

Работа с бездисковыми станциями

Текст обновил *Jean-François DockXs*.

Машина с FreeBSD может загружаться по сети и работать без наличия локального диска, используя файловые системы, монтируемые с сервера NFS. Кроме стандартных конфигурационных файлов, не нужны никакие модификации в системе. Такую систему легко настроить, потому что все необходимые элементы уже готовы:

- Имеется по крайней мере два возможных способа загрузки ядра по сети:
- *PXE*: Система от Intel (Preboot Execution Environment) является типом загрузочного ПЗУ, встроенного в некоторые сетевые адаптеры или материнские платы. Обратитесь к справочной странице по `pxeboot(8)` для получения более полной информации.
- Порт ***etherboot*** (net/etherboot) генерирует код, который может применяться в ПЗУ для загрузки ядра по сети. Код может быть либо прошит в загрузочный PROM на сетевом адаптере, либо загружен с локальной дискеты (или винчестера), или с работающей системы MS-DOS. Поддерживаются многие сетевые адаптеры.

- Примерный скрипт (`/usr/share/examples/diskless/clone_root`) облегчает создание и поддержку корневой файловой системы рабочей станции на сервере. Скрипт, скорее всего, потребует некоторых настроек, но он позволит вам быстро начать работу.
- Стандартные файлы начального запуска системы, располагающиеся в `/etc`, распознают и поддерживают загрузку системы в бездисковом варианте.
- Подкачка, если она нужна, может выполняться через файл NFS либо на локальный диск.

Существует много способов настройки бездисковой рабочей станции. При этом задействованы многие компоненты, и большинство из них могут быть настроены для удовлетворения ваших вкусов. Далее будет описана полная настройка системы, при этом упор будет делаться на простоту и совместимость с стандартной системой

скриптов начальной загрузки FreeBSD. Описываемая система имеет такие характеристики:

- Бездисковые рабочие станции совместно используют корневую файловую систему в режиме только чтения, а также используют /usr совместно тоже в режиме только чтения.

Корневая файловая система является копией стандартной корневой системы FreeBSD (обычно сервера), с некоторыми настроечными файлами, измененными кем-то специально для бездисковых операций или, возможно, для рабочей станции, которой она предназначена.

Части корневой файловой системы, которые должны быть доступны для записи, перекрываются файловыми системами mfs(8). Любые изменения будут потеряны при перезагрузках системы.

- Ядро загружается посредством **etherboot**, при помощи DHCP (или BOOTP) и TFTP.

Caution Как описано, эта система незащищена. Она должна располагаться в защищенной части сети, а другие хосты не должны на нее полагаться.

17.5.1. Инструкции по настройке

17.5.1.1. Настройка DHCP/BOOTP

Есть два протокола, часто используемых для загрузки рабочих станций, которые запрашивают свою конфигурацию по сети: BOOTP и DHCP. При начальной загрузке рабочей станции они используются в нескольких местах:

- **etherboot** использует (по умолчанию) DHCP или BOOTP (требуется опция при

настройке) для поиска ядра. (PXE использует DHCP).

- Ядро использует BOOTP для поиска корневой файловой системы в NFS.

Возможна настройка системы на использование исключительно BOOTP. Программа сервера bootpd(8) включена в базовую систему FreeBSD.

Однако DHCP имеет некоторый набор преимуществ перед BOOTP (лучше настроечные файлы, возможность использования PXE, плюс множество других, напрямую не связанных к работе без диска), и мы опишем как конфигурацию с чистым BOOTP, так и BOOTP+DHCP, с упором на последнее, в котором используется пакет DHCP от ISC.

17.5.1.1.1. Конфигурация с использованием ISC DHCP

Сервер **isc-dhcp** может обрабатывать как запросы BOOTP, так и запросы DHCP.

Начиная с релиза 4.4, **isc-dhcp 3.0** не включается в поставку системы. Сначала вам нужно будет установить порт net/isc-dhcp3 или соответствующий пакадж. Пожалуйста, обратитесь к [Chapter 4](#) для получения общего представления о портах и пакаджах.

После установки **isc-dhcp** ему для работы требуется конфигурационный файл (обычно называемый `/usr/local/etc/dhcpd.conf`). Вот прокомментированный пример:

```
default-lease-time 600;
```

```
max-lease-time 7200;
```

authoritative;

option domain-name "example.com";

option domain-name-servers 192.168.4.1;

option routers 192.168.4.1;

subnet 192.168.4.0 netmask 255.255.255.0 {

use-host-decl-names on;

option subnet-mask 255.255.255.0;

option broadcast-address 192.168.4.255;

host margaux {

```
hardware ethernet 01:23:45:67:89:ab;
```

```
fixed-address margaux.example.com;
```

```
next-server 192.168.4.4;
```

```
filename "/tftpboot/kernel.diskless";
```

```
option root-path "192.168.4.4:/data/misc/diskless";
```

```
}
```

```
}
```

Этот параметр указывает dhcpd посылать значения деклараций `host` как имя хоста для бездисковой машины. Альтернативным способом было бы добавление `option host-name margaux` внутри объявлений хоста.

Директива `next-server` определяет сервер TFTP (по умолчанию используется тот же самый хост, на котором расположен сервер DHCP).

Директива `filename` определяет файл, который будет загружать **etherboot** в качестве ядра.

Note: Похоже, что PXE предпочитает относительное имя файла, и он загружает `pxeboot`, а не ядро (`option filename "pxeboot"`).

Параметр `root-path` определяет путь к корневой файловой системе, в обычной нотации NFS.

17.5.1.1.2. Настройка с использованием BOOTP

Далее описана эквивалентная конфигурация с использованием `bootpd`. Она будет располагаться в `/etc/bootptab`.

Пожалуйста, отметьте, что **etherboot** должен быть откомпилирован с нестандартной опцией `NO_DHCP_SUPPORT` для того, чтобы можно было использовать BOOTP, и что для работы PXE *необходим DHCP*. Единственным очевидным преимуществом

bootpd

является его наличие в поставке системы.

.def100:

:hn:ht=1:sa=192.168.4.4:vm=rfc1048:

:sm=255.255.255.0:

:ds=192.168.4.1:

:gw=192.168.4.1:

:hd="/tftpboot":

:bf="/kernel.diskless":

:rp="192.168.4.4:/data/misc/diskless":

margaux:ha=0123456789ab:tc=.def100

17.5.1.2. Подготовка программы загрузки при помощи Etherboot

[Сайт Etherboot](#) содержит [подробную документацию](#), в основном предназначенную для систем Linux, но несомненно, она полезна. Далее будет просто кратко описано, как вы должны использовать **etherboot** в системе FreeBSD.

Сначала вы должны установить пакадж или порт [net/etherboot](#). Порт **etherboot** обычно расположен в /usr/ports/net/etherboot. Если в вашей системе установлено дерево портов, просто наберите make в этом каталоге, все остальное будет сделано за вас. Либо обратитесь к

[Chapter 4](#)

для получения информации о портах и пакаджах.

В нашей ситуации мы будем использовать загрузочную дискету. Для других методов (PROM или программа DOS) пожалуйста, обратитесь к документации по **etherboot**.

Для создания загрузочной дискеты, вставьте дискету в дисковод на машине, где установлен **etherboot**, затем перейдите в каталог src в дереве **etherboot** и наберите:

```
# gmake bin32/devicetype.fd0
```

devicetype зависит от типа адаптера Ethernet на бездисковой рабочей станции. Обратитесь к файлу NIC в том же самом каталоге для определения правильного значения для *devicetype*.

17.5.1.3. Настройка серверов TFTP и NFS

Вам нужно включить tftpd на сервере TFTP:

1. Создайте каталог, в котором tftpd будет работать с файлами, то есть: /tftpboot
2. Добавьте в ваш /etc/inetd.conf такую строку:

```
tftp dgram udp wait nobody /usr/libexec/tftpd tftpd /tftpboot
```

Note: Бывает, что некоторым версиям PXE требуется TCP-вариант TFTP. В таком случае добавьте вторую строку, заменяющую dgram udp на stream tcp.

1. Укажите inetd на повторное чтение своего конфигурационного файла:

```
# kill -HUP `cat /var/run/inetd.pid`
```

Вы можете поместить каталог tftpboot в любом месте на сервере. Проверьте, что это местоположение указано как в inetd.conf, так и в dhcpd.conf.

Вам также нужно включить NFS и экспортировать соответствующую файловую систему на сервере NFS.

1. Добавьте следующее в /etc/rc.conf:

```
nfs_server_enable="YES"
```

1. Проэкспортируйте файловую систему, в которой расположен корневой каталог для бездисковой рабочей станции, добавив следующую строку в /etc/exports (подправьте точку монтирования и замените *margaux* на имя бездисковой рабочей станции):

```
/data/misc -alldirs -ro margaux
```

1. Укажите mountd на повторное чтение настроечного файла. На самом деле если вам потребовалось на первом шаге включить NFS в /etc/rc.conf, то вам нужно будет выполнить перезагрузку.

```
# kill -HUP `cat /var/run/mountd.pid`
```

17.5.1.4. Построение ядра для бездисковой рабочей станции

Создайте конфигурационный файл ядра для бездискового клиента со следующими параметрами (вдобавок к обычным):

```
options    BOOTP          # Use BOOTP to obtain IP address/hostname
```

```
options    BOOTP_NFSROOT # NFS mount root filesystem using BOOTP info
```

```
options    BOOTP_COMPAT  # Workaround for broken bootp daemons.
```

Вам может потребоваться использовать BOOTP_NFSV3 и BOOTP_WIRED_TO (посмотрите LINT).

Постройте ядро (обратитесь к [Chapter 9](#)) и скопируйте его в каталог tftp под именем,

указанным в `dhcpd.conf`.

17.5.1.5. Подготовка корневой файловой системы

Вам нужно создать корневую файловую систему для бездисковых рабочих станций, в месоположении, заданном как `root-path` в `dhcpd.conf`.

Самым простым способом сделать это является использование скрипта `/usr/share/examples/diskless/clone_root`. Этот скрипт требуется настроить, по крайней мере, подправив место, где будет создана файловая система (переменная `DEST`).

Прочтите комментарии в начале скрипта для получения указаний. Там описано, как строится основная файловая система, и как файлы могут быть выборочно заменены версиями, предназначенными для работы без диска, для подсети или для отдельной рабочей станции. Также здесь даются примеры бездисковых файлов `/etc/fstab` и `/etc/rc.conf`.

Файлы `README` в `/usr/share/examples/diskless` много интересной информации, но вместе с другими примерами из каталога `diskless` они на самом деле описывают метод настройки, который отличается от того, что используется в `clone_root` и `/etc/rc.diskless[12]`, этим несколько запутывая дело. Используйте их только для справки, за исключением того случая, когда вы выберете метод, ими описываемый, и тогда вам нужны исправленные скрипты `rc`.

17.5.1.6. Настройка области подкачки

Если это нужно, то файл подкачки, расположенный на сервере, можно использовать посредством NFS. Точные параметры `bootptab` или `dhcpd.conf` пока плохо документированы. Сообщается, что работает следующая конфигурация с использованием `isc-dhcp 3.0rc11`.

1. Добавьте следующие строки в dhcpd.conf:
2. # Global section
3. option swap-path code 128 = string;
4. option swap-size code 129 = integer 32;
- 5.
6. host margaux {
7. ... # Стандартные строки, смотрите выше
8. option swap-path "192.168.4.4:/netswapvolume/netswap";
9. option swap-size 64000;
10. }

Идея заключается в том, что, по крайней мере для клиента FreeBSD, дополнительный параметр DHCP/BOOTP под кодом 128 является маршрутом к файлу подкачки NFS, а параметр с кодом 129 задает размер области подкачки в килобайтах. Более старые версии dhcpd разрешали использовать синтаксис типа option option-128 "...", который больше не работает.

Во /etc/bootptab будет использоваться такой синтаксис:

```
T128="192.168.4.4:/netswapvolume/netswap":T129=64000
```

11. На файловом сервере NFS создайте файл (или файлы) подкачки

1. # **mkdir /netswapvolume/netswap**
2. # **cd /netswapvolume/netswap**
3. # **dd if=/dev/zero bs=1024 count=64000 of=swap.192.168.4.6**
4. # **chmod 0600 swap.192.168.4.6**

192.168.4.6 является IP-адресом бездискового клиента.

16. На файловом сервере NFS, в /etc/exports добавьте такую строку:

1. */netswapvolume -maproot=0:10 -alldirs margaux*

Затем укажите **mountd** на повторное чтение файла exports, как описано ранее.

17.5.1.7. Различные проблемы

17.5.1.7.1. Работа с /usr, доступной только для чтения

Если бездисковая рабочая станция настроена на запуск X, вам нужно подправить настроечный файл для xdm, который по умолчанию помещает протокол ошибок в /usr.

17.5.1.7.2. Использование не-FreeBSD сервера

Если сервер с корневой файловой системой работает не под управлением FreeBSD, вам потребуется создать корневую файловую систему на машине FreeBSD, а затем скопировать ее в нужное место, при помощи tar или cpio.

В такой ситуации иногда возникают проблемы со специальными файлами в /dev из-за различной разрядности целых чисел для старшего/младшего чисел. Решением этой проблемы является экспортирование каталога с не-FreeBSD сервера, монтирование его на машине с FreeBSD и запуск скрипта MAKEDEV на машине с FreeBSD для создания

правильных файлов устройств.

оригинал: <http://freebsd.org.ru/handbook/diskless.html#CO-DHCP-FILENAME>