, этот список положений и следующий отказ от ответственности в первых строках этого файла в неизменном виде.

2. Распространяемые копии скомпилированных форм (преобразованные в другие DTD, конвертированные в PDF, PostScript, RTF и другие форматы) должны повторять вышеупомянутые объявления copyright, этот список положений и следующий отказ от ответственности в документации и/или других материалах, поставляемых с дистрибьюцией.

Важно

ЭТА ДОКУМЕНТАЦИЯ ПОСТАВЛЯЕТСЯ ПРОЕКТОМ ДОКУМЕНТАЦИИ FREEBSD "КАК ЕСТЬ" И ЛЮБЫЕ ЯВНЫЕ ИЛИ НЕЯВНЫЕ ГАРАНТИИ, ВКЛЮЧАЯ, НО НЕ ОГРАНИЧИВАЯСЬ НЕЯВНЫМИ ГАРАНТИЯМИ, КОММЕРЧЕ-

СКОЙ ЦЕННОСТИ И ПРИГОДНОСТИ ДЛЯ КОНКРЕТНОЙ ЦЕЛИ ОТРИЦАЮТСЯ. НИ ПРИ КАКИХ УСЛОВИЯХ ПРОЕКТ ДОКУМЕНТИРОВАНИЯ FREEBSD НЕ НЕСЕТ ОТВЕТСТВЕННОСТИ ЗА ЛЮБОЙ ПРЯМОЙ, КОСВЕННЫЙ, СЛУЧАЙНЫЙ, СПЕЦИАЛЬНЫЙ, ОБРАЗЦОВЫЙ ИЛИ ПОСЛЕДУЮЩИЙ УЩЕРБЫ (ВКЛЮЧАЯ, НО НЕ ОГРАНИЧИВАЯСЬ ПОСТАВКОЙ ТОВАРОВ ЗАМЕНЫ ИЛИ УСЛУГ; ПОТЕРЮ ДАННЫХ ИЛИ ИХ НЕПРАВИЛЬНУЮ ПЕРЕДАЧУ ИЛИ ПОТЕРИ; ПРИОСТАНОВЛЕНИЕ БИЗНЕСА), И ТЕМ НЕ МЕНЕЕ ВЫЗВАННЫЕ И В ЛЮБОЙ ТЕОРИИ ОТВЕТСТВЕННОСТИ, НЕЗАВИСИМО ОТ КОНТРАКТНОЙ, СТРОГОЙ ОТВЕТСТВЕННОСТИ, ИЛИ ПРАВОНАРУШЕНИИ (ВКЛЮЧАЯ ХАЛАТНОСТЬ ИЛИ ИНЫМ СПОСОБОМ), ВОЗНИКШЕМ ЛЮБЫМ ПУТЕМ ПРИ ИСПОЛЬЗОВАНИИ ЭТОЙ ДОКУМЕНТАЦИИ, ДАЖЕ ЕСЛИ БЫ БЫЛО СООБЩЕНО О ВОЗМОЖНОСТИ ТАКОГО УЩЕРБА.

2013-11-24 taras.

Аннотация

В этой статье описывается использование твердотельных дисковых устройств для создания встраиваемых систем на основе FreeBSD

Встраиваемые системы имеют преимущество в повышенной надежности по причине отсутствия в них движущихся частей (жестких дисков). Однако, следует принять во внимание, что системе, как правило, доступно очень малое дисковое пространство и ограниченный объем запоминающего устройства.

К отдельно рассматриваемым вопросам относятся типы и характеристики твердотельных носителей, подходящих для использования в качестве дисков во FreeBSD, параметры ядра, которые представляют интерес в таких условиях, механизмы `rc.initdiskless` , автоматизирующие инициализацию таких систем и удовлетворяющие требованиям файловых систем, доступных только для чтения, а также построение файловых систем с нуля. Статья заканчивается описанием некоторых общих стратегий для случаев малых систем FreeBSD и работ в режиме только для чтения.

Содержание

1. Твердотельные дисковые устройства 3
2. Параметры ядра 4
3. Подсистема rc и файловые системы в режиме только чтения 4
4. Создание файловой системы с нуля 5
5. Стратегии работы с системой для случаев небольших и доступных только для чтения файловых систем 7

1. Твердотельные дисковые устройства

Эта статья будет ограничиваться рассмотрением твердотельных дисковых устройств, которые делаются на основе флэш-памяти. Флэш-память является твердотельным (здесь нет движущихся частей) запоминающим устройством, которое является энергонезависимым (данные остаются в памяти даже после отключения всех источников питания). Флэш-память может быть нечувствительной к сильным физическим воздействиям и достаточно быстра (решения на основе флэш-памяти, описываемые в этой статье, гораздо медленнее, чем диски EIDE для операций записи, и гораздо быстрее их в случае выполнения операций чтения). Одним из очень важных свойств флэш-памяти, различные варианты которого будут рассмотрены далее в этой статье, является то, что каждый сектор имеет ограниченные возможности по перезаписыванию. Вы можете только записывать, стирать и снова записывать на сектор флэш-памяти определенное количество раз до того, как сектор станет полностью неработоспособным. Хотя многие продукты на основе флэш-памяти автоматически перенаправляют испорченные блоки, а некоторые даже распределяют операции записи по всему модулю, фактом является наличие ограничения на количество операций записи, которые могут выполняться с устройством. Современные модули имеют характеристики от 1,000,000 до 10,000,000 циклов записи на сектор. Эти характеристики могут зависеть от температуры рабочей среды.

В частности, мы обсудим компактные модули флэш-памяти, совместимые со стандартом ATA, которые стали весьма популярными в качестве носителя данных для цифровых камер. Особый интерес представляет тот факт, что они соответствуют шине IDE по контактам и совместимы с набором команд ATA. Таким образом, при помощи очень простого и дешевого адаптера такие устройства могут подключаться непосредственно к шине IDE компьютера. Если поступить таким образом, то такие операционные системы, как FreeBSD, распознают диск как обычный винчестер (весьма маленький).

Существуют и другие решения для твердотельных дисков, но их стоимость, безвестность и сравнительная сложность использования выводят их за рамки этой статьи.

2. Параметры ядра

Для тех, кто создает встраиваемую систему FreeBSD, интерес представляют несколько параметров ядра.

Все встраиваемые системы FreeBSD, которые используют флэш-память в качестве системного диска, заинтересованы в использовании дисков в памяти и файловых систем в памяти. Из-за ограниченного количества циклов записи, которые можно выполнить с флэш-памятью, диск и файловые системы на нем будут, скорее всего, монтироваться в режиме доступа только для чтения. В таком случае файловые системы типа `/tmp` и `/var` монтируются как файловые системы в памяти для того, чтобы позволить системе создать журналы и обновить счетчики и временные файлы. Файловые системы в памяти являются критическим компонентом успешной работы FreeBSD на твердотельных устройствах.

Вы должны удостовериться, что в конфигурационном файле вашего ядра присутствуют следующие строки:

```
options         MFS             # Memory Filesystem
options         MD_ROOT         # md device usable as a potential ↺
root device
pseudo-device   md              # memory disk
```

3. Подсистема `rc` и файловые системы в режиме только чтения

Инициализация встраиваемой системы FreeBSD после загрузки управляется `/etc/rc.initdiskless` .

`/etc/rc.d/var` монтирует `/var` как файловую систему в памяти, создает указываемый список каталогов в `/var` при помощи команды mkdir(1), изменяет режимы доступа на некоторые из этих каталогов. В процессе выполнения `/etc/rc.d/var` задействуется еще одна переменная `rc.conf` – `varsize`. Скрипт `/etc/rc.d/var` создает раздел `/var` на основе значения этой переменной из `rc.conf`:

```
varsize=8192
```

Запомните, что по умолчанию это значение указано в секторах.

Факт использования файловой системы `/var` в режиме чтения и записи является важным признаком, так как раздел `/` (и любые другие разделы, которые могут находиться на флэш-носителе) должен монтироваться в режиме только для чтения. Вспомните, что в Раздел 1, «Твердотельные дисковые устройства» мы касались ограничений флэш-памяти - особенно ограничений, касающихся возможностей записи. Важно не монтировать файловые системы на флэш-носителях в режимах чтения и записи, и важность отказа от файла подкачки не может быть переоценена.

Файл подкачки на загруженной системе может пережечь кусок флэш-носителя менее чем за год. Частое журналирование и создание временных файлов приводят к тому же результату. Поэтому, кроме удаления записи `swap` из вашего файла `/etc/fstab`, вы должны также изменить поле параметров каждой файловой системы на `ro` таким образом:

```
# Device                Mountpoint      FStype  Options         ↺
Dump    Pass#
/dev/ad0s1a             -/              ufs     ro              ↺
1       1
```

В результате этих изменений в среднестатистической системе несколько приложений немедленно перестанут работать. Например, cron не будет нормально запускаться в результате отсутствия таблиц для него в каталоге `/var`, созданном `/etc/rc.d/var` , а syslog и dhcp будут испытывать проблемы из-за доступа файловой системы только для чтения, а также отсутствия записей в `/var`, который был создан скриптом `/etc/rc.d/var` . Хотя эти проблемы являются временными и обсуждаются вместе с решением проблем с запуском распространенных программных пакетов, в Раздел 5, «Стратегии работы с системой для случаев небольших и доступных только для чтения файловых систем».

Важно помнить, что файловая система, которая была смонтирована только для чтения при помощи файла `/etc/fstab` , в любой момент может быть сделана доступной по чтению и записи выдачей команды:

```
# /sbin/mount --uw partition
```

и может быть возвращена к режиму доступа только для чтения по такой команде:

```
# /sbin/mount --ur partition
```

4. Создание файловой системы с нуля

Так как совместимые с ATA компактные флэш-карты распознаются во FreeBSD как обычные жесткие диски IDE, то теоретически вы можете установить FreeBSD по сети при помощи дискет kern и mfsroot или с компакт-диска.

Однако даже маленькая установка FreeBSD при помощи обычных процедур установки может привести к созданию системы размером, превышающим 200 мегабайт. Так как большинство людей используют устройства флэш-памяти меньшего размера (128 мегабайт считается весьма большим - 32 или даже 16 мегабайт используются гораздо чаще), то установка обычным образом не подходит—просто на диске нет места даже для самой минимальной установки.

Самым простым способом обойти это ограничение на объем является установка FreeBSD обычным образом на обычный жесткий диск. После окончания установки, обрежьте операционную систему до размера, который помещается на ваш фл-

эш-носитель, а затем полностью заархивируйте файловую систему. Следующие шаги поведут вас через процесс подготовки части флэш-памяти для вашей заархивированной файловой системы. Запомните, что из-за того, что обычная установка не выполнялась, такие операции, как разбиение на разделы, разметка, создание файловой системы и так далее должны быть выполнены вручную. Кроме дискет kern и mfsroot вам также нужно воспользоваться дискетой fixit.

1. Разбиение вашего флэш-носителя на разделы

   После загрузки при помощи дискет kern и mfsroot, выберите пункт `custom` из меню установки. Из следующего пункта меню выберите `partition` . В меню работы с разделами вы должны удалить все существующие разделы при помощи клавиши d. После удаления всех имеющихся разделов создайте раздел при помощи клавиши c и согласитесь с предлагаемым по умолчанию размером раздела. Когда вы будете опрошены на предмет типа раздела, удостоверьтесь, что значение типа равно `165`. Теперь запишите эту таблицу разделов на диск, нажав клавишу w (на этом экране эта опция скрыта). Если вы используете компактную флэш-карту, совместимую с ATA, вы должны выбрать FreeBSD Boot Manager. Теперь нажмите клавишу q для выхода из меню работы с разделами. Должно быть выдано еще раз меню для выбора менеджера загрузки - повторите то, что вы выбирали ранее.

2. Создание файловых систем на вашем устройстве флэш-памяти

   Выйдите из меню установки custom, и из главного меню установки выберите пункт `fixit`. После входа в режим работы fixit, введите следующую команду:

   ```
   # disklabel --e -/dev/ad0c
   ```

   В этот момент вы войдете в редактор vi из-под команды disklabel. Затем, вам нужно добавить строку `a:` в конце файла. Эта строка `a:` должна выглядеть примерно так:

   ```
   a:      123456   0       4.2BSD  0       0
   ```

   Здесь *123456* является числом, в точности совпадающим с тем, что характеризует размер имеющейся записи для `c:`. В общем, вы копируете существующую строку для `c:` для строки `a:`, не забывая определить fstype как `4.2BSD`. Сохраните файл и завершите редактирование.

   ```
   # disklabel --B --r -/dev/ad0c
   # newfs -/dev/ad0a
   ```

3. Размещение вашей файловой системы на флэш-носителе

   Смонтируйте только что подготовленный флэш-носитель:

   ```
   # mount -/dev/ad0a -/flash
   ```

Подключите эту машину к сети, чтобы можно было перенести наш tar-файл и распаковать его в файловую систему на флэш-носителе. Вот пример того, как это можно сделать:

```
# ifconfig xl0 192.168.0.10 netmask 255.255.255.0
# route add default 192.168.0.1
```

Теперь, когда машина находится в сети, перепишите ваш tar-файл. Здесь вы можете столкнуться с некоторой проблемой - если объем вашей флэш-памяти равен, к примеру, 128 мегабайтам, а ваш tar-файл превышает 64 мегабайта, то вы не можете одновременно разместить tar-файл на флэш-носителе и распаковать его - вам не хватит места. Одним из решений этой проблемы, если вы используете FTP, является распаковка файла во время его передачи по FTP. Если вы передаете файл именно так, то вы никогда не получите на диске одновременно архивный файл и его содержимое:

```
ftp> get tarfile.tar -"| tar xvf --"
```

Если ваш файл обработан утилитой gzip, вы также можете этого добиться:

```
ftp> get tarfile.tar -"| zcat -| tar xvf --"
```

После того, как вы получили содержимое вашей заархивированной файловой системы на файловой системе флэш-памяти, вы можете размонтировать флэш-память и выполнить перезагрузку:

```
# cd -/
# umount -/flash
# exit
```

Полагая, что вы правильно настроили вашу файловую систему при ее построении на обычном диске (с вашей файловой системой, смонтированной в режиме доступа только для чтения, и необходимыми параметрами, присутствующими в ядре) вы должны успешно загрузить вашу встраиваемую систему на основе FreeBSD.

5. Стратегии работы с системой для случаев небольших и доступных только для чтения файловых систем

В Раздел 3, «Подсистема rc и файловые системы в режиме только чтения» было указано, что файловая система `/var`, создаваемая скриптом `/etc/rc.d/var` , и наличие корневой файловой системы, доступной только для чтения, приводят к проблемам при работе многих распространенных программных пакетов, использу-

мых во FreeBSD. В этой статье будут даны рекомендации по настройке нормальной работы cron и syslog, установке портов и веб-сервера Apache.

5.1. cron

Во время загрузки содержимое каталога `/var` формируется скриптом `/etc/rc.d/var` используя данные из `/etc/mtree/BSD.var.dist` , поэтому в нем создается несколько стандартных каталогов, в числе которых — `cron`, `cron/tabs` , `at`.

Однако это не решает проблему с сохранением cron-таблиц между перезагрузками. Когда система перезагружается, то файловая система `/var`, которая располагается в памяти, будет уничтожена, вместе со всеми cron-таблицами, которые вы могли там иметь. Поэтому одним из решений может стать создание cron-таблиц для пользователей, которым они нужны, монтирование вашей файловой системы `/` в режиме чтения и записи, и копирование этих cron-таблиц в безопасное место, например, в `/etc/tabs` , и последующее добавление строки в конец скрипта `/etc/rc.initdiskless` для копирования этих cron-таблиц в каталог `/var/cron/tabs` после его создания во время инициализации системы. Вам может также потребоваться добавить строку, которая изменяет режимы доступа и права на каталоги, которые вы создали, и на файлы, которые вы скопировали в скрипте `/etc/rc.initdiskless` .

5.2. syslog

В файле `syslog.conf` задано местоположение некоторых файлов протоколов, которые имеются в каталоге `/var/log` . Эти файлы не создаются скриптом `/etc/rc.d/var` во время инициализации системы. Поэтому где-нибудь в скрипте `/etc/rc.d/var`, после секции, создающей каталоги в `/var`, вам нужно добавить нечто вроде следующего:

```
# touch -/var/log/security -/var/log/maillog -/var/log/cron -/var/log/
messages
# chmod 0644 -/var/log/*
```

5.3. Установка портов

Перед тем, как обсудить изменения, которые нужно сделать для успешного использования дерева портов, необходимо напомнить о том, что ваши файловые системы на флэш-носителях доступны только для чтения. Поэтому вам нужно временно монтировать их в режиме чтения и записи, используя параметры командной строки, как это показано в Раздел 3, «Подсистема rc и файловые системы в режиме только чтения». Вы всегда должны перемонтировать эти файловые системы в режим только для чтения после окончания работ - излишние записи на флеш носитель могут значительно сократить его срок эксплуатации.

Чтобы можно было войти в каталог с портами и успешно выполнить команду `make install` , необходимо создать каталог для пакаджей в файловой системе, не распо-

лагающейся в памяти, где будут храниться пакаджи между перезагрузками. Так как для установки пакаджа в любом случае требуется монтирование ваших файловых систем для чтения и записи, имеет смысл выделить область флэш-носителя также и для записи информации о пакадже.

Прежде всего создайте каталог с базой данных о пакаджах. Обычно это каталог `/var/db/pkg`, но мы не можем разместить базу именно здесь, так как она исчезнет после перезагрузки системы.

```
# mkdir -/etc/pkg
```

Теперь в скрипт `/etc/rc.d/var` добавьте строку, которая связывает каталог `/etc/pkg` с `/var/db/pkg`. Например:

```
# ln --s -/etc/pkg -/var/db/pkg
```

Теперь каждый раз при монтировании ваших файловых систем для чтения и записи и установки пакаджа, команда `make install` будет работать, а информация о пакадже будет успешно записана в каталог `/etc/pkg` (так как файловая система будет в это время смонтирована для чтения и записи), который всегда будет доступным операционной системе как `/var/db/pkg`.

5.4. Веб-сервер Apache

Примечание

Шаги, описанные в этой части статьи, необходимо выполнить лишь в том случае, если Apache настроен сохранять свой pid или лог файл вне каталога `/var`. С настройками по умолчанию Apache формирует свой pid файл в `/var/run/httpd.pid`, а лог файлы — в `/var/log`.

Далее в статье подразумевается, что Apache сохраняет свои лог файлы в каталог `apache_log_dir` вне каталога `/var`. Когда этот каталог расположен на файловой системе, смонтированной в режиме только для чтения, Apache не сможет сохранять лог файлы, что в свою очередь может вызывать проблемы в работе веб-сервера. В таком случае необходимо добавить новый каталог к списку каталогов из `/etc/rc.d/var` для их создания в каталоге `/var` и связать `apache_log_dir` с `/var/log/apache`. Нужно также задать права доступа и владельца нового каталога.

Сначала добавьте каталог `log/apache` к списку каталогов, создаваемых скриптом `/etc/rc.d/var`.

Затем добавьте в скрипт `/etc/rc.d/var` после секции создания каталогов такие команды:

```
# chmod 0774 -/var/log/apache
# chown nobody:nobody -/var/log/apache
```

И наконец, удалите существующий каталог `apache_install/logs` и замените его ссылкой:

```
# rm --rf apache_log_dir
# ln --s apache_log_dir
```