

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.

17 de Março de 2004 - Número #2

Editorial

Bem vindos a mais um **slackware zine!** -:) Com um dia de atraso, mas lançado em uma data bem mais festiva que um reles dia 16 de Março. Hoje é 17 de Março!!! Dia de Saint Patrick!!! Por isso, vista-se de verde, arranje um trevo e comemore esta data lendo o slackware zine e com uma caneca de cerveja! -:)

A recepção do primeiro zine foi muito boa e toda a equipe do **slackware zine** se sente muito feliz com isso! Aliás, a recepção primeiro zine foi tão boa, que muita gente copiou e publicou ele em outros lugares, sem nos comunicar e nem ao menos citar o site original (se você não sabe, é www.slackwarezine.com.br). Para evitar que isso ocorra novamente, colocamos logo aqui, na capa, um aviso de que a reprodução do material da revista pode ser feita, mas devem ser mantidos os créditos aos seus autores e a fonte do material. O aviso está logo ali, ao lado.

Muitos artigos foram enviados para a publicação e, acredito, os melhores artigos estão publicados aqui. Mesmo assim, nunca é demais lembrar que os artigos são única e exclusivamente responsabilidade de seus autores. Dúvidas e sugestões devem ser enviadas diretamente a eles. Os iniciantes devem gostar do artigo **O que é?** Feito pelo Ricardo Duarte Dalceno e do **Configurando ADSL no slackware** do gar0t0, enquanto os administradores devem aproveitar bastante do artigo sobre LDAP da dupla Deives Michellis e William Zanatta.

Se você pretende enviar um artigo, ele tem mais chances de ser publicado se utilizar pacotes da distro (quando disponível), ninguém quer perder tempo compilando algo que já está disponível e funcionando. Ganha pontos o artigo que for auto-contido (ou seja, não precisar de parte II, III, etc...). Se o seu artigo for de um pacote que não existe no **slackware** (ou existe mas é necessária alguma opção não disponível no pacote original), tente ir além de `./configure; make; make install`...

Espero que se divirtam com esta edição e aguardem ansiosamente a próxima...

Piter PUNK

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

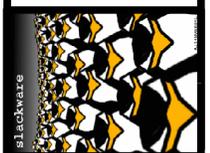
"Reproduzido da Slackware Zine #2 - www.slackwarezine.com.br"

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

índice

Montando um Repositório de Usuários no LDAP similar - Deives Michellis "thefallen" a um "Active Directory" (e sem precisar usar PAM) - William Zanatta	página 2
O que é? - Ricardo Duarte Dalceno	página 5
Como configurar uma bridge no slackware - Sulamita Garcia (Toskinha)	página 5
Imprimindo com o CUPS - Lorn	página 6
Impressoras e... X!?!? - Piter PUNK	página 7
Configurando SSL no Apache - toledo	página 9
Configurando ADSL no slackware - gar0t0	página 11

slack users



Montando um Repositório de Usuários no LDAP similar a um "Active Directory" (e sem precisar usar PAM)

1. Um pouquinho de Teoria

Vamos dar uma passada (bem por cima) para sabermos mais ou menos o que estaremos implementando aqui.

A autenticação de usuário num sistema Unix moderno acontece (a grosso modo) quando um programa faz uma chamada a função `getpwent` ou similares. Esta função consulta a base de usuários do sistema e retorna uma estrutura contendo os dados referentes ao dito usuário, como `username`, `UID`, `GID`, `HomeDirectory`, o `Shell` e, `tcharan`, o `password` criptografado do usuário.

Normalmente, basta consultarmos o `/etc/passwd` e `/etc/shadow` para obtermos esses dados. Simples não? Mas, e se eu quiser usar uma outra fonte de dados além do padrão do sistema? Era uma vez, a muito tempo, uma companhia chamada SUN Microsystems que teve uma idéia muito boa. Por que não termos uma biblioteca que cuida de obter esses dados?

Com isso, basta alterarmos a função `getpwent` e similares num único ponto do sistema, e todas as bibliotecas que usam essa função já poderão estar obtendo dados fornecidos pela nossa biblioteca. Estava nascendo o `NameService Switch`, uma biblioteca que forneceria serviços pra outras bibliotecas, obtendo dados dos arquivos do sistema, de um diretório NIS, de um banco de dados Berkeley DB, de qualquer coisa que tivesse um plugin para o `NameService Switch`. Entre eles, o LDAP. Assim reza o site do `NSS_LDAP` :)

Assim, podemos usar um repositório LDAP como "banco de dados" fornecendo informações para a biblioteca do NSS, que fornece informações para os programas de autenticação do sistema. "Pô, mas quanto intermediário, hein?". É o preço do sucesso :)

De fato, se acessarmos o diretório LDAP diretamente sem nenhum cache, as consultas ficam um pouco lentas (descobri isso quando fiz `ls -l` ehehe). Pra isso vamos contar com a ajuda do `nscd` - `NameService Caching Daemon`. Esse cara faz cache das consultas/resultados providos pelo NSS e melhora significativamente a performance do sistema.

Bom, vamos passar entao "ao que interessa" :)

2. Software utilizado

- * OpenLDAP - o "próprio" :)
<<http://www.openldap.org/>>
- * `nss_ldap`, a biblioteca do NSS que fala com o LDAP
<http://www.padl.com/OSS/nss_ldap.html>
- * `ladmin` - projeto de gerenciador de usuários
<<http://www.unitednerds.org/projects/ladmin>>

3. Instalação

Mãos à obra!

3.1. OpenLDAP

A instalação do OpenLDAP é bem simples. Para nossa instalação, não precisaremos de nenhuma opção especial. Não precisaremos compilar o OpenLDAP. Se quiser compilar para "ver como é que é", sintase a vontade, embora isso saia do escopo deste documento. Simplesmente pegue o pacotinho do linuxpackages.net e boa :) Apenas lembrando que o OpenLDAP 2.1.x requer a BerkeleyDB 4.x pra rodar...

Apenas uma dica: voce encontra a Berkeley DB 4 na sessao "extra" do CD/FTP do [slackware](#).

3.2. nss_ldap

```
# make
# make install
```

Vai instalar a biblioteca `/lib/libnss_ldap.so.2` e criar os arquivos `/etc/nsswitch.ldap` e `/etc/ldap.conf`.

4. Configurando o sistema

4.1. LDAP

Primeiro, vamos definir as configurações da base do diretório. Podemos usar o modelo de `dcObjects` (`dc=meudominio`, `dc=com`, `dc=br`) ou simplesmente o nome da Organization (`o=dominio`).

Normalmente uso o modo de Organization, pois além de mais simples, é bem mais fácil de digitar na hora de fazer as pesquisas :)

Edite o arquivo `/etc/openldap/slapd.conf`. Ele precisa conter as informações listadas no [quadro 1](#).

Feito isso, precisamos inicializar o diretório. Inicie o servidor com o comando `slapd` ou `/usr/libexec/slapd` (dependendo de como foi compilado seu LDAP)

Agora precisamos criar a entrada principal do diretório com o comando `ldapadd` ou `slapadd`. Finalize com 2 enters e um `Control+D`

DICA:

Se você **NÃO** passar a senha correta pro `ldapadd`, ele vai reclamar. Se a senha estiver correta, ele não vai aparecer nada, e pode já começar a digitar.

continua...

Montando um Repositório de Usuários no LDAP similar a um "Active Directory" (e sem precisar usar PAM)

```
thefallen@Ragnarok:~ $ ldapadd -D \  
    "cn=root, o=grupogeo" -x -W  
dn: o=grupogeo  
objectClass: top  
objectclass: Organization  
o: grupogeo
```

Observe que o `o: grupogeo` precisa ser exatamente igual ao `o=grupogeo` do `dn` (Distinguished Name). Feito isso, precisamos agora criar as sub-árvores que vão conter os grupos e usuários.

```
thefallen@Ragnarok:~ $ ldapadd -D \  
    "cn=root, o=grupogeo" -x -W  
dn: ou=People, o=grupogeo  
objectClass: top  
objectclass: OrganizationalUnit  
ou: People
```

```
dn: ou=Users, ou=People, o=grupogeo  
objectClass: top  
objectclass: OrganizationalUnit  
ou: Users
```

```
dn: ou=Groups, ou=People, o=grupogeo  
objectClass: top  
objectclass: OrganizationalUnit  
ou: Groups
```

```
dn: cn=nssuser, o=grupogeo  
objectClass: top  
objectClass: person  
cn: nssuser  
sn: Usuario do NSS_LDAP
```

Vamos agora por uma senha para o usuário `nssuser`. Este usuário vai ter acesso aos passwords (criptografados). Capriche!

```
thefallen@Ragnarok:~ $ ldappasswd -D \  
    "cn=root, o=grupogeo" -W \  
    "cn=nssuser, o=grupogeo"
```

Pronto. Agora estamos preparados pra deixar o script `ladmin` cuidar do resto :)

Se estiver em dúvida quanto ao significado disso que acabou de fazer, veja as definições de `ObjectClass` nos arquivos dentro do `/etc/openldap/schemas/`. São bastante interessantes.

4.2. nss_ldap

Podemos colocar o diretório LDAP como um mecanismo de autenticação primário, único, ou um fall back suplementar.

Para melhorar a performance do serviço, rode o `nscd` (Name Service Cache Daemon), que gera caches pros serviços do NS.

Edite o arquivo `/etc/nsswitch.conf` e mãos à obra!

```
/etc/nsswitch.conf:  
passwd:          files ldap  
shadow:         files ldap  
group:          files ldap
```

No caso do `slackware`, ficaria `passwd: compat ldap`. `compat` é uma macrodefinição de outros módulos do `nsswitch`, incluindo o `files` :) Este tutorial cobre apenas essas opções, embora seja possível por praticamente tudo dentro do LDAP.

Precisamos agora informar ao `nss_ldap` qual a senha do usuário `cn=nssuser, o=grupogeo`. Esse usuário é o único (além do `rootdn`) que tem permissão de ler os passwords guardados na base. Edite o arquivo `/etc/ldap.secret` e coloque a senha lá. O arquivo precisa estar com as permissões `0600`.

```
root@Ragnarok:~# echo \  
MinhaSenhaDoNSSUSER > /etc/ldap.secret  
root@Ragnarok:~# chmod 0600 /etc/ldap.secret
```

Pra configurar o `nscd`, edite o arquivo `/etc/nscd.conf` e adicione as opções do [quadro 2](#) (são default do `nscd`). Mude o `debug-level` para `0` depois de estabilizar todo o sistema :)

Agora é só startar o `nscd` com o comando `nscd -:`). Certifique-se que ele rodara no próximo boot. Rode o comandinho:

```
root@Ragnarok:~# which nscd >> /etc/rc.d/rc.M
```

Vamos agora ao arquivo `/etc/ldap.conf`, que é o arquivo que diz ao `nss_ldap` como lidar com o servidor LDAP. Edite o arquivo conforme as opções do [quadro 3](#)

Pronto! Agora vamos criar nossos usuários com o `ladmin`

4.3. ladmin

Esse é um pacote opcional; é uma ferramentinha que eu montei pra administrar usuários via comandos do shell. Você pode perfeitamente usar outras ferramentas, como o `YALA` ou outros administradores de LDAP. Você encontra um `MONTE` no `freshmeat.net` :)

Vamos primeiro configurá-lo. Edite o arquivo `/usr/lib/ladmin/config` e altere as opções necessárias.

```
/usr/lib/ladmin/config:  
BASEDN="o=grupogeo"  
USERDN="ou=People, o=grupogeo"  
GROUPDN="ou=Groups, o=grupogeo"  
ROOTDN="cn=root, o=grupogeo"  
MINID=20000  
HOST=localhost
```

continua...

Montando um Repositório de Usuários no LDAP similar a um "Active Directory" (e sem precisar usar PAM)

As outras opções devem ser deixadas como estão (a menos que você SAIBA o que esta fazendo...)

Se você estiver usando OpenLDAP<2.1.x, dê uma olhada no `/usr/lib/ladmin/functions`. Tem uma linha que precisa ser descomentada para que tudo "apareça" certinho. Agora podemos começar a criar nossos grupos e usuários.

Rode o comando `ladmin groups add` e crie um grupo no LDAP. Uma "feature" é que ele só aceita trabalhar com os grupos que já estão cadastrados no LDAP (alguém quer converter pra usar o `getpwnam` e funções afins?)

Feito isso, crie usuarios com o comando `ladmin users add`. As opções `xusers` e `xgroups` servem pra editar os grupos suplementares do usuário.

Quadro 2

```
/etc/nscd.conf:
logfile           /var/log/nscd.log
threads           3
server-user       nobody
debug-level       1

enable-cache      passwd           yes
positive-time-to-live  passwd       600
negative-time-to-live  passwd       20
suggested-size    passwd       211
check-files       passwd           yes

enable-cache      group            yes
positive-time-to-live  group        3600
negative-time-to-live  group        60
suggested-size    group          211
check-files       group            yes
```

Quadro 3

```
/etc/ldap.conf:
#
# O IP (IP, nao hostname; agiliza a pesquisa e,
# melhor ainda, fica independente de DNS)
#
host 192.168.0.10
#
# o "basedn" que voce colocou no
# /etc/openldap/slapd.conf
#
base o=grupogeo
#
# O usuario que criamos pra acessar o OpenLDAP
#
rootbinddn cn=nssuser, o=grupogeo
#
# Deixe-o procura em toda a sub-arvore
# do LDAP (base)
#
scope sub
bind_policy hard
```

Quadro 1

```
/etc/openldap/slapd.conf:
#
# Definicoes de ObjectClass e Attributes
#
include           /etc/openldap/schema/core.schema
include           /etc/openldap/schema/cosine.schema
include           /etc/openldap/schema/inetorgperson.schema
include           /etc/openldap/schema/nis.schema
#
# Local de armazenamento dos dados de PID e afins
#
pidfile           /var/lib/openldap/slapd.pid
argsfile          /var/lib/openldap/slapd.args
#
# Parametros para fazer o ldappasswd gerar hashes
# no formato Crypt/MD5
#
password-hash     {CRYPT}
password-crypt-salt-format "$1$%.8s"
#
# Permitir que clients mais antigos consigam se
# conectar ao servidor
#
allow bind_v2
#
# Modelo de ACLs pra "fechar" legal o acesso
# ao diretorio
#
# nssuser eh o usuario (com poucos privilegios) que
# o sistema vai usar pra pegar os dados do usuario.
#
access to attr=userPassword
by self +wx
by dn="cn=nssuser, o=grupogeo" +rsx
by anonymous auth
access to *
by users +xrs
by anonymous +xrs
#
# Tipo de backend que o OpenLDAP vai usar. Por
# padrao, eh bdb (Berkeley DB) no OpenLDAP 2.1.x
#
base             bdb
#
# Nome da organizacao
#
suffix           "o=grupogeo"
#
# Quem eh o super-usuario do diretorio
#
rootdn           "cn=root, o=grupogeo"
rootpw          MinhaSenhaAqui
directory        /var/lib/openldap
#
# Indices pra agilizar a pesquisa
#
index    objectClass    eq
index    uid             eq
index    gidNumber      eq
index    uidNumber      eq
```

Deives Michellis "thefallen" <dmichellis@yahoo.com>,
<thefallen@unitednerds.org>

William N. Zanatta <wzanatta@uol.com.br>,
<psych0byte@uol.com.br>

Esta seção é dedicada aos novatos que ainda se perdem com termos técnicos utilizados por todos os que se aventuram a utilizar algum sistema operacional diferente do Windows. Espero não precisar escrever esta seção por muito tempo (caso contrário, esta seção não terá sido de muita ajuda...)

O que é?

• Sistema Operacional?

É um conjunto de programas que permitem a você utilizar o computador de forma mais simples e para muitas tarefas diferentes. Um sistema operacional é formado por seu kernel, drivers de dispositivo, interpretador de comandos, aplicativos e sistema de arquivos.

No slackware Linux utilizamos o **kernel** Linux em conjunto com aplicativos desenvolvidos por diferentes empresas, grupos ou pessoas isoladas. A maioria destes programas é licenciado sob os termos da GPL, mas existem outras licenças de uso tão importantes quanto.

• Kernel?

Kernel é o núcleo de qualquer sistema operacional. É ele o responsável por "falar" com o hardware e gerenciar os recursos do hardware que você possui. O kernel faz a interface entre os programas utilizados por você, usuário, e o computador (hardware), de forma que você não necessite saber como, exatamente, o seu equipamento funciona.

Nos primórdios da computação, para se utilizar um computador, era necessário saber a linguagem daquele computador. O usuário era também programador e devia conhecer a linguagem de máquina apropriada para aquele computador. (N. do E.: aliás, nos primórdios **mesmo** para programar o computador você tinha que reconfigurar o hardware, trocando fios de lugar e outras aberrações...)

• Drivers de dispositivo?

São pequenos programas que interagem com o kernel e informam ao kernel do sistema como "conversar" com determinado hardware. Por exemplo, para utilizar uma placa de som, você deve ter o driver desta placa. Toda vez que você tentar acessar a placa de som (para ouvir seus mp3, por exemplo), o programa que você utiliza enviará ao kernel os dados que devem ser repassados à placa de som. O kernel, por sua vez, com a ajuda do driver, "conversa" com a placa, entregando os dados que ele recebeu do seu programa de mp3 à placa de som. Então, você tem sua música tocando no seu computador :)

Para a próxima edição, falarei sobre sistemas de arquivos, aplicativos e interpretador de comandos.

Ricardo D. Dalceno <j0k3r@drfinder.com.br>

slackware for the real nerds

Como configurar uma bridge no slackware

Esta é para os paranóicos por segurança ou aqueles que gostam de incrementar os servidores. Que tal tornar seu firewall inacessível para a rede externa? Como? Bom, se ele não possuir um endereço externo, como vai ser acessado? ;) Se configurarmos ele como uma bridge, ele até pode ter um servidor, mas vai funcionar como um filtro na rede que não pode ser acessado.

Acrescentamos um ip para gerenciamento remoto se quisermos, senão ele não pode ser acessado remotamente de forma alguma. Um firewall que não pode ser invadido, que tal?

Para podermos fazer isto, precisamos habilitar o kernel a trabalhar com bridge. Fazemos isto na selecionando a opção "< > 802.1d Ethernet Bridging" em "Networking Options", e recompilando o kernel. Se seu kernel padrão já suporta este protocolo, não há necessidade de recompilar.

Feito isto, podemos brincar de bridge.

Nosso primeiro procedimento será desabilitar toda configuração de rede na inicialização da máquina. Uma maneira simples pode ser tirar a permissão de execução do arquivo `/etc/rc.d/rc.inet1`. Este arquivo configura suas interfaces de rede, se ele não configurar, está correto.

Para trabalhar com a bridge, precisamos trabalhar com o `iproute2` (disponível no diretório `/extra do slackware`). Se você não tem instalado na sua máquina, instale. Aproveite e vá treinando, pois tudo indica que ele irá substituir o `ifconfig` e `route`. Baixe e instale o pacote `bridge-utils` da página <http://bridge.sourceforge.net/bridge-utils.html>. Com ele, vamos criar a interface que será a bridge:

```
# brctl addbr br0
```

Adicione as interfaces ethernet na bridge

```
# brctl addif br0 eth0
# brctl addif br0 eth1
```

Agora vamos habilitar as interfaces

```
# ip link set br0 up
# ip link set eth0 up
# ip link set eth1 up
```

Depois de um delay de aproximadamente 30seg, deve estar funcionando. Alternativamente, pode-se setar um ip para a bridge, mas sempre na interface de bridge, não na ethernet.

```
# ip addr add 1.2.3.4/24 brd + dev br0
# ip route add default via 1.2.3.5
```

Pronto! Agora você pode elaborar suas regras de iptables e fazer um firewall.

Sulamita Garcia <toskinha@noisemakers.org>

Imprimindo com o CUPS



Common
Unix
Printing
System

Introdução

Bom, se você algum dia precisou instalar uma impressora em algum *NIX teve ser encontrado em suas buscas algum site, ou alguém, falando do CUPS (Common Unix Printing System), além dele ser um dos mais conhecidos, é um dos que tem a interface mais fácil de mexer, tudo configurado via browser, chegando a ser até mais rápido e prático do que o famoso NNF (Next Next Finish) do outro systeminha lá. Chega de enrolação e vamos ao que interessa.

Instalação

Como não compensa você instalar o cups pelo código fonte, optei pela instalação via pacote:

```
# installpkg cups-1.1.19-i486-3.tgz
```

Você vai precisar instalar um "driver" auxiliar chamado gimp-print

```
# installpkg gimp-print-4.2.5-i386-1.tgz
```

Vamos instalar também o espgs que é o "ghostscript" do cups...

```
# installpkg espgs-7.05.6-i386-1.tgz
```

Agora você vai precisar instalar o pacote básico do X, "o que? do X?", sim o espgs utiliza uma lib do X libX*.so (alguma coisa assim).

```
# installpkg xfree86-4.3.0-i486-5.tgz
```

Todos esses pacotes podem ser encontrados no cd do slackware, ou em algum mirror de slackware.

Configuração

O arquivo de configuração do cups (/etc/cups/cups.conf) é bem simples, porém uma configuração adequada, aumenta a velocidade da impressão, se você vai instalar a impressora no seu desktop, e não quer compartilhar ela via rede, pule essa parte de configuração, pois não é necessária, ao menos que você tenha algum problema.

cups.conf

Seção "Server Identify"

É interessante descomentar a opção

```
#ServerAdmin root@your.domain.com
```

pois, se der algum problema no cups ele manda um email para o root@your.domain.com, e é claro o servidor de email tem que ser na mesma maquina do cups, ou estar configurado corretamente.

Seção "Server Options"

```
#ErrorLog /var/log/cups/error_log
```

Nem preciso dizer que é bom desabilitar essa opção, qualquer errinho que der no cups, ele vai gravar no /var/log/cups/error_log, e acredite esse log é muito bom, ainda mais se for usado em modo Debug (somente quando necessário), o modo debug é muito bom, pois mostra erros minuciosos do cups, como por exemplo o da libX*.so se vc não instalar o X, mais aviso muito cuidado ao habilitar esse modo, ele enche muito o log, portanto habilite, encontre o problema, resolva, desabilite.

Tipos de Log, como disse acima há varios tipos de log, normalmente eu deixo no info, em casos extremos deixo em debug2log

```
# debug2Log everything.  
# debugLog almost everything.  
# info Log all requests and state changes.  
# warn Log errors and warnings.  
# error Log only errors.  
# none Log nothing.  
LogLevel info
```

Seção "Filter Options"

Nessa parte precisamos definir com qual usuário e grupo o cups irá rodar, eu costumo deixar no usuário e grupo default:

```
User lp  
Group sys
```

Mais se você preferir outro usuário e grupo, gosto não se discute :)

Seção "Network Options"

Nessa parte se define em qual porta o cups rodará o seu daemon, se você não descomentar, ele vai usar a default, 631:

```
Port 631
```

Seção "Browsing Option"

Nessa parte eu não descomentei nada, simplesmente porque não vou usar nada, nessa parte você poderia configurar o cups para que a interface web dele não seja acessada somente pelo localhost, mais sim pela rede inteira.

continua...



Imprimindo com o CUPS

Seção "Security Options"

Essa opção você define qual grupo irá, ter "poderes" de root no cups eu coloquei o grupo root, mais você pode colocar qualquer outro.

```
SystemGroup root
```

Outra coisa interessante de se fazer aqui, é restringir apenas para sua rede poder imprimir.

```
<Location />  
  Order Deny,Allow  
  Deny From All  
  Allow From 127.0.0.1  
  Allow From 192.168.1.0/24  
</Location>
```

Com isso, eu "falo" para o cups que ele só pode aceitar impressão do localhost (127.0.0.1), e da rede 192.168.1.0/24

Instalando a impressora:

Inicie o cups, (/etc/rc.d/rc.cups start).

Caso você não tenha instalado o cups junto com o **slackware**, ele não irá criar o rc.cups, portanto, no **Quadro 1** vai uma versão "resumida" do original, é só colar ela no editor de sua preferência, copiar para o /etc/rc.d, e colocar uma referência a ele no rc.M, á não se esqueça de dar permissão para executá-lo (chmod +x rc.cups). Abra seu browser favorito, e se conecte ao localhost:631, no meu caso:

```
lynx localhost:631
```

Agora você verá, toda a interface "bonitinha" do cups, vá em Manage Printers e depois Add Printer, provavelmente ele vai pedir um usuário e senha agora, use o usuário e senha de root da sua maquina.

No campo Location, coloque /dev/lp0, que o dev responsável pelo controle da porta paralela. Depois continue.

Agora na parte Device, escolha Parallel Port #1 (Marca da sua impressora), Exemplo, Parallel Port #1 (EPSON).continue. novamente.

Agora escolha a marca de sua impressora. E selecione mais uma vez continue

Finalmente agora escolha um driver, escolha um driver tipo modelo_da_impresora, Cups+Gimp Print, Exemplo:

```
EpsonStilus, Cups+Gimp
```

Pronto! Finalmente terminou :)

"Oh, e agora? como faço para imprimir?"

Calma, calma...Se você usa OpenOffice, Abiword, etc..eles reconhecerão automaticamente a nova impressora, caso isso não aconteça, imprima com a opção "Printer Command", então digite lpr

Caso você queira compartilhar essa impressora em uma rede *nix, instale o cups na máquina que você irá compartilhar, inicie o daemon (cupsd), e então vá em /usr/lib/cups/daemon e execute:

```
# cups-polld ip porta tempo porta
```

Exemplo, cups-polld 192.168.23.10 631 30 631, com isso de 30 em 30 segundo ele irá verificar se a impressora está lá, e para imprimir faça a mesma coisa caso a impressora estivesse ligada na sua máquina.

continua...

Impressoras e... X!?!?

O slackware vem com duas versões do espgs (ESP GhostScript). Uma delas está linkada com o X, para uso do gv e outros programas que utilizam o ghostscript para renderizar documentos PostScript (ou PDF) diretamente no X. Já outra versão, não precisa do X para funcionar.

Quando instalamos o sistema para um desktop ou uma workstation, normalmente instalamos com o X. E, para facilitar a vida dos usuários de desktop, o ghostscript "padrão" é o que está linkado com as bibliotecas do X.

Mas, se formos instalar em uma máquina sem X, como a maior parte dos servidores, é muito importante "mudar" a versão do ghostscript para a que não utiliza o X (afinal, para quê instalar um pacote que não será usado?)

O binário linkado com a Xlib é o: /usr/bin/gs, enquanto o outro é convenientemente chamado de /usr/bin/gs-no-x11

Para trocar de binário, basta fazer a dança das cadeiras:

```
# cd /usr/bin  
# mv gs gs-with-x11  
# mv gs-no-x11 gs
```

E pronto! Agora você já pode utilizar tranqüilamente o seu ghostscript sem precisar instalar os pacotes do X. Afinal, não é todo servidor de impressão que tem todo este espaço no /usr não é? -;)

Lembre-se disto tanto ao usar o CUPS como ao usar o LPRng, já que os dois utilizam os filtros do ghostscript.

Piter PUNK



Imprimindo com o CUPS

Lindolfo Rodrigues / Lorn <lindolfo@frontservices.com.br>

Quadro 1

```
#!/bin/sh
#
# Startup/shutdown script for the Common UNIX Printing System (CUPS).
# Copyright 1997-2003 by Easy Software Products, all rights reserved.
#
# These coded instructions, statements, and computer programs are the property
# of Easy Software Products and are protected by Federal copyright law.
# Distribution and use rights are outlined in the file "LICENSE.txt" which
# should have been included with this file.  If this file is missing or damaged
# please contact Easy Software Products at:
#
#       Attn: CUPS Licensing Information
#       Easy Software Products
#       44141 Airport View Drive, Suite 204
#       Hollywood, Maryland 20636-3111 USA
#
#       Voice: (301) 373-9603
#       EMail: cups-info@cups.org
#       WWW: http://www.cups.org
#
# Esse script foi alterado por Lindolfo Rodrigues "Lorn", pois
# precisava que esse script funcione apenas no linux.
IS_ON=/bin/true

# See if the CUPS server (cupsd) is running...
pid=`ps ax | awk '{if (match($5, ".*cupsd$") || $5 == "cupsd") print $1}'` ;

# Start or stop the CUPS server based upon the first argument to the script.
case $1 in
  start|restart|reload)
    if $IS_ON cups; then
      if test "$pid" != ""; then
        kill -HUP $pid
      else
        /usr/sbin/cupsd
        if test $? != 0; then
          echo "cups: unable to $1 scheduler."
          exit 1
        fi
      fi
    fi
    echo "cups: ${1}ed scheduler."
  ;;
  stop)
    if test "$pid" != ""; then
      kill $pid
      echo "cups: stopped scheduler."
    fi
  ;;
  status)
    if test "$pid" != ""; then
      echo "cups: scheduler is running."
    else
      echo "cups: scheduler is not running."
    fi
  ;;
  *)
    echo "Usage: cups {reload|restart|start|status|stop}"
    exit 1
  ;;
esac

exit 0
```

Artigos técnicos,
de técnico para
técnico.

Finalmente, uma
revista que fala
mesmo a nossa
língua.

slackware zine

Sem estudos de
caso e sem papo
político.

Uma revista de
Linux para quem
gosta de
computador.



Configurando SSL no Apache

1. Introdução

Seja numa transferência bancária, efetuando uma compra pela internet (e-commerce) ou até mesmo num formulário de dados pessoais, é possível implementar e usar SSL junto ao seu webserver. A sigla SSL quer dizer Secure Socket Layer, com ela o fluxo de dados entre o cliente (você) e o servidor (sites de e-commerce e etc) é todo criptografado. Neste artigo vamos aprender como assegurar qualquer site usando o mod_ssl.

Atenção: Ao longo do artigo me refiro a alguns pacotes, entenda-se que estes são padrões do **slackware**.

2. Requerimentos

```
apache-1.3.29-i486-1      - Apache webserver
mod_ssl-2.8.16_1.3.29-i486-1- Módulo SSL para apache
openssl-0.9.7c-i486-2    - Criação de certificados
openssl-solibs-0.9.7c-i486-2- Libs do openssl
```

3. mod_ssl

O mod_ssl é um módulo do Apache baseado em OpenSSL e ele será a base do nosso artigo. É muito simples habilitar o mod_ssl no **slackware**, basta editar o /etc/apache/httpd.conf e no final do arquivo descomentar a seguinte linha:

```
#Include /etc/apache/mod_ssl.conf
```

Para descomentar basta tirar o # no começo da linha e reiniciar o apache. Como teremos que fazer mais algumas configurações, deixaremos para reiniciar o apache quando tivermos tudo ok!

4. Criando chaves privadas e certificados

Usaremos o comando openssl para gerar tudo o que precisaremos.

4.1. Chave privada RSA

Iremos gerar uma chave RSA composta por 1024 bit usando Triple-DES. Com o seguinte comando:

```
# openssl genrsa -des3 -out dominio.key 1024

Generating RSA private key, 1024 bit long modulus
.+++++
.....+++++
e is 65537 (0x10001)
Enter pass phrase for dominio.key:
```

Ao executar esse comando pedirá uma 'pass phrase', consiste em uma senha que será usada toda vez que for iniciado o Apache. Após digitar a senha, a chave será criada no arquivo dominio.key

DICA:

Para tirar o incômodo de ter que digitar a senha toda vez que o Apache subir, basta gerar a key tirando o parametro '-des3'

4.2. Certificado CSR (Certificate Signing Request)

O Certificado CSR é passado por um check de autenticidade e expiração, checado por algum órgão de Certificação Digital. Geralmente esses órgãos cobram entre 400 a 1500 dólares e os mais conhecidos são: Verisign (www.verisign.com) e Thawte(www.thawte.com)

Para gerar esse certificado 'baratinho' usamos o comando:

```
# openssl req -new -key dominio.key \
-out dominio.csr
```

Basta preencher as perguntas como no quadro abaixo, nada de outro mundo. As últimas duas perguntas, as que se referem a atributos extras, deixei-as em branco, mas você pode preencher se quiser.

continua... →

```
Enter pass phrase for dominio.key: aqui você coloca sua pass phrase escolhida acima
```

```
You are about to be asked to enter information that will be incorporated
into your certificate request.
```

```
What you are about to enter is what is called a Distinguished Name or a DN.
```

```
There are quite a few fields but you can leave some blank
```

```
For some fields there will be a default value,
```

```
If you enter '.', the field will be left blank.
```

```
-----
```

```
Country Name (2 letter code) [AU]:BR
```

```
State or Province Name (full name) [Some-State]:Sao Paulo
```

```
Locality Name (eg, city) []:Sao Paulo
```

```
Organization Name (eg, company) [Internet Widgits Pty Ltd]:SlackwareZine
```

```
Organizational Unit Name (eg, section) []:.
```

```
Common Name (eg, YOUR name) []:www.dominio.com.br
```

```
Email Address []:domain@dominio.com.br
```

```
Please enter the following 'extra' attributes
to be sent with your certificate request
```

```
A challenge password []:
```

```
An optional company name []:
```

Configurando o SSL no Apache

4.3. Certificado Self-Signed

Você pode testar seu Certificado sem usar um órgão de Certificações Digitais, usando o Self-Signed. Para criarmos um certificado temporariamente por 30 dias, digitamos:

```
# openssl x509 -req -days 30 -in dominio.csr \  
-signkey dominio.key -out dominio.crt
```

E pronto! Seu certificado está criado para testes. Toda vez que utilizar o Self-Signed aparecerá uma mensagem dizendo que o certificado não é totalmente seguro, essa mensagem é devido a não utilização de um órgão de Certificações Digitais.

5. Configurando o Apache para SSL

Com todos os certificados criados, precisamos agora configurar nosso Apache para que reconheça e use-os. Editamos o arquivo `/etc/apache/mod_ssl.conf`

```
<IfDefine SSL>  
<VirtualHost _default_:443>  
ServerAdmin dominio@dominio.com.br  
DocumentRoot /var/www/htdocs  
ServerName www.dominio.com.br  
SSLEngine on  
SSLCertificateFile /etc/apache/ssl.crt/dominio.crt  
SSLCertificateKeyFile /etc/apache/ssl.key/dominio.pem  
SetEnvIf User-Agent ".*MSIE.*" nokeepalive ssl-unclean-  
shutdown  
CustomLog /var/log/apache/ssl_request_log \  
"%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \  
"%r\" %b"  
</VirtualHost>  
</IfDefine>
```

Com essa configuração criamos um `VirtualHost` para `www.dominio.com.br`, podemos criar outros `VirtualHosts` também, bastando repetir a configuração trocando o `VirtualHost` (não entrarei em detalhes, pois não é o foco do artigo).

DICA:

Percebam que cada certificado eu criei e copiei para os diretórios padrões localizado em `/etc/apache/`, como `ssl.crt/`, `ssl.key/` e etc. É interessante seguir os padrões, torna-se mais fácil de você se achar mais tarde.

5.1. Inicializando nosso serviço SSL

Após todas essas gerações de certificados e configurações de apache, iremos iniciá-lo e testá-lo, vamos lá:

```
# apachectl stop (caso o webserver esteja no ar)  
# apachectl startssl (essa é a flag para  
podermos inicializá-lo com suporte a SSL)
```

Se tudo correr bem ele pedirá a 'pass phrase', basta colocar a mesma digitada para gerar os certificados.

E pronto!

Vamos aos testes. Abra seu browser preferido e tente conectar no seu domínio usando:

```
https://www.dominio.com.br
```

ou

```
http://www.dominio.com.br:443
```

Você verá uma mensagem de site seguro, juntamente com um cadeado fechado no canto inferior de determinados browsers. Agora você já pode colocar aquele seu projetinho de e-commerce no ar, com suporte a criptografia para pagamentos via site.

6. Conclusão

Não existe nada seguro, o que nós fazemos é aumentar o grau de segurança de nossas aplicações. Administrar um site de e-commerce ou afins é algo difícil, todo cuidado é pouco. Ler logs faz bem a vida de todos :) Uma segurança planejada e bem administrada é algo raro em nossa praia ;)

toledo <toledo@core-dumped.org>

Colocando o Apache+SSL para iniciar no boot da máquina

Para automatizarmos o serviço de apache com suporte a SSL iremos alterar alguns arquivos de inicialização (isto é para caso o servidor reinicie por algum motivo). Edite o arquivo `/etc/rc.d/rc.httpd` para que fique assim:

```
case "$1" in  
  'startssl')  
    /usr/sbin/apachectl startssl ;;  
  'start')  
    /usr/sbin/apachectl start ;;  
  'stop')  
    /usr/sbin/apachectl stop ;;  
  'restart')  
    /usr/sbin/apachectl restart ;;  
  *)  
    echo "usage $0 start|startssl|stop|  
restart" ;;  
esac
```

E, no `/etc/rc.d/rc.M` altere a parte que se refere ao apache:

```
# Start Web server:  
if [ -x /etc/rc.d/rc.httpd ]; then  
  . /etc/rc.d/rc.httpd startssl  
fi
```

LEMBRE-SE!!! Se você realizar estas alterações, e gerar a chave com o parâmetro `-3des`, será pedida uma senha toda vez que o seu computador rebootar.

Para fazer com que seja um processo REALMENTE automático, sem necessitar da sua intervenção, gere a sua chave com o comando:

```
# openssl genrsa -out dominio.key 1024
```

Autores

Deives Michellis "thefallen", Tecnólogo em Processamento de Dados pela FATEC/SP e Gerente de Desenvolvimento de Soluções Linux do Grupo GEO. Também nerd de carteirinha e ativista linux nas horas vagas.

Leandro Toledo, 19 anos, iniciou com computadores em 1993 e Linux em 1998, usando Slackware 3.4 kernel 2.0.30. Desde então, vem acompanhando a evolução desse maravilhoso sistema juntamente com toda a comunidade de softwares livres. Trabalha na área a 2 anos, atualmente, como sysadmin de redes linux numa agência de publicidade.

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.

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.

Ricardo Duarte Dalceno a.k.a. j0k3r, Técnico em Processamento de Dados, Estudante de Ciência da Computação, PUC-SP. Trabalhando com computação em geral desde 1993 e com Linux desde 1998. Estudo sempre focado em redes e Sistemas Distribuídos. Participante ativo do projeto Rau-Tu de Sistemas Distribuídos, Tópico de Sistemas Operacionais Distribuídos <http://rau-tu.fai.com.br/sd/>

Sulamita Garcia a.k.a. Toskinha, trabalha com Linux desde 1999 e com Slackware desde 2001, como Analista de Suporte. Certificada LPIC-1, é mantenedora do site de Alta Disponibilidade na Underlinux. Atualmente participa do projeto LinuxChix-Br, e trabalha como Projetista de Software da Cyclades Corporation.

Tiago Machado a.k.a. gar0t0, 21 anos, estudante de Ciência da Computação. Iniciou com computadores aos 14 anos usando um incrível XT, começou a interessar-se realmente aos 17 anos quando quebrou o computador e não tinha ninguém pra arrumar. 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.

William N. Zanatta a.k.a. psych0byte, 22 anos, estudante de Engenharia da Computação na FASP. Certificado LPIC-2. Começou com computadores ha cerca de 12 anos, em 1996 conheceu o Linux e o Slackware, única distribuição usada desde então. Atua nas áreas de desenvolvimento, administração de sistemas e atualmente também como consultor da Veritel Solutions.

Configurando ADSL no slackware

1 - Primeiro Passo - verificar a existência do pacote rp-pppoe

Para verificar eu vou usar o `slackpkg search`, que verifica se tenho o pacote instalado e caso eu não o tenha ele também ira reclamar.

No meu caso:

```
root@slack:~# slackpkg search rp-pppoe

[uninstalled] - rp-pppoe-3.5-i386-1
```

Este pacote não está instalado. então:

```
root@slack:~# slackpkg install rp-pppoe

rp-pppoe-3.5-i386-1.tgz
```

Do you wish to install those packages (Y/n)? Y

Package: rp-pppoe-3.5-i386-1.tgz

```
Downloading... - SUCCESS!
Checking... - SUCCESS!
Installing rp-pppoe-3.5-i386-1...
```

```
root@slack:~#
```

Pacote instalado vamos agora para parte de configuração.

2 - Segundo Passo - Configuração

Para configurar a sua conexão, use o comando `adsl-setup`. Vou explicar um pouquinho de cada coisa que ele pede, mesmo sendo tudo bem intuitivo.

2.1 - Nome do Usuário

```
USER NAME
>>> Enter your PPPoE user name (default bxxxxnxx@sympatico.ca):
```

Esta primeira parte só esta te dando um "welcome". e pede para você digitar o nome do usuário como no exemplo

2.2 - Interface (Placa de rede)

```
INTERFACE
>>> Enter the Ethernet interface connected to the ADSL modem
For Solaris, this is likely to be something like /dev/hme0.
For Linux, it will be ethn, where 'n' is a number.
(default eth0):
```

Nesta parte você só tem que especificar qual é a sua placa de rede. `eth0`, `eth1`, ... ai é com você. Por padrão a placa de rede default é `eth0`.

continua... ➔

Colocando o "." do teclado numérico para funcionar

Essa já tirou o sono de muita gente que usa o teclado ABNT2. Se o seu "." não funciona, apenas execute (dentro do X):

```
$ xmodmap -e 'keycode 91 = period'
```

tendo sucesso, inclua este comando no seu `.xinitrc`

Configurando ADSL no slackware

2.3 - Conexão por Demanda

Do you want the link to come up on demand, or stay up continuously?
If you want it to come up on demand, enter the idle time in seconds
after which the link should be dropped. If you want the link to
stay up permanently, enter 'no' (two letters, lower-case.)

NOTE: Demand-activated links do not interact well with dynamic IP
addresses. You may have some problems with demand-activated links.

>>> Enter the demand value (default no):

Aqui ele pergunta se vc quer tem uma conexão por demanda. Caso você coloque sim tem que saber o que está fazendo. Caso não saiba o que é aperte a tecla enter porque você não vai usar.

2.4 - DNS

Please enter the IP address of your ISP's primary DNS server.
If your ISP claims that 'the server will provide DNS addresses',
enter 'server' (all lower-case) here.

If you just press enter, I will assume you know what you are
doing and not modify your DNS setup.

>>> Enter the DNS information here:

Caso você tenha o DNS que seu provedor forneceu coloque aqui. Se não tem coloque apenas "server" e continue a instalação.

2.5 - PASSWORD

PASSWORD

>>> Please enter your PPPoE password:

Acho que esta parte não preciso falar muito, né? É só digitar a senha e confirmar, nada muito especial que ninguém nunca tenha feito.

2.6 - FIREWALLING

FIREWALLING

The firewall choices are:

0 - NONE: This script will not set any firewall rules. You are responsible
for ensuring the security of your machine. You are STRONGLY
recommended to use some kind of firewall rules.

1 - STANDALONE: Appropriate for a basic stand-alone web-surfing workstation

2 - MASQUERADE: Appropriate for a machine acting as an Internet gateway
for a LAN

>>> Choose a type of firewall (0-2):

Esta é a ultima opção é uma das mais importantes que você deve escolher. Aqui você vai falar se não quer firewall, se quer um firewall padrão para navegar na internet e o ultimo deve ser usado se você usa esta maquina como um gateway na sua rede interna. Caso você esteja na sua casa e não quer ter nenhum probleminha eu sugiro que selecione a opção 0. Caso você escolha alguma opção de Firewall certifique-se que o ipchains está instalado em seu sistema.

3 - Terceiro Passo (último) =P

Apenas confirme se com um y ou n se as informações que foram passadas no final estao corretas. Parabéns. Você terminou de configurar o seu "discador" ADSL. Execute o comando `adsl-start` para se conectar e boa navegação. Você pode adicionar `adsl-start` no seu `/etc/rc.d/rc.local`. Assim quando o `rc.local` for executado você já estará conectado a internet



II

Encontro
Nacional
LinuxChix
Br

1 e 2 de maio
das 9h00 as 18h00

**Palestras
voltadas
para o
público
técnico**

R\$10,00

**chega de
blábláblá**

www.linuxchix.org.br

