Para os novatos em FreeBSD e UNIX®

Annelise Anderson <andrsn@andrsn.stanford.edu >
Revisão: 43184

FreeBSD is a registered trademark of the FreeBSD Foundation.

IBM, AIX, OS/2, PowerPC, PS/2, S/390, and ThinkPad are trademarks of International Business Machines Corporation in the United States, other countries, or both.

Microsoft, IntelliMouse, MS-DOS, Outlook, Windows, Windows Media and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Netscape and the Netscape Navigator are registered trademarks of Netscape Communications Corporation in the U.S. and other countries.

Motif, OSF/1, and UNIX are registered trademarks and IT DialTone and The Open Group are trademarks of The Open Group in the United States and other countries.

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this document, and the FreeBSD Project was aware of the trademark claim, the designations have been followed by the “™” or the “®” symbol.

15 de agosto de 1997 por .

Resumo

Parabéns pela instalação do FreeBSD! Esta introdução é para os novatos no FreeBSD *e* no UNIX®—, então ela começa com o básico. Este artigo assume que você está usando a versão 2.0.5, ou mais atual, do FreeBSD distribuído pela FreeBSD.org, seu sistema, por agora, tem um único usuário (você) e você provavelmente está muito bem com o DOS/Windows® ou OS/2®.

Índice

1. Entrando e saindo do sistema 2
2. Adicionando um Usuário com Privilégios de Root 3
3. Explorando 3
4. Obtendo Ajuda e Informação 4
5. Editando Texto 6
6. Imprimindo Arquivos no DOS 8
7. Outros comando úteis 9
8. Próximos Passos 10
9. Seu Ambiente de Trabalho 11
10. Outros 13
11. Comentários São Bem-vindos 13

1. Entrando e saindo do sistema

Entre no sistema (quando você vê `login:`) como o usuário que você criou durante a instalação ou como `root`. (Sua instalação do FreeBSD já terá uma conta `root`; que pode ir para qualquer lugar e fazer qualquer coisa, incluindo remover arquivos essenciais, então seja muito cuidadoso!) Os símbolos `%` e `#` nos exemplos a seguir representam o `prompt` (o seu pode ser diferente), com o `%` indicando o `prompt` de um usuário comum e o `#` indicando o `prompt` do `root`.

Para sair do sistema (e obter uma novo `prompt` de `login:`) escreva:

```
# exit
```

quantas vezes forem necessárias. Você precisa pressionar enter após os comandos, e lembre-se que UNIX® é sensível a letras maiúsculas e minúsculas — `exit` não é o mesmo que `EXIT`.

Para desligar a máquina escreva:

```
# /sbin/shutdown --h now
```

Ou para reinicializar, escreva:

```
# /sbin/shutdown --r now
```

ou

```
# /sbin/reboot
```

Você também pode reiniciar com Ctrl+Alt+Delete. Dê-lhe um pouco de tempo para trabalhar. Isso é o equivalente ao `/sbin/reboot` nas versões recentes do FreeBSD e é muito, muito melhor do que pressionar o botão de `reset`. Você não quer ter que instalar tudo de novo, não é?

2. Adicionando um Usuário com Privilégios de Root

Se você não criou nenhum usuário durante a instalação do sistema e, portanto, está logado como `root`, você provavelmente precisa criar um usuário agora com:

```
# adduser
```

Na primeira vez que você usar o `adduser` , ele pode pedir por valores padrões para salvar. Você pode querer definir o `shell` padrão como csh(1) ao invés do sh(1), se ele sugerir o `sh` como padrão. Do contrário, apenas pressione enter para aceitar os valores padrões. Os valores padrões serão salvos em `/etc/adduser.conf` , o qual pode ser editado.

Suponha que você criou um usuário `jack`, cujo nome completo seja *Jack Benimble*. Dê a `jack` uma senha se segurança (mesmo crianças ao redor que possam por as mãos no teclado) é um problema. Quando for questionado se você deseja incluir `jack` em outros grupos, escreva `wheel`:

```
Login group is ``jack''. Invite jack into other groups: wheel
```

Isso tornará possível entrar no sistema como `jack` e usar o comando su(1) para tornar-se `root`. Então você não será mais repreendido por logar como `root`.

Você pode interromper o `adduser` à qualquer momento apenas pressionando Ctrl+C, e no fim você poderá aprovar o novo usuário ou simplesmente escrever n para não. Você pode querer criar um segundo usuário para o caso de algo sair errado na edição dos arquivos de `login` do usuário `jack`.

Uma vez que você tenha concluído, use `exit` para voltar ao `prompt` de `login` e entrar como o usuário `jack`. Em geral, é uma boa idéia fazer tudo quanto for possível como um usuário comum, que não tem o poder — e o risco — do `root`.

Se você já criou o usuário e quer que ele tenha permissão de utilizar o `su` para tornar-se `root`, você pode entrar como `root` e editar o arquivo `/etc/group` , adicionando `jack` ao grupo presente na primeira linha (o grupo `wheel`). Mas primeiro você precisa praticar com vi(1), o editor de texto instalado nas versões mais recentes do FreeBSD — ou usar um editor de texto mais simples, como o ee(1).

Para remover um usuário, use o comando `rmuser`.

3. Explorando

Ao entrar como um usuário comum, explore e tente alguns comandos que irão acessar as fontes de ajuda e informação do FreeBSD.

Aqui estão alguns comandos e o que eles fazem:

`id`
: Diz quem você é!

`pwd`
: Mostra onde você está — o diretório corrente de trabalho

`ls`
: Lista os arquivos no diretório corrente.

`ls -F`
: Lista os arquivos no diretório corrente com um * depois de arquivos executáveis, um `/` depois de diretórios, e um `@` depois de `links` simbólicos.

`ls -l`
: Lista os arquivos com detalhes — tamanho, data, e permissões.

`ls -a`
: Lista os arquivos ocultos, que iniciam com "ponto", com os outros. Se você está como `root`, os arquivos ocultos, que iniciam com "ponto", são mostrados sem a necessidade da opção `-a`.

`cd`
: Muda o diretório corrente. `cd ..` sobe um nível com relação ao diretório atual; note o espaço depois do `cd`. `cd /usr/local` entra no diretório especificado. `cd ~` entra no diretório `home` do usuário logado — e.g., `/usr/home/jack` . Tente `cd /cdrom`, e execute `ls`, para descobrir se seu CDROM está montado e funcionando.

`view filename`
: Permite que você visualize um arquivo (chamado *`filename`*) sem modificar seu conteúdo. Tente `view /etc/fstab` . Escreva `:q` para sair.

`cat filename`
: Mostra o conteúdo de *`filename`* na tela. Se ele é muito longo e você só consegue ver o final do arquivo, pressione ScrollLock e use up-arrow para navegar até o topo do arquivo; você pode usar ScrollLock também com páginas de manual. Pressione ScrollLock novamente para interromper o rolamento de conteúdo. Você pode querer experimentar o `cat` em alguns arquivos ocultos no seu diretório `home` — `cat .cshrc` , `cat .login` , `cat .profile` .

Você vai encontrar `aliases` em seu `.cshrc` para alguns comandos `ls` (estes são muito convenientes). Você pode criar outros `aliases` editando `.cshrc`. Você pode criar `aliases` disponíveis para todos os usuários colocando-os no arquivo de configuração principal do `csh` o qual afeta todo o sistema, o `/etc/csh.cshrc` .

4. Obtendo Ajuda e Informação

Aqui estão algumas fontes de ajuda úteis. *`Text`* representa um termo de sua escolha, para o qual você precisa de informação ou ajuda — usualmente um comando ou arquivo.

`apropos text`
: Tudo que contiver o texto *`text`* na whatis database.

`man text`
: Exibe a página de manual do *`text`*. A maior fonte de documentação para sistemas UNIX®. `man ls` vai lhe mostrar todos os detalhes de como usar o comando `ls`. Pressione Enter para navegar através do texto, Ctrl+B para voltar uma página, Ctrl+F para avançar uma página, q ou Ctrl+C para sair.

`which text`
: Informa onde, no `path` do usuário, o comando *`text`* é encontrado.

`locate text`
: Informa todos os caminhos onde o termo *`text`* é encontrado.

`whatis text`
: Informa o que o comando *`text`* faz e qual sua página de manual. Executar `whatis *` vai lhe informar sobre todos os binários no diretório corrente.

`whereis text`
: Encontra o arquivo *`text`*, informando seu caminho completo.

Você pode querer experimentar usar `whatis` em alguns comandos comuns, como `cat`, `more`, `grep`, `mv`, `find`, `tar`, `chmod`, `chown`, `date`, e `script`. `more` permite que você leia uma página de cada vez, do mesmo modo como no DOS, e.g., `ls -l | more` ou `more filename`. O * funciona como curinga — e.g., `ls w*` vai mostrar os arquivos que começam com `w`.

Algum desses programas não está trabalhando muito bem? Ambos, locate(1) e whatis(1), dependem de uma base de dados recompilada semanalmente. Se sua máquina não vai permanecer ligada (e rodando o FreeBSD) durante o final de semana, convém executar manualmente os comandos de manutenção diários, semanais, e mensais de vez em quando. Execute-os como `root` e, por agora, dê a cada um deles um tempo para finalizar antes de você iniciar o próximo.

```
# periodic daily
output omitted
# periodic weekly
output omitted
# periodic monthly
output omitted
```

Se você cansar de esperar, pressione Alt+F2 para acessar outro *virtual console*, e entre no sistema novamente. Afinal, ele é um sistema multiusuário e multitarefa. No entanto, é provável que esses comandos exibam mensagens na sua tela enquanto eles estão rodando; você pode executar `clear` em seu `prompt` para limpar a tela. Uma vez que eles tenham executado, você pode querer olhar em `/var/mail/root` e `/var/log/messages`.

Executar tais comandos é parte da administração do sistema — e como o único usuário do sistema UNIX®, você é seu próprio administrador de sistemas. Praticamente tudo que você precisa fazer como `root` será para administração de sistemas. Tais responsabilidades não são muito bem exploradas, mesmo nos grossos livros sobre UNIX®, que parecem dedicar um grande espaço para opções de menus em gerenciadores de janelas. Você pode querer obter um dos dois livros principais sobre administração de sistemas, ou Evi Nemeth et.al.'s *UNIX System Administration Handbook* (Prentice-Hall, 1995, ISBN 0-13-15051-7) — a segunda edição da capa vermelha; ou Æleen Frisch's *Essential System Administration* (O'Reilly & Associates, 2002, ISBN 0-596-00343-9). Eu uso Nemeth.

5. Editando Texto

Para configurar seu sistema, você precisará editar arquivos de texto. Muitos deles estarão no diretório `/etc`; e você precisará do `su` para tornar-se `root` e poder modificá-los. Você pode utilizar o `ee`, por ser fácil de usar, mas à longo prazo vale a pena aprender o editor de texto `vi`. Um excelente tutorial sobre o `vi` pode ser encontrado em `/usr/src/contrib/nvi/docs/tutorial` , se você tiver os fontes do sistema instalado.

Antes de editar um arquivo, você provavelmente deveria fazer um backup dele. Suponha que você queira editar o `/etc/rc.conf` . Você pode simplesmente usar `cd /etc` para ir até o diretório `/etc` e fazer:

```
# cp rc.conf rc.conf.orig
```

Isso vai copiar o `rc.conf` para `rc.conf.orig` , e depois você poderá copiar o `rc.conf.orig` para `rc.conf` para recuperar o original. Mas o melhor mesmo seria mover (renomear) e copiar novamente:

```
# mv rc.conf rc.conf.orig
# cp rc.conf.orig rc.conf
```

pelo fato do comando `mv` preservar a data e o dono originais do arquivo. Você pode agora editar o `rc.conf`. Se você quiser o original de volta, você faria **`mv rc.conf rc.conf.myedit`** (assumindo que você queira preservar a versão modificada) e então:

```
# mv rc.conf.orig rc.conf
```

para voltar as coisas do jeito que estavam.

Para editar um arquivo, faça:

```
# vi filename
```

Mova-se através do texto com as setas do teclado. Esc (a tecla de escape) coloca o `vi` em modo de comando. Aqui estão alguns comandos:

x
: Remove o caractere onde está o cursor

`dd`
: remove a linha inteira (mesmo que ela quebre na tela)

`i`
: para inserir texto a partir do cursor

`a`
: para inserir texto após o cursor

Uma vez que você digite `i` ou `a`, você pode inserir o texto. `Esc` coloca você de volta no modo de comando

`:w`
: para salvar suas modificações no disco e continuar editando

`:wq`
: para salvar as modificações e sair

`:q!`
: para sair sem salvar as modificações

`/text`
: para mover o curso para *`text`*; `/Enter` (a tecla enter) para encontrar a próxima ocorrência de *`text`*.

`G`
: vai para o final do arquivo

`nG`
: vai para a linha *n* no arquivo, onde *n* é o número.

Ctrl+L
: para redesenhar a tela

Ctrl+b e Ctrl+f
: volta e avança na tela, respectivamente, assim como fazem no `more` e `view`.

Pratique com o `vi` em seu diretório `home`, criando um novo arquivo com `vi filename`, adicionando e removendo texto, salvando o arquivo, e chamando-o de novo. `vi` oferece muitas surpresas, pois ele é realmente bastante complexo, e algumas vezes você vai inadvertidamente executar um comando que vai fazer alguma coisa que você não espera. (Algumas pessoas realmente gostam do `vi` — ele é mais poderoso que o DOS EDIT — procure sobre o comando `:r`.) Use Esc uma ou mais vezes para estar seguro de que você está no modo de comando e continuar a partir daí se você tiver problemas, salve frequentemente com `:w`, e use `:q!` para sair e começar novamente (a partir do seu último `:w`) quando você precisar.

Agora você pode entrar no diretório `/etc` com o `cd`, usar o `su` para tornar-se `root`, usar o `vi` para editar o arquivo `/etc/group` , e adicionar um usuário no grupo `wheel` para que ele tenha privilégios de `root`. Só adicione uma vírgula e o `login` do usuário no fim da primeira linha do arquivo, pressione Esc, e use `:wq` para escrever suas alterações no disco e sair. Efeito instantâneo. (Você não colocou um espaço depois da vírgula, colocou?)

6. Imprimindo Arquivos no DOS

Neste ponto você provavelmente não tem uma impressora funcionando, então aqui vai uma maneira de criar um arquivo a partir de uma página de manual, movê-lo para um disquete, e então imprimi-lo do DOS. Suponhamos que você queira ler cuidadosamente sobre mudança de permissões em arquivos (muito importante). Você pode usar `man chmod` para ler a respeito. O comando

```
% man chmod -| col --b > chmod.txt
```

irá remover códigos de formatação e enviar a página de manual para o arquivo `chmod.txt` em vez de mostrá-lo na tela. Agora coloque um disquete formatado no DOS em seu drive de disquete `a`, use o `su` para tornar-se `root`, e escreva

```
# /sbin/mount --t msdosfs -/dev/fd0 -/mnt
```

para montar o drive de disquete em `/mnt`.

Agora (você não precisa mais estar como `root`, e você pode executar `exit` para voltar para o usuário inicial `jack`) você pode ir até o diretório onde você criou o `chmod.txt` e copiar o arquivo para o disquete com:

```
% cp chmod.txt -/mnt
```

e usar `ls /mnt` para obter a listagem do diretório `/mnt`, que deveria mostrar o arquivo `chmod.txt` .

Você pode querer criar um arquivo a partir do `/sbin/dmesg` executando:

```
% /sbin/dmesg > dmesg.txt
```

e copiar o `dmesg.txt` para o disquete. `/sbin/dmesg` é o registro das mensagens de `boot`, e ele é útil para entender o que o FreeBSD encontra durante a inicialização. Se você enviar perguntas para a lista de discussão para perguntas gerais sobre o FreeBSD ou para o grupo da USENET — como "O FreeBSD não encontra a minha unidade de fita, o que eu faço?" — as pessoas vão querer saber o que o `dmesg` diz.

Você pode desmontar o drive de disquete agora (como `root`) para retirá-lo com:

```
# /sbin/umount -/mnt
```

e reiniciar para ir para o DOS. Copie os arquivos para um diretório do DOS, chame-os com o DOS EDIT, Windows® Notepad ou Wordpad, ou algum outro processador de texto, faça uma pequena alteração para o arquivo ser salvo, e imprima como você normalmente faz a partir do DOS ou Windows. Espero que funcione! Páginas de manual saem melhor se impressas com o comando `print` do DOS. (Copiar arquivos do FreeBSD para uma partição DOS montada ainda é, em alguns casos, um pouco arriscado.)

Obter uma impressora imprimindo do FreeBSD envolve criar uma entrada apropriada em `/etc/printcap` e criar um diretório de `spool` correspondente em `/var/spool/output`. Se sua impressora está na lpt0 (nos DOS é chamada de LPT1), você só precisa ir para `/var/spool/output` e (como `root`) criar o diretório `lpd` executando: `mkdir lpd`, se ele ainda não existe. Em seguida, a impressora deve responder se ela estiver ligada durante a inicialização, e `lp` ou `lpr` deve enviar um arquivo para a impressora. Se o arquivo vai ser impresso ou não, depende da configuração, esta é coberta no FreeBSD handbook.

7. Outros comando úteis

`df`
: mostra os dispositivos montados e o espaço em disco.

`ps aux`
: mostra os processos rodando. `ps ax` exibe uma lista compacta.

`rm filename`
: exclui o arquivo *`filename`*.

`rm -R dir`
: remove o diretório *`dir`* e todos os seus subdiretórios — seja cuidadoso!

`ls -R`
: lista os arquivos no diretório corrente e todos os subdiretórios; eu usei uma variante, `ls -AFR > where.txt`, para obter uma lista de todos os arquivos de `/` e (separadamente) `/usr`, antes de descobrir um jeito melhor de encontrar arquivos.

`passwd`
: para mudar a senha do usuário (ou a senha do `root`)

`man hier`
: exibe a página de manual sobre o sistema de arquivos do UNIX®

Use `find` para localizar `filename` em `/usr`, ou qualquer de seus subdiretórios, com

```
% find -/usr --name -"filename"
```

Você pode usar * como curinga em "*`filename`*" (que deve estar entre aspas). Se você diz para `find` procurar em `/`, em vez de `/usr`, ele vai procurar o arquivo em todos os dispositivos montados, incluindo CDROM e as partições DOS.

Um excelente livro que explica os comandos e utilitários UNIX® é Abrahams & Larson, *Unix for the Impatient* (2nd ed., Addison-Wesley, 1996). Existe também uma grande quantidade de informações sobre UNIX® na Internet.

8. Próximos Passos

Agora você deve ter as ferramentas que você precisa para explorar e editar arquivos, então você pode ter tudo ligado e funcionando. Existe uma grande quantidade de informações no FreeBSD handbook (que provavelmente está em seu disco rígido) e no web site do FreeBSD. Uma grande variedade de pacotes e `ports` estão disponível no CDROM, bem como no site web. O handbook diz mais sobre como usá-los (obter um pacote, se ele existir, com `pkg_add /cdrom/packages/All/nomepacote` , onde *`nomepacote`* é o nome do pacote). O CDROM tem uma lista dos pacotes e `ports` com uma breve descrição em `cdrom/packages/index` , `cdrom/packages/index.txt` , e `cdrom/ports/index` , com as descrições completas em `/cdrom/ports/*/*/pkg/DESCR` , onde os * representam subdiretórios das categorias e dos nomes dos programas, respectivamente.

Se você achar o handbook muito sofisticado (com `lndir` e tudo) sobre a instalação de `ports` a partir do CDROM, aqui está o que normalmente funciona:

Encontre o `port` que você quer, digamos que seja o `kermit`. Haverá um diretório para ele no CDROM. Copie o subdiretório para `/usr/local` (um bom lugar para adicionar programas que estarão disponíveis para todos os usuários) com:

```
# cp --R -/cdrom/ports/comm/kermit -/usr/local
```

Isso deve resultar em um subdiretório `/usr/local/kermit` onde estarão todos os arquivos do subdiretório `kermit` do CDROM.

Em seguida, crie o diretório `/usr/ports/distfiles` , se ele ainda não existe, usando `mkdir`. Agora verifique em `/cdrom/ports/distfiles` por um arquivo com o nome que indique o `port` que você quer. Copie o arquivo para `/usr/ports/distfiles` ; nas versões recentes você pode pular este passo, o FreeBSD vai fazer isso por você. No caso do `kermit` não existe `distfile`.

Vá até o subdiretório `/usr/local/kermit` , onde estará o arquivo `Makefile`. E execute

```
# make all install
```

Durante este processo o `port` vai obter a partir do FTP quaisquer arquivos compactados que sejam necessários e não estejam presentes no CDROM ou em `/usr/ports/distfiles` . Se sua rede ainda não está funcionando e não existe arquivo para o `port` em `/cdrom/ports/distfiles` , você vai precisar obter este arquivo de outra máquina e copiá-lo para `/usr/ports/distfiles` de um disquete ou de sua partição DOS. Leia o `Makefile` (com o `cat`, ou `more`, ou `view`) para descobrir aonde ir (o site de distribuição principal) para obter o arquivo e qual o seu nome. O nome do arquivo será truncado se ele for obtido a

partir do DOS, e depois de colocá-lo em `/usr/ports/distfiles` você precisa renomear o arquivo (com o comando `mv`) para seu nome original para que ele possa ser encontrado. (Use o modo de transferência binária!) Então volte para `/usr/local/kermit`, encontre o diretório com o `Makefile`, e execute `make all install`.

Outra coisa que pode acontecer quando da instalação de um `port` ou pacote é que algum outro programa seja necessário. Se a instalação parar com uma mensagem can't find unzip ou algo parecido, você pode precisar instalar o pacote ou `port` do unzip antes de continuar.

Uma vez instalado, execute `rehash` para fazer o FreeBSD reler os arquivos no `path`, então ele saberá o que existe lá. (Se você obter uma série de mensagens path not found quando usar `whereis` ou `which`, você pode querer adicionar entradas para a lista de diretórios na declaração do `path` no `.cshrc` em seu diretório `home`. A declaração do `path` no UNIX® funciona do mesmo modo que no DOS, exceto pelo diretório corrente que não é (por padrão) incluído no `path` por razões de segurança; você precisa digitar `./` antes do comando para executa-lo; sem espaços depois da barra.

Você pode querer obter a versão mais recente do Netscape® a partir de seu site FTP. (O Netscape® requer o `X Window System`.) Agora existe uma versão para o FreeBSD, então explore com cuidado. Basta usar o `gunzip filename` e `tar xvf filename` no arquivo, mover o binário para `/usr/local/bin` ou algum outro lugar onde os binários são mantidos, execute `rehash`, e então coloque as seguintes linhas no `.cshrc` de cada diretório `home` dos usuários ou (mais fácil) em `/etc/csh.cshrc`, o arquivo de configuração principal de inicialização do `csh`:

```
setenv XKEYSYMDB -/usr/X11R6/lib/X11/XKeysymDB
setenv XNLSPATH -/usr/X11R6/lib/X11/nls
```

Isso assume que o arquivo `XKeysymDB` e o diretório `nls` estão em `/usr/X11R6/lib/X11`; se eles não estiverem, encontre-os e coloque-os lá.

Se você instalou o Netscape® originalmente a partir de um `port` usando o CDROM (ou o site FTP), não substitua o `/usr/local/bin/netscape` com o novo binário do Netscape®; isso é só um `shell script` que define as variáveis de ambiente para você. Em vez disso, renomeie o novo binário para `netscape.bin` e mova o binário antigo para `/usr/local/netscape/netscape`.

9. Seu Ambiente de Trabalho

Seu `shell` é uma parte muito importante do seu ambiente de trabalho. No DOS, o `shell` usual é o `command.com`. O `shell` é quem interpreta os comandos que você executa na linha de comando, e, portanto, se comunica com o resto do sistema operacional. Você também pode escrever `shell scripts`, que são como arquivos de lotes do DOS: uma série de comando para serem executados sem sua intervenção.

O FreeBSD vem com dois `shells` instalados: `csh` e `sh`. `csh` é bom para trabalhar com a linha de comando, mas `scripts` devem ser escritos com `sh` (ou `bash`). Você pode descobrir qual `shell` você está utilizando executando `echo $SHELL`.

O `shell csh` é bom, mas `tcsh` faz tudo que o `csh` faz e mais. Ele aceita que você chame novamente os comandos, com as setas do teclado, e edite-os. Ele completa nomes de arquivos através da tecla TAB (`csh` usa a tecla Esc), e ele permite que você mude para seu último diretório com `cd -`. É muito fácil alternar seu `prompt` com `tcsh`. Ele torna sua vida muito fácil.

Aqui estão os passos para a instalação de um novo `shell`:

1. Instale o `shell` a partir de um `port` ou pacote, como você faria para qualquer outro `port` ou pacote. Use `rehash` e `which tcsh` (assumindo que você instalou o `tcsh`) para ter certeza que ele foi instalado.

2. Como `root`, edite o `/etc/shells`, adicionando uma linha no arquivo para o novo `shell`, nesse caso `/usr/local/bin/tcsh`, e salve o arquivo. (Alguns `ports` vão fazer isso por você.)

3. Use o comando `chsh` para mudar seu `shell` para o `tcsh` permanentemente, ou execute `tcsh` no `prompt` para mudar seu `shell` sem precisar realizar o `login` novamente.

Nota

Pode ser perigoso mudar o `shell` do `root` para qualquer coisa diferente de `sh` ou `csh` nas versões anteriores do FreeBSD e em muitas outras versões do UNIX®; você pode não ter um `shell` funcional quando o sistema colocar você em modo monousuário. A solução é usar `su -m` para tornar-se `root`, o que lhe dará o `tcsh` como `root`, pois o `shell` é parte do seu ambiente. Você pode tornar isso permanente adicionando para seu arquivo `.tcshrc` um `alias` com:

```
alias su su --m
```

Quando `tcsh` inicia, ele lê os arquivos `/etc/csh.cshrc` e `/etc/csh.login`, assim como o `csh`. Ele também irá ler o arquivo `.login` em seu diretório `home` e o arquivo `.cshrc`, a menos que você providencie um arquivo `.tcshrc`. Para isso você pode simplesmente copiar o `.cshrc` para `.tcshrc`.

Agora que você já instalou o `tcsh`, você pode ajustar o seu `prompt`. Você encontra os detalhes na página de manual do `tcsh`, mas aqui está uma linha para você incluir em

seu `.tcshrc` que vai lhe dizer quantos comandos você digitou, que horas são, e em que diretório você se encontra. Ele também mostra um > se você está como um usuário comum e um # se você está como `root`, mas `tcsh` vai fazer isso em qualquer caso:

set prompt = "%h %t %~ %# "

Isso deve ser incluído no mesmo lugar do `set prompt` existente, ou sob `"if($?prompt) then"` se este não existir. Comente a linha antiga; você poderá voltar para o `prompt` antigo se preferir. Não esqueça o espaço e as aspas. Você pode recarregar o `.tcshrc` executando `source .tcshrc` .

Você pode obter uma lista de outras variáveis de ambiente que foram definidas executando `env` no `prompt`. O resultado vai lhe mostrar seu editor padrão, paginador, e o tipo de terminal, e possivelmente muitos outros. Um comando útil se você está logado remotamente e não pode executar um programa, pois o terminal não é compatível, é `setenv TERM vt100` .

10. Outros

Como `root` você pode desmontar o CDROM com `/sbin/umount /cdrom` , retirá-lo do `drive`, inserir outro, e montá-lo com`/sbin/mount_cd9660 /dev/cd0a /cdrom`,assumindo que cd0a é o nome do seu dispositivo de CDROM. As versões mais recentes do FreeBSD deixam você montar o CDROM apenas com `/sbin/mount /cdrom` .

Usar o sistema de arquivos `live` — o segundo dos discos de CDROM do FreeBSD — é útil se você tem um espaço limitado. O conteúdo do sistema de arquivos `live` varia de versão para versão. Você pode tentar jogar a partir do CDROM. Isso envolve o uso do `lndir`, que é instalado com o `X Window System` , para informar aos programas onde encontrar os arquivos necessários, pois eles estão no sistema de arquivos de `/cdrom`, em vez de `/usr` e seus subdiretórios, que é o local onde eles esperam estar. Consulte `man lndir`.

11. Comentários São Bem-vindos

Se você usou este guia, eu estou interessado em saber onde ele não está claro, o que foi deixado de fora que você acha que deve ser incluído, e se ele foi útil. Meus agradecimentos para Eugene W. Stark, professor de ciência da computação em SUNY-Stony Brook, e John Fieber pelos comentários úteis.

Annelise Anderson, <andrsn@andrsn.stanford.edu >