

Construa seu próprio servidor de atualização do FreeBSD

Resumo

Este artigo descreve a construção de um servidor de atualizações do FreeBSD interno. O [freebsd-update-server](#) foi escrito por Colin Percival cperciva@FreeBSD.org, Oficial de Segurança Emérito do FreeBSD. Para usuários que acham conveniente atualizar seus sistemas em um servidor de atualização oficial, construir seu próprio FreeBSD Update Server pode ajudar a estender sua funcionalidade suportando versões do FreeBSD ajustadas manualmente ou fornecendo um espelho local que permitirá atualizações mais rápidas para várias máquinas.

Índice

1. Agradecimentos	1
2. Introdução	1
3. Pré-requisitos	1
4. Configuração: Instalação & Configuração	2
5. Compilando o Código de Atualização	4
6. Compilando um Patch	9
7. Dicas	14

1. Agradecimentos

Este artigo foi publicado posteriormente no [BSD Magazine](#).

2. Introdução

Usuários experientes ou administradores são muitas vezes responsáveis por várias máquinas ou ambientes. Eles entendem as difíceis demandas e desafios da manutenção de tal infraestrutura. A execução de um Servidor de Atualização do FreeBSD facilita a implantação de patches de segurança e software em máquinas de teste selecionadas antes de implementá-las nas máquinas em produção. Isso também significa que vários sistemas podem ser atualizados a partir da rede local, em vez de uma conexão de Internet potencialmente mais lenta. Este artigo descreve os passos envolvidos na criação de um Servidor de Atualização do FreeBSD interno.

3. Pré-requisitos

Para construir um Servidor de Atualização do FreeBSD interno, alguns requisitos devem ser atendidos.

- Um sistema FreeBSD em execução.



No mínimo, as atualizações requerem a criação de uma versão do FreeBSD maior ou igual a versão do release alvo para a distribuição.

- Uma conta de usuário com pelo menos 4 GB de espaço disponível. Isso permitirá a criação de atualizações para 7.1 e 7.2, mas os requisitos de espaço exatos podem mudar de versão para versão.
- Uma conta com acesso ao [ssh\(1\)](#) em uma máquina remota para carregar atualizações distribuídas.
- Um servidor web, como o [Apache](#), com mais da metade do espaço necessário para a construção. Por exemplo, as compilações de teste para 7.1 e 7.2 consomem uma quantidade total de 4 GB e o espaço do servidor da web necessário para distribuir essas atualizações é de 2.6 GB.
- Conhecimento básico de shell script com o Bourne shell, [sh\(1\)](#).

4. Configuração: Instalação & Configuração

Faça o download do software [freebsd-update-server](#) instalando [devel/subversion](#) e [security/ca_root_nss](#), e execute:

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

Atualize o `scripts/build.conf` apropriadamente. Ele é criado durante todas as operações de construção.

Aqui está o `build.conf` padrão, que deve ser modificado para se adequar ao seu ambiente.

```

# 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

```

Parâmetros para consideração seriam:

- ① Este é o local onde as imagens ISO são baixadas (pela sub-rotina `fetchiso()` do `scripts/build.subr`). A localização configurada não está limitada a URIs de FTP. Qualquer esquema de URI suportado pelo utilitário padrão `fetch()` deve funcionar bem. Personalizações para o código de `fetchiso()` podem ser instaladas copiando o script padrão `build.subr` para a área específica do release e da arquitetura em `scripts/RELEASE/ARCHITECTURE/build.subr` e aplicando alterações locais.
- ② O nome do host em construção. Esta informação será exibida em sistemas atualizados ao executar:

```
% uname -v
```
- ③ A chave SSH para fazer upload de arquivos para o servidor de atualizações. Um par de chaves pode ser criado digitando `ssh-keygen -t dsa`. Este parâmetro é opcional; a autenticação de senha padrão será usada como um método de autenticação secundário quando a `SSHKEY` não estiver definida. A página de manual do `ssh-keygen(1)` contém informações mais detalhadas sobre o SSH e as etapas apropriadas para criar e usar um.
- ④ Conta para fazer upload de arquivos para o servidor de atualização.
- ⑤ Diretório no servidor de atualização para o qual os arquivos são enviados.

O arquivo padrão `build.conf` fornecido com o código-fonte do `freebsd-update-server` é adequado para a criação de versões i386 do FreeBSD. Como um exemplo de criação de um servidor de atualização para outras arquiteturas, as etapas a seguir descrevem as alterações necessárias na configuração para o amd64:

1. Crie um ambiente de compilação para o amd64:

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

2. Instale um build.conf no diretório de criação recém-criado. As opções de configuração de compilação para o FreeBSD 7.2-RELEASE com arquitetura amd64 devem ser semelhantes 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 A chave [sha256\(1\)](#) usada para fazer o hash para a release desejada é publicada no respectivo [anúncio de release](#).
- ② Para gerar o número "End of Life" para o build.conf, consulte o "EOL estimado" publicado no [Site de Segurança do FreeBSD](#). O valor de `EOL` pode ser derivado da data listada no site, usando o utilitário [date\(1\)](#), por exemplo:

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

5. Compilando o Código de Atualização

O primeiro passo é executar o `scripts/make.sh`. Isso criará alguns binários, criará diretórios e irá gerar uma chave de assinatura RSA usada para aprovar as compilações. Nesta etapa, uma senha terá que ser fornecida para a criação final da chave de assinatura.

```
# 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:
```



Mantenha um backup do fingerprint gerado. Este valor é necessário para o arquivo `/etc/freebsd-update.conf` para as atualizações binárias.

Neste ponto, estamos prontos para montar uma construção.

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

O que se segue é uma amostra de uma execução da compilação *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
...
```

Então a compilação do world é executada novamente, com patches para world. Uma explicação mais detalhada pode ser encontrada em `scripts/build.subr`.



Durante este segundo ciclo de compilação, o daemon do protocolo de tempo de rede, [ntpd\(8\)](#), é desativado. Segundo o Colin Percival cperciva@FreeBSD.org, Oficial de segurança emérito do FreeBSD, "a compilação do código do [freebsd-update-server](#) precisa identificar os timestamps que são armazenados nos arquivos para que possam ser ignorados ao comparar builds para determinar quais arquivos precisam ser atualizados. Essa busca de timestamp trabalha com duas construções com 400 dias de diferença e compara os 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/hpttr.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, a construção é concluída.

```
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)
ntpddate 4.2.4p5-a Mon Aug 24 23:55:53 UTC 2009 (1)
ntpdcc 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.
```

Aprove a compilação se tudo estiver correto. Mais informações sobre como determinar isso podem ser encontradas no arquivo fonte distribuído chamado USAGE. Execute scripts/approve.sh, conforme indicado. Isso assinará a release e moverá os componentes para uma área de preparação adequada para o upload.

```
# 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
```

Após o processo de aprovação ser concluído, o procedimento de upload pode ser iniciado.

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



No caso de o código de atualização precisar ser reenviado, isso pode ser feito mudando para o diretório de distribuições públicas para o release alvo e atualizando os atributos do arquivo *carregado*.

```
# cd /usr/local/freebsd-update-server/pub/7.2-RELEASE/amd64
# touch -t 2008010101.01 uploaded
```

Os arquivos enviados precisarão estar no diretório de documentos raiz do servidor web para que as atualizações sejam distribuídas. A configuração exata irá variar dependendo do servidor web usado. Para o servidor web Apache, consulte a sessão [Configuração de servidores Apache](#) no Handbook.

Atualize o `KeyPrint` e `ServerName` do cliente no arquivo `/etc/freebsd-update.conf`, e execute as atualizações conforme instruído na [Seção de Atualização do FreeBSD](#) no Handbook.



Para que o Servidor de Atualização do FreeBSD funcione corretamente, atualizações para ambas releases *atual* e a release *que se deseja atualizar* precisam ser compilados. Isso é necessário para determinar as diferenças de arquivos entre as releases. Por exemplo, ao atualizar um sistema FreeBSD de 7.1-RELEASE para 7.2-RELEASE, as atualizações precisarão ser construídas e carregadas em seu servidor de distribuição para ambas as versões.

Para referência, toda a execução do `init.sh` é anexada.

6. Compilando um Patch

Toda vez que é anunciado um [aviso de segurança](#) ou uma [notificação de segurança](#), um patch de atualização pode ser construído.

Para este exemplo, o 7.1-RELEASE será usado.

Algumas suposições são feitas para uma versão diferente:

- Configure a estrutura de diretórios correta para a compilação inicial.
- Execute uma compilação inicial para o 7.1-RELEASE.

Crie o diretório de correção do respectivo release no diretório `/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
```

Como exemplo, pegue o patch para [named\(8\)](#). Leia o comunicado, e pegue o arquivo necessário de [Avisos de Segurança do FreeBSD](#). Mais informações sobre a interpretação do comunicado podem ser encontradas no [Handbook do FreeBSD](#).

No [resumo de segurança](#), este comunicado é chamado `SA-09:12.bind`. Depois de baixar o arquivo, é necessário renomear o arquivo para um nível de correção apropriado. Sugere-se manter isso consistente com os níveis oficiais de correção do FreeBSD, mas seu nome pode ser escolhido livremente. Para esta compilação, vamos seguir a prática atualmente estabelecida do FreeBSD e chamar isso de `p7`. Renomeie o arquivo:

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



Ao executar uma compilação em nível de patch, supõe-se que os patches anteriores estejam no lugar. Quando uma compilação de patch é executada, ela executará todas os patches contidos no diretório de patch.

Pode haver patches personalizados adicionados a qualquer compilação. Use o número zero ou qualquer outro número.



Cabe ao administrador do Servidor de Atualização do FreeBSD tomar as medidas apropriadas para verificar a autenticidade de cada patch.

Neste ponto, um *diff* está pronto para ser construído. O software verifica primeiro para ver se um `scripts/init.sh` foi executado na respectiva versão antes de executar a construção do *diff*.

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

O que se segue é um exemplo de uma execução de uma compilação *diferencial*.

```
# 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)
ntpddate 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
...
```

As atualizações são impressas e a aprovação é solicitada.

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.

Siga o mesmo processo descrito anteriormente para aprovar uma compilação:

```
# 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.

Depois de aprovar a compilação, faça o upload do software:

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

Para referência, toda a execução do [diff.sh](#) é anexada.

7. Dicas

- Se uma versão personalizada for criada usando o procedimento `make release nativo`, o `freebsd-update-server` funcionará a partir do seu release. Como exemplo, uma versão sem ports ou documentação pode ser construída limpando funcionalidades de limpeza pertinentes às sub-rotinas de documentação `findextradocs()`, `addextradocs()` e alterando o local de download em `fetchiso()`, respectivamente, em `scripts/build.subr`. Como último passo, altere o hash `sha256(1)` no arquivo `build.conf` sob sua respectiva versão e arquitetura e você estará pronto para criar sua versão personalizada.

```
# 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 () {
}
```

- Adicionando flags `-j NUMERO` para os alvos `buildworld` e `obj` no script `scripts/build.subr` pode acelerar o processamento, dependendo do hardware usado, no entanto, não é necessário. O uso dessas flags em outros alvos não é recomendado, pois pode tornar a construção não confiável.

```
        # 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
```

- Crie um registro `DNS` apropriado para o servidor de atualizações e coloque outros por trás dele com variáveis de pesos diferentes. O uso desse recurso fornecerá espelhos de atualização, no entanto, essa dica não é necessária, a menos que você deseje fornecer um serviço 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.
```