Подготовка релизов FreeBSD в части программных пакетов сторонних разработчиков

Steve Price <steve@FreeBSD.org >

Издание: 43234

2013-11-24 taras.

Аннотация

В этом документе описывается подход, используемый группой менеджеров портов FreeBSD для создания высококачественного набора пакетов, предназначенного для размещения на носителях с официальными релизами FreeBSD. Этот документ находится в состоянии доработки, и постепенно он покроет весь процесс построения целостного набора пакетов в «Кластере Портов» проекта FreeBSD.org, опишет, как настроить любой другой набор машин в качестве кластера портов, как разделить пакеты для размещения на носителях с релизом и как проверить целостность набора пакетов.

Содержание

1. Построение пакетов из Коллекции Портов ... 2
2. Разделение пакетов ... 2

1. Построение пакетов из Коллекции Портов

Коллекция портов FreeBSD представляет собой собрание более чем 24,000 программных пакетов сторонних разработчиков, доступных для FreeBSD. За поддержку дерева портов, которое может использоваться для подготовки бинарных пакетов, идущих с некоторым релизом FreeBSD, в рабочем состоянии, отвечает Группа Менеджеров Дерева Портов FreeBSD <portmgr@FreeBSD.org >.

1.1. Кластер портов

Для подготовки целостного набора пакетов сторонних разработчиков для релизов FreeBSD каждый порт строиться в отдельном окружении с изменённым корнем файловой системы, с изначально пустыми каталогами `/usr/local` и `/usr/X11R6`. Требуемые зависимости устанавливаются в виде пакетов до момента начала построения. Это обеспечивает *целостность* процесса построения пакетов. Запуская построение пакета в чистом окружении, мы можем гарантировать, что метаданные пакета (такие, как требуемые зависимости) корректны. Таким образом, мы никогда не создаём пакеты, которые, в зависимости от ранее установленного программного обеспечения, на одних системах работать могут, а на других работать не будут.

«Кластер портов» для архитектуры x86 в настоящее время состоит из центрального узла (Dual Pentium® III 733MHz) и 8 подчинённых узлов (Pentium® III 800MHz), выполняющих реальное построение пакетов. В такой конфигурации полное построение пакетов занимает более 24 часов. Эти машины размещены вместе с остальным оборудованием Проекта FreeBSD в углу с оборудованием компании Yahoo у провайдера Exodus в калифорнийском городе Санта-Клара.

«Кластер портов» для архитектуры Alpha состоит из 7 машин PWS 500A, предоставленных компанией Compaq, и они также расположены вместе с оборудованием компании Yahoo.

2. Разделение пакетов

Для FreeBSD 4.4 было создано более 4.1 гигабайт пакетов. Это привело к проблемам для распространения на CDROM, так как нам нужно записать как можно больше пакетов без необходимости заставлять пользователя вставлять другой диск для поиска зависимостей. Решением является создание «кластеров» близких пакетов с похожими зависимостями и группировать их на отдельные диски. В этом разделе

~~описывается программное обеспечение и методология, использованная при созда~~нии таких наборов пакетов для официальных дисков FreeBSD с релизами.

Скрипты и остальные файлы, необходимые для выполнения разделения пакетов, можно найти в дереве CVS в каталоге `ports/Tools/scripts/release` . Скопируйте этот каталог на машину с свободным дисковым пространством, достаточным для хранения 2 или 3 копий набора пакетов, который вы хотите разделить.

В этом каталоге находятся следующие скрипты:

`config`
: В этом файле находится свободное пространство на каждом диске и то, разрешено ли на этих дисках размещать пакеты, дистрибутивные файлы или оба типа файлов одновременно. Первая колонка содержит название диска. Оно должно быть в виде `disc[0-9a-z]` . На данный момент настройка выполнена на 10 дисков (4 для релиза и 6 для инструментального набора). При этом подразумевается дополнительный диск под названием «scratch», куда помещаются остальные дистрибутивные файлы и пакеты, если они больше никуда не помещаются. Вторая колонка может быть `1` или `0`, где `1` означает возможность размещения на этом диске пакетов. Третья колонка работает так же, но контролирует размещение на этом диске пакетов. Последняя колонка отмечает количество байтов свободного пространства на диске.

`doit.sh`
: Это рабочая лошадка. Как только все файлы будут на своих местах и всё будет полностью настроено, этот скрипт управляет процессом разделения пакетов. Имейте в виду, что он интерактивный, так что вам нужно присматривать за его работой. Подробности о том, что делает скрипт, будут даны позже.

`checkdeps.pl`
: Проверяет наличие всех зависимостей по файлу `INDEX` и каталогу с пакетами.

`oneshot.pl`
: Здесь происходит основное волшебство (я использую это слово с некоторой натяжкой, так как большей частью всё решается в лоб). Имея перечень требуемых пакетов для каждого диска и набор пакетов/дистрибутивных файлов, этот скрипт размещает пакеты или дистрибутивные файлы по дискам вместе со всеми их зависимостями.

`print-cdrom-packages.sh`
: Этот файл является копией `src/release/scripts/print-cdrom-packages.sh` из релиза, над которым вы работаете.

`scrubindex.pl`
: Этот скрипт удаляет строки из файла `INDEX` для отсутствующих пакетов. Он также удаляет зависимости XFree86™. ЗАМЕЧАНИЕ: вам нужно изменить значение переменной `xdep`, чтобы обеспечить правильность номера версии.

`setup.sh`
: Это вспомогательный скрипт, используемый мною в кластере сборки пакетов для получения копии дерева портов и соответствующего набора пакетов/дистрибутивных файлов.

Вот контрольный список того, что вам нужно будет проверить или настроить перед тем, как продолжить.

1. Отредактируйте `config`, отметив количество имеющихся у вас дисков, их размеры и собираетесь ли вы размещать на них пакеты, дистрибутивные файлы, оба типа файлов или ничего из них.

2. Проверьте, что вы удалили каталог `gen`, который мог остаться от предыдущих работ. Этот каталог содержит рабочие файлы, которые будут актуальны только для текущих действий.

3. При первом вашем прогоне лучше всего избежать копирования пакетов и дистрибутивных файлов. Это сохранит как время, так и дисковое пространство, пока вы сможете проделать пару пробных запусков, чтобы удостовериться в том, что всё помещается, и так далее. В файле `oneshot.pl` задайте переменную `fake` в значение 1 и вместо реального копирования файлов он будет только выполнять над ними команду touch(1). Обязательно выключите эту возможность или установите переменную `fake` в значение 0 перед тем, как передавать получающиеся диски тому, кто будет тиражировать диски, в противном случае каталоги будут заполнены файлами с нулевыми размерами.

4. Убедитесь, что у вас есть последняя копия файла `print-cdrom-packages.sh`, и он взят из соответствующего релиза.

5. Проверьте, что зависимость XFree86™ в файле `scrubindex.pl` имеет корректный номер версии. Вам также нужно будет проверять, что это значение правильно задано в файле `doit.sh`.

Теперь вам нужно сделать копию дерева портов, пакетов и дистрибутивных файлов из последнего построения кластера пакетов. Посмотрите файл `setup.sh` в качестве рабочего примера, но здесь описано всё необходимое.

1. Возьмите копию `ports.tar.gz` и распакуйте её в каталог `ports` отдельно от файла `doit.sh` и каталога `scripts`.

2. Удалите каталог packages/distfiles или символические ссылки на него. В bento имеются эти символические ссылки и вы будете получать смешанные результаты, если не избавитесь от них перед обработкой.

3. Создайте новый каталог ports/packages и скопируйте набор пакетов из кластера построения пакетов.

4. ~~Создайте новый каталог ports/distfiles и скопируйте дистрибутивные файлы из~~ кластера построения пакетов. ЗАМЕЧАНИЕ: если вы не хотите копировать дистрибутивные файлы, просто создайте каталог и оставьте его пустым. Этот каталог должен существовать, даже если в нём ничего нет.

Теперь мы окончательно готовы к весёлому занятию по реальному разделению пакетов. Запускайте обработку командой `./doit.sh`. Вот что она делает при первом её запуске.

1. Создаёт список ограниченных к распространению (которые не могут находиться на главном FTP-сервере) портов.

2. Запросит у вас, нужно ли удаление ограниченных к распространению портов. В большинстве случаев здесь вам лучше ответить (y)es.

3. Создаёт перечень пакетов/дистрибутивных файлов, которые не могут быть размещены на дисках.

4. Запросит у вас разрешение на удаление пакетов/дистрибутивных файлов, которые не размещаются на компакт-диск. В большинстве случаев здесь лучше ответить (y)es.

5. Копирует файл `INDEX` из каталога `ports` в каталог `gen`. При этом она удаляет строчки портов, для которых нет пакетов. Также проверяется наличие всех требуемых зависимых пакетов.

6. Создаёт список пакетов, требуемых для размещения на каждом диске.

7. Делает запрос на создание дисков. После генерации каждого диска она проверяет отсутствующие зависимости, вычищает файл `INDEX` и создаёт файл `CHECKSUM.MD5`.

8. Выполняет проверку того, что на каждом диске размещены требуемые пакеты и выдаёт общую статистику по размерам каждого диска.

Если вам повезёт с первого раза, то все требуемые пакеты будут построены и размещены по дискам. Всё, что вам нужно сделать, это задать в качестве значения переменной `fake` в `0` в файле `oneshot.pl` и перезапустить `./doit.sh`. При втором и последующем запусках шаги 1-5 выше будут пропускаться. Если вы хотите выполнить любые из тех шагов повторно, обратитесь к `doit.sh`, чтобы выяснить, какие файлы необходимо удалить для того, чтобы не пропустить эти этапы. Если вы хотите повторить все эти шаги, то проще всего выполнить `rm -rf gen`.

К моменту успешного завершения пакеты и дистрибутивные файлы будут располагаться в каталогах `disc*`, а оставшиеся файлы будут находиться в каталоге `scratch`.

Что делать, если что-то происходит не так? Вот некоторые распространённые проблем и их решения.

Отсутствие требуемых пакетов

Такое часто случается. Вам нужно подождать появления нового набора пакетов, где будут построены отсутствующие пакеты, либо попросить кого-нибудь перезапустить построение пакетов для вас. *Не пытайтесь* строить отсутствующие пакеты на собственной машине и добавлять их. Хотя вам может удастся обойтись этим, если вы очень осторожны, все же в большинстве случаев вы, скорее всего, пропустите какую-то маленькую деталь и простой процесс добавления одного пакета может привести к неожиданной неработоспособности сотен остальных.

Требуемые пакеты не помещаются

Такое тоже случается, и это относительно легко исправить. Просто отредактируйте файл `print-cdrom-packages.sh` , перенося пакеты, пока они не станут помещаться. Да, это итерационный процесс, и это одна из причин, по которой вы должны включать `fake` в `oneshot.pl` , пока не получите то, что хотите. Перезапустите `./doit.sh` после того, как внесли свои изменения.

Требуемые пакеты располагаются не на том (или ни на каком) диске

Обычно это означает, что вы не добавили их в файл `print-cdrom-packages.sh` или разместили их не на том диске. Этот скрипт является святым источником, по которому в целом определяется, где должен находиться пакет. Если вы хотите обеспечить расположение пакета на определённом диске, то это единственное место, гарантирующее требуемое вам расположение.

Если вы окончательно запутались, и не можете понять, почему всё работает неправильно и как это исправить, то пошлите письмо в адрес Steve Price <steve@FreeBSD.org > с просьбой о помощи.