

# Introdução ao **FreeBSD** para estações e servidores.

Fernando França <fernando@desconstruindo.eng.br>

versão 1.0.0, 31/05/2011

## Resumo

Este documento descreve um tutorial para instalação e configuração básicas do sistema operacional **FreeBSD** que pode ser utilizado para estações de trabalho e servidores.

## Sumário

<b>Lista de figuras</b>	<b>2</b>
<b>1 Notas do autor</b>	<b>2</b>
<b>2 Introdução</b>	<b>3</b>
<b>3 Por que <b>FreeBSD</b></b>	<b>3</b>
<b>4 Obtendo o <b>FreeBSD</b></b>	<b>4</b>
<b>5 Instalação</b>	<b>4</b>
5.1 Iniciando . . . . .	4
5.2 Discos e particionamento . . . . .	5
5.3 Tipo de instalação . . . . .	10
<b>6 Pós-instalação</b>	<b>11</b>
6.1 Configurações de rede . . . . .	11
6.2 Configurações de console . . . . .	13
6.3 Configurações de fuso horário . . . . .	14
6.4 Compatibilidade com Linux . . . . .	14
6.5 Configurações de mouse . . . . .	15
6.6 Instalação de pacotes . . . . .	15
6.7 Usuários e grupos . . . . .	16
6.8 Finalizando . . . . .	17
<b>7 Primeiro login e algumas configurações</b>	<b>17</b>
7.1 Shell e variáveis de ambiente . . . . .	17
7.2 O arquivo /etc/rc.conf . . . . .	18
7.3 O arquivo /etc/sysctl.conf . . . . .	19
7.4 O arquivo /etc/ttys . . . . .	19
7.5 O sistema de arquivos /proc . . . . .	19
<b>8 Suporte a hardware</b>	<b>20</b>
8.1 Módulos no kernel . . . . .	20

<b>9</b>	<b>Instalação e manutenção de programas</b>	<b>20</b>
9.1	Atualização do ports . . . . .	20
9.2	Manutenção de programas . . . . .	21
<b>10</b>	<b>Estações de trabalho</b>	<b>23</b>
10.1	Ambiente gráfico básico . . . . .	23
<b>11</b>	<b>Conclusão</b>	<b>24</b>
<b>12</b>	<b>Informações adicionais</b>	<b>24</b>
12.1	Referências . . . . .	24
12.2	Licença de uso . . . . .	25

## Lista de Figuras

1	Menu de inicialização . . . . .	4
2	Seleção de país e idioma . . . . .	5
3	Menu principal do sysinstall . . . . .	5
4	Seleção de discos para o fdisk . . . . .	6
5	Editor de partições fdisk . . . . .	6
6	Gerenciamento de boot . . . . .	7
7	Editor de slices disklabel . . . . .	8
8	Definindo o tamanho em disco . . . . .	8
9	Definindo o tipo do slice . . . . .	9
10	Definindo o ponto de montagem . . . . .	9
11	Finalizado o particionamento . . . . .	10
12	Escolhendo a distribuição . . . . .	10
13	Escolhendo a mídia de instalação . . . . .	11
14	Suporte a interface de rede . . . . .	12
15	Configurando a interface de rede . . . . .	12
16	Configurando serviços de rede . . . . .	13
17	Configurando opções do console . . . . .	14
18	Escolhendo descanso de tela . . . . .	14
19	Configurando o mouse . . . . .	15
20	Seleção de pacotes . . . . .	15
21	Selecionando o pacote bash para instalação . . . . .	16
22	Adicionando usuários e grupos . . . . .	16
23	Configurando novo usuário . . . . .	17

## 1 Notas do autor

Este documento é distribuído na esperança de ser útil porém **SEM NENHUMA GARANTIA** expressa ou implícita. O autor **NÃO ASSUME A RESPONSABILIDADE** por erros ou por danos em decorrência do uso das informações apresentadas.

A licença de uso deste documento é descrita na última seção, por favor revise-a pois ao prosseguir com a leitura estará automaticamente aceitando e concordando com seus termos.

## 2 Introdução

O objetivo deste tutorial é apresentar de forma resumida os passos necessários para instalação e configuração básicas do sistema operacional **FreeBSD** que poderá ser utilizado tanto como estação de trabalho quanto servidor de uso geral. Todo processo é descrito em forma de um roteiro e está baseado no **FreeBSD Handbook**<sup>1</sup> onde os tópicos são cobertos de forma completa e detalhada.

## 3 Por que **FreeBSD**

Existem alguns bons motivos para escolher o **FreeBSD**. Este sistema está longe de ser um desconhecido mas não tem (ainda) a popularidade que o Linux alcançou nos últimos anos quando falamos por exemplo de estações de trabalho, então ajudar a divulgar este sistema também é um dos objetivos deste tutorial.

Tratando-se de servidores sua popularidade é maior pois o **FreeBSD** é um sistema operacional que se encaixa muito bem para administradores de redes e sistemas pois além de ser rápido e ter grande compatibilidade com diferentes tipos e modelos de hardware, possui suporte a multiprocessamento (SMP) e tem inúmeros programas à disposição.

O **FreeBSD** pode ser executado tanto em máquinas Intel antigas, máquinas Spark e os mais novos processadores com vários um núcleos. A partir da versão 7.0, o **FreeBSD** após um trabalho de sete anos lançou o SMPng, um redesenho do suporte de seu kernel a sistemas multiprocessados. Muitas funcionalidades foram implementadas e desde então o **FreeBSD** se tornou um sistema totalmente preparado para processamento paralelo.

Testes<sup>2</sup> feitos em sistemas multiprocessados entre versões anteriores apresentam ganho de performance entre 35% e 45%, e quando comparado ao Linux (kernel 2.6.22) a performance é 15% maior trabalhando com os sistemas de gerenciamento de bancos de dados PostgreSQL e MySQL.

Outro bom motivo a se considerar é sua alta compatibilidade tanto com Unix quanto com Linux. Sendo um usuário Linux, muita similaridade será vista entre as estruturas de diretórios e aplicativos, sem falar no fato do **FreeBSD** possuir um modo de suporte e compatibilidade com binários Linux, permitindo que muitos programas compilados para Linux sejam executados sem problemas. Usuários do Mac OS X também devem se sentir muito confortáveis visto que o kernel executado neste sistema é o Darwin que é baseado no **FreeBSD**.

A segurança também deve ser mencionada. Realmente o **FreeBSD** não é como o OpenBSD mas implementa ótimos sistemas de segurança como listas de acesso de controle (ACLs), controle de acesso obrigatório (MAC), criptografia nos sistemas de arquivos (inclusive swap) entre outros. Seu sistemas de arquivos UFS também oferece ótima disponibilidade, segurança e desempenho.

Por último e não menos importante, documentação. A documentação do **FreeBSD** é altamente organizada, rica, legível e disponível em diversos formatos. O fato do desenvolvimento e documentação serem bem centralizados facilita na resolução de problemas. Inclusive este tutorial não teria um propósito se não fosse o do resumo e apresentação como formas de incentivo e divulgação do **FreeBSD** pois TODAS as informações contidas aqui saíram da documentação oficial e você sempre será encorajado a consultar nela, não é preciso reinventar a roda.

## 4 Obtendo o FreeBSD

Ao ser produzido este documento o FreeBSD se encontrava na versão 8.2, já com a versão 9.0 a caminho e ainda suportando a família 7 na versão 7.4. Podemos obter o FreeBSD diretamente através de seu sítio oficial<sup>3</sup>. É interessante escolher um servidor próximo (mirror) para que a carga seja feita em menor tempo.

Para esta instalação usaremos a versão RELEASE para a plataforma i386. Existe a opção de utilizar a versão CURRENT (atualizada constantemente) e outras plataformas (o FreeBSD suporta as plataformas de processadores: amd64, i386, ia64, pc98, powerpc e sparc64).

Existem também outros sistemas baseados no FreeBSD: PCBSD, DesktopBSD e FreeSBIE. Os dois primeiros voltados para estações de trabalho e o último um live cd. Este documento pretende uma instalação simples e limpa podendo ser aplicada para estações ou servidores.

## 5 Instalação

A partir deste ponto assume-se que você possua algum conhecimento mesmo que básico em sistemas operacionais pois este tutorial não é para leigos (para isso procure FreeBSD for Dummies na Internet :-). Todas as imagens do processo de instalação foram retiradas do FreeBSD Handbook<sup>1</sup>.

### 5.1 Iniciando

Após terminar a carga e gravar a imagem do FreeBSD em cd ou dvd, devemos fazer um levantamento de toda configuração da máquina, é necessário configurar a BIOS para o computador inicializar pelo cd ou dvd, executando em seguida o menu principal:



Figura 1: Menu de inicialização

Todo processo de instalação é bem claro e intuitivo, para a opção padrão de instalação basta apertar [Enter], ou escolha outra opção como por exemplo [2] para desabilitar o suporte a ACPI no kernel (pode ser útil para máquinas problemáticas). O processo de inicialização para máquinas Alpha e Spark é diferente e está descrito em detalhes no FreeBSD Handbook<sup>1</sup>.

Após o sistema realizar um mapeamento do hardware na inicia o programa de instalação **sysinstall**, nele usamos as setas e tab para navegar e barra de espaço ou enter para selecionar uma opção. Abaixo a seleção de país:



Figura 2: Seleção de país e idioma

Em seguida se apresenta a tela principal do sysinstall onde escolhemos a modalidade de instalação, opções de configuração e atualização do sistema. Para prosseguir com a instalação selecione a opção **Standard**:

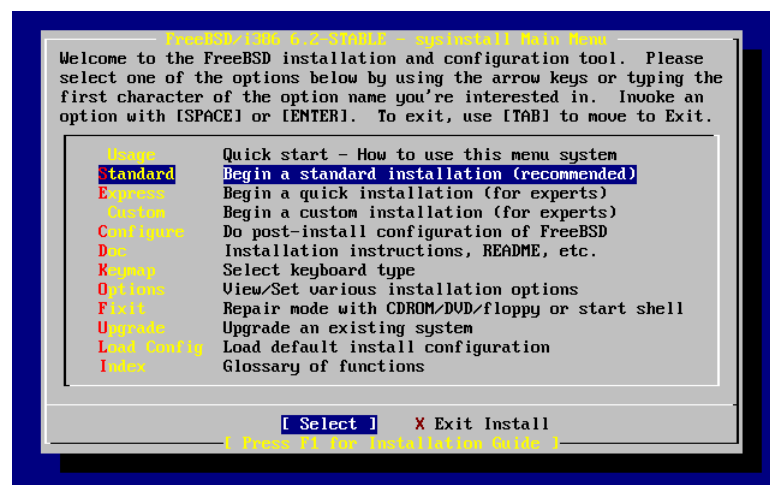


Figura 3: Menu principal do sysinstall

## 5.2 Discos e particionamento

O passo seguinte é a alocação de espaço em disco com o programa **fdisk**, usamos as setas para escolher o disco (caso exista mais de um) no qual queremos alocar espaço:

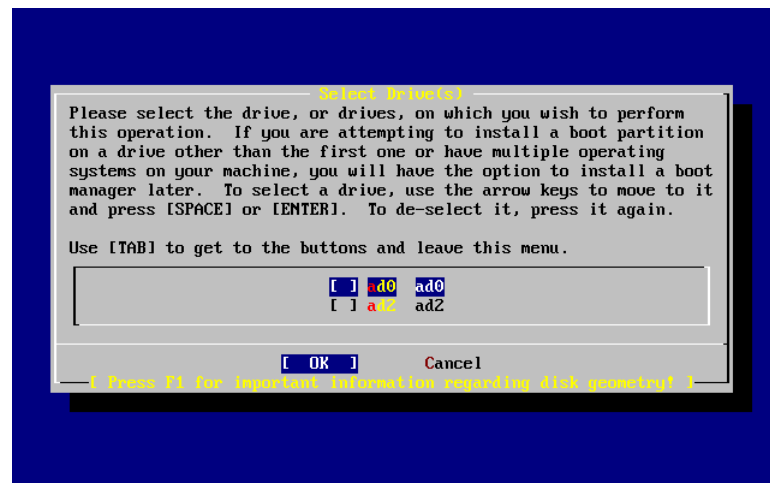


Figura 4: Seleção de discos para o fdisk

Acima percebemos que o primeiro disco está como **ad0** e o segundo como **ad2**. Isto acontece pois o **FreeBSD** se refere as controladoras onde os discos estão em ordem, **ad0** é o disco primário da primeira controladora e **ad2** é o disco primário da segunda controladora.

Esse comportamento é bem inteligente pois caso ele seguisse sequencialmente nomeando o disco primário da primeira controladora como ad0 e o primário da segunda controladora como ad1 e posteriormente você adicionasse um disco secundário na primeira controladora toda essa ordem mudaria e o disco secundário da primeira controladora se tornaria o ad1 e o disco primário da segunda controladora se tornaria o ad2 forçando você a reconfigurar seu sistema.

Ao seleccionar o disco somos direcionados para o programa fdisk onde visualizamos o disco e todos os *slices* existentes:

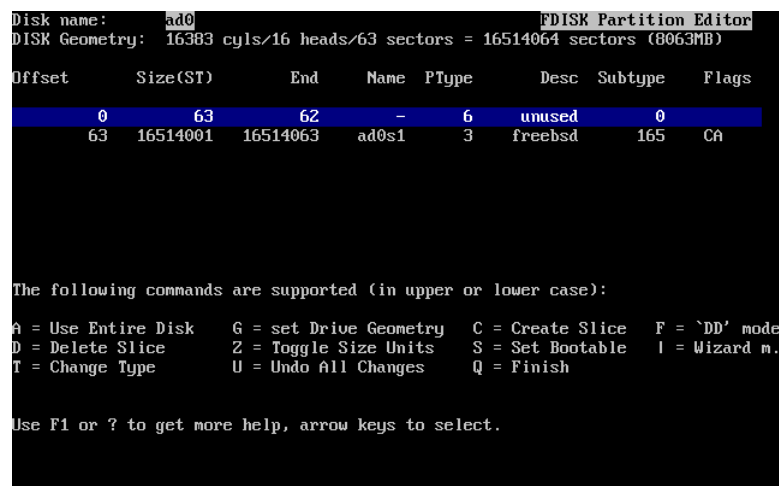


Figura 5: Editor de partições fdisk

Na parte superior está o nome do disco com qual estamos trabalhando. Seguindo os comandos que estão na parte inferior podemos usar o **D** para remover uma partição e o **C** para criar uma nova, alocando espaço para a instalação. Caso queira usar todo o disco na instalação, basta digitar **A**. Note que ainda não criamos os sistemas de arquivos (root, var, swap e etc.), somente

particionando o disco. Para finalizar usamos a opção **Q**.

Após particionar o disco vamos instalar um boot manager, caso tenha mais de um disco e esteja instalando o **FreeBSD** em um disco que não seja o primeiro na controladora primária (ad0) OU esteja instalando o **FreeBSD** no mesmo disco junto a outro sistema operacional (dual boot) escolha a opção **BootMgr**. Se o **FreeBSD** for instalado sozinho no primeiro disco da primeira controladora e for o único sistema operacional na máquina pode-se escolher a opção **Standard**. Por último, se estiver utilizando gerenciador de boot de terceiros, então escolha a opção **None**:

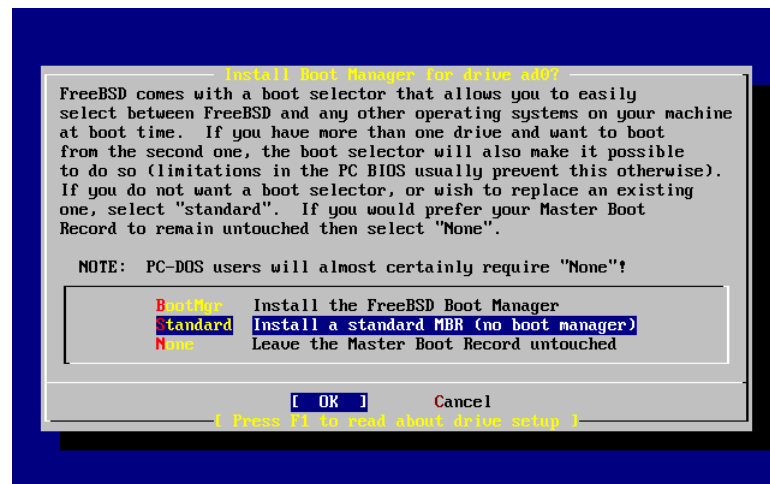


Figura 6: Gerenciamento de boot

Caso a máquina possua mais de um disco o processo anterior se repetirá, para a seleção de discos e particionamento (*ver figuras 4 e 5*) e todo processo deve ser repetido para que este disco também esteja pronto para receber o sistema. O único detalhe está no fato de caso o **FreeBSD** esteja sendo instalado em outro disco que não o primeiro disco, ambos devem ter o **FreeBSD** Boot Manager instalado (opção **BootMgr** na figura 6).

A próxima etapa é a criação dos sistemas de arquivos com o programa disklabel, novamente temos na parte superior a indicação do disco com o qual estamos trabalhando e abaixo os comandos disponíveis. Caso tenha mais de um disco na máquina usamos a seta para cima e baixo para selecionar em qual disco criaremos partição.

Para criar uma partição usamos a opção **C** e em seguida digitamos o tamanho que esta ocupará no disco:



Figura 7: Editor de slices disklabel



Figura 8: Definindo o tamanho em disco



Em seguida definimos se esta nova partição será usada para swap ou como parte do sistema:



Figura 9: Definindo o tipo do slice

Caso seja utilizada como parte do sistema de arquivos precisamos informar seu ponto de montagem:



Figura 10: Definindo o ponto de montagem

Este processo deve ser repetido para todas as partições existentes.

Podemos também optar por um particionamento padrão automático com a opção **A**. Após finalizar toda alocação do espaço em disco, digitamos **Q** para finalizar e sair:



Figura 11: Finalizado o particionamento

### 5.3 Tipo de instalação

Em seguida temos as opções de distribuição (pacotes de programas a serem instalados). Como foi dito o objetivo dessa instalação é ser mais limpa possível para permitir uma customização que atenda ao ambiente em questão e por conta disso escolhemos a opção **Minimal**. Todas as opções possuem uma detalhada descrição dos pacotes que a compõe:

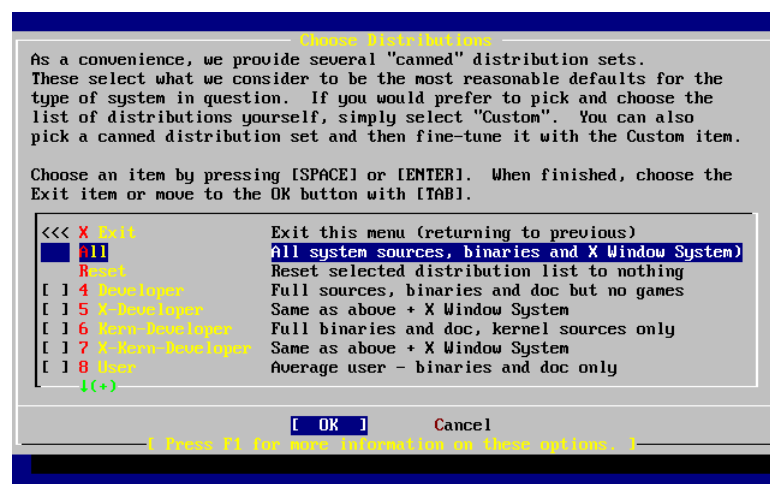


Figura 12: Escolhendo a distribuição

Após a escolha da distribuição o instalador oferece a opção para instalação da árvore de Ports, podemos selecionar **No** por hora pois vamos instalar mais tarde através do programa **portsnap**. Para sair da seleção de pacotes selecionamos a opção **X Exit**.

Na tela de seleção da mídia para instalação escolhemos a opção **1 CD/DVD** caso tenhamos a mídia completa ou a opção **2 FTP** para baixar a instalação através da Internet caso tenhamos um uma mídia do tipo bootonly:

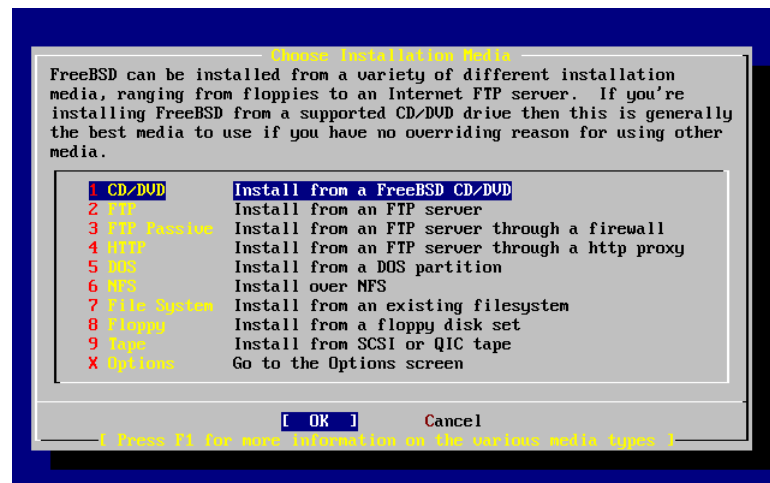


Figura 13: Escolhendo a mídia de instalação

Agora começa o processo de instalação, estando certo de tudo que foi definido basta selecionar **Yes** e a mensagem em seguida informa que a instalação ocorreu com sucesso OU que algo saiu errado, dando a possibilidade de voltar a tela principal do sysinstall (problemas que podem ocorrer na instalação são cobertos de forma detalhada no [FreeBSD Handbook](#)<sup>1</sup>).

## 6 Pós-instalação

Concluída a instalação com sucesso o sysinstall apresenta a configuração pós-instalação. Caso a máquina seja reinicializada por qualquer motivo é possível retornar a este menu com o comando **sysinstall** e em seguida a opção **Configure**.

### 6.1 Configurações de rede

A primeira configuração é a de rede, o sistema pergunta se desejamos configurar uma interface Ethernet ou SLIP/PPP, selecione **Yes**.

Os nomes das interfaces estão relacionados aos seus respectivos módulos carregados no kernel, por exemplo: **rl0** será a primeira interface de rede que é suportada pelo módulo **rl** (Realtek) que está carregado no kernel. Caso sua interface de rede não apareça, sendo exibidas somente as interfaces serial e/ou paralela, é porque o kernel não identificou/não suporta este modelo. Algo que se resolve quase sempre compilando o kernel.

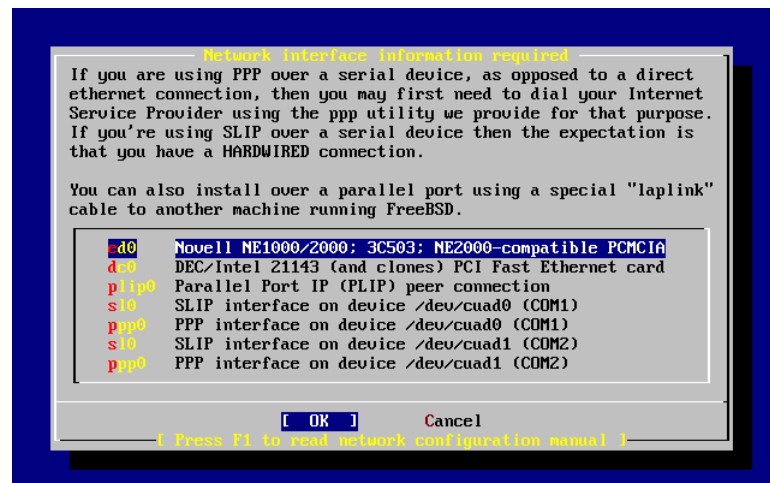


Figura 14: Suporte a interface de rede

Caso seja necessário suporte IPv6 esta opção será apresentada, aqui selecionamos **No**, e igualmente para a pergunta que segue sobre configurar a interface através de DHCP (a menos que sua rede tenha um servidor DHCP claro). A configuração manual da interface de rede é demonstrada abaixo, onde novamente utilizando **Tab** fazemos a navegação entre os parâmetros de configuração e ao finalizar selecionamos **Ok**:

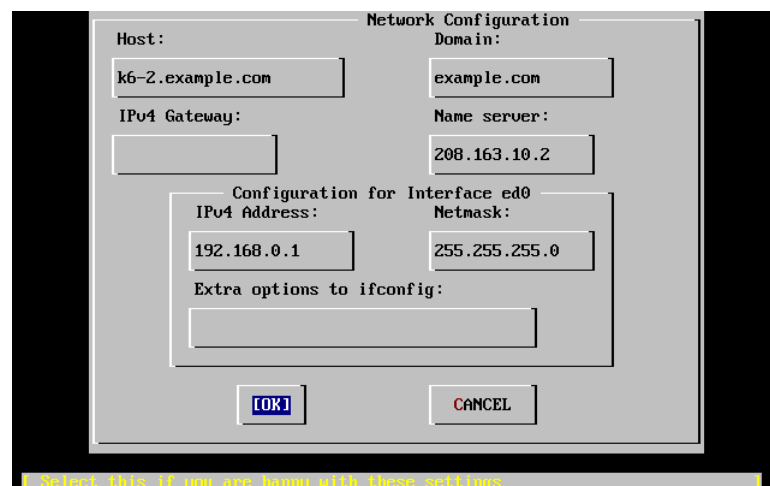


Figura 15: Configurando a interface de rede

Caso a máquina atue como gateway de rede, selecione **Yes** na tela seguinte: **Do you want this machine to function as a network gateway?**

O mesmo para a tela posterior: **Do you want to configure inetd and the network services that it provides?** Caso deseje prover serviços de rede como ftp, telnet. O instalador permitirá a configuração destes serviços através da edição do arquivo `inetd.conf` localizado no diretório `/etc`. Para habilitar um serviço basta descomentar a linha onde ele se encontra, ao finalizar pressione **Esc** para ter a opção de salvar e sair do editor:

```

^_ (escape) menu    ^_ search prompt    ^_ delete line    ^_ prev li    ^_ prev page
^_ o ascii code    ^_ x search    ^_ l undelete line    ^_ n next li    ^_ u next page
^_ u end of file    ^_ a begin of line    ^_ w delete word    ^_ b back 1 char
^_ t begin of file    ^_ e end of line    ^_ r restore word    ^_ f forward 1 char
^_ c command    ^_ d delete char    ^_ j undelete char    ^_ z next word

L: 1 C: 1 =====
# $FreeBSD: src/etc/inetd.conf,v 1.72 2006/08/31 17:15:10 obrien Exp $
#
# Internet server configuration database
#
# Define *both* IPv4 and IPv6 entries for dual-stack support.
# To disable a service, comment it out by prefixing the line with '#'.
# To enable a service, remove the '#' at the beginning of the line.
#
#ftp    stream  tcp    nowait  root    /usr/libexec/ftpd    ftpd -l
#ftp    stream  tcp6   nowait  root    /usr/libexec/ftpd    ftpd -l
#ftp    stream  tcp    nowait  root    /usr/libexec/lukemftpd  ftpd -l -r
#ftp    stream  tcp6   nowait  root    /usr/libexec/lukemftpd  ftpd -l -r
#ssh    stream  tcp    nowait  root    /usr/sbin/sshd        sshd -i -4
#ssh    stream  tcp6   nowait  root    /usr/sbin/sshd        sshd -i -6
#telnet stream  tcp    nowait  root    /usr/libexec/telnetd   telnetd
#telnet stream  tcp6   nowait  root    /usr/libexec/telnetd   telnetd
#shell  stream  tcp    nowait  root    /usr/libexec/rshd      rshd
#shell  stream  tcp6   nowait  root    /usr/libexec/rshd      rshd

```

Figura 16: Configurando serviços de rede

Em seguida o configurador pergunta se desejamos habilitar login via SSH, é recomendável escolher **Yes**. Caso queira configurar os detalhes do serviço basta editar o arquivo `sshd.config` dentro do diretório `/etc/ssh/`. Interessante por uma questão de segurança descomentar a opção **“PermitRootLogin no”** no arquivo de configuração do ssh. Para reiniciar o serviço execute o comando `/etc/rc.d/sshd restart`.

Para as opções dos serviços seguintes oferecidos pelo configurador: FTP anônimo, servidor NFS e cliente NFS escolheremos **No**. Esses serviços não são essenciais e podem ser facilmente configurados posteriormente. Veja o [FreeBSD Handbook](#)<sup>1</sup> para detalhes.

## 6.2 Configurações de console

Neste ponto definimos parâmetros das configurações do console como: tamanho da fonte, mapa de teclado e taxa de repetição, descanso de tela e quantidade de terminais disponíveis:

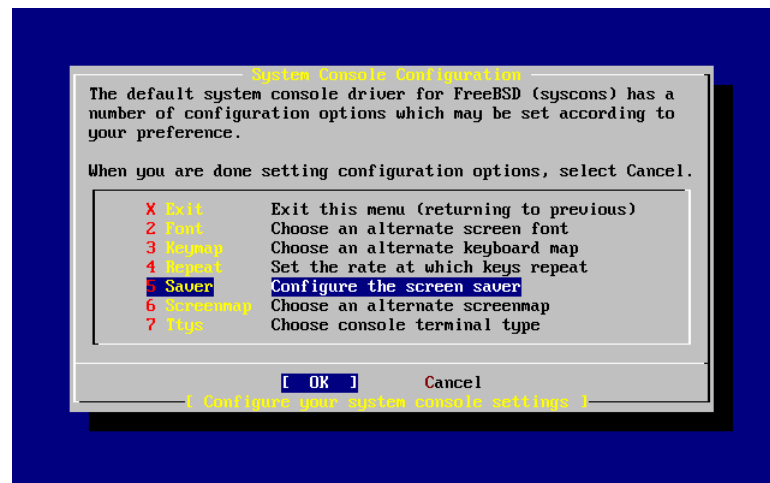


Figura 17: Configurando opções do console

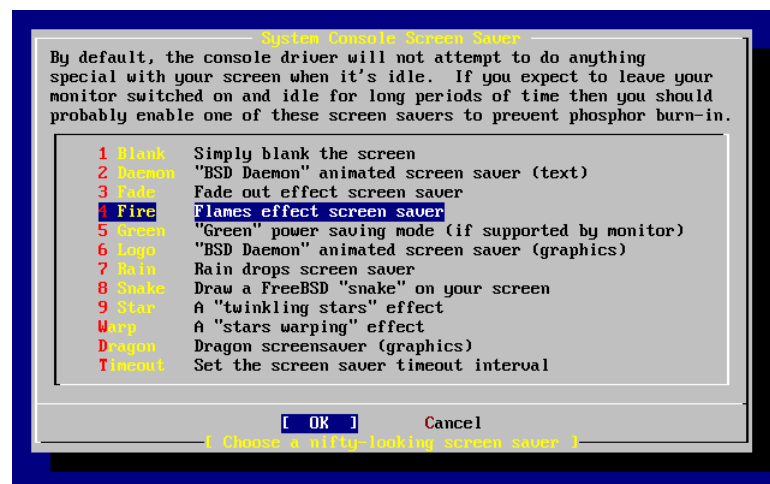


Figura 18: Escolhendo descanso de tela

Ao finalizar as configurações do prosseguimos com a opção **X Exit**.

### 6.3 Configurações de fuso horário

O configurador apresenta a opção para configurar fuso horário (timezone) da máquina. Para tal basta selecionar **Yes** e nas telas seguintes selecionar a região, o país e para finalizar o fuso horário. Caso a BIOS da máquina esteja configurada para hora UTC responda **Yes** para que a configuração de hora do sistema seja feita corretamente.

### 6.4 Compatibilidade com Linux

Em seguida temos a opção para habilitar a compatibilidade com binários Linux. Esta possibilidade é interessante e consiste basicamente da instalação de um pacote e carga do respectivo módulo no kernel, através desta funcionalidade podemos instalar softwares compilados para Linux.

## 6.5 Configurações de mouse

Caso o mouse seja ps/2 ou serial escolhemos **Yes** e caso seja usb ou não possua um mouse instalado, **No**. Através do menu que segue podemos habilitar, escolher o tipo e porta, testar ou desabilitar o mouse. Ao finalizar selecionamos **X Exit**:

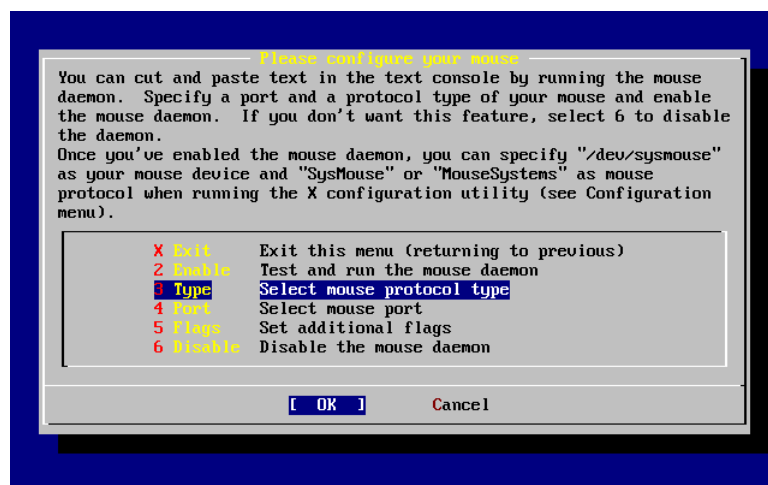


Figura 19: Configurando o mouse

## 6.6 Instalação de pacotes

Agora temos a possibilidade de instalar alguns pacotes no sistema, basta escolher **Yes** para visualizar a listagem. Pacotes são binários de programas já compilados e faremos a instalação de alguns poucos programas, posteriormente veremos com detalhes a instalação de programas no sistema através de pacotes e da árvore de Ports.

Os pacotes são separados por categoria, selecione o programa a ser instalado e em seguida **Ok** para retornar ao menu principal:

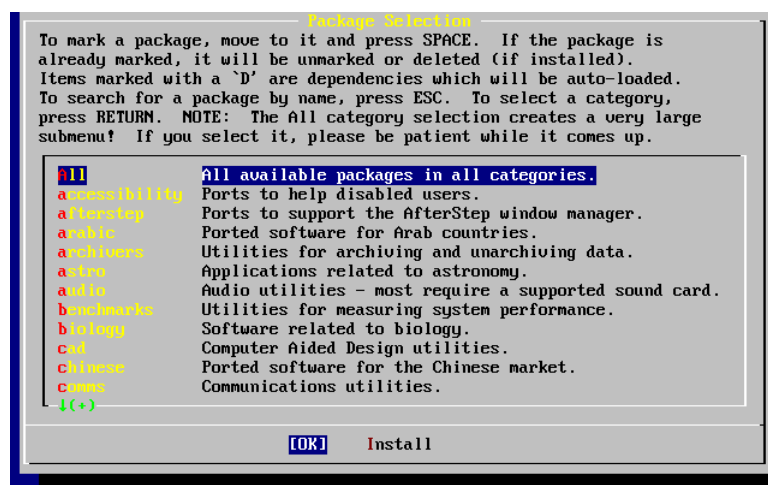


Figura 20: Seleção de pacotes

Na categoria **shells** ou através da categoria **All** encontramos o pacote do programa bash, vamos marcar para instalação pressionando a barra de espaço e sair selecionando **Ok**, o pacote

está pronto para ser instalado:

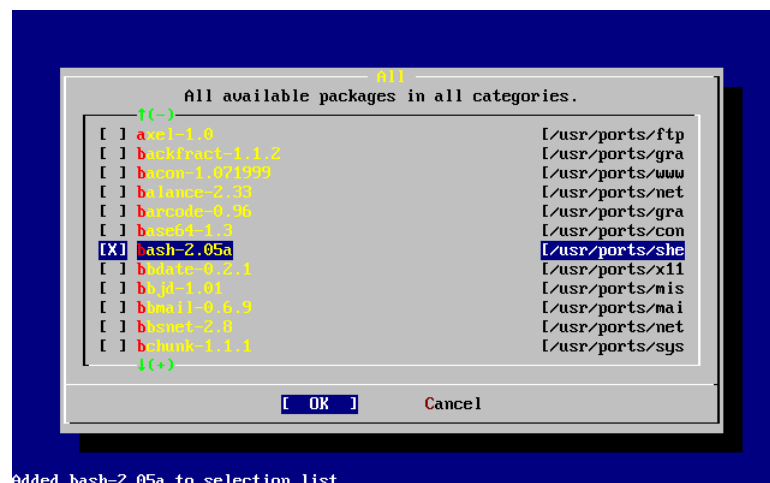


Figura 21: Selecionando o pacote bash para instalação

De volta ao menu anterior selecionamos **Install** e em seguida **Ok** para confirmar e proceder com a instalação dos pacotes.

## 6.7 Usuários e grupos

O próximo passo é adicionar usuários e grupos ao sistema. O menu do configurador apresenta as opções onde podemos inclusive definir o shell do usuário (como o bash que foi instalado anteriormente). Lembrando que o caminho para o bash é diferente para os usuários acostumados com o Linux e no **FreeBSD** se encontra em **/usr/local/bin/ bash**:



Figura 22: Adicionando usuários e grupos



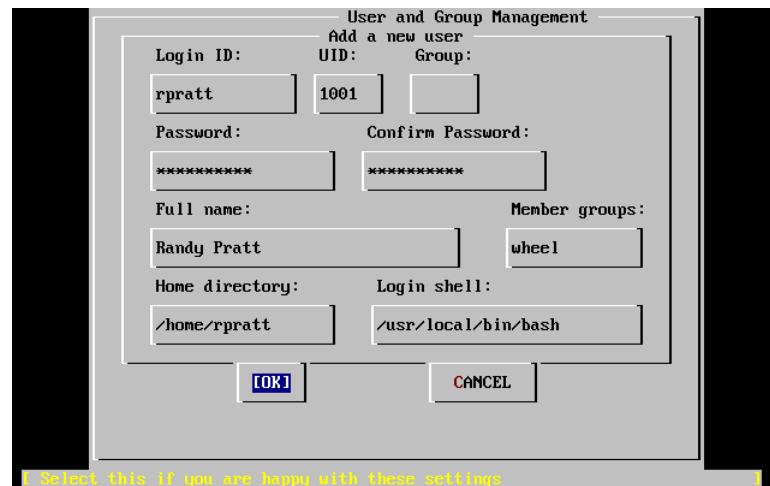


Figura 23: Configurando novo usuário

Terminada criação dos usuários e grupos selecione **X Exit** no menu principal, após selecionar **Ok** o configurador solicita que seja definida uma senha para o usuário *root*. Basta entrar com a senha e confirmá-la em seguida.

## 6.8 Finalizando

Para finalizar o processo de instalação (que é muito mais rápido de executar do que escrever) selecione **No** na janela de confirmação que pergunta: **Visit the general configuration menu for a chance to set any last options?**

Essas configurações dizem respeito a serviços, interfaces de rede e outros, é possível retornar a elas através do **sysinstall** a qualquer momento (na opção **C Configure**).

De volta a tela principal do **sysinstall** tudo se encerra selecionando **X Exit Install**, para que o sistema reinicie. Lembre de remover quaisquer dispositivos que possam atrapalhar a inicialização.

## 7 Primeiro login e algumas configurações

### 7.1 Shell e variáveis de ambiente

O shell padrão do **FreeBSD** é o **tsch** (Tenex C Shell, uma versão melhorada do C Shell) e durante o processo de instalação o pacote do **bash** (Bourne Again Shell) instalado. O **tsch** é um excelente shell mas caso esteja mais familiarizado com o **bash** pode facilmente alterar o shell padrão do usuário com o comando **chsh**:

```
sargas# chsh -s /usr/local/bin/bash usuario
```

O comando **setenv** é usado para exibir e definir valor para as variáveis de ambiente nos shells csh e tcsh. Por exemplo:

```
sargas# setenv
_=/bin/csh
HOSTTYPE=FreeBSD
EDITOR=ed
...
sargas# setenv EDITOR /usr/bin/vim
```

No bash, o comando para trabalhar com variáveis de ambiente é o **export**:

```
sargas# env
SHELL=/usr/local/bin/bash
TERM=xterm
EDITOR=ed
...
sargas# export EDITOR='/usr/bin/vim'
```

Após definir o shell do usuário pode ser interessante alterar a codificação e idioma que por padrão é C. Esta é uma forma de se alterar para um usuário em específico. Caso o usuário utilize o tcsh, editamos conforme abaixo o arquivo *.login.conf* no seu diretório pessoal (\$HOME):

```
me:\
    : charset=UTF-8;\
    : lang=en_US.UTF-8;
```

Para alterar em um usuário cujo shell seja o bash, também em seu diretório pessoal editamos o arquivo *.bashrc*:

```
export LANG='en_US.UTF-8';
export MM_CHARSET='UTF-8';
```

O **FreeBSD** suporta muitas codificações (como ISO-8859) e linguagens (como português, pt\_BR), mas infelizmente o console não suporta UTF-8 podendo ser substituído por ISO-8859 por exemplo. Particularmente não tenho problemas quanto a isso e utilizo mesmo assim UTF-8, já que quando acessamos o ambiente gráfico temos suporte a UTF-8, caso a máquina seja um servidor pode ser interessante configurar a codificação como ISO-8859. O capítulo sobre internacionalização do **FreeBSD Handbook**<sup>1</sup> é muito esclarecedor.

Editando os arquivos *.cshrc* e *.profile* para o csh ou *.bashrc*, *.bash\_profile* e *.bash\_logout* para o bash definimos outras variáveis de ambiente e comportamentos que serão configurados no momento que o usuário efetuar login no sistema. Consulte a documentação do seu shell favorito para maiores detalhes.

## 7.2 O arquivo /etc/rc.conf

O arquivo */etc/rc.conf* armazena configurações importantes para o sistema que serão lidas durante a inicialização. Abaixo um exemplo do arquivo */etc/rc.conf* que define os parâmetros de rede e alguns serviços:

```

hostname="sargas.scorpius"
# Mapa de teclado para o padrao americano
keymap="us.iso"
# Descanso de tela
5 saver="daemon"
# Rede usando DHCP
ifconfig_re0="DHCP"
# Rede com configuracoes manuais
# ifconfig_re0="inet 192.168.0.1 netmask 255.255.255.0"
10 # defaultrouter="192.168.0.254"
# Compatibilidade com Linux
linux_enable="YES"
# Mouse no terminal
moused_enable="YES"
15 # Servico ssh
sshd_enable="YES"

```

### 7.3 O arquivo /etc/sysctl.conf

Através do arquivo /etc/sysctl.conf definimos valores para o kernel, estes ajustes exigem conhecimentos avançados e geralmente são feitos objetivando melhor performance no sistema, para um sistema de gerenciamento de banco de dados por exemplo. O capítulo 11.11 do [FreeBSD Handbook](#)<sup>1</sup> trata sobre este assunto. O exemplo abaixo demonstra o seu uso para configuração da utilização de memória na comunicação entre processos (IPC):

```

kern.ipc.shmall=420186
kern.ipc.shmmax=1721083904

```

### 7.4 O arquivo /etc/ttys

Editando o arquivo /etc/ttys podemos definir o número de terminais virtuais, trocando o valor da coluna **status** de **on** para **off**. O exemplo abaixo deixa como ativos somente 4 terminais virtuais:

```

5 ttyv1  "/usr/libexec/getty Pc"      cons25  on  secure
  ttyv2  "/usr/libexec/getty Pc"      cons25  on  secure
  ttyv3  "/usr/libexec/getty Pc"      cons25  on  secure
  ttyv4  "/usr/libexec/getty Pc"      cons25  on  secure
  ttyv5  "/usr/libexec/getty Pc"      cons25  off secure
  ttyv6  "/usr/libexec/getty Pc"      cons25  off secure
  ttyv7  "/usr/libexec/getty Pc"      cons25  off secure
  ttyv8  "/usr/local/bin/xm -nodaemon" xterm   off secure

```

### 7.5 O sistema de arquivos /proc

O sistema de arquivos /proc não vem montado por padrão, para montá-lo adicione a seguinte linha no arquivo /etc/fstab:

```

proc    /proc    procfs    rw    0 0

```

## 8 Suporte a hardware

### 8.1 Módulos no kernel

Assim como no Linux o suporte a hardware no **FreeBSD** pode ser através da carga de módulo no kernel. O comando **kldstat** exibe os módulos carregados atualmente enquanto o comando **kldload módulo** permite a carga de um módulo. Abaixo um exemplo do comando **kldstat**:

sargas# kldstat					
Id	Refs	Address	Size	Name	
1	19	0xffffffff80100000	d188c0	kernel	
2	1	0xffffffff80e19000	23ed0	snd_hda.ko	
3	2	0xffffffff80e3d000	75708	sound.ko	
4	1	0xffffffff80eb3000	d44ab0	nvidia.ko	
5	2	0xffffffff81bf8000	413b0	linux.ko	
6	1	0xffffffff81e22000	11fe	daemon_saver.ko	

O módulo **snd\_hda.ko** suporta placas de som de chipset ICH7 enquanto o módulo **nvidia.ko** dá suporte a placas de vídeo nVidia. Para listar o hardware PCI usamos o comando **pciconf -lv** e a partir destas informações podemos buscar qual módulo suporta determinado hardware.

## 9 Instalação e manutenção de programas

A manutenção do sistema é uma das tarefas mais importantes. É necessário garantir que os programas instalados estejam sempre atualizados reduzindo assim a possibilidade de problemas por conta de vulnerabilidades e outras falhas. A tarefa de instalar programas e manter o sistema atualizado é muito facilitada no **FreeBSD** através do sistema de **packages** e da árvore de **ports**.

### 9.1 Atualização do ports

Começamos pela atualização do sistema instalado, para garantir que todo programa que esteja em sua última versão disponível.

Uma das formas de atualizar a árvore de **ports** é através do programa **portsnap**, sempre executada pelo usuário root:

```
sargas# portsnap fetch extract
```

Agora temos em **/usr/ports/** toda estrutura da árvore atualizada, antes de continuar consulte o arquivo **/usr/ports/UPDATING**, faça isso SEMPRE que atualizar a árvore de **ports**. O programa **portaudit** mantém uma base de auditoria que permite saber se algum dos programas instalados no sistema possui alguma vulnerabilidade conhecida, recomendando sua atualização.

Para instalar o **portaudit** (e qualquer outro programa) precisamos saber onde ele se encontra (dentro da árvore de ports). Em seguida, acessamos o diretório em questão e prosseguimos com a instalação:

```
sargas# whereis portsaudit
portaudit: /usr/ports/ports-mgmt/portaudit
sargas# cd /usr/ports/ports-mgmt/portaudit
sargas# make install clean
```

Concluída a instalação, podemos executar uma checagem nos programas instalados no sistema:

```
sargas# portaudit -Fa
auditfile.tbz                100% of   68 kB   52 kBps
New database installed.
0 problem(s) in your installed packages found.
```

E para checar os programas instalados estão atualizados em relação às últimas versões disponíveis na árvore de ports (importante sempre executar antes a atualização da árvore como demonstrado anteriormente com **portsnap fetch extract**):

```
sargas# pkg_version -v
portaudit-0.5.17             =   up-to-date   with port
```

## 9.2 Manutenção de programas

Estando com a árvore de **ports** atualizada é possível passar à manutenção de programas procedendo com a instalação, atualização e remoção.

A instalação de programas é uma tarefa simples e pode ser feita basicamente de duas formas: pacotes (packages, que são programas pré compilados) ou através da árvore de ports (código fonte a que será compilado na máquina). Através dos pacotes ganhamos agilidade e através dos ports ganhamos poder de customização. Importante lembrar que o **portaudit** deve sempre ser atualizado antes da instalação de qualquer programa.

Para instalar um programa através do pacote executamos como root:

```
sargas# pkg_add -r nome_do_programa
```

A opção **-r** indica que o sistema fará a carga do pacote do repositório na internet, útil pois não é necessário ter o arquivo. Para ver detalhes dos programas instalados ou de algum em particular:

```
sargas# pkg_info
sargas# pkg_info portaudit-0.5.17
Information for portaudit-0.5.17:

5 Comment:
Checks installed ports against a list of security vulnerabilities

Description:
10 portaudit provides a system to check if installed ports are listed in a
database of published security vulnerabilities.

After installation it will update this security database automatically and
include its reports in the output of the daily security run.
15 If you have found a vulnerability not listed in the database, please
contact
the FreeBSD Security Officer <security-officer@FreeBSD.org>. Refer to
```

```
20 http://www.freebsd.org/security/#sec
for more information.
WWW: http://people.freebsd.org/~eik/portaudit/
25 Oliver Eikemeier <eik@FreeBSD.org>
```

Para saber todos os arquivos que um determinado **port** instalou usamos a opção **-L**:

```
5 sargas# pkg_info -L portupgrade -2.4.8_1,2
Information for portupgrade -2.4.8_1,2:

Files:
/usr/local/man/man1/pkg_deinstall.1.gz
/usr/local/man/man1/pkg_fetch.1.gz
/usr/local/man/man1/pkg_glob.1.gz
...
```

Com **pkg\_info -D nome\_do\_programa** é possível reler as mensagens exibidas logo após sua instalação. Para consultar as informações de um port sem seu nome ou versão usamos a opção **-x**:

```
5 sargas# pkg_info -L -x portupgrade
Information for portupgrade -2.4.8_1,2:

Files:
/usr/local/man/man1/pkg_deinstall.1.gz
/usr/local/man/man1/pkg_fetch.1.gz
/usr/local/man/man1/pkg_glob.1.gz
...
```

Por exigir compilação, a instalação de programas através de ports é um pouco mais demorada mas igualmente simples. Todo processo é automatizado com todas as dependências sendo resolvidas por ele. Por exemplo, para a instalação do programa **portupgrade**:

```
sargas# whereis portupgrade
sargas# /usr/ports/ports-mgmt/portupgrade
sargas# cd /usr/ports/ports-mgmt/portupgrade
sargas# make install clean
```

O **portupgrade** é um dos programas que permitem a atualização de programas instalados no sistema com base na árvore de **ports**. Após verificar que algum programa não se encontra atualizado com o comando **pkg\_version -v**, podemos atualizá-lo:

```
sargas# portupgrade -R nome_do_programa
```

Onde a opção **-R** faz com que sejam atualizados primeiramente os programas que são dependências. Caso deseje que a atualização não seja feita com base na árvore de **ports** e sim através de pacotes existentes em um CD-ROM por exemplo, pode-se utilizar as opções **-PP** no **portupgrade**. A opção **-P** somente faz o **portupgrade** buscar pelo pacote localmente, caso não encontre ele usará a árvore de **ports**, porém com a opção **-PP** ele não busca no **ports**. Finalmente, é possível atualizar também todos os programas instalados:

```
sargas# portupgrade -a
```

Eventualmente alguma atualização automática pode não se comportar conforme esperado, nesses casos, com muita segurança e atenção nas dependências existentes entre os programas, é possível manualmente proceder com a remoção e instalação do programa usando a árvore de **ports**:

```
sargas# cd /usr/ports/ports-mgmt/portupgrade
sargas# make deinstall
sargas# make install clean
```

É necessário cuidado com esse tipo de procedimento para não gerar inconsistências na base de programas instalados.

Além da remoção manual de um programa (como demonstrado acima), é possível (e mais seguro), utilizar as ferramentas **pkg\_delete** e **pkg\_deinstall**, passando como argumento nome do programa. A diferença básica entre eles é que o **pkg\_deinstall** busca as dependências do programa e também as remove.

Após todos estes processos temos a possibilidade de limpar arquivos utilizados (diretório **work** do **port** e arquivos que não possuem referência no **/usr/ports/distfiles**), economizando espaço em disco:

```
sargas# portsclean -C
sargas# portsclean -D
sargas# portsclean -DD
```

Onde a opção **-C** limpa o diretório **work**, **-D** limpa o diretório **/usr/ports/distfiles**. A forma **portsclean -CDi** é a mais indicada pois trabalha de forma interativa.

A ferramenta **pkg\_cutleaves** encontra pacotes que não são referenciados por programas e permite que você os remova (através do **pkg\_delete**). Sempre atualize a base de pacotes antes:

```
sargas# pkgdb -u
sargas# pkg_cutleaves -l
```

A opção **-F** do programa **pkgdb** lhe permite corrigir determinadas dependências.

## 10 Estações de trabalho

Todo processo visto até agora pode ser aplicado em servidores e também estações de trabalho. Veremos especificamente a instalação de programas voltados para os usuários em estações de trabalho.

### 10.1 Ambiente gráfico básico

Para um ambiente gráfico básico (servidor X, gerenciador de janelas e navegador), podemos proceder com a instalação através de pacotes, para economizar tempo de compilação com **ports**:

```
sargas# pkg_add -r xorg-minimal
sargas# pkg_add -r dbus
sargas# pkg_add -r hal
sargas# pkg_add -r gdm
```

```
5 sargas# pkg_add -r gnome2-lite
sargas# pkg_add -r gnome-screensaver
sargas# pkg_add -r firefox
```

Após a conclusão da instalação, precisamos configurar os serviços que vão iniciar automaticamente, editando o arquivo **/etc/rc.conf**:

```
hald_enable="YES"
dbus_enable="YES"
gdm_enable="YES"
```

Por último, executar o comando **Xorg -configure** para que seja gerada uma configuração inicial do **xorg.conf**, este arquivo deve ser movido para **/etc/X11/**. Caso deseje testar o arquivo de configuração gerado, execute:

```
sargas# Xorg -config xorg.conf.new -retro
```

Todo processo de ambiente gráfico é apresentado em detalhes no **FreeBSD Handbook**<sup>4</sup>, capítulo 5.4 - X11 Configuration.

## 11 Conclusão

O processo de instalação do **FreeBSD** é muito bem documentado e o propósito deste tutorial foi apresentar uma introdução ao assunto de forma a divulgar o sistema, incentivando novos usuários a experimentá-lo, principalmente aqueles já familiarizados com sistemas Linux. Ambientes específicos requerem maior experiência para configurações específicas e os sistemas derivados do **FreeBSD** apresentados são ótimas opções, alguns com maior foco no usuário e portanto mais amigáveis.

## 12 Informações adicionais

### 12.1 Referências

1. **FreeBSD Handbook**. Disponível em [http://www.freebsd.org/doc/en\\_US.ISO8859-1/books/handbook/](http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/). Acesso em 31 mai. 2011.
2. KENNAWAY, Kris. The **FreeBSD** Project: Introducing **FreeBSD** 7.0.. Disponível em <http://people.freebsd.org/~kris/scaling/>. Acesso em 21 jul. 2010.
3. Getting **FreeBSD**. Disponível em <http://www.freebsd.org/where.html>. Acesso em 21 jul. 2010.
4. **FreeBSD Handbook - X11 Configuration**. Disponível em [http://www.freebsd.org/doc/en\\_US.ISO8859-1/books/handbook/x-config.html](http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/x-config.html). Acesso em 31 mai. 2011.
5. POHLMANN, Frank. Why **FreeBSD** - A quick tour of the BSD alternative. Disponível em <http://www.ibm.com/developerworks/opensource/library/os-freebsd/>. Acesso em 21 jul. 2010.



## 12.2 Licença de uso

Creative Commons alguns direitos reservados: Atribuição - Uso Não Comercial - Compartilhamento pela mesma Licença (by-nc-sa).

*É permitida adaptação, criação de obras derivadas, tradução e livre distribuição desta obra com fins não comerciais, contanto que atribuam o crédito e licenciem as novas criações sob os mesmos parâmetros. Toda nova obra feita com base nesta deverá ser licenciada com a mesma licença, de modo que qualquer obra derivada, por natureza, não poderá ser usada para fins comerciais.*