

CURSO DE LINUX BÁSICO

Versão 2.0

Copyright (c) 2001 Artur de Paula Coutinho

É dada permissão para copiar, distribuir e/ou modificar este documento sob os termos da Licença de Documentação Livre GNU, Versão 1.1 ou qualquer versão posterior publicada pela Free Software Foundation; sem Seções Invariantes, sem Textos de Capa da Frente e sem Textos da Quarta-Capa. Uma cópia da licença está inclusa na seção intitulada "Licença de Documentação Livre GNU".

ÍNDICE

- 0 – HISTÓRICO
- 1 – INTRODUÇÃO
 - 1.1 – O que é?
 - 1.2 – Características
 - 1.2.1 – Multiusuário
 - 1.2.2 – Multitarefa
 - 1.2.3 – Código aberto (GPL)
 - 1.2.4 – Custo
 - 1.3 – Distribuições
 - 1.4 – Informações
 - 1.4.1 – Manuais
 - 1.4.2 – HOWTO
 - 1.4.3 – Livros
 - 1.4.4 – Listas de discussão
 - 1.4.5 – Sites na Internet
 - 1.4.6 – Revistas
- 2 – INSTALAÇÃO
 - 2.1 – Precauções antes da instalação
 - 2.2 – Formas de instalação
 - 2.3 – Meios de instalação
 - 2.4 – Instalação
- 3 – PRIMEIRA UTILIZAÇÃO
 - 3.1 – Conceitos básicos
 - 3.1.1 – Carregamento do sistema (lilo)
 - 3.1.2 – Entrada no sistema (login)
 - 3.1.3 – Saída do sistema (logout ou exit)
 - 3.1.4 – Encerramento do sistema (shutdown)
 - 3.1.5 – Alternando entre consoles
- 4 – COMANDOS BÁSICOS I
 - 4.1 – ls
 - 4.2 – | (pipe)
 - 4.3 – more
 - 4.4 – less
 - 4.5 – cd
 - 4.6 – Comandos de data e hora
 - 4.7 – man
- 5 – EDITANDO TEXTOS (pico)
- 6 – COMANDOS BÁSICOS II
 - 6.1 – mkdir
 - 6.2 – rm
 - 6.3 – cp
 - 6.4 – mv
 - 6.5 – find
- 7 – TIPOS DE ARQUIVOS
 - 7.1 – Links
 - 7.2 – Metacaracteres
- 8 – COMANDOS BÁSICOS III
 - 8.1 – ln

9 – USUÁRIOS E GRUPOS

- 9.1 – Porque criar usuários?
- 9.2 – O conceito de grupo

10 – PERMISSÕES DE ACESSO

11 – COMANDOS BÁSICOS IV

- 11.1 – useradd
- 11.2 – passwd
- 11.3 – chown
- 11.4 – chgrp
- 11.5 – chmod
- 11.6 – su
- 11.7 – userdel

12 – SISTEMA DE ARQUIVOS

- 12.1 – Conceitos básicos
 - 12.1.1 – O que é?
 - 12.1.2 – Ponto de montagem
- 12.2 – Diretório raiz (/)
- 12.3 – Diretório /boot
- 12.4 – Diretório /bin
- 12.5 – Diretório /sbin
- 12.6 – Diretório /root
- 12.7 – Diretório /home
- 12.8 – Diretório /usr
 - 12.8.1 – Subdiretório /usr/X386
 - 12.8.2 – Subdiretório /usr/bin
 - 12.8.3 – Subdiretório /usr/dict
 - 12.8.4 – Subdiretório /usr/etc
 - 12.8.5 – Subdiretório /usr/include
 - 12.8.6 – Subdiretório /usr/lib
 - 12.8.7 – Subdiretório /usr/local
 - 12.8.8 – Subdiretório /usr/man
 - 12.8.9 – Subdiretório /usr/sbin
 - 12.8.10 – Subdiretório /usr/share
 - 12.8.11 – Subdiretório /usr/src
- 12.9 – Diretório /proc
- 12.10 – Diretório /dev
- 12.11 – Diretório /etc
- 12.12 – Diretório /lib
- 12.13 – Diretório /tmp
- 12.14 – Diretório /mnt
- 12.15 – Diretório /var
 - 12.15.1 – Subdiretório /var/adm
 - 12.15.2 – Subdiretório /var/catman
 - 12.15.3 – Subdiretório /var/lib
 - 12.15.4 – Subdiretório /var/local
 - 12.15.5 – Subdiretório /var/lock
 - 12.15.6 – Subdiretório /var/log
 - 12.15.7 – Subdiretório /var/named
 - 12.15.8 – Subdiretório /var/nis
 - 12.15.9 – Subdiretório /var/preview
 - 12.15.10 – Subdiretório /var/run
 - 12.15.11 – Subdiretório /var/spool
 - 12.15.12 – Subdiretório /var/tmp

Curso de Linux Básico

13 – ACESSANDO HD, CDROM E DISQUETE

- 13.1 – Nomes dos Dispositivos
- 13.2 – Montagem de Dispositivo
- 13.3 – Ponto de Montagem

14 – COMANDOS BÁSICOS V

- 14.1 – mount
- 14.2 – umount
- 14.3 – fdformat
- 14.4 – mkfs

15 – FORMAS DE ACESSO A DISQUETES

- 15.1 – Usando Dois Pontos de Montagem
- 15.2 – Usando Apenas um Ponto de Montagem

16 – ACESSANDO WIN9x

17 – MODO GRÁFICO (X) I

- 17.1 – Configurando o Servidor X
- 17.2 – Utilizando o Modo Gráfico (WindowMaker)
 - 17.2.1 – Configuração do WindowMaker
 - 17.2.1.1 – Criação e exclusão de ícones
 - 17.2.1.2 – Configuração do WindowMaker com WPrefs
 - 17.2.1.3 – Configuração do WindowMaker com Wmakerconf
 - 17.2.2 – XWC (Gerenciador de Arquivos)
- 17.3 – Utilizando o Modo Gráfico (KDE)

18 – INSTALAÇÃO DE PROGRAMAS

19 – ARQUIVAMENTO

20 – COMANDOS BÁSICOS VI

- 20.1 – rpm
 - 20.1.1 – Consulta
 - 20.1.2 – Instalação
 - 20.1.3 – Atualização
 - 20.1.4 – Desinstalação
- 20.2 – tar

21 – COMANDOS BÁSICOS VII

- 21.1 – ps
- 21.2 – kill

22 – MODO GRÁFICO (X) II

- 22.1 – O Configurador Linuxconf
- 22.2 – Configuração de Impressora com o Linuxconf
- 22.3 – Acesso à Internet / E-Mail
 - 22.3.1 – Configuração de Acesso Discado com o Kppp
 - 22.3.2 – Configuração do Kmail

23 – COMANDOS BÁSICOS VIII

- 23.1 – >
- 23.2 – >>
- 23.3 – &
- 23.4 – cat
- 23.5 – lpr
- 23.6 – lpq
- 23.7 – lprm
- 23.8 – lpc

Curso de Linux Básico

- 23.9 – pwd
- 23.10 – who
- 23.11 – df
- 23.12 – du

24 – STAROFFICE 5.1

- 24.1 – O que é?
- 24.2 – Instalando o StarOffice 5.1
- 24.3 – Configurando a Impressora para o StarOffice 5.1
- 24.4 – Usando o StarOffice 5.1

25 – TÓPICOS SOBRE SEGURANÇA

- 25.1 – Política de senhas
- 25.2 – Permissões de acesso
- 25.3 – Protegendo as senhas
- 25.4 – Configuração dos serviços TCP
- 25.5 – Acesso aos serviços do Linux
- 25.6 – Dicas de segurança
- 25.7 – Programas para segurança
- 25.8 – Considerações finais sobre segurança

26 – LICENÇA DE DOCUMENTAÇÃO LIVRE GNU

27 – BIBLIOGRAFIA

0 – HISTÓRICO

Versão 1.0:

- Esta versão existiu apenas para fins didáticos do próprio autor.
- Não estava sob a FDL.

Versão 2.0:

- Passou a estar sob FDL e disponível on-line.
- Alterado capítulo sobre Instalação para ficar mais geral.
- Incluída menção ao Grub como opção ao Lilo.
- Incluída menção aos novos nomes de dispositivos do kernel 2.4.
- Incluída montagem de dispositivos com parâmetro “auto” para tipo de sistema de arquivos.
- Incluída menção ao xf86cfg como opção de configuração do XFree86 versão 4.
- Incluída instrução para cópia de texto.
- Substituído o xfm pelo xwc.
- Incluída menção ao KDE2.
- Incluída configuração de impressora pelo Linuxconf em lugar do Control-Panel.
- Diversas pequenas correções de ortografia e sintaxe.

1 – INTRODUÇÃO

Este primeiro capítulo se propõe a responder perguntas básicas sobre o Linux, como sua origem, suas principais características e sua forma de distribuição aos usuários. Além disso, serão introduzidos alguns dos conceitos básicos do mundo Linux e listadas algumas das fontes de informação disponíveis sobre este sistema operacional.

1.1 – O QUE É?

O Linux é um sistema operacional originalmente desenvolvido em 1991 por Linus Torvalds, um finlandês do Departamento de Ciência da Computação da Universidade de Helsinki. A partir do seu lançamento, milhares de programadores espalhados por todo o mundo contribuíram e continuam contribuindo para o seu desenvolvimento, utilizando a Internet como a grande ferramenta que proporciona o intercâmbio de idéias e o gerenciamento de grupos de trabalho.

O Linux foi criado tendo por base o padrão POSIX, que é o mesmo que deu origem aos diversos "sabores" de UNIX. Sendo assim, podemos dizer que o Linux é um UNIX, mas não que ele é UNIX.

O Linux foi inicialmente desenvolvido para PCs baseados em CPUs x86 (como 386/486/Pentium, etc), porém atualmente existem versões para computadores Alpha da DEC, Sparcs da SUN, CPUs M68000 (semelhantes a Atari e Amiga), MIPS, PowerPCs (tais como iMac...), S/390 (IBM), I64 (Intel Itanium) e PDAs.

O sistema operacional em si é chamado kernel (núcleo) do Linux, e é o responsável por gerenciar todas as tarefas do sistema. Todos os demais serviços e programas fazem chamadas ao kernel durante sua execução.

Inicialmente o Linux suportava apenas uma interface por linha de comando, com recursos até hoje considerados poderosíssimos. Atualmente, existem servidores gráficos para Linux, como o "Servidor X", que é de livre distribuição, além de outros, comerciais. Para rodar no servidor X, existe uma infinidade de gerenciadores de janelas (window managers), com as mais diversas opções de interfaces gráficas, que imitam as interfaces do Win95, iMac, OS2, NeXT, etc, além daquelas criadas originalmente para o Linux.

Os principais usos do Linux anteriormente eram como servidor de páginas da Web, servidor FTP, servidor de e-mail, servidor de nomes (DNS) ou roteador (gateway) entre LANs e a Internet. Além destes usos, hoje, mais e mais sistemas Linux estão sendo utilizados como servidores de bancos de dados, sendo que a maioria dos mais populares pacotes de bancos de dados estão disponíveis nas versões nativas do Linux, incluindo produtos de empresas como Oracle, Informix, Sybase e IBM. Finalmente, o Linux vem sendo usado cada vez mais para estações de trabalho pessoais ou sistema desktop, utilizados para desenvolvimento de software, computação gráfica, acesso à internet, etc.

Atualmente tem sido desenvolvidos os mais variados aplicativos para a plataforma Linux, desde editores de texto simples até simuladores de circuitos eletrônicos, passando por jogos e aplicativos para internet, tornando cada vez mais popular e amigável este sistema operacional.

1.2 – CARACTERÍSTICAS

A seguir descreveremos algumas das características mais importantes do Linux, que o tornam um sistema estável, versátil e confiável.

1.2.1– Multiusuário:

Isto significa que o Linux instalado em uma máquina pode ser utilizado por mais de um usuário, seja na mesma máquina ou através de terminais remotos ligados a esta máquina. Tudo isso com privacidade, já que o acesso a cada arquivo ou diretório pode ser configurado (individualmente ou em grupos, como veremos mais a frente).

Cada usuário tem acesso ao sistema através do login (entrada no sistema), mediante o uso de senha. Por questões de segurança no Linux não é permitido (por vias normais) que alguém tenha acesso à máquina sem possuir autorização, ou seja, é necessário estar cadastrado no sistema para poder acessá-lo, ao contrário de outros sistemas.

Curso de Linux Básico

1.2.2– Multitarefa:

O Linux trabalha com multitarefa real, ou seja, ele pode gerenciar diversas tarefas sendo executadas "ao mesmo tempo" pela máquina. Para gerenciar estas tarefas, o Linux trabalha com o conceito de processos, nome dado a cada programa (ou parte dele) que está "rodando" na máquina. Cada vez que executamos um programa, serão criados um ou mais novos processos no kernel, separados dos demais, os quais serão gerenciados pelo Linux.

Quando inicializamos a máquina e o kernel do Linux está carregado, o próprio kernel já cria o primeiro processo, que é chamado *init*. Ao logarmos na máquina, o programa de login cria um processo para o usuário, que é a interface de linha de comando. A partir daí, o usuário pode criar novos processos, isto é, "rodar" programas.

É importante notar que um processo só pode ser criado a partir de outro (com exceção do primeiro). Ao processo de origem dá-se o nome de "processo-pai" e ao processo originado, chamamos "processo-filho".

Outro conceito importante é a "morte" de um processo. Um processo pode ser "morto" por outro processo, e existe um mecanismo que mantém o processo pai informado sobre seus processos-filhos que existam ou deixem de existir.

O principal aspecto que deve ser considerado, portanto, é que cada processo roda em um ambiente independente, de modo que se por algum motivo qualquer, seja necessário "matar" um processo, isto não irá afetar os demais processos (exceto os processos-filhos). Desta forma, os serviços disponibilizados pela máquina em questão continuariam sendo utilizados normalmente, sem maiores problemas.

1.2.3– Código Aberto (GPL):

O Linux não é um software de domínio público, mas é distribuído sob a GNU General Public License, que preserva a disponibilidade do seu código fonte. Ou seja, o código fonte do Linux deve estar sempre disponível para qualquer um. Alguém pode cobrar pela cópia do Linux, se desejar, desde que, com isso, não limite a distribuição do mesmo.

Por ser um software aberto, alertas para qualquer possível problema de segurança ou falha nos programas são distribuídos imediatamente pela Internet em busca de soluções, não sendo necessário esperar meses para que um fabricante ou desenvolvedor crie uma solução.

O Linux segue o modelo de desenvolvimento aberto e, por isso, cada nova versão disponibilizada ao público é considerada como um "produto de qualidade", pois qualquer um pode examinar e melhorar o código fonte. Para informar às pessoas se elas estão obtendo uma versão estável ou não, o seguinte esquema foi criado:

Versões r.x.y do kernel, onde x é um número par, são versões estáveis, e, enquanto o y é incrementado, apenas reparos de bugs são efetuados.

Versões r.x.y do kernel, onde x é um número ímpar, são versões beta destinadas apenas a desenvolvedores, podem ser instáveis e falhar, e estarão recebendo novas características o tempo todo.

De tempos em tempos, com o atual desenvolvimento do kernel sendo considerado "estável", x é mudado para um número par, e o desenvolvimento continua com uma nova versão (x ímpar).

1.2.4– Custo:

O custo do Linux varia de acordo com o que você espera obter. É possível obter o Linux pela Internet, pagando apenas o acesso (para fazer o download). Se você tiver acesso pela internet gratuita, o custo será zero!!! (se você considerar que tempo não é dinheiro...).

Se você optar por adquirir uma distribuição completa do Linux, receberá os CDs com os programas, manuais impressos e normalmente terá direito a um determinado tempo de suporte técnico gratuito.

Um ponto importante é que nestas distribuições completas incluem não só o sistema operacional, mas também uma infinidade de programas e aplicativos, inclusive pacotes "office", ferramentas gráficas, compiladores, servidores de aplicações, servidores de Internet, servidores de banco de dados, ERP, etc, tornando seu custo irrisório se comparado à quantidade de programas adquiridos.

Existem algumas "versões econômicas" de distribuições, bem mais baratas que as completas, nas quais você recebe apenas alguns CDs, não recebe os manuais impressos (ou nem todos eles) e normalmente não tem direito a suporte técnico.

Vale ressaltar que geralmente os CDs das distribuições Linux trazem os manuais em formato texto, html ou postscript, portanto, nesta situação pode ser uma boa idéia adquirir uma "versão econômica", principalmente se for para uso doméstico.

1.3 – DISTRIBUIÇÕES

Distribuição é o nome dado a um conjunto de programas constituído por um kernel do Linux e uma variedade de outros softwares para esta plataforma, chamados "pacotes", normalmente distribuídos de uma forma personalizada pela empresa ou grupo distribuidor. Existem diversas distribuições, dentre elas podemos citar:

<i>DISTRIBUIÇÃO</i>	<i>PAÍS DE ORIGEM</i>	<i>VERSÃO ATUAL</i>	<i>BASEADA EM:</i>
Slackware	Alemanha	7.1	
Red Hat	USA	7	
Debian	USA	2.2r2	
S.U.S.E	Alemanha	7.1	
Mandrake	França	8.0	Red Hat
OpenLinux (Caldera)	USA	2.4	
Corel Linux	Canadá	Second Edition	Debian
Conectiva Linux	Brasil	6.0	Red Hat

Cada distribuição tem suas particularidades, umas são mais amigáveis, outras menos, outras são mais fáceis de instalar ou atualizar, outras mais complicadas. Estas distribuições geralmente desenvolvem programas instaladores e configuradores personalizados, de forma a tornar mais fácil e amigável as tarefas no Linux, porém o núcleo do sistema (kernel) é o mesmo para todas elas. Mais adiante veremos as principais diferenças entre as principais distribuições.

1.4 – INFORMAÇÕES

1.4.1– Manuais:

Em geral todas as distribuições de Linux vêm com um ou mais manuais impressos, que trazem as informações necessárias para a instalação, configuração e operação do sistema.

Estes manuais costumam vir nos CDs, em formato texto, html, pdf ou ps (postscript), sendo sempre uma literatura quase obrigatória para o usuário do Linux.

Ao adquirir uma distribuição de Linux, é aconselhável verificar se os manuais estão em uma língua que você domina, para evitar problemas futuros.

1.4.2– HOWTO:

A cultura do Linux se utiliza bastante dos HOWTO, que são documentos que descrevem na prática "COMO FAZER" as principais atividades que um usuário, um administrador de sistemas ou redes possam vir a realizar.

Apesar da maioria dos HOWTOs originais estar em inglês, já existe atualmente um número razoável destes documentos em português.

Além dos HOWTOs, existem também os mini-HOWTOs, que apesar das semelhanças, estes últimos são mais

Curso de Linux Básico

resumidos e mais superficiais, geralmente utilizam exemplos ou casos individuais, gerando assim verdadeiras "receitas de bolo" para quem delas precisar.

Os HOWTOs e os mini-HOWTOs ficam disponíveis na Internet, sendo que de maneira geral são também incluídos nos CDs das diversas distribuições.

1.4.3- Livros:

Existem diversos livros sobre o Linux sendo editados hoje em dia, devido a sua popularização, o que facilita o acesso das pessoas às informações básicas sobre este sistema.

Porém os livros sobre Linux atualmente estão voltados, na sua maioria, para o público desenvolvedor, apesar de já existirem alguns do tipo "Linux Passo-a-Passo".

Bons livros que foram inclusive traduzidos para o português e também estão disponíveis na Internet, são os guias: "Guia do Usuário Linux", "Guia do Administrador de Sistemas" e "Guia do Administrador de Redes".

1.4.4- Listas de Discussão:

Nas listas de discussão sobre Linux, você tem acesso a uma fonte de informações muitíssimo importante, pois neste ambiente há uma intensa troca de idéias e experiências sobre o dia a dia de quem trabalha ou se diverte usando Linux.

No Brasil temos boas listas de discussão sobre Linux, dentre elas, a da Conectiva (linux-br) e da Unicamp. Existem ainda as lista de discussão da Usernet (newsgroups), tipo comp.os.linux, porém a maioria é em inglês. O volume de mensagens nestas lista é em torno de 100 a 200 ou mais mensagens por dia!!!

1.4.5- Sites na Internet:

Existem inúmeros sites na Internet sobre o Linux, onde podem ser encontradas informações sobre este sistema operacional. Vamos relacionar alguns deles:

<http://www.conectiva.com.br>

<http://linusp.com.br>

<http://www.caldera.com>

<http://www.slackware.com>

<http://www.suse.com>

<http://www.redhat.com>

<http://www.debian.org>

<http://www.linux-mandrake.com>

<http://www.linux.org>

<http://www.windowmaker.com.br>

<http://www.linuxing.org>

<http://pontobr.org>

<http://www.revistalinux.com.br>

<http://www.revistadolinux.com.br>

<http://www.linux.trix.net>

<http://www.planetarium.com.br>

<http://www.tlm.com.br>

1.4.6– Revistas:

Atualmente diversas revistas têm publicado matérias sobre o Linux, inclusive distribuindo CDs com versões deste sistema. A mais direcionada delas é a Revista do Linux, que é editada pela Conectiva, distribuidora de Linux brasileira. Esta revista traz bastante informação e dicas para o usuário de Linux. Além dela podemos citar também a PCMaster, com boas matérias sobre Linux.

2 – INSTALAÇÃO

Neste capítulo serão tratados os pontos básicos do processo de instalação do Linux em máquinas baseadas em CPUs x86 (plataforma PC).

Serão discutidas as ações pré-instalação e os conceitos envolvidos na instalação de um sistema Linux.

2.1 – PRECAUÇÕES ANTES DA INSTALAÇÃO

O Linux pode ser instalado em um HD separado ou pode compartilhar um HD com outros sistemas operacionais (SO). A quantidade de espaço (no HD) para o Linux depende do modo de instalação escolhido, sendo que este assunto será abordado mais adiante, quando serão discutidas as formas de instalação.

É importante salientar que o Linux tem uma convivência amigável com outros sistemas operacionais, apesar do contrário nem sempre ser verdadeiro. Portanto, é comum termos o Linux instalado em uma máquina ou sistema, juntamente com DOS, Windows, OS/2, Novell Netware, etc.

Assim sendo, antes de instalar o Linux, a primeira coisa com que devemos nos ocupar é com relação à preservação dos dados de outros sistemas operacionais que porventura estejam instalados na mesma máquina. Ou seja, devemos definir onde o Linux será instalado na máquina e a necessidade de realizar backup de alguma informação, na hipótese de reparticionamento de HDs.

No caso em que se dispõe de um HD exclusivo para o Linux, o backup dos dados que possam estar em outros HDs na mesma máquina não é obrigatório, porém o bom senso nos diz que esta é uma boa medida preventiva para evitar dores de cabeça...

Caso o Linux vá compartilhar o HD com outro SO, é preciso criar o espaço necessário (partição) para ele, se ainda não existir, criando pelo menos duas partições, sendo uma do tipo Linux Nativa e outra de Swap (arquivos de troca). Para criarmos estas partições temos três alternativas:

a) No HD existe espaço não particionado

Neste caso, o espaço não atribuído a nenhuma partição será utilizado para criar as partições do Linux. Por exemplo, num HD de 8.4GB em que se tem apenas uma partição de 4.2GB com Windows instalado, os 4.2GB restantes (não particionados) poderão ser usados pelo Linux.

b) No HD existe uma ou mais partições sem uso

Se pelo menos uma das partições do HD estiver sem uso ou puder ter seus dados transferidos para outras partições (de modo a torná-la não usada), esta partição sem uso poderá ser excluída para dar lugar às partições do Linux.

c) Existe espaço livre numa partição já utilizada

Para conseguirmos as partições necessárias ao Linux nestas condições, temos duas opções:

– Reparticionamento não-destrutivo: Isto é feito utilizando softwares específicos que conseguem alterar a tabela de alocação de arquivos (FAT) do HD, diminuindo o tamanho da partição e criando uma nova partição apenas com espaço livre.

Antes de usar este método é recomendado fazer BACKUP de todos os dados importantes que houverem no HD, em seguida desfragmentar o HD, de modo que todos os dados fiquem contidos (compactados) em uma área restrita do HD e utilizar o software específico para "encolher" a partição existente e criar a nova partição. Se a nova partição criada não for do tipo apropriada para Linux, deve-se excluí-la e em seguida recriá-la de modo que possa ser utilizada pelo Linux.

Um software para reparticionamento não-destrutivo normalmente encontrado nas distribuições de Linux é o fips.

Curso de Linux Básico

OBS.: Pode-se encontrar problemas se, mesmo após a desfragmentação do HD, alguns dados não tiverem sido movidos, permanecendo em áreas que reduzam o tamanho da nova partição ou até mesmo impeçam o reparticionamento não-destrutivo.

– Reparticionamento destrutivo: Este é o processo mais frequentemente utilizado, sendo um dos mais radicais, pois é preciso excluir a partição existente e criar as novas partições.

Portanto, antes de excluir a partição existente, é necessário fazer BACKUP de todos os dados contidos no HD e verificar se todos os originais dos programas instalados estão disponíveis, já que deverão ser reinstalados posteriormente, incluindo o sistema operacional.

Deve-se então excluir a partição existente e criar as novas partições, usando utilitários como o fdisk do DOS ou o fdisk do Linux para este fim. Observe que apesar dos nomes iguais, estes dois utilitários são bastante diferentes, sendo que o primeiro roda em DOS e o segundo em Linux. Todo o cuidado é pouco ao usar estes programas, pois o efeito de uma ação impensada pode ser uma enorme dor de cabeça!!!

Além destes dois programas, atualmente as distribuições têm trazido diversas outras opções, inclusive com interfaces gráficas, para o gerenciamento das partições de HDs. Por exemplo: Disk Druid (Conectiva Linux), DiskDrake (Mandrake), entre outros.

Nota: Atualmente, além das opções de instalar o Linux em partições próprias (nativas), diversas distribuições já possuem o recurso de se instalar o Linux em uma partição DOS. Neste caso é criada uma pasta (diretório) dentro do Windows e esta pasta conterá todo o sistema de arquivos do Linux. Desta forma, portanto, não é necessário reparticionar o HD, porém, é esperada uma queda na performance devido à emulação do sistema de arquivos.

Além das precauções discutidas acima, é igualmente importante dispormos das informações sobre o Hardware da máquina em que o Linux será instalado (placas de vídeo, rede, impressora, teclado, mouse, monitor, etc), caso sejam solicitadas durante a instalação / configuração do sistema. Se a máquina tiver Windows instalado, estas informações podem ser obtidas no Painel de Controle, acessando o ícone Sistema.

2.2 – FORMAS DE INSTALAÇÃO

Geralmente as distribuições de Linux costumam disponibilizar formas de instalação pré-definidas, com algumas particularidades e recursos a mais ou a menos em relação às demais.

Além destas, normalmente é disponibilizada uma instalação personalizada, que enfatiza as necessidades específicas do usuário, proporcionando muita flexibilidade. Pode-se ter completo controle sobre os pacotes que serão instalados no sistema, assim como determinar se será ou não usada dupla inicialização.

Esta forma de instalação é recomendada para quem já tem familiaridade com o Linux e com a manipulação de partições. Dependendo de como será feita a seleção dos pacotes a serem instalados, esta forma de instalação pode ser bem mais demorada que as demais.

2.3 – MEIOS DE INSTALAÇÃO

Resumidamente os seguintes meios podem ser utilizados na instalação do Linux:

- CDROM Local: Se você tem um drive de CDROM e o CD de uma distribuição do Linux.
- Disco Rígido Local: Somente se os arquivos do Linux tenham sido copiados para o disco rígido.
- NFS (via Rede): Se a instalação for efetuada pela rede, será necessário montar o CD do Linux em uma máquina que suporte o padrão ISO-9660 para sistemas de arquivos com extensões Rock Ridge. Esse equipamento deverá suportar ainda NFS. O CDROM deverá ser exportado através do NFS, assim como será necessário conhecer o endereço IP e o caminho do CDROM ou ter o servidor de nomes configurado. Este método requer um disquete extra para suporte à inicialização via rede.
- FTP: Este método requer um disquete extra para suporte à inicialização via rede, o nome ou o endereço IP do

Curso de Linux Básico

servidor FTP a ser utilizado e o diretório onde residem os arquivos da distribuição do Linux.

– HTTP: Este método requer um disquete extra para suporte à inicialização via rede, o nome ou o endereço IP do servidor HTTP a ser utilizado e o diretório onde residem os arquivos da distribuição do Linux.

A maioria dos CDs de Linux já são inicializáveis, porém pode ser que a máquina não aceite o boot pelo CDROM por limitação da BIOS e então será necessário criar um disquete de boot, cuja imagem é normalmente distribuída com o Linux. As distribuições de Linux geralmente incluem um disquete de boot em seus pacotes.

Este disquete poderá ser de grande utilidade caso ocorra alguma falha no sistema de dupla inicialização e não seja possível o boot normal pelo HD. Isto é muito comum quando se tem um HD compartilhado entre Windows e Linux e é feita a reinstalação do Windows, pois o mesmo remove a dupla inicialização do MBR (Master Boot Record, ou Registro Mestre de Inicialização).

Para a criação deste disquete, tradicionalmente é usado o utilitário rawrite, que deve ser executado no DOS.

Inicialmente etiquete um disco formatado de 3 ½ polegadas com o nome de disco de inicialização local ou algo similar e insira na unidade de disco flexível. Após, execute os seguintes comandos (presumindo que o seu CD seja o drive d:):

```
C:> d:
D:> cd dosutils
D:dosutils>rawrite.exe
Enter disk image source file name: ..imagens\boot.img
Enter target diskette drive: a:
Please insert a formatted diskette into drive A: and press <ENTER>
```

```
D:dosutils>
```

O utilitário inicialmente solicitará o nome do arquivo do disco imagem, (informar por exemplo boot.img). Após solicitará o dispositivo de gravação, onde deverá ser informado a:. Para gerar um disco adicional, etiquete um segundo disco e execute o rawrite novamente, informando o nome do arquivo imagem desejado.

2.4 – INSTALAÇÃO

Para iniciar a instalação do Linux, insira o disquete de inicialização (ou o CD-ROM, caso a BIOS aceite inicialização do sistema via CD-ROM), no drive, reinicialize o computador e siga os passos do programa de instalação.

A seguir serão listadas resumidamente as partes da instalação do Linux:

- Seleção do idioma;
- Seleção do layout do teclado;
- Configuração do Mouse;
- Configuração da Rede;
- Configuração do Fuso Horário;
- Configuração de Impressora;
- Definição da Senha do Superusuário (root);
- Criação do Disco de Inicialização;
- Configuração do Modo Gráfico:
 - Placa de vídeo;
 - Monitor de vídeo;
 - No. de cores / Resolução;
 - Inicialização Gráfica.

3 – PRIMEIRA UTILIZAÇÃO

Neste capítulo trataremos alguns conceitos básicos antes de iniciarmos o uso do Linux, como o gerenciamento de boot, a entrada e saída do sistema e a utilização de consoles.

3.1 – CONCEITOS BÁSICOS

3.1.1– Carregamento do Sistema (lilo):

Como já foi citado anteriormente, o Linux pode conviver amigavelmente com outros sistemas operacionais instalados numa mesma máquina. Para permitir ao usuário escolher qual sistema operacional será usado a cada vez que a máquina é ligada, o Linux possui gerenciadores de boot.

O gerenciador de boot mais tradicionalmente utilizado no Linux é o LILO (Linux Loader), cujas características veremos brevemente nesta seção. Versões mais atualizadas utilizam o Grub como gerenciador de boot.

Normalmente é recomendado que o gerenciador de boot seja carregado no MBR (Master Boot Record ou Registro Mestre de Inicialização) do HD, pois este é o primeiro dado a ser lido pelo BIOS (Basic I/O System) da máquina. Ele portanto informará ao BIOS qual o próximo endereço a ser lido no HD, que será aquele correspondente ao sistema operacional selecionado pelo usuário. Esta opção é recomendada quando não há outros gerenciadores de boot, como o Boot Manager do OS/2 ou System Commander, já instalados no MBR.

Se no MBR já houver um gerenciador de boot instalado, existe a opção de instalação do mesmo no setor de boot de uma partição marcada como "ativa" (ou seja, capaz de dar boot), pois se no MBR não houver nenhum redirecionamento para algum sistema operacional, o próximo setor a ser lido pelo BIOS é o setor de boot de uma partição "ativa". Da mesma forma, o gerenciador de boot redirecionará a leitura para o sistema operacional escolhido pelo usuário.

O LILO está sujeito a algumas limitações impostas pelo BIOS. Geralmente, a maioria dos BIOS não pode acessar mais de dois discos rígidos e eles não podem acessar qualquer dado armazenado além do cilindro 1023 de qualquer dispositivo. Note que alguns BIOS novos não têm estas limitações, mas isto não é universal.

O Grub por sua vez não possui esta limitação, além da característica de ser executado em modo gráfico.

Após instalarmos o Linux, o gerenciador de boot assumirá este sistema operacional como o default, sendo que esta opção pode ser alterada posteriormente.

Portanto, após ligada a máquina e o BIOS ser carregado, se o LILO estiver instalado será apresentada a seguinte mensagem:

lilo boot:

Se pressionarmos a tecla Tab, serão mostradas no vídeo as opções de inicialização (boot) do sistema. Podemos digitar qual a opção desejada ou aguardarmos o tempo definido para que o LILO carregue o sistema operacional definido como padrão.

Caso o gerenciador de boot instalado seja o Grub, aparecerá sua tela gráfica, com as opções disponíveis. Também possui um default que é executado após um tempo predefinido.

Após isto, o sistema operacional escolhido será carregado na máquina.

3.1.2– Entrada no Sistema (login):

Após o sistema operacional Linux ter sido carregado, o primeiro processo (para o usuário, pois diversos outros serviços já estarão inicializados, como impressão, e-mail, etc) estará rodando na máquina, que é o getty. Ele fornece a tela de login e passa a informação digitada ao sistema para autenticação do usuário, que é feita pelo

Curso de Linux Básico

programa login.

Como já mencionamos anteriormente, o Linux só permite acesso do usuário mediante sua identificação ao sistema e a informação de sua respectiva senha.

Após a inicialização aparecerá no vídeo algo como:

Conectiva Linux Versão 4.0
Kernel 2.2.12-5cl

localhost login: _

A primeira vez que se acessa o Conectiva Linux, o acesso deverá ser realizado com o superusuário root. Este é o nome da conta que tem acesso completo a todos os componentes do sistema.

Normalmente, a conta de superusuário é somente utilizada na execução de tarefas de administração do sistema, como a criação de novas contas, desligar o sistema, etc. Isso se deve ao fato de que o acesso irrestrito do superusuário quando mal utilizado poderá provocar grandes estragos ao sistema.

Então seja cuidadoso ao acessar o sistema como root e use a conta de superusuário somente quando realmente for necessário.

Para o acesso inicial, informe root na linha de comando login: e pressione "Enter". Aparecerá uma linha de comando Password: , como abaixo:

Conectiva Linux Versão 4.0
Kernel 2.2.12-5cl

localhost login: root
Password: _

Digite a mesma senha criada durante a instalação, pressionando "Enter" ao terminar. Deverá então surgir algo como:

[root@localhost /root]#

Chegamos portanto ao bash (Bourne Again Shell), o shell (interpretador de comandos) padrão do Linux. Já podemos então utilizar a máquina digitando comandos que serão interpretados pelo bash e passados ao kernel para execução.

3.1.3- Saída do Sistema (logout ou exit):

A saída do sistema pode ser realizada de duas formas, usando o comando **exit** ou o comando **logout**. A diferença entre os dois é que **exit** encerra o shell de comandos corrente e **logout** encerra a sessão.

Vale lembrar que no Conectiva Linux existe já definido um atalho para o comando **logout**, bastando pressionar **Ctrl + d**.

3.1.4 – Encerramento do Sistema (shutdown):

Se o sistema deve ser desligado ou reinicializado, devemos utilizar o comando **shutdown** para fazer isto. Este comando se encarrega dos detalhes do desligamento, de modo que tudo ocorra em ordem, sem danos ao sistema. Ele pode inclusive avisar aos demais usuários com antecedência de que o sistema será paralisado e toma automaticamente providências para a finalização.

Curso de Linux Básico

Sintaxe do comando **shutdown**:

```
shutdown [-t segundos] [-rkhncf] tempo [mensagem de alerta]
```

Opções:

- k Não desliga realmente o sistema, somente envia mensagens de aviso a todos os usuários.
- r Reinicializa após o desligamento do sistema.
- h Desliga o sistema após a execução do comando.
- c Cancela a execução de um programa shutdown. Não necessita do argumento tempo.

tempo Quanto o sistema deverá ser executado, antes da ação do comando shutdown.

msg-de-aviso

Mensagem a ser enviada a todos os usuários.

OBS.: O argumento tempo pode ter diferentes formatos. Primeiro, ele pode ser informado em um formato absoluto no formato hh:mm, na qual hh é a hora (com 1 ou 2 dígitos) e mm são os minutos da hora (com dois dígitos). O segundo formato tem o formato +m, no qual m é o número de minutos a serem aguardados. A palavra **now** é um nome alternativo para +0.

Exemplo:

```
shutdown -f -h +2 "Falha na energia elétrica; Sistema sendo desligado"  
shutdown -c "Energia elétrica restaurada; Desligamento Cancelado"
```

Vale lembrar que no Conectiva Linux 4.0 existe um atalho para reinicializar o sistema, bastando pressionar Ctrl + Alt + Del e será executado o comando **shutdown -t3 -r now**.

3.1.5 – Alternando Entre Consoles:

Os sistemas Linux permitem que se trabalhe com mais de um console na mesma máquina. No Conectiva Linux, por default temos acesso a seis consoles, além de mais seis sessões do X Window, sendo que estes parâmetros podem ser alterados no sistema.

Ao inicializar a máquina e logar, normalmente utilizamos (por default) o primeiro console. Se quisermos alternar para o segundo console, basta pressionar **Alt** juntamente com uma das seis primeiras teclas de função (F1 a F6).

Portanto, Alt + F1 corresponde ao primeiro console, Alt + F2 ao segundo, até Alt + F6 que corresponde ao sexto console. As combinações de Alt + F7 a Alt + F12 são reservadas para alternar entre sessões do X Window (interface gráfica), como veremos mais adiante.

4 – COMANDOS BÁSICOS I

Neste capítulo, iniciaremos o contato com os comandos mais comuns e úteis do Linux. Nem todas as opções sobre cada comando serão descritas aqui, apenas as de uso mais prático para a maioria dos usuários.

Para conhecer o conjunto completo das opções de um determinado comando, o usuário é encorajado a buscar estas informações nas páginas de manual (man pages), conforme será descrito a seguir.

Os comandos serão tratados em diversos capítulos, para conciliar a introdução de novos conceitos e a sua aplicação na prática pelo usuário.

4.1 – ls

O comando **ls** lista o conteúdo de um diretório. Quando usado sem opções, lista todos os arquivos não ocultos do diretório, em ordem alfabética, preenchendo tantas colunas quantas couber na tela.

Opções:

- a Lista todos os arquivos presentes nos diretórios, inclusive os ocultos.
- k Caso o tamanho do arquivo seja listado, mostra-o em Kbytes.
- l Além do nome de cada arquivo, lista o tipo, permissões, número de ligações diretas, nome do dono, nome do grupo, tamanho em bytes e data (da modificação, a menos que outra data seja selecionada). Para arquivos com uma data anterior a 6 meses ou com mais de 1 hora no futuro, a data conterà o ano ao invés da hora e dia.
- t Ordena o conteúdo dos diretórios pela data ao invés da ordem alfabética, com os arquivos mais recentes listados no início.
- u Ordena o conteúdo dos diretórios de acordo com a data de último acesso ao invés da data de modificação. No formato longo de listagem, apresenta a data de último acesso ao invés da data de modificação.
- R Lista o conteúdo de todos os diretórios recursivamente.
- X Ordena o conteúdo dos diretórios alfabeticamente pelo nome da extensão (caracteres após o último '.'). Arquivos sem extensão são listados no início.
- 1 Lista um arquivo por linha.
- color
Colore os nomes dos arquivos dependendo do tipo.

4.2 – | (pipe)

O pipe (ou duto) é utilizado como conexão de utilitários. É uma maneira de redirecionar as entradas e saídas, de modo que a saída de um comando torna-se a entrada do comando seguinte. Pode-se usar vários pipes em uma mesma linha de comando, de maneira que é possível combinar tantos comandos quantos forem necessários.

Veremos exemplos de aplicação do pipe logo em seguida, depois de conhecermos mais alguns outros comandos do Linux.

4.3 – more

O comando **more** consiste de um filtro para uso na visualização de arquivos em terminais. Este comando só pode pagnar o texto para frente (do início para o fim).

Curso de Linux Básico

Opções:

`-d` O `more` irá solicitar instruções ao usuário através da mensagem "[Pressione espaço para continuar, 'q' para finalizar.]" e irá apresentar a mensagem "[Pressione 'h' para instruções.]" ao invés de emitir sinal sonoro quando uma tecla ilegal for pressionada.

Exemplos:

```
more .Xdefaults
more -d .Xdefaults
ls -al --color | more -d
```

4.4 – less

O comando **less** é um comando similar ao **more**, porém ele permite pagnar para frente e para trás no texto.

Para movimentar dentro do texto, utilize as teclas Page Up, Page Down, Home, End e as setas de direção. Se desejar recorrer à ajuda dentro do **less**, basta pressionar a tecla **h**. Para sair, pressione a tecla **q**.

Exemplos:

```
less .Xdefaults
ls -al | less
```

4.5 – cd

O comando **cd** (abreviatura de change directory) é utilizado para mudar o diretório corrente. Permite mudar do diretório atual para outro especificado pelo usuário. Se for usado sem argumentos, muda para o diretório pessoal do usuário. A opção "`cd -`" volta ao diretório anterior, enquanto a opção "`cd ..`" sobe um nível no sistema de arquivos (árvore de diretórios).

O argumento do comando, ou seja, a especificação do diretório para onde se quer mudar, pode ser relativo à posição em que se está ou baseado no diretório raiz (/). Neste último caso, o nome do diretório desejado deve ser precedido de uma "/".

Exemplos:

```
cd /etc/X11      (especificação baseada no diretório raiz)
cd ..
cd X11          (especificação relativa à posição em que se está [/etc])
cd -
cd
```

4.6 – COMANDOS DE DATA E HORA

O comando **date** é utilizado para mostrar a data e a hora do sistema. Exibe a data e hora corrente, desde que aplicado sem parâmetros. Somente o superusuário pode, através do uso de parâmetros associados ao comando **date**, alterar a data e hora do sistema.

Se for dado algum argumento que não comece com '+', o comando **date** acerta o relógio do sistema com o tempo e data especificados pelo argumento. O argumento deve consistir somente de dígitos, que tenham o seguinte significado:

MM	mês;
DD	dia do mês;
hh	hora;
mm	minuto;
CC	primeiros dois dígitos do ano (opcional);
YY	últimos dois dígitos do ano (opcional);
ss	segundos (opcional).

Exemplos:

```
date
date 05221600200015
date '+Hoje é %A, dia %d de %B'
date '+Agora são %T'
```

4.7 – man

O comando **man** é utilizado para formatar e exibir as páginas de manual on–line, que são textos descrevendo em detalhes como usar um comando especificado.

Exemplos:

```
man ls
man more
```

5 – EDITANDO TEXTOS (pico)

Neste capítulo conheceremos um editor de textos básico do Linux, o pico. Ele é um dos mais simples editores, porém satisfaz aos iniciantes justamente pela sua facilidade de uso, sendo ideal para pequenas edições em arquivos de configuração, pequenos textos, etc.

Vale citar que neste campo existem dois poderosíssimos editores rivais: o Emacs e o VI, sendo que ambos possuem versões em modo texto e com front-end gráfico, para serem utilizados sob o X Window. Apesar de toda uma rivalidade entre os simpatizantes de cada um destes dois programas, ambos são excelentes editores, sendo bastante avançados em relação ao pico e aos demais editores encontrados no ambiente Linux.

Sintaxe:

pico [opções] [nome_do_arquivo]

Principais opções:

+n

Faz o editor ser iniciado com o cursor localizado *n* linhas dentro do arquivo.

-b

Habilita a opção de substituir textos encontrados pelo comando Ctrl+W (Where is).

-e

Habilita a função de auto-completar nomes de arquivos no manipulador de arquivos.

-j

Habilita o comando "Goto" no manipulador de arquivos.

-k

Faz com que o comando Ctrl+K (Cut Text) remova caracteres da posição do cursor até o fim da linha, ao invés de remover a linha inteira.

-m

Habilita as funcionalidades do mouse. Isto funciona apenas quando o pico é executado em um terminal dentro do X Window.

-rn

Especifica a coluna usada para limitar os comandos de justificação (margem direita).

-s *speller*

Especifica um programa alternativo de correção ortográfica, quando for utilizado corretor ortográfico.

-v

Apenas permite visualizar o arquivo, desabilitando qualquer edição do mesmo.

-w

Desabilita quebra de linha (desta forma, permite edição de linhas longas).

-x

Desabilita o menu das teclas de comando no rodapé da tela.

Descrição das funcionalidades do **pico**:

Os comandos são mostrados no rodapé da tela e uma ajuda sensível ao contexto é oferecida. Assim que os caracteres são digitados eles são imediatamente inseridos no texto.

Os comandos de edição são entrados usando combinações de teclas com a tecla CONTROL (Ctrl). O editor tem cinco funções básicas: justificação de parágrafos, pesquisa, cortar / colar blocos, um corretor ortográfico e um manipulador de arquivos.

A justificação de parágrafos utiliza o comando Ctrl+J e ocorre no parágrafo onde está o cursor ou no parágrafo imediatamente abaixo, se o cursor está entre as linhas. Os parágrafos são delimitados por linhas em branco ou por linhas começando com um espaço ou uma tabulação (Tab). O comando de justificação pode ser desfeito imediatamente após a sua execução, utilizando a combinação das teclas Ctrl+U.

As pesquisas de strings são executadas por meio do comando Ctrl+W e não são sensíveis à maiúsculas / minúsculas. A pesquisa é iniciada na posição corrente do cursor e vai até o final do texto. A string mais recentemente pesquisada é oferecida como default na pesquisa seguinte.

Blocos de texto podem ser movidos, copiados ou deletados com o uso criativo dos comandos Ctrl+^ (para marcar), Ctrl+K (para deletar) e Ctrl+U (para desfazer). O comando Ctrl+K removerá o texto que se encontra entre a "marca" feita e a posição corrente do cursor e o colocará no buffer. O comando Ctrl+U faz uma "colagem" do texto do buffer a partir da posição corrente do cursor.

O corretor ortográfico examina todas as palavras do texto. Ele então mostra cada uma das palavras para correção, enquanto a destaca no texto. A correção ortográfica pode ser cancelada a qualquer momento. Alternativamente, o editor poderá substituir a rotina de correção ortográfica padrão por uma outra, definida pela variável de ambiente SPELL.

O manipulador de arquivos é oferecido como uma opção nos prompts dos comandos Ctrl+R (Read File) e Ctrl+O (Write Out). Ele é destinado a ajudar na busca por arquivos específicos e na navegação de hierarquias de diretórios. Nomes de arquivos com seus tamanhos e nomes de diretórios no diretório de trabalho corrente são apresentados para seleção. O diretório de trabalho corrente é mostrado na linha mais acima na tela, enquanto a lista de comandos disponíveis aparece nas duas linhas de baixo. Algumas funções básicas para manipulação de arquivos são suportadas: renomear, copiar e deletar.

Mais ajuda específica está disponível no "Help OnLine" do pico (em inglês), que pode ser acessada pelo comando Ctrl+G. Para sair do Help, deve ser utilizado o comando Ctrl+X.

Para finalizar o **pico**, é utilizado o comando Ctrl+X, o qual apresentará uma mensagem perguntando se deseja salvar as modificações feitas e ainda não salvas, caso o arquivo sendo editado tenha sido alterado. Caso seja respondido "N", as alterações feitas serão perdidas (não serão salvas).

Quando o pico está sendo executado e é desconectado, ele salvará o trabalho corrente (caso necessário) antes de ser encerrado. O trabalho será salvo com o nome corrente do arquivo, sendo acrescentado .save a este nome. Se o trabalho corrente não possuir nome, ele será salvo como "pico.save".

A maneira que as linhas mais longas que a largura da tela são repartidas não é imediatamente óbvia. Linhas que continuam além da largura da tela são indicadas por um caracter "\$" ao fim da linha. Linhas longas são roladas horizontalmente assim que o cursor se move através delas.

Exemplos:

```
pico nome_do_arquivo  
pico +5 nome_do_arquivo  
pico -b -k nome_do_arquivo  
pico -v nome_do_arquivo  
pico -wx nome_do_arquivo
```

6 – COMANDOS BÁSICOS II

Neste capítulo veremos mais alguns dos principais comandos do Linux, utilizados para criar diretórios, remover, copiar, mover e procurar arquivos.

6.1 – mkdir

Este comando é utilizado para a criação de diretórios. Sua sintaxe é:

```
mkdir [opções] <caminho>
```

As opções mais utilizadas são:

- p Cria todos os diretórios especificados no caminho;
- m Especifica as permissões de acesso do novo diretório.

Exemplos:

```
mkdir meu_diretorio  
mkdir -p um dois tres
```

6.2 – rm

Este comando é utilizado para remover arquivos. Pode remover também diretórios.

Sintaxe:

```
rm [opções] <arquivos>
```

Opções mais utilizadas:

- f Não solicita confirmação.
- i Solicita confirmação. (Caso sejam informados -f e -i, somente o último terá efeito).
- r Remove as árvores de diretórios recursivamente.
- R Remove as árvores de diretórios recursivamente.

Exemplos:

```
rm meu_diretorio  
rm -ir meu_diretorio  
rm -rf um dois tres
```

6.3 – cp

Comando utilizado para copiar arquivos e diretórios. Pode-se copiar um arquivo para um destino informado, ou copiar arbitrariamente muitos arquivos para o diretório de destino.

Sintaxe:

```
cp [opções] <origem> <destino>
```

Principais opções:

- f Remove um arquivo de destino já existente.
- i Pergunta se deve regravar arquivos já existentes.
- R Copia diretórios recursivamente, preservando arquivos que não sejam diretórios.

Caso o último argumento denomine um diretório existente, este comando copiará cada arquivo de destino naquele diretório (mantendo o mesmo nome). Caso dois arquivos sejam informados, ele copiará o primeiro no segundo.

Exemplos:

```
cp /etc/fstab fstab
cp -R /etc/rc.d /root/etc
```

6.4 – mv

Comando utilizado para mover e renomear arquivos e diretórios.

Sintaxe:

```
mv [opções] <origem> <destino>
```

Principais opções:

- f Remove os arquivos de destino, sem solicitar a confirmação pelo usuário.
- i Solicita confirmação para sobrescrever arquivos de destino.
- v Lista o nome de cada arquivo antes de removê-lo.

Caso o último argumento seja o nome de um diretório existente, este comando moverá cada arquivo informado para o diretório, mantendo o nome original. Por outro lado, caso somente dois arquivos sejam informados, altera o nome do primeiro para o segundo.

Exemplos:

```
mv /etc/fstab /etc/fstab.bak
mv -v /etc/fstab.bak /etc/fstab
```

6.5 – find

Comando utilizado para pesquisar arquivos em uma hierarquia de diretórios.

O comando **find** pesquisa a árvore de diretórios com raiz dada pelo nome do arquivo fornecido para avaliação, através de uma expressão avaliada da esquerda para a direita, de acordo com as regras de precedência, até que o resultado seja conhecido. Neste ponto **find** vai para o próximo nome de arquivo.

Sintaxe:

Curso de Linux Básico

`find [caminho] [expressão]`

Algumas opções:

`-name pattern` O nome do arquivo (o caminho à frente do nome do arquivo não é considerado) deve coincidir com os padrões informados em `pattern`. Os metacaracteres ('*', '?', e '[']) não combinam com um '.' no início do nome do arquivo.

`-iname pattern` Como `-name`, mas o teste de padrão não é sensível a maiúsculas e minúsculas. Por exemplo, os padrões 'fo*' e 'F??' coincidem com os nomes de arquivos 'Foo', 'FOO', 'foo', 'fOo', etc.

O primeiro argumento que começar com '-', '(', ')', ';', ou '!' é colocado no início da expressão. Quaisquer argumentos antes disso são caminhos para pesquisa e quaisquer argumentos após constituem o restante da expressão. Caso nenhum argumento seja fornecido, o diretório atual será utilizado.

Exemplos:

```
find -name login
find /bin -name login
find -iname Netscape
find -iname *Navigator
```

7 – TIPOS DE ARQUIVOS

Neste capítulo conheceremos os tipos de arquivos permitidos pelo Linux e suas características.

Os arquivos no Linux podem ter nomes com até 255 caracteres e múltiplas extensões (partes separadas por um ponto "."), sendo que não existem padrões de extensão que forcem o arquivo a ser de um determinado tipo (como ".exe", ".com" e ".bak" no DOS).

Alguns caracteres não devem ser utilizados em nomes de arquivos, como "!", "*", "\$" e "&". Espaços em branco são permitidos, porém não recomendáveis. Ao utilizar nome de arquivo que contenha espaços, o mesmo deve ser digitado entre aspas duplas: "nome do arquivo", por exemplo.

Além disto, devemos estar atentos para o fato de que o Linux é "case sensitive", ou seja, faz distinção entre maiúsculas e minúsculas. Por exemplo, "nome_do_arquivo" é diferente de "Nome_do_arquivo", que por sua vez é diferente de "Nome_do_Arquivo".

Basicamente, o Linux suporta quatro tipos de arquivos: regulares, de diretório, especiais de caracteres e especiais blocados. A seguir, veremos as particularidades de cada um destes tipos.

Os arquivos regulares são aqueles que contêm informações de usuários, por exemplo, tipo ASCII.

Diretórios são arquivos usados na manutenção do sistema de arquivos.

Arquivos especiais de caracteres estão diretamente ligados à entrada / saída e são usados para dispositivos seriais de entrada / saída, tais como terminais, impressoras e placas de rede.

Os arquivos especiais blocados são usados para modelar dispositivos.

Alguns arquivos, apesar de se incluírem num destes tipos, possuem características particulares que os tornam um pouco diferentes, por isto vamos comentá-los aqui:

- ✓ Arquivos de backup: terminam com o caracter "~";
- ✓ Links: São ponteiros para outro arquivo (ou diretório). Muito semelhantes aos "atalhos" do Windows.
- ✓ Arquivos ocultos: são aqueles que têm nomes iniciados por um ponto (".").

Ao executarmos o comando `ls -F`, um caracter será adicionado ao final do nome de cada arquivo (exceto para arquivos comuns), indicando o seu tipo, conforme abaixo:

- '*' Para arquivos comuns que sejam executáveis;
- '/' Para diretórios;
- '@' Para ligações simbólicas (links);
- '|' Para FIFOs;
- '=' Para sockets.

7.1 – Links

Há dois conceitos de 'links' no Linux, normalmente chamados link direto (hard link) e link simbólico (soft link). Um link direto é somente um nome para um arquivo (e um arquivo pode ter diversos nomes).

Ele será removido do disco quando o último arquivo for removido. Não há algo como um nome original, ou seja, todos os nomes têm o mesmo status.

Normalmente, mas não necessariamente, todos os nomes do arquivo são encontrados no mesmo sistema de arquivos em que o arquivo está.

Já um link simbólico ou soft link, é uma entidade totalmente diferente: é um pequeno arquivo especial que contém um caminho.

Além disto, os links simbólicos podem apontar para arquivos em diferentes sistemas de arquivo (possivelmente arquivos NFS montados a partir de diferentes máquinas), e não necessitam apontar para arquivos realmente existentes.

7.2 – Metacaracteres

Metacaracteres são caracteres que representam o nome de um grupo de arquivos. Vejamos os exemplos a seguir:

```
# ls
doc1  doc2  sessao1  sessao2  sessao3
```

Asterisco (“*”): Substitui por 0 ou mais caracteres quaisquer.

```
ls se*
sessao1  sessao2  sessao3
```

```
# ls *1
doc1 sessao1
```

Intervalo de caracteres (“[]”)

```
# ls sessao[12]          lista arquivos terminados por 1 e 2
sessao1  sessao2
```

```
# ls sessao[1-9]        lista arquivos terminados por 1 até 9
sessao1  sessao2  sessao3
```

Interrogação (“?”): Substitui por um caractere qualquer.

```
# ls doc?
doc1  doc2
```

8 – COMANDOS BÁSICOS III

Neste capítulo conheceremos o comando `ln`, utilizado para criar links.

8.1 – ln

O comando `ln` é utilizado para criação de links (ligações) entre arquivos. Se for usado sem opções, por default ele cria links diretos.

Sintaxe:

ln [opções] origem [destino]
ln [opções] origem... diretório

As opções mais utilizadas são:

- f Remove arquivos de destino já existentes.
- i Solicita confirmação antes de remover os arquivos de destino.
- s Cria um link simbólico ao invés de links diretos.
- v Lista o nome de cada arquivo antes de criar a ligação.

Caso somente um arquivo seja informado, ele liga o arquivo no diretório atual, isto é, cria uma ligação para aquele arquivo no diretório atual, com o nome igual ao nome daquele arquivo.

De outra forma, caso o último argumento seja um diretório existente, `ln` criará uma ligação para cada arquivo mencionado na origem naquele diretório, com o nome igual ao nome do arquivo de origem.

Se somente dois arquivos forem informados, ele cria uma ligação chamada destino para o arquivo origem.

Ocorrerá um erro se o último argumento não for um diretório e mais de dois arquivos forem informados.

Exemplos:

```
ln /etc/fstab /etc/fstab2
ln /etc/fstab
ln -s /etc/fstab /etc/fstab2
ln -s /etc dir_etc
```

9 – USUÁRIOS E GRUPOS

Veremos a seguir o que são usuários e grupos do sistema, bem como sua utilização no Linux. Mais adiante, aprenderemos como adicionar e remover usuários e grupos do sistema.

9.1 – Porque Criar Usuários?

Num primeiro momento, quem inicia no uso do Linux pode questionar a necessidade de se criar usuários, principalmente se já tem experiências utilizando DOS / WINDOWS.

A pergunta clássica seria: "Se posso fazer tudo como root, porque preciso criar um outro usuário?"

Realmente, como root você pode fazer TUDO mesmo, inclusive danificar o sistema acidentalmente.

É para evitar isto que deve-se criar usuários com menos "poderes" dentro do sistema, de modo a torná-lo menos vulnerável a este tipo de problema. Isto também aumenta consideravelmente a segurança do sistema, pois qualquer invasor, se não estiver como root, pouco dano poderá causar ao sistema.

Um outro objetivo que alcançamos ao criarmos vários usuários, é que cada um deles pode manter sigilo absoluto em relação aos demais, se desejado. Isto inclui todos os arquivos pessoais do usuário, inclusive e-mail, news, etc.

Concluindo, é altamente recomendado que sempre se acesse o sistema como usuário comum, utilizando-se do root apenas quando estritamente necessário.

OBS.: Diz-se que o usuário cadastrado no sistema possui uma "conta", a qual muitas vezes é referenciada como se fosse o usuário. Portanto, é comum dizer "criar uma nova conta" ao invés de "cadastrar um novo usuário".

9.2 – O Conceito de Grupo

Todos os usuários pertencem a um ou mais grupos. Como veremos mais adiante, no Linux cada arquivo tem um dono específico. Por conseqüência, cada arquivo pertence ao mesmo grupo do usuário proprietário.

O grupo pode ser exclusivo do dono do arquivo, ou compartilhado por diversos usuários. A habilidade de ler, gravar ou executar um arquivo pode ser atribuído a um grupo, separadamente das permissões do dono do arquivo. Por exemplo, o dono do arquivo pode ser capaz de gravar um documento, enquanto os membros do grupo somente poderão lê-lo.

10 – PERMISSÕES DE ACESSO

Para cada arquivo ou diretório, consideram-se três categorias de usuários:

- ✓ Dono: Quem criou o arquivo.
- ✓ Grupo: Grupo ao qual pertence o dono do arquivo.
- ✓ Outros: Usuários que não se enquadrem nas categorias anteriores.

Todo arquivo determina quais usuários têm acesso a ele e com que finalidade.

Cada conjunto de permissões de acesso significa presença ou ausência de permissões para: leitura (r); escrita (w); execução (x), conforme a tabela abaixo:

Modo de Acesso	Arquivo comum/especial	Diretório
Leitura "r"	examinar conteúdo de arquivo	listar arquivos do diretório
Escrita "w"	alterar o conteúdo do arquivo	escrever no diretório
Execução "x"	executa o arquivo como comando	pesquisar o diretório

Codificação utilizada para as permissões de acesso:

Tipo	Proprietário			Grupo			Sistema		
	leitura	escrita	execução	leitura	escrita	execução	leitura	escrita	execução
d	r	w	x	r	w	x	r	w	x
l	-	-	-	-	-	-	-	-	-
-									
1	2	3	4	5	6	7	8	9	10

- | | |
|--------------------------------|---|
| 1 – informa o tipo de arquivo | (d para diretório, l para link, – para demais arquivos) |
| 2 – Permissões do Proprietário | (r leitura permitida, – não permitida leitura) |
| 3 – Permissões do Proprietário | (w escrita permitida, – não permitida escrita) |
| 4 – Permissões do Proprietário | (x execução permitida, – não permitida execução) |
| 5 – Permissões do Grupo | (r leitura permitida, – não permitida leitura) |
| 6 – Permissões do Grupo | (w escrita permitida, – não permitida escrita) |
| 7 – Permissões do Grupo | (x execução permitida, – não permitida execução) |
| 8 – Permissões do Sistema | (r leitura permitida, – não permitida leitura) |
| 9 – Permissões do Sistema | (w escrita permitida, – não permitida escrita) |
| 10 – Permissões do sistema | (x execução permitida, – não permitida execução) |

11 – COMANDOS BÁSICOS IV

Neste capítulo trabalharemos com os comandos utilizados para adicionar e remover usuários e grupos do sistema, definir e alterar senhas, alterar as permissões de arquivos e alguns outros recursos para a administração de um sistema Linux.

11.1 – useradd

O comando `useradd` é utilizado para criar um novo usuário no sistema ou para atualizar os dados de um novo usuário.

Quando utilizado sem a opção `-D`, este comando criará uma nova conta usando os valores especificados na linha de comando e os defaults do sistema. A nova conta passará a fazer parte do sistema de arquivos, um diretório pessoal (`home`) para o novo usuário será criado e os arquivos iniciais serão copiados, dependendo das demais opções da linha de comando. Em algumas versões deste comando (como da Red Hat e derivados) será criado também um grupo para cada usuário adicionado ao sistema, a menos que a opção `-n` seja usada.

Quando utilizado com a opção `-D`, o comando **useradd** mostrará os dados default atuais do sistema ou atualizará estes dados, conforme informado na linha de comando.

Sintaxes:

useradd [opções] *nome_da_conta*

Principais opções:

`-d` *home_dir*

Criará o diretório *home_dir* para o novo usuário criado. Se esta opção não for utilizada, o nome do diretório criado será o mesmo da conta.

`-e` *expire_date*

A data na qual a conta do usuário será desabilitada. A data *expire_date* deve ser especificada no formato MM/DD/YY. Para habilitar novamente a conta, deve ser utilizado o comando **usermod** `-e` *expire_date*, onde *expire_date* é a nova data de expiração da conta do usuário.

`-n`

Um grupo com o mesmo nome do novo usuário será criado por default.

`-u` *uid*

O valor numérico de identificação do usuário. Este valor deve ser único e não negativo. Se não usada esta opção, será usado, por default, o menor ID disponível que seja maior que 99 e maior que os demais IDs dos demais usuários. Os valores entre 0 e 99 são tipicamente reservados para contas do sistema (system accounts).

useradd `-D` [opções]

`-b` *default_home*

Permite especificar o caminho inicial dos diretórios pessoais (`home`) dos novos usuários, normalmente `/home`.

`-e` *default_expire_date*

Permite especificar a data na qual a conta do usuário será desativada.

Curso de Linux Básico

`-f default_inactive`

Permite especificar quantos dias após uma senha ter expirado devem ser aguardados, antes da conta ser desativada.

`-g default_group`

Permite especificar o nome do grupo ou ID para ser o grupo inicial de novos usuários.

`-s default_shell`

Permite especificar o nome do shell para os novos usuários.

As informações sobre usuários, senhas, grupos, etc, ficam localizadas nos seguintes arquivos:

<code>/etc/passwd</code>	Informações de contas de usuário
<code>/etc/shadow</code>	Informações de contas de usuário seguras
<code>/etc/group</code>	Informações de grupos
<code>/etc/default/useradd</code>	Informações default do sistema
<code>/etc/login.defs</code>	Parâmetros do sistema
<code>/etc/skel</code>	Diretório que contém os arquivos default

OBS.: No Conectiva Linux existe por default um link para o comando **useradd**, que é chamado **adduser**. Portanto, neste sistema pode-se utilizar tanto **useradd**, que é o original, como **adduser**.

Exemplos:

```
useradd -D
useradd seu_nome
useradd -d meu_diretorio meu_nome
useradd -e 07/28/00 sua_conta
```

11.2 – passwd

O comando **passwd** é utilizado para atualizar a senha de autenticação de um usuário. Normalmente cada usuário pode alterar somente a sua própria senha, exceto o superusuário (root) que pode atualizar a senha de outro usuário, fornecendo o *nome_do_usuario*. Se não for especificado o *nome_do_usuario*, será atualizada a senha do usuário corrente.

Sintaxe:

passwd [opções] [*nome_do_usuario*]

Opções:

`-u`

Indica que a atualização somente pode ser efetuada para senhas expiradas, mantendo-se a senha atual até a data de sua expiração.

Exemplos:

```
passwd
passwd seu_nome
```

11.3 – chown

O comando **chown** é utilizado para alterar o dono e/ou grupo de arquivos.

Ele muda o dono e / ou o grupo de um determinado arquivo, de acordo com o primeiro argumento não opcional informado, o qual é interpretado conforme abaixo:

Caso somente um nome de usuário ou identificação numérica de usuário for informada, o usuário é transformado no dono do arquivo informado e o grupo do arquivo não é alterado.

Se o nome do usuário é seguido por dois pontos ou ponto e um nome de grupo ou identificação numérica de um grupo é fornecido, sem espaços entre eles, o grupo do arquivo também será alterado.

Se os dois pontos ou o ponto e o grupo são informados, mas o nome de usuário é omitido, somente o grupo do arquivo será alterado.

Sintaxe:

```
chown [opções] [usuário] [:] [grupo] arquivo...
```

Principais opções:

- v Descreve as mudanças de propriedade realizadas.
- R Altera as propriedades dos diretórios e seus conteúdos de maneira recursiva.

Exemplos:

```
chown -v root /home/meu_diretorio
chown -v :root /home/meu_diretorio
chown -v root:root /home/sua_conta
```

11.4 – chgrp

O comando **chgrp** é usado para alterar o grupo ao qual pertencem os arquivos. Ele muda o grupo de cada arquivo passando-os para um novo grupo, o qual é informado através de seu nome ou de sua identificação numérica.

Sintaxe:

```
chgrp [opções] grupo arquivo...
```

Principais opções:

- v Descreve as mudanças de propriedade realizadas.
- R Recursivamente muda a propriedade dos diretórios e seus conteúdos.

Exemplo:

```
chgrp -v root /home/seu_nome
```

11.5 – chmod

O comando **chmod** é utilizado para alterar as permissões de acesso aos arquivos para as novas definições informadas.

Pode ser utilizado de dois modos: utilizando uma representação simbólica das mudanças a serem feitas ou utilizando um número em formato octal que represente o padrão de bits das novas permissões.

O formato do modo simbólico é '[ugoa...][[+|=][rwxXstugo...][...][,...]'.

Múltiplas operações simbólicas podem ser informadas, separadas por vírgulas.

A combinação das letras 'ugoa' controla quais usuários podem acessar o arquivo que será alterado: o dono do arquivo (u), outros usuários do grupo do arquivo (g), outros usuários não pertencentes ao grupo do arquivo (o), ou todo e qualquer usuário (a). Caso nenhum desses seja informado, o comando assume a opção 'a', porém bits configurados através da opção umask não serão afetados.

O operador '+' causa a adição das permissões informadas às permissões existentes para o arquivo; '-' provoca a sua remoção, e '=' provoca a mudança completa das permissões para as informadas.

As letras 'rwxXstugo' selecionam as novas permissões para os usuários definidos:

<i>Letra</i>	<i>Significado</i>
r	Leitura.
w	Gravação.
x	Execução (ou acesso a diretórios).
X	Execução somente se o arquivo ou diretório já tem permissão de execução para algum usuário.
s	Configurar o usuário ou identificação do grupo durante a execução.
t	Salvar a área de texto do programa na área de swap.
u	As permissões que o usuário proprietário do arquivo possui atualmente.
g	As permissões que outros usuários do grupo do arquivo têm para acessá-lo.
o	As permissões que outros usuários não pertencentes ao grupo do arquivo têm.

Pode-se utilizar também o modo numérico com um a quatro dígitos da base octal (0-7), derivados da adição dos bits com valores 4, 2, e 1, conforme tabela abaixo:

<i>Decimal</i>	4	2	1
0	–	–	–
1	–	–	♦
2	–	♦	–
3	–	♦	♦
4	♦	–	–
5	♦	–	♦
6	♦	♦	–
7	♦	♦	♦

Qualquer bit omitido é assumido como tendo o valor zero.

Curso de Linux Básico

O primeiro dígito seleciona a identificação de usuário (4), a seleção do grupo (2) e o salvamento dos atributos da imagem do arquivo (1).

O segundo dígito seleciona as permissões para o dono do arquivo: leitura (4), gravação (2) e execução (1).

O terceiro dígito seleciona as permissões de outros usuários do grupo do arquivo, com os mesmos valores do dono.

O quarto dígito faz o mesmo para outros usuários que não estejam no grupo do arquivo, também como os mesmos valores.

O `chmod` nunca muda as permissões de links simbólicos; a chamada ao sistema `chmod` não pode fazê-lo. Isso não é um problema desde que as permissões de links simbólicos nunca sejam utilizadas. Porém, para cada link simbólico informado na linha de comando, o `chmod` altera as informações do arquivo apontado pela ligação. Por outro lado, o `chmod` ignora links simbólicos encontrados durante a opção recursiva em diretórios.

Sintaxe:

chmod [opções] *modo arquivo...*

Principais opções:

- v Descreve as permissões alteradas.
- R Altera as permissões de diretórios e seus conteúdos de forma recursiva.

Exemplos:

```
chmod -v 777 /home/seu_nome
chmod -v 100 /home/seu_nome
chmod -v 400 /home/seu_nome
chmod -v u+w, u+x /home/seu_nome
chmod -v g+w /home/seu_nome
chmod -v g-w /home/seu_nome
```

11.6 – su

O comando **su** é usado para permitir que um usuário torne-se outro temporariamente. Ele executa um interpretador com a identificação real e efetiva de usuário, identificação de grupo e grupos suplementares do usuário. Caso o usuário não seja informado na linha de comando, o padrão é o superusuário (root). O interpretador executado é o especificado para o usuário no arquivo `passwd`, ou `/bin/sh` caso nenhum seja especificado.

Caso o novo usuário tenha senha, `su` solicita a senha, a menos que se tenha a identificação real de usuário igual a 0 (superusuário).

Sintaxe:

su [opções] [usuário]

Principais opções:

- s interpretador

Executa INTERPRETADOR ao invés do interpretador definido em `/etc/passwd`, a menos que o usuário que esteja executando `su` não seja o superusuário e o interpretador do usuário não seja restrito.

OBS.: Para retornar ao usuário original, basta digitar "exit".

Exemplos:

```
su
su seu_nome
su -s /bin/tcsh seu_nome
```

11.7 – userdel

O comando **userdel** é usado para remover uma conta de usuário e os arquivos relacionados a ele. Este comando modifica os arquivos de contas, apagando todas as entradas que se referem ao usuário.

Sintaxe:

userdel [-r] nome_do_usuario

Se a opção `-r` for utilizada, os arquivos no diretório home do usuário serão removidos junto com o próprio diretório. Os arquivos localizados em outros sistemas de arquivos terão de ser procurados e deletados manualmente.

OBS.: O comando **userdel** não permite exclusão de uma conta se o usuário estiver correntemente logado.

Exemplos:

```
userdel seu_nome
userdel -r meu_nome
```

12 – SISTEMA DE ARQUIVOS

Neste capítulo iremos tratar do sistema de arquivos utilizado pelo Linux e veremos também alguns conceitos necessários para entendermos melhor o sistema de arquivos. Veremos quais os diretórios básicos presentes na maioria dos sistemas Linux e as suas respectivas aplicações.

12.1 – CONCEITOS BÁSICOS

Antes de aprofundarmos sobre o sistema de arquivos do Linux, vamos esclarecer um pouco mais sobre partições. O Linux é um sistema operacional que suporta partições, ou seja, partes de um HD que são tratadas pela máquina como unidades de disco independentes. As partições podem ser criadas segundo diversos padrões, conforme o sistema operacional utilizado, e cada um deles utiliza um tipo de sistema de arquivo. Temos, portanto, um grande número de tipos de partições diferentes, sendo que dentre elas, podemos citar: FAT16 (DOS e WIN95), FAT32 (WIN98), NTFS (WIN NT), HPFS (OS/2).

12.1.1 – O que é?

Um sistema de arquivos é o método e a estrutura de dados que um sistema operacional utiliza para administrar arquivos em um disco ou partição, ou seja, a forma pela qual os arquivos estão organizados em um disco. A expressão também é utilizada para se referenciar a uma partição ou disco que seja usado para armazenar os arquivos ou outros tipos de sistemas de arquivos. Alguém pode dizer “eu tenho dois sistemas de arquivos”, significando que tem duas partições nas quais armazena arquivos ou aquela pessoa está usando o “sistema de arquivo estendido”, exemplificando o tipo do sistema de arquivo.

A diferença entre um disco ou partição e um sistema de arquivos é bastante significativa. Poucos programas (inclusive os programas que criam sistemas de arquivos) operam diretamente em setores não inicializados de um disco ou partição, e caso exista um sistema de arquivos ele será destruído ou danificado seriamente. A maioria dos programas trabalham em um sistema de arquivos e não funcionam em uma partição que não contenha um (ou que contenha um de tipo errado).

Antes de uma partição ou disco ser usado como um sistema de arquivos ele necessita ser inicializado, e a estrutura básica de dados necessita ser gravada no disco. Este processo é chamado criação de um sistema de arquivos.

O Linux suporta diversos tipos de sistemas de arquivos. Dentre esses destacamos:

minix

O mais antigo e presumivelmente o mais confiável, mas bastante limitado em características (algumas datas não aparecem, máximo de 30 caracteres para nome de arquivos, etc...) e restrito em armazenamento (no máximo 64 Mb por sistema de arquivos).

ext2

O mais poderoso e popular sistema de arquivos nativo do Linux. Desenhado para ser facilmente compatível com os avanços das novas versões, sem a necessidade de criar novamente os sistemas de arquivos já existentes.

Adicionalmente há o suporte a diversos outros sistemas de arquivos, para simplificar a troca de informações com outros sistemas operacionais. Estes sistemas de arquivos funcionam como se fossem nativos, exceto pela perda de algumas facilidades presentes no UNIX, ou apresentam algumas particularidades.

msdos

Compatibilidade com MS-DOS (e OS/2 e Windows NT) através de sistemas de arquivos FAT/FAT32.

umsdos

Sistemas de arquivos MS-DOS estendidos para suportar nomes longos, donos, permissões, links e arquivos de dispositivos do Linux. Isso permite que um sistema de arquivos msdos possa ser usado como se fosse um sistema Linux, removendo a necessidade de uma partição distinta para o Linux.

iso9660

Curso de Linux Básico

O sistema de arquivos padrão do CD-ROM. A extensão Rock Ridge que permite nomes longos também é suportada automaticamente.

nfs

Sistemas de arquivos em redes que permitem o compartilhamento e o fácil acesso aos arquivos entre diversos computadores da rede.

hpfs

O sistema de arquivos do OS/2.

A opção do sistema de arquivos a ser usado depende da situação. Caso a compatibilidade ou outras razões tornem um dos sistemas de arquivos não nativos necessário, então este deve ser utilizado. Caso a opção seja livre, então provavelmente a decisão mais acertada seja usar o ext2, uma vez que ele traz diversas facilidades sem sofrer perda de performance.

12.1.2 – Ponto de Montagem

Antes de um sistema de arquivos poder ser utilizado, ele necessita ser montado. O sistema operacional executa diversas verificações para estar seguro de que tudo está funcionando bem. Uma vez que todos os arquivos no Linux estão em uma única árvore de diretórios, a operação de montagem fará com que o novo sistema de arquivos pareça um subdiretório existente em algum sistema de arquivos já montado.

Como veremos mais à frente, o comando para montagem de um sistema de arquivos possui dois argumentos. O primeiro é o arquivo de dispositivo correspondente ao disco ou partição que contenha o sistema de arquivos. O segundo é o diretório sob o qual ele será montado. Após a execução do comando, dizemos então que “/dev/hda2 está montado no /home”, por exemplo.

Para examinar estes sistemas de arquivos, pode-se acessar estes diretórios exatamente da mesma forma que qualquer outro, como veremos mais adiante. É importante ressaltar a diferença entre o **dispositivo** /dev/hda2 e o **diretório montado** /home. Enquanto o primeiro dá acesso aos dados brutos do disco, o segundo permite o acesso aos arquivos contidos no mesmo disco. O diretório montado é chamado **ponto de montagem**.

12.2 – DIRETÓRIO RAIZ (/)

O primeiro sistema de arquivos (chamado raiz, por conter o diretório raiz (/)) é montado, não a partir de outros sistemas de arquivos, mas sim, de maneira automática durante a inicialização do sistema operacional, podendo-se estar certo de que ele sempre estará disponível, pois de outra forma o sistema não poderá ser inicializado.

A composição do diretório raiz de um sistema Linux típico pode ser representado pela tabela abaixo:

<i>DIRETÓRIO</i>	<i>CONTEÚDO</i>
boot	Arquivos estáticos de boot de inicialização (boot-loader).
bin	Arquivos executáveis (binários) de comandos essenciais pertencentes ao sistema e que são usados com frequência.
sbin	Arquivos de sistema essenciais.
root	Diretório local do superusuário (root).
home	Diretórios locais (home) dos usuários.
usr	Todos os arquivos de usuários devem estar aqui (segunda maior hierarquia).
proc	Não existe no HD, apenas no kernel. Permite acessar informações sobre a máquina e sobre os processos.
dev	Arquivos de dispositivos de entrada/saída (I/O).
etc	Arquivos de configuração do sistema da máquina local com arquivos diversos para a administração do sistema.

Curso de Linux Básico

<i>DIRETÓRIO</i>	<i>CONTEÚDO</i>
lib	Arquivos das bibliotecas compartilhadas usados com frequência.
tmp	Arquivos temporários gerados por alguns utilitários.
mnt	Ponto de montagem de partição temporária.
var	Informação variável.

Cada diretório listado será discutido em detalhes mais adiante.

O kernel do Linux normalmente está localizado na raiz / ou no /boot. Se estiver localizado em / é recomendado usar o nome vmlinux ou vmlinuz, sendo este último o nome que deverá ser usado em programas fonte do kernel do Linux atualmente.

12.3 – DIRETÓRIO /boot

Este diretório contém tudo que é necessário para carregar o sistema, exceto os arquivos de configuração e o gerenciador de boot.

O /boot é utilizado para qualquer coisa antes do kernel executar /sbin/init. Isto inclui setores master de inicialização (master boot sectors) guardados, arquivos de mapa de setor e qualquer outra coisa que não é editada manualmente.

Como exposto acima, o kernel do Linux pode estar localizado em / ou /boot. Se estiver em /boot, é recomendado usar um nome mais descritivo.

12.4 – DIRETÓRIO /bin

Contém os comandos que podem ser utilizados por todos os usuários e pelo administrador do sistema, os quais são requeridos no modo mono-usuário (single-user mode).

Pode também conter comandos que são utilizados indiretamente por alguns scripts.

Não é recomendado abrir subdiretórios dentro do /bin.

Os arquivos dos comandos que não são suficientemente essenciais para estar em /bin estarão localizados em /usr/bin. Os elementos que são utilizados pelos usuários isoladamente (porém não pelo root, como mail, chsh, etc) não são suficientemente essenciais para estar dentro da partição /bin.

Os seguintes comandos essenciais devem estar em /bin: arch, cat, chgrp, chmod, chown, cp, date, dd, df, dmeg, echo, ed, false, kill, in, login, mkdir, mknod, more, mount, mv, ps, pwd, rm, rmdir, sed, setserial, sh, sftc, seu, sinc, true, umount, uname.

12.5 – DIRETÓRIO /sbin

Tipicamente, /sbin contém arquivos essenciais para dar boot ao sistema, além dos arquivos em /bin. Qualquer coisa que se executar após /usr ter sido montado (quando não há problemas) deveria estar em /usr/sbin. Os arquivos de administração do sistema root local devem estar em /usr/local/sbin.

Decidir que arquivos vão no diretório /sbin é difícil. Se o usuário necessitar executar tais arquivos, estes deverão ir para outro diretório. Se somente o administrador do sistema ou o root necessitarem executar, tais como scripts da administração, então deverão ir para /sbin (não /usr/sbin ou /usr/local/sbin, pois o arquivo não é vital para a operação do sistema).

Arquivos e/ou comandos armazenados em /sbin são dos seguintes tipos :

- Comandos gerais (clock, getty, init, update, mkswap, swapon, swapoff, telinit);
- Comandos de saída (fastboot, fasthalt, halt, reboot, shutdown);
- Comandos de manipular sistema de arquivos (fdisk, fsck, fsck.*, mkfs, mkfs.*, onde * = é um dos seguinte: ext, ext2 minix, msdos, xia, e talvez outros, badblocks, dumpe2fs, e2fsck, mke2fs, mkost+found, tune2fs);
- O gerenciador de boot de inicialização (lilo);
- Comandos de rede (arp, ifconfig, route).

Além destes, existem outros arquivos que são opcionais em /sbin: sln, nc estático e ldconfig.

12.6 – DIRETÓRIO /root

Este diretório é opcional no Linux. Nos sistemas UNIX o diretório local do usuário root é tradicionalmente o diretório /. Em muitos sistemas Linux e em alguns sistemas UNIX utiliza-se o diretório /root.

O diretório local da conta do usuário root pode ser determinada por preferências locais. As possibilidades óbvias incluem /, /root, e /home/root. Se o diretório local do root não está armazenado na partição raiz, será necessário assegurar-se que tome / por default caso não seja localizado.

Não é recomendado o uso da conta root para coisas corriqueiras tal como ler o e-mail e ver as notícias (mail & news), recomenda-se que seja usada somente para a administração do sistema. Por esta razão recomendamos que não apareçam subdiretórios como Mail e News no diretório local da conta do usuário root. É recomendado que o mail para root seja redirecionado a um usuário mais adequado.

12.7 – DIRETÓRIO /home

O diretório /home é claramente um sistema de arquivos específico do diretório local. A regra de criá-lo difere de máquina para máquina. Descreve uma localização sugerida para os diretórios local dos usuários, assim, recomendamos que todas as distribuições LINUX usem este lugar como localização default dos diretórios locais.

Em sistemas pequenos, cada diretório de usuário é um dos subdiretórios abaixo do /home, como por exemplo: /home/dirson, /home/raulson, /home/weslei, etc.

Em sistemas maiores (especialmente quando os diretórios /home são compartilhados entre várias máquinas via rede) é útil subdividir os diretórios locais. A subdivisão pode ser implementada utilizando subdiretórios tais como /home/apoio, /home/docs, /home/cartas, etc.

12.8 – DIRETÓRIO /usr

O diretório /usr é a segunda maior seção do sistema de arquivos. /usr é informação compartilhada, somente de leitura, isto significa que /usr deve ser compartilhada entre várias máquinas que utilizam o Linux e não deve exibir qualquer informação local de uma máquina ou que varia com o tempo.

Nenhum pacote grande (como TeX ou GNU Emacs) deve utilizar o subdiretório direto abaixo de /usr, em vez disso, deve haver um subdiretório dentro de /usr/lib (o /usr/local/lib caso tenha sido instalado localmente). A propósito, para o sistema X Window faz-se uma exceção permitindo um considerável precedente, sendo esta prática amplamente aceita.

Em um sistema típico teremos mais ou menos os seguintes diretórios abaixo de /usr:

X11R6	Sistema X Window Versão 11 release 6
X386	Sistema X Window Versão 11 release 5 em plataformas X 386
bin	A maioria dos comandos de usuário
dict	Listas de palavras
doc	Documentação miscelânea
etc	Configuração do sistema
games	Jogos e arquivos educacionais
include	Arquivos header (cabeçalhos) incluídos por programas C
info	Diretório primário, o sistema GNU Info
lib	Bibliotecas
local	Hierarquia local
man	Manual on line
sbin	Arquivos de administração do sistema não vitais
share	Informação independente da arquitetura
src	Código fonte

Alguns links simbólicos a diretórios podem estar presentes. Esta possibilidade baseia-se na necessidade de preservar a compatibilidade com sistemas anteriores, haja visto que em todas as implementações podemos assumir o uso da hierarquia /var.

Poderão existir os seguintes links :

```
/usr/adm           ----->    /var/adm
/usr/spool         ----->    /var/spool
/usr/tmp          ----->    /var/tmp
/var/spool/locks  ----->    /var/lock
```

Deve existir também um link /usr/X11 apontando para a hierarquia do sistema X Window atual.

Veremos a seguir qual o conteúdo de cada um dos diretórios contidos em /usr.

12.8.1 – Subdiretório /usr/X386

Composto do sistema X Window, versão 11 release 5 em plataformas X86, este subdiretório é geralmente idêntico a /usr/X11R6, exceto que os links simbólicos de /usr devem estar ausentes se estiver instalado /usr/X11R6.

12.8.2 – Subdiretório /usr/bin

Composto pela maioria dos comandos do usuário, este é o diretório principal de comandos executáveis no sistema. Possui o mh (comandos para o sistema de manipular e-mail MH) e o X11 (link simbólico até /usr/X11R6/bin).

12.8.3 – Subdiretório /usr/dict

Tradicionalmente este diretório contém somente arquivos de palavras inglesas, o qual é utilizado por vários programas de ortografia.

As listas de palavras para outras linguagem podem usar o nome em inglês para a linguagem, por exemplo, /usr/dict/french, /usr/dict/danish, etc. Estes devem, se possível, utilizar o conjunto de caracteres ISO 8859 que é

Curso de Linux Básico

apropriado para a linguagem em questão ou, se possível, o conjunto de caracteres ISO 8859–1 (latin1).

É razoável ter aqui só as listas de palavras, e definir que elas serão os únicos arquivos comuns a todos os verificadores de ortografia.

12.8.4 – Subdiretório /usr/etc

Contém a configuração do sistema, porém armazenar a configuração /usr/etc do software que se encontra em /usr/bin e /usr/sbin é um problema. Montar /usr somente para leitura de um CD-ROM ou através de NFS é difícil no melhor dos casos.

Uma possível solução considerada foi eliminar completamente /usr/etc e especificar que todas as configurações se armazenem em /etc. Acontece que existe a possibilidade de que muitos podem querer ter alguns arquivos de configuração que não estejam na sua máquina local.

Eventualmente, decide-se que /etc deverá ser o único diretório que seja referenciado pelos programas (isto é, todos devem buscar configurações em /etc e não em /usr/etc). Qualquer arquivo de configuração que seja necessário para todo o sistema e que não era necessário antes de montar /usr (em uma situação de emergência) deve estar localizado em /usr/etc.

Arquivos específicos em /etc, em máquinas específicas, podem ter ou não um link simbólico aos arquivos de configuração localizados em /usr/etc. Isto também significa que /usr/etc é tecnicamente um diretório opcional no sentido restrito, mesmo assim é recomendado que todos os sistemas Linux o incorporem.

Não é recomendado que /usr/etc contenha links simbólicos que apontem para arquivos /etc. Isto é desnecessário e interferem no controle local das máquinas que compartilhem o diretório /usr.

12.8.5 – Subdiretório /usr/include

Neste diretório é onde todos os arquivos include de uso geral do sistema para programação em linguagem C e C++ podem ser localizados.

Descrição dos principais subdiretórios de /usr/include:

X11	Link simbólico até /usr/X11R6/include/X11.
arpa	Definição do protocolo definido por ARPNET.
asm	Link simbólico até /usr/src/linux/include/asm-<arch>.
bsd	Arquivos include de compatibilidade com BSD.
g++	Arquivos include de GNU C++.
gnu	Arquivos include GNU.
linux	Link simbólico a /usr/src/linux/include/linux.
net	Definições genéricas relacionadas com rede.
netax25	Definições específicas a +AX25 (ARRL AX25).
Netinet	Definições específicas a TCP/IP.
netipx	Definições específicas a +IPX (Novo IPX/SPX).
protocols	Definições de protocolos (baseadas em INET).
readline	A biblioteca readline GNU.
rpc	Definição RPC da Sun Microsystem (para uso de nfs).
Rpcsvc	Definição de serviços RPC da Sun Microsystem (para uso de nfs).
sys	Arquivos include de geração de sistemas.

Curso de Linux Básico

12.8.6 – Subdiretório /usr/lib

Inclui as bibliotecas para programas e pacotes, as bibliotecas objeto, arquivos de programa compilador, informação estática de vários casos, códigos executáveis (por exemplo os arquivos internos do gcc estão localizados abaixo de /usr/lib/gcc-lib) e outros tipos de informação.

Descrição dos principais subdiretórios de /usr/lib:

/usr/lib/ – bibliotecas para programação e pacotes.

X11	Link simbólico para /usr/X11R6/lib/X11.
emacs	Arquivos de suporte estáticos para o editor GNU Emacs.
games	Arquivos de dados estáticos para /usr/games.
groff	Bibliotecas e diretórios para GNU groff.
gcc-lib	Arquivos e diretórios específicos do sistema para gcc.
kbd	Tabelas de tradução de teclado e informação relacionada.
Mh	Bibliotecas para o sistema de manipular e-mail MH.
news	Cnews/INN.
smail	Smail.
terminfo	Diretórios para a base de dados terminfo.
texmf	TeX/MF (e ATeX) bibliotecas de informação.
uucp	Comandos de UUCP.
zoneinfo	Configuração e informação da zona horária.

Qualquer pacote de programa que contenha e precisa informação que não necessite ser modificada deve armazenar tal informação em /usr/lib. Recomenda-se a utilização de um subdiretório em /usr/lib para este propósito.

Nota: nenhuma informação específica de host para o sistema X Window deve ser armazenada em /usr/lib/X11 (que é realmente /usr/X11R6/lib/X11). Os arquivos de configuração específicos do host tal como Xconfig ou XF86Config devem ser armazenados em /etc/X11.

12.8.7 – Subdiretório /usr/local

O subdiretório /usr/local deve ser utilizado pelo administrador de sistemas quando instala o Linux localmente. Necessita ficar a salvo de ser sobrescrito quando o software do sistema se atualiza. Pode ser usado por programas e por informação que são compatível entre um grupo de máquinas, pois não se encontram em /usr.

Descrição dos principais subdiretórios de /usr/local:

bin	Arquivos.
doc	Documentação local.
etc	Arquivos de configuração utilizados somente no local.
games	Jogos instalados localmente.
lib	Bibliotecas para /usr/local.
info	Páginas de informação local.
man	Hierarquias de páginas de manual para /usr/local.
sbin	Administração do sistema.
scr	Código fonte local.

Este diretório deve estar vazio ao terminar de instalar o Linux pela primeira vez. Não deve haver exceções a esta regra, exceto talvez aos subdiretórios vazios listados.

O software instalado localmente deve estar localizado dentro de /usr/local, em vez de /usr, a menos que esteja sendo instalado para reimplantar ou atualizar software em /usr.

Note que o software localizado em / ou em /usr pode ser sobrescrito por atualizações do sistema (assim mesmo, é recomendado que as distribuições não sobrescrevam informações de /etc fora destas circunstâncias). Por esta razão, o software local não deve ser colocado fora de /usr/local sem uma boa causa.

12.8.8 – Subdiretório /usr/man

Inclui as páginas do manual, detalha a organização das páginas do manual através do sistema.

As páginas de manual ficam em diretórios de nomes man [1–9], conforme abaixo:

man1	Programas para usuários.
man2	Chamadas do sistema.
man3	Subrotinas e funções de biblioteca.
man4	Dispositivos.
man5	Formatos arquivos.
man6	Jogos.
man7	Miscelâneas.
man8	Administração do sistema.
man9	Funções e variáveis internas do kernel.

A prática de colocar as páginas do manual de diferentes idiomas nos subdiretórios apropriados de /usr/man também se aplica às outras hierarquias de páginas do manual, tais como /usr/local/man e /usr/X11R6/man. Isto também é aplicável à estrutura opcional de /var/catman, mostrada no subdiretório /var.

12.8.9 – Subdiretório /usr/sbin

Este diretório contém quaisquer arquivos não essenciais utilizados exclusivamente pelo administrador do sistema.

Os programas de administração do sistema que sejam utilizados para a reparação do sistema, montado no /usr, outras funções essenciais devem localizar-se em /sbin em vez de /usr/bin.

Tipicamente /usr/sbin contém os daemons de rede, quaisquer ferramentas de administração não essenciais e

Curso de Linux Básico

arquivos para programas servidores não-críticos. Isto inclui os daemons da internet que são chamados por inetd (chamados in.*) tais como in.telnetd e in.fingerd e os daemons baseados em rpc manipulados por portmap (chamados rcp.*), tais como rcp.infsd e rcp.mountd.

Os programas administrativos instalados localmente devem estar localizados em /usr/local/sbin.

12.8.10 – Subdiretório /usr/share

São informações que independem da arquitetura, qualquer especificação para /usr/share será incluída em um documento suplementar ao FSSTND, de acordo com a Linux Organization. Os pesquisadores do FSSTND acreditam não ser necessário /usr/share na maioria dos sistemas Linux.

12.8.11 – Subdiretório /usr/src

Contém o código fonte do kernel do Linux, qualquer código fonte não local deve localizar-se neste diretório. O único código fonte que sempre deve localizar-se em um lugar específico é o código do kernel (quando exista ou esteja enlaçado como parte de uma estrutura /usr/include).

Podem-se usar subdiretórios como desejar.

/usr/include deve conter links aos diretórios /usr/src/linux/include/asm-<arch> e /usr/src/linux/include/linux, cujos dados são necessários ao compilador C. Pelo menos estes arquivos include devem sempre ser distribuídos nas instalações que incluem um compilador C. Devem ser distribuídos no diretório /usr/src/linux de forma a não haver problemas quando os administradores do sistema atualizarem sua versão do kernel pela primeira vez.

/usr/src/linux pode também ser um link simbólico a uma árvore de código fonte do kernel.

12.9 – DIRETÓRIO /proc

O sistema de arquivos /proc é um sistema de arquivos virtual utilizado para manipular informação de processos e de sistema. É recomendada sua utilização para o armazenamento e obtenção de informação de processos, assim como outras informação do kernel ou da memória.

12.10 – DIRETÓRIO /dev

Este é o diretório dos dispositivos de entrada/saída. Contém um arquivo para cada dispositivo que o kernel do Linux pode suportar.

Também contém um script chamado MAKEDEV, o qual pode criar dispositivos quando necessário. Pode conter um MAKEDEV local para dispositivos locais.

Os links simbólicos não devem ser distribuídos nos sistemas Linux, somente como previsto na lista de dispositivos de Linux. Isto é porque as instalações locais diferem daquelas da máquina do administrador. Além disso, se um script de instalação configurar links simbólicos na instalação, estes links seguramente não se atualizarão se houverem trocas locais no hardware. Quando utilizados responsabilmente são de bom uso.

A lista de dispositivos compatíveis com o Linux é mantida por Peter Anvin (peter.anvin@linux.org) e está disponível no endereço eletrônico da internet ftp.eggdrassiml.com, no diretório /pub/device-list.

12.11 – DIRETÓRIO /etc

Contém arquivos e diretórios de configuração do sistema da máquina local com arquivos diversos para a administração de sistema.

Tipicamente /etc possui dois subdiretórios:

X11	Arquivos de configuração para o X11
skel	Esqueletos da configuração de usuários

O /etc/skel é a localização para os chamados arquivos esqueletos de usuários, que são os dados por default quando um novo usuário recebe uma conta. Este diretório pode conter subdiretórios para diferentes grupos de usuários (/etc/skel/apoio, /etc/skel/usuários).

O /etc/X11 é o lugar recomendado para todos os arquivos de configuração X11 locais da máquina. Este diretório é necessário para permitir o controle local se /usr for colocado somente para leitura. Os arquivos que devem ir neste diretório incluem Xconfig (e/ou XF86Config) e Xmodmap. O /etc/X11/xdm contém os arquivos de configuração xdm. Estes são a maioria dos arquivos normalmente gravados em /usr/lib/X11/xdm.

Esta descrição do conteúdo é genérica, portanto não está totalmente completa, pode haver algumas variações dependendo do distribuidor do Linux ou do administrador de sistema. Os arquivos /etc são composto de:

Arquivos gerais, necessários na maioria dos sistemas LINUX, tais como: adjtime, fstab, group, inittab, issue, ld.so.conf, lilo.conf, motd, mtab, mtools, passwd, profile, shells, termcap.

Arquivos de rede: exports, ftpusers, gateway, hosts, host.conf, host.equiv, host.lpd, inetd.conf, networks, printcap, protocols, resolv.conf, rpc, service.

Os sistemas com senhas sombreadas (shadow password) terão arquivos de configuração adicionais, em /etc (/etc/shadow e outros) e /usr/bin (useradd, usermod, e outros).

12.12 – DIRETÓRIO /lib

O diretório /lib contém aquelas bibliotecas compartilhadas que são necessárias para carregar o sistema e executar os comandos do sistema de arquivos raiz, além de módulos essenciais do kernel.

As bibliotecas que são necessárias somente pelos arquivos /usr (como qualquer arquivo X Window) não pertencem a /lib. Só as bibliotecas compartilhadas requeridas para executar os arquivos dentro de /bin e /sbin devem estar ali. A biblioteca libm.so.* poderia estar localizada em /usr/lib se não é utilizada de nenhuma forma em /bin ou /sbin.

Por razão de compatibilidade, /lib/cpp necessita existir como uma referência ao pre-processador C instalado no sistema. A localização usual do arquivo é /usr/lib/gcc-lib/ <target>/<versão>/cpp. Podem existir links /lib/cpp apontando para estes arquivos ou a qualquer outra referência a ele que exista no sistema de arquivos. (Por exemplo, /usr/bin/cpp é utilizado frequentemente). A especificação para /lib/module ainda não foi definida, pois ainda não há um consenso na comunidade Linux.

12.13 – DIRETÓRIO /tmp

O /tmp é utilizado para arquivos temporários gerados por alguns arquivos utilitários, preferencialmente em dispositivo rápido (um sistema de arquivos baseado em memória, por exemplo).

A "permanência" da informação que é armazenada em /tmp é diferente daquela que é armazenada em /var/tmp.

Curso de Linux Básico

/tmp pode ser limpo em cada inicialização ou a intervalos relativamente frequentes. Portanto, não se deve operar a informação armazenada em /tmp permanecendo por algum período determinado de tempo.

Os programas devem utilizar /tmp ou /var/tmp (que era originalmente /usr/tmp) de acordo os requisitos esperados da informação, pois não devem colocar nenhum arquivo particular em qualquer diretório de armazenamento temporário.

Os administradores de sistemas podem querer ajuntar /tmp a algum outro diretório, tal como /var/tmp. Isto é útil, por exemplo, para conservar espaço na partição raiz. Se isto for feito, então a permanência de arquivos em /var/tmp deve ser mesmo tão grande como a de /tmp.

O subdiretório /tmp pode estar na memória RAM.

12.14 – DIRETÓRIO /mnt

Este diretório foi previsto para o administrador poder montar temporariamente sistemas de arquivos quando necessitar. O conteúdo deste diretório é um assunto local e não deve afetar a maneira que executamos nenhum programa. É recomendável a não utilização deste diretório para programas de instalação.

12.15 – DIRETÓRIO /var

O diretório /var contém arquivos com informação variável. Inclui arquivos e diretórios em fila de execução, informação de ordem administrativa e arquivos temporários e transitórios.

Diretórios contidos em /var:

adm	Informações administrativas do sistema (obsoleto). Link simbólico até /var/log.
catman	Páginas do manual formatadas localmente.
lib	Informação do estado das aplicações.
local	Informação variável do software de /usr/local.
lock	Arquivos de bloqueio.
log	Arquivos de Agenda (log).
named	Arquivos DNS, somente rede.
nis	Arquivos base de dados NIS.
run	Arquivos relevantes a processos execução do sistema.
spool	Diretórios de trabalhos em fila para realizar-se depois.
tmp	Arquivos temporários, utilizado para manter /tmp menor possível.

/var é especificada aqui para fazer possível montar /usr somente para leitura. Tudo aquilo que alguma vez ficou em /usr e é escrito durante a operação normal do sistema (não durante a instalação e manutenção do software) deve ir em /var.

12.15.1 – Subdiretório /var/adm:

Este subdiretório era utilizado para armazenar a agenda do sistema (logs) e arquivos contabilizados (obsoleto). Tem sido repassado para /var/log e outros diretórios. Deve ser um link simbólico a /var/log até que todos os programas não se refiram mais a algum arquivo em /var/adm.

12.15.2 – Subdiretório /var/catman:

Curso de Linux Básico

Este diretório contém páginas do manual formatadas localmente (opcional) e proporciona uma localização padrão para os computadores que utilizam uma partição `/usr` somente para leitura, pois desejam permitir o armazenamento temporário de páginas do manual formatadas localmente.

12.15.3 – Subdiretório `/var/lib`

O subdiretório `/var/lib` contém diretórios com arquivos de informação de estado das aplicações.

<code>emacs</code>	Diretório do estado do Emacs.
<code>games</code>	Informação variável de jogos.
<code>news</code>	Arquivos variáveis de Cnews/INN.
<code>texmf</code>	Informação variável associada com TeX.
<code>xdm</code>	Arquivos de autenticação e código de erros de aplicações X Window.
<code>/var/lib/<nome></code>	É o lugar apropriado para o suporte de empacotamento de todas as distribuições. Diferentes distribuições de Linux podem utilizar diferentes nomes para suporte.

12.15.4 – Subdiretório `/var/local`:

Este subdiretório contém toda a informação variável que está relacionada com o software que se encontra em `/usr/local`. Naturalmente a implementação deste subdiretório é prerrogativa do administrador do sistema. Como a informação pode estar noutra lugar do diretório `/var`, não se deve colocá-la em `/var/local`. Por exemplo, todos os arquivos de bloqueios estarão em `/var/lock`.

12.15.5 – Subdiretório `/var/lock`:

Os arquivos de bloqueio devem ser armazenados dentro de uma estrutura do subdiretório `/var/lock`.

Para preservar a habilidade de montar `/usr` somente para leitura, não se deverá colocar os arquivos de bloqueio na partição `/usr`.

Os arquivos de bloqueio dos dispositivos, tais como os arquivos de bloqueio dos dispositivos seriais, que antes se encontravam em `/usr/spool/lock` ou em `/usr/spool/uucp` devem agora ser armazenado em `/var/lock`. A convenção para a nomenclatura que deve utilizar-se é LCK., seguido do nome base do dispositivo. Por exemplo, para bloquear `/dev/cua0` se deverá criar o arquivo `LCK.cua0`.

12.15.6 – Subdiretório `/var/log`:

Este subdiretório contém arquivos agenda miscelâneos. A maioria dos arquivos agenda se devem exibir neste diretório ou subdiretórios apropriados.

Alguns arquivos contidos em `/var/log` e seu conteúdo:

<code>lastlog</code>	Registro do último acesso de cada usuário.
<code>messages</code>	Mensagens do sistema desde que está logado.
<code>wtmp</code>	Registro de todos os acessos e saídas.

Pode requerer um link simbólico de `/var/log/utmp` até `/var/run/utmp`. Basta que nenhum programa se refira a `/var/adm/utmp` (`/var/adm` é em si mesmo um link simbólico transicional até `/var/log`).

12.15.7 – Subdiretório `/var/named`:

Curso de Linux Básico

Este subdiretório contém todos os arquivos de trabalho do servidor de nomes Internet (DNS), named. Recomendamos que /etc/named.boot seja um link simbólico até /var/named/named.boot, dado que /etc/named.boot é o arquivo de inicialização default, caso não sejam fornecidos argumentos a named.

12.15.8 – Subdiretório /var/nis:

Arquivos da base de dados do serviço de informação de rede (NIS).

12.15.9 – Subdiretório /var/preview:

Este subdiretório contém os arquivos que são armazenados antes de qualquer terminação não esperada.

12.15.10 – Subdiretório /var/run:

Este subdiretório contém arquivos variáveis de tempo de execução, com informação do sistema que o descrevem desde que inicializou. Geralmente os arquivos neste subdiretório devem ser deletados (removidos ou truncados) ao começar o processo de inicialização.

Os arquivos identificados do processo (PID), que estavam originalmente em /etc, devem ser colocados em /var/run. A convenção de nomenclatura dos arquivos PID é <nome-programa>.pid, por exemplo o arquivo PID de crond se chama /var/run/crond.pid.

O arquivo utmp, que armazena informação sobre quem está atualmente utilizando o sistema se localiza neste subdiretório.

Os programas que mantenham sockets transitórios de domínio UNIX, devem colocá-los neste subdiretório.

12.15.11 – Subdiretório /var/spool:

Subdiretórios de fila de trabalhos para processamento posterior. /var/spool é tradicionalmente utilizado para a informação local de máquina que é enviada para processo depois.

Por exemplo, trabalhos de impressão que são armazenados aqui para entrega posterior ao daemon da impressora, o e-mail que sai é armazenado aqui para entrega a sistemas remotos e os trabalhos de at e cron são armazenados aqui para execução posterior pelo daemon cron.

Subdiretórios de /var/spool e seu conteúdo:

at	Trabalhos de at
cron	Trabalhos de cron
lpd	Diretório de impressora *
mail	arquivos caixa-postal dos usuários
mqueue	Fila de espera dos correio
news	Diretório de noticias *
rwhod	arquivos rwhod
smail	Diretório de smail *
uucp	Diretório de UUCP

* Significa fila de trabalhos para processamento posterior.

12.15.12 – Subdiretório /var/tmp:

Contém arquivos temporários, utilizado para manter /tmp pequeno. Os arquivos que estão no /var/tmp estão

Curso de Linux Básico

armazenados por uma duração não específica (lembre-se que os diretórios temporários do sistema não garantiram manter a informação por nenhum período particular).

A informação armazenada em `/var/tmp` tipicamente está numa "forma definida localmente", pois usualmente é menos frequente que `/tmp`.

Deve existir um link simbólico de `/usr/tmp` até `var/tmp` por razão de compatibilidade.

13 – ACESSANDO HD, CDROM E DISQUETE

Neste capítulo trataremos da forma como o Linux faz acesso aos dados contidos em Hds, CDROMs e disquetes.

13.1 – Nomes dos Dispositivos

O Linux possui uma identificação bastante diferente do DOS / Windows quanto às unidades de disquete, Hds e CDROMs, bem como aos demais dispositivos.

A seguir, veremos como são identificados alguns dos dispositivos mais comuns no Linux:

<i>Descrição</i>	<i>Identificação</i>
Winchesters (HDs) e CDROMs (IDE)	/dev/hda, /dev/hdb, ... ou /dev/ide0/..., /dev/ide1/...
Floppy drives	/dev/fd0H1440, /dev/fd0h1200, /dev/fd1H1440, ...
Dispositivos SCSI	/dev/sda, /dev/sdb, /dev/sdc. ...
Impressoras paralelas	/dev/lp0, /dev/lp1, /dev/lp2, ...
Portas seriais (COM 1, COM 2, etc)	/dev/cua0, /dev/cua, ...1 ou /dev/ttyS0, /dev/ttyS1, ...

OBS.:

- 1) No kernel 2.4 os nomes dos dispositivos foram alterados. Por exemplo: /dev/hda passou a ser /dev/ide0/...
- 2) Normalmente no diretório /dev existem diversos arquivos de dispositivos, não significando que todos estes dispositivos estejam instalados. Isto facilita a instalação de novos componentes de hardware no sistema, pois não há necessidade de encontrar os parâmetros corretos para o dispositivo sendo instalado.

13.2 – Montagem de Dispositivo

No Linux, os arquivos em qualquer dispositivo de armazenamento (disquete, CDROM ou HD), devem fazer parte do sistema de arquivos para poderem ser acessados. Ou seja, é diferente do DOS / Windows, que trata cada unidade (a:, c:, etc) como um sistema de arquivos isolado.

Para que os arquivos armazenados em um disquete, CDROM ou HD possam fazer parte do sistema de arquivos, é preciso realizar uma operação chamada "montagem". Basicamente a montagem consiste em informar ao sistema os parâmetros necessários para o acesso aos dados, como o ponto de montagem, o dispositivo utilizado, o tipo do sistema de arquivo, entre outros.

A operação de montagem pode ser realizada manualmente ou pode ser automatizada para que seja realizada em determinadas circunstâncias, especialmente na inicialização do sistema.

Da mesma forma, quando um sistema de arquivos em um dispositivo não é mais necessário, ele deve ser "desmontado" antes de ser fisicamente removido, de modo a não fazer mais parte do sistema de arquivos raiz (/).

O arquivo /etc/fstab contém informações que permitem sintetizar os comandos de montagem de dispositivos, já que as informações contidas nele não precisarão ser informadas ao sistema durante a operação de montagem dos dispositivos nele relacionados. Mais adiante veremos como isto funciona.

13.3 – Ponto de Montagem

Como vimos anteriormente, "ponto de montagem" é a denominação do local onde um sistema de arquivos está montado.

No Linux, o diretório destinado à montagem de sistemas de arquivos temporários é o /mnt. Porém nada impede que os mesmos sejam montados em outros diretórios, atentando apenas para o fato de estarmos saindo um pouco fora dos padrões adotados em consenso pela comunidade Linux.

Veremos a seguir os principais pontos de montagem normalmente utilizados nos diversos sistemas Linux:

<i>Dispositivo</i>	<i>Ponto de Montagem</i>
Floppy drive	/mnt/floppy
CDROM	/mnt/cdrom

Desta forma, se quisermos acessar o conteúdo de um disquete, devemos montá-lo e teremos acesso a seus dados através o diretório /mnt/floppy. Da mesma forma com os dados de um CDROM, porém neste caso o diretório que nos dará o acesso aos dados é /mnt/cdrom.

14 – COMANDOS BÁSICOS V

Neste capítulo veremos alguns comandos utilizados para montar e desmontar sistemas de arquivos a partir de um dispositivo, bem como para formatar disquetes e criar sistemas de arquivos.

14.1 – mount

Todos os arquivos acessíveis em um sistema Linux estão organizados em uma grande árvore, a hierarquia de arquivos, iniciada pelo raiz simbolizado como */*. Estes arquivos podem estar distribuídos por diversos dispositivos. O comando `mount` destina-se a incluir o sistema de arquivos encontrado em algum dispositivo à grande árvore de arquivos raiz (*/*).

O formato padrão do comando `mount` é **mount** *-t tipo dispositivo dir*. Isso indica ao kernel para incluir o sistema de arquivos encontrado em *dispositivo* (o qual é do tipo *tipo*) nominando-o como diretório *dir*.

Sintaxe:

mount *-a [-fnrvw] [-t tipo]*

mount *[-fnrvw] [-o opções [...]] dispositivo | dir*

mount *[-fnrvw] [-t tipo] [-o opções] dispositivo dir*

Opções disponíveis para o comando `mount` :

-v

Modo de mensagens ativas.

-a

Monta todos os sistemas de arquivos (ou aqueles com os tipos mencionados) descritos em `fstab`.

-n

Montagem sem gravação de `/etc/mstab`. Isso é necessário por exemplo quando o sistema de arquivos `/etc` está somente com permissões de leitura.

-f

Faz com que tudo seja executado exceto a montagem efetiva em si. Apesar de não ser tão óbvia, esta opção permite que falsas montagens sejam realizadas, e é útil quando em conjunto com *-v* permite determinar o que o comando `mount` está tentando fazer. Pode ainda ser usado para adicionar entradas para dispositivos que foram montados anteriormente com a opção *-n*.

-r

Monta o sistema de arquivos somente com permissões de leitura. Um sinônimo é *-o ro*.

-w

Monta o sistema de arquivos com permissões de leitura e gravação. Este é o padrão. É um sinônimo de *-o rw*.

-t tipo

O argumento seguinte a -t é usado para indicar o tipo do sistema de arquivo. Uma relação dos tipos suportados pelo Linux pode ser encontrada em `linux/fs/filesystems.c`, quais sejam: `minix`, `ext`, `ext2`, `xiafs`, `hpfs`, `msdos`, `umdos`, `vfat`, `proc`, `nfs`, `iso9660`, `smbfs`, `ncpfs`, `affs`, `ufs`, `romfs`, `sysv`, `xenix`, `coherent`. Note que os últimos três são equivalentes e que `xenix` e `coherent` serão descontinuados em algum momento no futuro. Sugere-se o uso de `sysv` em seu lugar. Desde o kernel 2.1.21 os tipos `ext` e `xiafs` foram descontinuados.

O tipo `iso9660` é o padrão. Se nenhuma opção -t for apresentada, ou se o tipo `auto` for especificado, o superbloco será testado para verificação do tipo do sistema de arquivos (`minix`, `ext`, `ext2`, `xiafs`, `iso9660`, `romfs` são suportados). Caso este teste falhe e `/proc/filesystems` exista, então todos os sistemas de arquivos listados serão testados, exceto aqueles que estejam marcados como "nodev" (por exemplo `proc` e `nfs`). Note que o tipo `auto` pode ser útil para unidades de disquetes montadas pelos usuários. Porém atente que o teste usa um método heurístico (a presença de um número mágico) e pode reconhecer de forma equivocada o tipo do sistema de arquivos).

Mais que um tipo pode ser especificado com uma vírgula como separador. A lista dos tipos de sistema de arquivos pode ser precedida pela palavra **no** para especificar tipos de sistemas que não devem ser utilizados nos testes. (Isso pode não ter sentido com a opção -a *option*.)

-o

Opções são especificadas com um indicador -o seguido por vírgula como separador. Algumas dessas opções são úteis somente quando aparecem no arquivo `/etc/fstab`. As opções a seguir aplicam-se a qualquer sistema de arquivos que esteja sendo montado:

<code>auto</code>	Pode ser montado com a opção -a .
<code>defaults</code>	Usa as opções padrão: <code>rw</code> , <code>suid</code> , <code>dev</code> , <code>exec</code> , <code>auto</code> , <code>nouser</code> e <code>async</code> .
<code>dev</code>	Interpreta dispositivos especiais de blocos ou caracter no sistema de arquivos.
<code>exec</code>	Permite a execução de binários.
<code>noauto</code>	O arquivo somente pode ser montado explicitamente.
<code>nodev</code>	Dispositivos especiais de blocos ou caracter não devem ser interpretados.
<code>noexec</code>	Não permite a execução de qualquer binário no sistema de arquivos montado.
<code>nosuid</code>	Não permite usar os bits de configuração de identificação de usuário ou grupo.
<code>nouser</code>	Proíbe que um usuário comum monte o sistema de arquivos. Este é o padrão.
<code>remount</code>	Tenta remontar um sistema de arquivos já montado.
<code>ro</code>	Monta o sistema de arquivos somente para leitura.
<code>rw</code>	Monta o sistema de arquivos com permissão de leitura e gravação.
<code>suid</code>	Permite o uso dos bits de configuração de identificação do usuário e do grupo.
<code>user</code>	Permite que um usuário normal possa montar o sistema de arquivos.

Além das citadas, existem várias opções especiais (que se seguem ao parâmetro -o) que se aplicam aos diversos tipos de sistemas de arquivos. Para mais informações, deve-se consultar `man mount`.

Na tabela abaixo estão os principais arquivos relacionados ao comando mount:

Arquivo	Descrição
/etc/fstab	Tabela de sistemas de arquivos
/etc/mtab	Tabela de sistemas de arquivos montados
/etc/mtab~	Arquivo de lock
/etc/mtab.tmp	Arquivo temporário

O arquivo /etc/fstab pode conter linhas descrevendo quais dispositivos são usualmente montados, e com quais opções. O arquivo é usado de três formas:

- a) O comando mount `-a [-t tipo]` faz com que todos os sistemas de arquivos indicados em fstab (de tipo apropriado) sejam montados conforme indicado, exceto para aqueles cujas linhas contenham a palavra chave `noauto`.
- b) Quando estiver montando um sistema de arquivos mencionado em fstab, é suficiente fornecer somente o dispositivo, ou somente o ponto de montagem.
- c) Quando fstab contém a opção `user` na linha, qualquer usuário poderá montar este sistema.

Exemplos:

```
mount /dev/fd0
mount /mnt/floppy
mount -a -t nomsdos,ext
```

14.2 – umount

O comando `umount` retira o sistema de arquivos indicado da hierarquia de arquivos. O sistema de arquivos a ser “desmontado” pode ser especificado tanto através da informação do diretório onde ele foi montado, quanto pelo nome do dispositivo onde ele reside.

Devemos notar que um sistema de arquivos não pode ser desmontado quando ele está em uso – por exemplo quando há arquivos abertos ou quando alguns processos tenham seu diretório de trabalho nele, ou quando um arquivo de swap esteja em uso.

Sintaxes:

```
umount -a [-nrv] [-t tipo]
```

```
umount [-nrv] dispositivo | dir [...]
```

Principais opções:

- `-v` Modo de apresentação de mensagens.
- `-n` Desmontar sem escrever em /etc/mtab.
- `-r` No caso da desmontagem falhar, tenta remontar somente para leitura.
- `-a` Todos os sistemas de arquivos descritos em /etc/mtab são desmontados.
- `-t tipo` Indica que as ações podem ser realizadas nos sistemas de arquivos do tipo especificado. Mais de um tipo pode ser especificado, separados por vírgulas. A lista de tipos de sistemas de arquivos pode ter um prefixo **no** para especificar os tipos de sistemas de arquivos nos quais as ações não podem ser exercidas.

Exemplos:

```
umount /dev/fd0
umount -v /mnt/floppy
umount -a -t noext2
```

14.3 – fdformat

O comando **fdformat** executa uma formatação de baixo nível em um disquete. O parâmetro *dispositivo* é normalmente um dos seguintes: /dev/fd0d360, /dev/fd0h1200, /dev/fd0D360, /dev/fd0H360, /dev/fd0D720, /dev/fd0H720, /dev/fd0h360, /dev/fd0h720, /dev/fd0H1440, /dev/fd1d360, /dev/fd1h1200, /dev/fd1D360, /dev/fd1H360, /dev/fd1D720, /dev/fd1H720, /dev/fd1h360, /dev/fd1h720 ou /dev/fd1H1440.

Os dispositivos de disquetes genéricos, /dev/fd0 e /dev/fd1, não funcionarão com fdformat quando um formato não padrão estiver sendo usado, ou o formato não seja auto detectado.

Sintaxe:

```
fdformat [-n ] dispositivo
```

Opções:

-n Não verificar. Esta opção desabilitará a verificação que é realizada após a formatação.

14.4 – mkfs

O comando **mkfs** constrói um sistema de arquivos Linux em um dispositivo, geralmente uma partição de um disco rígido ou disquete. O parâmetro *sistema-arq* pode ser o nome do dispositivo (por exemplo /dev/hda1, /dev/fd0H1440) ou o ponto de montagem (por exemplo /, /usr, /home) para o sistema de arquivos. O parâmetro opcional [blocos] é a quantidade de blocos a ser utilizada pelo sistema de arquivos.

Os códigos de retorno do comando **mkfs** são: 0 em caso de sucesso e 1 em caso de erro.

Sintaxe:

```
mkfs [-V ] [-t tipo] [opções] sistema-arq [blocos ]
```

Principais Opções:

-V

Exibe informações detalhadas sobre os comandos executados, incluindo os comandos específicos de cada sistema de arquivos. Passando esta opção mais de uma vez inibe-se a execução de comandos específicos ao sistema de arquivos. Isso é útil durante a realização de testes.

-t tipo

Especifica o tipo de sistema de arquivos a ser criado. Se não especificado o tipo padrão de sistema de arquivo (atualmente ext2) é criado.

Curso de Linux Básico

Opções

Opções específicas do sistema de arquivos a ser passado ao construtor de sistemas de arquivos. Embora não seja garantido, geralmente as opções seguintes são suportadas por muitos construtores de sistemas de arquivos.

-c

Checa o dispositivo por blocos defeituosos durante a criação do sistema de arquivos.

-l *nomearq*

Lê a lista de blocos defeituoso a partir de *nomearq*.

-v

Mostra o que está sendo feito (modo detalhado, do inglês "verbose").

Exemplos:

```
mkfs -t ext2 /dev/fd0H1440
```

```
mkfs -t msdos /dev/fd0H1440
```

15 – FORMAS DE ACESSO A DISQUETES

Neste capítulo veremos duas formas diferentes para acessarmos os dados contidos em disquetes, quando trabalhamos com dois ou mais sistemas de arquivos de tipos diferentes nestes disquetes.

15.1 – Usando Dois Pontos de Montagem

A primeira forma, consiste em usar mais de um ponto de montagem para o mesmo dispositivo, sendo que cada um dos pontos de montagem será utilizado para acessar um sistema de arquivos de tipo diferente.

Para realizarmos isto, devemos:

1. Criar os pontos de montagem, o que significa criar dois diretórios diferentes, através dos quais os dados serão acessados. Normalmente estes diretórios devem ser criados dentro de /mnt.
2. Editar o arquivo /etc/fstab e informar os dois pontos de montagem, correspondendo ao mesmo dispositivo, porém com tipos de sistemas de arquivos diferentes.

Exemplo:

```
cd /mnt
mkdir disk
mkdir floppy
pico /etc/fstab
```

Dentro de fstab:

```
/dev/fd0      /mnt/floppy    ext2 noauto  0 0
/dev/fd0      /mnt/disk      vfat noauto  0 0
```

Desta forma, portanto, temos um mesmo dispositivo (/dev/fd0) que pode ser montado em dois pontos diferentes, sendo /mnt/floppy o ponto de montagem correspondente ao sistema de arquivos ext2 e /mnt/disk o correspondente a vfat.

Assim, ao executarmos o comando `mount /mnt/floppy` o sistema tentará encontrar no dispositivo /dev/fd0 um sistema de arquivos do tipo ext2. Ao executarmos `mount /mnt/disk` o sistema tentará encontrar no dispositivo /dev/fd0 um sistema de arquivos do tipo vfat.

É bom observarmos que neste caso não podemos usar o comando `mount` informando apenas o dispositivo (/dev/fd0), pois o sistema não saberá a qual dos dois pontos de montagem estamos nos referindo, especialmente porque cada um deles foi definido para corresponder a um determinado tipo de sistema de arquivos.

15.2 – Usando Apenas um Ponto de Montagem

Esta segunda forma de acesso a disquetes tende a ser mais simples que a primeira, pois usa apenas um ponto de montagem, com identificação automática do tipo de sistema de arquivo que está sendo utilizado.

Para isto, o arquivo /etc/fstab deve conter uma linha como abaixo, sendo obrigatório o uso do parâmetro *auto*:

```
/dev/fd0      /mnt/floppy    auto noauto  0 0
```

Para montar o dispositivo, o comando será: `mount /mnt/floppy`

16 – ACESSANDO WIN9x

É bastante freqüente termos na mesma máquina o Linux e o DOS ou Windows, normalmente cada um em uma partição ou mesmo em HDs separados. Portanto, é comum a necessidade de acessar pelo Linux os dados que estão na partição DOS / Windows.

Para possibilitarmos isto é muito simples, basta criar o ponto de montagem e em seguida montar o dispositivo correspondente à partição onde o acesso é desejado.

Exemplo:

```
mkdir /mnt/win  
mount -t vfat /dev/hda1 /mnt/win
```

Se quisermos facilitar ainda mais, devemos editar o arquivo `/etc/fstab` e adicionar uma linha semelhante à descrita abaixo, para informarmos ao sistema os parâmetros que devem ser utilizados pelo comando `mount`:

```
/dev/hda1          /mnt/win          vfat noauto,user  0 0
```

Note que as opções *noauto,user* indicam que este sistema de arquivos não é montado automaticamente (*noauto*) e que ele pode ser montado pelos usuários (*user*) e não apenas pelo root. Se desejar que este sistema de arquivos seja montado automaticamente, substitua as opções *noauto,user* por *defaults*. Para saber mais sobre o arquivo `fstab` e as opções que podem ser utilizadas, utilize o comando `man fstab`.

Utilizando a linha descrita acima no arquivo `/etc/fstab`, para montarmos o sistema de arquivos basta informarmos o dispositivo ou o ponto de montagem:

```
mount /dev/hda1          ou          mount /mnt/win
```

Se utilizarmos a opção *defaults* em lugar de *noauto,user* o sistema de arquivos será montado automaticamente durante a inicialização do sistema e poderemos acessar seus dados simplesmente adentrando no ponto de montagem.

17 – MODO GRÁFICO (X) I

No Linux, o modo gráfico é dividido em três componentes: o servidor gráfico, o gerenciador de janelas e a aplicação.

O servidor gráfico, ou servidor X é responsável por fornecer um acesso amigável a dispositivos de hardware, como teclado, mouse e vídeo. Ele permite que as aplicações mostrem suas informações de forma gráfica na tela, através das janelas. Neste caso, a maioria dos objetos da interface, como botões, menus e barras de rolagem fazem parte de janelas. No Linux o servidor X da maior parte das distribuições é o Xfree86 (gratuito), apesar de existirem outros, inclusive comerciais.

O gerenciador de janelas, por sua vez, tem como função controlar o layout da janela na tela. Ele fornece a barra de título, a barra de ajuste de tamanho, os menus de aplicação, ícones e demais janelas. No Linux existe uma enorme variedade de gerenciadores de janelas, entre os quais podemos citar: Gnome, Kde, WindowMaker, AfterStep, ICEWM, BlackBox, FVWM, Enlightenment.

Uma aplicação é qualquer programa que rode no modo gráfico do Linux. Geralmente são criadas através de uma biblioteca de desenvolvimento (como GTK, qt, etc).

17.1 – Configurando o Servidor X

Durante a instalação do Conectiva Linux o servidor X já foi configurado, porém às vezes é necessário refinar as configurações iniciais, ou mesmo reconfigurá-lo novamente, devido à configuração inicial ter sido incorreta.

Também pode ser necessário reconfigurá-lo ao efetuar alterações no hardware da máquina, como troca de teclado, mouse, placa de vídeo ou monitor, por outros de características diferentes dos anteriormente instalados.

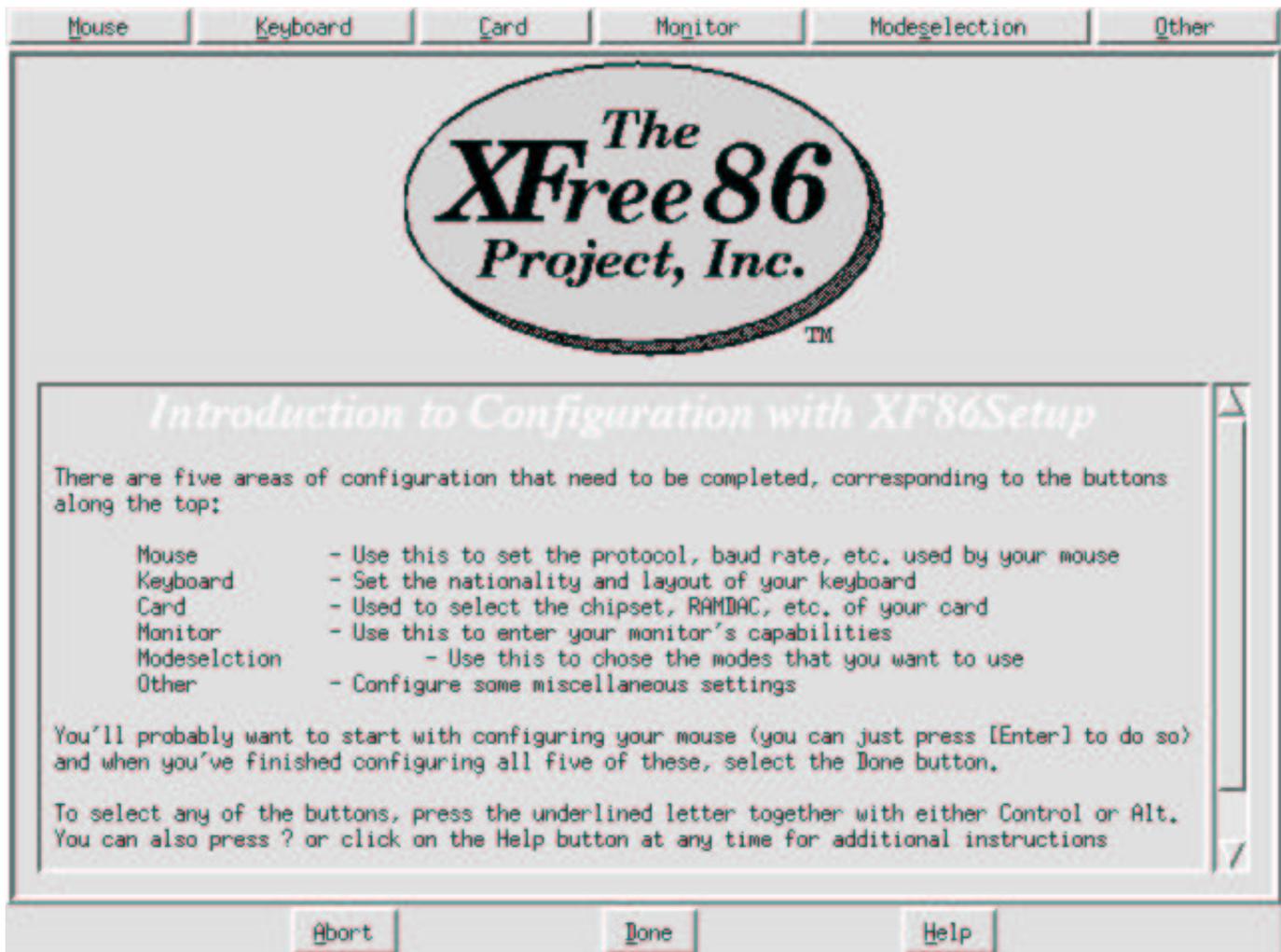
Além disso, pode ser necessário corrigir o posicionamento da janela dentro da tela, principalmente se o computador é utilizado com outros sistemas operacionais, para evitar ajustes no monitor de vídeo a cada vez que se alterne entre o modo texto e o modo gráfico, ou entre o Linux e outros sistemas operacionais.

As configurações do sistema X Window ficam armazenadas no arquivo *XF86Config* (ou *XF86Config-4* na versão 4 do *XFree86*), o qual geralmente está localizado em */usr/X11R6/lib/X11* ou em */etc/X11*.

Para auxiliar na tarefa de configuração do X Window, foram desenvolvidos alguns programas, dentre os quais podemos citar o Xconfigurator, o XF86Setup e mais recentemente, o xf86cfg.

O Xconfigurator é o mais simples dos dois, e é o mesmo programa que é utilizado na configuração inicial do modo gráfico do Conectiva Linux, durante o processo de instalação. Procura detectar os dispositivos automaticamente e permite que as configurações sejam selecionadas por meio de caixas de listagem contendo as opções suportadas.

Já o XF86Setup, cuja tela inicial está mostrada abaixo, é bem mais flexível, permitindo que sejam informados parâmetros genéricos, principalmente para placas de vídeo, além de possuir mais opções e parâmetros que o Xconfigurator.



O XF86Setup, além de configurar o servidor X, pode chamar, se desejável, o programa **xvidtune**, que permite que se faça ajustes no posicionamento da janela gráfica na tela do monitor de vídeo. Conforme alerta o próprio criador do programa, ele deve ser utilizado com bastante cuidado e sempre sabendo o que se está fazendo, pois o uso incorreto deste programa pode danificar a placa de vídeo ou mesmo o monitor de vídeo.

Para utilizá-lo deve-se antes ajustar o monitor de vídeo para que esteja centralizado e ocupando a maior área possível da tela, isto no modo texto.

Após chamar o programa **xvidtune**, devemos utilizar os botões *Left*, *Right*, *Wider* e *Narrower* para deslocar a tela para a esquerda (*Left*) ou para a direita (*Right*), aumentar (*Wider*) ou diminuir (*Narrower*) a largura e os botões *Up*, *Down*, *Shorter*, *Taller* para deslocar a tela para cima (*Up*), para baixo (*Down*), diminuir (*Shorter*) ou aumentar (*Taller*) a altura da tela.

O botão *Auto* deve estar ativado se for desejável que as alterações realizadas sejam aplicadas automaticamente, caso contrário, à medida em que os ajustes forem sendo feitos, os mesmos devem ser testados, usando para isto o botão *Test*. O botão *Show* mostra (num terminal texto) os valores selecionados. Após a finalização dos ajustes, deve-se aplicar (*Apply*) as configurações e sair (*Quit*) do programa.

As configurações feitas através do **xvidtune** devem ser salvas na seção "Monitor", encontrada no arquivo `/etc/X11/XF86Config` (ou `/etc/X11/XF86Config-4`).

A figura a seguir mostra a tela do **xvidtune**:

HDisplay: 800	VDisplay: 600
HSyncStart: 840	VSyncStart: 601
<input type="text"/>	<input type="text"/>
HSyncEnd: 968	VSyncEnd: 605
<input type="text"/>	<input type="text"/>
HTotal: 1056	VTotat: 628
<input type="text"/>	<input type="text"/>
<input type="button" value="Left"/> <input type="button" value="Right"/> <input type="button" value="Wider"/> <input type="button" value="Narrower"/>	<input type="button" value="Up"/> <input type="button" value="Down"/> <input type="button" value="Shorter"/> <input type="button" value="Taller"/>
Flags (hex): <input type="text" value="0005"/>	Pixel Clock (MHz): 40.00
<input type="button" value="Quit"/> <input type="button" value="Apply"/> <input type="button" value="Auto"/> <input type="button" value="Test"/> <input type="button" value="Restore"/>	Horizontal Sync (kHz): 37.88
<input type="button" value="Fetch"/> <input type="button" value="Show"/> <input type="button" value="Next"/> <input type="button" value="Prev"/>	Vertical Sync (Hz): 60.31

Independentemente do programa utilizado para a configuração da interface gráfica, é importante ter em mente que se ocorrerem problemas na configuração, mais especificamente por falta de informação sobre o hardware (monitor, placa de vídeo, etc...), devem ser tentadas inicialmente as configurações mais modestas (menos resolução de vídeo, menos bits de cores, menos memória de vídeo, menores frequências de varredura do monitor, etc...), fazendo testes e alterando para configurações mais exigentes, sempre testando, até alcançar as configurações ótimas para o hardware em questão.

Durante a utilização do modo gráfico podem ocorrer situações em que algumas configurações de teclado eventualmente necessitem ser refinadas, principalmente se a distribuição utilizada não for específica para suportar o Português. São comuns problemas na digitação de acentos e do cedilha. Normalmente estes pequenos inconvenientes não são corrigidos pelo Xconfigurator, XF86Setup ou xf86cfg, devendo a correção ser feita manualmente, editando-se os arquivos de configuração correspondentes.

Por exemplo, para que a acentuação e o cedilha fiquem corretos no StarOffice, edite o arquivo `/etc/X11/XF86Config` (ou `/etc/X11/XF86Config-4`) e inclua (ou descomente) a linha contendo `XkbDisable` na seção "keyboard"(ou "Input devices").

Para conseguir que as teclas `delete` e `back space` funcionem corretamente, edite o arquivo `/etc/X11/xinit/Xmodmap` e acrescente os códigos corretos para estas teclas:

```
keycode 22 = BackSpace  
keycode 107 = Delete
```

17.2 – Utilizando o Modo Gráfico (WindowMaker)

O WindowMaker é um gerenciador de janelas bastante leve e com muitos recursos, tendo sido desenvolvido pelo brasileiro Alfredo Kojima, baseado no After Step, um outro gerenciador de janelas.

No Conectiva Linux, o WindowMaker é o gerenciador de janelas padrão do sistema, bastando digitar **startx** para que seja iniciado. Para sair, além da opção `sair` no menu, pode ser utilizada a combinação das teclas **Ctrl + Alt + ↵** (Backspace), a qual encerra o servidor X.



A figura acima ilustra a tela do WindowMaker, onde podemos perceber os seguintes componentes:

- O Dock, coluna de ícones localizado do lado direito da tela, abaixo do qual existem aplicações acopladas;
- O Clip, ícone localizado do lado esquerdo da tela, abaixo do qual existem ícones de aplicativos;
- As janelas miniaturizadas e os ícones de aplicação, localizadas abaixo na tela e que correspondem a janelas de aplicativos minimizadas e também a ícones de aplicações sendo executadas;
- Barras de menus, como Aplicações e Conectiva Linux;
- Aplicativos maximizados, como a Calculadora mostrada na figura.

A função do Dock é acoplar ícones e aplicações (Dock apps), que são pequenos programas executados dentro do próprio ícone, como relógios, indicadores de consumo de memória, indicadores da atividade da CPU, etc.

O Clip é um tipo de Dock mais sofisticado. A função do Clip é permitir alternar entre as diversas áreas de trabalho criadas, bem como permitir que aplicações sejam iniciadas (lançadas) mediante configuração do comando e posterior lançamento.

As janelas miniaturizadas e os ícones de aplicação podem ser confundidos num primeiro momento, pois são muito parecidos. Uma diferença é que as janelas miniaturizadas possuem uma pequena barra de título, ao contrário dos ícones de aplicações. Estes, por sua vez, possuem no canto inferior esquerdo pequenos pontos indicadores de estado, podendo ser três pontos quando a aplicação não está sendo executada, dois pontos quando a janela da aplicação está escondida dentro do ícone ou nenhum ponto quando a aplicação está sendo executada.

O menu é utilizado para executar programas ou comandos. Para acionar o menu, basta clicar com o botão direito do mouse sobre uma área livre do desktop (tela). Também pode ser acionado a partir da tecla F12. Para fazer desaparecer o menu, basta clicar com o botão esquerdo do mouse sobre uma área livre do desktop ou teclar **Esc**. Também é possível manter um menu sempre visível, bastando clicar com o botão esquerdo do mouse sobre a sua barra de título. Aparecerá um X no canto superior direito do menu, o qual servirá para fechar o mesmo,

quando desejarmos.

Para executar um aplicativo, além da opção através do menu, podemos utilizar o duplo clique com o botão esquerdo do mouse sobre o ícone do aplicativo, se ele possuí-lo.

A janela de uma aplicação pode ser maximizada, minimizada, restaurada, escondida ou redimensionada, por meio das funções disponíveis ao clicar com o botão direito do mouse sobre a barra de título da respectiva janela. As tarefas de minimizar e fechar a aplicação também estão disponíveis como botões na barra de título da janela, bastando clicar com o botão esquerdo do mouse sobre o (do lado esquerdo da barra de título) para minimizar ou no X (do lado direito da barra de título) para fechar a aplicação.

As janelas de aplicação podem ser movidas dentro do desktop. Isto pode ser feito clicando-se com botão esquerdo do mouse sobre a barra de título da janela e arrastando-a para o local desejado. Pode-se também fazer isto pressionando a tecla **Alt**, clicando-se em qualquer área da janela e arrastando-a, devendo a tecla **Alt** permanecer apertada durante a operação.

Já vimos que uma aplicação minimizada aparece no desktop na forma de uma janela miniaturizada, na parte inferior da tela. Para restaurar uma aplicação minimizada, basta clicar duas vezes com o botão esquerdo do mouse sobre a janela miniaturizada correspondente.

Os ícones associados ao Clip podem ser escondidos, sendo que para isso deve-se utilizar o duplo clique com o botão esquerdo do mouse sobre o Clip. Ao ser repetida esta operação, os ícones voltarão a aparecer na tela.

Uma lista de todas as janelas de aplicações que estão abertas (sendo executadas) pode ser vista ao se clicar com o botão do meio do mouse em uma área livre da tela. O mesmo efeito pode ser conseguido pressionando a tecla F11. Clicando-se novamente sobre uma das aplicações listadas, a respectiva janela será maximizada.

Podem ser criadas várias áreas de trabalho diferentes, bastando utilizar o menu *Áreas de Trabalho, Novo*. Para alternar entre as diversas áreas de trabalho, além da utilização das setas do Clip, podemos utilizar também a combinação de teclas *Ctrl + Alt + seta direita/esquerda* ou ainda, *Alt + número* (do teclado alfanumérico), como por exemplo, *Alt + 1*, *Alt + 2*, etc.

É possível levar uma janela de aplicação aberta em uma área de trabalho para outra. Para isto, deve-se pressionar a tecla Shift e com ela pressionada, clicar com o botão esquerdo do mouse sobre a barra de título da janela que se deseja transportar para outra área de trabalho. Assim esta janela será selecionada. Em seguida, deve-se alternar para a área de trabalho desejada e, com a tecla Shift pressionada, clicar com o botão esquerdo do mouse novamente sobre a barra de título da janela de aplicação. Pronto. A janela está na outra área de trabalho.

Para copiar texto no modo gráfico do Linux, basta selecionar o texto desejado, arrastando o mouse sobre ele com o botão esquerdo pressionado e em seguida, clicar com o botão do meio do mouse na posição onde o texto copiado deve ser colado.

17.2.1 – Configuração do WindowMaker:

Veremos a seguir como configurar os diversos itens que o WindowMaker nos permite alterar ou personalizar. Algumas delas, como a criação e eliminação de ícones na área de trabalho, são bastante simples e podem ser realizados com recursos do próprio WindowMaker, porém outras, como edição dos menus, são mais facilmente realizadas por meio de programas escritos especificamente para configurar este gerenciador de janelas. No nosso caso, veremos dois destes programas: o **Wprefs** e o **wmakerconf**.

17.2.1.1 – Criação e exclusão de ícones:

Veremos a seguir como criar e destruir ícones na área de trabalho do WindowMaker. Estas tarefas são bastante simples e são feitas utilizando recursos do próprio gerenciador de janelas.

Para criar novos ícones no desktop, tudo o que se tem a fazer é executar o programa desejado, seja através do menu ou linha de comando. Se o aplicativo executado tiver sido criado para rodar em modo gráfico, seu ícone correspondente aparecerá na parte inferior da tela. Deve-se então, com o botão esquerdo do mouse, arrastar

Curso de Linux Básico

este ícone (veja bem, estamos nos referindo ao ícone da aplicação e não à janela miniaturizada) para perto de uma das colunas de ícones (abaixo do Clip ou abaixo do Dock) e o ícone será fixado ali.

Para que a alteração tenha efeito permanente, é preciso salvar a alteração. Para isso, deve-se utilizar o menu *Área de Trabalho, Salvar Sessão*. Após isto, o WindowMaker sempre iniciará com o ícone adicionado junto aos demais já existentes.

Para excluir um ícone do conjunto de ícones do desktop, basta, com o botão esquerdo do mouse, arrastar o ícone que se deseja excluir para o centro da tela. Assim que o botão do mouse for solto no centro da tela, o ícone arrastado para lá será destruído. Vale dizer que esta alteração só será permanente caso seja salva, utilizando o menu *Área de Trabalho, Salvar Sessão*.

17.2.1.2 – Configuração do WindowMaker com WPrefs:

O programa **Wprefs** é o meio mais tradicional de se configurar o ambiente gráfico do WindowMaker. Veremos aqui apenas algumas das configurações mais interessantes, visto que são muitas as opções oferecidas ao usuário. Se houver interesse por parte do usuário em alterar outras configurações não descritas aqui, o mesmo não deverá encontrar dificuldades, pois o **Wprefs** está em Português e possui um sistema de "ajuda em balões" que pode ser ativado na sua tela inicial, como mostrado abaixo:



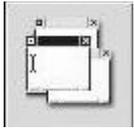
O Wprefs é composto de treze ícones, que se encontram na parte superior da janela, cada um correspondendo a um conjunto de opções que podem ser selecionadas ou configuradas. Obs.: A versão apresentada aqui é a 0.40, sendo que para outras versões os ícones podem ter sua aparência diferente das ilustradas aqui.

A seguir veremos a correspondência entre cada ícone e as respectivas opções de configuração disponíveis:



Opções de manuseio de janelas.
Posicionamento inicial, borda resistente, movimento opaco, etc.

Curso de Linux Básico



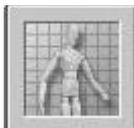
Política de mudança de foco de teclado, mapa de cores para displays de 8 bpp e outras opções relacionadas.



Opções de uso de menu. Velocidade de rolagem, alinhamento de submenus, etc.



Opções de manuseio de ícones e minijanelas. Área de posicionamento, tamanho dos ícones, estilo de animação.



Várias opções, como texto em balões, geometria, etc.



Caminhos de procura quando exibir pixmaps e ícones.



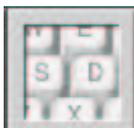
Propriedades de navegação da área de trabalho. Permite habilitar e desabilitar o Dock e o Clip.



Velocidade das animações, estilos de barras de título, várias opções e número de cores para reservar para o WindowMaker em displays 8 bits.



Edita o menu para a execução de aplicações.



Muda as teclas de atalho para ações como mudança de áreas de trabalho e abrir menus.



Velocidade de aceleração do mouse, velocidade do duplo clique, atalhos nas teclas do mouse, etc.



Configuração da textura de fundo para as janelas, menus e ícones.



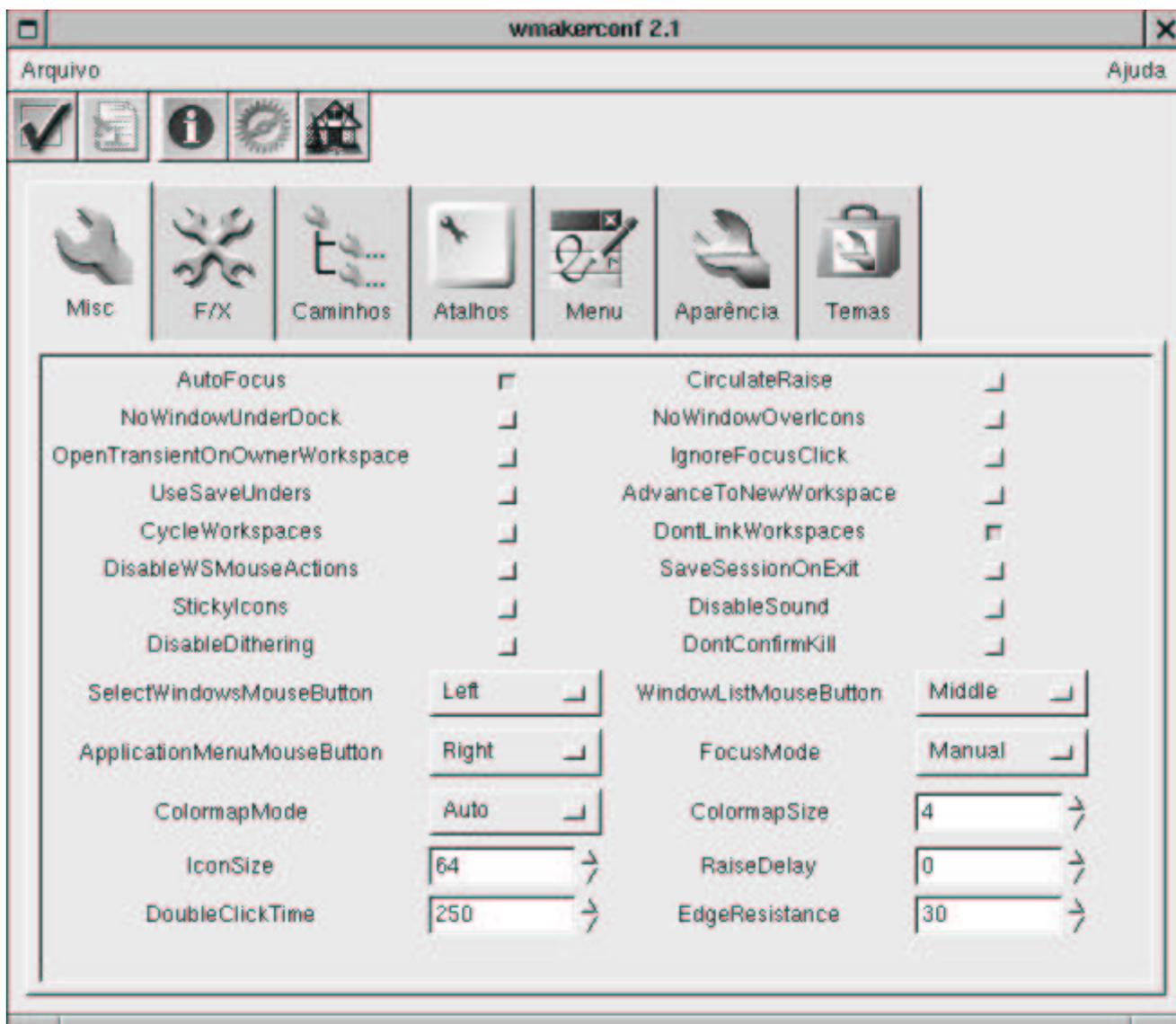
Opções diversas (avançadas).

Curso de Linux Básico

Uma das opções mais interessantes é a correspondente ao nono ícone, o qual permite mudanças na barra de menus. Utilizando este recurso, é possível acrescentar novos comandos e programas, bem como remover outros que podem não ser necessários.

17.2.1.3 – Configuração do WindowMaker com Wmakerconf:

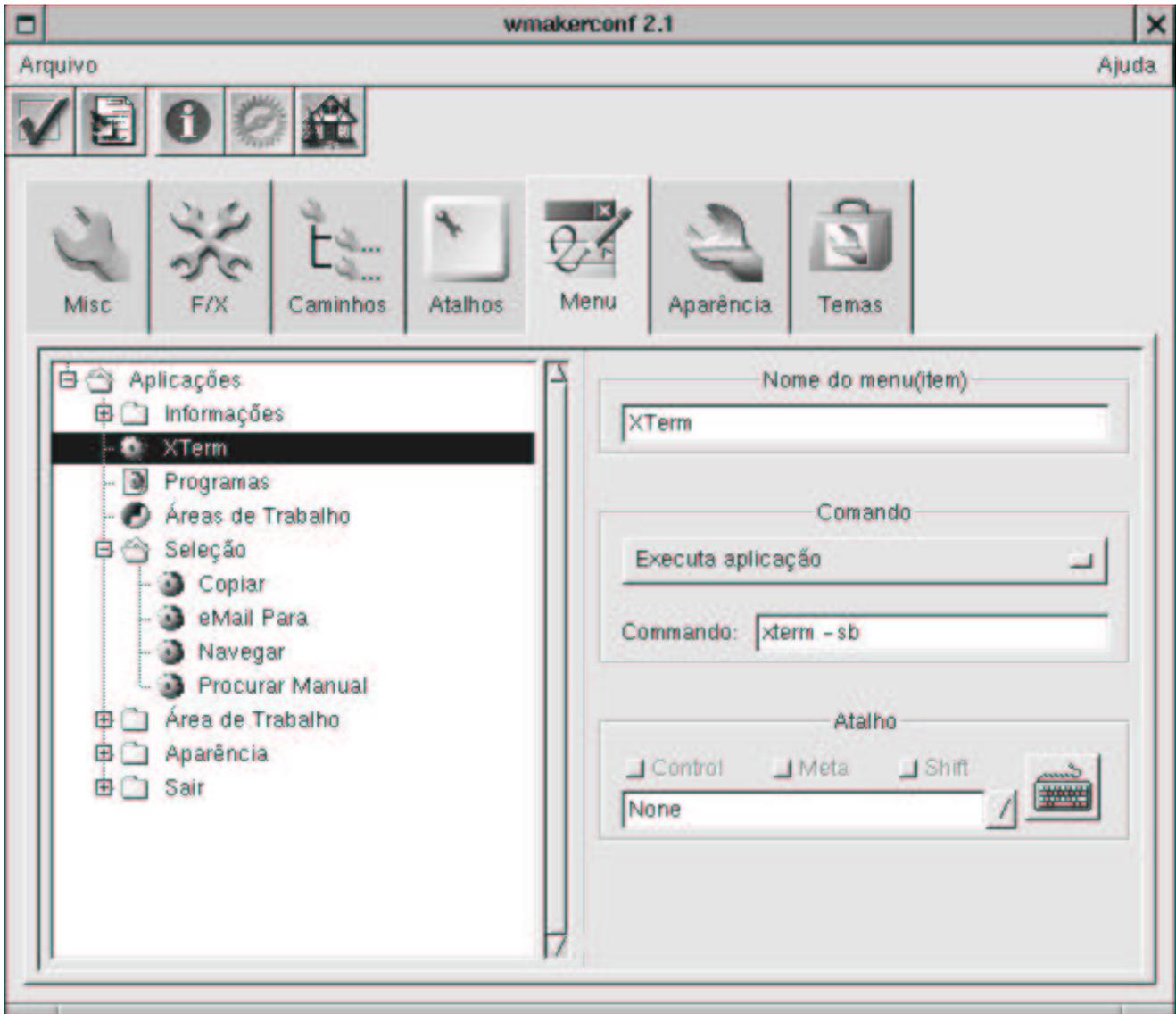
O programa **Wmakerconf** é outro programa utilizado para configurar o WindowMaker, possuindo basicamente todos as opções contidas no Wprefs, porém com algumas particularidades em relação a este último. Abaixo, uma ilustração da tela inicial do Wmakerconf versão 2.1:



Uma desvantagem do Wmakerconf em relação ao Wprefs é que ele não tem todas as opções traduzidas para o Português, o que torna seu uso um pouco mais difícil para quem não domina o Inglês.

As vantagens deste programa são bastante perceptíveis nos três botões de opções da direita (Menu, Aparência e Temas).

A opção Menu, cuja tela é mostrada a seguir, permite a personalização da barra de menus, seja acrescentando, removendo ou alterando menus e submenus, de uma forma bastante intuitiva, já que utiliza uma forma gráfica do tipo árvore.



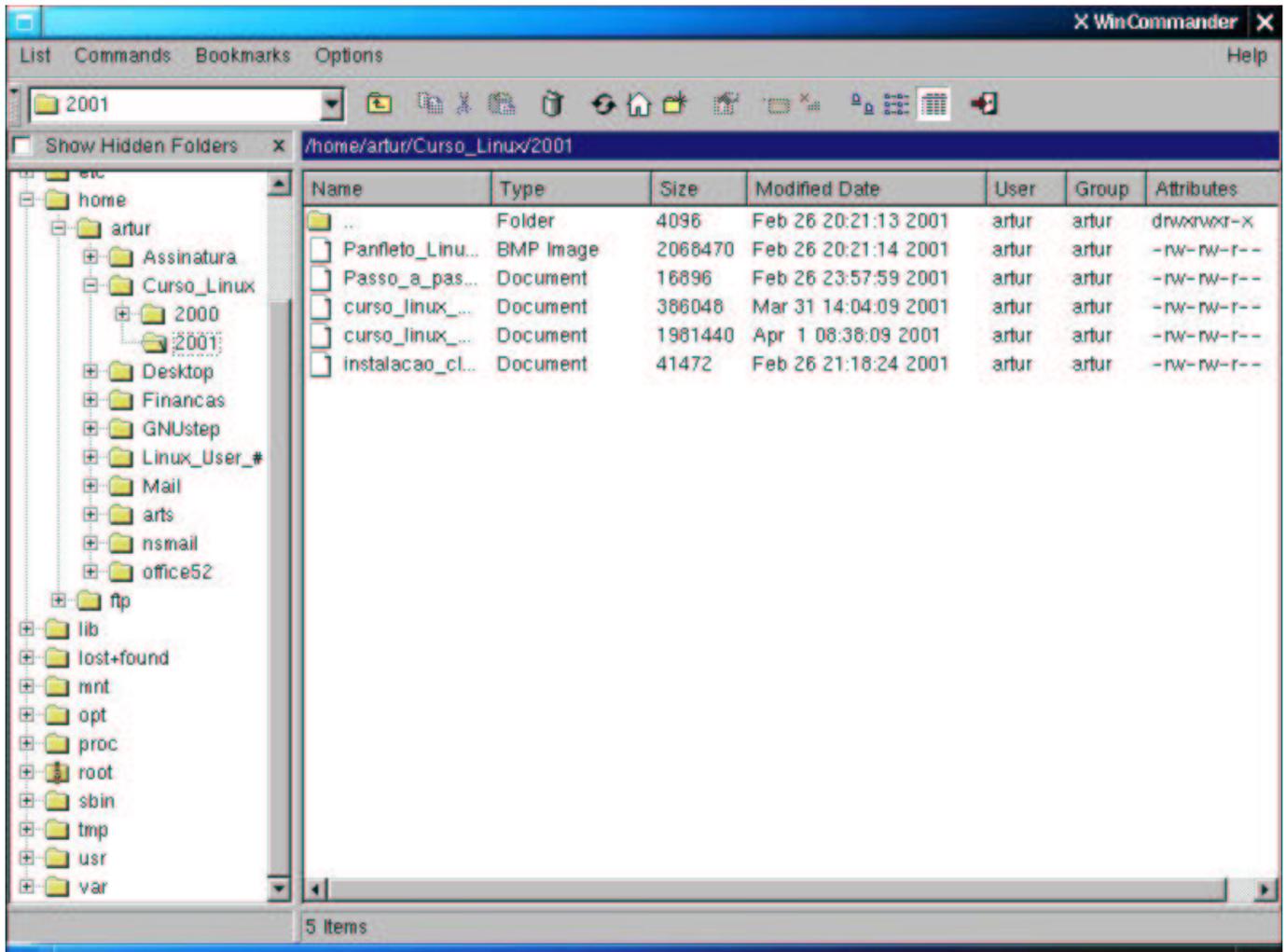
A opção Aparência permite a alteração dos padrões das áreas de trabalho e dos menus, como papel de parede, cores, estilo, fontes, etc. Permite inclusive a definição de um padrão diferente para cada área de trabalho.

A opção Temas permite que sejam carregados temas para a área de trabalho, contendo estilos, cores, figuras para papel de parede, sons, etc.

Após todos os itens terem sido configurados, os parâmetros devem ser salvos, antes de finalizar o Wmakerconf.

17.2.2 – XWC (Gerenciador de Arquivos):

Existem diversos programas gerenciadores de arquivos para o Linux, sendo o **xwc** um deles, utilizado em modo gráfico. A tela do **xwc** lembra o Explorer do Windows 9x, como pode ser observado na figura a seguir:



O funcionamento do xwc é muito semelhante ao Explorer do Windows 9x, possuindo visualização detalhada ou na forma de ícones, possui botão para selecionar visualização ou não dos arquivos ocultos, etc.

17.3 – Utilizando o Modo Gráfico (KDE)

O KDE é um gerenciador de janelas com características bastante semelhantes ao Windows 9x, possuindo uma enorme gama de recursos. É bem mais pesado que outros gerenciadores de janelas, tais como o WindowMaker, exigindo bem mais da máquina que este último. Sua principal proposta é tornar o Linux mais fácil, utilizando uma interface gráfica amigável.

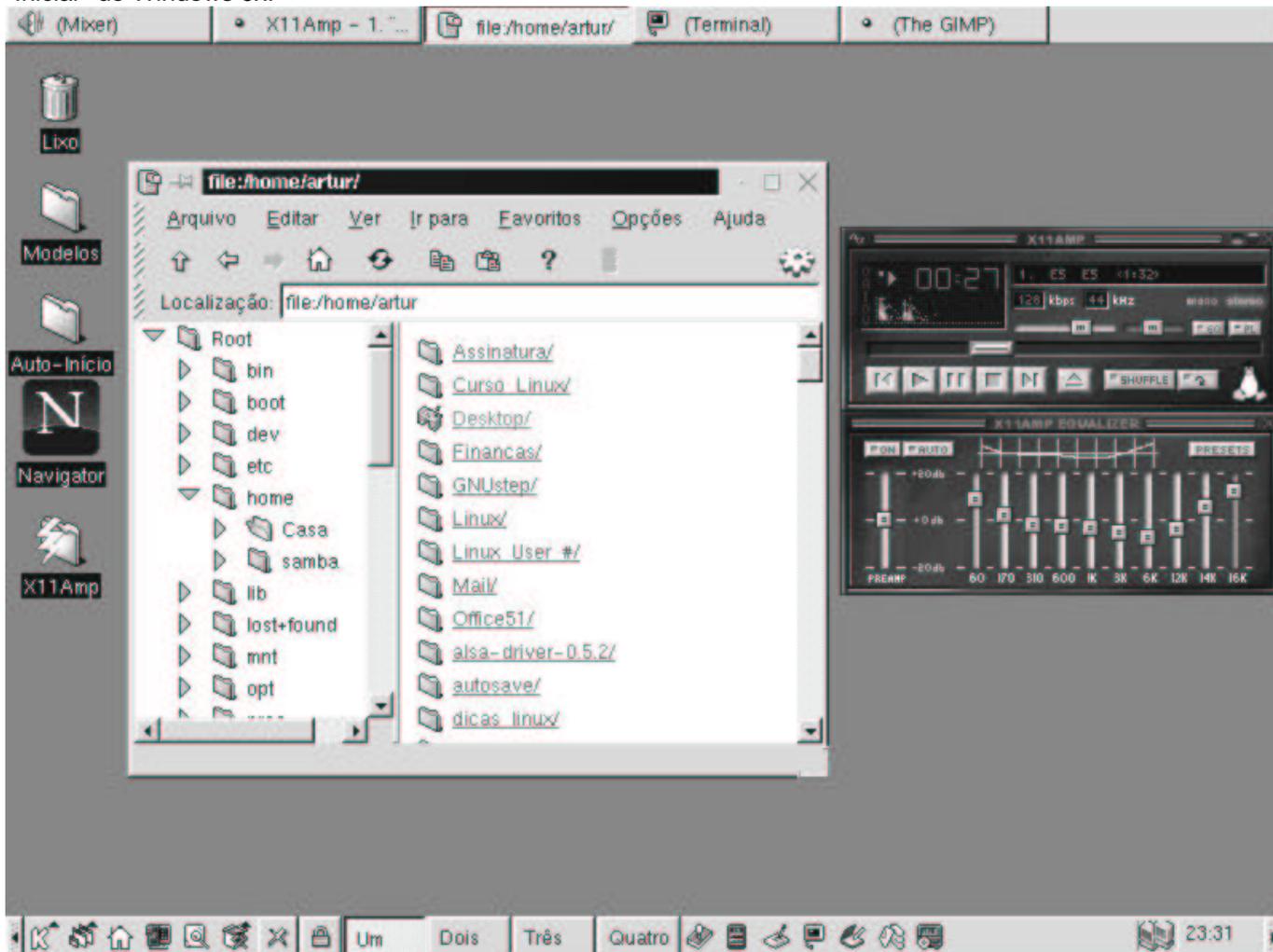
Na figura a seguir, ilustramos uma tela do KDE. Conforme podemos observar, o KDE possui uma barra com ícones, chamado painel (abaixo), uma barra de tarefas (acima), além de diversos ícones, aplicações (X11Amp) e gerenciador de arquivos na área de trabalho.

A utilização do KDE é bastante simples, também muito semelhante ao Windows 9x. Os ícones, tanto da área de trabalho quanto do painel, são acessados por um clique único do botão esquerdo do mouse. As janelas podem ser maximizadas, desmaximizadas, minimizadas ou fechadas utilizando os botões correspondentes na parte superior direita da janela (barra de título). Quando minimizadas, as janelas vão para a barra de tarefas (na figura, mostrada na parte superior da janela). As janelas também podem ser maximizadas ou minimizadas clicando-se com o botão direito do mouse sobre o ícone correspondente na barra de tarefas. Clicando-se com o botão do meio do mouse em uma área livre no desktop, abre-se uma janela com a lista de aplicações abertas, permitindo alternar entre elas (maximizá-la). A combinação das teclas Alt + Tab também permite alternar entre as janelas de aplicações abertas, como no Windows 9x. No painel do KDE existem também alguns ícones que correspondem às diversas áreas de trabalho que podem ser utilizadas neste gerenciador.

Curso de Linux Básico

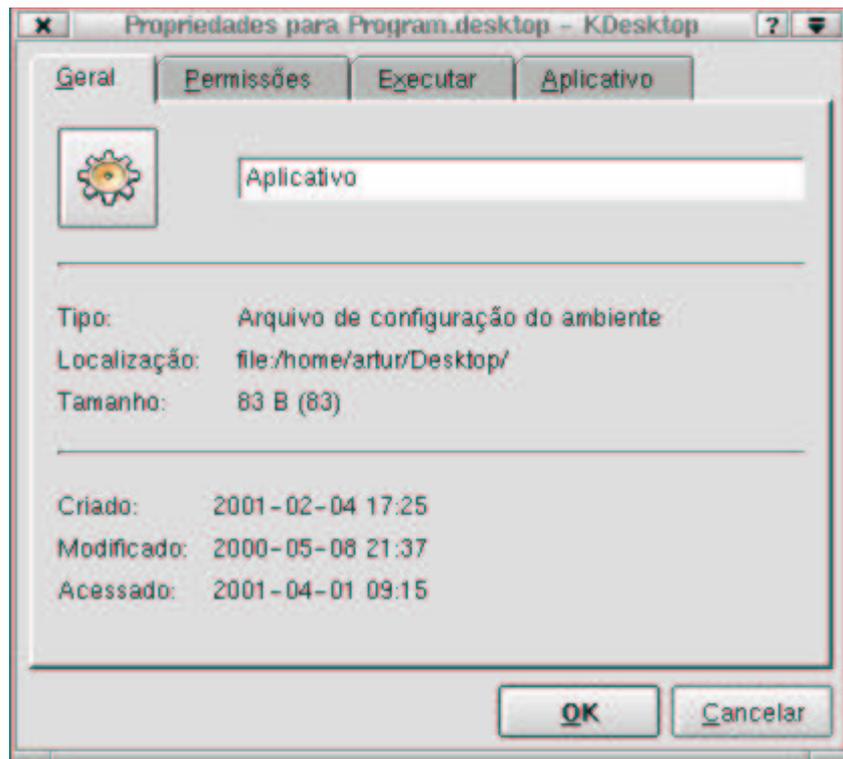
O KDE possui um gerenciador de arquivos muito amigável, o Kfm (no KDE1) ou o Konqueror (no KDE2), sendo que o primeiro é mostrado na figura acima. Ambos podem ser configurados, através de seus menus, para exibir as informações dos arquivos e diretórios de diversas formas. As pastas podem ser abertas através de um único clique com o botão esquerdo do mouse, os arquivos podem ser abertos também com um único clique com o botão esquerdo do mouse, desde que o sistema reconheça qual aplicativo deve ser utilizado para abri-los.

As aplicações e programas que não possuem ícones na área de trabalho ou no painel poderão ser encontrados acessando o ícone em forma de **K**, na extrema esquerda do painel. Podemos dizer que ele corresponde ao "Iniciar" do Windows 9x.



O KDE é totalmente configurável através de seus próprios recursos, os quais podem ser acessados pelos ícones disponíveis no painel. O próprio painel pode ser configurado clicando-se com o botão direito do mouse sobre uma área livre de ícones do painel e escolhendo a opção *configurar* do menu.

Para criar um novo ícone na área de trabalho, basta clicar com o botão direito do mouse sobre uma área livre da mesma e selecionar *criar novo* no menu. Normalmente deve-se escolher *aplicativo* no próximo menu, porém outras opções estão disponíveis. Será apresentada uma tela semelhante a esta:



Na aba *Geral* deve-se trocar a palavra *Aplicativo* pelo nome que se deseja atribuir ao ícone sendo criado e confirmar com o botão OK. Se desejar alterar o ícone que será apresentado na tela, selecione outro, clicando no ícone em forma de engrenagem. Após tudo configurado, basta confirmar com OK. Em seguida, devemos selecionar a guia *Executar* e no campo onde está o cursor, deve-se informar o nome da aplicação ou programa desejado (ou utiliza-se o botão *Explorar* para pesquisar o sistema de arquivos).

Para adicionar um ícone ao painel, basta utilizar no menu a opção menu do *Painel*, *Adicionar Aplicativo* e selecionar a aplicação desejada dentre as opções do menu.

O painel pode ser escondido ou mostrado na tela, fazendo uso das setas nas suas extremidades.

Assim como outros gerenciadores de janela, o KDE também possui múltiplas áreas de trabalho, as quais podem ser alternadas clicando-se com o botão esquerdo do mouse sobre os respectivos botões no painel. Por padrão, as áreas de trabalho têm os nomes na sequência 1, 2, 3, etc. Porém estes nomes podem ser alterados na configuração do painel.

Uma ferramenta de configuração do KDE é o Centro de Controle KDE, que pode ser acessado via menu ou via painel. Ele permite modificar praticamente tudo neste gerenciador de janelas, além de fornecer inúmeras informações sobre o sistema.

O KDE possui ainda um recurso de travar a tela, o qual pode ser utilizado através da opção *Bloquear Tela*, no menu. Quando acionado, a proteção de tela, se estiver definida, é ativada. Se não houver uma proteção de tela definida, a tela ficará escura. O retorno à tela do KDE é feito mediante a digitação da senha do usuário.

18 – INSTALAÇÃO DE PROGRAMAS

Existem diversas maneiras de se instalar programas no Linux. Podemos dizer basicamente que elas se dividem em duas formas: baseadas em código fonte e baseadas em pacotes.

A primeira delas, mais tradicional e ausente na maior parte dos demais sistemas operacionais, é aquela onde se dispõe do código fonte do programa e a instalação consiste em compilar este código, gerando o código objeto (também chamado de binário ou executável). Muitas vezes, junto com o código fonte é fornecido um script (semelhante a um arquivo de lote .bat do DOS) para facilitar a compilação do programa. Geralmente, isto é feito utilizando-se a seguinte sequência de comandos:

```
./configure  
make  
make install
```

A segunda forma é baseada em pacotes, ou seja, o programa já compilado (código objeto ou executável) é "empacotado", junto com os demais arquivos que sejam necessários para seu funcionamento. Existem diversos tipos de pacotes, como rpm (usado pela Red Hat e derivados), deb (Debian e derivados) e tgz (Slackware).

Aqui vale uma observação: existe uma certa confusão entre os formatos tgz e tar.gz. O primeiro é o formato dos pacotes de programas utilizado pelo Slackware, enquanto o segundo são apenas arquivos agrupados pelo comando *tar* e compactados com *gzip*.

Praticamente todas as principais distribuições de Linux utilizam algum programa gerenciador de pacotes, existindo também programas para conversão de um tipo de pacote em outro. Exemplos de gerenciadores de pacotes: RPMDrake, GnoRPM, pkgtool, etc.

Exemplos de conversores de pacotes: alien, rpmtotgz, etc.

19 – ARQUIVAMENTO

A cópia de segurança de arquivos (backup) é uma tarefa muito importante em redes e também nas aplicações onde máquinas individuais necessitam preservar informações (dados) de valor.

O processo de backup mais comum é o que utiliza cópias simples, a qual consiste em copiar tudo uma única vez e nas próximas vezes, copiar apenas os arquivos que foram alterados após a cópia inicial. A primeira cópia é chamada cópia total ou completa, enquanto as seguintes são chamadas cópias incrementais.

Além deste, existe o processo de backup que utiliza cópias em diversos níveis (multinível), sendo mais adequado para tarefas mais complexas.

O que deve ser arquivado? Esta pergunta deve ser respondida pelo usuário da máquina ou pelo administrador da rede, baseado em seu conhecimento sobre o conteúdo armazenado em cada máquina. Porém, é bastante claro que devem ser copiados os arquivos de usuários (normalmente em /home) e os arquivos de configuração do sistema (normalmente em /etc, podendo haver outros arquivos espalhados por todo o sistema de arquivos).

O Linux dispõe de alguns comandos que podem ser utilizados para gerar e restaurar cópias de arquivos (backup), sendo os mais comuns o *tar*, *cpio* e *dump*. Eles permitem criar arquivos de backup em fitas, disquetes, discos rígidos e outros formatos de mídia.

Uma forma de diminuir o espaço gasto na mídia com os arquivos de backup é a compactação dos mesmos. Porém a compactação do backup, além de tornar o processo mais lento, pode contribuir para que todo o arquivo seja inutilizado, caso haja ocorrência de um grande número de erros no backup compactado.

A restauração da cópia de segurança pode ser feita de maneira completa (restaura todos os arquivos do backup) ou parcial, bastando especificar os nomes dos arquivos desejados ao executar o comando para restauração.

20 – COMANDOS BÁSICOS VI

Neste capítulo veremos alguns comandos utilizados para a instalação, atualização, desinstalação e consulta de pacotes e também o comando utilizado para backup do sistema.

20.1 – rpm

O rpm é um poderoso gerenciador de pacotes, que pode ser utilizado para instalar, consultar, atualizar e desinstalar pacotes de software. Além destas funções, que serão abordadas aqui, o rpm também pode fazer verificação, validação de assinatura, construção, reconstrução do banco de dados, ajustar permissões, ajustar donos e grupos e exibir configuração. Cada um destes modos básicos de operação aceita um conjunto diferente de opções.

20.1.1 – Consulta:

A sintaxe geral para o modo de consulta é:

```
rpm -q [opções-de-consulta]
```

Há dois subconjuntos de opções de consulta: seleção de pacotes e seleção de informações.

Opções de seleção de pacotes:

- <nome_do_pacote> Consulta o pacote instalado de nome <nome_do_pacote>.
- a Consulta todos os pacotes instalados.
- f <arquivo> Consulta o pacote do qual <arquivo> faz parte.
- p <arquivo_pacote> Consulta um arquivo de pacote (desinstalado) de nome <arquivo_pacote>.

Opções de seleção de informações:

- i Exibe informações sobre o pacote, incluindo nome, versão e descrição.
- R Lista os pacotes dos quais este depende (o mesmo que --requires).
- provides Lista as capacidades que este pacote fornece.
- changelog Exibe informações sobre as mudanças neste pacote.
- l Lista os arquivos contidos no pacote.
- s Exibe os estados dos arquivos no pacote (implica -l). O estado de cada arquivo é normal, não instalado (uninstalled), ou substituído (replaced).
- d Lista apenas os arquivos de documentação (implica -l).
- c Lista apenas os arquivos de configuração (implica -l).
- scripts Lista os scripts de shell do pacote, que são usados no processo de instalação e desinstalação, se existirem.

Curso de Linux Básico

Exemplos:

```
rpm -qa | less
rpm -qa | grep less
rpm -qf /usr/bin/less
rpm -qi less-332-9cl
rpm -q less-332-9cl -R
rpm -q glibc-2.1.1-11cl --provides | more
rpm -q less-332-9cl -l
rpm -q less-332-9cl -ld
rpm -q less-332-9cl -lc
```

20.1.2 – Instalação:

A sintaxe geral para o modo de instalação é:

```
rpm -i [opções-de-instalação] <arquivo_pacote>
```

Opções de instalação:

- `--force` O mesmo que usar `--replacepkgs`, `--replacefiles` e `--oldpackage`.
- `-h`, `--hash` Exibe 50 caracteres # (hash) à medida que o arquivo é descompactado. Usar em conjunto com `-v` para uma exibição interessante.
- `--oldpackage` Permite que uma atualização substitua um pacote por uma versão anterior.
- `--percent` Exibe porcentagens à medida que os arquivos são descompactados.
- `--replacefiles` Instala os pacotes mesmo que eles substituam arquivos de outros pacotes, já instalados.
- `--replacepkgs` Instala os pacotes mesmo que alguns deles já estejam instalados no sistema.

- `--allfiles` Instala ou atualiza todos os arquivos do pacote que estão faltando, independente deles existirem ou não.
- `--nodeps` Não verifica as dependências antes de instalar ou atualizar um pacote.
- `--noscripts` Não executa os scripts de pré ou pós instalação.
- `--excludedocs` Não instala nenhum arquivo marcado como documentação (o que inclui as páginas de manual e documentos texinfo).
- `--test` Não instala o pacote, apenas verifica e avisa sobre possíveis conflitos.

Exemplos:

```
rpm -ivh --test tree-1.2-8cl.rpm
rpm -i --percent tree-1.2-8cl.rpm
rpm -ivh nt-1.06-1.rpm
rpm -ivh xcircuit-2.0a11-1.i386.rpm
```

20.1.3 – Atualização:

Quando usada a sintaxe de atualização do rpm, ele instala ou atualiza o pacote atualmente instalado para a versão do novo RPM. Isso é o mesmo que instalar, exceto que todas as versões anteriores dos pacotes serão removidas do sistema após a atualização.

A sintaxe geral para o modo de atualização é:

```
rpm -U [opções-de-instalação] <arquivo_pacote>
```

As opções de instalação são as mesmas descritas acima, no item instalação.

Exemplos:

```
rpm -Uvh man-1.5g-7cl.i386.rpm
```

20.1.4 – Desinstalação:

A sintaxe geral utilizada para desinstalação é:

```
rpm -e <nome_do_pacote>
```

Opções de desinstalação:

- `--allmatches` Remove todas as versões do pacote que casarem com <nome_do_pacote>. Normalmente um erro é exibido se <nome_do_pacote> casar com múltiplos pacotes.
- `--noscripts` Não executa os scripts de pré e pós desinstalação.
- `--nodeps` Não verifica se dependências serão quebradas antes de desinstalar o pacote.
- `--test` Não desinstala nada, apenas simula todos os movimentos.

Exemplos:

```
rpm -evv tree-1.2-8cl --test
rpm -evv tree-1.2-8cl
rpm -e xcircuit-2.0a11-1.i386.rpm
```

20.2 – tar

O **tar** é um programa de arquivamento desenvolvido para armazenar e extrair arquivos de um arquivo **tar** (que contém os demais) conhecido como tarfile. O tarfile pode ser construído em uma fita magnética, ou também, o que é comum, gravar-se um tarfile em um arquivo normal.

O primeiro argumento para **tar** deve ser uma das seguintes opções: `Acdrdux`, seguido por uma das seguintes funções adicionais.

Os argumentos finais do **tar** são os nomes dos arquivos ou diretórios nos quais eles podem ser arquivados. O uso de um nome de diretório implica sempre que os subdiretórios sob ele serão incluídos no arquivo.

Curso de Linux Básico

Sintaxe:

```
tar [opções] arquivo1 [ arquivo2, ... arquivoN ] diretório1 [ diretório2, ...diretórioN ]
```

Uma das seguintes opções deve ser usada:

- A Anexar os arquivos tar a um arquivo
- c Criar um novo arquivo tar
- d Encontrar as diferenças entre um arquivo tar e um sistema de arquivos
- delete Apagar do arquivo tar (não pode ser usado para fitas magnéticas!)
- r Anexar arquivos ao final do arquivo tar
- t Lista o conteúdo de um arquivo tar
- u Somente anexa arquivos mais novos que a cópia presente no arquivo tar
- x Extrai arquivos de um arquivo tar

Opções adicionais:

- atime-preserve Não altera a data de acesso dos arquivos copiados
- b, --block-size N Tamanho do bloco Nx512 bytes (padrão N=20)
- B, --read-full-blocks Redefine o tamanho do bloco enquanto lê (para leitura de pipes 4.2.BSD)
- C, --directory DIR Mudar para o diretório DIR
- checkpoint Imprimir os nomes dos diretórios enquanto lê o arquivo tar
- f, --file [HOSTNAME:]F Usar o arquivo file ou o dispositivo F (padrão /dev/rmt0)
- force-local Arquivo tar será local mesmo que tenhas vírgulas
- G, --incremental Cria/lista/extrai no formato GNU antigo de cópia de segurança incremental.
- g, --listed-incremental F Cria/lista/extrai no formato GNU novo de cópia de segurança incremental.
- h, --dereference Não copia ligações simbólicas, mas sim os arquivos que elas apontam.
- i, --ignore-zeros Ignorar blocos com zeros no arquivo tar (normalmente significam fim de arquivo)
- ignore-failed-read Não finalizar com status diferente de zeros quando houver arquivos que não possam ser lidos
- k, --keep-old-files Mantém os arquivos existentes, não regrava a partir do arquivo tar
- K, --starting-file F Começa no arquivo F do arquivo tar
- l, --one-file-system Manter-se no sistema de arquivos local ao criar um arquivo tar
- L, --tape-length N Muda a fita após gravar N*1024 bytes
- m, --modification-time Não extrair a data de modificação dos arquivos

Curso de Linux Básico

-M, --multi-volume	Cria / lista / extrai arquivos multivolumes
-N, --after-date DATA, --newer DATA	Somente armazena arquivos mais recentes que DATA
-o, --old-archive, --portability	Grava o arquivo no formato V7, ao invés do formato ANSI
-O, --to-stdout	Extrai arquivos para a saída padrão
-p, --same-permissions, --preserve-permissions	Extrai todas as informações de proteção
-P, --absolute-paths	Não retirar os caracteres '/' do início do nome dos arquivos
--preserve	Similar -p -s
-R, --record-number	Mostra o número do registro dentro do arquivo tar em cada mensagem
--remove-files	Remover os arquivos após adicioná-los ao arquivo tar
-s, --same-order, --preserve-order	Ordena a lista de nomes a serem extraídos para comparar com o arquivo tar
--same-owner	Cria os arquivos extraídos com a mesma propriedade.
-S, --sparse	Manuseia arquivos segmentados eficientemente
-T, --files-from F	Obtém os nomes a serem extraídos ou criados no arquivo F
--null	-T lê nomes terminados com caracter nulo, desabilita -C
--totals	Lista o total de bytes gravados com --create
-v, --verbose	Mostra a lista dos arquivos processados
-V, --label NOME	Cria um arquivo tar com o nome de volume igual a NOME
-w, --interactive, --confirmation	Solicita uma confirmação para cada ação
-W, --verify	Tenta verificar um arquivo após gravá-lo
--exclude ARQUIVO	Excluir arquivo ARQUIVO
-X, --exclude-from ARQUIVO	Excluir arquivos listados em ARQUIVO
-Z, --compress, --uncompress	Filtra o arquivo através de compactação
-z, --gzip, --ungzip	Filtra os arquivos através de gzip
--use-compress-program PROG	Filtra os arquivos através de PROG (o qual deve aceitar -d)
--block-compress	Bloco de saída de programas de compactação de fitas magnéticas
-[0-7][lmh]	Especifica o dispositivo e a densidade.

Exemplos:

```
tar -cf arquivo.tar /home
tar -czf arquivo.tar.gz /home
tar -tf arquivo.tar
tar -tzf arquivo.tar.gz
tar xvzf xcircuit-2.0b1.tar.gz
tar xvf arquivo.tar --directory backup arquivo_a_extrair
```

21 – COMANDOS BÁSICOS VII

Neste capítulo veremos dois comandos utilizados para gerenciamento de processos no Linux, o ps e o kill.

21.1 – ps

O comando **ps** relata a situação dos processos, fornecendo uma imagem dos processos atuais. Caso se deseje uma atualização repetitiva da situação, deve-se usar o comando top.

Sintaxe:

```
ps [-] [lujsvmaxScewhrnu] [ttx] [O[+|-]k1[[+|-]k2...]] [pids]
```

Descrição das opções longas no estilo GNU:

Opção	Descrição
l	Formato longo
u	Formato de usuário: fornece o nome do usuário e o horário de início do processo
j	Formato de processos: pgid sid
s	Formato de sinal
v	Formato vm
m	Lista informações de memória (combinado com o indicador p pode informar o número de páginas).
f	Formato de árvores da família na linha de comando
a	Mostra também processos de outros usuários
x	Mostra processos sem controle de terminal
S	Adiciona o tempo de processamento e falhas nas páginas dos processos filhos.
c	Nome do comando a partir de task_struct
e	Mostra o ambiente após a linha de comando e ' + '
w	Saída larga: não trunca as linhas de comandos para que caibam em uma única linha. Para ser exato, cada w que é especificado irá adicionar outra linha possível na saída. Caso o espaço não seja necessário, ele não será utilizado. Pode-se ter até 100 w's.
h	Sem cabeçalhos
s	Somente os processos em execução
n	Saída em formato numérico para USER e WCHAN.
ttx	Somente processos controlando terminais tty xx; para xx pode-se usar ou o nome do dispositivo em "/dev" ou o nome com o qual tty é dividido. Este é o reverso heurístico do processo usado por ps para listar o nome abreviado no campo TT, e.g. ps -t1.

Opção	Descrição
O[+ -]k1[, [+ -]k2[, ...]]	Ordena a lista de processos de acordo com a ordem multinível especificada na seqüência de chaves curtas em SORT KEYS, k1, k2,... Especificações padrão de ordem existem para cada um dos vários formatos de ps. Eles podem ser sobrepostos por uma ordem especificada pelo usuário. O sinal '+' é opcional, e significa somente a direção padrão de uma chave. O sinal '-' reverte a direção da chave que seja precedida por ele. Assim como t e pids, a opção O deve ser a última opção em um argumento simples de comando, mas especificações em argumentos sucessivos podem ser concatenados.
pids	Lista somente os processos especificados; separados por vírgulas. A lista deve ser informada imediatamente após a última opção em um argumento de linha de comando, sem espaços, como por exemplo ps -j1,4,5. Listas especificadas em argumentos subseqüentes serão tratadas de forma diferenciada, por exemplo ps -l 1,2,3,4,5,6 irá listar todos os processos de 1 a 6 no formato longo. Caso as identificações dos processos sejam informadas, eles serão listados, não importa de que forma. Caso um terminal tty seja informado, todos os processos associados a ele serão listados. Estas opções sobrepõem-se aos indicadores 'a' e 'x'.

O campo **STAT**, que aparece na lista gerada pelo comando ps, possui o seguinte significado: O primeiro campo será R para em execução, S para aguardando, D para aguardando sem interrupção, T para parado ou em depuração, ou Z para um processo zumbi. O segundo campo contém W caso o processo não tenha páginas residentes. O terceiro campo será N caso o processo tenha um valor de prioridade positivo.

NOTA:

%CPU mostra o percentual de cputime/realtime. Não deve ser superior a 100%. É igual ao tempo usado dividido pelos processos que estão sendo executados.

Exemplos:

```
ps ux
ps fx
ps aux
ps awx
```

21.2 – kill

O comando **kill** permite finalizar (matar) um processo em execução. **kill** envia um sinal específico para um determinado processo. Caso nenhum sinal seja especificado, o sinal TERM é enviado. Este sinal irá finalizar processos que esperam este tipo de mensagem. Para outros processos, pode ser necessário usar o sinal KILL (9), uma vez que este sinal não pode ser ignorado.

Sintaxe:

```
kill [ -s sinal | -p ] pid ...
```

```
kill -l [ sinal ]
```

Curso de Linux Básico

Opções:

Opção	Descrição
pid ...	Especifica a lista de processos para os quais kill deve sinalizar. Cada pid pode ser um entre quatro opções: Um nome de processo no qual o processo nomeado receberá o sinal. n onde n é maior que 0. O processo com o pid (número de identificação) n receberá o sinal. -1 onde todos os processos de MAX_INT a 2 receberão o sinal, se permitido pelo dono do processo. -n onde n é maior que 1, e todos os processos do grupo n receberão o sinal. Caso um sinal negativo seja informado, o sinal obrigatoriamente deve ser especificado antes, de outra forma será interpretado como o sinal a ser enviado.
-s	Especifica o sinal a ser enviado. O sinal pode ser informado como um dígito ou como um número.
-p	Especifica que kill pode somente listar a identificação do processo (pid) do processo nomeado, e não deve enviar-lhe um sinal.
-l	Lista uma relação dos nomes de sinais. Eles podem ser encontrados em /usr/include/linux/signal.h

Exemplos:

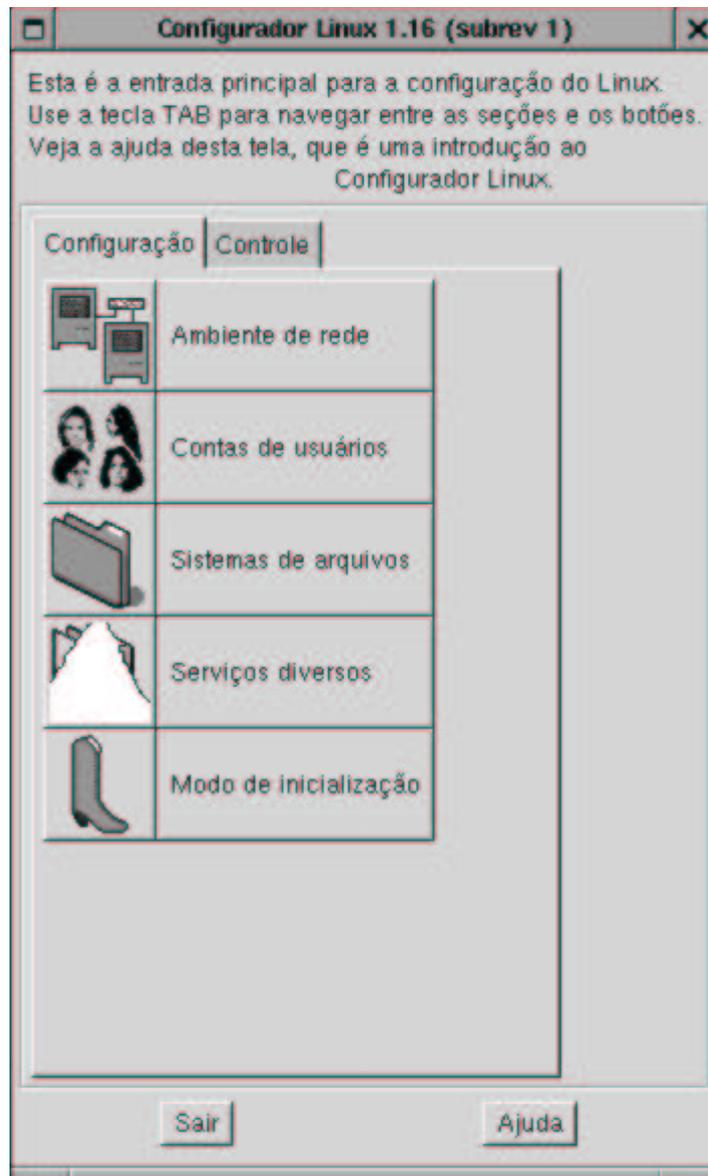
```
kill -l  
kill -p 572  
kill -9 572
```

22 – MODO GRÁFICO (X) II

Neste capítulo trataremos de algumas ferramentas disponíveis no modo gráfico, na sua maioria utilizadas para a configuração do sistema ou de seus periféricos, como impressoras e modems. Veremos também como configurar uma conexão com a internet através de um modem e como configurar o KMail, um leitor de correio eletrônico em modo gráfico.

22.1 – O Configurador Linuxconf

O Linuxconf é um programa criado para ser um configurador do sistema Linux, podendo ser utilizado tanto no modo texto quanto no modo gráfico, já que possui front-end gráfico. Sua tela inicial (gráfica) pode ser observada na figura abaixo (pode mudar conforme a versão utilizada):



O Linuxconf é basicamente um aplicativo destinado à configuração e administração do sistema.

- Em Ambiente de Rede é possível configurar todo um ambiente de redes, como DNS, roteadores, conexões ppp e slip.
- Em Contas de Usuários, podemos manipular contas de usuários, alterar senhas e configurar contas especiais, além de podermos definir uma política de senhas e contas.
- Em Sistemas de Arquivos podemos definir as partições que queremos ter acesso, bem como alterar outros parâmetros do sistema de arquivos.
- Em Serviços Diversos temos acesso a configurações de modems e impressoras, além de podermos visualizar os registros do sistema.
- Em Modo de Inicialização, teremos acesso às configurações do lilo e também podemos alterar o modo de inicialização padrão.
- Em Grub boot loader teremos acesso às configurações do gerenciador de boot Grub (alternativa gráfica ao Lilo).
- Em Painel de Controle, podemos executar diversas operações no sistema, como reinicializar a máquina, arquivar as configurações, ativar ou desativar uma conexão discada, montar ou desmontar sistemas de arquivos, entre outras tarefas.
- Em Arquivos de Controle e Sistemas, podemos configurar os comandos e programas residentes, alterar os arquivos de configuração do sistema e inclusive configurar os módulos do próprio Linuxconf.
- Em Data & Horário, podemos facilmente configurar a data e a hora do sistema.
- Em Características, temos acesso à configuração do mapa do teclado e do idioma, entre outros itens.
- Em Gerenciamento de Pacotes RPM, podemos realizar tudo o que o comando rpm faz, em um ambiente gráfico.
- Em Registros, podemos verificar as mensagens de sistema, desde que o sistema esteja configurado para gerar tais mensagens.
- Em Ver o Estado do Sistema, podemos obter as mais variadas informações sobre o sistema, como qual o processador utilizado, uso da memória, uso do espaço em disco, quais os processos estão rodando, os mapas de E/S (I/O) e de IRQs, estado das interfaces de rede, módulos do kernel carregados, etc.

Para ser executado, o Linuxconf normalmente exige estar logado como root, devido às tarefas que executa no sistema. Portanto, um usuário comum não poderá executar o Linuxconf.

Um fato importante é que o Linuxconf permite a inclusão de módulos a si próprio, como por exemplo, os módulos configuradores do Apache (servidor http), do Sendmail (servidor de e-mail), do squid (servidor proxy), entre outros.

Um ponto que deve ser lembrado é que apesar das facilidades de uso de ferramentas de configuração, tais como o Linuxconf, estes têm como desvantagem a ausência de vínculo com os programas que configuram, podendo ocorrer problemas se houver uma mudança na forma de configuração devido a novas versões dos programas configurados, as quais podem não ser tratadas corretamente pelo configurador.

22.2 – Configuração de Impressora com o Linuxconf

Selecione na aba Configuração do Linuxconf o botão Serviços Diversos e em seguida Impressora. Clique no botão Adicionar / Editar impressoras. Para adicionar uma nova impressora, clique no botão adicionar.

Devemos definir o nome da impressora, que normalmente deve ser o nome da porta onde está conectada. Por exemplo: lp0, que corresponde a LPT1 do DOS / Windows.

Devemos definir também como está conectada (normalmente local). Após clicar em Aceitar e OK, devemos clicar sobre seu nome para definir suas propriedades, como tamanho da fila de impressão, porta onde está conectada, o tipo de filtro correspondente à impressora (responsável por transformar o conteúdo a ser impresso para a linguagem da impressora), etc.

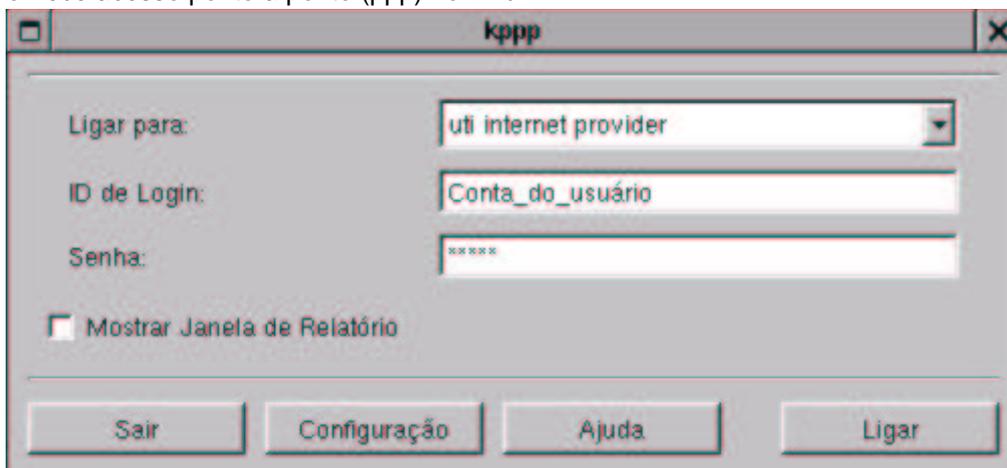
A partir desta tela também é possível executar um teste da impressora instalada.

22.3 – Acesso à Internet / E-Mail

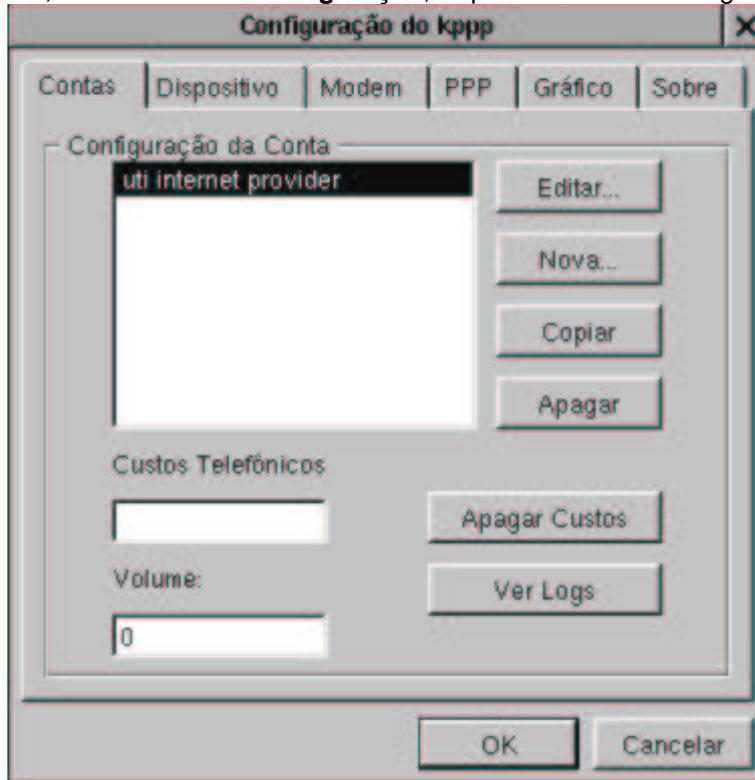
Dois dos serviços básicos desejados pela maioria dos usuários de micros desktop são o acesso à Internet e à correspondência eletrônica (e-mail). O Linux oferece diversas opções para os dois casos, porém aqui abordaremos apenas o Kppp e o Kmail, pois são dois utilitários simples de configurar e bastante amigáveis para este perfil de usuário.

22.3.1 – Configuração de Acesso Discado com o Kppp:

A figura abaixo ilustra a tela inicial do Kppp, que nada mais é que um front-end gráfico para o pppd, um dos daemons que fornece acesso ponto a ponto (ppp) no Linux:



Para configurar as conexões, basta clicar em **Configuração**, o qual abrirá a tela a seguir:

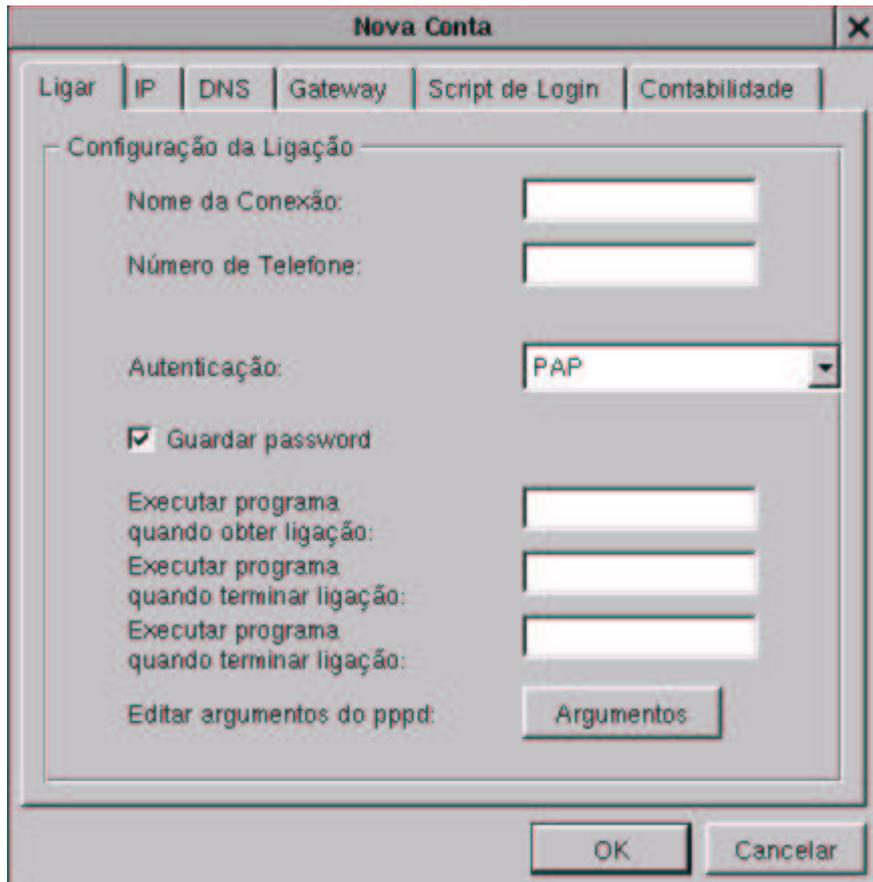


Para adicionar uma nova conta, basta clicar em Nova, e preencher os campos das telas mostradas a seguir:

Em Nome da Conexão deve-se colocar o nome de identificação da conta sendo criada, e em Número do Telefone deve ser informado o número do provedor de acesso desejado.

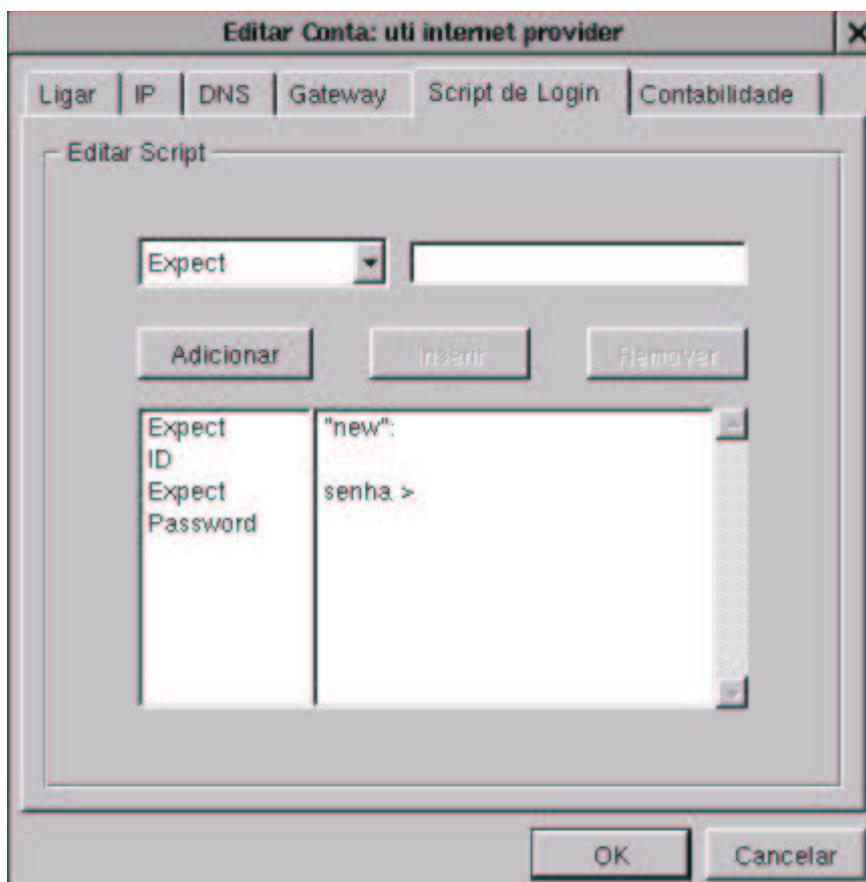
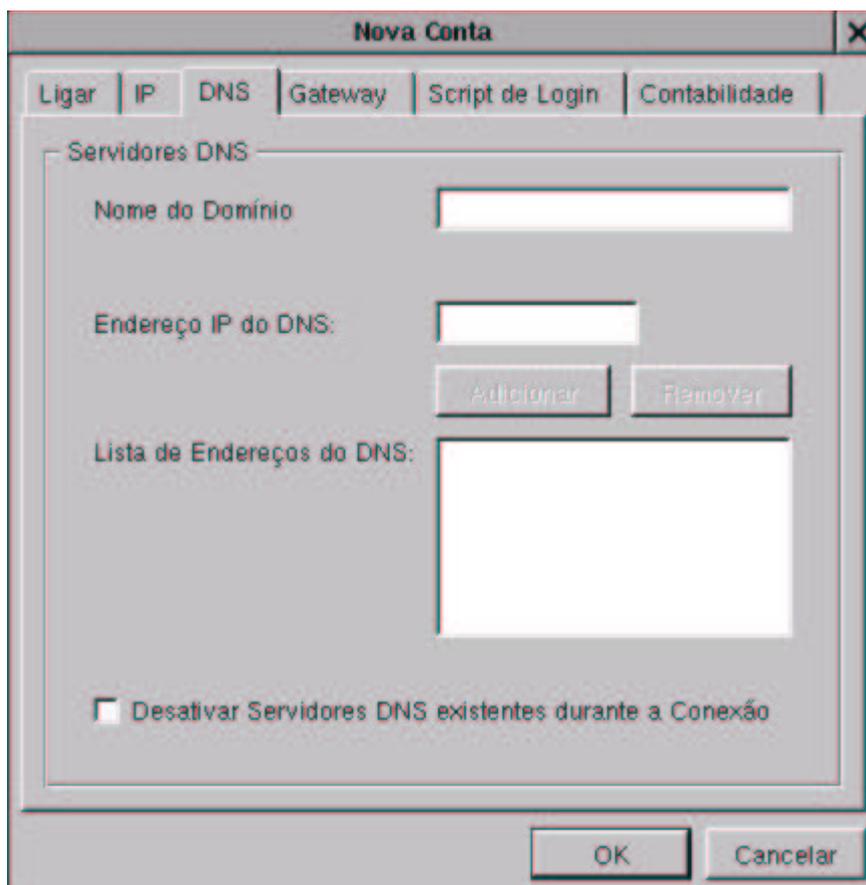
No campo Autenticação, deve ser escolhido um tipo compatível com o que é utilizado pelo provedor em questão. O mais utilizado atualmente é o PAP. Porém, em caso de dúvidas deve-se entrar em contato com o suporte técnico do provedor de acesso para obter esta informação.

Pode-se também selecionar a opção Guardar Password para que o sistema guarde a senha cadastrada, evitando ter que digitá-la a cada conexão.



Na aba IP, deve ser configurado conforme a situação. Se o provedor de acesso lhe fornece um número diferente de IP a cada conexão, selecione Endereço IP Dinâmico, caso contrário selecione a opção Endereço IP Estático e forneça este número, bem como a máscara de subrede. Se houver dúvidas quanto a estas informações, procure obtê-las junto ao suporte técnico do provedor de acesso. A opção **Auto-configurar nome da host a partir deste IP** deve ficar desmarcada.

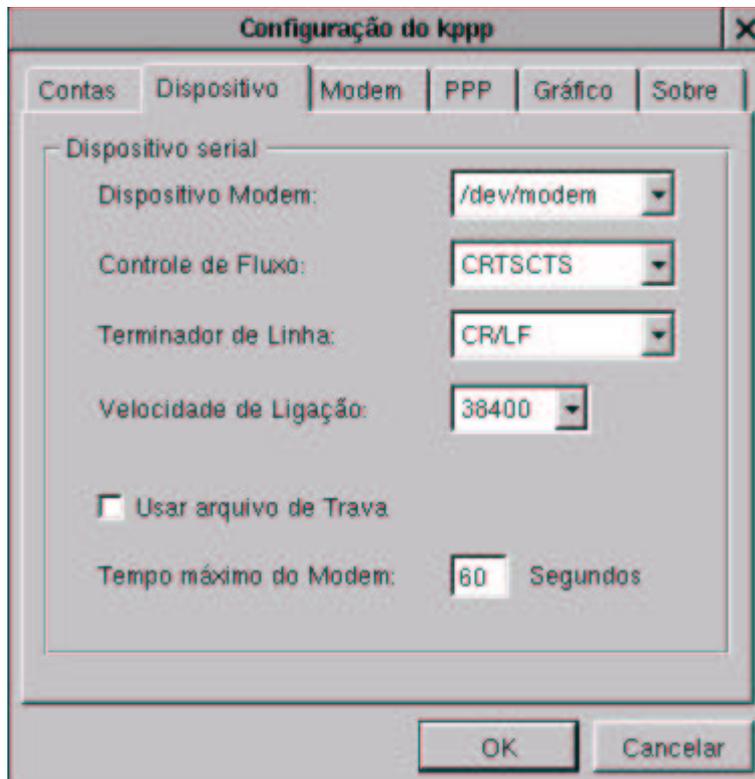
Na figura a seguir, devemos configurar o servidor DNS. Para isso, devemos informar o Nome do Domínio e adicionar os Endereços IP do DNS. Estas informações devem ser obtidas junto ao suporte técnico do provedor de acesso.



Curso de Linux Básico

Na figura anterior, há um exemplo simples de script de login, utilizado para conexão com provedores que possuam este tipo de autenticação. Se utilizada autenticação PAP, o script não é necessário.

Na tela da figura a seguir, deve-se especificar as configurações a serem utilizadas pelo modem instalado na máquina. Deve-se observar que /dev/modem é um link simbólico que aponta para o modem instalado em uma porta serial (/dev/cua0, /dev/cua1,... ou /dev/ttyS0, /dev/ttyS1,...).



Além destas configurações, podem ser acrescentadas outras, porém são apenas ajustes para um funcionamento personalizado, que não impedem o estabelecimento da conexão.

Deve-se apenas atentar para a configuração dos comandos do modem (aba Modem), pois se os comandos não forem aceitos pelo modem, será impossível a conexão. Estas informações podem variar de um modem para outro, apesar de que o padrão (default) configurado atende a maioria dos modems convencionais. Em caso de dúvida, consulte a documentação de seu modem.

Uma observação importante é sobre o suporte do Linux a modems baseados em software, como os Winmodems, HSP modems, Softmodems, etc, pois estes pseudo-modems em geral não são suportados pelo Linux devido à arquitetura fechada de seu hardware aliado ao software proprietário utilizado para emular um modem, normalmente disponibilizado pelos fabricantes apenas (infelizmente) para a plataforma Windows.

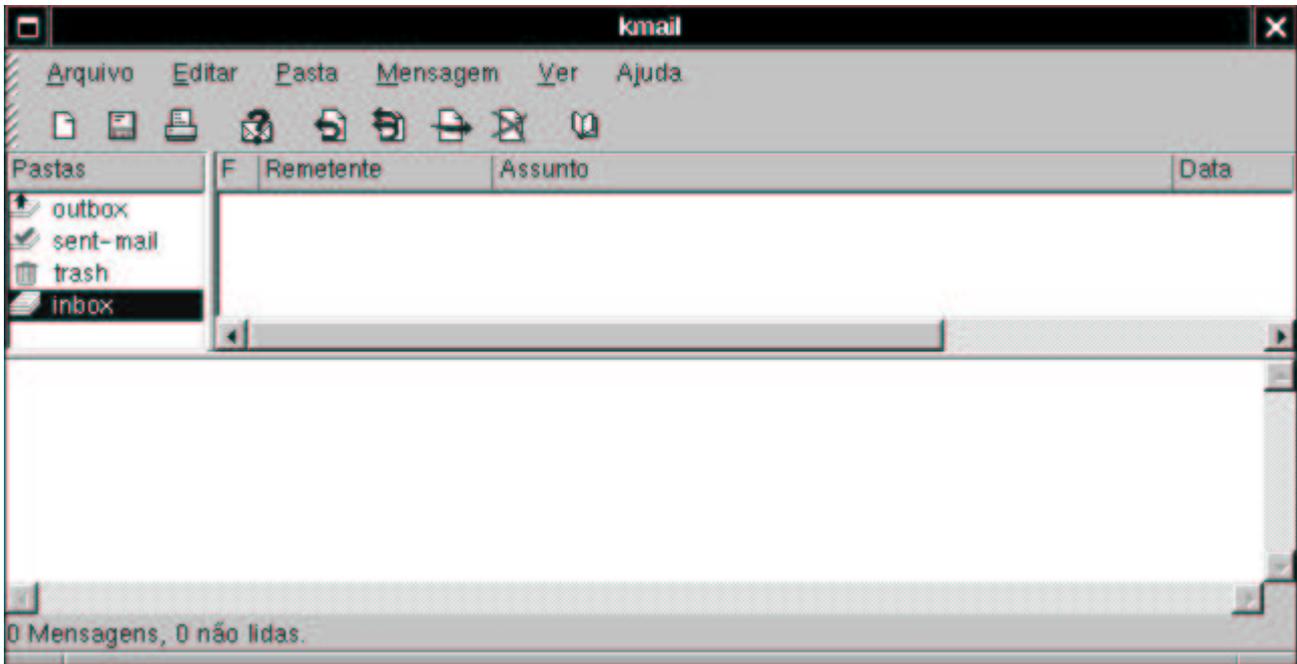
Atualmente, alguns fabricantes de chipsets de modems, como Lucent e PCTEL já liberaram drivers de seus produtos para o Linux, porém ainda estão em fase de amadurecimento. Maiores informações sobre o assunto pode ser obtido na Internet em www.linmodems.org ou www.winmodems.com.

A navegação pela WEB pode ser realizada por qualquer browser disponível para Linux, como o tradicional Netscape, o recente navegador comercial Opera ou os navegadores incluídos nas suítes do KDE (o Konqueror) e do StarOffice.

Além destes, que atuam em modo gráfico, existem outros, para navegação em modo texto: o Lynx, o Links, o W3m, entre outros.

22.3.2 – Configuração do KMail:

A figura abaixo ilustra a tela inicial do Kmail, um utilitário para leitura e envio de e-mail no Linux:



Na primeira vez que o Kmail é executado, será informado que o diretório ~/Mail não existe e será solicitado confirmação para que o próprio Kmail crie este diretório, destinado a armazenar a correspondência. Para isto, basta clicar no botão OK.

Será aberta então a tela ilustrada na Figura 1, a qual possui diversas abas.

Na opção Identidade, devemos informar os dados pessoais de usuário, como exemplificado. Também podemos especificar um arquivo contendo uma assinatura, que nada mais é que um arquivo texto ou gráfico, que poderá ser inserido nas mensagens enviadas, personalizando-as.

Ao clicar na opção Rede, aparecerá uma tela semelhante à ilustrada na Figura 2, que deve ser preenchida conforme mostrado. No campo servidor, localhost deve ser substituído pelo servidor SMTP do provedor.

Ao adicionar uma nova conta será solicitado escolher entre Caixa de Correio Local ou Pop3. Normalmente, deve-se escolher pop3 e clicar no botão OK, conforme a caixa de diálogo ilustrada na Figura 3.

Os dados da conta pop3 devem ser preenchidos conforme ilustra a Figura 4. As informações necessárias para o preenchimento devem ser obtidas do provedor de acesso.

Figura 1:

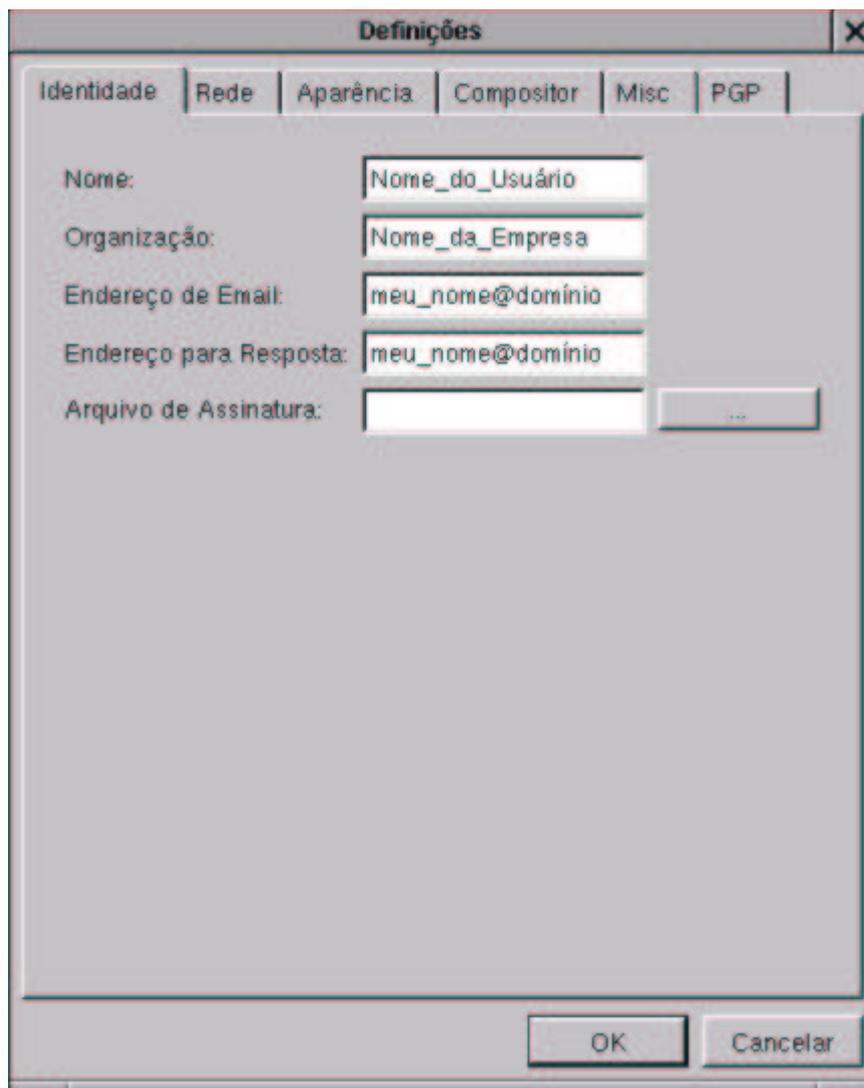


Figura 2:

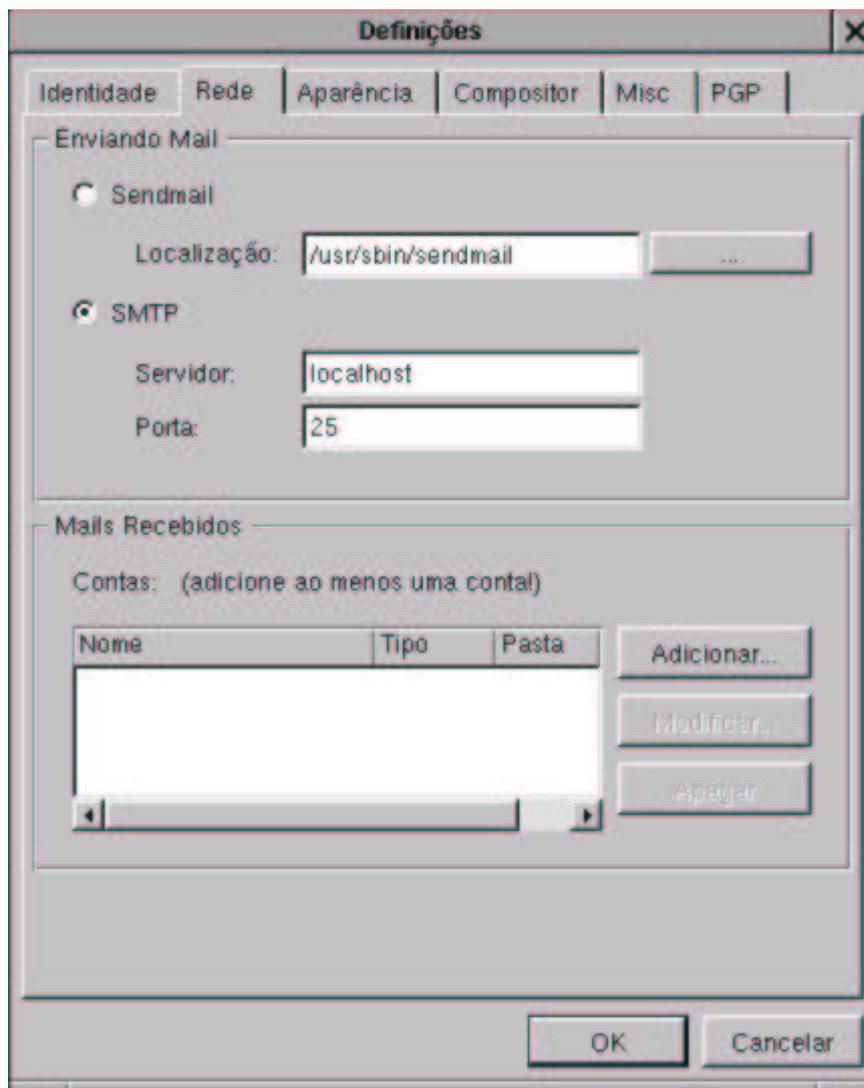


Figura 3:

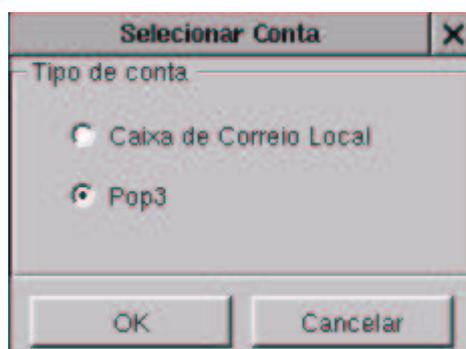
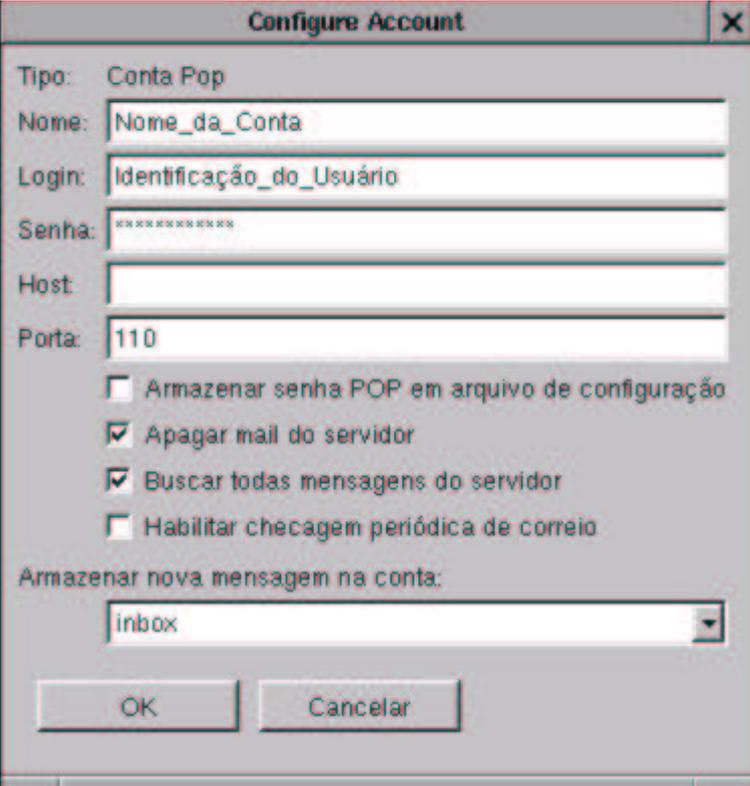


Figura 4:



Configure Account

Tipo: Conta Pop

Nome: Nome_da_Conta

Login: Identificação_do_Usuário

Senha: *****

Host:

Porta: 110

Armazenar senha POP em arquivo de configuração

Apagar mail do servidor

Buscar todas mensagens do servidor

Habilitar checagem periódica de correio

Armazenar nova mensagem na conta:

inbox

OK Cancelar

Além destas opções, diversas outras podem ser configuradas, de modo a personalizar o programa, porém não são essenciais para o seu funcionamento. Basta navegar pelas abas de configuração disponíveis e marcar as opções desejadas.

23 – COMANDOS BÁSICOS VIII

Neste capítulo trataremos dos comandos utilizados para impressão de arquivos e seu gerenciamento no Linux, além de alguns outros comandos variados.

23.1 – >

Este comando é um redirecionador, ou seja, redireciona a saída de um comando para um arquivo.

Exemplo:

```
ls -al /sbin/* > ~/meu_arq
```

O comando acima criará um arquivo contendo o resultado do comando ls.

23.2 – >>

Este comando também é um redirecionador, porém acrescenta a saída de um comando no fim de um arquivo.

Exemplo:

```
ls -al /bin/* >> ~/meu_arq
```

23.3 – &

Este caracter, acrescentado no final de uma linha de comando, informa ao sistema que aquela linha de comando deve ser executada em segundo plano (background). É interessante seu uso, pois libera o prompt, permitindo que outros comandos sejam entrados, enquanto os comandos anteriores são processados em segundo plano pelo sistema.

Exemplo:

Para este exemplo, deve-se estar no modo gráfico e executar o comando abaixo em um terminal virtual (xterm ou outro):

```
/usr/bin/netscape &
```

23.4 – cat

Este comando concatena um arquivo e lista o resultado na saída padrão (normalmente o vídeo). Pode ser utilizado na visualização, cópia e impressão de arquivos como veremos nos exemplos a seguir.

Exemplos:

cat meu_arq more	Lista o conteúdo de <i>meu_arq</i> na tela.
cat meu_arq > seu_arq	Copia o conteúdo de <i>meu_arq</i> para <i>seu_arq</i> .
cat meu_arq > /dev/lp0	Copia <i>meu_arq</i> para a impressora lp0 (impressão direta).

23.5 – lpr

lpr é a interface entre a fila de impressão e os demais processos da máquina. Geralmente uma tarefa de impressão é iniciada com o comando:

```
lpr [-P fila] arquivo_texto
```

Caso se omita o parâmetro `-P` que indica o nome da fila, o padrão será obtido através da variável de ambiente `$PRINTER`. Caso ela não esteja configurada, o nome padrão `lp` será utilizado.

Exemplo:

```
lpr -P lp0 meu_arq
```

23.6 – lpq

O comando **lpq** mostra as tarefas de impressão dos usuários.

Exemplo:

```
lpq -P lp0  
lpq -P lp0 marcos
```

A listagem resultante será algo como:

```
lp0 está pronta e imprimindo  
Ordem  Dono   Tarefa  Arquivo   Tamanho Total  
ativo           marcos   678   texto.txt  428934 bytes  
1st           marcos   679   texto2.txt  859345 bytes  
2nd           marcos   684   texto3.txt  985903 bytes
```

23.7 – lprm

O comando **lprm** remove as tarefas de impressão do usuário na fila. Caso nenhum número de tarefa seja informado, a tarefa ativa ou em impressão no momento será cancelada.

Exemplo:

```
lprm -P lp0 679  
lprm -P lp0 marcos
```

23.8 – lpc

O comando **lpc** inicia uma interação com o superusuário, disponibilizando comandos de administração da impressora, como habilitar, suspender, alterar a ordem de impressão, etc. Mais informações podem ser obtidas com *man lpc*.

Exemplos:

```
#lpc
lpc> status lp0
lpc> topq lp0 684
lpc> topq lp0 marcos
lpc> clean all
lpc> exit
```

23.9 – pwd

O comando **pwd** informa o caminho completo do diretório corrente (atual).

23.10 – who

O comando **who** informa quem está conectado.

Exemplos:

who	Informa todos os usuários conectados.
whoami	Informa sob qual usuário você está conectado.
who -q	Informa quem são os usuários conectados e quantos são.
who -i	Informa quem são os usuários conectados e o tempo que estão ociosos no sistema. "." significa que está ativo e "old", que está ocioso há mais de 24 horas.

23.11 – df

O comando **df** informa quais são os sistemas de arquivos montados e qual a porcentagem de utilização do espaço em disco correspondente a cada um deles.

Exemplo:

df	
df -k	Informa o tamanho em blocos de 1k, se este não for o default.
df -a	Informa inclusive os sistemas de arquivos virtuais (que não ocupam o disco).

23.12 – du

O comando **du** informa o espaço ocupado pelos arquivos ou diretórios. Se não for informado o local, serão exibidas informações do diretório corrente.

Exemplos:

du -b /etc/fstab	Informa o tamanho do arquivo /etc/fstab em Bytes.
du -k /etc/fstab	Informa o tamanho do arquivo /etc/fstab em KBytes.
du -m /etc/fstab	Informa o tamanho do arquivo /etc/fstab em MBytes.
du -h /etc/fstab	Informa o tamanho do arquivo /etc/fstab na unidade mais adequada.
du /etc more	Informa os tamanhos dos arquivos e diretórios localizados em /etc.
du	Informa os tamanhos dos arquivos e diretórios do diretório corrente.

24 – STAROFFICE 5.1

Neste capítulo trataremos da instalação, configuração e utilização do StarOffice 5.1.

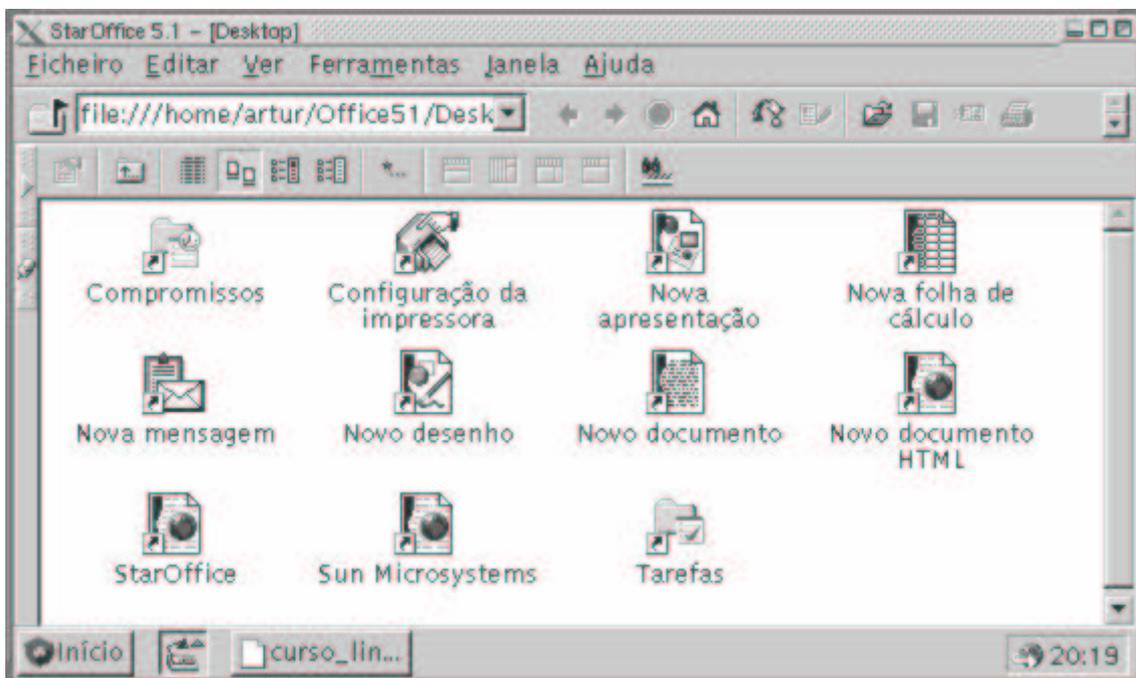
24.1 – O Que É?

O StarOffice é uma suíte para escritório, tal como os pacotes Office da Microsoft, Lotus, Corel, etc. Foi originalmente desenvolvido por uma empresa alemã, a StarDivision, que foi posteriormente adquirida pela Sun.

O StarOffice possui basicamente um editor de textos (muito semelhante ao Word), uma planilha eletrônica (muito semelhante ao Excel), um programa para criação de apresentações (muito semelhante ao PowerPoint).

Além destes, possui agenda, cliente de e-mail, programas gráficos (vetorial e bitmap), banco de dados, programa para geração de fórmulas matemáticas e navegador Web.

A figura abaixo ilustra a tela inicial do StarOffice 5.1:



Alguns fatos são relevantes:

- Possui versões para Linux, Solaris, OS/2 e Windows;
- É gratuito, tanto para uso pessoal quanto em empresas;
- Pode trabalhar com os formatos do MS Office, como .doc, .xls e .ppt;
- Recentemente a Sun liberou seu código fonte, o que assegurará seu desenvolvimento;
- Infelizmente, até a versão 5.2, sua interface ainda está escrita em Português de Portugal.

24.2 – Instalando o StarOffice 5.1

O StarOffice deve ser instalado em duas etapas. A primeira etapa consiste em instalar o pacote rpm que contém os arquivos do programa.

Isto pode ser feito simplesmente utilizando a linha de comando abaixo, estando como root, dentro do diretório onde se encontra o arquivo .rpm:

```
rpm -ivh StarOffice-5.1*
```

Isto feito, os programas já estarão na máquina, porém ainda não instalados. A próxima etapa consiste em instalar os executáveis do StarOffice.

Para esta tarefa, devemos estar no modo gráfico (como usuário comum) e executar o binário **setup** (normalmente no diretório */opt/Office51/bin*).

Ao ser executado, uma nova janela aparecerá, mostrando uma mensagem de boas vindas. Siga as instruções na tela e vá clicando no botão "Seguinte >>" até encontrar uma tela solicitando escolher entre o modo "Estação de Trabalho" e "Instalação Local".

Nesta tela, selecione "Estação de Trabalho" e prossiga na instalação. Os arquivos necessários serão copiados e o programa de instalação será finalizado.

O StarOffice já está instalado, bastando acessá-lo, executando o binário **soffice**, localizado no diretório home do usuário, dentro do subdiretório *Office51/bin*. Na instalação o StarOffice já é adicionado aos menus do KDE, porém se estiver utilizando outro gerenciador de janelas que não tenha acesso aos menus do KDE, deverá personalizar seu menu ou seu desktop para adicionar o StarOffice.

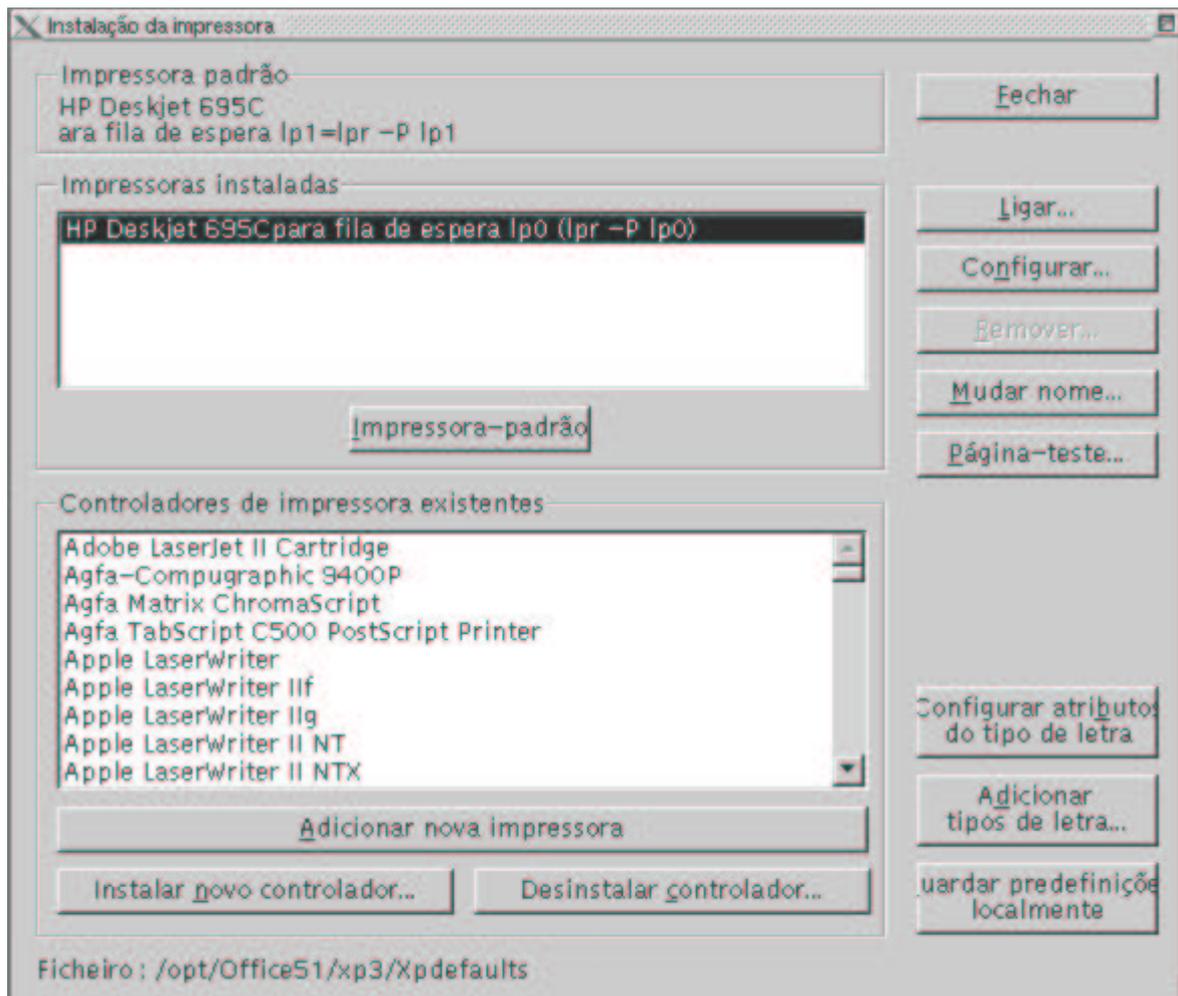
Na primeira utilização, será aberta uma tela solicitando a configuração da internet. Caso não deseje utilizar a internet, informe isto ao programa e o StarOffice estará pronto para ser utilizado. Caso deseje configurar a internet, continue o processo, informando as configurações do servidor proxy (se houver), as configurações da sua conta de e-mail (se houver) e as configurações de uma conta de notícias (news), se houver.

Isto feito, a área de trabalho do StarOffice estará na tela, pronta para a utilização pelo usuário.

24.3 – Configurando a Impressora Para o StarOffice 5.1

Para a utilização de uma impressora com o StarOffice, além de já ter uma impressora definida e configurada no sistema, devemos configurá-la dentro do StarOffice.

Isto pode ser realizado acessando o ícone "Configuração da Impressora" na tela inicial do StarOffice. Será aberta uma janela como a figura a seguir:



Deve-se seleccionar um dos modelos de impressoras listados ou caso a sua impressora não apareça na lista, seleccione "Generic Printer".

Clique no botão "Mudar Nome" e troque o nome por um mais sugestivo. Isto não é necessário, porém facilita a identificação da impressora, em caso de dúvidas.

Clique no botão "Configurar" e defina os padrões para serem utilizados com esta impressora.

Clique no botão "Ligar" e no campo "Filas de espera existentes" digite: **lp0=lpr -P lp0** e clique no botão "OK". Observe que este comando só é válido se a impressora estiver conectada em /dev/lp0 (correspondente a LPT1 do DOS / Windows). Altere os textos "lp0" do comando conforme a necessidade e a conexão da impressora instalada no sistema.

24.4 – Utilizando o StarOffice 5.1

Os detalhes da utilização do StarOffice não serão vistos aqui, supondo que o usuário já tenha conhecimento prévio do funcionamento básico de outras suítes Office disponíveis no mercado, como o MS Office, Corel WordPerfect, Lotus SmartSuite, etc.

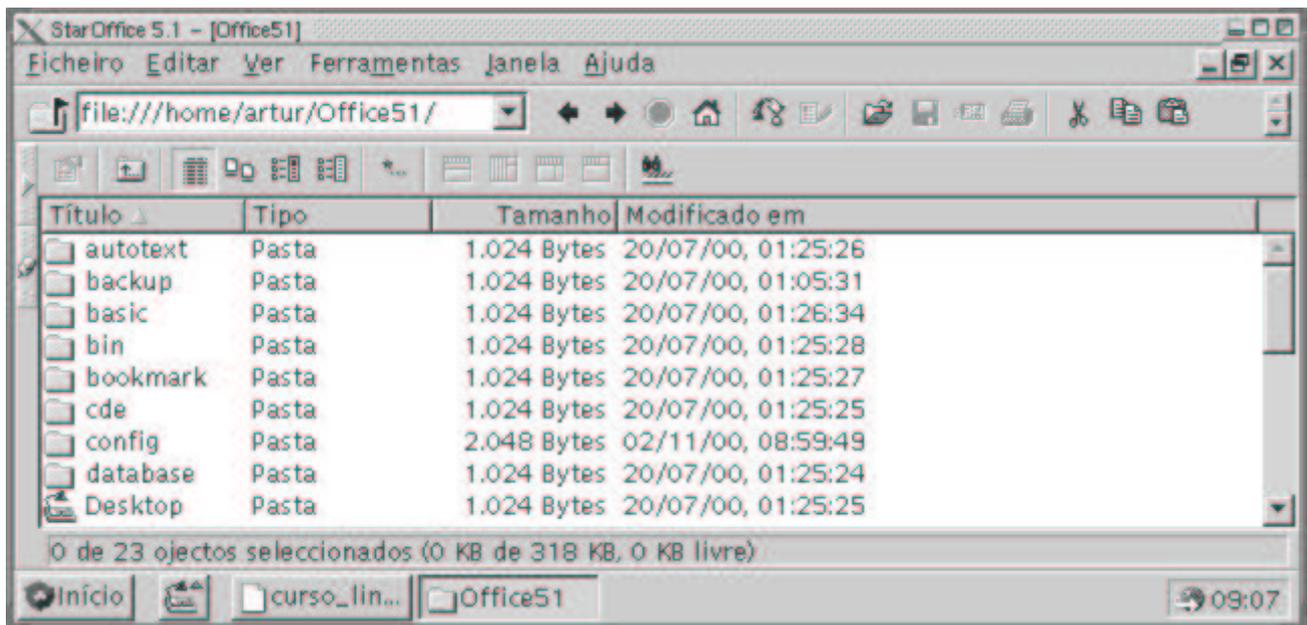
Isto porque o StarOffice é um programa similar, não devendo o usuário encontrar dificuldades em relacionar as funções do StarOffice às funções de outro pacote office, especialmente o MS Office. Basicamente, todas as principais funções encontradas no MS Word, MS Excel, MS PowerPoint, estão disponíveis no StarOffice de uma forma bastante parecida, senão igual.

Veremos então algumas particularidades do StarOffice em relação aos demais pacotes Office disponíveis.

Curso de Linux Básico

A área de trabalho criada pelo StarOffice possui uma grande semelhança com o desktop do Windows, com um botão início, pelo qual podemos ter acesso a várias funções interessantes, como localizar documentos ou executar comandos. Os documentos abertos aparecem na "barra de tarefas", ao lado do botão início. Esta barra de tarefas funciona da mesma forma que no Windows, permitindo alternar entre os documentos abertos. É possível integrar completamente o StarOffice ao gerenciador de janelas, utilizando a opção "Desktop Integrado" no menu "Ver".

Nesta área de trabalho também está disponível um navegador de arquivos, que também pode navegar na internet. A figura abaixo ilustra seu aspecto:



O StarOffice possui um ambiente de trabalho integrado, portanto de dentro da mesma interface podemos abrir todos os tipos de documentos disponíveis, como planilha, texto, desenho, apresentação, etc. Os programas que tratam cada tipo de documento possuem basicamente a mesma interface.

Para abrir qualquer documento, basta utilizar o menu "Ficheiro", opção "Abrir" e localizar o documento a ser aberto. Isto também pode ser feito usando o ícone correspondente na barra de ferramentas.

25 – TÓPICOS SOBRE SEGURANÇA

Neste capítulo teremos noções básicas sobre segurança em sistemas Linux.

Longe de querermos esgotar o assunto, serão abordados de forma breve os principais aspectos a serem observados e devidamente configurados, visando tornar o sistema seguro.

Também serão colocadas algumas dicas práticas para manter a segurança de um sistema Linux, sendo na maioria simples e de muita utilidade.

Muitas das informações apresentadas aqui só terão aplicação em casos específicos (caso a máquina seja um servidor de rede, um provedor Internet), enquanto as demais aplicar-se-ão a máquinas desktop isoladas ou participantes de uma rede (que pode ser a Internet).

A maior parte do material deste capítulo está baseado no TLM (The Linux Manual) , versão 3.4, de autoria de Hugo Cysneiros.

25.1 – Política de Senhas

Um sistema seguro pressupõe senhas seguras, já que praticamente todo acesso ao sistema é feito mediante senhas.

Portanto, é importante evitar o uso de senhas fáceis de serem descobertas, como partes do nome do usuário, datas relacionadas, placa do carro, número do telefone, etc...

Também é importante evitar como senhas palavras comuns que podem ser encontradas em dicionários, pois podem ser descobertas facilmente pelo método de tentativa e erro.

É altamente recomendado o uso de senhas com oito ou mais caracteres alfanuméricos (ou seja, letras e números), valendo-se ainda de símbolos, caracteres maiúsculos e minúsculos, e seguindo as recomendações acima.

A estrutura das senhas pode ser automatizada no sistema, de modo que as que não seguirem a estrutura predefinida não serão aceitas.

25.2 – Permissões de Acesso

Para um sistema seguro, as permissões de acesso aos diversos diretórios e arquivos devem ser cuidadosamente estudadas e configuradas, pois toda a segurança do sistema depende destas permissões.

É importante que cada usuário tenha acesso somente aos diretórios que realmente necessita, caso contrário pode ser uma entrada para o sistema, uma vez que sua senha seja descoberta.

Também é importante evitar que usuários comuns possam alterar as configurações do sistema (exceto suas próprias preferências, em seu ambiente de trabalho) ou executar comandos de administração, que possam afetar todo o sistema.

25.3 – Protegendo as Senhas

Não basta ter senhas difíceis de serem descobertas, se elas estiverem disponíveis para quem quiser ver. Para evitar isto, existem pacotes que permitem que as senhas fiquem escondidas (shadowed).

Em ambos os casos, as senhas são armazenadas criptografadas, porém, instalando o pacote Shadow Utils, as

Curso de Linux Básico

senhas não mais ficarão armazenadas em `/etc/passwd` e sim em `/etc/shadow`, sendo que normalmente este arquivo só possui permissão de leitura, exclusivamente para o root.

Também é importante proteger, além da senha de login no sistema, as demais senhas utilizadas pelos usuários, como as utilizadas para conexões à internet via modem (ppp), entre outras.

25.4 – Configuração dos Serviços TCP

Alguns serviços disponibilizados pelo protocolo TCP, se habilitados podem fornecer importantes informações a um possível invasor. São eles:

- **Netstat (tcp/15):** Informa todas conexões atuais (endereços, dns, portas, etc)
- **Systat (tcp/11):** Mostra qualquer / todos os processos que estão sendo rodados na máquina. Se acessado telnet na porta 11 da máquina, qualquer pessoa pode ver esses processos, e com isso, saber o que se está fazendo no exato momento em que se faz!
- **Finger (tcp/79):** Apresenta informações completas de usuários logados / não-logados no sistema (shells, diretórios, logins, etc).

Para deixar seu sistema mais seguro, edite o arquivo `/etc/inetd.conf` e comente(#) as seguintes linhas:

```
#finger stream tcp  nowait nobody /usr/sbin/tcpd  in.fingerd -w
#systat stream tcp  nowait nobody /usr/sbin/tcpd  /bin/ps -auwwx
#netstat  stream tcp  nowait root   /usr/sbin/tcpd /bin/netstat -a
```

Com isso, os 3 serviços estarão desabilitados, e ninguém poderá acessar além do root, claro.

25.5 – Acesso aos Serviços do Linux

O controle de acesso aos serviços do Linux que podem ou não ser acessados pelos usuários, é feito por meio de regras colocadas nos arquivos `/etc/hosts.allow` e `/etc/hosts.deny`.

Para o acesso a algum serviço do Linux , a seqüência é a seguinte :

- Verifica-se se o IP e o serviço estão no `hosts.allow`, se estiverem, acesso garantido (mesmo se o IP e o serviço estiverem no `hosts.deny`, este não será checado se houver referência em `hosts.allow`).
- Verifica-se se o IP e o serviço estão no `hosts.deny`, se não estiverem, acesso garantido.
- Se o IP não estiver no `hosts.allow` mas também não estiver no `hosts.deny`, o acesso é garantido.

hosts.deny: Contém os hosts que **NÃO** podem acessar certos serviços na máquina.

Sintaxe :
serviço:IP

Exemplo :
in.fingerd:200.00.00.00

```
|      |
|      | O IP da máquina que não poderá acessar tal serviço
|- Serviço , no caso o finger
```

Obs: Podemos usar alguns Wildcards como All ou Local

Ex:
in.fingerd:All Significa que nenhum IP poderá acessar o serviço em questão.
All:All except local Significa que nenhum IP exceto IPs locais possam acessar o serviço, no caso todos os serviços disponíveis.

hosts.allow: É o arquivo que contém os serviços que certos IPs podem acessar.

Sintaxe: Parecida com o do hosts.deny só que essa linha garante acesso
Serviço:IP

Exemplo :
in.fingerd:200.1.1.1
 Garante acesso do IP 200.1.1.1 ao serviço finger

All:200.20.20.20
 Garante acesso de 200.20.20.20 a todos os serviços disponíveis

Obs: É importante ressaltar que podemos colocar várias linhas em ambos os arquivos, podendo então anular ataques de certos IPs.

25.6 – Dicas de Segurança

Aqui serão listadas algumas dicas básicas que ajudam a manter um sistema seguro.

1 – Limitar o número de programas SUID root

Programas SUID root são programas que quando rodam, rodam no nível de root. Os programas SUID root podem fazer qualquer coisa que o root pode, tendo um alto grau de responsabilidade a nível de segurança.

Quando um programa deste tipo falha, usuários podem usar o root para fazer coisas que nem se pode imaginar, utilizando os exploits. Um exploit é um programa ou script que usa o SUID root para executar ações não permitidas a ele como usuário ou até mesmo ilegais, como usar shells de root, copiar arquivos de senhas, ler mails de outras pessoas, deletar arquivos, etc.

2 – Rodar programas com privilégio mínimo no acesso

Alguns programas não precisam de root para rodá-los, mas precisam de um alto acesso para o usuário normal. Aqui é onde começa a idéia do privilégio mínimo de acesso.

Por exemplo, a LP (linha de impressora) possui comandos que precisam de alto acesso para o usuário normal (para acessar a impressora), mas não precisa rodá-los como root. Então, uma pequena coisa a fazer é criar um usuário (tendo /bin/true como shell) e um grupo chamado **lp** e fazer com que qualquer usuário possa rodar qualquer dos comandos de LP e fazer tudo com os comandos LP que tiverem como proprietário e grupo o **lp**.

Isto fará com que o **lp** possa fazer seu trabalho (administrar as impressoras) e caso tenha sua segurança comprometida, o invasor não conseguirá dar um passo de root no sistema.

Para alguns programas que são SUID root, crie um usuário e um grupo para cada um deles, de forma que sejam rodados por usuários com privilégio mínimo.

Curso de Linux Básico

3 – Desabilitar serviços desnecessários

Se você não usa `rpc.mountd`, `rpc.nfsd` ou outros daemons parecidos, não rode-os. Simplesmente `kill -9 xxxx`, edite os scripts em `/etc/rc.d` e deixe-os comentados.

Além de aumentar a memória livre e o uso da CPU, é um meio de se prevenir contra invasores que tentam obter informações sobre o sistema e, claro, pegar root nele.

4 – Encriptar as conexões

Esta dica é útil no caso de conexões remotas. Se estas conexões não estiverem encriptadas, um usuário utilizando um sniffer (farejador) pode facilmente interceptar a conexão e com isto obter passwords (senhas) do sistema. Na lista de programas sobre segurança a seguir apresentamos um destes programas para encriptar conexões, porém existem outros.

5 – Instalar wrappers para `/bin/login` e outros programas

Wrappers são programas pequenos mas muito eficientes que filtram o que está sendo enviado para o programa. O wrapper para login remove todas as instâncias de diversas variáveis do ambiente e o wrapper do sendmail faz mais ou menos o mesmo.

6 – Manter os programas atualizados e o kernel na última versão estável

Esta dica se aplica aos sistemas que possuem vários usuários.

Procure estar sempre informado das novas versões dos programas de sua distribuição, atualizando-os assim que informado pelo distribuidor. Muitas das vezes estas atualizações são para corrigir falhas de segurança, que podem ser exploradas por um possível invasor. Geralmente as distribuições de Linux mantêm serviços de e-mail com o propósito de informar aos usuários sobre as atualizações do sistema, procure conhecê-lo e assiná-lo.

Também é importante manter o kernel do sistema sempre na última versão estável. Kernels antigos possuem seus bugs conhecidos por todos e às vezes são muito instáveis. Por exemplo, kernels 2.0.X tendem a ser mais rápidos que os 1.2.X e, é claro, mais estáveis.

7 – Manter no kernel apenas o código necessário

Ao recompilar o kernel do sistema, configure-o e compile-o somente com o código necessário.

Quatro razões vêm em mente:

- O kernel ficará mais rápido (menos códigos para rodar);
- O sistema terá mais memória livre;
- O sistema ficará mais estável;
- As partes desnecessárias não estarão disponíveis para ser usadas por um invasor.

8 – Permitir o mínimo possível de informações sobre o sistema

Um simples finger para o sistema da vítima pode revelar muitas coisas sobre seu sistema: quantos usuários, quando o admin está logado, ver o que ele está fazendo, quem ele é, quem usa o sistema e informações pessoais que podem ajudar um invasor a conseguir senhas de usuários.

Para evitar isto, deve ser utilizado um potente finger daemon, que limite quem pode se conectar ao sistema e exiba o mínimo possível sobre ele.

9 – Limitar o acesso ao sistema

Se puder, limite quem pode se conectar ao sistema. Se possível, bloqueie o acesso telnet de fora da subrede. Certamente o sistema ficará mais seguro e se evitará ser danificado por estranhos.

25.7 – Programas Para Segurança

Aqui serão listadas alguns programas úteis para um sistema seguro, porém existem muitos outros, ficando ao critério do administrador do sistema analisar as opções mais adequadas a cada necessidade.

As informações sobre versão dos programas não deve ser levada em consideração, pois podem estar desatualizadas no momento da leitura. Busque as versões mais atuais, geralmente encontradas na Internet.

Tcp Wrappers:

Antes que qualquer aplicativo TCP possa conectar, este pode ser processado pelo Tcp Wrappers para checar se o usuário que esta chamando está proibido de se conectar (via /etc/hosts.allow e /etc/hosts.deny). Ele é muito bom quando se deseja limitar quem pode conectar aos seus sites. Roda na maioria dos sistemas Linux.

Crack 5.0a:

Apenas porque possui as senhas escondidas (shadowed), não quer dizer que alguns usuários não possam pegá-las. Se os usuários forem forçados a escolher boas senhas, as chances de alguém pegar o arquivo de senhas shadow e conseguir algumas senhas para acesso no sistema diminuem bastante.

Este programa resolve o problema crackeando as senhas com dicionários e exibe informações sobre senhas encontradas no arquivo passwd.

Tripwire:

Com este programa instalado no sistema, configurado corretamente, e tendo um arquivo de configuração atualizado, você poderá saber se o sistema foi invadido e quais arquivos foram modificados ou instalados (principalmente cavalos de tróia!).

Tripwire pesquisa o sistema e verifica o tamanho, data e outras coisas sobre os arquivos, exibindo uma lista de todos os que tiveram alguma de suas características alteradas.

COPS 1.04:

Este pacote contém alguns programas de segurança e também alguns scripts que ajudam a proteger o sistema.

Secure Shell Home Page Secure shell daemon:

Utilitário para encriptação de conexões. Útil para evitar que hackers observem suas conexões. Provê autenticação, faz segurança em conexões de Xwindow e em alguns casos faz segurança em conexões telnet, entre outros usos.

Fefe's finger daemon:

Um simples e prático finger daemon. As opções mais interessantes são:

- Não revela muito sobre o usuário (último acesso, mail, shell);
- Previne certos tipos de ataques de usuários locais;
- Não necessita de root para rodar;
- Não permite finger no sistema;
- Faz log total.

Xinetd:

Um programa para substituir o famoso inetd. Inetd observa portas e inicializa programas quando algum usuário se conecta em alguma das portas. Xinetd oferece mais controle e configuração sobre as portas do que o inetd e é mais seguro.

Lsof:

Lista todos os arquivos abertos no sistema. Bom para prevenir algum tipo de ataque. Este programa informa quando algum usuário está rodando um sniffer e guarda todos os dados sobre o invasor nos arquivos de log.

Pidentd 2.5.1:

Outro inetd daemon que é mais configurável e seguro que o simples inetd do Linux.

rhosts.dodgy:

Este script verifica o .rhosts por '+' e outras coisas que nunca poderiam estar nele. É escrito em perl e facilmente configurado. Uma dica é colocá-lo no contrab, para que ele faça uma verificação periódica (por exemplo, a cada 15 minutos).

ICMPinfo 1.11:

Este programa detecta qualquer tipo de pacote ICMP enviado à sua máquina. Idem ao TCP Dump, exceto que trabalha com pacotes ICMP (Ping).

TCP Dump:

Mostra uma grande quantidade de informações sobre todas as conexões via TCP. Isto serve para administradores de sistema. Se rodado durante um ataque, o atacante será exibido e logado.

25.8 – Considerações Finais Sobre Segurança

É importante lembrarmos que um bom sistema de backup faz parte das medidas de segurança de um sistema, pois é a partir dele que um sistema invadido ou danificado poderá voltar a funcionar corretamente, portanto, deve estar sempre atualizado e protegido.

Também devemos nos preocupar com o acesso físico ao sistema, dependendo da situação. Muitas vezes este item é desprezado, podendo vir a ser a porta de entrada (literalmente) para um invasor.

Um cuidado especial deve ser tomado na utilização de certos tipos de programas, hoje bastante difundidos, como ICQ, IRC, CHAT, Napster, Gnutella, além de programas que tratam Real Audio, Real Video, etc. Estes programas quando executados mantêm uma determinada porta do TCP (ou UDP) sempre aberta, por onde um possível invasor pode entrar no sistema. As portas utilizadas podem ser verificadas através do comando netstat, quando o programa estiver sendo executado.

26 – LICENÇA DE DOCUMENTAÇÃO LIVRE GNU

GNU Free Documentation License

Tradução de **João S. O. Bueno** – Email: gwidion@mpc.com.br

Esta é uma tradução não oficial da Licença de Documentação Livre GNU em Português Brasileiro. Ela não é publicada pela Free Software Foundation, e não se aplica legalmente a distribuição de textos que usem a GFDL – apenas o texto original em Inglês da GNU FDL faz isso. Entretanto, nós esperamos que esta tradução ajudará falantes de português a entenderem melhor a GFDL.

This is an unofficial translation of the GNU General Documentation License into Brazilian Portuguese. It was not published by the Free Software Foundation, and does not legally state the distribution terms for software that uses the GFDL—only the original English text of the GFDL does that. However, we hope that this translation will help Portuguese speakers understand the GFDL better.

Licença de Documentação Livre GNU Versão 1.1, Março de 2000

Copyright (C) 2000 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111–1307 USA

É permitido a qualquer um copiar e distribuir cópias exatas deste documento de licença, mas não é permitido alterá-lo.

0. INTRODUÇÃO

O propósito desta Licença é deixar um manual, livro–texto ou outro documento escrito “livre” no sentido de liberdade: assegurar a qualquer um a efetiva liberdade de copiar ou redistribuí-lo, com ou sem modificações, comercialmente ou não. Secundariamente, esta Licença mantém para o autor e editor uma forma de ter crédito por seu trabalho, sem ser considerado responsável pelas modificações feitas por terceiros.

Esta Licença é um tipo de “copyleft” (“direitos revertidos”), o que significa que derivações do documento precisam ser livres no mesmo sentido. Ela complementa a GNU Licença Pública Geral (GNU GPL), que é um copyleft para software livre.

Nós fizemos esta Licença para que seja usada em manuais de software livre, por que software livre precisa de documentação livre: um programa livre deve ser acompanhado de manuais que provenham as mesmas liberdades que o software possui. Mas esta Licença não restringe a manuais de software; ela pode ser usada para qualquer trabalho em texto, independentemente do assunto ou se ele é publicado como um livro impresso. Nós recomendamos esta Licença principalmente para trabalhos cujo propósito seja de instrução ou referência.

1. APLICABILIDADE E DEFINIÇÕES

Esta Licença se aplica a qualquer manual ou outro texto que contenha uma nota colocada pelo detentor dos direitos autorais dizendo que ele pode ser distribuído sob os termos desta Licença. O “Documento”, abaixo, se refere a qualquer tal manual ou texto. Qualquer pessoa do público é um licenciado e é referida como “você”.

Uma “Versão Modificada” do Documento se refere a qualquer trabalho contendo o documento ou uma parte dele, quer copiada exatamente, quer com modificações e/ou traduzida em outra língua.

Uma “Seção Secundária” é um apêndice ou uma seção inicial do Documento que trata exclusivamente da relação dos editores ou dos autores do Documento com o assunto geral do Documento (ou assuntos relacionados) e não contém nada que poderia ser incluído diretamente nesse assunto geral. (Por exemplo, se o Documento é em parte um livro texto de matemática, a Seção Secundária pode não explicar nada de matemática).

Essa relação poderia ser uma questão de ligação histórica com o assunto, ou matérias relacionadas, ou de posições legais, comerciais, filosóficas, éticas ou políticas relacionadas ao mesmo.

As “Seções Invariantes” são certas Seções Secundárias cujos títulos são designados, como sendo de Seções Invariantes, na nota que diz que o Documento é publicado sob esta Licença.

Os "Textos de Capa" são certos trechos curtos de texto que são listados, como Textos de Capa Frontal ou Textos da Quarta Capa, na nota que diz que o texto é publicado sob esta Licença.

Uma cópia "Transparente" do Documento significa uma cópia que pode ser lida automaticamente, representada num formato cuja especificação esteja disponível ao público geral, cujos conteúdos possam ser vistos e editados diretamente e sem mecanismos especiais com editores de texto genéricos ou (para imagens compostas de pixels) programas de pintura genéricos ou (para desenhos) por algum editor de desenhos de grandemente difundido, e que seja passível de servir como entrada a formatadores de texto ou para tradução automática para uma variedade de formatos que sirvam de entrada para formatadores de texto. Uma cópia feita em um formato de arquivo outrossim Transparente cuja constituição tenha sido projetada para atrapalhar ou desencorajar modificações subsequentes pelos leitores não é Transparente. Uma cópia que não é "Transparente" é chamada de "Opaca".

Exemplos de formatos que podem ser usados para cópias Transparentes incluem ASCII simples sem marcações, formato de entrada do Texinfo, formato de entrada do LaTeX, SGML ou XML usando uma DTD disponibilizada publicamente, e HTML simples, compatível com os padrões, e projetado para ser modificado por pessoas. Formatos opacos incluem PostScript, PDF, formatos proprietários que podem ser lidos e editados apenas com processadores de texto proprietários, SGML ou XML para os quais a DTD e/ou ferramentas de processamento e edição não estejam disponíveis para o público, e HTML gerado automaticamente por alguns editores de texto com finalidade apenas de saída.

A "Página do Título" significa, para um livro impresso, a página do título propriamente dita, mais quaisquer páginas subsequentes quantas forem necessárias para conter, de forma legível, o material que esta Licença requer que apareça na página do título. Para trabalhos que não tenham uma tal página do título, "Página do Título" significa o texto próximo da aparição mais proeminente do título do trabalho, precedendo o início do corpo do texto.

2. FAZENDO CÓPIAS EXATAS

Você pode copiar e distribuir o Documento em qualquer meio, de forma comercial ou não comercial, desde que esta Licença, as notas de copyright, e a nota de licença dizendo que esta Licença se aplica ao documento estejam reproduzidas em todas as cópias, e que você não acrescente nenhuma outra condição quaisquer que sejam às desta Licença.

Você não pode usar medidas técnicas para obstruir ou controlar a leitura ou confecção de cópias subsequentes das cópias que você fizer ou distribuir. Entretanto, você pode aceitar compensação em troca de cópias. Se você distribuir uma quantidade grande o suficiente de cópias, você também precisa respeitar as condições da seção 3.

Você também pode emprestar cópias, sob as mesmas condições colocadas acima, e você também pode exibir cópias publicamente.

3. FAZENDO CÓPIAS EM QUANTIDADE

Se você publicar cópias do Documento em número maior que 100, e a nota de licença do Documento obrigar Textos de Capa, você precisa incluir as cópias em capas que tragam, clara e legivelmente, todos esses Textos de Capa: Textos de Capa da Frente na capa da frente, e Textos da Quarta Capa na capa de trás. Ambas as capas também precisam identificar clara e legivelmente você como o editor dessas cópias. A capa da frente precisa apresentar o título completo com todas as palavras do título igualmente proeminentes e visíveis. Você pode adicionar outros materiais às capas. Fazer cópias com modificações limitadas às capas, tanto quanto estas preservem o título do documento e satisfçam a essas condições, pode tratado como cópia exata em outros aspectos.

Se os textos requeridos em qualquer das capas for muito volumoso para caber de forma legível, você deve colocar os primeiros (tantos quantos couberem de forma razoável) na capa verdadeira, e continuar os outros nas páginas adjacentes.

Se você publicar ou distribuir cópias Opacas do Documento em número maior que 100, voce precisa ou incluir uma cópia Transparente que possa ser lida automaticamente com cada cópia Opaca, ou informar em ou com cada cópia Opaca a localização de uma cópia Transparente completa do Documento acessível publicamente em uma rede de computadores, à qual o público usuário de redes tenha acesso a download gratuito e anônimo

utilizando padrões públicos de protocolos de rede. Se você utilizar o segundo método, você precisa tomar cuidados razoavelmente prudentes, quando iniciar a distribuição de cópias Opacas em quantidade, para assegurar que esta cópia Transparente vai permanecer acessível desta forma na localização especificada por pelo menos um ano depois da última vez em que você distribuir uma cópia Opaca (diretamente ou através de seus agentes ou distribuidores) daquela edição para o público.

É pedido, mas não é obrigatório, que você contate os autores do Documento bem antes de redistribuir qualquer grande número de cópias, para lhes dar uma oportunidade de prover você com uma versão atualizada do Documento.

4. MODIFICAÇÕES

Você pode copiar e distribuir uma Versão Modificada do Documento sob as condições das seções 2 e 3 acima, desde que você publique a Versão Modificada estritamente sob esta Licença, com a Versão Modificada tomando o papel do Documento, de forma a licenciar a distribuição e modificação da Versão Modificada para quem quer que possua uma cópia da mesma. Além disso, você precisa fazer o seguinte na versão modificada:

A. Usar na Página de Título (e nas capas, se alguma) um título distinto daquele do Documento, e daqueles de versões anteriores (que deveriam, se houvesse algum, estarem listados na seção Histórico do Documento). Você pode usar o mesmo título de uma versão anterior se o editor original daquela versão lhe der permissão.

B. Listar na Página de Título, como autores, uma ou mais das pessoas ou entidades responsáveis pela autoria das modificações na Versão Modificada, conjuntamente com pelo menos cinco dos autores principais do Documento (todos os seus autores principais, se ele tiver menos que cinco).

C. Colocar na Página de Título o nome do editor da Versão Modificada, como o editor.

D. Preservar todas as notas de copyright do Documento.

E. Adicionar uma nota de copyright apropriada para suas próprias modificações adjacente às outras notas de copyright.

F. Incluir, imediatamente depois das notas de copyright, uma nota de licença dando ao público o direito de usar a Versão Modificada sob os termos desta Licença, na forma mostrada no Adendo abaixo.

G. Preservar nessa nota de licença as lisats completas das Seções Invariantes e os Textos de Capa requeridos dados na nota de licença do Documento.

H. Incluir uma cópia inalterada desta Licença.

I. Preservar a seção intitulada "Histórico", e seu título, e adicionar à mesma um item dizendo pelo menos o título, ano, novos autores e editor da Versão Modificada como dados na Página de Título. Se não houver uma sessão denominada "Histórico" no Documento, criar uma dizendo o título, ano, autores, e editor do Documento como dados em sua Página de Título, então adicionar um item descrevendo a Versão Modificada, tal como descrito na sentença anterior.

J. Preservar o endereço de rede, se algum, dado no Documento para acesso público a uma cópia Transparente do Documento, e da mesma forma, as localizações de rede dadas no Documento para as versões anteriores em que ele foi baseado. Elas podem ser colocadas na seção "Histórico". Você pode omitir uma localização na rede para um trabalho que tenha sido publicado pelo menos quatro anos antes do Documento, ou se o editor original da versão a que ela se refira der sua permissão.

K. Em qualquer seção intitulada "Agradecimentos" ou "Dedicatórias", preservar o título da seção e preservar a seção em toda substância e tim de cada um dos agradecimentos de contribuidores e/ou dedicatórias dados.

L. Preservar todas as Seções Invariantes do Documento, inalteradas em seus textos ou em seus títulos. Números de seção ou equivalentes não são considerados parte dos títulos da seção.

M. Apagar qualquer seção intitulada "Endossos". Tal sessão não pode ser incluída na Versão Modificada.

Curso de Linux Básico

N. Não re–entitular qualquer seção existente com o título “Endossos” ou com qualquer outro título dado a uma Seção Invariante.

Se a Versão Modificada incluir novas seções iniciais ou apêndices que se qualifiquem como Seções Secundárias e não contenham nenhum material copiado do Documento, você pode optar por designar alguma ou todas aquelas seções como invariantes. Para fazer isso, adicione seus títulos à lista de Seções Invariantes na nota de licença da Versão Modificada. Esses títulos precisam ser diferentes de qualquer outro título de seção.

Você pode adicionar uma seção intitulada “Endossos”, desde que ela não contenha qualquer coisa além de endossos da sua Versão Modificada por várias pessoas ou entidades – por exemplo, declarações de revisores ou de que o texto foi aprovado por uma organização como a definição oficial de um padrão.

Você pode adicionar uma passagem de até cinco palavras como um Texto de Capa da Frente, e uma passagem de até 25 palavras como um Texto de Quarta Capa, ao final da lista de Textos de Capa na Versão Modificada. Somente uma passagem de Texto da Capa da Frente e uma de Texto da Quarta Capa podem ser adicionados por (ou por acordos feitos por) qualquer entidade. Se o Documento já incluir um texto de capa para a mesma capa, adicionado previamente por você ou por acordo feito com alguma entidade para a qual você esteja agindo, você não pode adicionar um outro; mas você pode trocar o antigo, com permissão explícita do editor anterior que adicionou a passagem antiga.

O(s) autor(es) e editor(es) do Documento não dão permissão por esta Licença para que seus nomes sejam usados para publicidade ou para assegurar ou implicar endossamento de qualquer Versão Modificada.

5. COMBINANDO DOCUMENTOS

Você pode combinar o Documento com outros documentos publicados sob esta Licença, sob os termos definidos na seção 4 acima para versões modificadas, desde que você inclua na combinação todas as Seções Invariantes de todos os documentos originais, sem modificações, e liste todas elas como Seções Invariantes de seu trabalho combinado em sua nota de licença.

O trabalho combinado precisa conter apenas uma cópia desta Licença, e Seções Invariantes Idênticas com múltiplas ocorrências podem ser substituídas por apenas uma cópia. Se houver múltiplas Seções Invariantes com o mesmo nome mas com conteúdos distintos, faça o título de cada seção único adicionando ao final do mesmo, em parênteses, o nome do autor ou editor original daquela seção, se for conhecido, ou um número que seja único. Faça o mesmo ajuste nos títulos de seção na lista de Seções Invariantes nota de licença do trabalho combinado.

Na combinação, você precisa combinar quaisquer seções intituladas “Histórico” dos diversos documentos originais, formando uma seção intitulada “Histórico”; da mesma forma combine quaisquer seções intituladas “Agradecimentos”, ou “Dedicações”. Você precisa apagar todas as seções intituladas como “Endosso”.

6. COLETÂNEAS DE DOCUMENTOS

Você pode fazer uma coletânea consistindo do Documento e outros documentos publicados sob esta Licença, e substituir as cópias individuais desta Licença nos vários documentos com uma única cópia incluída na coletânea, desde que você siga as regras desta Licença para cópia exata de cada um dos Documentos em todos os outros aspectos.

Você pode extrair um único documento de tal coletânea, e distribuí-lo individualmente sob esta Licença, desde que você insira uma cópia desta Licença no documento extraído, e siga esta Licença em todos os outros aspectos relacionados à cópia exata daquele documento.

7. AGREGAÇÃO COM TRABALHOS INDEPENDENTES

Uma compilação do Documento ou derivados dele com outros trabalhos ou documentos separados e independentes, em um volume ou mídia de distribuição, não conta como uma Versão Modificada do Documento, desde que não seja reclamado nenhum copyright de compilação seja reclamado pela compilação. Tal compilação é chamada um “agregado”, e esta Licença não se aplica aos outros trabalhos auto–contidos compilados junto com o Documento, só por conta de terem sido assim compilados, e eles não são trabalhos derivados do Documento.

Se o requerido para o Texto de Capa na seção 3 for aplicável a essas cópias do Documento, então, se o

Documento constituir menos de um quarto de todo o agregado, os Textos de Capa do Documento podem ser colocados em capas adjacentes ao Documento dentro do agregado. Senão eles precisam aparecer nas capas de todo o agregado.

8. TRADUÇÃO

Tradução é considerada como um tipo de modificação, então você pode distribuir traduções do Documento sob os termos da seção 4. A substituição de Seções Invariantes por traduções requer uma permissão especial dos detentores do copyright das mesmas, mas você pode incluir traduções de algumas ou de todas as Seções Invariantes em adição às versões originais dessas Seções Invariantes. Você pode incluir uma tradução desta Licença desde que você também inclua a versão original em Inglês desta Licença. No caso de discordância entre a tradução e a versão original em Inglês desta Licença, a versão original em Inglês prevalecerá.

9. TÉRMINO

Você não pode copiar, modificar, sublicenciar, ou distribuir o Documento exceto como expressamente especificado sob esta Licença. Qualquer outra tentativa de copiar, modificar, sublicenciar, ou distribuir o Documento é nula, e resultará automaticamente no término de seus direitos sob esta Licença. Entretanto, terceiros que tenham recebido cópias, ou direitos, de você sob esta Licença não terão suas licenças terminadas tanto quanto esses terceiros permaneçam em total acordo com esta Licença.

10. REVISÕES FUTURAS DESTA LICENÇA

A Free Software Foundation pode publicar novas versões revisadas da Licença de Documentação Livre GNU de tempos em tempos. Tais novas versões serão similares em espírito à versão presente, mas podem diferir em detalhes ao abordarem novos problemas e preocupações. Veja <http://www.gnu.org/copyleft/>.

A cada versão da Licença é dado um número de versão distinto. Se o Documento especificar que uma versão particular desta Licença "ou qualquer versão posterior" se aplica ao mesmo, você tem a opção de seguir os termos e condições daquela versão específica, ou de qualquer versão posterior que tenha sido publicada (não como rascunho) pela Free Software Foundation. Se o Documento não especificar um número de Versão desta Licença, você pode escolher qualquer versão já publicada (não como rascunho) pela Free Software Foundation.

ADENDO: Como usar esta Licença para seus documentos

Para usar esta Licença num documento que você escreveu, inclua uma cópia desta Licença no documento e ponha as seguintes notas de copyright e licenças logo após a página de título:

Copyright (c) ANO SEU NOME.

É dada permissão para copiar, distribuir e/ou modificar este documento sob os termos da Licença de Documentação Livre GNU, Versão 1.1 ou qualquer versão posterior publicada pela Free Software Foundation; com as Seções Invariantes sendo LISTE SEUS TÍTULOS, com os Textos da Capa da Frente sendo LISTE, e com os Textos da Quarta-Capa sendo LISTE. Uma cópia da licença em está inclusa na seção intitulada "Licença de Documentação Livre GNU".

Se você não tiver nenhuma Seção Invariante, escreva "sem Seções Invariantes" ao invés de dizer quais são invariantes. Se você não tiver Textos de Capa da Frente, escreva "sem Textos de Capa da Frente" ao invés de "com os Textos da Capa da Frente sendo LISTE"; o mesmo para os Textos da Quarta Capa.

Se o seu documento contiver exemplos não triviais de código de programas, nos recomendamos a publicação desses exemplos em paralelo sob a sua escolha de licença de software livre, tal como a GNU General Public License, para permitir o seu uso em software livre.

27 – BIBLIOGRAFIA

- Guia de Instalação Conectiva Linux, Versão 1.0, 23 de Junho de 1999.
Conectiva Informática Ltda.
- Guia do Usuário Conectiva Linux, Versão 1.0, 3 de agosto de 1999.
Conectiva Informática Ltda.
- Linux Total (Guia de Referência Linux), Versão 1.0, 30 de Dezembro de 1999.
Conectiva Informática Ltda.
- Guia do Administrador de Sistemas Linux, Versão 0.6.2, 9 de Setembro de 1999.
Lars Wirzenius, tradução Conectiva Informática Ltda.
- Guia do Administrador de Redes Linux, Versão beta 0.21, Abril de 1999.
Olaf Kirch, tradução Conectiva Informática Ltda.
- TLM (The Linux Manual), Versão 3.4
Hugo Cisneiros
- Linux The Book
- Linux Soluções Simples (Coleção Informática Descomplicada, No. 13)
André Luiz Dias e Wagner Pratti
Editora Escala
- Linux Dicas e Macetes (Coleção Informática Descomplicada, No. 17)
André Luiz Dias e Wagner Pratti – Equipe Frente!
Editora Escala
- Programando Para Linux (Coleção Informática Descomplicada, No. 18)
André Luiz Dias e Wagner Pratti – Equipe Frente!
Editora Escala
- Coleção Informática Passo-a-Passo, No 2 (OpenLinux)
Editora Escala
- Revista do Linux
Conectiva Informática Ltda.
- Revista PCMaster
Editora Europa
- Revista Geek
Digerati Editorial
- Revista Arquivo Linux
Digerati Editorial
- Revista PC Expert
CD Expert Editora e Distribuidora Ltda.
- Revista Linux Prático
Editora Europa
- Outros: Lista de Discussão linux-br, HOWTOs diversos, Páginas man, Páginas info.