Csup - обновляем систему

Опубликовано muff в Сб, 2009-08-15 19:44

FreeBSD прекратило распространение портов через CVS и CVSup

Начиная с 28 февраля 2013 года порты FreeBSD перестанут быть доступны через CVS, т.е. их нельзя будет зеркалировать или синхронизировать для поддержания в актуальном виде через такие инструменты как CVSup. Причиной прекращения поддержки CVS является переход к использованию Subversion для разработки и поддержания дерева портов, в то время как используемые средства экспорта SVN в CVS имеют ряд ограничений, приводящих к затруднениям при организации работы зеркал.

Пользователям предлагается спланировать переход на portsnap или svn для обновления дерева портов.

Пора заняться апдейтом сервера. Сначала, пока не было никаких изменений, советую обновить саму FreeBSD до более новой ветки. Для этого воспользуемся утилитой csup, которая устанавливается непосредственно с системой.

Проверяем текущую версию:

```
# uname -a

FreeBSD web0.domain.ua 7.0-RELEASE FreeBSD 7.0-RELEASE #0: Sun Feb 24 10:35:36 UTC 2008
root [at] driscoll [dot] cse [dot] buffalo [dot] edu:/usr/obj/usr/src/sys/GENERIC amd64
```

Все верно. 7.0-RELEASE, как мы и устанавливали... Чуда не произошло ;)
Что ж, это нужно исправить. Первым делом рекомендую создать папку, куда мы положим файлы конфигурации.

```
# mkdir /etc/csup
```

Создаем файл конфигурации следующего содержания:

```
# cat /etc/csup/src-all

*default host=cvsup3.ua.FreeBSD.org
*default base=/var/db
*default prefix=/usr
*default release=cvs tag=RELENG_7
*default delete use-rel-suffix
*default compress src-all
```

Коротко о опциях:
- default host – хост, с которого мы и будем скачивать обновления для нашей системы (поскольку я живу в Украине, соответственно выбрал украинский ресурс – "cvsup3.ua.FreeBSD.org");
- default base – место, где система будет хранить инфу о файлах, которые будут загружены на наш сервер. Настройки "/var/db" сгенерируют эту информацию в "/var/db/sup". Можно изменить данный путь, запустив csup с опцией "-b base". Указанная директория должна существовать!
- default prefix – место, где будут храниться запрошенные файлы. Указав "/usr", файлы будут размещены в "/usr/src", "/usr/src/bin", "/usr/src/lib" соответственно. Каталог должен существовать!
- default release – собственно, версия, до которой мы будем обновлять систему. "RELENG_7_0" – вытянет файлы для последней версии 7.0, " RELENG_7_0_0_RELEASE" – последние исходные коды 7.0-RELEASE. Суффикс "RELENG_7" позволит загрузить файлы для последней версии в 7 ветке.

- default compress – включаем компрессию и вытягиваем все исходные коды ядра, а не какую-то конкретную часть.
Можно сразу создать файлы для получения обновлений портов, документации и исходных кодов системы:

Supfile для обновления портов:

```
# cat /etc/csup/ports-all
*default host=cvsup3.ua.FreeBSD.org
*default base=/var/db
*default prefix=/usr
*default release=cvs tag=.
*default delete use-rel-suffix
*default compress ports-all
```

Supfile для обновления документации:

```
# cat /etc/csup/doc-all
*default host=cvsup2.ua.FreeBSD.org
*default base=/var/db
*default prefix=/usr
*default release=cvs tag=.
*default delete use-rel-suffix
*default compress doc-all
```

Supfile для обновления исходных кодов ядра:

```
# cat /etc/csup/src-sys

 *default host=cvsup3.ua.FreeBSD.org
*default base=/var/db
*default prefix=/usr
*default release=cvs tag=RELENG_7
*default delete use-rel-suffix
*default compress src-sys
```

Supfile для обновления исходных кодов всей системы:

```
# cat /etc/csup/src-all

*default host=cvsup3.ua.FreeBSD.org
*default base=/var/db
*default prefix=/usr
*default release=cvs tag=RELENG_7
*default delete use-rel-suffix
*default compress src-all
```

В принципе, обновлять регулярно нужно только коллекцию портов. Обновлять исходные коды ядра можно перед его очередной пересборкой, если появилась такая необходимость. Ну, а исходные коды всей системы – перед переходом на новую версию.

Для обновления коллекции портов регулярно, в "*/etc/crontab*" нужно добавить строчку:

```
25 2 * * * root /usr/bin/csup -g -L 2 /etc/csup/ports-all > /dev/null 2>&1
```

Итак, вернемся к нашим баранам... Запускаем обновление исходных кодов ядра:

```
# csup -g -L 2 /etc/csup/src-all
```

Если на экране видим следующий вывод

```
Connecting to cvsup3.ua.FreeBSD.org
Connected to 212.42.64.9
Server software version: SNAP_16_1h
Negotiating file attribute support
Exchanging collection information
Establishing multiplexed-mode data connection
Running
```

значит все нормально. Ждем окончания обновления. Тем, у кого каналы в Интернет низкоскоростные придется ждать порядочно...

Дождались? Поздравляю самых терпеливых... Исходные коды подтянуты ;)

```
Shutting down connection to server
Finished successfully
```

Рассмотрим, что же нам нужно сделать дальше... Эту информацию можно почерпнуть из файла "/usr/src/Makefile".
Рекомендую ознакомиться полностью со всем файлом. Но на данный момент нас интересует обновление системы. Соответственно делаем выборку того, что нам нужно сделать:

```
# 1. `cd /usr/src'       (or to the directory containing your source tree).
# 2. `make buildworld'
# 3. `make buildkernel KERNCONF=YOUR_KERNEL_HERE'     (default is GENERIC).
# 4. `make installkernel KERNCONF=YOUR_KERNEL_HERE'   (default is GENERIC).
#       [steps 3. & 4. can be combined by using the "kernel" target]
# 5. `reboot'        (in single user mode: boot -s from the loader prompt).
# 6. `mergemaster -p'
# 7. `make installworld'
# 8. `make delete-old'
# 9. `mergemaster'                        (you may wish to use -U or -ai).
# 10. `reboot'
# 11.  `make delete-old-libs' (in case no 3rd party program uses them anymore)
```

Итак приступим...

```
# cd /usr/src/
# make -j4 buildworld
```

Построение мира занимает много времени, поэтому запаситесь пивом и терпением. Обратите также внимание на то, что запуск был сделан с ключем -j4 (сборка в несколько потоков), что должно ускорить процесс сборки.

Итак... Свершилось чудо:

```
------------------------------------------------------------
>>> World build completed on Sat Aug 8 03:48:39 EEST 2009
------------------------------------------------------------
```

Теперь можно приступать к сборке и установки ядра. Пропускать эти шаги НЕЛЬЗЯ, даже мотивируя тем, что "я когда-то уже собирал ядро..."
Чтобы запустить сборку со стандартным ядром GENERIC, достаточно набрать "make kernel" в директории "/usr/src". Ну а поскольку мы легких путей не ищем, то будем собирать свой вариант ядра.
Для начала скопируем стандартный конфиг ядра в свой:

```
# cp /usr/src/sys/amd64/conf/GENERIC /usr/src/sys/amd64/conf/Web0
```

Обратите внимание на "amd64" – это платформа в моем частном случае. В вашем случае может отличаться. Чтобы определить какая у вас платформа, воспользуйтесь командой "uname –a"
Приступим к редактированию созданного нами файла:

```
# ee /usr/src/sys/amd64/conf/Web0
```

Не буду расписывать каждую опцию, а приведу только свой конфиг (документации по сборке ядра более чем достаточно, возможно даже будет частично доступна на этом ресурсе)

```
# $FreeBSD: src/sys/amd64/conf/GENERIC,v 1.484.2.21 2009/07/15 08:32:19 ed Exp $

cpu             HAMMER
ident           Web0

options         SCHED_ULE               # ULE scheduler
options         PREEMPTION              # Enable kernel thread preemption
options         INET                    # InterNETworking
options         FFS                     # Berkeley Fast Filesystem
options         SOFTUPDATES             # Enable FFS soft updates support
options         UFS_ACL                 # Support for access control lists
options         UFS_DIRHASH             # Improve performance on big directories
options         UFS_GJOURNAL            # Enable gjournal-based UFS journaling
options         GEOM_PART_GPT           # GUID Partition Tables.
options         GEOM_LABEL              # Provides labelization
options         COMPAT_43TTY            # BSD 4.3 TTY compat [KEEP THIS!]
options         COMPAT_IA32             # Compatible with i386 binaries
options         COMPAT_FREEBSD4         # Compatible with FreeBSD4
options         COMPAT_FREEBSD5         # Compatible with FreeBSD5
options         COMPAT_FREEBSD6         # Compatible with FreeBSD6
options         KTRACE                  # ktrace(1) support
options         STACK                   # stack(9) support
options         SYSVSHM                 # SYSV-style shared memory
options         SYSVMSG                 # SYSV-style message queues
options         SYSVSEM                 # SYSV-style semaphores
options         P1003_1B_SEMAPHORES     # POSIX-style semaphores
options         _KPOSIX_PRIORITY_SCHEDULING # POSIX P1003_1B real-time extensions
options         KBD_INSTALL_CDEV        # install a CDEV entry in /dev
options         ADAPTIVE_GIANT          # Giant mutex is adaptive.
options         STOP_NMI                # Stop CPUS using NMI instead of IPI
options         AUDIT                   # Security event auditing
#options        KDTRACE_FRAME           # Ensure frames are compiled in
#options        KDTRACE_HOOKS           # Kernel DTrace hooks

# Make an SMP-capable kernel by default
options         SMP                     # Symmetric MultiProcessor Kernel

# CPU frequency control
device          cpufreq

# Bus support.
device          acpi
device          pci

# ATA and ATAPI devices
device          ata
```

```
device          atadisk         # ATA disk drives
options         ATA_STATIC_ID   # Static device numbering

# atkbdc0 controls both the keyboard and the PS/2 mouse
device          atkbdc          # AT keyboard controller
device          atkbd           # AT keyboard
device          kbdmux          # keyboard multiplexer

device          vga             # VGA video card driver

device          splash          # Splash screen and screen saver support

# syscons is the default console driver, resembling an SCO console
device          sc

# PCI Ethernet NICs that use the common MII bus controller code.
# NOTE: Be sure to keep the 'device miibus' line in order to use these NICs!
device          miibus          # MII bus support
device          nfe             # nVidia nForce MCP on-board Ethernet
device          sk              # SysKonnect SK-984x & SK-982x gigabit Ethernet

# Pseudo devices.
device          loop            # Network loopback
device          random          # Entropy device
device          ether           # Ethernet support
device          pty             # Pseudo-ttys (telnet etc)
device          firmware        # firmware assist module
device          vlan            # 802.1Q VLAN

# The `bpf' device enables the Berkeley Packet Filter.
# Be aware of the administrative consequences of enabling this!
# Note that 'bpf' is required for DHCP.
device          bpf             # Berkeley packet filter

# My options
options         IPFIREWALL
options         IPFIREWALL_VERBOSE
options         IPFIREWALL_VERBOSE_LIMIT=100
options         IPFIREWALL_DEFAULT_TO_ACCEPT
options         IPFIREWALL_FORWARD
options         DUMMYNET
options         QUOTA
options         ACCEPT_FILTER_HTTP
```

Исходя из "/usr/src/Makefile", шаги 3 и 4 можно совместить в одну команду, что мы и делаем:

```
# make kernel KERNCONF=Web0
```

Если процесс сборки и установки ядра прошел без ошибок, пускаем сервер в ребут:

```
# shutdown -r now
```

Если никаких ошибок не допустили, то через некоторое время, необходимое серверу для перезагрузки, можно опять приконектиться к серверу и продолжить начатое.
Если же ядро почему-то не загрузилось, то вам сюда:
http://www.freebsd.org/doc/ru/books/handbook/kernelconfig-trouble.html [1]

Теперь начинаются различия обновления локального и удаленного серверов. Рассмотрим два этих варианта:

Локальный сервер

Для безопасного и корректного обновления необходимо ввести машину в single mode:

Перезагрузкой и выбором в стартовом меню загрузку в single user mode (номер 4).

```
# shutdown -r now
```

Удаленный сервер

Вам необходимо отключить в /etc/rc.conf все службы, в которых нет необходимости для запуска системы (sshd не выключать!). Это СУРБД, FTP, Apache и т.д. и перезагрузится. Можно и самим остановить эти службы без перезагрузки системы.

Продолжаем обновление
При запуске системы, если это локальный сервер, необходимо смонтировать корневой каталог и остальные разделы. А так же запустим mergemaster для создания начальной точки работы:

```
# /sbin/mount -u /
# /sbin/mount -a
# su
# mergemaster –p
```

Установим мир.

```
# cd /usr/src
# make installworld
```

Запустим mergemaster для синхронизации файлов, библиотек, конфигурационных файлов с новыми версиями.

```
# mergemaster -cv
```

(c-контекстный diffs вместо унифицированного)
mergemaster сравнивает существующие конфигурационные файлы с теми, которые предлагаются новой версией FreeBSD. И если существуют отличия, то появляется сообщение об этом. Первой строкой в нем идет имя файла, который не соответствует новым требованиям, а ниже сами отличия.
Знаки:
- - этим знаком помечаются строки, которые mergemaster собирается удалить.
+ - этим знаком помечаются строки, которые будут добавлены.
Есть несколько вариантов управления и в низу mergemaster предлагают следующие варианты работы с текущим файлом:
d - удалить предлагаемый вариант оставить старый.
i - установить предлагаемый вариант, удалив старый.
m - сравнить и объединить при необходимости построчно старый и новый вариант.
v - посмотреть отличия в файлах снова.
При построчном сравнивании экран делится на две части, слева строки из старого файла, а справа из предлагаемого. При нажатии на "Enter" выводятся возможные варианты последующих действий:
l - использовать вариант с левой стороны.
r - использовать вариант с правой стороны.
и т.д. После того как сравнение файла закончилось, предлагаются следующие варианты:
i - установить получившийся файл.
r - повторить сравнение снова.
v - просмотреть получившийся файл.
Бывает, что с новой версией системы появляются новые файлы, которых до этого не было. В таком случае предлагается 2 варианта:
d - удалить предлагаемый вариант и оставить старый.
i - установить предлагаемый вариант, удалив старый.
И так сравниваются все файлы.
В конце mergemaster предложить удалить все, что осталось в */var/tmp/temproot* – соглашаемся («yes»), а также, напоследок, обновить базу aliases – соглашаемся («yes»).

Закончив с mergemaster, перезапускаем систему и удаляем старые библиотеки.

```
# shutdown -r now
# cd /usr/src/
# make delete-old-libs
```

Проверяем версию системы:

```
# uname -a

FreeBSD web0.domain.ua 7.2-STABLE FreeBSD 7.2-STABLE #0: Sat Aug 8 04:58:58 EEST 2009
muff [at] web0 [dot] domain [dot] ua:/usr/obj/usr/src/sys/Web0 amd64
```

Теперь осталось еще закачать новые версии портов...

```
# csup -g -L 2 /etc/csup/ports-all
Shutting down connection to server
Finished successfully
```

Порты установлены. Можно приступать к установке софта.

Источник (получено *2026-11-02 08:15*): http://muff.kiev.ua/content/csup-obnovlyaem-sistemu

Ссылки:
[1] http://www.freebsd.org/doc/ru/books/handbook/kernelconfig-trouble.html