

Manual para Iniciantes em UNIX

versão 1.0

O objetivo deste manual é servir de referência inicial a usuários iniciais de Unix, e mais particularmente do Sistema Operacional Linux, desenvolvido para Pcs. Estão contidos aqui, os comandos mais utilizados numa primeira interação com esse sistema. O manual foi desenvolvido baseado em livros e na prática em cima de máquinas Linux da área de Internet e Laboratório de Informática da Unicruz.

O que é o UNIX?

R. É um sistema operacional; uma coleção de programas projetados para controlar as interações das funções de baixo nível da máquina com os programas de aplicação.

Como é tratado um arquivo no UNIX?

R. Um arquivo pode estar disponível somente para o próprio usuário ou ser compartilhado com outros usuário. Quem decide é o usuário que cria-o. Ao digitar-se o comando “ls -l”, todos os arquivos do diretório corrente serão listados. Nota-se que na primeira coluna da esquerda para direita, teremos 10 caracteres. Estes caracteres são, em ordem:

d, l, c, b - Se estiver setado, quer dizer que esse é um diretório / link / dispositivo caracterer / dispositivo bloco.

r - Se estiver setado, quer dizer que o usuário pode ler o conteúdo do arquivo.

w - Se estiver setado, quer dizer que o usuário pode escrever no arquivo.

x - Se estiver setado, quer dizer que o usuário pode executar o arquivo.

r - Se estiver setado, quer dizer que os integrantes do grupo o qual encontra-se o usuário podem ler o conteúdo do arquivo.

w - Se estiver setado, quer dizer que os integrantes do grupo o qual encontra-se o usuário podem escrever no arquivo.

x - Se estiver setado, quer dizer que os integrantes do grupo o qual encontra-se o usuário podem executar o arquivo.

r - Se estiver setado, quer dizer que os outros usuários podem ler o conteúdo do arquivo.

w - Se estiver setado, quer dizer que os outros usuários podem escrever no arquivo.

x - Se estiver setado, quer dizer que os outros usuários podem executar o arquivo.

Ex: drwxrwxrwx

Não estar setado, significa que em vez de ter uma letra na coluna correspondente, teria um “-”.

Ex: -rwxr--r--

Este exemplo quer dizer que é um arquivo, onde o usuário tem permissão de leitura, escrita e execução do mesmo; onde o grupo só pode ler o arquivo, não podendo escrever e executar; e onde os outros usuário também só podem ler o arquivo, ficando sem a opção de escrever e executar o mesmo.

O usuário pode mudar essas restrições, usando o comando *chmod*, que será explicado posteriormente.

Comandos básicos do UNIX:

adduser - Cadastro de usuários. Comando válido somente para o superusuário.

banner - É usado para criar palavras com letras gigantes.

Ex: *banner Ola Mundo*

Normalmente a saída do banner é o vídeo, mas pode-se redirecionar para um arquivo:

Ex: *banner Ola Mundo ola.mundo*

cat - Esse comando é similar ao TYPE do DOS. Visualiza o arquivo, sem abri-lo em um editor de texto.

Pode-se usar o comando “[more]”, que visualiza o arquivo em páginas.

cd - Passar entre diretórios.

Ex: *cd ..*

Esse comando volta para o diretório mãe do diretório onde estava.

Ex: *cd*

Esse comando volta para o diretório home.

Ex: *cd/users*

Esse comando passa do diretório atual para o diretório users se ele existir.

chfn - Esse comando muda informações do *finger*.

Ex: *chfn*

Muda informações como nome, trabalho, telefone do trabalho e telefone de casa.

chgrp - Esse comando é usado para mudar o atributo de pertinência de grupo de um arquivo.

Ex: *chgrp novo_grupo nome_do_arquivo*

chmod - Esse comando é usado para mudar o modo de acesso de um arquivo

Ex: *chmod quem+-que nome_do_arquivo*

Ex: *chmod go-rw arquivo*

Essa linha de comando faz com que o arquivo “arquivo” retire(-) as permissões de leitura(r) e escrita(w) do grupo(g) e dos outros(o). Quer dizer que os usuários do mesmo grupo e os outros não terão mais acesso sobre o arquivo.

Opções:

“quem” => u - usuário; g - grupo; o - outros.

“+-” => + dá permissão; - retira permissão.

“que” => r - ler(read); w - escrever(write); x - executar.

chown - Esse comando é usado para passar a propriedade de um arquivo para outra pessoa. Para mudar a propriedade de um arquivo, o usuário tem de ser dono do arquivo. Se o usuário mudar acidentalmente a propriedade, tem de pedir ao novo usuário que mude a propriedade de volta.

Ex: *chown novo_dono nome_do_arquivo*

clear - Limpa tela.

cp - O comando **cp** é usado para duplicar arquivos. Copia um ou vários arquivos.

Ex: *cp arquivo1 arquivo2*

Arquivo1 é o arquivo de entrada da operação de cópia, e o arquivo2 é a

saída produzida. Arquivo1 e arquivo2 devem ter nomes distintos; se tiverem o mesmo nome, então será emitida uma mensagem de diagnóstico indicando que são idênticos, e o arquivo não será copiado sobre si mesmo. Se o arquivo2 já existia, seu conteúdo será substituído pelo conteúdo do arquivo1.

Sintaxe: cp [-ipr] <arquivo> [arquivo ...] <destino>

Parâmetros:

-i - Pede confirmação para cada arquivo a ser copiado.

-p - Mantém na cópia as datas de modificação e permissões do arquivo original.

-r - Copia recursivamente arquivos e diretórios. Neste caso destino deve se referir a um diretório.

Exemplo: cp -r ~/leonardo/html/ /www

date - Esse comando permite ver a data.

Ex: *date*

O comando **date** também permite trocar a hora do sistema.

Ex: *date 1225123097*

Este comando quer dizer que o relógio do sistema passa a marcar 12:30 horas do dia 25 do mês de dezembro (12) de 1997 (97).

Exercício:

Verifique a data do sistema, e troque, se for necessário.

env - Comando usado para obter uma lista do conteúdo do ambiente shell do usuário. O conteúdo é formado por cadeias que compõem as variáveis do shell e seus valores.

Ex: *env*

find - Este comando é uma maneira fácil e poderosa de localizar objetos no sistema de arquivos do UNIX. O comando **find** tem muitas opções.

Ex: *find / -name Systems -print*

Esse exemplo instrui o comando **find** para começar a busca na raiz do sistema de arquivos (/), localizar todas as ocorrências de arquivos chamados

Systems (- name *Systems*), e exibir os resultados na saída padrão (-print).

Ex: *find /usr/tsm -name Sis* -print*

Este outro exemplo, procura a partir do diretório */usr/tsm* todos os arquivos que tenham começo com *Sys*.

ftp - Protocolo de Transmissão de arquivos.

Ao entrar em algum endereço de *ftp*, e o mesmo pedir um cadastro e uma senha, o usuário deve-se cadastrar como **anonymous**, e entrar como senha qualquer palavra, seguido de **@**, pois ele estará pedindo um e-mail como senha.

Para copiar arquivos que o usuário tenha no sistema LINUX para o sistema W95, o usuário deve entrar no aplicativo **ftp** (Iniciar - Programas - Internet - FTP), acessar a máquina *dinf*, com o comando **open dinf**, e digitando o seu login e sua correspondente senha. O usuário automaticamente irá para seu diretório *home*, onde o usuário tem permissão de gravar arquivos. Então, é só copia-los, como se estivessem em uma máquina qualquer (*bin - hash - get* ou *mget*).

Principais comandos:

open <nome_máquina> => permite acesso na máquina especificada.

bin => Seta modo de transferência binário.

cd <diretorio> => Muda de diretório.

get remote.file <local.file> => Download o arquivo.

hash => Mostra o sinal: #, a cada bloco transmitido.

lcd <drive: diretório> => Seta o diretório local.

ls -lF |more => Lista os arquivos com paradas na tela.

mget <arquivos> => Download de vários arquivos

put local.file <remote.file> => Upload o arquivo.

bye => Sair.

Exercício:

Buscar no endereço da Universidade de Santa Maria (<ftp.ufsm.br>) os arquivos *listao.txt.gz* e *listao.arj*, que localizão-se no diretório */pub/vest97*.

grep - É a base da família de comandos **grep(grep, egrep, fgrep)**. É um filtro que examina os arquivos de entrada em busca de padrões. Quando um batimento é encontrado, a linha que o contém é gravada na saída padrão, a menos que seja impedido por uma das opções. Múltiplos arquivos podem ser

processados concorrentemente especificando-se seus nomes na linha de comando, embora usualmente seja usado em um arquivo por vez.

Ex: *grep opções expressão arquivo(s)*

As opções são:

* -c : produz contagem das linhas que contém o padrão;

* -i : instrui o grep a não lvar e conta a diferença entre letras maiúsculas e minúsculas.

* -l : especifica que somente devem ser exibidos os nomes dos arquivos que contém o padrão. Útil quando se processa grande quantidade de arquivos.

* -n : ativa a numeração de linhas, na exibição das linhas que batem.

* -s : causa a supressão das mensagens de erro quando são encontrados arquivos que não podem ser lidos ou quando não são encontrados os arquivos especificados.

* -v : instrui o grep a imprimir na saída padrão todas as linhas, exceto aquelas que não contém o padrão.

Se o usuário quiser ver sua entrada em um arquivo como o /etc/passwd, não é necessário listar o arquivo inteiro, ou editá-lo. O comando a seguir poderia ser usado para produzir a informação desejada:

Ex: *grep 'tsm' /etc/passwd*

Tabela de Tipos de Arquivos:

Compactador	Descompactador	Extensão
Zip	Unzip	.zip
Pack	Unpack	.z
Compress	Uncompress	.Z
Gzip	Gunzip ou Gzip -d	.gz

Obs.:

zip e unzip são os equivalentes ao pkzip e pkunzip para DOS / Windows.

gunzip - Esse comando é usado para descompactar arquivos com extensões “.gz”, “.tgz”, “.taz”, “.tar.gz” e “tar.Z”.

Ex: *gunzip nome_do_arquivo*

Esse comando pode ser substituído por **gzip -d** ou **zcat**.

Exercício:

Descompactar o arquivo *listao.txt.gz*, que foi buscado por **ftp** na UFSM. Logo após, copie o arquivo para o diretório **c:** e visualize-o no MS-WordPad.

gzip - Esse comando é usado para compactar arquivos. É criada então a extensão “.gz”.

Ex: *gzip nome_do_arquivo*

Cria-se então, o arquivo “nome_do_arquivo.gz”.

Para descompactá-lo, usa-se o comando **gzip -d**, **gunzip** ou **zcat**.

Exercício:

Compactar o arquivo *listao.txt*, que foi buscado na UFSM e descompactado por *gunzip*.

Outros:

Awk	Formatar saídas para aparecer determinadas colunas
Cb	Embelezador de programas para C.
Split:	Ex.: <i>split -b 1440k arquivo.exe arquivo</i> (Linux) Ex.: <i>copy /b arquivo.* arquivo.exe</i>
Cal	Mostra um calendário
Sed	Substitui cadeias em arquivos E.: <i>sed s/4/9/g arqfonte arqdestino</i>

OBS.:

Arquivos .ps PostScript é da Adobe Systems usado para traçar gráficos tridimensionais ou não em impressoras e vídeos.

kill - Esse é o comando usado para encerrar processos ativos no sistema UNIX. Não há um programa equivalente no DOS, porque o DOS é um sistema monotarefa. Para eliminar um processo, o usuário deve ter autorização. Apenas os processos iniciados pelo usuário podem ser por ele eliminados. O superusuário tem autorização para eliminar qualquer processo, inclusive o 0.

Ex: *kill -9 137*

Neste exemplo, o processo com ID 137 seria imediatamente encerrado e retirado da memória. A opção -9 é um tiro imediato e fulminante.

Para saber o número do processo a ser excluído, o comando é **ps**.

ln - Esse comando é usado para criar ligações (nomes alternativos) para outro arquivo. Quando uma ligação é criada para um arquivo ou outra ligação, todas as mudanças nas ligações são, na realidade, mudanças no arquivo ao qual estão ligadas. Cria links a arquivos ou diretórios.

A instrução **ln** cria o nome *meuperfil* para o arquivo */usr/tsm/.profil*. Qualquer referência ao arquivo *meuperfil* é, na realidade, uma referência à */usr/tsm/.profile*. A ligação pode ser removida com qualquer um dos seguintes comandos:

* *unlink meuperfil*

* *rm meuperfil*

Ex: *ln /usrtsm/.profile meuperfil*

ln [-fs]

ln [-fs] [arquivo ...]

Parâmetros:

-f Cria o link mesmo se o arquivo destino não exista ou não estiver acessível.

-s Cria um link simbólico (soft link).

Obs: **ln** pode criar tanto links simbólicos (soft links) como diretos (hard links); **ln** cria links diretos por default.

ls - É a listagem dos arquivos e diretórios do corrente diretório.

Ex: *ls*

Opções:

* -l : listagem em formato de lista;

* -a : listagem dos arquivos ocultos (arquivos que inicial com “.”);

* -d : listagem dos diretórios;

* -r : ordem reversa;

Ex: *ls -la*

Esse comando faz uma listagem de todos os arquivos, inclusive os ocultos, em forma de lista.

lynx - Esse comando faz você navegar pelas páginas da www. Basta colocar um endereço após o comando **lynx**, para acessá-lo.

Ex: *lynx www.unicruz.tche.br*

Exercício:

Acessar a página da Universidade Federal do Rio Grande do Sul (www.penta.ufrgs.br) e entrar no site “Redes de Computadores”.

mail - Esse comando permite ver os mail's recebidos.

Ex: *mail*

man - Comando que chama o manual com a descrição de todos os parâmetros do comando fornecido.

Ex.: *man <comando>*

Exercício:

Ver o manual do comando **lynx**.

mcopy - Comando este, que transfere um ou mais arquivos para um drive específico. Comando específico para trabalhar com disquetes em formato DOS.

Ex: *mcopy arq a:*

Exercício:

Copiar o arquivo *listao.txt*, que foi buscado do site da UFSM, e descompactado com *gunzip*, para um disquete do drive **a:** .

mkdir - Comando usado para criar diretórios. É um dos comandos fundamentais do UNIX, e todos os usuários devem conhecê-lo.

Ex: *mkdir novodir*

Para criar um diretório em um local específico, é o seguinte.

Ex: *mkdir /usr/tsm/novodir*

Exercício:

Crie o diretório *download* no seu diretório HOME.

mv - O comando é usado para mover um arquivo de um local para outro. Ele funciona de forma similar ao comando **cp**, exceto que o fonte é apagado depois do arquivo ser copiado. O comando **mv** deve ser usado com cuidado. Se o arquiv de destino já existir, o **mv** gravará em cima incondicionalmente. Além disso, se vários arquivos estiverem sendo movidos ao mesmo tempo, o alvo deve existir; caso contrário, o comando **mv** emitirá um diagnóstico.

Ex: *mv /usr/tsm/mbox /usr/wbu*

Nesse exemplo, o arquivo que está sendo copiado é */usr/tsm/mbox*, e o destino é o diretório */usr/wbu*, que já existia. No finalda operação, *mbox* não existirá mais no diretório */usr/tsm*.

Exercício:

Mover o arquivo *listao.txt* para o diretório */download/*.

passwd - Esse comando permite ao usuário trocar sua própria password.

OBS: Essa nova password deve conter números e letras, e não apenas um deles. O comando **passwd** não aceita password's "fáceis" de descobrir.

Ex: *passwd*

Depois de dar o comando, o sistema pedirá para digitar a password antiga. Após digitar, o sistema pedirá a nova password. Então digite o que você quiser. Repita a password. Se a mensagem for "Password Changed", deu certo.

pine - Aplicativo que visualiza, envia, recebe e escreve-se fax.

Ex: *pine*

Principais Funções:

- ? Help;
- C Composição de Mensagens;
- I Visualizar Mensagens que estão na Pasta Corrente;
- L Selecionar Pasta de Mail's;
- A Endereços Particulares;

S Setup;
Q Exit.

C Composição de Mensagens:

To : <endereço do destinatário>
Cc : <endereço de quem manda>
Attachment : <arquivos que o usuário deseja enviar com o mail>
Subject : <título do mail>
----- Message Text -----
<mensagem>

L Selecionar Pasta de Mail's:

Para visualizar um mail novo, selecione a Pasta **INBOX**.

Exercício:

Mandar um mail para um usuário vizinho.

ping - Esse comando vê a comunicação entre seu terminal e o endereço desejado.

Ex: *ping www.ibm.com*

Esse comando é necessário saber, pois com ele, verifica-se a velocidade da rede e se ela está parada. Para sair, click Ctrl-C.

Exercício:

Compare as velocidades de transmissão de dados dos endereços:

- *www.microsoft.com*
- *www.unijui.tche.br*

ps - A finalidade do comando **ps** é reportar a situação de processos ativos no UNIX. O comando **ps** tem várias opções. As opções -e e -f são de maior interesse para os desenvolvedores, já que reportam todas as informações disponíveis para todos os processos. Essas opções são especificadas como -ef, que produz oito colunas de informação, como segue (as principais):

* Coluna UID: identifica a ID do dono do processo. A ID correspondente ao nome de conexão do usuário, como especificado em seu registro no etc/passwd.

* Coluna PID: identifica o número de ID do processo. É preciso conhecer esse número para tomar qualquer atitude com relação ao processo, como usar

o comando **kill**.

- * Coluna PPID: identifica o pai do processo.

- * Coluna STIME: indica a hora em que o processo foi iniciado.

- * Coluna TTY: indica o terminal de controle associado ao processo, e podem conter o caracter ?, significando que o processo não tem um terminal de controle.

- * Coluna TIME: indica o tempo total de execução que o processo acumulou desde quando foi começado.

- * Coluna COMMAND: descreve o nome do processo, indicando qual comando que está sendo executado, bem como seus argumentos.

Ex: *ps -ef*

pwd - Esse comando é usado para mostrar o diretório corrente, e é, na realidade, um acrônimo de *print working directory* (mostre o diretório de trabalho).

Ex: *pwd*

rm - Este comando é usado para remover arquivos do sistema de arquivos do UNIX. As opções são -f, -r e -i. A ação padrão, quando nenhuma opção é indicada, é tentar remover os objetos especificados. Cuidado com o uso de curingas, pois pode ter conseqüências desastrosas! Ao entrar este comando, esteja absolutamente certo do diretório em que será executado.

Ex: *rm **

Este comando deleta tudo o que estiver no diretório corrente. Cuidado com esse comando.

rmdir - Esse comando é usado para remover somente diretórios vazios, e é uma forma relativamente segura de remover diretórios. Supondo que o diretório chamado “/usr/src/cmd/lixo” não contenha arquivos, o comando a seguir pode ser usado para removê-lo do sistema de arquivos:

Ex: *rmdir /usr/src/cmd/lixo*

Nesse exemplo, lixo deve ser um diretório, senão haverá uma mensagem de erro.

talk - Esse comando dá a permissão de conversar com outro usuário por meio do teclado. O outro usuário necessariamente deve estar logado no siste-

ma UNIX.

Ex: *talk <nome_do_usuario>@dinf.unicruz.tche.br*

Para sair, Ctrl-C.

Exercício:

Dê um talk para seu vizinho.

tar - Compacta vários arquivos, transformando em apenas um.

Ex: *tar cvf backup.tar **

“Backup.tar” é o nome do arquivo criado.

* são os arquivos a serem compactados.

v => mostra os arquivos processados;

telnet - Terminal remoto. Emula um terminal virtual do servidor remoto. Possibilita conectar-se em outros computadores da Internet com plataforma Unix.

Ex: *telnet www.unijui.tche.br*

Esse comando conecta o usuário com o servidor “www.unijui.tche.br”. Ao receber mensagem indicando conexão, o sistema irá pedir o seu *login*. Se o usuário estiver cadastrado no servidor, somente digita-se seu login e sua password. Se estiverem corretas, o usuário entrará no sistema automaticamente em seu diretório HOME.

traceroute - Esse comando traça toda a rota que o servidor faz para chegar em um endereço www.

Ex: *traceroute www.ibm.com*

O comando acima traça a rota de servidores até chegar no endereço ibm.com.

Exercício:

Traçar a rota para o endereço da Microsoft. (www.microsoft.com).

vi - O comando chama o editor visual, que é um editor de texto de tela cheia. Esse editor é falto de alguns recursos e carece do esplendor disponível nos ambientes DOS e IBM de grande porte, mas mesmo assim, o **vi** é uma fer-

ramenta útil e poderosa.

Comandos:

k - seta cima;	j - seta baixo
h - seta esquerda	f - seta direita
^ - inicio de linha	\$ - fim de linha
w - avança palavra	3w - avança 3 palavras
b - retorna palavra	3b - retorna palavra
e - fim da palavra	fx - avança até caracter x
Fx - retorna até caracter x	^d - desce 1/2 tela
^u - sobe 1/2 tela	^f - desce 1 tela
^d - sobe 1 tela	H - topo da tela
M - meio de linha	L - fim da tela
xG - vai para linha x	x - deleta caracter
r - substitui caracter	X - deleta caracter anterior
:! - comando shell	:sh - shell
u - restaura arquivo	U - restaura a linha

Pesquisa:

/cadeia - à frente	?cadeia - para trás
n - repete mesmo sentido	N - repete sentido contrário
xyy - coloca linha corrente + (x-1) linhas no buffer	P - coloca buffer apos linha do cursor (até 26 buffer's endereçáveis)
:set list - lista variáveis do vi	:set (no) number - com ou sem numeração se linhas
% - desloca o cursor para o parênteses ou chave que casa com a que está sob o cursor.	:set (no) sm - casamento de parenteses e chaves
:set all - mostra a setagem das variáveis	:set (no) ai - com ou sem auto-identificação
S - substitui caracter e abre inserção	R - substitui caracter (contínuo)
J - junta linhas	dd - deleta linha (buffer)

cw - substitui word	c\$ - substitui até o final da linha
---------------------	--------------------------------------

:q - quit	:q! - saída forçada
:next - próximo arquivo	:rewind - arquivo anterior
:x,y s/velho/novo/g - substitui string	x,y - linha e coluna
g - todas as ocorrências	:w - grava arquivo
:x - grava e sai	dx - deleta x linhas

put - põe conteúdo no buffer	y,x - guarda no buffer x linhas apartir da linha do cursor.
:e! edição forçada, abandona mudanças	
:wq - grava e sai	\ - help

Inserção:

i - antes do cursor	I - início da linha
o - linha abaixo	O - linha acima
A - fim da linha	

Copiar um bloco de texto:

* posiciona no início do bloco;
* dá o comando: 10yy (10 linhas);
* posiciona o cursor e pressiona “p”

w - Esse comando lista os usuários que estão logados no sistema UNIX. A lista mostra o login, a hora em que foi o usuário logou-se, o aplicativo em que encontra-se e a quanto tempo, entre outras informações.

Ex: *w*

wc - O comando **wc** é também chamado o comando de contagem de palavras. Ele pode contar e informar a quantidade de caracteres, palavras e linhas do(s) arquivo(s) especificado(s).

Ex: *wc arquivo*

who - O comando **who** é usado para determinar a quantidade e identidade dos usuários que estão utilizando o sistema UNIX no momento. Além do nome dos usuários, pode também informar a hora de abertura da sessão; o terminal (registro/dev); o PID do interpretador de comando do usuário; e outras informações, todas obtidas no arquivo /etc/utmp que é dinamicamente modificado à medida que cada usuário abre ou fecha uma sessão.

Ex: *who*

Para saber quem está logado no terminal, digite:

Ex: *whoami*

Comandos Unix:

Listagem das funções e comandos está em /usr/man/whatis

pwd

ls

mv

cp

rm

mkdir

rmdir

chmod

chown

chgrp

grep

ln

users | wc -w

loop.c

&

ps

kill

Quando digitamos ^Z mandamos um sinal SIGSTOP. bg irá reiniciar o processo em background.

A lista com os processos ativos é o comando jobs -l. fg coloca-o em foreground

kill -l = mostra os sinais.

Uso da área de swap (memória secundária)

Exemplo de processo em background: remetende de uma mensagem. As mensagens são transferidas no formato texto.

Processos = são gerenciados pelo núcleo do sistema UNIX.

Dutos ou pipes = é um meio de conexão entre a saída de uma aplicação e a entrada de outra.

Executáveis = são os shell scripts e arquivos compilados.

Multitarefa = não há programas residentes mas há processos sendo executados

Arquivos = Cada arquivo está ligado a um inodo. Temos que usar shutdown para evitar discrepâncias na contagem de blocos livres e alocação de inodos que afetarão arquivos e diretórios modificados durante a última sessão operacional do UNIX. O superbloco seria a grossíssimo modo uma FAT, e é gravado periodicamente pelo núcleo.

Multiusuários = programas devem considerar o compartilhamento de arquivos

Disquetes = 1. Deve ter sido formatado; 2. O sistema de arquivo ser válido; 3. Deve ser montado.

Shell Scripts = são os equivalentes aos arquivos .bat do DOS, mas identificados aqui com a extensão *.sh*.

Siglas

FTP - File Transfer Protocol

HTML - Hipertext Markup Language

HTTP - Hipertext Transfer Protocol

ICMP - Internet Control Message Protocol. Controle entre gateways e hosts.

IRC - Internet Relay Chat

TCP-IP - Transmission Control Protocol / Internet Protocol.

ARP, Rarp, ICMP, IGMP, UDP, SMTP, RPC, FTP, Telnet, DNS, SNMP, TFTP.

TCP - Serviço de transporte orientado à conexão

IP- Serviço de rede não orientado à conexão (protocolo do tipo datagrama)

NFS – Network File System – Esquema de compartilhamento de diretórios em rede.

Slip / PPP - Protocolos utilizados na conexão entre dois computadores via modem e linha telefônica.

SLIP - Serial Line IP. Comunicação ponto a ponto assíncrono.

SMTP – Simple Mail Transfer Protocol.

UDP - User Datagram Protocol. Funcionalidades mais simplificadas que o TCP. ex.: DNS

URL - Uniform Resource Locators

WWW - World Wide Web

Bibliografia:

- Redes d Computadores. Das Lans Mans Wans às Redes ATM. 2ª edição. Luiz Fernando Gomes Soares. Guido Lemos. Sérgio Colcher. Editora Campus.

Computes Networks. Third Edition. Andrew S. Tanenbaum. Prentice Hall.

Referências:

<ftp://penta.ufrgs.br/pub/virus/fp-222.zip>
www.procergs.com.br/alap/veiculo.html
www.doc/unicamp.br/~cmrubira/aacesta
www.utexas.edu/world/lecture/cs