

Um “sabor” para todos os gostos

# Gentoo Linux

O Gentoo Linux nada mais é do que uma distribuição Linux voltada para a otimização e personalização de pacotes. Seu alto desempenho, configurabilidade e uma enorme equipe de desenvolvedores espalhados pelo mundo fizeram desta uma das distribuições mais populares.

POR MARCELO VERIATO LIMA E WILLIAN FERRAZ

Neste artigo vamos apresentar o *Gentoo Linux* aos novos usuários, ensinar a instalar o sistema e desfazer alguns mitos a respeito dessa distribuição. O Gentoo Linux nasceu de uma idéia, a princípio solitária, de Daniel Robbins, que estava descontente com as distribuições que existiam à época, com sua falta de otimizações e gerenciamento de pacotes problemático, entre outras coisas. Porém o principal motivo era que em nenhuma distribuição o usuário podia ter certeza do que estava instalando através de um gerenciador de pacotes. Então, além de projetar o Gentoo, Daniel Robbins também projetou a *Portage*, segundo ele uma ferramenta “mais do que perfeita” comparada com as que existiam na época. Ela foi escrita com foco na flexibilidade do gerenciamento de pacotes.

## Portage

Para entender o que é a *Portage* faremos uma analogia com o corpo humano: a *Portage* é comparada às veias que levam o sangue ao coração da distribuição. Através dela é feito todo o trabalho de gerenciamento de pacotes, dependências, busca por informações e atualização de tudo o que está instalado na distribuição. Como o próprio Daniel Robbins disse, “o povo gostou do conceito da *Portage*” e com isso o Gentoo cresceu rapidamente, tornando-se conhecido como a distribuição *From Source* (A partir do

código fonte), o que é um dos aspectos fundamentais e mais importantes do Gentoo Linux.

No início foi complicado ter uma distribuição baseada no código fonte, mas hoje isso é feito sem problemas. Há quem pense que o Gentoo baseia-se apenas no código-fonte, o que não é exatamente verdade: existem vários repositórios não oficiais com pacotes binários para as mais diversas arquiteturas. O projeto da *Portage-ng* (*New Generation*) está caminhando de forma rápida e, logo logo, haverá binários de toda a árvore da *Portage*. O suporte a pacotes binários é muito importante porque eles são largamente utilizados pela comunidade Linux, já que grande parte dos usuários não pode esperar pela compilação de um determinado pacote.

A filosofia do Gentoo se resume em um parágrafo: permitir que o usuário faça o que bem entender com a máquina e desenvolver ferramentas que possibilitem a ele trabalhar de forma agradável e eficiente na instalação de novos pacotes e manutenção do sistema. Nossas ferramentas devem ajudar o usuário a apreciar o Linux e a comunidade do Software Livre. Isso só é possível quando a ferramenta é projetada para refletir e transmitir a vontade do usuário. Se a ferramenta o força a fazer coisas de um modo pré-determinado, ela causará desconforto e estará trabalhando contra o usuário, ao invés de para ele.

Em muitos casos, uma ferramenta parece impor ao usuário “sua vontade”. Com a *Portage* é diferente: absolutamente todas as definições do pacote são atribuídas ao usuário. Outra coisa interessante: a filosofia do Gentoo Linux visa criar ferramentas melhores. Quando uma ferramenta está fazendo o seu trabalho perfeitamente, você não fica muito ciente de sua presença porque a mesma não interfere nas suas decisões, nem o força a interagir com ela quando não quer. A ferramenta serve melhor ao usuário do que o usuário à ferramenta. No futuro, o objetivo do Gentoo é continuar a criar ferramentas quase ideais para as necessidades de todos, ou seja, que possam acomodar tudo de que diferentes usuários possam precisar. Você gosta de uma distribuição ou de uma ferramenta que faz realmente o que você quer fazer? A missão do Gentoo Linux é dar a quem o utiliza essa sensação de saber tudo o que existe instalado no computador e utilizar as ferramentas que preferir.

## Guia de instalação

O que muitas pessoas sentem falta no Gentoo é de um instalador amigável. Porém se pegarmos o *Handbook* [1] (o manual oficial de instalação) e o lermos com calma veremos que um instalador não é realmente necessário. O guia é muito didático e mesmo quem nunca trabalhou com Linux já conhecerá um pouco mais sobre o sistema e seus prin-

cipais arquivos apenas com essa leitura. Alguns reclamam que o Handbook “enche lingüiça”, ou seja, enrola demais até chegar ao ponto. Para esses, existe o *Quick Installation Reference* [2], um guia rápido de instalação muito bom.

## Desenvolvedores

Qualquer pessoa pode se tornar um desenvolvedor do Gentoo Linux – não é preciso ser um programador nato. Basta testar novos pacotes, informar sobre erros e problemas de compilação ou escrever ebuilds (receitas de compilação em Shell Script) para novos programas. Para isso contamos com um sistema *Bugzilla* muito completo. É só se cadastrar [3] e começar a postar relatórios de erros ou correções. O mais interessante é observar a rapidez com que os bugs são corrigidos e um novo patch ou ebuild é colocado na árvore da Portage.

## Documentação

Outra coisa que o pessoal considera excelente é a forma da organização de toda a documentação do Gentoo Linux, dividida por áreas: desde a instalação, documentação para o usuário (Desktop), administração de rede (Serviços), desenvolvimento de ebuilds e como criar e/ou traduzir documentos. Há também um fórum [4] onde os problemas mais comuns, dicas e *how-tos* são encontrados facilmente; existem ainda os canais IRC dentro da rede Freenode.net ([irc.freenode.net](http://irc.freenode.net)) onde todos os desenvolvedores se encontram, como o [#gentoo-dev](http://irc.freenode.net) e o [#gentoo-bugs](http://irc.freenode.net). Claro que existem muitos outros, como canais divididos por arquitetura e grupos como o [#gentoo-sound](http://irc.freenode.net), que cuida apenas do conjunto de ferramentas relacionadas a som dentro da árvore Portage. A documentação em Português do Brasil ainda está sendo feita. Quem tiver vontade de contribuir deve mandar um email para o líder do projeto *Gentoo-docs pt\_BR*, no endereço [anderson@gentoo.org](mailto:anderson@gentoo.org).

## Grupo GentooBR

O grupo de usuários *Gentoo Linux Brasil* (GentooBR), do qual fazemos parte, foi formado pelos usuários Otávio Piske e Marcos Vieira há dois anos e meio para atender aos usuários da distribuição dentro do Brasil. Não somos muitos, porém todos se ajudam, conversam e contribuem com a distribuição. Divulgamos o Gentoo Linux em palestras e eventos relacionados ao Software Livre, nos quais muitos usuários nos abordam para conversar sobre o sistema, saber onde podem conseguir-lo e como instalar.

Contamos com um portal [5] onde postamos dicas, how-tos, comentários e demais coisas relacionadas ao Gentoo. Seguimos algumas filosofias, como não ramificar o grupo por região: o GentooBR sempre será o mesmo em qualquer lugar do Brasil. Nosso canal IRC é também o canal oficial do Gentoo Linux no Brasil: [#GentooBR](http://irc.freenode.net) na mesma rede [irc.freenode.net](http://irc.freenode.net).

## Instalando o Gentoo Linux

Vamos dar início à instalação, seguindo o passo-a-passo da forma como está no Handbook. A versão usada é a 2005.0, encartada como CD do Mês nesta edição da Linux Magazine Brasil.

A instalação do Gentoo Linux pode ser feita a partir de 3 *Stages* (estágios). Os Stages são como pontos de partida por onde iremos começar a instalação. Veja os prós e contras de cada um:

⇒ **Stage 1** – O usuário terá que compilar toda a distribuição, desde o compilador `gcc` até a `glibc` (processo chamado *bootstrap*), ou seja, toda a base do sistema. Esta opção é ideal para quem quer o máximo desempenho e tem tempo para instalar. Ela permite que você tenha controle total sobre os ajustes de otimização e personalização da distribuição. É apropriada para os *power users* que sabem o que está sendo feito ou para quem gosta de “fuçar”, e permite que você aprenda mais sobre

o funcionamento interno do Gentoo. Entretanto, instalar o Gentoo Linux pelo Stage 1 é demorado e, se você não pretende fazer ajustes finos de compilação, é provavelmente um desperdício do tempo. Não é possível instalar a partir do Stage 1 sem acesso à Internet.

⇒ **Stage 2** – É o Stage 1 pré-compilado para um processador específico e com otimizações feitas pela equipe do Gentoo Linux. Com ele não é necessário realizar o bootstrap, é mais rápido do que começar pelo Stage 1 e você ainda pode personalizar parte do sistema (com o comando `emerge system`). Entretanto, não é possível personalizá-lo tanto como no Stage 1. Essa não é a maneira a mais rápida instalar o Gentoo e você tem que aceitar as otimizações que foram escolhidas para o bootstrap pela equipe de desenvolvedores do Gentoo. Também não é possível instalar a partir do Stage 2 sem acesso à Internet.

⇒ **Stage 3** – Escolhendo o Stage 3 a instalação será a mais rápida, porém as otimizações serão as escolhidas pela equipe de desenvolvimento do Gentoo Linux. Para sermos honestos, os ajustes foram escolhidos com cuidado para realçar o desempenho ao mesmo tempo em que mantém a estabilidade. Utilize esse Stage caso você não tenha uma conexão à Internet durante a instalação. Apesar de ser a maneira mais rápida de instalar o Gentoo Linux, você não pode personalizar o sistema base, pois ele já vem pronto.

Nossa instalação será feita a partir do Stage 1, portanto precisamos de acesso à Internet, de preferência via banda larga (ou muita paciência, caso contrário).

## Carregando o LiveCD

Há duas imagens ISO do Gentoo Linux nos *mirrors* oficiais, os CDs *minimal* e *universal*. No *minimal* vêm apenas ferramentas de boot e alguns utilitários; teríamos que fazer o download do stage

a partir do qual desejamos instalar e também de um *snapshot* da árvore da Portage. Já a imagem universal vem com os 3 Stages, um snapshot da Portage e os *distfiles* (código fonte dos programas) para uma instalação sem Internet. Vamos utilizar o CD universal.

Após reiniciar o computador pelo LiveCD podemos escolher quatro imagens do kernel diferentes: *gentoo* (kernel com suporte a *framebuffer*), *gentoo-nofb* (kernel sem suporte a *framebuffer*), *smp* (kernel para máquinas com dois ou mais processadores, com *framebuffer*) e *smp-nofb* (para máquinas “multi-processadas”, sem *framebuffer*). Algumas opções extras também estão disponíveis: tecla [F2] e confira. Caso necessite de alguma delas use-as na forma *kernel opção*, como por exemplo: *gentoo-nofb nodhcp nopcmcia*.

## Particionando o disco

Vamos particionar nosso disco como mostrado no Handbook. Digite o comando abaixo:

```
# cfdisk /dev/hda
```

Onde */dev/hda* é o disco *master* na primeira interface IDE. O *cfdisk* é um utilitário de particionamento mais amigável que o popular *fdisk*.

Vamos criar uma partição de “boot” */dev/hda1* com 80 MB e marcá-la como *Bootable* (inicializável). A seguir criamos uma partição de *swap* (também conhecida como memória virtual) */dev/hda2*, tipo *Linux Swap*, com um tamanho igual ao dobro de memória RAM instalada no computador. Pessoalmente, acho que o ideal é seguir essa regra até um limite de 512 MB de *swap*. Por último uma partição */dev/hda3* com o resto do espaço livre no disco. A tabela de particionamento ficará como na [tabela 1](#).

## Criando o sistema de arquivos e montando unidades

Particularmente, gosto de mesclar entre os sistemas de arquivo *ReiserFS* e *Ext3*. Vamos criar o sistema de arquivos de nossa primeira partição (boot) como *Ext3*. Digite o comando:

```
# mke2fs -j /dev/hda1
```

**Tabela 1: Tabela de Particionamento**

Partição	Opções	Tipo part.	Tipo SA	Size (MB)
hda1	Bootable	Primária	Linux	80,00
hda2		Primária	Linux Swap	2x a memória RAM
hda3		Primária	Linux	Resto do disco

A seguir, vamos criar um sistema de arquivos *ReiserFS* em nossa partição raiz:

```
# mkreiserfs /dev/hda3
```

Confirme a alteração teclando [y]. Por último, vamos criar nossa partição de *swap*:

```
# mkswap /dev/hda2
```

E em seguida, vamos ativá-la, com o comando:

```
# swapon /dev/hda2
```

Depois de criar os sistemas de arquivos nas partições, vamos montá-las dentro do diretório */mnt/gentoo*:

```
# mount /dev/hda3 /mnt/gentoo
# mkdir /mnt/gentoo/boot
# mount /dev/hda1 /mnt/gentoo/boot
```



## Configurando a rede

Como estamos instalando o Gentoo pelo Stage 1, necessitamos de uma conexão à Internet, de preferência de banda larga. Vamos explicar os dois casos que podem ocorrer freqüentemente:

**1)** Caso você possua um servidor DHCP na rede, o Gentoo Linux irá pegar um endereço IP automaticamente. Claro que você também pode configurar o endereço IP manualmente:

```
# ifconfig <eth0> <endereço IP da rede> netmask <máscara> up
```

`eth0` é a interface de rede, `endereço IP da rede` é o endereço de comunicação na rede interna e `netmask` a máscara de rede. Veja:

```
ifconfig eth0 192.168.0.5 netmask 255.255.255.0 up
```

Vamos agora configurar nosso gateway padrão, na forma:

```
# route add default gw <ip do Gateway>
```

Exemplo:

```
# route add default gw 192.168.0.1
```

Depois de configurar o gateway padrão edite o arquivo `/etc/resolv.conf`. No Gentoo Linux o editor de texto padrão ainda é o `nano`, um editor simples e com poucos recursos, mas que faz o serviço. Digite:

```
# nano -w /etc/resolv.conf
```

E adicione a linha a seguir ao arquivo:

```
nameserver 200.176.2.10
```

Esse endereço IP é um servidor DNS público do Terra Networks. Salve o arquivo teclando `[Ctrl]+[x]`.

Para ter certeza de que sua conexão está funcionando digite:

```
# ping www.meudominiopreferido.com.br
```

**2)** Caso sua conexão à Internet seja feita usando o protocolo PPPoE (comum em links ADSL), utilizaremos o software `RP-PPPoE` para configurá-la. Digite o comando:

```
# adsl-setup
```

Siga os passos até o final e, logo em seguida, digite

```
# adsl-start
```

Se o resultado for a mensagem `Connected!`, meus parabéns! Você está na Internet!

## Instalando o Stage 1 e o snapshot da Portage

Antes de tudo, verifique a data do seu sistema com o comando `date`. Caso ela esteja incorreta vamos acertá-la. `date` recebe como parâmetro uma data no formato `MMDDHHMnMnA-AAA`, ou seja, **Mês**, **Dia**, **Hora**, **Minuto** e **Ano**, cada um com dois dígitos (exceto ano, com quatro). Por exemplo, a data de *22 de Abril de 2005 às 23:52* é representada por: `date 042223542005`.

Após acertar a data do sistema, vamos entrar no diretório `/mnt/gentoo` (com o comando `cd /mnt/gentoo`) e descompactar o arquivo contendo o Stage 1. Digite:

```
# tar -xvjpg /mnt/cdrom/stages/stage1*
```

Pronto! Agora que o Stage 1 já se encontra no diretório correto, vamos instalar a árvore da Portage:

```
# tar -xvzf /mnt/cdrom/snapshots/portage* -C /mnt/gentoo/usr
```



## Configurando as flags para o compilador

A partir de agora iremos configurar as *flags* para o compilador C com os parâmetros de compilação, *USE Flags* e demais coisas. Digite o comando:

```
# nano -w /mnt/gentoo/etc/make.conf
```

A variável *USE* é responsável pelas *USE Flags*; por exemplo, se você deseja o suporte a ALSA em seus pacotes, adicione a entrada "alsa" a essa variável. Outras *USE Flags* podem ser adicionadas, mas na nossa instalação básica isso não fará diferença alguma.

Uma lista das variáveis *USE* pode ser encontrada no arquivo `/usr/portage/profile/use.desc`.

Na variável *CHOST* definimos a arquitetura para a qual o GCC irá compilar os programas. Para máquinas i686 (Pentium II em diante) utilizamos `i686-pc-linux-gnu`, para i386 usamos `i386-pc-linux-gnu`. As flags para as demais arquiteturas estão disponíveis no próprio Handbook.

As *CFLAGS* e *CXXFLAGS* são as mais importantes e devem ser configuradas com cuidado. Na variável *CFLAGS* indicamos a arquitetura de nosso processador (com `-march=`) e flags de otimização para o compilador GCC. Eu particularmente utilizo apenas três flags: `-O2`, `-pipe` e `-fomit-frame-pointer`.

Então, se meu processador for um Intel Pentium 4, utilizo as flags:

```
CFLAGS="-march=pentium4 -O2 -pipe"
```

Essas *CFLAGS* indicam que o sistema será compilado com otimizações específicas (e exclusivas) para processadores Intel Pentium 4 (`-march=pentium4`), procurará otimizar ainda mais o código fonte gerado (`-O2`) e utilizará *pipes* (`-pipe`) para a comunicação entre os estágios da compilação, o que acelera um pouco o processo.

**Atenção:** Lembre-se de que programas compilados com o parâmetro `-march=<alvo>` irão funcionar apenas na CPU especificada, no nosso caso Pentium 4. Verifique com o comando `cat /proc/cpuinfo` qual o seu processador e indique-o. Para gerar código mais genérico utilize `-mcpu=i686` ou `-mcpu=i386`, dependendo da arquitetura de seu processador, no lugar de `-march`. No caso de processadores Duron, Sempron ou Celeron, o parâmetro `-march` varia de acordo com o core/núcleo do chip.

No site da comunidade Gentoobr.org você encontra uma referência muito boa sobre essas otimizações [6].

Na variável *MAKEOPTS* colocamos `-j2`; ela define quantas compilações paralelas poderemos ter. Para máquinas com dois ou mais processadores informe o número de processadores mais 1. Por exemplo: `-j3` para uma máquina com dois processadores.

## Criando o ambiente

O primeiro passo é selecionar um dos servidores espelho (*mirrors*). Digite o comando a seguir:

```
# mirrorselect -i -o >> /mnt/gentoo/etc/make.conf
```

Escolha os mirrors que bem entender, todos são relativamente bons. O próximo passo é copiar o arquivo `/etc/resolv.conf` para `/mnt/gentoo/etc`:

```
# cp -L /etc/resolv.conf /mnt/gentoo/etc/
```

Agora monte o diretório `/proc` sob o diretório `/mnt/gentoo`:

```
# mount -t proc none /mnt/gentoo/proc
```

Agora vem a etapa que tanto esperávamos: criar o ambiente básico para começar a compilação do sistema. Digite:

```
# chroot /mnt/gentoo /bin/bash
```

E exporte as variáveis de ambiente com o comando abaixo:

```
# env-update
# source /etc/profile
```

Parabéns, estamos dentro do ambiente do Gentoo Linux! Claro, ainda faltam alguns passos para terminar a instalação. Sincronize a lista de "pacotes" com as listas oficiais online:

```
# emerge sync
```

O comando anterior utiliza o protocolo *rsync*. Caso tenha problemas com alguma espécie de firewall utilize:

```
# emerge-websync
```

## Bootstrap e instalação do sistema

Nesta etapa iremos compilar toda a base do sistema incluindo o GCC, *glibc*, *ncurses*, *zlib* e demais ferramentas necessárias ao funcionamento do sistema. Digite:

```
# cd /usr/portage
# scripts/bootstrap.sh
```

Dependendo de sua conexão com a Internet e da velocidade de seu processador, esse estágio pode demorar um pouco, em média de duas a três horas.

Lembramos que o bootstrap só é necessário para quem vai instalar o Gentoo pelo Stage 1, como é o nosso caso.

Logo após o fim do bootstrap, vamos atualizar as variáveis de ambiente. Digite o comando:

```
# env-update
# source /etc/profile
```

Agora vem outro passo muito importante, a instalação das ferramentas básicas necessárias para que o sistema funcione corretamente:

```
# emerge system
```

Esse processo também demora em torno de duas a três horas (mais se sua máquina for antiga), assim como o bootstrap. Ignore os avisos (*warning*) mostrados na tela. Após o término do processo de compilação, vamos atualizar alguns arquivos de configuração com o comando:

```
# etc-update
```

## Configurando o kernel

Antes de tudo escolha o fuso horário (*timezone*), no nosso caso *Brazil-East*:

```
# ln -sf /usr/share/zoneinfo/Brazil/East /etc/localtime
```

Podemos utilizar várias versões do kernel em nosso sistema Gentoo Linux. Vou explicar quatro delas:

- *vanilla-sources* – Kernel 2.4 “padrão” criado pelos desenvolvedores oficiais do kernel Linux, sem patches.
- *gentoo-sources* – Kernel 2.4 com vários patches que corrigem falhas de segurança, melhoram o desempenho e trazem atualizações, preparado pelos desenvolvedores do Gentoo.
- *development-sources* – Kernel 2.6 “padrão” sem patches, assim como o *vanilla-sources*.
- *gentoo-dev-sources* – O mesmo que o *gentoo-sources*, mas na versão 2.6. É o recomendado e o que vamos utilizar. Digite o comando:

```
# emerge gentoo-dev-sources
```

Lembre-se de conferir se o link simbólico `/usr/src/linux` aponta para o diretório correto contendo o código fonte do kernel Linux.

```
# ls -l /usr/src/
lrwxrwxrwx 1 root root 22 Set 30 19:52 linux -> linux-2.6.XX-gentoo-rX
```

Caso o link não esteja correto, execute os comandos abaixo:

```
# rm -rf /usr/src/linux
# cd /usr/src
# ln -s linux-2.6.XX-gentoo-rX linux
```

Troque os “X” pela versão corrente do seu kernel. Para quem sabe compilar o kernel manualmente os passos são os mesmos de sempre:

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

Configure as opções do kernel ao seu gosto. Lembre-se apenas de preencher alguns requisitos básicos, mostrados a seguir. Um \* em frente à uma opção significa que ela está selecionada:

```
Code maturity level options ->
[*] Prompt for development and/or incomplete code/drivers
File systems ->
Pseudo Filesystems ->
[*] /proc file system support
[*] /dev file system support (OBSOLETE)
[*] Automatically mount at boot
[*] Virtual memory file system support (former shm fs)
<*> Reiserfs support
<*> Ext3 journalling file system support
<*> Second extended fs support
```

Caso tenha uma conexão ADSL via protocolo PPP (PPPoE, como no *Speedy*), é necessário habilitar opções extras:

```
Device Drivers ->
Networking support ->
<*> PPP (point-to-point protocol) support
<*> PPP support for async serial ports
<*> PPP support for sync tty ports
```

Compile o kernel digitando:

```
# make && make modules_install
# cp arch/i386/boot/bzImage /boot/kernel-2.6.XX-gentoo-rX
```

Para quem não sabe configurar o kernel há uma alternativa, o *genkernel*, uma ferramenta que ajudará a compilar o kernel automaticamente. Digite:

```
# emerge genkernel
```

Você pode fazer uma compilação totalmente automática do kernel (digite: `genkernel all`) ou acessar um menu de configuração (digite: `genkernel -menuconfig all`). Lembramos que no caso da instalação automática o kernel será totalmente modular.

Para hardware especial ou mais recente há alguns passos a mais. Veja na [tabela 2](#) o que é necessário instalar em cada caso:

Para carregar os módulos, como suporte à placa de rede, áudio ou outros periféricos, digite o comando:

```
# nano -w /etc/modules.autoload.d/kernel-2.6
```

Adicione uma nova linha com a entrada correspondente ao seu módulo; por exemplo, para placas de rede RTL8139 digite

**Tabela 2: Módulos extras do kernel**

Ebuild	O que é	Comando
nvidia-kernel	Aceleração 3D para placas NVIDIA	<code>emerge nvidia-kernel</code>
nforce-net	Rede on-board nas placas-mãe NVIDIA Nforce2	<code>emerge nforce-net</code>
nforce-audio	Som on-board nas placas-mãe NVIDIA Nforce2	<code>emerge nforce-audio</code>
e100	Placa de rede Intel e100	<code>emerge e100</code>
e1000	Placa de rede Intel e1000	<code>emerge e1000</code>
ati-drivers	Aceleração gráfica para placas ATI	<code>emerge ati-drivers</code>
ati-drivers-extra	Ferramentas gráficas para ATI	<code>emerge ati-drivers-extra</code>

### Dica do Geek Slack

Após instalar o sistema instale um pacote chamado *esearch*:

```
# emerge esearch
# eupdatedb
```

Esse pacote faz um “banco de dados” dos pacotes instalados no sistema e após o *esync* (*emerge -sync*) relaciona os pacotes que foram recentemente atualizados na Árvore da Portage. Para maiores informações sobre o uso do *esearch* consulte a página de manual do programa (com o comando *man esearch*).

8139too, para placas Via-Rhine digite *via-rhine* etc. Salve o arquivo e atualize a lista de módulos a carregar:

```
# modules-update
```

### Configurações finais

Vamos configurar o *fstab*, arquivo que contém as informações sobre as partições que devem ser montadas automaticamente e os pontos de montagem. Digite:

```
# nano -w /etc/fstab
```

Altere as opções como mostrado na **tabela 3** e salve o arquivo. Vamos definir o *hostname* da máquina. No nosso exemplo, o nome será *larry*. *Larry* é o nome do mascote do Gentoo Linux, *Larry, The Cow*:

```
# echo larry > /etc/hostname
```

### Configurando a rede:

Edite o arquivo de configuração da rede com o comando a seguir:

```
# nano -w /etc/conf.d/net
```

Altere os parâmetros conforme sua rede e suas interfaces; altere também o seu gateway. Caso utilize o protocolo PPP para conexão, deixe o gateway comentado. Para fazer com que a placa de rede “pegue” um endereço IP automaticamente durante o boot, digite:

```
# rc-update add net.eth0 default
```

Caso possua mais de uma placa de rede, você deve digitar:

```
# cd /etc/init.d
# ln -s net.eth0 net.eth1
# rc-update add net.eth1 default
```

Agora configure os hosts no arquivo */etc/hosts*:

```
# nano -w /etc/hosts
127.0.0.1 localhost
192.168.0.5 larry larry.cow
```

Para alterar a senha de root digite o comando *passwd*, informe a senha de root e a confirme.

Agora configure algumas opções do sistema no arquivo */etc/rc.conf*

```
# nano -w /etc/rc.conf
```

Neste arquivo você pode alterar o layout do seu teclado no console para *br-abnt2*, seu editor padrão, seu *Display Manager* etc. Em seguida, vamos instalar as últimas ferramentas (*syslog reiser progs* e *cron*):

```
# emerge syslogd reiserfsprogs vixie-cron
```

```
# rc-update add syslogd default
# rc-update add vixie-cron default
```

Se você tem uma conexão ADSL via PPPoE (como o *Speedy*, da *Telefonica*), não esqueça disto:

```
# USE="-X" emerge rp-pppoe
```

### Instalando o gerenciador de boot

O gerenciador de boot é responsável por “carregar” o sistema operacional. Vamos ensinar a configurar o *Lilo*, embora também seja possível usar o *GRUB*. Digite:

```
# emerge lilo
```

Edite o arquivo de configuração:

```
# nano -w /etc/lilo.conf
```

E deixe-o como segue:

```
boot=/dev/hda # Instala na MBR
prompt # Menu de seleção do S.O.
timeout=50 # Aguarda 5 segundos
default=gentoo # Sistema padrão é gentoo
vga=791
# Para quem não utilizou o genkernel
image=/boot/kernel-2.6.XX-gentoo-rX
label=gentoo # Nome da sessão
read-only
root=/dev/hda3
# Para quem utilizou o genkernel
image=/boot/kernel-2.6.XX-gentoo-rX
label=gentoo
read-only
root=/dev/hda3
append="init=/linuxrc"
initrd=/boot/initrd-2.6.XX-gentoo-rX
```

Salve o arquivo e rode o comando:

```
# /sbin/lilo
```

Hora de reiniciar o sistema e ver se tudo correu bem. Teclê:

Tabela 3: arquivo */etc/fstab*

/dev/hda1	/boot	ext2	noauto,noatime	1 2
/dev/hda2	none	swap	sw	0 0
/dev/hda3	/	reiserfs	noatime	0 1
none	/proc	proc	defaults	0 0
none	/dev/shm	tmpfs	defaults	0 0
/dev/cdroms/cdrom0	/mnt/cdrom	auto	noauto,user	0 0

```
# exit
cdimage ~# cd
cdimage ~# umount /mnt/gentoo/boot
/mnt/gentoo/proc /mnt/gentoo
cdimage ~# reboot
```

Pode ser que o sistema não reinicie com algumas placas-mãe (a máquina travará quando descarregar os módulos USB). Nesse caso aperte o botão de reset que estará tudo bem.

Não se esqueça de tirar o CD do Gentoo do drive de CD-ROM. Se tudo correu bem, seu sistema deverá reiniciar sem problemas. Faça “login” como root e adicione um usuário:

```
# useradd joao -m -G users,wheel,audio,tty
-s /bin/bash
```

Finalmente o Gentoo estará rodando e instalado a partir do Stage 1. Para dominar os comandos básicos, `emerge` e `portage`, recomendamos a leitura dos manuais (`man emerge` e `man portage`)

## Instalando pacotes

Como exemplo, vamos instalar o Gnome. Primeiro vejamos as USE-Flags que o pacote disponibiliza:

```
# emerge -vp gnome
```

O parâmetro `-v` mostra as USE-Flags suportadas e `-p` mostra todos os pacotes fundamentais para o funcionamento do Gnome. Caso queira alguma USE-Flag, utilize a variável USE. Por exemplo: `USE="xml2" emerge gnome`. Nesse caso a variável USE colocará suporte a xml2. Caso queira tirar o suporte a xml2, `USE="-xml2" emerge gnome` e assim por diante.

O gerenciamento de pacotes do Gentoo Linux funciona da seguinte forma: visualize as USE-Flags suportadas, passe a variável USE se necessário e retire o parâmetro `-vp` para iniciar a instalação. Para procurar algum pacote dentro da árvore Portage utilize o comando `emerge -s <pacote>`. Por exemplo: `emerge -s apache`. O resultado serão todos os pacotes que contenham a palavra apa-



**Figura 01:** A comunidade GentooBR [5] está sempre disposta a ajudar os iniciantes.

**Tabela 4: Os principais parâmetros do *emerge***

Comando	Resultado
<code>emerge &lt;pacote&gt;</code>	Instala um determinado pacote.
<code>emerge -f &lt;pacote&gt;</code>	Faz o download do pacote. Para instalar basta remover o parâmetro “-f”.
<code>emerge -s &lt;palavra&gt;</code>	Faz uma busca por nome do pacote; suporta expressões regulares.
<code>emerge -S &lt;palavra&gt;</code>	Procura na descrição dos pacotes uma determinada palavra; um pouco demorado.
<code>emerge -p &lt;pacote&gt;</code>	Visualiza os pacotes que serão instalados.
<code>emerge -vp &lt;pacote&gt;</code>	Visualiza os pacotes que serão instalados e também as USE-Flags suportadas.
<code>emerge -u &lt;pacote&gt;</code>	Atualiza um pacote.
<code>emerge -uD world</code>	Atualiza toda a distribuição.
<code>emerge -O &lt;pacote&gt;</code>	Instala ignorando as dependências. Cuidado!
<code>emerge -C &lt;pacote&gt;</code>	Desinstala o pacote.
<code>emerge -i &lt;pacote&gt;</code>	Engana a portage fazendo-a pensar que o pacote está instalado. Muito útil quando instalamos um pacote ao modo antigo e queremos que a portage não fique nos avisando sobre as dependências desse pacote.

che no nome. Não se esqueça de ler os manuais do `emerge` e da `portage`. Para sanar dúvidas acesse o canal de IRC do GentooBR, [#gentoobr](#) no servidor [irc.free-node.net](#). Todos os participantes estarão à disposição para ajudá-lo. ■

## S.O.S

O CD-ROM que acompanha a Linux Magazine foi testado e, até onde pudemos constatar, se encontra livre de qualquer tipo de vírus ou conteúdo malicioso e de defeitos. Não nos responsabilizamos por qualquer perda de dados ou dano resultante do uso deste CD-ROM ou de software nele incluído. A Linux Magazine não oferece suporte técnico ao conteúdo do CD.

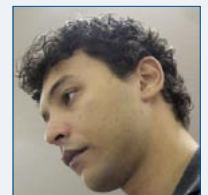
Este CD contém o *Stage 3* do *Gentoo Linux 2005.0* para a arquitetura x86, otimizado para os processadores Pentium III, x86 e Athlon.

## INFORMAÇÕES

- [1] Gentoo Install Handbook
- [2] Quick Installation Reference: [www.gentoo.org/doc/en/gentoo-x86-quickinstall.xml](http://www.gentoo.org/doc/en/gentoo-x86-quickinstall.xml)
- [3] Bugzilla do Gentoo: [bugs.gentoo.org](http://bugs.gentoo.org)
- [4] Fórum do Gentoo: [forums.gentoo.org](http://forums.gentoo.org)
- [5] Gentoo Linux Brasil: [www.gentoo.org](http://www.gentoo.org)
- [6] Parâmetros de otimização: [www.gentoo.org/modules.php?op=modload&name=Sections&file=index&req=viewarticle&artid=15&page=1](http://www.gentoo.org/modules.php?op=modload&name=Sections&file=index&req=viewarticle&artid=15&page=1)

## SOBRE OS AUTORES

*William Ferraz (Geek\_Slack) é Administrador de Redes e editor da comunidade GentooBR, além de tradutor da documentação*



*para o site oficial do projeto Gentoo Linux e da Gentoo Weekly Newsletter.*



*Marcelo Veriato Lima (marcelo\_) é Administrador de Redes e editor da comunidade GentooBR.*