Einen eigenen FreeBSD Update Server bauen

Zusammenfassung

Dieser Artikel beschreibt den Bau eines internen FreeBSD Update Server. Die freebsd-update-server Software wurde von Colin Percival <cperciva@FreeBSD.org>, emeritierter Security Officer von FreeBSD, geschrieben. Benutzer, die es als vorteilhaft ansehen ihre Systeme über einen offiziellen Update-Server zu aktualisieren, können mit Hilfe eines selbst erstellten FreeBSD Update Server die Funktionalität über manuell optimierte FreeBSD Releases oder über Bereitstellung eines lokalen Mirror, welcher schnellere Updates ermöglicht, erweitern.

Inhaltsverzeichnis

1. Danksagung . . . 1
2. Einführung . . . 1
3. Voraussetzungen . . . 2
4. Konfiguration: Installation & Setup . . . 2
5. Den Update Code bauen . . . 4
6. Eine Fehlerkorrektur erstellen . . . 8
7. Tipps . . . 12

Übersetzt von Björn Heidotting <bhd@FreeBSD.org>.

1. Danksagung

Dieser Artikel wurde anschließend im BSD Magazine gedruckt.

2. Einführung

Erfahrene Benutzer oder Administratoren sind häufig für etliche Maschinen oder Umgebungen verantwortlich. Sie verstehen die schwierigen Anforderungen und Herausforderungen der Aufrechterhaltung einer solchen Infrastruktur. Ein FreeBSD Update Server macht es einfacher, Sicherheits- und Software-Korrekturen für ausgewählte Test-Maschinen bereitzustellen, bevor diese dann auf den Produktionssystemen ausgerollt werden. Es bedeutet auch, dass eine Reihe von Systemen über das lokale Netzwerk, anstatt über eine langsame Internet-Verbindung, aktualisiert werden können. Dieser Artikel beschreibt die Vorgehensweise zum Erstellen eines eigenen FreeBSD Update Server.

3. Voraussetzungen

Für den Bau eines internen FreeBSD Update Server sollten einige Anforderungen erfüllt werden.

- Ein laufendes FreeBSD System.



Als Minimum, muss das zu verteilende Ziel-Release auf einer gleichen, oder höheren FreeBSD Version gebaut werden.

- Ein Benutzerkonto mit mindestens 4 GB freiem Speicherplatz. Dies erlaubt die Erstellung der Updates für 7.1 und 7.2. Der genaue Platzbedarf kann sich aber von Version zu Version ändern.
- Ein ssh(1) Konto auf einem entfernten System, um die später zu verteilenden Updates hochzuladen.
- Einen Webserver, wie Apache, wobei über die Hälfte des Platzes für den Bau benötigt wird. Als Beispiel benötigt der Bau von 7.1 und 7.2 insgesamt 4 GB. Der Speicherplatz, den der Webserver für die Verteilung dieser Updates benötigt, würde 2.6 GB betragen.
- Grundlegende Kenntnisse im Shell Skripting mit der Bourne Shell, sh(1).

4. Konfiguration: Installation & Setup

Laden Sie die freebsd-update-server Software durch die Installation von devel/subversion sowie security/ca_root_nss, und starten Sie:

```
% svn co https://svn.freebsd.org/base/user/cperciva/freebsd-update-build freebsd-
update-server
```

Passen Sie scripts/build.conf an Ihre Bedürfnisse an. Diese Datei wird bei jedem Bau mit einbezogen.

Hier ist die Standardeinstellung für build.conf, welche Sie für Ihre Umgebung anpassen sollten.

```
# 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
```

Parameter, die zu berücksichtigen sind:

① Dies ist der Ort, von dem die ISO Abbilder (über die `fetchiso()` in scripts/build.subr) heruntergeladen werden. Der Ort ist nicht auf FTP URIs beschränkt. Jedes URI-Schema, welches von fetch(1) unterstützt wird, sollte hier gut funktionieren. Anpassungen am `fetchiso()` Code können Sie vornehmen, indem Sie das Standardskript build.subr in den Release- und Architektur-spezifischen Bereich in scripts/RELEASE/ARCHITECTURE/build.subr kopieren und dort lokale Änderungen vornehmen.

② Der Name des Build-Hosts. Auf aktualisierten Systemen können Sie diese Information wie folgt ausgeben:

```
% uname -v
```

③ Der SSH Schlüssel für das Hochladen der Dateien auf den Update Server. Ein Schlüsselpaar kann durch die Eingabe von `ssh-keygen -t dsa` erstellt werden. Dieser Parameter ist jedoch optional; Standard Password Authentifizierung wird als Fallback-Methode benutzt wenn `SSHKEY` nicht definiert ist. Die ssh-keygen(1) Manualpage enthält detaillierte Informationen zu SSH und die entsprechenden Schritte zur Erstellung und Verwendung von Schlüsseln.

④ Benutzerkonto zum Hochladen von Dateien auf den Update-Server.

⑤ Verzeichnis auf dem Update-Server, in welches die Dateien hochgeladen werden.

Die Standard build.conf, die mit den `freebsd-update-server` Quellen ausgeliefert wird ist geeignet um i386 Releases von FreeBSD zu bauen. Als Beispiel für den Aufbau eines Update-Servers für andere Architekturen beschreiben die folgenden Schritte die Konfiguration für amd64:

1. Erstellen Sie eine Bau-Umgebung für amd64:

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

2. Installieren Sie eine build.conf in das neu erstellte Verzeichnis. Die Konfigurationsoptionen für FreeBSD 7.2-RELEASE auf amd64 sollten ähnlich wie die folgenden sein:

   ```
   # 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 Der sha256(1) Fingerabdruck für die gewünschte Version wird innerhalb der jeweiligen Release-Ankündigung veröffentlicht.

② Um die "End of Life" Nummer für die build.conf zu generieren, beziehen Sie sich bitte auf "Estimated EOL" auf der FreeBSD Security Webseite. Der Wert für EOL kann aus dem Datum, das auf der Webseite veröffentlicht ist, abgeleitet werden. Benutzen Sie dafür das Werkzeug date(1). Dazu ein Beispiel:

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

5. Den Update Code bauen

Der erste Schritt ist das Ausführen von scripts/make.sh. Dieses Skript baut einige Binärdateien, erstellt Verzeichnisse und einen RSA Signaturschlüssel für die Genehmigung des Bau. In diesem Schritt müssen Sie auch eine Passphrase für die Erstellung des Signaturschlüssels angeben.

```
# 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:
```



Notieren Sie sich den Fingerabdruck des erzeugten Schlüssels. Dieser Wert wird in /etc/freebsd-update.conf für die binären Updates benötigt.

An dieser Stelle sind wir bereit, den Bauprozess zu starten.

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

Hier folgt ein Beispiel für einen *ersten* Bauprozess.

```
# 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
```

```
...
```

Anschließend wird das Basissystem mit den dazugehörigen Patches erneut gebaut. Eine detaillierte Erklärung dazu finden Sie in scripts/build.subr.



Während der zweiten Bauphase wird der Network Time Protocol Dienst, ntpd(8), ausgeschaltet. Per Colin Percival <cperciva@FreeBSD.org>, emeritierter Security Officer von FreeBSD, "Der freebsd-update-server Code muss Zeitstempel, welche in Dateien gespeichert sind, identifizieren, sodass festgestellt werden kann, welche Dateien aktualisiert werden müssen. Dies geschieht, indem zwei Builds erstellt werden die 400 Tage auseinander liegen und anschließend die Ergebnisse verglichen werden."

```
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
...
```

Schlussendlich wird der Bauprozess fertiggestellt.

```
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.
```

Genehmigen Sie den Bau, wenn alles korrekt ist. Weitere Informationen zur korrekten Bestimmung finden Sie in der Quelldatei namens USAGE. Führen Sie, wie angegeben scripts/approve.sh aus. Dieser Schritt unterschreibt das Release und verschiebt die Komponenten an einen Sammelpunkt, wo sie für den Upload verwendet werden können.

```
# 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
```

Nachdem der Genehmigungsprozess abgeschlossen ist, kann der Upload gestartet werden.

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



Wenn der Update-Code erneut hochgeladen werden muss, kann dies durch die Änderung des öffentlichen Distributionsverzeichnisses für das Ziel-Release und der Aktualisierung der Attribute für die *hochgeladene* Datei geschehen.

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

Um die Updates zu verteilen, müssen die hochgeladenen Dateien im Document Root des Webservers liegen. Die genaue Konfiguration hängt von dem verwendeten Webserver ab. Für den Apache Webserver, beziehen Sie sich bitte auf das Kapitel Konfiguration des Apache Servers im Handbuch.

Aktualisieren Sie `KeyPrint` und `ServerName` in der /etc/freebsd-update.conf des Clients und führen Sie das Update, wie im Kapitel FreeBSD Update des Handbuchs beschrieben, aus.



Damit FreeBSD Update Server ordnungsgemäß funktioniert, muss sowohl das *current* Release als auch das Release *auf welches Sie aktualisieren wollen* neu gebaut werden. Dies ist notwendig, um die Unterschiede von Dateien zwischen den beiden Releases bestimmen zu können. Zum Beispiel beim Upgrade eines FreeBSD Systems von 7.1-RELEASE auf 7.2-RELEASE, müssen für beide Versionen Updates gebaut und auf den Webserver hochgeladen werden.

Als Referenz wird der gesamte Verlauf von init.sh beigefügt.

6. Eine Fehlerkorrektur erstellen

Jedes Mal, wenn ein Sicherheits-Hinweis oder ein Fehler-Hinweis angekündigt wird, kann eine Fehlerkorrektur gebaut werden.

Für dieses Beispiel wird 7.1-RELEASE benutzt.

Für den Bau eines anderen Release werden ein paar Annahmen getroffen:

- Richten Sie die korrekte Verzeichnisstruktur für den ersten Bau ein.
- Führen Sie einen ersten Bau für 7.1-RELEASE aus.

Erstellen Sie das Korrekturverzeichnis des jeweiligen Releases unter /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
```

Als Beispiel nehmen Sie die Korrektur für named(8). Lesen Sie den Hinweis und laden Sie die erforderliche Datei von FreeBSD Sicherheits-Hinweise herunter. Weitere Informationen zur Interpretation der Sicherheitshinweise finden Sie im FreeBSD Handbuch.

In der Sicherheits Anweisung, nennt sich dieser Hinweis SA-09:12.bind. Nach dem Herunterladen der Datei, ist es erforderlich, die Datei auf einen geeigneten Patch-Level umzubenennen. Es steht Ihnen frei den Namen frei zu wählen, es wird jedoch nahegelegt, diesen im Einklang mit dem offiziellen FreeBSD Patch-Level zu halten. Für diesen Bau folgen wir der derzeit gängigen Praxis von FreeBSD und benennen sie p7. Benennen Sie die Datei um:

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



Wenn ein Patch-Level gebaut wird, wird davon ausgegangen, dass die bisherigen Korrekturen bereits vorhanden sind. Wenn der Bau läuft, werden alle Korrekturen aus dem Patchverzeichnis mit gebaut.

Es können auch selbsterstellte Korrekturen zum Bau hinzugefügt werden. Benutzen Sie die Zahl Null, oder jede andere Zahl.



Es liegt in der Verantwortung des Administrators des FreeBSD Update Server geeignete Maßnahmen zu treffen, um die Authentizität jeder Fehlerkorrektur zu überprüfen.

An dieser Stelle sind wir bereit, einen *Diff* zu bauen. Die Software prüft zunächst, ob scripts/init.sh für das jeweilige Release gelaufen ist, bevor mit dem Bau des Diff begonnen wird.

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

Es folgt ein Beispiel für einen *Diff* Bauprozess.

```
# 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
...
```

Die Updates werden angezeigt und warten auf Genehmigung.

```
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.
```

Folgen Sie dem zuvor erwähnten Verfahren für die Genehmigung des Bauprozesses:

```
# 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.
```

Nachdem Sie den Bau genehmigt haben, starten Sie den Upload der Software:

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

Als Referenz wird der gesamte Verlauf von diff.sh beigefügt.

7. Tipps

- Wenn Sie ein selbst erstelltes Release über die native `make release` Prozedur bauen, wir der `freebsd-update-server` Code Ihr Release unterstützen. Als Beispiel können Sie ein Release ohne Ports oder Dokumentation bauen, indem Sie betreffende Funktionalität der Subroutinen `findextradocs ()`, `addextradocs ()` entfernen und eine Veränderung des Download-Verzeichnisses in `fetchiso ()`, in scripts/build.subr. Als letzten Schritt ändern Sie den sha256(1) Hash in build.conf für Ihr jeweiliges Release und Architektur damit Sie bereit sind, Ihr benutzerdefiniertes Release zu bauen.

  ```
  # Compare ${WORKDIR}/release and ${WORKDIR}/$1, identify which parts
  # of the world|doc subcomponent are missing from the latter, and
  # build a tarball out of them.
  findextradocs () {
  }
  # Add extra docs to ${WORKDIR}/$1
  addextradocs () {
  ```

```
}
```

- Durch das Hinzufügen von `-j NUMMER` zu den `buildworld` und `obj` Zielen in scripts/build.subr kann die Verarbeitung, abhängig von der eingesetzten Hardware, beschleunigt werden. Die Benutzung dieser Optionen auf andere Ziele wird jedoch nicht empfohlen, da sie den Bau unzuverlässig machen können.

```
>               # 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
```

- Erstellen Sie einen geeigneten DNS SRV Datensatz für den Update-Server, und fügen Sie weitere Server mit verschiedenen Gewichtungen hinzu. Sie können diese Möglichkeit nutzen um Update-Mirror hinzuzufügen. Dieser Tipp ist jedoch nicht notwendig solange Sie keinen redundanten Service anbieten möchten.

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