

O slackware é a distribuição linux mais antiga ainda em atividade. Tendo sido criada por Patrick Volkerding em 1993, a partir da SLS.

Em todos esses anos, a distro conquistou ardorosos utilizadores, principalmente graças à sua filosofia de simplicidade e estabilidade.

Um produto de extrema qualidade para usuários com esta mesma característica. E este zine é de slacker para slacker



# slackware zine

Slackware is a registered trademark of Slackware Linux, Inc.

21 de Setembro de 2004 - Número 5

## Editorial

De novo atrasados. Isso está se tornando um péssimo hábito e pedimos desculpas a todos os nossos leitores. Agora que já fizemos um mea culpa, vamos à parte interessante...

Neste número do slackwarezine temos a estréia de mais um colaborador: o Clayton, com um artigo sobre a instalação e configuração do mplayer. Mesmo com o slackware já vindo com players de mídia, o mplayer possui grandes vantagens, e quem usa, aprova.

Dois artigos do Deives "thefallen" Michellis, um sobre ambientes chrooted (se você não sabe do que estou falando, vá ler o artigo) e outro com dicas sobre o mutt agradam tanto aos administradores preocupados com segurança como os usuários amantes do modo texto.

Um artigo sobre configuração de mouses, cobrindo desde o básico sobre os tipos de mouse até a instalação de múltiplos mouses e um sobre como usar o ndiswrapper do Yucatan Costa completam essa edição.

Espero que gostem dessa edição!!

Piter PUNK

PS> Continuem mandando os seus artigos! Afinal, esta revista é feita a partir da contribuição de todos. Os artigos devem ser enviados para: [editor@slackwarezine.com.br](mailto:editor@slackwarezine.com.br), mande o seu minicurrículo junto, para a

## Índice

<b>Instalando o Mplayer e o mplayer-plugin no slackware.....</b>	<b>02</b>
Clayton Eduardo dos Santos	
<b>Brincando com Ratos.....</b>	<b>04</b>
Piter PUNK	
<b>Colocando serviços em chroot.....</b>	<b>07</b>
Deives "thefallen" Michellis	
<b>Recuperando a senha de root.....</b>	<b>09</b>
Piter PUNK	
<b>Dicas para o Mutt.....</b>	<b>10</b>
Deives "thefallen" Michellis	
<b>Wireless sem suporte no Linux? Ndiswrapper nele!!!.....</b>	<b>12</b>
Yucatan Costa	

## Errata

No artigo sobre a instalação do kernel 2.6, o autor esqueceu de um detalhe, caso ajam mais de um diretório de módulos na máquina, é necessário indicar quais módulos devemos usar, isso é feito com a opção `-k versão_do_kernel` na linha de comando do `mkinitrd`. Exemplo:

```
# mkinitrd -c -k 2.6.8.1 -m jbd:ext3
```

Assim seria criado um initrd com os módulos necessários para ler partições ext3, pegando esses módulos do `/lib/modules/2.6.8.1`

Reprodução do material contido nesta revista é permitida desde que se incluam os créditos aos autores e a frase:

**"Reproduzida da Slackware Zine #5-  
[www.slackwarezine.com.br](http://www.slackwarezine.com.br)"**

com fonte igual ou maior à do corpo do texto e em local visível



slack  
users

# Instalando Mplayer e mplayer-plugin no slackware

Atualmente, no **slackware** 10, os players de vídeo padrão incluídos no sistema são o **xine** e o **totem**. No entanto, eu particularmente, não gosto muito desses players em função das limitações presentes em cada um deles, como a pequena quantidade de codecs suportados (ao menos na instalação default do **slackware**) e de alguns outros problemas que podem ser percebidos com o decorrer do tempo.

Um dos players mais bacanas para o linux, em minha opinião, é o **MPlayer**. A quantidade de codecs suportados é bastante interessante e o desenvolvimento do player está em constante evolução. Além disso, a integração entre o **MPlayer** e o browser **mozilla/plugin** é 100% funcional, funcionando muito bem com vídeos e áudio, inclusive streaming. Vamos ao trabalho... =]

## Instalando os codecs

Seguindo a ordem descrita na documentação do **Mplayer**, o primeiro codec é o **FFmpeg**, cujos fontes podem ser encontrados no site

<http://ffmpeg.sourceforge.net>. O processo de instalação é trivial:

```
tar -xvzf ffmpeg-0.4.9-pre1.tar.gz
./configure
make
checkinstall
```

### ATENÇÃO:

Todo mundo deve saber, mas não custa lembrar, a instalação de programas (`checkinstall`, `make install`, etc...) deve ser feita como `root`.

O próximo codec citado na documentação do **Mplayer** é responsável pela decodificação dos codecs **win32**, em especial arquivos codificados com codecs **WMV9/WMA9**.

O arquivo compactado com os codecs pode ser encontrado em:

<http://www.mplayerhq.hu/MPlayer/releases/codecs>, e o nome dele é `windows-all-20040809.zip`

Em seguida, descompacte esse arquivo para que possamos copiar o conteúdo do diretório (como usuário `root`) criado para o diretório `/usr/local/lib/codecs`.

Isto pode ser feito com a seguinte seqüência:

```
unzip windows-all-20040809.zip
su -
mkdir /usr/local/lib/codecs
cp /home/clayton/windows-all-20040809/* \
    /usr/local/lib/codecs
```

As `dlls` que dão suporte ao formato **Quick Time**, já estão presentes no pacote de `libs win32`, desse modo, o download dos pacotes avulsos referentes ao **Quick Time**

não deverá ser realizado. O mesmo ocorre com o suporte a vídeos no formato **DivX4/DivX5**, provido pela `lib FFmpeg`.

Os codecs do **XAnim**, podem ser baixados em:

<http://www.mplayerhq.hu/MPlayer/releases/codecs>, eo nome do arquivo é `xanimdlls-20040626.tar.bz2`

Em seguida,

```
tar -xvzf xanimdlls-20040626.tar.bz2
su -
cp /home/clayton/xanimdlls-20040626/* /
usr/local/lib/codecs
```

São necessários vários arquivos para dar suporte aos codecs do **RealVideo/RealAudio**. Todos estes arquivos podem ser baixados no mesmo site em que baixamos os do **XAnim**. Neste site devemos baixar os seguintes arquivos:

- `rp8codecs-20040626.tar.bz2`
- `rp8codecs-win32-20040626.tar.bz2`
- `rp9codecs-20040626.tar.bz2`
- `rp9codecs-win32-20040626.tar.bz2`

Descompacte/desempacote todos os codecs (`tar -xvzf nome_do_arquivo.tar.bz2`). E, em seguida, copie-os para o diretório em que estamos colocando os codecs:

```
su -
cd /usr/local/lib/codecs
cp /home/clayton/rp8codecs-20040626/* .
cp /home/clayton/rp8codecs-win32-20040626/* .
cp /home/clayton/rp9codecs-20040626/* .
cp /home/clayton/rp9codecs-win32-20040626/* .
```

## Baixando, compilando e instalando o Mplayer

A primeira coisa a fazer é baixar o arquivo `Mplayer-1.0pre5.tar.bz2` do endereço

<http://ftp5.mplayerhq.hu/mplayer/releases>

Tendo o arquivo em mãos:

```
tar -xvzf MPlayer-1.0pre5.tar.bz2
cd MPlayer-1.0pre5
./configure --enable-gui
make
checkinstall
```

Baixe uma fonte e uma skin para seu player na página oficial do **MPlayer**, eu sugiro os seguintes:

- `Skin: Blue-1.4.tar.bz2`
- `Fonte: font-arial-iso-8859-1.tar.bz2`

Utilize o bom e velho `tar -xvzf` para descompactá-los, conforme fizemos com os arquivos anteriores.

## Instalando Mplayer e mplayer-plugin no slackware

Após a descompactação, escolha o tamanho de fonte que mais lhe agrada, e como usuário root, copie o **CONTEÚDO** do diretório escolhido para o diretório /usr/local/share/mplayer/font/

Exemplo:

```
su -
cd /usr/local/share/mplayer/font
cp /home/clayton/font-arial-iso-8859-1/\
font-arial-14-iso-8859-1/* .
```

O mesmo acontece com a Skin utilizada na interface gráfica:

```
su -
cd /usr/local/share/mplayer/Skin
mkdir default
cd default
cp -r /home/clayton/Blue/* .
```

Nesse caso, a interface utilizada estará no diretório chamado default. Para facilitar a troca de skins, a utilização de links simbólicos é indicada. =]

Tudo pronto!!!

Agora é só chamar o mplayer via console:

```
mplayer nome_do_arquivo.avi
```

Ou ainda:

```
gmplayer
```

Em alguns casos, é possível que o mplayer mostre um erro dizendo que a saída default não é válida, para

corrigir o problema, basta clicar com o botão direito do mouse sobre a interface do gmplayer, escolher preferences/video e mudar a saída padrão para a opção xv (se a sua placa de vídeo suportar XVideo)

### Instalando o mplayer-plugin para mozilla/firefox

Baixe o arquivo em

<http://mplayerplugin.sourceforge.net/download.html>

Precisaremos também do gecko-sdk ou do mozilla-devel, como estou baseando essa instalação no slackware 10 (com Mozilla 1.7 sem mozilla-devel), precisei baixar o gecko-sdk em:

```
http://ftp.mozilla.org/pub/mozilla.org/mozilla/releases/mozilla1.7/gecko-sdk-i686-pc-linux-gnu-1.7.tar.gz
```

Após baixar o arquivo, execute os seguintes passos:

```
tar -xvzf gecko-sdk-i686-pc-linux-gnu-1.7.tar.gz
tar -xvzf mplayerplug-in-2.66.tar.gz
cd mplayerplug-in
./configure --with-gecko-sdk=./gecko-sdk
make
```

Pronto, agora é só copiar os arquivos mplayerplug-in.so e mplayerplug-in.xpt gerados na compilação, para os diretórios plugins e components de seu navegador preferido (mozilla ou firefox).

Reinicie o browser e...Bom divertimento!!!

Clayton Eduardo dos Santos  
<claytones@terra.com.br>

if you can read  
you can use

# slackware

# Brincando com Ratos

Este é um artigo simples, mostrando os tipos de mouses, protocolos e como configurá-los. Aproveitamos para mostrar como mapear botões e como configurar dois mouses ao mesmo tempo, o que é bem útil para usuários de notebooks.

## Tipos de Mouse

A primeira coisa a fazer é como se comunicar com o seu mouse, obviamente não estamos falando em usá-lo como microfone como o Scotty no Jornada nas Estrelas 4 ([slackwarezine](#) - to the **REAL** nerds), mas queremos saber qual o protocolo devemos usar.

Além do protocolo, os mouses ainda variam quanto à forma de conexão. Ou seja, os conectores dos mouses são diferentes fisicamente entre si. Nesse quesito, existem basicamente quatro tipos de mouse:

- **Mouses Seriais:** com um conector igual ao de joystick de atari, conhecido como DB9. Se você tem um mouse serial realmente antigo, pode ser que ele tenha um conector DB25 do tamanho de um conector de impressora!!!
- **Mouses PS/2:** com um conector redondo, conhecido como PS/2 ou miniDIN.
- **Mouses USB:** com um conector retangular na ponta, muito do sem graça.
- **Mouses de Bus:** Ah... a diversidade e o caos...

No mundo real, existem só os três primeiros tipos. Mouses de bus são raros, e mais raros ainda os que funcionam. Os mouses seriais caminham a passos largos para o desaparecimento, mas ainda existe uma base instalada razoável deles.

A nossa comunicação com algum hardware se dá através de arquivos localizados dentro do `/dev`. Assim, o `/dev/hda` é o seu disco rígido, o `/dev/rtc` é o chip de relógio e por aí vai... nessa linha, temos as entradas em que os mouses são ligados.

- **Seriais:** entradas seriais: `/dev/ttyS*`
- **PS/2:** entrada PS2: `/dev/psaux`
- **USB:** entrada USB: `/dev/input/mice`

No `/dev/ttyS*` o `*` pode ser 0, 1, 2 ou 3. Normalmente é o 0. Existe um link no `/dev`, chamado `/dev/mouse` que aponta qual o dispositivo está sendo utilizado para a sua comunicação com o mouse. É possível saber essa configuração com um simples `ls`:

```
punk@rachael:~$ ls -l /dev/mouse
lrwxrwxrwx 1 root root 10 \
2004-07-30 02:03 /dev/mouse -> input/mice
```

Ou seja, o meu mouse padrão é um mouse USB. Nessa máquina tenho conectado também um mouse PS2 (é, só para escrever este artigo), mas o principal é o USB. Para mudar o link de um mouse para outro, é possível fazer

isso "na mão" com o comando `ln` ou utilizar um utilitário como o `mouseconfig`, o que acaba facilitando bastante a vida de quem está começando.

### ATENÇÃO:

O `mouseconfig` **não** configura o mouse no X. Repetindo, o `mouseconfig` **NÃO** configura o mouse no X. Espalhem essa informação para seus amigos e parentes.

Agora podemos pensar no segundo passo, o protocolo. Por incrível que pareça, dois mouses com o mesmo conector podem não falar a mesma língua. Principalmente se o seu mouse for um mouse de "rodinha". Ah! Para ajudar, os nomes dos protocolos são diferentes no X e no `gpm` (console).

### Seriais:

- **gpm:** bare, ms, pnp e ms3\*
- **X:** Microsoft, IntelliMouse\*

### PS/2:

- **gpm:** ps2, imps2\*, exps2\*, netmouse
- **X:** PS/2, IMPS/2\*, ExplorerPS/2\*, NetScrollPS/2\*, NetMousePS/2

### USB:

Praticamente os mesmos do PS/2. Nunca vi um que não seja IMPS/2 no X e imps2 no `gpm`, mas devem existir.

Todos os protocolos que eu listei com um `*` do lado são protocolos para mouses com "rodinha". Apenas se lembre-que o `gpm` ignora solenemente a rodinha do mouse, apesar de suportar o protocolo utilizado por elas.

Os protocolos mais comuns são os primeiros das listas, ficando para o fim os protocolos mais exóticos. Existem uma série de outros protocolos, mas coloquei na lista apenas os mais "encontráveis", se estiver no X e não souber o protocolo do seu mouse, utilize o protocolo "Auto".

## Configuração do GPM

Sem maiores segredos, praticamente é só rodar o `mouseconfig` e fazer:

```
/etc/rc.d/rc.gpm restart
```

Para quem quiser fazer "na mão" a configuração, vale a pena uma olhadinha no próprio `/etc/rc.d/rc.gpm`. Costuma ser suficiente, para os preguiçosos, a sintaxe de uma linha de comando para chamar o `gpm` seria algo assim:

```
gpm -t protocolo -m dispositivo
```

No caso de um mouse USB por exemplo:

```
gpm -t imps2 -m /dev/input/mice
```

## Brincando com Ratos

Existem várias outras opções, mas a maior parte delas não costuma ser utilizada. Se quiser conhecer quais as outras opções do `gpm`, lembre de acessar a página de manual desse comando. Ela é bem completa e explicativa.

### Configuração do X

Aqui a coisa é mais interessante, principalmente porque neste caso a temível "rodinha" do mouse está presente. Para quem não sabe, o padrão PS2 original previa apenas 3 botões. Nada de rodinhas e de 5, 6 ou 7 botões em um mouse. E isso causou um verdadeiro samba do crioulo doido, já que cada fabricante fez um "puxadinho" no protocolo para suportar a rodinha e os botões extras.

O mais comum desses "puxadinhos" é o IMPS/2, o segundo mais comum é o ExplorerPS/2. Geralmente os mouses com muitos botões obedecem ao segundo protocolo, mas isso não é uma regra.

Uma entrada de mouse dentro do `/etc/X11/xorg.conf` se parece com isso:

```
Section "InputDevice"
    Identifier "Mouse0"
    Driver "mouse"
    Option "Device" "/dev/psaux"
    Option "Protocol" "IMPS/2"
    Option "Buttons" "5"
    Option "ZAxisMapping" "4 5"
EndSection
```

Cada uma das linhas aí dentro tem uma utilidade:

- **Identifier:** O nome pelo qual esse mouse será identificado
- **Driver:** O que é esse dispositivo, no caso de mouses, sempre será "mouse"
- **Device:** O dispositivo utilizado, nesse caso, é um mouse conectado à porta PS/2
- **Protocolo:** O protocolo do mouse
- **Buttons:** Quantidade de botões
- **ZaxisMapping:** Quais botões são utilizados para a rodinha.

#### DICA:

Se o seu mouse aparece no X e vai correndo para o canto da tela, sozinho, isso significa que o protocolo está errado, edite o `xorg.conf` e troque-o para "Microsoft"

Deu para ver que essa sintaxe é referente a um mouse que possui a rodinha. Ele é um mouse com três botões, o da esquerda, a rodinha e o da direita. Se fosse um mouse com dois botões apenas, é interessante colocar a opção para emular três botões:

```
Option "Emulate3Buttons" "On"
```

Com isso, é possível simular um terceiro botão apertando simultaneamente os botões da direita e da esquerda. Muito útil, já que a função de "colar" no X é feita através do botão do meio (sim, o terceiro botão é o do meio)

Voltando ao exemplo, mesmo o mouse tendo 3 botões, nós colocamos "Buttons" "5", para avisar que existem dois botões "virtuais", o movimento da rodinha para cima e o movimento da rodinha para baixo.

Se o seu mouse possuir 5 botões: o esquerdo, a rodinha, o direito e mais dois laterais (por exemplo), você deve colocar "Buttons" "7" se quiser usar os dois botões laterais e mapear a rodinha para os botões 6 e 7. Esta é a configuração que uso para um Dr.Hank nessas condições:

```
Section "InputDevice"
    Identifier "Mouse1"
    Driver "mouse"
    Option "Device" "/dev/psaux"
    Option "Protocol" "ExplorerPS/2"
    Option "Buttons" "7"
    Option "ZAxisMapping" "6 7"
EndSection
```

Vemos aí o protocolo diferente (ExplorerPS/2) e a mudança na quantidade de botões e no mapeamento deles. Mas surge agora um novo problema, muitos gerenciadores de janelas e programas por aí só mapeiam o movimento da rodinha nos botões 5 e 6, o q fazer?

Uma solução seria simplesmente ignorar os últimos botões e colocar "Buttons" "5" e "ZAxisMapping" "4 5". Mas, e se quisermos usar todos os botões?

Simples, vamos "trocar" os botões do mouse de lugar! Vamos primeiro imprimir como esses botões estão mapeados:

```
punk@rachael:~$ xmodmap -pp
There are 7 pointer buttons defined.
```

Physical Button	Button Code
1	1
2	2
3	3
4	4
5	5
6	6
7	7

Esse monte de coisas estranhas está dizendo: Olha, o botão 1 do mouse manda o código do botão 1, o botão 2 manda o do 2 e por aí vai... Vamos mudar essa tabela para que ele diga algo assim: O botão 1 manda o código 1, o 2 manda o 2, mas o 4 manda o código 6, o 5 manda o 7 e vice-versa. Confuso? Não se preocupe, isso piora.

```
punk@rachael:~$ xmodmap -e \
    "pointer = 1 2 3 6 7 4 5"
```

Pronto! Com isso trocamos os botões de lugar, perceba que ao invés do típico: 1 2 3 4 5 6 7, invertemos as posições do 6 com o 4 e do 5 com o 7. Essa nova disposição faz com que quando usemos a rodinha (botões 6 e 7) o X entenda que estamos usando os botões 4 e 5. E todos os programas que usam a rodinha do mouse funcionam felizes :-)

## Brincando com Ratos

### DICA:

Para inverter o mouse de um canhoto é só fazer a mesma coisa, ao invés da ordem "normal" 1 2 3, coloque 3 2 1.

```
xmodmap -e "pointer = 3 2 1"
```

Se mesmo assim a rodinha não funcionou, você pode trocar o protocolo que está no `xorg.conf` por outro (ou até mesmo pelo "Auto" para poder descobrir que raios de mouse é esse seu). Ah! Se a rodinha funcionou (e você precisou do `xmodmap`) crie um arquivo `.Xmodmap` na sua área e coloque dentro dele o comando que você enviou para o `xmodmap`. O meu está assim:

```
punk@rachael:~$ more .Xmodmap
keycode 113 = Mode_switch
keycode 47 = semicolon colon ccedilla Ccedilla
pointer = 1 2 3 6 7 4 5
```

Esse arquivo será lido sempre que você entrar no X. As configurações aqui foram testadas em mouses da A4Tech (USB), MTek (USB e PS2), DrHank (PS2) e Compaq (PS2) e Troni (PS2). Todos, menos o da Compaq, com rodinha.

O da Troni, mesmo tendo 5 botões e insistindo que é ExplorerPS2 não aceita a configuração que indiquei, dois botões dele se comportam sempre como sendo o botão 2 (botão do meio). Então preferi deixar configurado com 5 botões mesmo.

### Dois mouses

Quem tem um notebook sabe o quanto aquela superfície que faz as vezes de mouse enche o saco. Quem tem o mouse estilo "clitório" entre o J e o K padece mais ainda. É muito comum a pessoa ter um segundo mouse, externo, para passar um pouco menos de raiva.

O que nem todo mundo sabe, é que dá para configurar dois mouses de modo simultâneo. Assim, você pode continuar usando o mouse do seu notebook e um mouse externo sem precisar ficar editando arquivos de configuração toda vez que troca de mouse.

Antes de começar, os dois mouses devem acessar dispositivos diferentes, um pode ser USB e outro PS/2, um Serial e um USB, um Serial e um PS/2 ou, até mesmo, dois mouses USB.

A maior parte dos mouses embutidos em notebooks são PS2, então irei considerar que o mouse "padrão" é um PS2 e o externo é um USB. Caso este não seja o seu caso, basta usar um pouco a cabeça e adaptar as configurações.

```
Section "InputDevice"
    Identifier "MousePadrao"
    Driver "mouse"
    Option "Device" "/dev/psaux"
    Option "Protocol" "PS/2"
EndSection
```

```
Section "InputDevice"
    Identifier "MouseExterno"
    Driver "mouse"
    Option "Device" "/dev/input/mice"
    Option "Protocol" "IMPS/2"
    Option "Buttons" "5"
    Option "ZAxisMapping" "4 5"
    Option "Resolution" "256"
EndSection
```

Estas são as entradas para os dois mouses. Nas entradas em si, não existe nada que indique qual mouse é qual e o que eles estão fazendo, isso é feito por outra seção no `xorg.conf`:

```
Section "ServerLayout"
    Identifier "Simple Layout"
    Screen "Screen 1"
    InputDevice "MousePadrao" "CorePointer"
    InputDevice "Keyboard1" "CoreKeyboard"
    InputDevice "MouseExterno" "AlwaysCore"
EndSection
```

Pronto! As duas linhas que falam dos mouses fazem com que os dois sejam "obedecidos". O `MousePadrao` é o "CorePointer" e o segundo Mouse envia dados para ele. Agora você pode inclusive fazer uma "guerra" de mouses com cada um puxando o mouse para um lado.

No segundo mouse (`MouseExterno`) existe uma opção "Resolution", ela informa qual a resolução do mouse. Quanto maior a resolução, mais preciso o seu mouse. Mexer nesse parâmetro altera a velocidade do mouse na tela, se você está irritado com isso, esse é o bom lugar para mudar.

### Conclusões

São algumas configurações simples para o mouse, mas que fazem toda a diferença. Ainda existe muito o que falar sobre esse assunto, inclusive pela quantidade de modelos e tipos de mouses que existem no mercado. Mas, este artigo, já dá uma boa "introdução" ao assunto.

Ah! Eu sei que o plural de mouse é "mice", mas no documento preferi utilizar mouses por que eu quis e pronto.

Piter PUNK <piterpk@terra.com.br>

**slackware**  
read the docs  
and be happy.

# Colocando serviços em Chroot

## 1. Introdução

Aqui vão algumas dicas de como colocar alguns serviços em chroot, para aumentar a segurança do sistema.

## 2. Um pouquinho de Teoria

Dentro de um sistema Unix ou suas variantes (como o nosso amado **slackware** Linux :D), é possível colocar um processo dentro de um ambiente virtual de diretórios, fazendo com que ele enxergue um diretório como se fosse o diretório raiz (o diretório / ). Repare que é daí que vem o nome "chroot" de "change root" (mudar o (diretório) raiz).

Dessa maneira, o processo e todos os subprocessos que se originem dele ficam efetivamente presos dentro desse diretório, sem poder acessar o ambiente externo de sistema de arquivos.

Veja agora as "letrinhas miúdas" do contrato; o processo fica completamente incapaz de acessar o ambiente de sistema de arquivos. Isso vale pra TUDO. O programa será incapaz de ler bibliotecas ou arquivos de configuração que ficaram "do lado de fora"; também não conseguirá acessar os unix sockets do syslog (que fica no /dev/log normalmente). Tudo que você tem disponível são a rede e o que esta no seu diretório de chroot.

Assim, para "chrootar" um programa precisamos copiar todas as bibliotecas que ele utiliza lá pra dentro, para o diretório /LUGAR/DO/CHROOT/lib/, os arquivos de configuração para /LUGAR/DO/CHROOT/etc/, e assim por diante, criando uma imagem de sistema do que queremos que o programa enxergue dentro do chroot. Para ver a lista de bibliotecas necessárias, rode o comando "ldd" no executável. Ex: ldd /var/chroot/apache/bin/httpd

Há algumas exceções a essa regra. Alguns programas que já possuem recursos de chroot internamente dispensam a cópia de bibliotecas para dentro do chroot (como o BIND, por exemplo). Isso acontece por que lá dentro de seu código há a instrução de chroot depois que o programa já esta rodando, e obviamente já carregou todas as bibliotecas necessárias antes de rodar :)

Uma outra exceção seria termos programas compilados estaticamente (usando-se o comando "export LDFLAGS=-wl,-static" antes de rodar o ./configure, por exemplo). Programas compilados estaticamente já possuem todas as chamadas de biblioteca necessárias dentro do seu código executável, dispensando o carregamento de bibliotecas.

O ponto mais forte do chroot é que dentro do ambiente virtual não há shells disponíveis nem comandos a serem executados. Assim, na eventualidade de alguém ser bem sucedido em explorar alguma vulnerabilidade do serviço, ele não terá nenhum programa para rodar seus comandos (o serviço do shell), o que dificultará consideravelmente a ação do tal sujeito.

Outra consideração importante é usar um esquema de descarte de privilégios junto com o chroot. Fazer chroot de um serviço rodando com os poderes do root ainda deixa abertas opções de se voltar ao diretório raiz original do sistema, furando todo seu trabalho de aumento de segurança :)

Portanto, use chroot junto com descarte de privilégios (drop priv, setuid para um usuário desprivilegiado ou algo similar) para fazer valer o seu trabalho de aumento de segurança :)

## 3. Chrootando os serviços

Mãos a obra! Aqui vão algumas receitas de bolo de como colocar serviços em chroot.

Nos exemplos a seguir, uso a nomenclatura /var/chroot/SERVICO como diretório padrão de chroot.

apache+PHP representa o jeito mais difícil de fazer o chroot, sendo necessário criar um ambiente mínimo dentro do diretório de chroot. Os outros exemplos já serão mais tranquilos de se fazer.

### 3.1. Apache 2

Esse howto foi escrito para apache 2 (2.0.50). Com algumas pequenas modificações, deve rodar sem problemas com o apache 1.3.x. Veja as flags disponíveis no ./configure do apache 1.3.

#### 3.1.1. Compilando

Vamos compilar o apache 2 "esticamente", ou seja, todas as chamadas a bibliotecas serão colocadas DENTRO do programa ao invés de apenas deixar uma referência a chamada (por exemplo: chame a função XYZ na biblioteca libUVW --> copie a função XYZ da biblioteca libUVW para dentro do meu binário)

```
LDFLAGS="-Wl,-static -static /usr/lib/libcrypto.a" \
./configure --prefix=/usr --sysconfdir=/etc \
--enable-proxy --enable-ssl \
--enable-static-support --enable-info \
--enable-suexec --enable-cgi \
--enable-rewrite --enable-so && make
```

A inclusão do "/usr/lib/libcrypto.a" é para evitar conflitos com a libcrypto na hora da linkagem estática (ele fica reclamando de re-definição da função "crypt"). Verifique se a libcrypto se encontra de fato no /usr/lib/libcrypto.a ou se está em outro diretório.

#### 3.1.2. Instalando

Se tudo correr bem na compilação, instale o Apache com o seguinte comando:

```
LD_LIBRARY_PATH=/var/chroot/apache/usr/lib/ \
make INSTALL_ROOT=/var/chroot/apache/ \
DESTDIR=/var/chroot/apache install
```

## Colocando Serviços em chroot

Isso já vai jogar o apache dentro do diretório de chroot.

Para que ele consiga rodar, precisamos ainda copiar algumas bibliotecas "de apoio" (com as bibliotecas do NameServiceSwitch ou NSS e alguns devices).

Vamos precisar (inicialmente) da libc, o ld, e as libs do nss. Quando for copiar as bibliotecas, use o comando "cp -a" para manter o tipo de arquivo (se for um link, ele ira copiar como link, e não copiar o arquivo linkado).

```
root@Ragnarok:/var/chroot/apache/lib# ls
ld-2.2.5.so*          libnss_compat-2.2.5.so*
libnss_dns-2.2.5.so* libnss_hesiod-2.2.5.so*
libnss_nis-2.2.5.so* ld-linux.so.2@
libnss_compat.so.2@  libnss_dns.so.2@
libnss_hesiod.so.2@ libnss_nis.so.2@
libc-2.2.5.so*       libnss_db-2.2.so*
libnss_files-2.2.5.so* libnss_ldap-2.2.5.so*
libnss_nisplus-2.2.5.so* libc.so.6@
libnss_db.so.2@      libnss_files.so.2@
libnss_ldap.so.2@    libnss_nisplus.so.2@
```

Ah, copie os arquivos /etc/group, /etc/host.conf, /etc/hosts, /etc/localtime, /etc/nsswitch.conf, /etc/passwd e /etc/resolv.conf pra dentro do /var/chroot/apache/etc/ (senão o apache não vai nem saber qual o UID do nobody :D). Esses podem (e devem) ser copiados SEM o -a (o localtime normalmente é um link pra /usr/share/algumacoisa... não queremos o link, e sim o arquivo em si)

Agora, vamos criar o mínimo de devices necessários. Basicamente o /dev/null e o /dev/zero hehehe

```
root@Ragnarok:/var/chroot/apache# mkdir dev
root@Ragnarok:/var/chroot/apache# cp -a \
/dev/{null,zero} dev
```

Para facilitar a instalação de outros modulos no Apache (como o módulo do PHP ou outro qualquer que use o apxs), vamos fazer algumas alterações no utilitario do Apache "apxs".

Edite as seguintes linhas no seu

```
/var/chroot/apache/usr/bin/apxs:
```

```
my $installbuilddir = "/var/chroot/apache/usr/build";
get_config_vars("$installbuilddir/config_vars.mk", \%
config_vars);
```

```
# read the configuration variables once
my $prefix          = "/var/chroot/apache/";
```

E adicione as seguintes linhas logo após a seqüência abaixo:

```
my $ltflags          = $ENV{'LTFLAGS'};
$ltflags or $ltflags = "--silent";
```

```
$config_vars{'sbindir'}=$prefix."/usr/bin/";
$config_vars{'APU_BINDIR'}=$prefix."/usr/bin/";
$config_vars{'APR_BINDIR'}=$prefix."/usr/bin/";
```

```
# activate module via LoadModule/AddModule
directive
```

```
# Adicione a linha abaixo também...
```

```
$CFG_SYSCONFDIR="/var/chroot/apache/etc/";
```

```
if ($opt_a or $opt_A) {
if (not -f "$CFG_SYSCONFDIR/$CFG_TARGET.conf") {
    error("Config file
$CFG_SYSCONFDIR/$CFG_TARGET.conf not found");
exit(1);
```

Feito isso, rode o seguinte comando para acertar o diretório de "build" do apxs:

```
ln -si /var/chroot/apache/usr/build/ \
/var/chroot/apache/usr/bin/
```

Edite as configurações do Apache se quiser e rode-o com o seguinte script de rc:

```
#!/bin/sh
# Script rc.apache chrootado by Deives Michellis
case "$1" in
test)
    chroot /var/chroot/apache/ /usr/bin/httpd -t
;;
start)
    chroot /var/chroot/apache/ \
        /usr/bin/httpd -k start
;;
stop)
    chroot /var/chroot/apache/ \
        /usr/bin/httpd -k stop
;;
restart)
    chroot /var/chroot/apache/ \
        /usr/bin/httpd -k restart
;;
*)
    echo "$0 (start|stop|restart|test)"
;;
esac
```

### 3.2. PHP 4.3.x

Para instalarmos o PHP como módulo do apache faremos algo parecido com o que está acima:

```
make clean
CFLAGS=-I/var/chroot/apache/usr/include/ \
LD_LIBRARY_PATH=/var/chroot/apache/usr/lib/ \
./configure \
--with-apsx2=/var/chroot/apache/usr/bin/apxs \
--with-mysql --prefix=/usr \
--sysconfdir=/etc/php && make
make install INSTALL_ROOT=/var/chroot/apache/
```

O "make install" já deverá colocar o modulo do PHP dentro do diretório /var/chroot/apache/. Basta editar o httpd.conf para chamar o PHP e boa.

### 3.3. BIND 9 (named)

É bastante simples colocarmos o BIND 9 (named) em chroot. Ele já provê mecanismos de descarte de privilegios e uso de chroot, o que facilita bastante nosso trabalho.

O diretório de chroot usado será /var/chroot/named. Dentro desse diretório vamos precisar dos diretórios etc, dev e var/named/. Crie-os da seguinte forma:



## Colocando Serviços em chroot

```
root@KlingonRealm:~# mkdir -p \  
  /var/chroot/named/etc \  
  /var/chroot/named/dev \  
  /var/chroot/named/var/named/  
root@KlingonRealm:~# cp -a /dev/null \  
  /dev/random /dev/urandom \  
  /var/chroot/named/dev/  
root@KlingonRealm:~# cp /etc/localtime \  
  /var/chroot/named/etc/  
root@KlingonRealm:~# useradd -g daemon -d \  
  /dev/null -s /bin/false named  
root@KlingonRealm:~# chown -R named:daemon \  
  /var/chroot/named/var/named/
```

O diretório de chroot já esta pronto pra receber o named. Basta iniciá-lo com o comando:

```
root@KlingonRealm:~# named -u named \  
  -t /var/chroot/named
```

Caso algo aconteça e o bind não funcione, rode-o com a flag -g. Isso fara com que ele exiba informações de debug (como arquivos que não conseguiu abrir ou permissões negadas), facilitando seu trabalho :)

### 3.4. ProFTPD

O ProFTPD é ridiculamente simples de colocar o usuário em chroot. Basta adicionar a seguinte linha no seu proftpd.conf:

```
DefaultRoot ~
```

Só mandar um killall -HUP proftpd e pronto :)

Deives Michellis <thefallen@Unitednerds.org>

## Autores:

**Clayton Eduardo dos Santos**, Trabalha com Linux a cerca de um ano e meio e com Slackware a cerca de um ano. É matemático, mestre em Engenharia Elétrica pela USP de São Carlos e atualmente desenvolve seu projeto de pesquisa de Doutorado no Departamento de Engenharia Elétrica na USP de São Carlos , utilizando ferramentas 100% baseadas em software livre.

**Deives Michellis "thefallen"** é Tecnólogo em Processamento de Dados pela FATEC e Gerente de Desenvolvimento de Soluções Linux do Grup o GEO. Também nerd de carteirinha e ativista linux nas horas vagas.

**Piter PUNK**, é usuário de Linux desde 1996, e trabalhou durante vários anos como administrador de sistemas. É o principal desenvolvedor do slackpkg, ferramenta para atualização de pacotes no Slackware. Escrever documentação e brincar com computadores MSX são dois de seus hobbies.

**Yucatan Costa "Kenjiro"**, Bacharel em Ciência da Computação e Pós-Graduando (Especialização) em Programação Avançada e Redes. Trabalha como Administrador de Redes da Escola Técnica Alto Jacuí. Árduo defensor do Slackware Linux, botou as mãos em um computador pela primeira vez em 1985 (um CP-500), mas só teve contato com Linux em 1996 (Slackware Linux 3.0)

# Recuperando a Senha de ROOT

Ok, todo mundo sabe que senha de root é mais importante que senha de banco, e que não se deve perdê-la sob hipótese nenhuma. Mas sempre existe algum amigo (claro, nunca você) que por um acaso esqueceu a senha de root.

Normalmente é bem simples de recuperar. Reboote o seu sistema, no prompt do LILO digite:

```
LILO: Linux init=/bin/sh rw
```

Isso irá iniciar o sistema sem carregar o init. O init é o responsável por gerenciar a inicialização, é ele que põe os prompts de login e senha nos terminais virtuais tanto quando se inicia no modo multiusuário (init 3) quanto no modo monousuário (init 1).

Trocando o init pelo /bin/sh, você irá cair direto na shell do sistema. Agora é só trocar a senha, use o comando passwd e coloque a sua nova senha de root (ops, sua não, do seu amigo).

Claro, existe a chance de você ter colocado senha no

LILO e de ter esquecido essa senha também. Nesse caso, o melhor a fazer é bootar com um disco de boot, serve o próprio CD de instalação do **slackware**.

No prompt do SYSLINUX (aquele escrito boot:), digite o seguinte:

```
boot:bare.i root=/dev/hdXn init=/bin/sh noinitrd rw
```

É praticamente a mesma coisa do comando anterior, só estamos pedindo para carregar o kernel bare.i (o kernel default. Se você utilizar um kernel especial, esta é a hora de decidir: adaptec.s, raid.s são alguns dos mais usados). Em seguida indicamos qual será a partição /, o x deve ser substituído por uma letra e o n por um número. O init=/bin/sh já explicamos para o quê serve e o noinitrd faz com que o sistema não carregue a ramdisk do CD.

Agora, se além de tudo isso seu amigo perdeu também a senha da BIOS e não pode bootar pelo CD, recomendo resetar a BIOS e comprar uma dose considerável de Memoriol

Piter PUNK <piterpk@terra.com.br>

# Dicas para o Mutt

## Introdução

O Mutt é um cliente de email em modo texto, com uma quantidade de recursos considerável, inclusive com facilidade para expandir o leque de opções e funcionalidades. Segundo o próprio autor, "All mail clients suck. This one just sucks less." :)

A seguir vamos ver como colocar o GnuPG pra funcionar com ele e também como acessar um catálogo de endereços externo, como um LDAP por exemplo.

## 1. Configurando o GnuPG no Mutt

Aqui vão umas dicas rápidas pra colocar o GnuPG pra trabalhar com o Mutt.

Para quem não sabe, GnuPG (GNU Privacy Guard) é uma implementação aberta dos protocolos do PGP, e permite enviar emails com assinaturas digitais baseadas em certificados ou mesmo emails com criptografia forte.

Esse howto sub-entende que você já possui o gnupg rodando em seu computador, que já gerou as chaves e já enviou para o keyserver, só falta integrá-lo ao mutt.

### 1.1. Configurando o Mutt

É uma boa idéia separarmos as diferentes configurações do Mutt em arquivos distintos, para facilitar a leitura ou correção dessas configurações depois.

No final do seu ~/.muttrc adicione a linha

```
source ~/.mutt-gnupgrc
```

Iremos criar um arquivo ~/.mutt-gnupgrc com o conteúdo do **Quadro 1**. Lembrando que as linhas com \ no final foram quebradas apenas para facilitar a diagramação, devendo ser digitadas em uma linha só

Altere a Key ID para a de sua chave. Minha chave possui o Key ID 0x73AB3459.

Para obter esse ID basta rodar o comando

```
thefallen@KlingonRealm:~$ gpg \  
--list-secret-keys \  
/home/thefallen/.gnupg/secring.gpg  
  
sec 1024D/73AB3459 2003-08-26 Deives Michellis  
(Personal Key)<dmichellis@yahoo.com>
```

### 1.2. Usando

Quando for enviar um email, vai aparecer a opção do PGP na tela de envio.

```
PGP: Sign  
sign as: 0x73AB3459
```

Você pode alterar essas opções apertando a tecla "p"

(e)ncrypt, (s)ign, sign (a)s, (b)oth, or (f)orget it?

Caso você erre a Passphrase do seu certificado, você pode fazer o mutt tirá-la do cache de passphrase apertando Ctrl+F.

Ah, e a função "encrypt" criptografa o email com a chave pública da pessoa para a qual você esta mandando email. O Mutt exibe uma lista de chaves para você escolher a correta para enviar o email.

### Quadro 1 - arquivo ~/.mutt-gnupgrc

```
set pgp_decode_command="gpg %?p?--passphrase-fd 0? --no-verbose --batch --output - %f"  
set pgp_verify_command="gpg --no-verbose --batch --output - --verify %s %f"  
set pgp_decrypt_command="gpg --passphrase-fd 0 --no-verbose --batch --output - %f"  
set pgp_sign_command="gpg --no-verbose --batch --output - --passphrase-fd 0 --armor \  
--detach-sign --textmode %?a?-u %a? %f"  
set pgp_clearsign_command="gpg --no-verbose --batch --output - --passphrase-fd 0 --armor \  
--textmode --clearsign %?a?-u %a? %f"  
set pgp_encrypt_only_command="pgpwrap gpg --batch --quiet --no-verbose --output - \  
--encrypt --textmode --armor --always-trust --encrypt-to 0x73AB3459 -- -r %r -- %f"  
set pgp_encrypt_sign_command="pgpwrap gpg --passphrase-fd 0 --batch --quiet --no-verbose \  
--textmode --output - --encrypt --sign %?a?-u %a? --armor --always-trust \  
--encrypt-to 0x73AB3459 -- -r %r -- %f"  
set pgp_import_command="gpg --no-verbose --import -v %f"  
set pgp_export_command="gpg --no-verbose --export --armor %r"  
set pgp_verify_key_command="gpg --no-verbose --batch --fingerprint --check-sigs %r"  
set pgp_list_pubring_command="gpg --no-verbose --batch --with-colons --list-keys %r"  
set pgp_list_secring_command="gpg --no-verbose --batch --with-colons --list-secret-keys %r"  
set pgp_autosign=yes  
set pgp_sign_as=0x73AB3459  
set pgp_replyencrypt=yes  
set pgp_timeout=1800  
set pgp_good_sign="^gpg: Good signature from"
```

## Dicas para o Mutt

Na hora de receber mensagens, alguns emails vem no formato de PGP clássico, ou seja, não há informações de header indicando ao Mutt que ele deve passar a mensagem pelo gpg (aparece "BEGIN PGP MESSAGE" ou "BEGIN PGP SIGNED MESSAGE" sem a mensagem de que o gpg verificou a assinatura). Basta usar a seqüência de teclas "<ESC> P".

### 2. Usando um catalogo de endereços externo no Mutt

Agora vão algumas dicas e scripts rápidos pra acessar um catálogo de endereços externo ao Mutt, mesmo um catálogo LDAP.

No meu caso específico, eu estou acessando 2 LDAPs (um Linux e um Exchange) para Catálogo de Endereços e o meu arquivo local de Alias do mutt.

Se você tiver um Thunderbird, Mozilla, Netscape, Evolution, etc etc etc, basta montar os shell scripts para acessá-los.

#### 2.1. Um pouco de Teoria

Entre os muitos, talvez infundáveis, recursos do Mutt, encontra-se a possibilidade de rodar um comando de "query" (pressionando-se ^t na hora de digitar os endereços). Esta variável maravilhosa chama-se "query\_command". Quando se digita Ctrl-T (^t), o Mutt chama esse comando passando como parâmetro o nome parcial já digitado, e espera uma saída no formato "EMAIL <TAB> NOME COMPLETO <TAB> O QUE MAIS VOCE QUISER"

No caso do script utilizado, ele manda como 3o. atributo a "Origem" da informação, para facilitar na hora de escolher qual endereço utilizar.

#### 2.2. Configurando o muttrc

Vamos primeiro adicionar a configuração do query\_command. Edite seu ~/.muttrc e adicione as linhas:

```
# chama o comando de query
set query_command = "~/bin/muttldap %s"
# já altera o comportamento da tecla
bind editor <tab> complete-query
# TAB pra chamar direto o query_command;
# Pode causar problemas com outros
# auto-completar (de nomes de arquivos,
# por exemplo)
```

Se quiser aproveitar que já esta "com a mão na massa", edite também a formatação de "forward" do mutt (a padrão é [ fulano@dominio.com.br: assunto ] no Assunto "forwardado" da mensagem) e também uma caixa de "Enviados"...

```
# Muda a formatação do forward
set forward_format = "FW: %s"
# guarda uma cópia das mensagens enviadas no
# diretório ~/Mail/Sent/ - formato Maildir;
# pra mailbox, basta remover o ultimo /
set record = ~/Mail/Sent/
```

```
# o mutt via perguntar se quer forwardar a
# mensagem INTEIRA (inclusive headers) como
# anexo;
set mime_forward = ask-no
```

Bom, feito isso, o Mutt já ta pronto pra rodar nosso comandinho... Vamos agora passar ao Script

### 2.3. Script de Lookup em LDAP

O script encontra-se disponível em <http://www.unitednerds.org/projects/mail/> com o nome muttldap.bz2. Trata-se de um shell script. Você deve descompactá-lo em algum caminho de execução (como /usr/bin, /usr/local/bin, ou mesmo ~/bin) e lhe dar permissões de execução (Chmod 755 muttldap, por exemplo).

Edite agora o arquivo de configuração ~/.mutt-sources (conforme solicitado pelo script). Um exemplo segue:

```
# Informa que a fonte de dados é um LDAP
LDAP
# O Host
Host: 10.10.0.2
# A BaseDN do LDAP
BaseDN: o=geobrasil, c=br
# SourceName é o nome que aparece no catalogo
SourceName: GEOMail
# Caso seja necessário fazer login no LDAP
#BindPW:
#BindDN:
# Linha em branco para encerrar a fonte de dados

Ldap
host: 10.1.0.5
basedn:
sourcename: Exchange
binddn: cn=FULANO, dc=LALAS, dc=com, dc=br
bindpw: NONONONONO

ALIAS
aliasfile: ~/.mutt-aliasrc
sourcename: Local Alias

# Deixe SEMPRE uma linha em branco no fim do arquivo
```

Agora faça o teste com o comando : muttldap ALGUMA COISA

#### 2.4. Dentro do Mutt

Tudo Ok, agora o teste final: digite "m" para enviar um novo email, e coloque as iniciais de um nome que você sabe que esta no LDAP. Se houver mais de uma ocorrência da palavra/string/nome nos backends, o Mutt automaticamente mostra uma lista, inclusive com a origem do dado :). Se houve apenas uma ocorrência, o Mutt já a coloca no campo "to" que você esta editando.

### 3. Bibliografia

<http://codesorcery.net/mutt/mutt-gnupg-howto>  
<http://mutt.org/>

# Wireless sem suporte no Linux? ndiswrapper nele!!

Já se "defrontaram" com uma placa de rede wireless que não é suportada pelo kernel ou cujos drivers para Linux não compilam em sistemas com kernel atual?

Ao passar por essa experiência (tinha que fazer uma Realtek 8180 funcionar) conversei com uns amigos do #slackware-br e ao mesmo tempo pesquisei no Google. Várias dicas convergiram para o uso do 'ndiswrapper' (<http://ndiswrapper.sourceforge.net/>). Resolvi dar uma chance ao dito cujo.

Abaixo seguem passos para se chegar a colocar a Realtek 8180 pra funcionar:

1- Baixe o ndiswrapper, descompacte e instale:

```
# tar -zxvf ndiswrapper-X.Y.tar.gz
# cd nsdiswrapper-X.Y
# make install
```

2- Baixe o driver para Windows XP da Realtek (<http://www.realtek.com.tw>)

**OBS:** Também achei ridículo ter que usar um driver para Windows XP, mas fazer o quê se o fabricante não tem competência para fazer um driver para Linux que preste?

3- Descompacte o driver para Windows XP:

```
# unzip "ndis5x-8180(170).zip"
```

**OBS2:** O nome do arquivo do driver pode mudar, dependendo de quando você for baixá-lo

4- Utilize o ndiswrapper para ler o arquivo .INF:

```
# ndiswrapper NET8180.INF
```

5- Se tudo deu certo até aqui, então basta carregar o ndiswrapper como módulo:

```
# modprobe ndiswrapper
```

**OBS3:** Para verificar se tudo correu bem até aqui, utilize o dmesg. No meu caso:

```
ndiswrapper: using irq 11
```

```
wlan0: ndiswrapper ethernet device\
      00:30:4f:32:55:5a using driver rtl8180.sys
ndiswrapper device wlan0 supports WPA with
AES/CCMP and TKIP ciphers
ndiswrapper: driver rtl8180.sys
(Realtek,07/09/2004,5.170.0709.2004) added
```

6- Agora é configurar a placa WiFi utilizando o iwconfig. No meu caso, como era apenas um teste, utilizei o seguinte:

```
# iwconfig wlan0 essid "ABSLA" channel 3 mode \
      Managed nick "lalala" rate auto key off
```

OBS.4 -> Porém, no meu caso, a placa não estava aceitando trocar o ESSID e nem mesmo o CHANNEL. Tentei, retentei e não tinha jeito. O que solucionou o problema foi "baixar" a interface wlan0, descarregar o módulo do ndiswrapper e carregá-lo novamente.

```
# iwconfig
lo          no wireless extensions.

wlan0 IEEE 802.11b  ESSID:"ABSLA"  Nickname:"lalala"
Mode:Managed  Frequency:2.442GHz  \
      Access Point: 00:02:78:F4:41:F7
Bit Rate=11Mb/s  Tx-Power:20 dBm  \
      Sensitivity=0/3
RTS thr=2432 B  Fragment thr=2432 B
Encryption key:off
Power Management:off
Link Quality:100/100  Signal level:-48 dBm \
      Noise level:-256 dBm
Rx invalid nwid:0  Rx invalid crypt:0  Rx \
      invalid frag:0
Tx excessive retries:0  Invalid misc:0 \
      Missed beacon:0
```

7- Por último, utilize o ifconfig para setar IP, netmask e broadcast da placa e o route para acertar a rota dos pacotes:

```
# ifconfig wlan0 192.168.0.2 \
      netmask 255.255.255.0 broadcast 192.168.0.255
# route add default gw 192.168.0.1 metric 1
```

Pronto, sua placa Realtek 8180 já deve estar voando.

Yucatan Costa "Kenjiro" <kenjiro@brturbo.com>

Cansou de blá blá blá?

# slackwarezine

a revista técnica de técnicos para técnicos.