Construye tu propio servidor de actualizaciones de FreeBSD

Resumen

Este artículo describe cómo construir un Servidor de FreeBSD Update interno. El freebsd-update-server está escrito por Colin Percival <cperciva@FreeBSD.org>, Security Officer Emérito de FreeBSD. Para los usuarios que consideran adecuado actualizar sus sistemas contra un servidor de actualizaciones oficial, construir su propio Servidor de FreeBSD Update podría ayudar a extender su funcionalidad mediante el soporte de versiones de FreeBSD ajustadas a mano o proporcionando un mirror local que permitirá actualizaciones más rápidas para un determinado número de máquinas.

Tabla de contenidos

1. Reconocimientos . . . 1
2. Introducción . . . 1
3. Prerequisitos . . . 2
4. Configuración: Instalación & Preparación . . . 2
5. Compilando el código de actualización . . . 4
6. Compilar un parche . . . 8
7. Consejos . . . 12

1. Reconocimientos

Este artículo fue imprimido posteriormente en BSD Magazine.

2. Introducción

Los usuarios experimentados o administradores a menudo son responsables de múltiples máquinas o entornos. Entienden las complicadas demandas y los desafíos de mantener una infraestructura de este tipo. Ejecutar un Servidor de Actualizaciones de FreeBSD facilita la implementación de parches de seguridad y software en las máquinas de pruebas seleccionadas antes de implementarlos en producción. Esto también significa que se pueden actualizar varios sistemas desde la red local en lugar de una conexión a Internet la cual sea potencialmente más lenta. Este artículo describe los pasos necesarios para crear un servidor de actualizaciones de FreeBSD propio.

3. Prerequisitos

Para construir un Servidor de Actualizaciones de FreeBSD se tienen que cumplir algunos requisitos.

- Un sistema FreeBSD en ejecución.



Como mínimo, las actualizaciones requieren que cree una versión de FreeBSD que sea mayor o igual a la versión de la versión de destino para la distribución.

- Una cuenta de usuario con al menos 4 GB de espacio disponible. Esto permitirá la creación de actualizaciones para las versiones 7.1 y 7.2, los requisitos de espacio exactos pueden cambiar de una versión a otra.
- Una cuneta ssh(1) en una máquina remota para subir los parches distribuidos.
- Un servidor web, como Apache, con más de la mitad del espacio requerido para la construcción. Por ejemplo, construcciones de prueba para 7.1 y 7.2 consumen un total de 4 GB, y el espacio necesario en el servidor web para distribuir estas actualizaciones es de 2.6 GB.
- Conocimiento básico de shell scripting con el shell Bourne, sh(1).

4. Configuración: Instalación & Preparación

Descarga el software freebsd-update-server instalando los paquetes devel/git y security/ca_root_nss y ejecuta:

```
% git clone https://github.com/freebsd/freebsd-update-build.git freebsd-update-server
```

Actualiza el fichero scripts/build.conf apropiadamente. Se crea durante todas las operaciones de construcción (build).

Aquí está el archivo build.conf predeterminado, que debe ser modificado para adaptarse a su entorno.

```
# Main configuration file for FreeBSD Update builds.  The
# release-specific configuration data is lower down in
# the scripts tree.

# Location from which to fetch releases
export FTP=ftp://ftp2.freebsd.org/pub/FreeBSD/releases ①

# Host platform
export HOSTPLATFORM=`uname -m`

# Host name to use inside jails
export BUILDHOSTNAME=${HOSTPLATFORM}-builder.daemonology.net ②

# Location of SSH key
```

```
export SSHKEY=/root/.ssh/id_dsa ③

# SSH account into which files are uploaded
MASTERACCT=builder@wadham.daemonology.net ④

# Directory into which files are uploaded
MASTERDIR=update-master.freebsd.org ⑤
```

Los parámetros a considerar son:

① Este es el lugar desde donde se descargan las imágenes ISO (por la subrutina `fetchiso()`de scripts/build.subr). La localización configurada no se limita a URIs FTP. Cualquier esquema URI soportado por la utilidad estándar fetch(1) debería funcionar bien. Se pueden instalar adaptaciones al código de `fetchiso()`copiando el script por defecto build.subr a los áreas de release y específicos de la arquitectura en scripts/RELEASE/ARCHITECTURE/build.subr y aplicando cambios locales.

② El nombre del host de compilación. Esta información se mostrará en los sistemas que estén actualizados cuando se ejecute el comando:

```
% uname -v
```

③ La clave SSH utilizada para subir los ficheros al servidor de actualizaciones. Se tiene que crear un par de claves tecleando `ssh-keygen -t dsa`. Este parámetro es opcional; se utilizará autenticación estándar por contraseña como mecanismo de reserva cuando `SSHKEY` no está definida. La página de manual de ssh-keygen(1) tiene información más detallada acerca de SSH y los pasos apropiados para crear y utilizar una.

④ Cuenta para subir archivos al servidor de actualizaciones.

⑤ Directorio en el servidor de actualización donde se subirán los archivos.

El archivo build.conf predeterminado incluido con el código fuente de freebsd-update-server es adecuado para compilar versiones i386 de FreeBSD. Como ejemplo de creación de un servidor de actualizaciones para otras arquitecturas, los siguientes pasos describen los cambios necesarios en la configuración para amd64:

1. Cree un entorno de compilación para amd64:

   ```
   % mkdir -p /usr/local/freebsd-update-server/scripts/7.2-RELEASE/amd64
   ```

2. Coloque el archivo build.conf en el directorio de compilación que acaba de crear. Las opciones de configuración de la compilación para FreeBSD 7.2-RELEASE en la arquitectura amd64 deberían de ser similares a:

   ```
   # SHA256 hash of RELEASE disc1.iso image.
   export RELH=1ea1f6f652d7c5f5eab7ef9f8edbed50cb664b08ed761850f95f48e86cc71ef5 ①
   # Components of the world, source, and kernels
   ```

```
export WORLDPARTS="base catpages dict doc games info manpages proflibs lib32"
export SOURCEPARTS="base bin contrib crypto etc games gnu include krb5  \
                lib libexec release rescue sbin secure share sys tools  \
                ubin usbin cddl"
export KERNELPARTS="generic"

# EOL date
export EOL=1275289200 ②
```

① La clave hash sha256(1) para la versión deseada se publica con el respectivo anuncio de lanzamiento.

② Para generar el número de "End of Life" para build.conf, utiliza el "Estimated EOL" publicado en el FreeBSD Security Website. El valor de EOL se puede derivar de la fecha establecida en el sitio web, usando la utilidad date(1), por ejemplo:

```
% date -j -f '%Y%m%d-%H%M%S' '20090401-000000' '+%s'
```

5. Compilando el código de actualización

El primer paso es ejecutar scripts/make.sh. Esto creará algunos binarios, creará directorios y generará una clave RSA utilizada para aprobar las compilaciones. En este paso, se deberá proporcionar una contraseña para la creación de la clave de firma.

```
# sh scripts/make.sh
cc -O2 -fno-strict-aliasing -pipe   findstamps.c  -o findstamps
findstamps.c: In function 'usage':
findstamps.c:45: warning: incompatible implicit declaration of built-in function
'exit'
cc -O2 -fno-strict-aliasing -pipe   unstamp.c  -o unstamp
install findstamps ../bin
install unstamp ../bin
rm -f findstamps unstamp
Generating RSA private key, 4096 bit long modulus
.....................................................................++
...................++
e is 65537 (0x10001)

Public key fingerprint:
27ef53e48dc869eea6c3136091cc6ab8589f967559824779e855d58a2294de9e

Encrypting signing key for root
enter aes-256-cbc encryption password:
Verifying - enter aes-256-cbc encryption password:
```



Guarda una copia de seguridad de la huella digital generada. Este valor es necesario en el archivo /etc/freebsd-update.conf para las actualizaciones binarias.

En este punto, estamos listos para empezar el build.

```
# cd /usr/local/freebsd-update-server
# sh scripts/init.sh amd64 7.2-RELEASE
```

Lo que se muestra a continuación es un ejemplo de una ejecución de construcción *inicial*.

```
# sh scripts/init.sh amd64 7.2-RELEASE
Mon Aug 24 16:04:36 PDT 2009 Starting fetch for FreeBSD/amd64 7.2-RELEASE
/usr/local/freebsd-update-server/work/7.2-RELE100 of  588 MB  359 kBps 00m00s
Mon Aug 24 16:32:38 PDT 2009 Verifying disc1 hash for FreeBSD/amd64 7.2-RELEASE
Mon Aug 24 16:32:44 PDT 2009 Extracting components for FreeBSD/amd64 7.2-RELEASE
Mon Aug 24 16:34:05 PDT 2009 Constructing world+src image for FreeBSD/amd64 7.2-
RELEASE
Mon Aug 24 16:35:57 PDT 2009 Extracting world+src for FreeBSD/amd64 7.2-RELEASE
Mon Aug 24 23:36:24 UTC 2009 Building world for FreeBSD/amd64 7.2-RELEASE
Tue Aug 25 00:31:29 UTC 2009 Distributing world for FreeBSD/amd64 7.2-RELEASE
Tue Aug 25 00:32:36 UTC 2009 Building and distributing kernels for FreeBSD/amd64 7.2-
RELEASE
Tue Aug 25 00:44:44 UTC 2009 Constructing world components for FreeBSD/amd64 7.2-
RELEASE
Tue Aug 25 00:44:56 UTC 2009 Distributing source for FreeBSD/amd64 7.2-RELEASE
Mon Aug 24 17:46:18 PDT 2009 Moving components into staging area for FreeBSD/amd64
7.2-RELEASE
Mon Aug 24 17:46:33 PDT 2009 Identifying extra documentation for FreeBSD/amd64 7.2-
RELEASE
Mon Aug 24 17:47:13 PDT 2009 Extracting extra docs for FreeBSD/amd64 7.2-RELEASE
Mon Aug 24 17:47:18 PDT 2009 Indexing release for FreeBSD/amd64 7.2-RELEASE
Mon Aug 24 17:50:44 PDT 2009 Indexing world0 for FreeBSD/amd64 7.2-RELEASE

Files built but not released:
Files released but not built:
Files which differ by more than contents:
Files which differ between release and build:
kernel|generic|/GENERIC/hptrr.ko
kernel|generic|/GENERIC/kernel
src|sys|/sys/conf/newvers.sh
world|base|/boot/loader
world|base|/boot/pxeboot
world|base|/etc/mail/freebsd.cf
world|base|/etc/mail/freebsd.submit.cf
world|base|/etc/mail/sendmail.cf
world|base|/etc/mail/submit.cf
world|base|/lib/libcrypto.so.5
world|base|/usr/bin/ntpq
world|base|/usr/lib/libalias.a
world|base|/usr/lib/libalias_cuseeme.a
world|base|/usr/lib/libalias_dummy.a
world|base|/usr/lib/libalias_ftp.a
```

```
...
```

Entonces, se vuelve a realizar la construcción de todo (build world), con los parches de world. Puedes encontrar una explicación más detallada en scripts/build.subr.



Durante este segundo ciclo de construcción, el demonio del protocolo de tiempo de red, ntpd(8), se desactiva. Según Colin Percival <cperciva@FreeBSD.org>, Security Officer Emérito de FreeBSD, "el código de construcción de freebsd-update-server necesita identificar timestamps que se almacenan en ficheros de forma que se puedan ignorar cuando se comparan distintas construcciones para determinar qué ficheros se han actualizado. Este trabajo de encontrar los timestamps funciona haciendo dos construcciones separadas 400 días y comparando los resultados."

```
Mon Aug 24 17:54:07 PDT 2009 Extracting world+src for FreeBSD/amd64 7.2-RELEASE
Wed Sep 29 00:54:34 UTC 2010 Building world for FreeBSD/amd64 7.2-RELEASE
Wed Sep 29 01:49:42 UTC 2010 Distributing world for FreeBSD/amd64 7.2-RELEASE
Wed Sep 29 01:50:50 UTC 2010 Building and distributing kernels for FreeBSD/amd64 7.2-
RELEASE
Wed Sep 29 02:02:56 UTC 2010 Constructing world components for FreeBSD/amd64 7.2-
RELEASE
Wed Sep 29 02:03:08 UTC 2010 Distributing source for FreeBSD/amd64 7.2-RELEASE
Tue Sep 28 19:04:31 PDT 2010 Moving components into staging area for FreeBSD/amd64
7.2-RELEASE
Mon Aug 24 19:04:46 PDT 2009 Extracting extra docs for FreeBSD/amd64 7.2-RELEASE
Mon Aug 24 19:04:51 PDT 2009 Indexing world1 for FreeBSD/amd64 7.2-RELEASE
Mon Aug 24 19:08:04 PDT 2009 Locating build stamps for FreeBSD/amd64 7.2-RELEASE
Mon Aug 24 19:10:19 PDT 2009 Cleaning staging area for FreeBSD/amd64 7.2-RELEASE
Mon Aug 24 19:10:19 PDT 2009 Preparing to copy files into staging area for
FreeBSD/amd64 7.2-RELEASE
Mon Aug 24 19:10:20 PDT 2009 Copying data files into staging area for FreeBSD/amd64
7.2-RELEASE
Mon Aug 24 12:16:57 PDT 2009 Copying metadata files into staging area for
FreeBSD/amd64 7.2-RELEASE
Mon Aug 24 12:16:59 PDT 2009 Constructing metadata index and tag for FreeBSD/amd64
7.2-RELEASE

Files found which include build stamps:
kernel|generic|/GENERIC/hptrr.ko
kernel|generic|/GENERIC/kernel
world|base|/boot/loader
world|base|/boot/pxeboot
world|base|/etc/mail/freebsd.cf
world|base|/etc/mail/freebsd.submit.cf
world|base|/etc/mail/sendmail.cf
world|base|/etc/mail/submit.cf
world|base|/lib/libcrypto.so.5
world|base|/usr/bin/ntpq
world|base|/usr/include/osreldate.h
world|base|/usr/lib/libalias.a
```

```
world|base|/usr/lib/libalias_cuseeme.a
world|base|/usr/lib/libalias_dummy.a
world|base|/usr/lib/libalias_ftp.a
...
```

Finalmente, el build termina.

```
Values of build stamps, excluding library archive headers:
v1.2 (Aug 25 2009 00:40:36)
v1.2 (Aug 25 2009 00:38:22)
@()FreeBSD 7.2-RELEASE 0: Tue Aug 25 00:38:29 UTC 2009
FreeBSD 7.2-RELEASE 0: Tue Aug 25 00:38:29 UTC 2009
    root@server.myhost.com:/usr/obj/usr/src/sys/GENERIC
7.2-RELEASE
Mon Aug 24 23:55:25 UTC 2009
Mon Aug 24 23:55:25 UTC 2009
 built by root@server.myhost.com on Tue Aug 25 00:16:15 UTC 2009
 built by root@server.myhost.com on Tue Aug 25 00:16:15 UTC 2009
 built by root@server.myhost.com on Tue Aug 25 00:16:15 UTC 2009
 built by root@server.myhost.com on Tue Aug 25 00:16:15 UTC 2009
Mon Aug 24 23:46:47 UTC 2009
ntpq 4.2.4p5-a Mon Aug 24 23:55:53 UTC 2009 (1)
 * Copyright (c) 1992-2009 The FreeBSD Project.
Mon Aug 24 23:46:47 UTC 2009
Mon Aug 24 23:55:40 UTC 2009
Aug 25 2009
ntpd 4.2.4p5-a Mon Aug 24 23:55:52 UTC 2009 (1)
ntpdate 4.2.4p5-a Mon Aug 24 23:55:53 UTC 2009 (1)
ntpdc 4.2.4p5-a Mon Aug 24 23:55:53 UTC 2009 (1)
Tue Aug 25 00:21:21 UTC 2009
Tue Aug 25 00:21:21 UTC 2009
Tue Aug 25 00:21:21 UTC 2009
Mon Aug 24 23:46:47 UTC 2009

FreeBSD/amd64 7.2-RELEASE initialization build complete.  Please
review the list of build stamps printed above to confirm that
they look sensible, then run
 sh -e approve.sh amd64 7.2-RELEASE
to sign the release.
```

Aprobar la construcción si todo es correcto. Puedes encontrar más información sobre cómo determinar si es correcto en el archivo USAGE. Ejecuta scripts/approve.sh, como se indica. Esto firmará la versión y moverá los componentes a un área de preparación adecuada para la subida.

```
# cd /usr/local/freebsd-update-server
# sh scripts/mountkey.sh
```

```
# sh -e scripts/approve.sh amd64 7.2-RELEASE
Wed Aug 26 12:50:06 PDT 2009 Signing build for FreeBSD/amd64 7.2-RELEASE
Wed Aug 26 12:50:06 PDT 2009 Copying files to patch source directories for
FreeBSD/amd64 7.2-RELEASE
Wed Aug 26 12:50:06 PDT 2009 Copying files to upload staging area for FreeBSD/amd64
7.2-RELEASE
Wed Aug 26 12:50:07 PDT 2009 Updating databases for FreeBSD/amd64 7.2-RELEASE
Wed Aug 26 12:50:07 PDT 2009 Cleaning staging area for FreeBSD/amd64 7.2-RELEASE
```

Una vez finalizado el proceso de aprobación, se puede iniciar el proceso de subida.

```
# cd /usr/local/freebsd-update-server
# sh scripts/upload.sh amd64 7.2-RELEASE
```



En el caso de que código de actualización necesite ser subido de nuevo, se puede hacer cambiando al directorio público de las distribuciones para la versión que se quiera y actualizando los atributos del fichero *uploaded*.

```
# cd /usr/local/freebsd-update-server/pub/7.2-RELEASE/amd64
# touch -t 200801010101.01 uploaded
```

Se necesita que los ficheros subidos se encuentren en el documento raíz del servidor web para que las actualizaciones sean distribuidas. La configuración exacta variará dependiendo del servidor web utilizado. Para el servidor web Apache, por favor lee la sección Configuration of Apache servers en el Handbook.

Actualiza `KeyPrint` y `ServerName` del cliente en /etc/freebsd-update.conf y actualiza según se indica en la sección FreeBSD Update del Handbook.



Para que el Servidor de Actualizaciones de FreeBSD funcione correctamente, se necesita construir tanto la versión *actual* como la versión *a la que se quiere actualizar*. Esto es necesario para determinar las diferencias entre dos versiones. Por ejemplo, cuando se actualiza un sistema FreeBSD de 7.1-RELEASE a 7.2-RELEASE, las actualizaciones necesitan construirse y subirse a tu servidor de distribución para ambas versiones.

Como referencia, se adjunta la ejecución completa de init.sh.

6. Compilar un parche

Cada vez que se anuncia un security advisory o un security notice se puede construir un parche de actualización.

Para este ejemplo, se utilizará 7.1-RELEASE.

Para una versión de lanzamiento diferente, se da por hecho un par de suposiciones:

- Que ha configurado la estructura de directorios correcta para la compilación inicial.
- Que ha realizado una compilación inicial para la versión 7.1-RELEASE.

Cree el directorio de parches para la versión correspondiente en /usr/local/freebsd-update-server/patches/.

```
% mkdir -p /usr/local/freebsd-update-server/patches/7.1-RELEASE/
% cd /usr/local/freebsd-update-server/patches/7.1-RELEASE
```

Toma como ejemplo el parche para named(8). Lee la advertencia de seguridad y obtén el fichero necesario de FreeBSD Security Advisories. Se puede encontrar más información acerca de cómo interpretar las advertencias en FreeBSD Handbook.

En el security brief, esta advertencia se llama `SA-09:12.bind`. Después de descargar el fichero, se necesita renombrar el fichero a un nivel de parcheado apropiado. Se sugiere mantenerlo consistente con los niveles de parche oficiales de FreeBSD, pero su nombre se puede escoger libremente. Para esta construcción, sigamos la práctica actualmente establecida en FreeBSD y llamemos a este `p7`. Renombra el fichero:

```
% cd /usr/local/freebsd-update-server/patches/7.1-RELEASE/; mv bind.patch 7-SA-
09:12.bind
```



Cuando se ejecuta la compilación del parche, se asume que los parches anteriores están en su sitio. Cuando se ejecute la compilación de los parches, se ejecutarán todos los parches que se encuentren en el directorio de parches.

Puede haber parches personalizados adjuntos a cualquier compilación. Usa el número cero, o cualquier otro número.



Depende del administrador del Servidor de Actualizaciones de FreeBSD tomar las medidas apropiadas para verificar la autenticidad de cada parche.

En este punto, ya se puede construir un *diff*. El software comprueba primero si se ha ejecutado un scripts/init.sh en la versión respectiva antes de ejecutar la construcción del diff.

```
# cd /usr/local/freebsd-update-server
# sh scripts/diff.sh amd64 7.1-RELEASE 7
```

Lo siguiente es una muestra de una ejecución de construcción *differencial*.

```
# sh -e scripts/diff.sh amd64 7.1-RELEASE 7
Wed Aug 26 10:09:59 PDT 2009 Extracting world+src for FreeBSD/amd64 7.1-RELEASE-p7
Wed Aug 26 17:10:25 UTC 2009 Building world for FreeBSD/amd64 7.1-RELEASE-p7
```

```
Wed Aug 26 18:05:11 UTC 2009 Distributing world for FreeBSD/amd64 7.1-RELEASE-p7
Wed Aug 26 18:06:16 UTC 2009 Building and distributing kernels for FreeBSD/amd64 7.1-
RELEASE-p7
Wed Aug 26 18:17:50 UTC 2009 Constructing world components for FreeBSD/amd64 7.1-
RELEASE-p7
Wed Aug 26 18:18:02 UTC 2009 Distributing source for FreeBSD/amd64 7.1-RELEASE-p7
Wed Aug 26 11:19:23 PDT 2009 Moving components into staging area for FreeBSD/amd64
7.1-RELEASE-p7
Wed Aug 26 11:19:37 PDT 2009 Extracting extra docs for FreeBSD/amd64 7.1-RELEASE-p7
Wed Aug 26 11:19:42 PDT 2009 Indexing world0 for FreeBSD/amd64 7.1-RELEASE-p7
Wed Aug 26 11:23:02 PDT 2009 Extracting world+src for FreeBSD/amd64 7.1-RELEASE-p7
Thu Sep 30 18:23:29 UTC 2010 Building world for FreeBSD/amd64 7.1-RELEASE-p7
Thu Sep 30 19:18:15 UTC 2010 Distributing world for FreeBSD/amd64 7.1-RELEASE-p7
Thu Sep 30 19:19:18 UTC 2010 Building and distributing kernels for FreeBSD/amd64 7.1-
RELEASE-p7
Thu Sep 30 19:30:52 UTC 2010 Constructing world components for FreeBSD/amd64 7.1-
RELEASE-p7
Thu Sep 30 19:31:03 UTC 2010 Distributing source for FreeBSD/amd64 7.1-RELEASE-p7
Thu Sep 30 12:32:25 PDT 2010 Moving components into staging area for FreeBSD/amd64
7.1-RELEASE-p7
Wed Aug 26 12:32:39 PDT 2009 Extracting extra docs for FreeBSD/amd64 7.1-RELEASE-p7
Wed Aug 26 12:32:43 PDT 2009 Indexing world1 for FreeBSD/amd64 7.1-RELEASE-p7
Wed Aug 26 12:35:54 PDT 2009 Locating build stamps for FreeBSD/amd64 7.1-RELEASE-p7
Wed Aug 26 12:36:58 PDT 2009 Reverting changes due to build stamps for FreeBSD/amd64
7.1-RELEASE-p7
Wed Aug 26 12:37:14 PDT 2009 Cleaning staging area for FreeBSD/amd64 7.1-RELEASE-p7
Wed Aug 26 12:37:14 PDT 2009 Preparing to copy files into staging area for
FreeBSD/amd64 7.1-RELEASE-p7
Wed Aug 26 12:37:15 PDT 2009 Copying data files into staging area for FreeBSD/amd64
7.1-RELEASE-p7
Wed Aug 26 12:43:23 PDT 2009 Copying metadata files into staging area for
FreeBSD/amd64 7.1-RELEASE-p7
Wed Aug 26 12:43:25 PDT 2009 Constructing metadata index and tag for FreeBSD/amd64
7.1-RELEASE-p7
...
Files found which include build stamps:
kernel|generic|/GENERIC/hptrr.ko
kernel|generic|/GENERIC/kernel
world|base|/boot/loader
world|base|/boot/pxeboot
world|base|/etc/mail/freebsd.cf
world|base|/etc/mail/freebsd.submit.cf
world|base|/etc/mail/sendmail.cf
world|base|/etc/mail/submit.cf
world|base|/lib/libcrypto.so.5
world|base|/usr/bin/ntpq
world|base|/usr/include/osreldate.h
world|base|/usr/lib/libalias.a
world|base|/usr/lib/libalias_cuseeme.a
world|base|/usr/lib/libalias_dummy.a
world|base|/usr/lib/libalias_ftp.a
```

```
...
Values of build stamps, excluding library archive headers:
v1.2 (Aug 26 2009 18:13:46)
v1.2 (Aug 26 2009 18:11:44)
@()FreeBSD 7.1-RELEASE-p7 0: Wed Aug 26 18:11:50 UTC 2009
FreeBSD 7.1-RELEASE-p7 0: Wed Aug 26 18:11:50 UTC 2009
    root@server.myhost.com:/usr/obj/usr/src/sys/GENERIC
7.1-RELEASE-p7
Wed Aug 26 17:29:15 UTC 2009
Wed Aug 26 17:29:15 UTC 2009
 built by root@server.myhost.com on Wed Aug 26 17:49:58 UTC 2009
 built by root@server.myhost.com on Wed Aug 26 17:49:58 UTC 2009
 built by root@server.myhost.com on Wed Aug 26 17:49:58 UTC 2009
 built by root@server.myhost.com on Wed Aug 26 17:49:58 UTC 2009
Wed Aug 26 17:20:39 UTC 2009
ntpq 4.2.4p5-a Wed Aug 26 17:29:42 UTC 2009 (1)
 * Copyright (c) 1992-2009 The FreeBSD Project.
Wed Aug 26 17:20:39 UTC 2009
Wed Aug 26 17:29:30 UTC 2009
Aug 26 2009
ntpd 4.2.4p5-a Wed Aug 26 17:29:41 UTC 2009 (1)
ntpdate 4.2.4p5-a Wed Aug 26 17:29:42 UTC 2009 (1)
ntpdc 4.2.4p5-a Wed Aug 26 17:29:42 UTC 2009 (1)
Wed Aug 26 17:55:02 UTC 2009
Wed Aug 26 17:55:02 UTC 2009
Wed Aug 26 17:55:02 UTC 2009
Wed Aug 26 17:20:39 UTC 2009
...
```

Se muestran las actualizaciones y se te solicita aprobación.

```
New updates:
kernel|generic|/GENERIC/kernel.symbols|f|0|0|0555|0|7c8dc176763f96ced0a57fc04e7c1b8d79
3f27e006dd13e0b499e1474ac47e10|
kernel|generic|/GENERIC/kernel|f|0|0|0555|0|33197e8cf15bbbac263d17f39c153c9d489348c2c5
34f7ca1120a1183dec67b1|
kernel|generic|/|d|0|0|0755|0||
src|base|/|d|0|0|0755|0||
src|bin|/|d|0|0|0755|0||
src|cddl|/|d|0|0|0755|0||
src|contrib|/contrib/bind9/bin/named/update.c|f|0|10000|0644|0|4d434abf0983df9bc474356
70d307fa882ef4b348ed8ca90928d250f42ea0757|
src|contrib|/contrib/bind9/lib/dns/openssldsa_link.c|f|0|10000|0644|0|c6805c39f3da2a06
dd3f163f26c314a4692d4cd9a2d929c0acc88d736324f550|
src|contrib|/contrib/bind9/lib/dns/opensslrsa_link.c|f|0|10000|0644|0|fa0f7417ee9da42c
c8d0fd96ad24e7a34125e05b5ae075bd6e3238f1c022a712|
...
FreeBSD/amd64 7.1-RELEASE update build complete.  Please review
the list of build stamps printed above and the list of updated
files to confirm that they look sensible, then run
```

```
# sh -e approve.sh amd64 7.1-RELEASE
to sign the build.
```

Sigue el mismo proceso descrito anteriormente para dar tu aprobación a una compilación:

```
# sh -e scripts/approve.sh amd64 7.1-RELEASE
Wed Aug 26 12:50:06 PDT 2009 Signing build for FreeBSD/amd64 7.1-RELEASE
Wed Aug 26 12:50:06 PDT 2009 Copying files to patch source directories for
FreeBSD/amd64 7.1-RELEASE
Wed Aug 26 12:50:06 PDT 2009 Copying files to upload staging area for FreeBSD/amd64
7.1-RELEASE
Wed Aug 26 12:50:07 PDT 2009 Updating databases for FreeBSD/amd64 7.1-RELEASE
Wed Aug 26 12:50:07 PDT 2009 Cleaning staging area for FreeBSD/amd64 7.1-RELEASE

The FreeBSD/amd64 7.1-RELEASE update build has been signed and is
ready to be uploaded.  Remember to run
 sh -e umountkey.sh
to unmount the decrypted key once you have finished signing all
the new builds.
```

Después de dar tu aprobación a la compilación, sube el software al servidor de actualizaciones:

```
# cd /usr/local/freebsd-update-server
# sh scripts/upload.sh amd64 7.1-RELEASE
```

Se adjunta como referencia el resultado de la ejecución completa de diff.sh.

7. Consejos

- Si se construye una versión personalizada utilizando el comando nativo `make release` procedure, el código de freebsd-update-server funcionará desde tu versión. Como ejemplo, se puede construir una versión sin ports o documentación eliminando funcionalidad que hace referencia a las rutinas de documentación `findextradocs ()`, `addextradocs ()` y alterando el lugar de descarga en `fetchiso ()`, respectivamente, en in scripts/build.subr. Como último paso, cambia el hash sha256(1) en build.conf en tu versión y arquitectura específica y ya estás listo para construir tu versión personalizada.

  ```
  # Compare ${WORKDIR}/release y ${WORKDIR}/$1, identifique qué partes
  # del subcomponente world|doc faltan en este último, y
  # construya el tarball con ellas.
  findextradocs () {
  }
  # Añadir documentos extra a ${WORKDIR}/$1
  addextradocs () {
  }
  ```

- Añadir flags `-j NUMERO` a los objetivos `buildworld` y `obj` en el script scripts/build.subr puede acelerar el proceso dependiendo del hardware utilizado, sin embargo no es necesario. Utilizar estos flags en otros objetivos no está recomendado, ya que puede provocar que la construcción no sea correcta.

```
            # Build the world
          log "Building world"
          cd /usr/src &&
          make -j 2 ${COMPATFLAGS} buildworld 2>&1
        # Distribute the world
          log "Distributing world"
          cd /usr/src/release &&
          make -j 2 obj &&
          make ${COMPATFLAGS} release.1 release.2 2>&1
```

- Crea un registro DNS SRV para el servidor de actualizaciones y pon otros detrás de él con pesos variables. Utilizar este mecanismo proporcionará mirrors de actualización, sin embargo, este truco no es necesario a menos que desees proporcionar servicio redundante.

```
_http._tcp.update.myserver.com.     IN SRV   0 2 80   host1.myserver.com.
                    IN SRV   0 1 80   host2.myserver.com.
                    IN SRV   0 0 80   host3.myserver.com.
```