



## Upgrade de kernel

**Linux in Brazil**

Documentação original  
e de qualidade  
em bom português

[Dúvidas comuns](#) | [Perguntar no Fórum](#) | [Notícias](#) | [Mais documentos](#) | [Contato](#)

Este texto foi escrito pelo João Alexandre Voss de Oliveira (joalex@pro.via-rs.com.br), a princípio como uma mensagem na lista Linux-BR, mas ele concordou em compartilhar as dicas com os leitores do Linux in Brazil

O foco do João, e os exemplos que ele usa, são todos para um upgrade do kernel 2.2.13 para o 2.2.14. Mas todos os procedimentos expostos são genéricos o suficiente para qualquer upgrade da série 2.2, e eventualmente até mesmo das séries posteriores.

Agradecemos pela contribuição do João, que é usuário de Slackware e membro do GULE - Grupo de Usuarios Linux de Erechim - RS - BRASIL.

---

### O mini-HOWTO de atualização do kernel

#### 1. INTRODUÇÃO

Peço desculpas pela pretensão, mas como tenho observado várias dúvidas de companheiros quanto à atualização do kernel de suas distros, resolvi sair do ghost e enviar este "mini - HOWTO" indicando os passos para a atualização do kernel ...

IMHO, a única necessidade "real" de recompilar o kernel, para o usuário novato, é a de adequar o kernel recém instalado a seu tipo de processador, para não subutilizar a máquina. Para isto é interessante uma primeira compilação.

Claro que a adequação do kernel ao tipo de hardware do usuário também é um bom motivo, mas o usuário comum pode esperar um pouco até dominar um pouco mais o Linux.

Entretanto, há aqueles que não se contentam com uma simples compilação, e adoram perder horas de sono e convívio familiar, somente pelo prazer de "fuçar" na máquina. Felizmente (?) eu me enquadro nessa categoria, e senti aquela "comichão" quando vi lançado o kernel 2.2.14 - meu slackware 7.0 não podia ficar para trás :)

Apesar de eu não acreditar muito em "recipes", seguem os passos que eu executei para a atualização do kernel 2.2.13 para o 2.2.14, no slackware.

#### 2. A RECEITA ...

Os passos a seguir dizem respeito à compilação do kernel, com a construção e instalação de novos módulos, adequados às necessidades do usuário. Você deve estar logado como root (passo "b" em diante):

- a. Baixar o linux-2.2.14.tar.gz do [www.kernel.org](http://www.kernel.org) ou algum mirror;
- b. copiar o linux-2.2.14.tar.gz para /usr/src: `cp linux-2.2.14.tar.gz /usr/src`
- c. em /usr/src, excluir o link simbólico "linux": `rm linux`
- d. descompactar o fonte do kernel copiado em /usr/src - será criado um diretório chamado "linux": `tar -zxvf linux-2.2.14.tar.gz`
- e. renomear o diretório "linux" recém descompactado: `mv /usr/src/linux /usr/src/linux-2.2.14`
- f. criar um novo link simbólico "linux", desta vez apontando para o novo diretório linux-2.2.14: `ln -s /usr/src/linux-2.2.14 linux`

Neste ponto você deve ter, em /usr/src, os dois diretórios e um symlink:

```
# pwd
# /usr/src
# ls
# linux@ linux-2.2.13 linux-2.2.14 rpm sendmail
```

Essa é a listagem em /usr/src em meu sistema.

O link simbólico deve apontar para o diretório linux-2.2.14. É nele que o trabalho a seguir deve ser feito. Não delete o diretório linux-2.2.13 ainda, voce pode precisar dele mais tarde, se as coisas derem erradas.

Continuando, os próximos passos DEVEM ser executados em /usr/src/linux;

```
# cd /usr/src/linux
# make menuconfig
```

O comando acima iniciará a configuração em modo gráfico do kernel.

Opcionalmente, estando em algum ambiente X (KDE, Windowmaker, Blackbox ...), o comando poderá ser:

```
# make xconfig
```

Escolha a opção "Store configuration to file" e salve a configuração atual em / com um nome bem claro, tipo "config.original" ou algo parecido.

Mais tarde, se vc precisar restaurar a configuração original, basta rodar o make menuconfig e carregar os parâmetros a partir deste arquivo.

Em seguida escolha a opção "Quit without save" e saia.

Continuando, digite (em /usr/src/linux):

```
# make mrproper
```

Isto apagará a configuração anterior do kernel no sistema.

Em seguida, novamente:

```
# make menuconfig (ou make xconfig, se for o caso)
```

Escolha as opções que quiser, adapte o kernel ao seu hardware. Salve a configuração.

Novamente no prompt:

```
# make dep
# make clean
# make zImage
```

Oa comandos acima criam as dependências necessárias, limpam arquivos temporários e de instalação e criam a imagem compactada do novo kernel, em /usr/src/linux/arch/i386/boot.

No meu caso, como a imagem do kernel a ser gerado é muito grande, tenho de usar "make bzImage" ao invés de "make zImage". Significa, literalmente, "make a BIG zImage" ... :)

Após algum tempo (que varia de acordo com a máquina), a imagem é gerada e gravada no diretório acima referido.

Vá agora para o diretório /lib/modules. O comando ls dá a listagem do conteúdo:

```
# cd /lib/modules
# ls
# 2.2.13
```

O único diretório listado, 2.2.13, é onde encontram-se os módulos originais da primeira instalação. Caso vc esteja apenas compilando um kernel existente, renomeie para, por exemplo, 2.2.13.old. Caso seja uma atualização, deixe intacto, pois a seguir será criado um novo diretório.

Volte para /usr/src/linux:

```
# cd /usr/src/linux
```

Digite os comandos:

```
# make modules
# make modules_install
```

Os comandos acima criarão e instalarão os novos módulos em /lib/modules. No caso da atualização desta receita, será criado um diretório "2.2.14" em /lib/modules.

O trabalho de criação dos módulos pode demorar um pouco, dependendo do tipo de máquina (processador, RAM disponível, etc).

Após o prompt ser novamente liberado (!), mova a imagem compactada do kernel, de /usr/src/linux/arch/i386/boot para o diretório raiz (/):

```
# mv /usr/src/linux/arch/i386/boot/bzImage /
```

Vá para o diretório raiz, e renomeie o novo kernel recém compilado para um nome facilmente identificável:

```
# cd /
# mv bzImage vmlinuz-2.2.14
```

O novo kernel, agora, terá o nome acima, vmlinuz-2.2.14.

Observe que será necessário indicar, ao LILO, qual a imagem compactada que deverá ser inicializada no boot da máquina. Vá para /etc e edite o arquivo "lilo.conf":

```
# cd /etc
# pico lilo.conf
```

Meu /etc/lilo.conf se parece com isto:

```
# LILO configuration file
# generated by 'liloconfig'
#
# Start LILO global section
boot = /dev/hda
message = /boot/boot_message.txt
prompt
timeout = 20
vga = normal
image = /vmlinuz
    root = /dev/hda1
    label = Linux
    read-only
# Linux bootable partition config ends
```

Veja que apenas uma imagem pode ser inicializada. Para adicionar a nova imagem, na seção:

```
image = /vmlinuz
root = /dev/hda1
label = Linux
read-only
```

adicione mais uma entrada:

```
image = /vmlinuz-2.2.14
root = /dev/hda1
label = New
read-only
```

O arquivo lilo.conf deve ficar assim:

```
# LILO configuration file
# generated by 'liloconfig'
```

```
#
# Start LILO global section
boot = /dev/hda
message = /boot/boot_message.txt
prompt
timeout = 20
vga = normal
image = /vmlinuz
    root = /dev/hda1
    label = Linux
    read-only
image = /vmlinuz-2.2.14
    root = /dev/hda1
    label = New
    read-only
# Linux bootable partition config ends
```

Salve o arquivo lilo.conf. No prompt, digite:

```
# lilo
```

para salvar as alterações. Você poderá ver:

```
Added
Linux *
New
```

indicando que a imagem vmlinuz-2.2.14 poderá ser inicializada no boot da máquina digitando-se "new" no prompt do lilo. Posteriormente vc poderá mesmo eliminar a imagem antiga (vmlinuz) em "/" e mesmo deletar o diretório /lib/modules/2.2.13. Mas espere alguns dias, e observe se não ocorrem erros em módulos, e se todas as funcionalidades estão operantes (ppp, etc).

Efetue um reboot na máquina, e a nova inicialização deverá apresentar

```
Welcome to linux 2.2.14
```

indicando que tudo correu como devia e o novo kernel está instalado.

boa diversão!

### 3. SE AS COISAS DEREM ERRADO ...

Observe as mensagens de inicialização, às vezes alguns módulos teimam em não ser instalados corretamente ...

Em caso de kernel panic, inicialize a partir de um disco de inicialização (você não tem um? o que está esperando para fazer um?) e efetue as alterações necessárias. É possível voltar ao antigo kernel 2.2.13, simplesmente desfazendo as alterações. Lembre-se que o comando "linux" digitado no prompt do lilo inicializa a imagem antiga do kernel 2.2.13, e o comando "new" a nova imagem, vmlinuz-2.2.14.

### 4. CONCLUSÃO

Não deixe de consultar a documentação referente ao kernel, em especial o "kernel HOWTO" (em linuxdoc.org) e o guia para o kernel 2.2.XX.

Como dito, esta é uma receita simples, que visa orientar os iniciantes a trilharem o caminho das pedras ... não tem a pretensão de ser nada mais além disto.

Espero ter sido útil, estamos à disposição para eventuais esclarecimentos (e mesmo correções por parte dos "mais velhos" da lista ...

```
+-----+
|  .^.   | João Alexandre Voss de Oliveira          |
|  /V\   | Registered Linux User # 147244 - Linux Machine # 63929 |
| // \\\ | e-mail: joalex@pro.via-rs.com.br //  joalex@via-rs.net |
| /(. .)\ | GULE - Grupo de Usuarios Linux de Erechim - RS - BRASIL |
| ^^--^^ | "God is in His heaven. All`s right with the world"      |
| #147244 | Powered by Slackware Linux 7.0 Kernel 2.2.14          |
+-----+
```



Usando a mini distribuição "TOMSRTBT" para recuperar seu Boot Loader

tomsrtbt is "The most GNU/Linux on one floppy disk" for:

- rescue recovery panic & emergencies
- tools to keep in your shirt pockets
- whenever you can't use a hard drive
- > <http://www.toms.net/rb/>

Geralmente a perda de boot loader acontece quando você instala primeiro alguma distribuição GNU/Linux e depois algum tipo de Windows, o qual irá sobrescrever a MBR sem lhe falar nada. Isso é um problema muito comun entre os usuários que estão iniciando no ambiente GNU/Linux !!!! Mas, é claro tem resolução. Então levando-se em conta que essa é sua situação e que você só consegue entrar no Windows, faça o download do arquivo para DOS, veja os links abaixo:

Arquivo para Linux

<http://www.tux.org/pub/distributions/tinylinux/tomsrtbt/tomsrtbt-2.0.103.tar.gz>

Arquivo para DOS

<http://www.tux.org/pub/distributions/tinylinux/tomsrtbt/tomsrtbt-2.0.103.dos.zip>

Para usar o arquivo para DOS, você deve:

- descompactá-lo em alguma pasta do seu sistema (Winzip, Filzip, ...)
- diskete em mãos
- vá para a pasta onde descompactou o arquivo .zip e digite:  
install
- siga as instruções na tela e pronto....você terá seu diskete de recuperação
- se tiver mais dúvidas leia a FAQ que tem no arquivo zip

Para usar o arquivo para Linux, você deve:

- faça tudo como usuário root no sistema
- descompactá-lo em algum diretório do seu sistema  
cd diretório (onde está o arquivo .tar.gz)  
tar xzvf tomsrtbt-2.0.103.tar.gz
- diskete em mãos

- digite:

```
./install.s
```

- siga as instruções na tela e pronto....você terá seu diskete de recuperação

- se tiver mais dúvidas leia a FAQ que tem no arquivo .tar.gz

Feito o diskete, vamos proseguir:

- dê um boot com o diskete criado

- no final do boot entre com o usuário

```
root
```

- a senha é

```
xxxx
```

Feito o login no sistema, vamos prosseguir:

- execute

```
fdisk -l
```

- com isso vc vai ver as partições do seu sistema, bem nesse caso você deve conhecer como é a nomenclatura para dispositivos em ambiente GNU/Linux - não irei abordar este assunto aqui.

- se seu boot loader é o LILLO, e você tem uma partição que representa o "/" e uma partição que representa a "swap"....faça o seguinte:

```
mkdir /mnt/hd1
```

```
mount -t ext2 (ou ext3) /dev/hda3 /mnt/hd1
```

\_\_\_\_|\_|\_\_\_\_ -aqui tá o problema, você tem que descobrir a sua partição através do comando fdisk -l, como foi dito logo acima !!

obs: é ext2 ou ext3

depois disso faça:

```
chroot /mnt/hd1 /sbin/lilo
```

pronto está gerado seu boot loader novamente na MBR do disco !

-> -> você deve atentar aqui para que não tenha o /etc em partição separada, pois com essa configuração a recuperação ocorre de outra forma !!!

- Agora, se seu boot loader for o GRUB, então faça o seguinte:  
chroot /mnt/hd1 /usr/sbin/grub-install

pronto está gerado seu boot loader novamente na MBR do disco !

-> você deve atentar aqui para que não tenha o /boot em partição separada,  
pois com essa configuração a recuperação ocorre de outra forma !!!

Até mais !!!

Autor:  
Eduardo de Oliveira Hernandes  
eduardo@kees.com.br  
Kees Informática LTDA