

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

19 de Maio de 2003 - Número #3

Editorial

Antes de mais nada, as mais sinceras desculpas pelo atraso. Mas ele valeu a pena, nesta edição temos vários artigos interessantes e alguns novos colaboradores.

Depois do artigo tratando do CUPS, um novo artigo sobre o mais tradicional sistema de impressão, o LPRng (herdeiro do BSD LPD). Agora não existem mais desculpas para não botar uma impressora para funcionar.

Outros dois ótimos artigos tratam do pkgsrc, o sistema de gerenciamento de pacotes do NetBSD. Enquanto alguns se "gabam" por poder usar um clone do ports que emergiu do nada, podemos ter a própria ferramenta de pacotes dos BSD rodando no **slackware**. Ao contrário do ports do Freebie, o pkgsrc foi criado pensando na portabilidade para outros sistemas. Um artigo do Jumpi descreve o pkgsrc e outro do UdontKnow nos mostra como instalá-lo.

Para quem usa o Emacs, uma dica de como configurar a "rodinha" do mouse para funcionar com este editor de textos (sim, ele também edita textos). É a estréia do r_linux na **slackwarezine**.

E, por fim, um artigo ensinando como instalar o Apache2 mais PHP com o TomCat, cada vez mais pedido pelas empresas. Outro artigo de um estreante, o Márcio Rodrigo Miranda.

Espero que gostem desta edição e que se animem com a alta participação de estreantes e mandem os seus artigos!

Piter PUNK

PS> Queria aproveitar e agradecer em nome da equipe editorial do zine a todos que contribuíram com este número.

índice

Configurando a Impressora usando APSFilter+LPRng (Piter PUNK)	2
Configurando a placa de rede no slackware (gar0t0)	4
Ativando a rodinha do mouse no Emacs (r_linux e misfit)	5
Problemas rodando o Emacs sem o X11 (Piter PUNK)	6
pkgsrc (Jumpi)	6
Instalando o pkgsrc do NetBSD no slackware Linux (UdontKnow)	7
Tutorial Apache+TomCat+PHP (Marcio_LED)	9
Conectando-se a uma VPN do MS Windows 2000 usando PPTP (Lorn e Wolvie)	11

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

**"Reproduzida da Slackware Zine #3 -
www.slackware.com.br"**

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



slack
users



Configurando a Impressora usando o ApsFilter+LPRng

1. Introdução

O LPRng é compatível com o LPD, o sistema de impressão "padrão" do mundo *NIX. É simples de se utilizar e de configurar, além de ser extremamente leve.

O maior problema de se utilizar o LPRng é na hora de selecionar e configurar o filtro de impressão. Fazer este serviço manualmente é um dos trabalhos mais terríveis do administrador do sistema, que dirá para um usuário doméstico.

Para resolver este problema foi criado o Apsfilter, uma ferramenta capaz de configurar impressoras com facilidade tanto para o sistema LPRng como para o BSD LPD. Através de menus e respondendo a algumas questões podemos configurar impressoras paralelas, seriais, em redes UNIX, em redes SMB, etc...

2. Antes de Começar

A primeira coisa a verificar é se todos os pacotes necessários estão instalados no sistema:

```
. apsfiler-7.2.5-i386-2
. lprng-3.8.22-i486-1
. espgs-7.05.6-i486-2
. gimp-print-4.2.5-i386-1
. hpijs-1.4.1-i486-1
. ifhp-3.5.10-i386-1
```

Os três primeiros são essenciais, enquanto os últimos são apenas *altamente* recomendados. Os números de versão dos pacotes são os encontrados no Slackware 9.1, se você tiver uma versão mais nova, melhor para você -;)

Agora é a hora de verificar para qual sistema de impressão os comandos `lpr`, `lpc`, `lpq`, `lp*`, estão configurados. Isso é muito importante, muitas pessoas ficam trocando do CUPS para o LPRng e do LPRng para o CUPS várias vezes, e acabam ficando com links errados. Verifique com o comando:

```
$ ls -la `which lpr` | grep -c "lprng"
1
```

Este comando verifica se o link do `/usr/bin/lpr` aponta para `lpr-lprng`. Se apontar, devolve "1", caso contrário, devolve "0". Caso a resposta tenha sido "0", a melhor maneira de corrigir os links é simplesmente reinstalando o pacote do LPRng.

Outra coisa que devemos fazer é desativar o daemon do CUPS, se ele estiver sendo executado:

```
# /etc/rc.d/rc.cups stop
# chmod -x /etc/rc.d/rc.cups
```

O primeiro pára o CUPS, enquanto o segundo faz com que ele não seja reativado no próximo boot da máquina. Agora iremos ativar o LPRng e prepará-lo para reaparecer no próximo boot:

```
# chmod +x /etc/rc.d/rc.lprng
# /etc/rc.d/rc.lprng start
```

Vamos confirmar se tudo está mesmo no ar:

```
# ps ax | grep lpd
761 ?          S          0:00 lpd Waiting
```

E pronto! Podemos passar para a próxima fase! -:)

3. Usando o ApsFilter

Vamos lá, o `apsfilter` é um script que vai dar uma mãozinha com as suas configurações. Ele possui uma licença especial, em que você se compromete a enviar um cartão postal ao autor. Pela facilidade de uso que o `apsfilter` traz, vale a pena pedir o endereço do autor e enviar o cartão postal. Não custa quase nada...

O primeiro passo é executar o comando `SETUP`:

```
# /usr/share/apsfilter/SETUP
```

Logo em seguida, você é questionado se aceita a licença. Seja uma pessoa gentil e aceite-a. A próxima tela pede a confirmação para que seja enviado um e-mail ao autor solicitando o seu endereço, novamente seja uma pessoa educada e envie o e-mail. Se ocorrer algum problema, envie o e-mail manualmente, o endereço do sujeito é: `apsfilter-snailmail@apsfilter.org`.

As próximas duas telas são telas de "propaganda", a primeira seria a "splash screen" do `apsfilter`, dizendo quem fez, a URL e em quais sistemas roda e a segunda explica o que faz exatamente o `apsfilter`. Para passar por estas telas, aperte `<enter>`.

3.1. Os primeiros passos

Finalmente começamos a configuração propriamente dita. A primeira etapa é verificar o `spool` e as permissões dele. Exceto em casos especiais, as permissões do `spool` devem ser `drwx-----` e deve ter como `owner` e `group` o `lp`. Se estiver correto (e geralmente está), apenas confirme com "y" (de Yes).

A próxima tela pergunta se você deseja adicionar uma nova impressora ou se deseja sobrescrever o arquivo de configuração (ou seja, apagar tudo que já existe e escrever de novo). Se você quiser apenas adicionar a nova impressora escolha "a" (add), se deseja sobrescrever, escolha "o" (overwrite).

3.2. Escolhendo um Filtro...

Agora começou a parte importante. Iremos escolher o filtro de impressão. Se você instalou todos os pacotes recomendados, terá dúzias de filtros para escolher e ser feliz. Selecione a opção (1) Printer Driver Selection, e vamos ver a lista de filtros disponíveis...

Difícilmente você terá uma impressora PostScript (talvez na sua empresa), o mais provável é que a sua impressora seja suportada pelos drivers da lista 3, 5 ou 12; respectivamente filtros do GhostScript, do gimp-print e do HPIJS.

Para impressoras Epson, Canon e algumas Lexmarks eu recomendo usar o gimp-print; para impressoras HP recomendo o HPIJS e, se o seu modelo não estiver listado nem no gimp-print nem no HPIJS ou você não gostar do filtro, utilize os filtros do próprio GhostScript.

Após escolher a lista que deseja, percorra-a até achar o modelo da sua impressora. Ao ver o modelo da sua impressora, pressione "q" e, em seguida o número da sua impressora na lista, o apsfiler irá pedir uma confirmação, aperte Y para confirmar. Se a sua impressora não apareceu na lista ou você desejar olhar alguma das outras listas, selecione "0" e escolha outra lista de drivers.

3.3. Onde está a sua impressora?

O próximo passo é selecionar onde a sua impressora está conectada. Até pouco tempo atrás, praticamente todas as impressoras locais estavam conectadas via porta paralela (existiam algumas na porta serial, mas eram as "ovelhas negras" da família). Atualmente, as impressoras USB estão se tornando a cada dia mais comuns, em termos de configuração, ambas são iguais.

Então, escolha a opção (2) Interface Setupe, dentro desta opção, escolha (1) para configurar a sua impressora local. Agora, você deve escrever o nome do dispositivo que a sua impressora está usando... se for uma impressora paralela, deve estar utilizando o /dev/lp0 (primeira porta paralela, a maior parte dos computadores só tem uma mesmo), se for USB, grandes chances do device ser /dev/usb/lp0.

Se a sua impressora estiver na rede, você deve escolher 3) para impressoras disponíveis via LPD ou 4) para impressoras disponíveis via SMB. Você deve ter percebido que deixamos 5) Appletalk e 6) Novell "de fora" das configurações, fizemos isso porque são redes incomuns e o autor deste artigo (eu) não obteve acesso a uma delas para testar.

No caso da sua impressora na rede estar sendo exportada via LPD, você deve conhecer o nome da impressora na rede e o servidor em que ela está:

```
Machine name for remote printer: SERVIDOR
Remote Printer Name: NOME_DA_IMPRESSORA
```

Se você não conhece o nome da impressora, pode usar o comando lpstat para descobrir... faça: lpstat -s servidor, e verá uma lista das impressoras disponíveis naquele servidor.

continua

Agora, se a sua impressora estiver disponível através de uma rede SMB (Windows ou uma máquina Linux/*BSD/*NIX rodando samba), prepare-se para dúzias de perguntas:

```
NetBIOS name of Windows Server: NOME_DO_SERVIDOR
Windows Server IP Address      : IP_DO_SERVIDOR
Printer Share Name             : NOME_IMPRESSORA
Workgroup Name                 : GRUPO_TRABALHO
```

Neste momento aparece uma pergunta diferente, ele pergunta se você irá imprimir como usuário GUEST, é bem comum as impressoras de rede aceitarem este usuário, se for este o caso, selecione Y. Caso contrário, responda:

```
Windows Username              : NOME_DO_USUÁRIO
Windows Password              : SENHA
```

CUIDADO!!! A senha aparece na tela enquanto você preenche este campo! Preste muita atenção se não há ninguém olhando por cima do seu ombro.

3.4. Papel, cores, qualidade, etc...

Esta seção é optativa. Normalmente, cada driver já tem as suas configurações "padrão" para estes parâmetros. Mas, você pode não gostar, pode querer economizar uma tinta ou ainda pode querer imprimir em outro tipo de papel.

Vou passar rapidamente por estas opções:

(3) Paper Format:

Seleciona o tamanho da folha de papel, os mais comuns são o A4 e o US letter. Se você quiser medir a sua folha de papel, lembre que 1 polegada (in) são 25,4mm. Boa Sorte -;)

(4) Printing Quality

Aqui você pode escolher entre 5 opções:

1. draft (esboço): baixíssima resolução, correção de um nível de cor para o outro feita "nas côxas".
2. low (baixa): baixa resolução, transição entre níveis de cor mais suave.
3. medium (média) Média resolução e transição de cores normal.
4. high (alta): Alta resolução e ótima transição de cores...
5. photo (Foto): Altíssima resolução! Transição de cores soberba!

Obviamente, o 1 é a opção que possui a impressão mais rápida e econômica, enquanto o 5 é a mais lenta e "gastona"

(5) Color Mode

Três opções, 1) Colorido, 2) Escalas de Cinza e 3) Preto e Branco

IMPORTANTE!!!

Se você planeja imprimir em uma rede Windows/Samba, lembre de instalar o pacote do samba e editar o /etc/samba/smb.conf. Caso contrário, nada irá funcionar.

continua

continua

(6) Print Resolutions in "dots per inch"

Quantidade de pontos em cada polegada. Muita ATENÇÃO: em vários filtros, a Qualidade de Impressão tem prioridade sobre essa configuração, então você deve alterar lá a resolução.

3.5. Testando e Instalando...

Depois de todas estas configurações, devemos testar a impressora para ver se tudo está OK. Para testar, devemos utilizar a opção (T) Print Test Page. Na seqüência, devemos escolher T novamente (se tivermos certeza de que queremos imprimir uma página de teste) ou qualquer tecla (se quisermos voltar ao menu principal).

Após isso, será apresentado um comando gigante na tela explicando o que o sistema irá tentar fazer para imprimir. Sinceramente, se você não conhece o GhostScript, não tem muitas chances de entender a linha de comando, então é melhor simplesmente apertar "Y", confirmando. Aguarde um pouco que às vezes a impressão demora bastante, principalmente se você escolheu uma alta qualidade para a impressão...

Se tudo correu bem, basta instalar a impressora (I). Caso contrário, selecione um novo driver, ou altere as configurações do driver que está usando e tente novamente, às vezes isso resolve.

Você deve escolher um nome para a impressora (depois de ter selecionado I, para instalar uma impressora). O ideal é que a sua impressora padrão se chame "lp", caso contrário, sempre terá que indicar o nome da impressora quando for imprimir algo.

Se quiser instalar outras impressoras, basta seguir novamente todos os passos e, quando selecionar I, escolher outro nome para a impressora. Quando houver terminado, utilize o comando "Q". E aperte <enter> várias vezes, passando por telas e mais telas de "dicas" e propagandas da banda do autor do apsfiler.

4. Toque Final

Finalmente, reinicie o lpd:

```
# /etc/rc.d/rc.lprng restart
```

ou

```
# lpc reread
```

E pronto! Sua impressora já está configurada e pronta para uso! Espero que você tenha ótimas impressões :-). Este artigo foi testado com várias impressoras: Epson Stylus C42SX, Epson Stylus Color 777 (USB), HP DeskJet 660C, HP LaserJet 1200 (USB) e HP LaserJet 1100A. Algumas testadas com instalação local e outras em rede, tanto via LPD como via SMB.

Não foram testadas especificamente com este artigo, mas já foram configuradas com o apsfiler + LPRng/LPD: Epson ActionPrinter AP2000, Epson LX810, Epson LX810Q+, Canon BJC245, HP Deskjet 610 e outras que não lembro agora, mas com estas já é possível ter uma idéia da quantidade de dispositivos suportados.

Piter PUNK <piterpk@terra.com.br>

continua

Configurando placa de rede no slackware

Como você já deve ter percebido escrevo coisas para newbies (como eu mesmo). Você acha muita coisa no google facilmente mas muitas vezes elas não funcionam como gostaríamos.

Configurar uma placa de rede é muito fácil mas quando estamos começando é um pouco difícil de se perceber as coisas.

No **slackware** tem um **programa** chamado **netconfig**. É em modo texto, mas é muito simples de ser utilizado...

A primeira coisa que o netconfig vai te perguntar é o hostname. Este hostname vai ser o nome da sua máquina. Por exemplo, em: gar0t0@orac:~\$; o nome da minha máquina é orac, e o que vem antes é o meu username =D

A segunda pergunta do netconfig é o Domain Name é o "Domínio" que sua máquina pertence, por exemplo: orac.lala.com, o domínio é o lala.com

A próxima pergunta é como você vai configurar sua placa de rede. Se vai ter um IP estático (você que determina), DHCP (um servidor lhe dá o seu IP) ou loopback (sem rede).

1 - Static IP

Primeira pergunta: Qual o IP que você vai atribuir a Placa de Rede ?

Segunda pergunta: Máscara de Sub-Rede

Terceira pergunta: Informar o número de IP do gateway

Quarta pergunta: Aparece uma caixinha de diálogo perguntando se você acessa servidor de nomes ou não. Aqui você coloca o seu DNS

Quinta e última pergunta: Tudo que você adicionou esta certo ? caso esteja selecione Yes caso contrario No e execute novamente o netconfig

2 - DHCP

Primeira pergunta: ele está pedindo para você informar qual o IP do servidor DHCP

Última pergunta: Se você quer que ele adicione as configurações na placa de rede ou não

3 - Loopback

Utilize esta opção no caso de não possuir placa de rede e mesmo assim quiser os serviços de rede configurados. Máquinas isoladas devem configurar a rede com esta opção.

Bom.. acho que o que eu deveria falar esta aí. Espero que te ajude =) Se tudo der certo sua placa de rede será reconhecida e instalada automaticamente pelo próprio netconfig.

gar0t0 <gar0t0@ig.com.br>

Ativando a rodinha do mouse no Emacs

Mesmo se ativarmos o suporte a rodinha do mouse no X, existem aplicativos que não usam o suporte nativo do X por padrão, mas muitos deles capturam os eventos, portanto podem ser configurados para ativar tal suporte.

No caso do Emacs podemos ativar este suporte de varias maneiras. No **slackware** o Emacs já vem com o arquivo `mwheel.el` que fornece o suporte a MS Intellimouse.

Examinando e ativando a variável

Para examinar o valor da variável use 'C-h v' e em 'Describe variable' você coloca `mouse-wheel-mode`, click com o botão esquerdo no 'customize' e aperte o ENTER. Na próxima tela você ativa, desativa, salva para a sessão corrente, para as sessões futuras, etc.

Você pode ativar também o suporte com 'M-x' `mouse-wheel-mode`, mas só vai funcionar para a sessão atual.

Editando o .emacs

A opção também pode ser ativada diretamente editando o arquivo de configuração local do Emacs, o `~/.emacs`. Na sessão 'custom-set-variables' você coloca a opção `mouse-wheel-mode` para não nula.

```
(custom-set-variables
 '(mouse-wheel-mode t nil (mwheel))
)
```

Note que ele usa o `mwheel.el`.

Customizando manualmente

Se as opções anteriores não o satisfizes, você quer que ele pule um determinado numero de linhas de uma vez, ou escolher outra forma de ativar a rolagem, você pode escrever as funções em seu `~/.emacs`.

No exemplo a seguir, queremos que a rolagem seja de 5 em 5 linhas:

```
(defun up-slightly ()
  "Rolando para cima 5 linhas"
  (interactive)
  (scroll-up 5)
)
(defun down-slightly ()
  "Rolando para baixo 5 linhas"
  (interactive)
  (scroll-down 5)
)
;; Relaciona a rolagem da rodinha com
;; as 2 funções acima
(global-set-key [mouse-4] 'down-slightly)
(global-set-key [mouse-5] 'up-slightly)
```

Neste próximo exemplo, com a tecla Shift pressionada, a rodinha vai rolar uma linha:

```
(defun up-one ()
  "Rolando para cima 1 linha"
  (interactive)
  (scroll-up 1)
)
(defun down-one ()
  "Rolando para baixo 1 linha"
  (interactive)
  (scroll-down 1)
)
;; Relaciona a rolagem da rodinha com o
;; Shift pressionado com as 2 funções acima
(global-set-key [S-mouse-4] 'down-one)
(global-set-key [S-mouse-5] 'up-one)
```

Agora uma função que com a tecla Ctrl pressionada a rodinha vai rolar página a página:

```
(defun up-a-lot ()
  "Rolando uma página para cima"
  (interactive)
  (scroll-up)
)
(defun down-a-lot ()
  "Rolando uma página para baixo"
  (interactive)
  (scroll-down)
)
;; Relaciona a rolagem da rodinha com o
;; Ctrl pressionado com as 2 funções acima
(global-set-key [C-mouse-4] 'down-a-lot)
(global-set-key [C-mouse-5] 'up-a-lot)
```

Podem existir dúvidas quanto ao funcionamento das funções descritas acima, pois o aprendizado de Lisp para Emacs esta fora do escopo deste artigo.

Conclusão

O Emacs é uma ferramenta incrivelmente poderosa. Aqui mostrei apenas uma parte insignificante no meio de uma enorme quantidade de coisas que esta ferramenta pode oferecer em relação a otimização. Procurei apenas explicar como fazer para ativar o botão de rolagem do mouse no Emacs, mas como foi dito no começo deste artigo, qualquer programa que não tenha este suporte automaticamente, pode ser explorado para tentar achar uma forma de configurar o suporte.

r_linux <r_linux@yahoo.com>
misfit <misfit@linuxmail.org>

LEMBRETE:

Para usar este artigo, lembre de instalar o Emacs. Os pacotes se encontram disponíveis na série **E** do **slackware** e podem ser instalados com o comando `installpkg`

Problemas rodando Emacs sem o X11

Apesar de já haver avisado no número anterior, nunca é demais lembrar que alguns binários do **slackware** vem em duas versões, uma compilada com as bibliotecas do X e outra sem.

Este é o caso de dois softwares utilizados nos artigos anteriores, o GhostScript (pacote `espgs`) e o Emacs (pacote `emacs-*`). Se você está montando uma máquina sem X e pretende utilizar os dois, deve utilizar as versões sem as bibliotecas do X.

No caso do Emacs, basta instalar não instalar o pacote `emacs` propriamente dito e sim o `emacs-nox`. Se você já instalou os dois, pode consertar o "erro" de duas maneiras. A primeira consiste em simplesmente instalar o pacote `emacs-nox` novamente, isso irá corrigir o link errado.

A outra maneira é corrigir o link manualmente. Fácil de fazer, já que é apenas um:

```
# rm -rf /usr/bin/emacs
# ln -s /usr/bin/emacs-21.3-no-x11 \
    /usr/bin/emacs
```

E pronto! O seu emacs está pronto para funcionar sem o X! O caso do GhostScript nós já falamos no número passado, mas o rápido resumo é este:

```
# mv /usr/bin/gs /usr/bin/gs-with-x
# ln -s /usr/bin/gs-no-x11 /usr/bin/gs
```

Qualquer dúvida, consulte a edição anterior..

Piter PUNK <piterpk@terra.com.br>.

pkgsrc

O `pkgsrc` é um framework para facilitar a instalação, configuração e manutenção de aplicativos no NetBSD e em outros sistemas UNIX-like e contém em sua árvore cerca de 4000 pacotes pré-definidos para instalação.

Sua árvore é dividida por categorias: audio, chat, comms, databases, devel, editors, emulators, finance, fonts, games, lang, mail, textproc, time, wm, www, x11, etc... Esta divisão tem por tarefa facilitar a localização de pacotes, por exemplo... se eu preciso instalar o `apache`, basta eu acessar /`usr/pkgsrc/www/apache` ou `apache2`.

Bem... vamos a uma breve explicação, /`usr/pkgsrc` - onde se encontra a árvore de pacotes, `www` - aplicativos para trabalhar com world wide web em geral, e `apache`, se quiser instalar o `apache` serie 1.x, ou `apache2` para instalar da serie 2.x, tornando assim a tarefa de instalação intuitiva...

Com o `pkgsrc` eu consigo automatizar a instalação do software através dos fontes, e também posso criar pacotes binários a partir desta instalação para facilitá-la ainda mais. Os fontes e os últimos patches são baixados automaticamente a partir de uma lista de sites disponíveis para download em cada pacote, tem a verificação de checksum para ver a autenticidade do pacote e assim ele é instalado no sistema e, se houver alguma dependência que ainda não esteja instalada, a mesma é baixada e instalada no sistema automaticamente, agilizando o trabalho de instalação.

Todos os pacotes são instalados em um diretório padrão, por default, /`usr/pkg`, onde estão inclusos binários, libs, man pages e qualquer outra documentação do software.

Por ter sido derivado do NetBSD, que se preocupa com a portabilidade do sistema, o `pkgsrc` segue a mesma filosofia: prover portabilidade e consistencia entre todos os sistemas que ele for configurado.

Atualmente as plataformas suportadas pelo `pkgsrc` são: (NetBSD, MacOS X, FreeBSD, IRIX, Linux (**slackware** principalmente), OpenBSD e Solaris e ainda existem ports futuros para BSDi, AIX e Windows Services for Unix.

Bem, em todas plataformas, menos no NetBSD, para trabalharmos com o `pkgsrc` necessitamos baixar o Bootstrap Kit e o Binary Kit, sendo que o bootstrap kit consiste nas ferramentas de administração do pacote, mais precisamente o `pkg_tools`, que vai permitir a construção dos pacotes e o binary kit, seria a árvore propriamente dita do `pkgsrc`.

Continuando o exemplo, para instalar um pacote é muito facil, estando dentro do diretório basta digitar o comando `make install`, e eu terei o `apache` pronto para trabalhar, se eu quiser remover o `apache`, basta eu entrar no diretório de instalação e digitar `make deinstall` para que ele remova o pacote para mim.

Se eu quiser verificar se existe uma versão mais nova do software, basta eu utilizar o `lintpkgsrc` e se houver algo para atualizar, entrar no diretório referente no `pkgsrc` e digitar `make update`, para que seja baixada a nova versão e todas as novas dependências sejam instaladas.

Posso também utilizar outras ferramentas para facilitar o meu trabalho e a administração do `pkgsrc`, tais como `cpuflags` (que verifica quais as melhores flags do compilador para otimizar a compilação do source), `dfdisk` (que baixa os pacotes a partir de vários locais ao mesmo tempo), enfim, um sistema de pacotes estável, totalmente portátil e consistente para o seu **slackware**, tornando assim a tarefa de atualização que antes era árdua, bem mais prática e fácil.

Alan Silva (Jumpi) <jumpi@netbsd.com.br>



Instalando o pkgsrc do NetBSD no slackware Linux

1. Introdução

Este artigo guiará o usuário no processo de instalação do sistema de gerenciamento de pacotes do NetBSD, o `pkgsrc`. Também será abordada uma breve introdução ao gerenciamento do `pkgsrc`.

2. Requisitos

Sistema Operacional **slackware** Linux já instalado e com um conjunto mínimo de bibliotecas e utilitários necessários para o funcionamento do `pkgsrc`
Conexão com a internet funcionando
`pkgsrc` obtido do repositório do NetBSD

2.1. Obtendo e Instalando `pkgsrc`

Para este artigo, o autor utilizou `cv`s para obter os fontes. Há outras formas de obter o `pkgsrc`, mas não serão discutidas neste documento. Maiores informações podem ser encontradas na página do `pkgsrc`.

Exporte a variável `CVSR`OOT para facilitar os comandos subseqüentes:

```
# export CVSROOT=:pserver:anoncv
```

s@anoncvs.netbsd.org/cvsroot

Efetuar login:

```
# cv
```

s login
Senha: anoncvs

Faça download do `pkgsrc` para o diretório correto:

AVISO: Este comando criará um diretório `/usr/pkgsrc` utilizado para armazenar todos os pacotes obtidos da internet, fontes descompactados, objetos e pacotes criados.

```
# cd /usr && cv
```

s co -P pkgsrc

Instale o `pkgsrc`:

AVISO: Isto criará os diretórios `/usr/pkg` para armazenamento de binários e `/var/db/pkg` para banco de dados de pacotes.

```
# cd /usr/pkgsrc/bootstrap && ./bootstrap
```

Adicione ao `PATH`: crie um arquivo `/etc/profile.d/pkgsrc.sh` com o seguinte conteúdo:

```
if [ "`id -u`" = "0" ]; then  
    PATH=/usr/pkg/sbin:$PATH  
fi  
PATH=/usr/pkg/bin:$PATH  
export PATH
```

Faça-o executável:

```
# chmod 755 /etc/profile.d/pkgsrc.sh
```

Adicione o `PATH` à sessão atual:

```
# source /etc/profile.d/pkgsrc.sh
```

Pronto, a base do `pkgsrc` está instalada. Prosseguiremos para a configuração do ambiente.

continua →

www.slackware.com/getslack



Instalando o pkgsrc do NetBSD no slackware Linux

3. Configurando pkgsrc

O pkgsrc vem com um conjunto padrão de configurações, que deve ser adaptado às suas necessidades.

Edite o `/etc/pkgsrc/mk/bsd.pkg.defaults.mk` e altere as opções que forem necessárias. Otimização e flags adicionais devem ser inseridas em `/etc/mk.conf`.

O manual completo do `mk.conf` pode ser encontrado em netbsd.gw.com/cgi-bin/man-cgi?mk.conf++NetBSD-current

3.1. Diferenças de sintaxe

Linux utiliza ferramentas GNU como base, e NetBSD usa ferramentas BSD. GNU `make` é essencial a sistemas Linux, e incompatível com BSD `make`, utilizado pelo pkgsrc. por este motivo, BSD `make` é renomeado `bmake`.

Instalando pacotes essenciais

O processo de instalação do pkgsrc instala quase todos os pacotes essenciais, exceto dois: `audit-packages` e `pkglint`

Instalando `audit-packages`:

```
# cd /usr/pkgsrc/security/audit-packages && \
  bmake && bmake install
```

AVISO: o correto uso do `audit-packages` é descrito ao final do comando `bmake install`, incluindo-se a linha que deve ser configurada no `crontab`

Limpar arquivos-objeto desnecessários:

```
# bmake clean clean-depends
```

Executar o programa de atualização da lista de vulnerabilidades pela primeira vez:

```
# /usr/pkg/sbin/download-vulnerability-list
```

Instalando `pkglint`:

```
# cd /usr/pkgsrc/pkgtools/pkglint && bmake \
  && bmake install
# bmake clean clean-depends
```

Usando `pkglint`:

```
lintpkgsrc -i
  para checar por atualizações na árvore do pkgsrc
lintpkgsrc -o
  para checar por arquivos de distribuição obsoletos
lintpkgsrc -or
  para remover arquivos de distribuição obsoletos
```

Atualizando a árvore do pkgsrc

Após a instalação inicial, a atualização é trivial:

Exportar `CVSROOT` para apontar o repositório do NetBSD, conforme foi feito na seção 2.1.

Atualizar:

```
# cd /usr/pkgsrc && cvs update -dP
```

Informações Adicionais:

O manual completo do pkgsrc pode ser encontrado em The NetBSD Packages Collection (pkgsrc) disponível na URL: www.netbsd.org/Documentation/pkgsrc/

Evaldo Gardenali (UdontKnow)
<evaldo@gardenali.biz>

continua

#ZERO, #2.5 e...

Prepare-se para mais uma edição especial...



slackware zine

Slackware is a registered trademark of Slackware Linux, Inc

...no V FISL, estaremos lá!

De que você precisa?

continua →

Tutorial Apache2 + Tomcat + PHP

Requisitos

- [slackware](#) 9.1
- Apache 2.0.49
- TomCat 5.0.19
- mod_jk
- PHP 4.3.6

Instalação

Apache

Descompacte o arquivo `httpd-2.0.49.tar.gz`

```
# tar -zxvf httpd-2.0.49.tar.gz
```

Entre no diretório criado e execute:

```
# ./configure --enable-cgi \  
--with-static-htpasswd \  
--enable-module=so --enable-ssl
```

Obs : A opção “`--enable-module=so`” é necessária para carga de módulos via `httpd.conf`, que será utilizada na carga dos módulos do php e `mod_jk`, as outras opções utilizadas são de necessidades particulares de cada um, outras opções deste comando podem ser adquiridas através do `./configure --help`.

Após o término desse comando, executar `make` e depois `make install`.

PHP

Descompactar o arquivo, entrar no diretório e executar o seguinte comando:

```
# ./configure \  
--with-apxs2=/usr/local/apache2/bin/apxs
```

Onde a opção indica o caminho de aplicações do Apache, que é necessário para que o PHP seja executado no servidor.

Por fim, criar o arquivo `php.ini`:

```
# cp php.ini-dist /usr/local/lib/php.ini
```

Toda configuração referente ao PHP pode ser feita neste arquivo.

Tomcat

Instalar a última versão disponível do JDK Standard (j2sdk). Definir as seguintes variáveis de ambiente que podem ser incluídas no arquivo `/etc/profile`:

```
#####  
# Variaveis utilizadas pelo Tomcat #  
#####  
JAVA_HOME=/usr/lib/java  
CATALINA_HOME=/usr/local/tomcat  
export JAVA_HOME CATALINA_HOME  
#####
```

Obs: No caso do [slackware](#) 9.1 a variável `JAVA_HOME` não é necessária, pois já é setada por default na instalação se você optar por instalar o Ambiente Java que vem com a distribuição, para verificar se já está setada faça o seguinte:

```
# echo $JAVA_HOME
```

e analise o resultado.

Para instalar o servidor Tomcat, nada é mais necessário do que descompactar o arquivo `.tar.gz` e movê-lo para o diretório que se deseja, isto é claro, se você baixou a versão binária. Depois disso criar o usuário e grupo `tomcat` e atribuí-lo como dono do diretório do `tomcat`:

```
# chown -R tomcat.tomcat /usr/local/tomcat
```

e depois de setadas as variáveis de ambiente, iniciá-lo com o comando:

```
# su tomcat -c \  
/usr/local/tomcat/bin/startup.sh
```

e testar no browser com a URL:

`http://localhost:8080`, que será exibida sua página inicial.

Obs: Edite o arquivo `$TOMCAT_HOME/conf/tomcat-users.xml` e adicione a linha abaixo para poder administrar o servidor:

```
<tomcat-users>  
...  
<user username="admin" password="sua_senha"  
roles="admin,manager"/>  
</tomcat-users>
```

e reinicie o Tomcat e teste.

continua →

segurança? velocidade? estabilidade? poder?

← continua

Configuração

Apache + PHP

É necessário o "mod_so" habilitado no Apache, editar o arquivo `httpd.conf` localizado em `$APACHE_HOME/conf`, e adicionar as seguintes linhas:

```
LoadModule php4_module modules/libphp4.so
AddType application/x-httpd-php .php
```

na cláusula `DirectoryIndex`, adicionar `index.php`.

Após isso reiniciar o Apache e criar o arquivo `phpinfo.php` na pasta de documentos do servidor (`/usr/local/apache2/htdocs`), com o seguinte conteúdo:

```
<? phpinfo(); ?>
```

e depois acesse o seguinte endereço para testar se está tudo funcionando: `http://localhost/phpinfo.php`, se tudo ocorreu bem deverão ser exibidas informações sobre o PHP, senão, será necessário revisar os passos anteriores e corrigir possíveis falhas.

mod_jk

Os passos a seguir instalam e configuram o conector `apache/tomcat`:

Descompacte o arquivo `jakarta-tomcat-connectors-jk2-src-current.tar.gz`

```
# tar -zxvf \
jakarta-tomcat-connectors-jk2-src-current.tar.gz
```

Entre no diretório criado, na pasta `jk/native2` e execute:

```
# ./configure \
--with-apxs2=/usr/local/apache2/bin/apxs
# make
# cd ../build/jk2/apache2
# /usr/local/apache2bin/apxs -n jk2 \
-i mod_jk2.so
```

Adicione a linha abaixo ao `httpd.conf`:

```
LoadModule jk2_module modules/mod_jk2.so
```

Crie o arquivo `workers2.properties` na pasta de configurações do Apache com o seguinte conteúdo:

```
[channel.socket:localhost:8009]
port=8009
host=127.0.0.1
[ajp13:localhost:8009]
channel=channel.socket:localhost:8009

[uri:/servlets-examples/*]
worker=ajp13:localhost:8009
```

Reinicie o Apache:

```
# $APACHE_HOME/bin/apachectl graceful
```

Acesse a URL `http://localhost/servlets-examples`

Se tudo estiver correto, deverá ser exibida a página de exemplos de servlets do Tomcat, e para incluir uma nova aplicação neste esquema basta adicioná-la ao Tomcat, incluí-la no arquivo `/usr/local/apache2/conf/workers2.properties` como foi incluída a aplicação "examples".

Concluindo

Para iniciar os serviços a cada carga do servidor faça o seguinte:

Em `/etc/rc.d/rc.local` adicione as seguintes linhas:

```
echo "Iniciando o Tomcat..."
export JAVA_HOME=/usr/lib/java
export CATALINA_HOME=/usr/local/tomcat
su tomcat -c /
usr/local/tomcat/bin/startup.sh

echo "Iniciando o Apache..."
/usr/local/apache2/bin/apachectl start
```

Bom é isso aí, espero que este documento ajude a quem precisar desta estrutura em sua empresa ou casa.

Marcio_LED <marcio@automat.com.br>

slackware

Tudo que você precisa.

Conectando-se a uma VPN do MS Windows 2000 usando o PPTP

1. Introdução

Neste artigo, irei mostrar como é facil (porém trabalhoso se você nunca fez antes) fazer uma VPN entre slackware e Windows 2000, usando pptp.

O Linux estará sendo usado com o client e o Windows 2000 como server, infelizmente (ou felizmente? :P) a configuração do server tá fora do escopo do tutorial.

2. Instalação

Vá ao site pptpclient.sourceforge.net e faça o download dos seguintes arquivos :

- [ppp-2.4.2_cvs_20040102.tar.gz](#)
- [pptp-linux-1.4.0.tar.gz](#)
- [pptp-extras-1.4.0.tar.gz](#)

O último deles você só irá encontrar no site do projeto no SourceForge:

sourceforge.net/project/showfiles.php?group_id=33063

2.1. ppp-2.4.2

```
# tar xvzf ppp-2.4.2_cvs_20040102.tar.gz \
-C /usr/src/
# cd /usr/src/ppp-2.4.2_cvs_20040102/linux/mppe
```

Agora vamos aplicar o patch no kernel, apesar de que na documentação estar escrito que esse patch só funciona no 2.4.18 e 2.4.19, estou usando em um 2.4.22 e está funcionando normalmente.

```
# sh mppeinstall.sh
```

Ele faz tudo automático, reconhece a versão do seu kernel, e aplica o patch. É recomendável que vc tenha um link simbólico chamado 'linux' para o diretório do kernel, Ex: /usr/src/linux -> /usr/src/linux-2.4.22

Após aplicarmos o patch para o mppe, aplicaremos o patch para o PPP aceitar o mppe.

```
# cd /usr/src/ppp-2.4.2_cvs_20040102/linux/mppe
# sh kinstall.sh
```

Esse funciona em qualquer kernel da série 2.4.

Depois de tudo isso, vamos recompilar o kernel e colocar o mppe como módulo.

```
-> Network device support
-> <M> PPP MPPE compression (encryption)
```

Recompile o kernel:

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

Depois

```
# modprobe ppp_mpppe
```

Agora vamos compilar o "novo" pptp:

```
# cd /usr/src/ppp-2.4.2_cvs_20040102/
# ./configure
# make
# make install
```

2.2. pptp-linux-1.4.0

```
# tar xvzf pptp-linux-1.4.0.tar.gz \
-C /usr/src/
# cd /usr/src/pptp-linux-1.4.0/
# make
# make install
```

3. Configuração

Primeiro vamos criar uma pasta no /etc/ppp/ chamada "peers", que será onde o pptp-command gravará as configurações da VPN.

```
# mkdir /etc/ppp/peers
```

Agora vamos descompactar e instalar o pptp-command:

```
# tar -xvzf pptp-extras-1.4.0.tar.gz \
-C /usr/src
```

Como o pptp-extras é apenas uma "coleção" de programinhas que ajudam a criar VPN mais facilmente, iremos apenas copiar arquivos.

```
# cd /usr/src/pptp-extras-1.4.0/
# make install
```

Vamos editar o options.pptp, e deixar somente as seguintes opções:

```
lock
noauth
nobsdcomp
nodeflate
refuse-eap

require-mppe-40
require-mppe-128
nomppe-stateful
```

Agora com o pptp-command iremos configurar as VPNs:

```
# pptp-command setup
```

Entre na opção:

4.) Add a NEW PPTP Tunnel

Digite o nome da sua VPN:

```
Tunnel Name: teste
```

O ip:

```
Server IP: 200.100.200.200
```

A minha rota eu deixei em branco, não precisei usar rota nenhuma e olha que estou atrás de NAT:

```
route: <Enter>
```

Agora vamos configurar o usuário para se conectar na VPN, o local name. O nome que estará no /etc/ppp/chap-secrets então pode ser qualquer um, desde que você saiba qual é qual.

```
Local Name: nome_no_chaps
```

Esse sim é o usuário da VPN

```
Remote Name : usuario_da_vpn
```

Apesar do ppp-command ter uma interface para se configurar a senha, (1.) Chap Manager), prefiro configurar na mão é bem mais fácil.

```
# vim /etc/ppp/chap-secrets
```

Esse arquivo é muito simples de entender veja só:

```
# Secrets for authentication using CHAP
# client      server      secret IP addresses
nome_no_chaps usuario_da_vpn 123456 200.100.200.200
```

Simple né?

Testaremos agora com o comando:

```
# ppp-command start
```

Se não der nenhum erro, e aparecer a interface ppp0 no ifconfig, significa que funcionou :)

4. Conclusão

Com o ppp-command, fica bem mais fácil criar/conectar/gerenciar as suas VPNs, tirando a parte de aplicar os patches no kernel, o resto é bem fácil, é só prestar bastante atenção para não escrever nada errado.

5. Agradecimentos

Quero agradecer ao wolvie, menos conhecido como Thomas, que me ajudou a configurar o ppp. E cuja revisão deste artigo foi muito útil

Lindolfo Rodrigues aka Lorn <lindolfo@frontservices.com.br>
wolvie aka Thomas <wolvie@unitednerds.org>

continua

AUTORES

Lindolfo Rodrigues / Lorn, é usuário de Linux desde 2000, começou com slackware 8.0, atualmente trabalha na FrontServices, como Analista de Suporte e está no 2o. Semestre de Ciências da Computação na FASP.

Marcio Rodrigo Miranda, estudante de Tecnologia em Desenvolvimento WEB, utilizo Linux desde 1998, mas só a partir de 2003 profissionalmente. Atualmente trabalho como administrador de redes em uma empresa de engenharia de automação onde estou realizando um projeto de migração e concluí meu curso na metade deste ano. Utilizo Slackware desde de junho de 2003.

Piter PUNK, é mantenedor e principal desenvolvedor do slackpkg. Possui experiência com UNIX e Linux desde '96 tendo escrito diversos artigos em revistas da área, atualmente, trabalha como administrador de redes na 3WT Corporation.

Reinaldo Nolasco Sanches (r_linux), atualmente trabalhando na Mandic LTDA no desenvolvimento de aplicações para a Web. Graduando-se em Bacharel de Sistemas de Informações pela Universidade Presbiteriana Mackenzie. Meus interesses são C/C++, Qt, Linux, Sistemas Distribuídos, Programação de Games, Inteligência Artificial, Computação Evolutiva e Heavy Metal.

Tiago Machado a.k.a. gar0t0, 22 anos, estudante de Ciência da Computação. Usuário Slackware desde junho de 2003 quando realmente tomou coragem e apagou o velho windows da máquina. Atualmente, trabalha em uma Faculdade administrando algumas máquinas.

wolvie a.k.a thomas, usa linux desde 98 com uma (BEM) rápida passagem por algumas distribuições fo até encontrar a luz, sobreviveu a 2 anos do curso de m.p.c.e na FATEC-SP e hoje trabalha (ou finge que) com linux e infra-estrutura de redes. dizem as más linguas também que dá pitacos em C, python, shell script e asm-x86. toca guitarra mal e porcamente e tenta tocar um banda de som não definido.

Alan Silva aka jumpi, Membro e um dos fundadores grupo de usuários NetBSD no Brasil (NetBSD-BR). Atualmente trabalha na Takano Online, desenvolvendo aplicações multimídia para web, também cuida da parte de segurança e administração de sistemas e redes da empresa, o vulgo 1001 utilidades. E, às vezes, invade festinhas promovidas por algumas comunidades linux tentando convencer que o diabinho pode ser tão útil quanto o pingüim

Evaldo Gardenali (UdontKnow), usuário slackware desde 1993, estudante de Ciencia da Computacao pela unesp (Bauru-SP), e atua em consultoria, principalmente envolvendo redes (controle de trafego/QoS e outros), Bancos de dados (Informix, PostgreSQL, Sybase, MySQL, oracle), Sistemas Linux, *BSD, Clusters Beowulf e Mosix, e nas horas vagas programa em ruby e auxilia na administracao da rede freenode