Uma Introdução ao Uso da Rede do IME

versão 2

Marco Dimas Gubitoso
gubi@ime.usp.br

25 de março de 1997

Sumário

1 Diferenças com os micros **3**

2 Entrando no sistema **3**
2.1 Terminais 4

3 Sistema de arquivos **5**
3.1 Comandos Básicos 5

4 Processos **6**
4.1 Terminais 'Travados' 7

5 Manuais **7**

6 Outros Usuários **8**

7 Sistemas de Janelas **9**

8 Preparação de Documentos **9**

9 Imprimindo Arquivos **10**

10 Rede **11**
10.1 Mail (Correio eletrônico) 11
10.2 WWW 12
10.3 Mais internet 12

11 Tópicos Avançados **12**
11.1 Shells 13
11.2 Compiladores 13

A 'Netiquete' **14**
A.1 Utilização de recursos 14
A.2 Privacidade 14

B Uma Sessão Inicial **15**

C Exemplo de uma página de manual **18**

Introdução

O UNIX é um sistema operacional multi-usuário que, como tal, organiza e controla o acesso e os programas executados no computador.

As estações de trabalho do IME têm o UNIX como sistema operacional. O objetivo deste texto é apresentar as características básicas de seu uso para aqueles que tem pouca ou nenhuma familiaridade com as estações de trabalho. Em alguns pontos será feita uma comparação com o MS-DOS, embora não seja necessária nenhuma experiência naquele sistema para acompanhar o que segue. Os trechos deste manual que pressupõem uma experiência maior serão escritas com este desenho de letra.

Pelo volume de informações não seria possível incluir tudo num texto como este, existem várias informações disponíveis nas próprias estações de trabalho que podem e *devem* ser consultados, em particular, vale a pena olhar o programa *info*. Além deste, existe um manual 'on line' para consulta rápida, veja a seção 5.

Nos exemplos que seguem, o tipo `typewrite` será usado para designar a impressão do computador na tela. O tipo SMALL CAPS indicará o tipo de informação que você deve fornecer naquele momento e o tipo *inclinado* deve ser digitado exatamente como aparece. Teclas especiais aparecerão como, por exemplo, **ENTER**. A notação **Ctrl**+**tecla** significa que deve-se pressionar a tecla **Ctrl** e — sem retirar o dedo da tecla — pressionar **tecla**, esta combinação é representada geralmente como ˆ tecla em vários textos e no terminal, em geral[1].

Para simplificar o texto, usarei a palavra terminal tanto para consoles como para terminais, usando o termo console apenas onde a diferença for fundamental.

1 Diferenças com os micros

A principal diferença com relação aos micros rodando MS-DOS é que nas estações de trabalho pode existir mais de uma pessoa usando ao mesmo tempo e em geral existem vários programas rodando concorrentemente. Apesar destas características, tem-se a impressão de que se está sozinho no computador.

Qualquer problema com terminal travado ou não respondendo deve ser resolvido de acordo com instruções no manual (preferencialmente) ou pedindo-se ajuda a alguem mais experiente, em último caso.

Deve-se ter em mente que qualquer programa 'esquecido' rodando atrapalha o rendimento dos outros programas; além disso não é possível para outras pessoas parar um programa que não foi iniciado por elas.

2 Entrando no sistema

Para poder entrar nas estações você precisa primeiro de uma conta, obtendo um código de usuário (usercode) e uma senha. Isto é feito através de um 'patrocinador' ou responsável que solicita a abertura eletronicamente. Você pode escolher tanto o código de usuário[2] como a senha. A senha deve ser mantida em segredo, pois com ela qualquer pessoa terá acesso aos seus arquivos e aos processos que você estiver executando. Mais gravemente ainda, a divulgação indiscriminada de um par código de usuário/senha permitirá o acesso aos computadores a pessoas não autorizadas.

De posse destes dados, você pode se dirigir a um terminal desocupado e

[1] Na verdade, esta combinação de teclas gera caracteres de controle, que não são passíveis de impressão, mas um deles provoca a limpeza da tela, outro a mudança de linha no texto, outro indica final de arquivo texto (ˆ D), etc

[2] Eventualmente você pode receber um código de usuário diferente do escolhido, se houver coincidência com outro já existente

iniciar uma sessão. A tela deverá conter a palavra `login:`[3] precedida pelo nome da máquina e seguida pelo cursor, indicando que o sistema pode aceitar um usuário naquele terminal. Usaremos como exemplo a máquina 'fuzarca', mas sua conta vale para qualquer máquina da rede

Entre os dados como segue:

> Exemplo:
> `fuzarca login:`SEU CÓDIGO DE USUÁRIO **Enter**
> `Password:`SUA SENHA **Enter**

A senha não aparecerá na tela, por motivos de segurança.

Se tudo estiver correto, aparecerão algumas mensagens relativas à sessão em que você está, eventualmente algumas informações do sistema, e, se for o caso, o aviso que você tem correspondência (mail).

Depois disto o sistema fica a espera de comandos, o que é indicado por um 'prompt', que pode ser o símbolo `$`, ou algo do tipo `[/home/map/fulano]`, dependendo do sistema e, de qualquer forma, é sempre seguido do cursor.

Seu primeiro passo deve ser executar o programa `cadastro`, e preencher os dados requisitados

Para terminar uma sessão, digite **Ctrl**+**D** ou *exit*. O sistema responderá com um `logout` e o terminal ficará disponível a outro usuário.

NÃO se esqueça de terminar a sessão! Caso contrário você estará permitindo que outras pessoas tenham acesso a seus arquivos além de inibir o uso do terminal, pois outros usuários não tem como saber se você vai voltar logo ou se abandou a máquina, antes de decidirem 'remove-lo'. Se precisar se ausentar por *pouco* tempo, use o *xlock*, mas leia antes apêndice A.

2.1 Terminais

Cada terminal tem características próprias, como número de linhas e colunas, comandos para posicionamento de cursor e disposição das teclas. Por isso, se a tela não estiver se comportando como esperado, pode ser que o terminal não esteja definido corretamente para o programa que está rodando.

Em geral o sistema estará configurado corretamente, mas não estranhe se algumas teclas funcionarem de modo um pouco diferente em algumas situações. O caso mais comum é a tecla para apagar o último caractere digitado,

[3] Se a tela estiver apagada, toque qualquer tecla e o texto ou imagem deverá aparecer.

em alguns programas é $\boxed{\textbf{BackSpace}}$ e em outros $\boxed{\textbf{Delete}}$. Para "arrumar" o terminal a seu gosto veja o comando 'stty'.

Nos terminais gráficos, bem como nos consoles das estações de trabalho, pode-se usar um sistema de janelas para facilitar o uso do equipamento e aproveitar melhor os recursos do UNIX. Veja a seção 7.

3 Sistema de arquivos

Tudo que você produzir no computador, como textos, programas, correspondência, etc, é gravado em "arquivos" para que você tenha acesso posteriormente. Existem comandos para criar, listar, apagar, mover e trocar o nome de arquivos de um modo bastante similar ao MS-DOS.

Os arquivos são agrupados no que se chama *diretórios*. Pode-se imaginar um diretório como uma espécie de pasta onde são colocados os arquivos e/ou outras pastas (subdiretórios)[4]. O comando *pwd* mostrará qual o sub-diretório em que os seus arquivos residem.

Como podem existir diretórios dentro de diretórios, pode-se imaginar que os arquivos ficam distribuídos em forma de árvore, como no MS-DOS. Apesar de isto não ser verdade, pois podem haver conexões entre subdiretórios em 'ramos' diferentes, esta idéia funciona bastante bem para o uso do dia a dia.

3.1 Comandos Básicos

Alguns comandos básicos para manipulação de arquivos e diretórios vêm a seguir. Na maior parte deles pode-se especificar um conjunto de arquivos através dos caracteres coringa '`*`', '`?`'. O caractere '`*`' significa qualquer sequência (vazia ou não) de caracteres, enquanto '`?`' especifica qualquer caractere. Além destes pode-se indicar um caractere dentro de um subconjunto, colocando o conjunto entre colchetes, como por exemplo `[abc]` ou `[0-9]`, neste último caso, estamos especificando qualquer dígito.

ls Lista os arquivos no diretório corrente, é similar ao comando 'dir /w' do MS-DOS e FILES do CANDE. Você pode fornecer o nome de um outro diretório para ver seu conteúdo, por exemplo *ls /* listará os arquivos e subdiretórios do diretório principal da máquina (raiz).

[4] Na verdade a coisa é mais complicada, pois pode-se ter ligações entre diretórios e além disso alguns podem residir em outro disco, ou ainda, em outra máquina.

ll É uma variação do *ls* mais parecida com o 'dir', apresentando os arquivos com mais informações, como tamanho, data e hora da última modificação, além de apresentar o nome do dono as permissões de acesso. Na verdade *ll* é um apelido (ou *alias*) para *ls -l*. Você pode construir seus próprios aliases, se quiser, veja a página 13.

mkdir Constrói um novo subdiretório. Você só pode criar e remover diretórios dentro de seu diretório pessoal. É recomendável a colocação de arquivos em diretórios diferentes, de acordo com a finalidade, para fins de organização. Este comando é equivalente ao comando 'md' do MS-DOS.

rmdir Remove subdiretório. O subdiretório tem que estar vazio para poder ser removido.

rm Remove arquivos. *Atenção ao usar este comando, pois ele é irreversível, ao contrário do que contece no MS-DOS!*.

cp Copia arquivos, na forma *cp <origem> <destino>*.

more Mostra um arquivo na tela (de preferência contendo texto), parando após cada página.

Você só pode agir sobre arquivos de outros usuários se tiver acesso liberado por eles. Para alterar a permissão de acesso a arquivos, use o comando 'chmod'. Com ele você pode impedir que outras pessoas leiam e/ou escrevam em seus arquivos ou ainda impedir a alteração de um arquivo acidentalmente.

O diretório onde seus arquivos se encontram é chamado de 'home'; você pode referenciá-lo através da abreviação ~/. O diretório de outro usuário tem abreviação ~<código do usuário>. Veja o apêndice B.

Outros comandos interessantes : *diff*, *grep*.

4 Processos

Tudo o que é executado num ambiente UNIX tem o nome de processo e possui um número associado. Você pode olhar quais os seus processos através do comando *ps*.

Por ser multi-tarefa, o UNIX permite que você tenha mais de um processo rodando ao mesmo tempo. O modo mais fácil de se obter isto é através de um sistema de janelas, como o OpenWindows (seção 7).

Outra opção é colocar o programa para rodar em 'background', colocando-se um '&' no final da linha de comando, o que pode ser útil também num sistema de janelas, como nos exemplos da seção 8 e do apêndice B. Para obter informações mais específicas consulte o manual sobre o *bash*, *ksh* ou *csh*.

Algumas vezes pode acontecer de um processo ficar suspenso ou parado sem motivo aparente. A causa pode ser, entre outras coisas: erro no programa, problemas no sistema ou algum dispositivo que não está respondendo. Neste caso, para não deixar o terminal 'travado' ou para conseguir finalizar sua sessão, você pode tentar matar o processo. Para tanto use o comando

Exemplo:
kill -9 NÚMERO DO PROCESSO

O número do processo pode ser obtido pelo comando *ps*.

4.1 Terminais 'Travados'

Se o terminal estiver travado, entre na mesma máquina através de um outro terminal (inicie uma nova sessão) e então mate os processos anteriores. Para entrar na mesma máquina a partir de outra veja os comandos *rlogin* e *telnet* no manual do sistema. Em geral, para máquinas da rede do IME, basta digitar o nome da máquina para fazer uma conexão remota.

Lembre-se no entanto que o uso do kill é uma solução drástica. O melhor é tentar achar a causa do problema e tentar saná-la antes de partir para a força bruta. Um caso mais ou menos comum de terminal 'travado" é quando se pressiona **Ctrl**+**S**, na verdade esta combinação de teclas não trava mas congela o terminal, e é útil quando se quer interromper a saída de um programa (por exemplo, uma lista de arquivos de um subdiretório tão grande que não caiba na tela). Para descongelar o terminal usa-se a tecla **Ctrl**+**Q**.

5 Manuais

Existem vários manuais nas proximidades das estações de trabalho para consulta, desde aqueles para iniciantes como manuais mais técnicos. Provavelmente você irá querer consultar com mais frequência aqueles que descrevem os comandos básicos do sistema. Existe uma versão destes manuais para consulta direta na tela através do comando *man*. A sintaxe é imediata :

Exemplo: *man* ÍTEM PROCURADO

O ítem pode ser um nome de comando, uma função de biblioteca de programação, formato de arquivos, etc. Cada entrada no manual é chamada de 'página' muito embora possa ocupar várias delas. Um exemplo de uma destas páginas se encontra no apêndice C.

O nome do ítem pode ser precedido opcionalmente pelo número da seção do manual. Quase sempre uma página do manual faz referência a outras colocando o nome da página seguido pelo número da seção entre parênteses.

Caso não se tenha uma idéia muito clara de qual ítem buscar, pode-se usar a opção *-k* seguida de uma palavra chave, como no exemplo:

Exemplo: *man -k print*

que trará uma lista de resumos de entradas do manual com referência a impressão.

Para iniciar, porque não executar *man man*?

6 Outros Usuários

Como dissemos, pode haver outras pessoas usando o sistema ao mesmo tempo que você. Assim, se houver mais gente na mesma máquina onde você está rodando seus programas, procure ser razoável no seu uso, não rode joguinhos ou 'demos' se houver gente fazendo coisas mais importantes. Se você precisar rodar uma tarefa menos urgente (um cálculo demorado, por exemplo) enquanto outra pessoa estiver usando a máquina para algo imediato ou interativo (digamos, editando a versão final do paper para a conferência de amanhã), considere o uso do comando *nice (1)*[5].

O UNIX é um sistema relativamente aberto, no sentido que você pode ter acesso aos arquivos de outros usuários, até mesmo saber o que eles andam fazendo no momento. No entanto, em geral, você não pode alterar os arquivos nem matar processos de outras pessoas. Na verdade espera-se também que não se fique bisbilhotando nas coisas alheias (veja o apêndice A).

No entanto existe um usuário todo-poderoso, conhecido como super usuário (superuser) que tem acesso total aos arquivos e processos, além de

[5] Isto é, comando *nice* descrito na seção 1 do manual : *man 1 nice*

controlar os processos básicos de manutenção do sistema. Eventualmente você pode receber mensagens avisando sobre problemas ou alterações do sistema, enviados por este usuário, também conhecido como 'root'.

7 Sistemas de Janelas

Nos consoles das máquinas pode ser executado um sistema de interação com usuários baseado em 'janelas'. Uma janela é uma área retangular na tela onde ocorre uma apresentação ou troca de informações com o usuário, por exemplo: pode haver o desenho de um relógio, mostrando a hora corrente, ou então a tela de um editor de textos, ou ainda um 'mini' terminal onde você pode rodar os seus programas.

A vida janelada é muito mais fácil. Imagine que você está escrevendo um texto usando o LaTeX (veja 8), numa janela você tem o texto no editor, noutra você roda o latex e numa terceira você vê o resultado final, ao mesmo tempo, de modo que alterações podem ser feitas com mais rapidez.

O sistemas de janelas para estações de trabalho mais padronizado é o Xwindows, desenvolvido no MIT e que pode ser rodado em várias máquinas diferentes. Para usar o Xwindows digite *startx*.

No IME também usamos uma versão adaptada para as Sun's chamada Openwindows. Se você digitar o comando *openwin* ou *xw* no console de qualquer uma das estações você estará ativando este sistema.

8 Preparação de Documentos

Para editar artigos, teses ou mesmo cartas, existe disponível o formatador de textos LaTeX, baseado no TeX de Donald Knuth e conhecido daqueles que tem usado os micros para escrever seus trabalhos.

Este formatador contém embutidas várias regras de tipografia e comandos para distribuição de texto de forma adequada. Em particular, este manual foi preparado pelo *patex*, que é uma versão do LaTeXpara textos em portugues.

O texto deve ser escrito num editor de textos e ter comandos avisando sobre o início de seções, formatos de tabelas, etc. A partir daí, o texto é passado para o formatador, que divide as linhas, páginas, controla o espaçamento entre os parágrafos, e assim por diante. Algumas pessoas se sentem chateadas em deixar um programa decidir o formato final, mas na verdade tem-se bastante controle sobre o resultado e, de qualquer forma, o programa tem mais informação sobre convenções tipográficas do que a maioria de nós. Existe um manual muito bem escrito sobre o LaTeX disponível para consulta. Os arquivos a serem preparados por este formatador em geral terminam com a extensão *.tex*, por exemplo *tese4.tex*, no caso do *patex*, a extensão *.pat* também é aceita. O resultado formatado possui a final *.dvi* (no exemplo *tese4.dvi*).

Para ver na tela o resultado de uma formatação feita pelo LaTeX use o comando

Exemplo:
xdvi SEU ARQUIVO&

de dentro do Xwindows. Não é necessário escrever o '.dvi' no final do nome. O '&', permite que a janela onde você chamou o comando possa ser usada para outros comandos ao mesmo tempo em que o *xdvi* está rodando.

Quanto ao editor de textos propriamente dito, recomendamos **fortemente** o Emacs, que aparece na configuração default do Xwindows e que também pode ser executado a partir de qualquer terminal. O Emacs possui um sistema de ajuda interna que facilita o seu aprendizado, para ter acesso a este recurso, digite **Ctrl**+**H** **i** de dentro dele. Sugiro que na primeira vez você digite **h** após entrar no info. Para entrar num tutorial sobre o Emacs digite **Ctrl**+**H** **t**. Existe também um manual para consulta. Por ser um editor bastante poderoso e complexo, sugiro que se aprenda os comandos à medida da necessidade.

9 Imprimindo Arquivos

Para imprimir arquivos formatados pelo LaTeX, use o comando *printdvi* seguido do nome do arquivo, sem extensões. No exemplo da seção anterior digitaríamos *printdvi tese4*.

Para imprimir outros arquivos contendo texto puro, pode-se usar o comando *lpr*. Veja também, na página correspondente ao lpr, como verificar a fila de impressão e como cancelar a impressão de alguns de seus arquivos.

10 Rede

As estações de trabalho estão ligadas numa rede entre si e com conexão com outras partes do mundo. Assim é possível se comunicar com outros usuários, pegar arquivos de outras máquinas e trocar correspondência, via 'Internet'.

Para enviar e receber arquivos de outras máquinas, existe o programa *ftp*; para usá-lo, você deve saber o endereço da outra máquina e ter uma conta lá[6]. Este programa pode ser particularmente útil para aqueles que tem uma conta no exterior e precisa pegar algum arquivo. Note no entanto que a conexão para o exterior via ftp é lenta, sendo preferível que se utilize à noite.

Para abrir sessão em outra máquina, existem os comandos *telnet* e *rlogin*. O telnet transforma o seu terminal num terminal da outra máquina. O rlogin faz o mesmo, mas já apresenta o seu código de usuário e eventualmente até dispensa o uso de senha, se a máquina destino tiver um método de acesso adequado.

10.1 Mail (Correio eletrônico)

Provavelmente o recurso que será mais usado com relação a redes será o de correio eletrônico. Este é um método rápido de troca de cartas com pesquisadores de praticamente todo o mundo.

Na verdade, existem vários métodos para enviar e receber mensagens. Se você tiver vontade de usar o emacs, recomendamos *fortemente* o 'vm'. Caso contrário, existe o *mailtool*, disponível num dos menus do OpenWindows.

O vm apresenta o cabeçalho e o conteúdo da mensagem separadamente, permite que se responda diretamente a quem mandou, sem necessidade de se re-escrever o endereço e o código do destinatário, permite salvar cada mensagem num arquivo diferente, se houver necessidade entre outras coisas e, por fazer parte do emacs, tem todos os recursos do editor de textos presente.

Para trocar correspondência com alguém, tudo o que você precisa saber é qual o endereço eletrônico do destinatário. Este endereço em geral tem a forma

Exemplo:
CÓDIGO DE USUÁRIO@MÁQUINA E REDE

Seu endereço na rede do IME é

[6] Vários lugares possuem um usuário especial, chamado "anonymous" que permite o uso do ftp por qualquer pessoa. Estes 'sítios' em geral são centros de distribuição de software.

Exemplo: Seu código@ime.usp.br

veja o meu endereço na primeira página deste manual, para um exemplo concreto.

Ainda sobre endereços, existe uma forma de se especificar 'apelidos' para os endereços que você usar mais frequentemente. Por exemplo, suponha que o endereço do 'Joca' seja 'jstol@stupid.machine.name.edu'. Ao invés de digitar este endereço comprido toda vez que enviar uma mensagem para o Joca, você pode opcionalmente digitar 'joca', desde que tenha registrado o apelido 'joca'. Os registros dos apelidos ficam num arquivo chamado '.mailrc'; se ele não existir no seu diretório, crie-o com um editor de texto. Cada apelido deve ficar numa linha seguindo o modelo:

Exemplo: a joca jstol@stupid.machine.name.edu

Procure mais detalhes na página 'Mail' do manual.

10.2 WWW

O acesso à Teia Mundial (World Wide Web) pode ser feito através dos programas 'netscape' ou 'lynx'.

O netscape deve ser executado dentro de um sistemas de janelas (7), já o lynx trabalha apenas em modo texto.

10.3 Mais internet

Para mais informações sorbe a Internet, procure o livro 'Introdução à Internet e Seu Uso'.

11 Tópicos Avançados

Para aqueles que queiram ou precisem mexer um pouco mais, aqui estão algumas informações mais específicas.

11.1 Shells

O programa que interage com o usuário chama-se 'shell'; é ele que interpreta os comandos que você digita antes de carregar um programa. O correspondente no MS-DOS é o 'command.com' ou o '4dos.com'. No UNIX, existem vários 'shells', cada um com um conjunto de comandos particulares. No IME existem vários: *sh*, *csh*, *bash* e *ksh* e o bash é usado como default para todos por ser mais completo.

É possível escrever programas interpretáveis pelo shell (batch files), para pequenas tarefas. Em particular o arquivo .bash_profile é um programa executado pelo bash toda vez que você entra na máquina (como o autoexec.bat, só que o .bash_profile é rodado como parte do 'login' e não ao ligar a estação). Outros arquivos parecidos são : .bashrc, .cshrc e .xinitrc, que são executados toda vez que se roda um novo shell ou o Xwindows. Como pode se imaginar pelo nome, o .bashrc é a inicialização do bash e o .cshrc a do csh.

Você pode escolher apelidos para seus comandos através do comando *alias*, que tem a sintaxe

Exemplo:
alias -x <novo nome> <linha de comando>

Por exemplo, o comando *ll* está definido como *alias -x ll ls -l*. Para fazer os seus apelidos serem definidos toda vez que iniciar uma sessão, adcione as declarações no arquivo '.bash_profile'.

11.2 Compiladores

Existem alguns compiladores disponíveis, para várias linguagens. O mais usado no UNIX é o compilador C, que é a linguagem materna do sistema. O compilador C é o *cc*, sendo que também existe *gcc*.

Lembre-se também que não existe um sistema integrado. Entretanto, emacs permite rodar o compilador de dentro de si, existem as janelas e o programa *make* para cuidar de projetos.

Os interessados em projetos maiores deveriam ler o manual sobre o próprio *make*, o ligador *ld* e o mantenedor de bibliotecas *ar*, além, obviamente, dos próprios compiladores. Existem muitos outros manuais a serem lidos, de acordo com as necessidades, não se acanhem neste ponto.

A ‘Netiquete’

O fato de termos um sistema multi-usuário implica em ‘regras de convivência’. Acreditamos que a regra básica é o bom senso e que ninguém vai querer intencionalmente prejudicar alguém, no entanto a experiência mostra que é muito fácil atrapalhar pessoas quando se é novo num sistema destes, mesmo que não se tenha tal intenção.

Para minimizar problemas potenciais, aqui estão alguns pontos do que se considera um comportamento razoável.

A.1 Utilização de recursos

- Por enquanto não temos muitas impressoras disponíveis e a capacidade de papel nelas é relativamente pequeno. Desta forma evite imprimir documentos muito grandes se houver muitas pessoas querendo imprimir alguma coisa. Você sempre pode imprimir num outro horário.

  Se acabar o papel da impressora, providencie para que ele seja reposto.

  Se houver alguma listagem ‘esquecida’ na impressora, não custa nada colocá-la no armário ao lado.

  Procure não esquecer suas listagens na máquina.

  Quanto a listagens velhas, existe uma caixa ao lado da impressora onde devem ser colocados os papéis para reciclagem.

- O número de usuários que podem estar ativos em cada máquina é finito, procure não deixar suas sessões abertas e em particular não abuse do *xlock*. Principalmente se considerando que a maior parte das vezes se abre uma sessão do console da máquina, com isto você estará impedindo que outra pessoa possa usufruir do sistema.

- Os computadores têm alguns jogos e demonstrações interessantes e que estão aí para serem jogados e vistos, mas procure não fazer barulho se outras pessoas estiverem trabalhando e evite o uso se houver uma carga grande no sistema.

A.2 Privacidade

- As estações de trabalho possuem mecanismos para proteger a privacidade dos usuários, restringindo a leitura de alguns arquivos, por exemplo.

Como nem sempre o usuário se preocupa ou lembra de proteger algo que deseja manter privado (texto de uma prova é um bom exemplo). Obviamente bisbilhotar não é considerado razoável.

Ah, sim, não se esqueça de proteger o texto de suas provas.

- Não divulge a sua senha, veja o ítem anterior.

B Uma Sessão Inicial

Este é um roteiro para uma primeira sessão, onde se vê um pouco das coisas comentadas aqui e se dão algumas dicas a mais para quem está iniciando. Vamos supor que seu código de usuário seja 'pipoca' e sua senha inicial '43excuse', e que estamos na 'feijoada'.

O primeiro passo é entrar na máquina. Como foi visto, procure uma que estiver desocupada e digite seu código de usuário e sua senha inicial (lembre-se que a senha não aparecerá na tela):

> Exemplo:
> `feijoada login:`*pipoca* **Enter**
> `Password:`*43excuse* **Enter**

Feito isto, aparecerão na tela as algumas mensagens indicando a última vez em que você entrou na máquina (se é que você entrou alguma vez), a versão do sistema operacional, a data e a hora e eventualmente um aviso de que existe correspondência.

Na linha seguinte aparecerá um 'prompt', mostrando o número do comando, o nome da máquina e o seu diretório 'home':

> Exemplo:
> <1>`[feijoada:/home/depto/pipoca]` >

Feito isso, vamos experimentar alguns comandos.

> Exemplo:
> $< n >$`[feijoada:/home/depto/pipoca]` >*ls* **Enter**

Serão mostrados os arquivos presentes no seu subdiretório. Veja como fica a versão *ll*, que fornece uma lista completa.

Podemos aproveitar para mudar a sua senha. Escolha uma senha que tenha pelo menos 7 caracteres, misture letras com dígitos e/ou símbolos (&, %, @), e use o seguinte comando

> Exemplo:
> $< n >$`[feijoada:/home/depto/pipoca]` $>$*passwd* **Enter**
> `Changing NIS password for pipoca on feijoada.`
> `Old password:`*43excuse*
> `New password:`

Você tem que saber a senha antiga para poder trocá-la, é por isso que o sistema pede novamente. Aqui também ela não aparecerá na tela. Depois disso digite a senha nova seguida por um **Enter** e confirme.

Agora vamos entrar no sistema de janelas:

> Exemplo:
> $< n >$`[feijoada:/home/depto/pipoca]` $>$*xw*

A tela piscará, mostrando que o 'OpenWindows' está sendo carregado (às vezes demora alguns segundos, não se assuste) e aparecerão algumas janelas: uma 'cmdtool(Console)' no canto superior esquerdo, um 'File Manager' logo abaixo e um 'Tutorial' no canto superior direito. Aparecerá também uma setinha, indicando a posição apontada pelo 'mouse'[7].

Procure movimentar o 'mouse' e veja como a seta se move na tela. A seta será usada para selecionar a janela de trabalho e para ativar as várias opções disponíveis.

Vamos selecionar a janela de tutorial: movimente a seta até que ela entre na janela e pressione o botão da esquerda. A barra com o título da janela mudará de cor. Leia e siga as instruções até se sentir mais à vontade com o sistema.

Existe um menu de programas que pode ser ativado pressionando-se o botão da direita enquanto a seta estiver apontando para o 'fundo' da tela, isto é, fora de qualquer janela. Se você fizer isto, deverá aparecer uma janelinha com as seguintes opções:

[7]Para quem não sabe, é aquela caixinha com 3 botões e um fio comprido que fica ao lado das estações, sobre uma placa azul.

- xterm
- Programs
- Utilities
- EMACS
- Properties...
- New Xdefaults
- Restart...
- Exit...

A opção xterm trará uma nova janela para trabalho, como se fosse mais um terminal.

Os itens 'Programs' e 'Utilities', se selecionados, abrem novos menus. O 'mailtool' citado anteriormente, pode ser ativado de dentro do submenu 'Programs'. Para ativar uma opção, posicione a seta sobre ela e aperte o btão da direita, como descrito no tutorial. Verifique as opões dentro do 'Programs'. E dê uma olhada também na opção 'Properties...', principalmente se você estiver numa estação com monitor colorido.

Por ora vamos ver como funciona o Emacs...

Selecione a opção 'EMACS'. Aparecerá uma janela relativamente grande no canto inferior direito da tela. Para aprender a mexer neste editor, digite **Ctrl**+**H** **t**, e você terá um tutorial para seguir. Para interromper, digite **Ctrl**+**X** **Ctrl**+**Z**, e responda as eventuais questões de acordo.

Como brincadeira final, vá até a janela 'cmdtool' e digite

Exemplo:
< *n* >`[feijoada:/home/depto/pipoca]` >*xdvi ~gubi/manual/unix&*

para ver a versão mais recente deste manual e observe que a janela 'cmdtool' continua disponível para outros comandos. Saia do *xdvi* selecionando 'Quit' e tente a linha de cima sem o '&' no final e veja a diferença.

Para sair do sistema de janelas, acione a opção 'Exit...' do menu principal. Não se esqueça que sua sessão ainda estará aberta, digite *exit*, ou **Ctrl**+**D** para terminar.

C Exemplo de uma página de manual

```
rtfm(l)              UNIX Programmer´s Manual              rtfm(l)



NAME
     rtfm - read the fucking manual

SYNOPSIS
     rtfm

OPTIONS
     None, you have to read the manual for an answer.

DESCRIPTION
     Used when lazy people ask stupid questions.  Normaly cried
     out in vain.

FILES
     /dev/null

ENVIRONMENT
     Any.

SEE ALSO
     man(1)

DIAGNOSTICS
     Is a  diagnostic.  Since you are reading this you are get-
     ting the idea.

BUGS
     Ha!


Printed 1/25/90               local                           1
```