

Gerenciamento de pacotes no Slackware

Embrulha pra viagem!

Uma afirmação freqüente é que o Slackware não dispõe de um sistema de gerenciamento de pacotes. Ele existe, e neste artigo vamos aprender a utilizá-lo corretamente e a desenvolver nossos próprios pacotes. **POR SULAMITA GARCIA**

o fez, e em dois dias o número de downloads foi tão grande que derrubou o servidor ftp. Nascia o Slackware Linux.

Os pacotes .tgz

No Slackware trabalha-se sempre tendo em mente a filosofia KISS: "Keep It Simple, Stupid!" (mantenha-o simples, estúpido!). O sistema de gerenciamento de pacotes também é assim. Os pacotes são, em sua essência, a estrutura completa que será instalada: a árvore de diretórios, contendo os binários, documentação e arquivos de configuração. É realmente um arquivo concatenado (*tar*) e compactado (*gzip*).

Experimente pegar algum pacote para o Slackware e expandi-lo com *tar zxvf*. Se isto for feito no diretório raiz, metade do trabalho de instalação está feito.

Porém não basta só isto. Junto à estrutura do pacote está o diretório *install/*, que contém os arquivos *slack-desc* e o *do-install.sh*. O *do-install.sh* é utilizado quando há a necessidade de alguma configuração extra: adicionar um usuário, configurar uma permissão de acesso especial em algum arquivo, criar links, etc. Já o *slack-desc* contém a descrição do pacote.

Dependências entre pacotes

Antes de realmente começarmos a por as mãos no teclado, é importante fazer algumas considerações. Não, os pacotes do Slackware não checam dependências. Para alguns esta é uma desvantagem, para outros uma vantagem. Cada um encara de uma maneira, e os usuários são tão diferentes quanto são diferentes as múltiplas distribuições Linux.

Em defesa da falta de checagem de dependências, o Slackware tem a característica de deixar todas as decisões do

sistema por conta do administrador. Por exemplo, a autenticação do sistema pode ser feita via LDAP, PAM ou outro sistema de autenticação qualquer, e muitas distribuições colocam um ou mais destes como um requisito para outras partes do sistema. Isto, no entanto, não é essencial: a autora deste artigo, por exemplo, não utiliza nenhum dos sistemas de autenticação citados. Então, quem pode dizer o que realmente é dependência ou não é o usuário da distribuição.

Aqui há, entretanto, algumas alternativas. Primeiro, se você acredita que não pode viver sem que o sistema calcule as dependências para você, deve procurar uma ferramenta chamada *swaret* [3], disponível no diretório *extra* do Slackware, ou alternativas como o *slapt-get* [4] e *SlackUpdate* [5].

Porém, se você quer manter tudo sob seu controle, a solução é bastante simples: se determinado programa não estiver funcionando e você desconfiar que falta alguma dependência para ele, utilize o *ldd*. Digite *ldd programa*, e se aparecer alguma linha indicando um "not found", pesquise o arquivo *Manifest*, no CD do Slackware, para saber se ele está em algum pacote. Se não, procure no Google e instale o software que fornece aquela biblioteca. Assim, o problema das dependências será resolvido.

GLOSSÁRIO

SLS: abreviação de "Soft Landing System" ou "Softlanding Linux System". Foi a primeira tentativa – ou segunda, juntamente com a MCC (Manchester Computing Center distribution) – de organizar um sistema GNU/Linux no formato que conhecemos hoje como distribuição. Ambas foram criadas em 1992. O Slackware (1993) nasceu do esforço em melhorar a SLS, mas foi muito além, sendo a mais antiga distribuição ainda disponível, uma vez que tanto SLS quanto MCC deixaram de existir.

Em 1993, um ano após o lançamento da versão 1.0 do Linux, um estudante universitário chamado Patrick Volkerding começou a utilizar uma das primeiras versões do sistema organizado em forma de distribuição, o **SLS**, para um trabalho. Percebendo que várias coisas poderiam ser melhoradas, Patrick realizou diversas modificações no sistema. Ao terminá-las, mostrou-as ao seu orientador, que o incentivou a disponibilizar o sistema na internet. Ele



O princípio de tudo: *pkgtool*

Você já se encontrou com ele, mas nem notou: o *pkgtool* é a ferramenta utilizada durante a instalação do Slackware. Se você executá-la, ela irá lhe pedir para escolher o que quer fazer: instalar ou remover pacotes, ou verificar os já instalados. Por exemplo, se você executá-la dentro do diretório *slackware/k/* do CD-ROM do Slackware, que contém os pacotes com fontes do kernel, ele lhe mostrará as opções indicadas na Figura 2.

Desta forma você pode navegar pelos pacotes instalados, selecionar outros para instalar, ou mostrar informações sobre cada pacote e decidir se quer instalá-lo ou não.

Instalando ou atualizando pacotes

O *pkgtool* tem vários “irmãozinhos”. O principal deles é o *installpkg*, que instala um pacote: ele o descompacta, executa o script *do-install.sh* e cria um arquivo no diretório */var/log/packages/* com o nome do pacote, contendo sua descrição e a lista dos arquivos instalados. Desta forma, você tem um registro de todos os pacotes de seu sistema e pode procurar facilmente o nome de um pacote em particular, a qual pacote pertence certo arquivo, etc.

Uma outra ferramenta muito útil é o *upgradepkg*. Em qualquer mirror do Slackware você encontrará o diretório *patches/packages/*, onde estão as correções (patches) dos pacotes da distribuição. Logo após uma instalação, é sempre recomendável dar uma olhada neste diretório e verificar quais atualizações são aplicáveis ao seu sistema.

Com estas atualizações em mãos, utilize o comando *upgradepkg* para atualizar os pacotes. Também é recomendável se inscrever na lista de discussão *slackware-security* [5] para ser informado via e-mail sempre que um pacote for atualizado por questões de segurança.

```
slackware http://www.slackware.com
linux
toska@toskinha:~/irssi-0.8.6/temp$ ls
irssi-0.8.6-1386-1.tgz
toska@toskinha:~/irssi-0.8.6/temp$ tar tzvf irssi-0.8.6-1386-1.tgz |more
drwxr-xr-x root/root      0 2004-05-15 15:40:13 ./
drwxr-xr-x root/root      0 2004-05-15 15:38:26 ./usr/
drwxr-xr-x root/root      0 2004-05-15 15:38:26 ./usr/etc/
-rw-r--r-- root/root    6931 2004-05-15 15:38:24 ./usr/etc/irssi.conf
drwxr-xr-x root/root      0 2004-05-15 15:38:29 ./usr/bin/
-rwxr-xr-x root/root  1632560 2004-05-15 15:38:21 ./usr/bin/irssi
drwxr-xr-x root/root      0 2004-05-15 15:38:26 ./usr/doc/
drwxr-xr-x root/root      0 2004-05-15 15:38:26 ./usr/doc/irssi-0.8.6/
-rw-rw-r-- root/root    494 2002-02-25 17:25:08 ./usr/doc/irssi-0.8.6/README
E.cygwin
-r-xr-xr-x root/root    48396 2004-04-16 20:32:42 ./usr/lib/perl5/site_perl/5.8.0/1486-linux/auto/irssi/AI/UI.so
-rw-r--r-- root/root     178 2004-05-15 15:39:18 ./usr/lib/perl5/site_perl/5.8.0/1486-linux/auto/irssi/AI/.packlist
-rw-r--r-- root/root     175 2004-05-15 15:39:18 ./usr/lib/perl5/site_perl/5.8.0/1486-linux/auto/irssi/.packlist
drwxr-xr-x root/root      0 2004-05-15 15:40:13 ./install/
-rw-r--r-- root/root     61 2004-05-15 15:40:13 ./install/description
-rw-r--r-- root/root     277 2004-05-15 15:40:13 ./install/doinst.sh
toska@toskinha:~/irssi-0.8.6/temp$
```

Figura 1: A estrutura de um pacote para o Slackware.

```
Slackware Package Tool (kgthall@osuna.06123)
Welcome to the Slackware package tool.

Which option would you like?

  [U]rge[n]t  Install packages from the current directory.
  [O]ther    Install packages from some other directory.
  [F]loppy   Install packages from floppy disks.
  [R]emove   Remove packages that are currently installed.
  [V]iew     View the list of files contained in a package.
  [S]elect   Choose Slackware installation scripts to run again.
  [E]xit     Exit Pkgtool.

  <Reitar>  <Cancelar>
```

Figura 2: O utilitário *pkgtool*.

```
PACKAGE NAME:      irssi-0.8.6-1386-1
COMPRESSED PACKAGE SIZE:  1105 K
UNCOMPRESSED PACKAGE SIZE: 3000 K
PACKAGE LOCATION: irssi-0.8.6-1386-1.tgz
PACKAGE DESCRIPTION:
irssi-0.8.6-1386-1: Irssi is a simple irc text-based client.
irssi-0.8.6-1386-1:
irssi-0.8.6-1386-1: Date 04/14/04
irssi-0.8.6-1386-1: Sulanita Garcia
irssi-0.8.6-1386-1: sulazarcia@yahoo.com.br
FILE LIST:
./
./install/
install/slack-desc
install/doinst.sh
usr/
usr/man/
usr/man/man1/
usr/man/man1/irssi.1.gz
usr/lib/
usr/lib/perl5/
```

Figura 3: O registro de um pacote em */var/log/packages*.

Removendo pacotes

Muita gente acha que no Slackware não há outra maneira de remover um pacote a não ser “na unha”. Isto é mais um mito: os pacotes instalados podem ser facilmente removidos utilizando-se o comando *removepkg*. Simples e indolor.

Criando pacotes

Agora vem a parte mais interessante da coisa: fazer seus próprios pacotes. Digamos que você tenha uma dúzia de servidores, e queira instalar um software

em todos eles. É claro que existe uma saída mais elegante do que sair recompilando o código em todas as máquinas: você pode configurar e compilar o software como quiser em uma máquina qualquer, fazer um pacote e instalá-lo em todas as outras. Há duas formas de se fazer isto: com *checkinstall* ou *makepkg*.

No diretório *extra/* da árvore do Slackware existe o programa *checkinstall*. Ele é uma ferramenta que pode criar pacotes nos formatos *tgz*, *deb* ou *rpm*. Funciona da seguinte maneira: com os fontes de um programa que você deseja instalar, execute os passos usuais para a instalação, ou seja, *./configure --opções && make*. A forma certa de fazer isso é indicar no arquivo *./configure* os diretórios corretos para binários (*--bindir=/usr/bin*), páginas de manual (man pages: *--mandir=/usr/man*), configuração (*--confdir=/etc/ exemplo*), diretório de bibliotecas (*--libdir=/usr/lib - muito importante*), etc. Rode *./configure --help* para ver uma lista com todas as opções disponíveis.

Após configurar tudo corretamente, inicie o processo de compilação digitando *make*. Depois de tudo pronto, substitua o comando *make install* por *checkinstall*. O programa vai lhe

fazer algumas perguntas e criar um pacote no formato *tgz*. O nome do arquivo será composto pelo nome do programa, a versão, a arquitetura e o release, separados por hífen.

Já disse que um pacote do Slackware é formado pelos arquivos necessários ao software, um script de pós-instalação e um arquivo de informações sobre o pacote, o *slack-desc*. O *checkinstall* cria, ao invés do *slack-desc*, um arquivo chamado *description*. É preciso corrigir isto se quisermos um pacote “oficial”.

O arquivo *do-install.sh*, que é um shell script e também não é específico do Slackware, foi preparado para trabalhar com o *slack-desc*. Para criar um pacote no formato oficial devemos modificá-lo. Não realizar tais alterações não implica no mal funcionamento do pacote, apenas irá deixá-lo no formato “incorreto”.

Dissemos anteriormente que um pacote é um arquivo concatenado e compactado no formato *tar* e *gzip*. Podemos extrair seu conteúdo com o comando *tar zxvf* ou usar uma outra ferramenta da turma do *pkgtool*, o *explodepkg*. No diretório *install/*, que foi criado durante a descompactação, devemos remover o arquivo *description* e criar nele um arquivo chamado *slack-desc*. Veja mais sobre sua estrutura no Quadro 1.

E para reempacotar o programa? O *checkinstall* é um bom utilitário para a criação de pacotes, mas a ferramenta certa para isso é o *makepkg*. Este, porém, espera a estrutura criada pelo *checkinstall*, neste caso o(s) diretório(s) de instalação e o diretório *install/*. Como esta estrutura já foi descompactada pelo *explodepkg*, podemos utilizar o *makepkg* para finalizar a geração do pacote.

```
6 - Group: [ Applications/System ]
7 - Architecture: [ i386 ]
8 - Source location: [ inssi-0.8.6 ]
9 - Alternate source location: [ ]

Enter a number to change any of them or press ENTER to continue:

*****
*** Slackware package creation selected ***
*****

Preparing Slackware install directory...OK
Writing package description...OK
Writing Slackware install script...OK
Creating package inssi-0.8.6-1386-1.tgz
Installing package...
```

Figura 4: Criando um pacote com o *checkinstall*.

```
root@toskinha:/home/toska/temp# ls
install usr
root@toskinha:/home/toska/temp# makepkg inssi-0.8.6-1386-1.tgz

Slackware package maker, version 2.1.
Searching for symbolic links:

No symbolic links were found, so we won't make an installation script.
You can make your own later in ./install/doinst.sh and rebuild the
package if you like.

This next step is optional - you can set the directories in your package
to some sane permissions. If any of the directories in your package have
special permissions, then DO NOT reset them here!

Would you like to reset all directory permissions to 755 (drwxr-xr-x) and
directory ownerships to root.root (Y/n)?
```

Figura 5: O utilitário *makepkg* em ação.

Pronto. Agora temos um pacote que não existe na árvore oficial do Slackware pronto para ser instalado. Na verdade ele já foi até instalado pelo *checkinstall*, que é um pouco intrometido. Porém podemos removê-lo facilmente com o utilitário *removepkg* e instalar o novo pacote com o *installpkg*. Depois disto, verifique o arquivo */var/log/packages/pacote*, e você verá que a lista de todos os arquivos instalados e a descrição do seu novo pacote estão lá.

Assim temos controle total sobre todos os pacotes instalados. É uma excelente idéia criar um pacote sempre que for preciso instalar um software, já que assim você terá acesso fácil às informações sobre todos os arquivos que foram colocados em seu sistema. Com o comando *grep "arquivo" /var/log/packages/** você descobre a qual pacote pertence *arquivo*. Tudo isso sem perder de vista a filosofia de simplicidade que governa o Slackware.

As instruções de empacotamento mostradas neste artigo servem para a maioria dos softwares, mas sabemos que toda regra tem sua exceção: alguns programas, como o *gmail* ou o *courrier-imap*, usam métodos não padronizados de instalação. Como este

é um assunto no mínimo tão extenso quanto este artigo, vamos deixá-lo para uma próxima edição. Até lá, treine os comandos aqui mostrados e veja como o gerenciamento de pacotes no Slackware pode ser simples e muito eficiente. ■

QUADRO 1: O ARQUIVO SLACK-DESC

Um arquivo *slack-desc* é composto por informações sobre o pacote (nome, versão, a arquitetura e o release, separados por hífen), terminadas por “:”, seguidas da descrição do pacote. Na descrição você pode colocar um pequeno resumo da função do pacote, data de criação e um endereço de contato com o autor, para que dúvidas ou problemas possam ser relatados. Exemplo:

```
An installation script was detected in ./install/doinst.sh, but
was not executed.
root@toskinha:/home/toska/temp# cd install/
root@toskinha:/home/toska/temp/install# ls
description doinst.sh
root@toskinha:/home/toska/temp/install# rm description
root@toskinha:/home/toska/temp/install# ls -l
total 4
-rw-r--r-- 1 root root 277 Abr 16 20:38 doinst.sh
root@toskinha:/home/toska/temp/install# vi slack-desc
skipping 6 old session files

wrote slack-desc, 6 lines, 216 chars
root@toskinha:/home/toska/temp/install# ls
doinst.sh slack-desc
root@toskinha:/home/toska/temp/install# cat slack-desc
inssi-0.8.6-1386-1: Inssi is a simple irc text-based client.
inssi-0.8.6-1386-1:
inssi-0.8.6-1386-1: Date 04/14/04
inssi-0.8.6-1386-1:
inssi-0.8.6-1386-1: Sulamita Garcia
inssi-0.8.6-1386-1: sulagarcia@yahoo.com.br
root@toskinha:/home/toska/temp/install#
```

INFORMAÇÕES

- [1] Página oficial do Slackware:
<http://www.slackware.com/>
- [2] Swaret:
<http://www.swaret.org/>
- [3] slapt-get:
<http://software.jaos.org/>
- [4] SlackUpdate:
<http://128.173.184.249/slackupdate/>
- [5] Listas de discussão oficiais:
<http://www.slackware.com/lists/>
- [6] Checkinstall:
<http://asic-linux.com.mx/~izto/checkinstall/>

SOBRE A AUTORA

Sulamita Garcia interessou-se por computadores ao trabalhar como auxiliar num escritório onde não a deixavam chegar perto de um. É formada em Ciências da Computação pela UFSC, onde conheceu o Linux e não largou mais. Participa do projeto LinuxChix e mantém o site de Alta Disponibilidade na UnderLinux, possuindo também certificação LPIC I. Trabalha atualmente na Cyclades Brasil como Software Designer.

