
GNU/Linux
debian

ÍNDICE

O DEBIAN

O Debian é um sistema operacional (SO) livre para seu computador. Um sistema operacional é um conjunto de programas básicos e utilitários que fazem seu computador funcionar. O Debian usa o Kernel (núcleo de um sistema operacional), Linux, mas a maior parte das ferramentas do SO vêm do projeto GNU; daí o nome Debian GNU/Linux.

O Debian GNU/Linux é mais que um simples SO: ele vem com mais de 15490 pacotes contendo softwares pré-compilados e distribuídos em um bom formato, que torna fácil a instalação deles na sua máquina.

O Debian foi iniciado em agosto de 1993 por Ian Murdock, como uma nova distribuição que seria feita abertamente, no espírito do Linux e do projeto GNU.

Ele começou como um grupo pequeno de desenvolvedores de Software Livre e cresceu gradualmente para se tornar uma comunidade grande e bem-organizada de desenvolvedores e usuários.

O nome Debian vem do nome de seu criador, Ian Murdock, e sua esposa, Debra.

Observem a seguinte linha do tempo para identificar o surgimento do Debian:

• 1969: Desenvolvimento do Unix, inicialmente por Kenneth Thompson e Dennis Ritchie.

• 1974: Ampla divulgação do Unix e de outros softwares com disponibilização do código.

• 1978: Os novos Unix param de vir com seu código disponibilizado.

• 1984: Insatisfeito com o fim da era do código aberto, Richard Stallman demite-se do MIT (Massachusetts Institute of Technology) e decide fazer um SO compatível com o Unix e totalmente aberto (projeto GNU).

• 1984: Richard Stallman introduz o conceito de free software “ou” software livre ““.

• 1985: Richard Stallman cria a Free Software Foundation (FSF) para arrecadar fun dos para o projeto GNU.

• 1985: A FSF cria a licença GNU GPL, que especifica se um software é livre ou não.

• 1986: Andre Tanenbaun, um professor universitário da Holanda, desenvolve o Minix, um SO para fins de estudos acadêmicos.

• 1991: O projeto GNU está avançado, mas não possui um Kernel.

• 1991: Linus Torvals, um universitário finlandês, estuda o Minix e desenvolve, como projeto pessoal, um Kernel de SO, compatível com o Unix.

• 1991: Linus testa o seu Kernel com programas da FSF.

• 1992: Theodore Tsó, um jovem programador C e usuário de Unix, junta o Kernel Linux e vários programas da FSF em disquetes e vende numa lista de discussão por US$ 2,50. Essa foi a primeira distribuição Linux.

• 1993: Surgem as distribuições Slakware e Debian.

Debian Linux (Stable releases)		
Versão	**Nome**	**Data**
0.93R6	-	26 de outubro de 1995
1.1	Buzz	17 de Junho de 1996
1.2	Rex	12 de Dezembro de 1996
1.3	Bo	5 de Junho de 1997
2.0	Hamm	24 de Julho de 1998
2.1	Slink	9 de Março de 1999
2.2	Potato	15 de Agosto de 2000
3.0	Woody	19 de Julho de 2002
3.1	Sarge	6 de Junho de 2005
?	Etch	-

DISTRIBUIÇÕES

Quando ocorre a junção do kernel Linux a programas criados por empresas, universidades e programadores independentes, além da FSF, temos uma distribuição. As maiores e mais antigas distribuições ainda em produção são: Slackware, Debian, Suse e Red Hat. Muitas distribuições são derivadas dessas.

VERSÕES DO DEBIAN

O Debian tem sempre três versões em manutenção constante: “stable”, “testing” e “unstable”.

STABLE

A distribuição “stable” contém a última distribuição oficialmente lançada pela equipe Debian. Essa é a versão de produção do Debian, ela que é recomendada primeiramente.

A distribuição “stable” do Debian GNU/Linux está atualmente na versão 3.1r0 e seu codinome é *sarge*. Ela foi lançada em 6 de Junho de 2005.

TESTING

A distribuição “testing” contém pacotes que não foram aceitos numa versão “stable” ainda, mas eles já estão na fila para serem aceitos. A principal vantagem de usar essa distribuição é que ela tem versões mais novas dos softwares.

A distribuição “testing” atual chama-se *etch*.

UNSTABLE

É na distribuição “unstable” que o desenvolvimento ininterrupto do Debian ocorre. Geralmente, os usuários dessa distribuição são os próprios desenvolvedores e pessoas que gostam de emoções fortes.
A distribuição “unstable” atual chama-se *sid*.

IMAGENS ISO

As imagens ISO, além de classificadas em stable, testing e unstable, poderão ser do tipo:

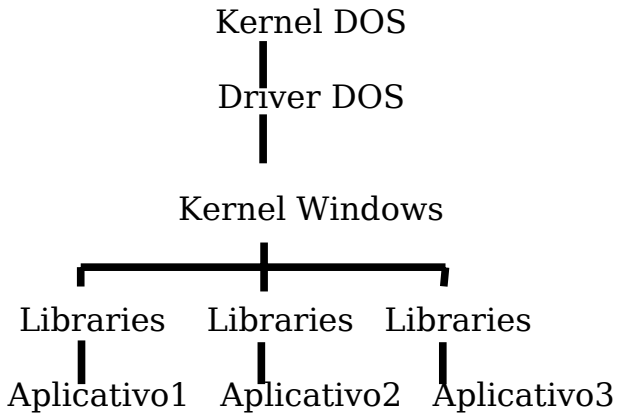
- DVD: imagem em DVD contendo toda a distribição
- CD inteiro: vários CDs contendo todo o Debian.
- Netinst: um único CD, com cerca de 100 MB, contendo a base do sistema. o restante do sistema será instalado via internet. Essa modalidade tambem estará disponivel para pendrive (40 MB) ou disquete (3 disquetes).

ARQUITETURAS SUPORTADAS

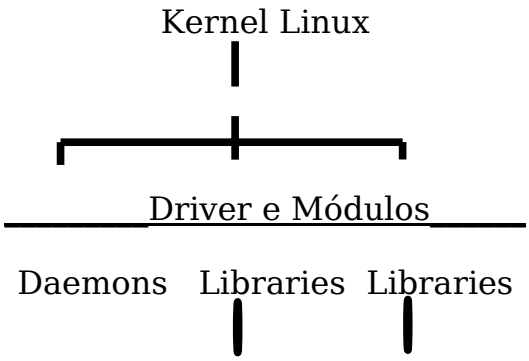
As seguintes arquiteturas de computadores são suportadas nesta versão: Alpha, ARM, HP, PA-RISC, Intel x86, Intel IA-64, Motorola 680x0, MIPS, MIPS (DEC), Powerpc, IBM S/390, SPARC.

ARQUITETURA LINUX X WINDOWS

Arquitetura Windows 95/98



Arquitetura Linux



PREPARAÇÃO PARA INSTALAÇÃO

A instalação do sistema poderá ocorrer sob três situações básicas:

- O HD não possui nenhum sistema operacional (SO) instalado.
- O HD já possui um ou mais SO instalados, mas há um espaço livre, sem partição.

  Nos dois casos acima não haverá problemas para instalação.
- O HD já possui um ou mais SO Instalados e não á espaço livre.

Neste caso deverá ser utilizado um software para reparticionar o HD para criação de um espaço livre no HD.

O Debian irá requerer, no mínimo, 600 MB de espaço em disco para sua instalação (incluindo swap).

- No caso da instalação de dois SO em um mesmo HD, instale primeiro o Windows e depois o Linux. Porque o Linux grava o GRUB no setor de boot do HD (MBR) e o Windows sobrescreve esse valor, apagando-o.

INSTALAÇÃO DO SISTEMA

Inicialmente devemos saber qual versão do Debian iremos instalar, existem três versões disponíveis:

- Stable: Versão estável de um projeto, sem bugs ou folhas de segurança. Esta versão não recebe novos pacotes, somente atualizações de pacotes para sanar falhas de segurança.
- Testing: Contém pacotes que já passaram por um teste inicial da equipe Debian, eles são criticados pelos usuários e ficam em manutenção, até estarem prontos para serem portados pela versão stable.
- Unstable: Contém programas que não foram testados pela equipe Debian.

A instalação do Debian pode ser realizada em 4 (quatro) fases:

1. Baixar imagem ISO que será utilizada;

As 14 imagens do sistema completo está disponível em: http://cdimage.debian.org/debian-cd/3.1_r0a/i386/iso-cd/

No link acima também podemos baixar a imagem do CD com o sistema mínimo, para instalação via rede, chamado de netinst.

A versão Debian-BR-CDD, distribuição baseada no Debian Sarge pode ser baixado em http://cdd.debian-br.org/iso/1.0_pre5/debian-br-cdd_1.0pre5.iso

Obs.: Utilize o wget para baiar as imagens, ou alguma outra ferramenta para dowload evitando assim perda de pacotes.

Lembre-se após fazer o dowload da imagem usar o md5sum para verificar a integridade da imagem.

1. Gravar um CD-ROM a partir da imagem baixada;
2. Instalar o sistema;
3. Realizar os ajustes finais pós-instalação;
4. Realizar o Checklist de segurança pós-instalação.

INSTALAÇÃO 1° FASE

Configure a máquina para, onde o debian será instalada, para dar o boot pelo cd-rom.

- Coloque o cd-rom na máquina.
- Inicialize a maquina.
- Aparecerá a tela inicial da instalação.
- Pressionando as telas de F1 a F7, serão mostradas opções específicas para diversos tipos de inicialização da instalação.
- Tecle <enter> para inicializar a instalação.

FASE	*AÇÃO*	*OBS*
Escolha da linguagem	Escollher "portuguese (brasil)"	
Seleção de layout do teclado	Escolher o teclado correto	“Português ABNT2" ou "Português brasileiro americano"
Detecção cd-rom		
Carrregamento de componentes		
Configuração da rede (1)	Detecção da placa de rede	
Configuração da rede (2)	Configuração da rede	
Configuração da rede (3)	Configurar nome da máquina	
Configuração da rede (4)	Configurar o domínio da máquina	
Detecção de hardware		
Particionamento de disco (1)	Selecionar "editar manualmente a tabela de partições"	
Particionamento de disco (2)	Realizar particionamento de disco	Criar partições do tipo reiserfs e swap com pelo menos 256 mb de tamanho
Particionamento de disco (3)	finalizar particionamento	Ao final selecionar "finalizar particionamento e gravar as mudanças no disco"
Particionamento de disco (4)	Gravar as mudanças em disco.	
Formatação de partições		

FASE	*AÇÃO*	*OBS*
Instalação do sistema básico		
Instalação do GRUB	Instalar o GRUB no registro de inicialização principal.	Selecionar “Sim”. Será detectado outro sistema operacional.
Finalizando a instalação (1)	Retirar o CD_ROM e selecionar continuar	
Finalizando a instalação (2)	Reboot	

- Haverá um reboot do sistema. Após esse reboot, o sistema seguirá para a segunda parte da instalação.

INSTALAÇÃO PROPRIAMENTE DITA - 2° FASE

FASE	*AÇÃO*	*OBS*
Tela de Boas vindas	Digite Enter	
Configuração do Fuso Horário (1)	Selecione o tipo de ajuste	O relógio de hardware esta configurado para GMT? SIM: permitirá a escolha por diferença de fuso horário global. NÃO: permitirá a escolha por região do mundo (no caso, o Brasil).
Configuração do Fuso Horário (2)	Selecionar o fuso correto, caso deseje alterar o fuso mais tarde, bastará utilizar os comandos tzsetup, tzconfig ou tzselect.	Brasília corresponde a LESTE ou GMT +3
Configuração da senha de root	Entrar com a senha	será pedido que entre com a senha novamente para confirmá-la
Criação de conta de usuário comum	não fazê-lo. selecionar cancela	
Configurar o apt	Selecione a opção “Configurar o apt”	
Configuracao do apt (1)	Selecionar “HTTP” como método de acesso ao repositório	
Configuracao do apt (2)	Selecionar um país como fonte do apt	A melhor opção será “Estados Unidos”, onde estão os repositórios mais rápidos e estáveis.
Configuracao do apt (3)	Selecione o mirror a ser utilizado	um dos mais rápidos é o “ftp.us.debian.org”. Após a seleção, o Debian fará um download da relação de programas lá existentes.
Configuracao do apt (4)	Indicar o Proxy da rede, caso haja um.	Se não houver Proxy ou se o mesmo for transparente, bastará um ENTER. No caso de existir um Proxy, utilize a sintaxe mostrada na tela para defini-lo.
Configuracao do apt (5)	Aguarde o download do mirror	o Debian fará um download da relação de programas existentes no mirror selecionado.

FASE	*AÇÃO*	*OBS*
Configuracao do apt (6)	Adicionar outra fonte apt?	Não é necessário adicionar outra fonte. Caso deseje adicionar qualquer outra fonte mais tarde bastará utilizar o comando apt-setup
Configuracao do apt (7)	Adicionar a fonte de atualização de segurança	Responda SIM para a pergunta “Utilizar atualizações de segurança de security.debian.org”
Configuracao do apt (8)	Aguardar o download do índice do servidor de segurança	O Debian fará um download da relação de atualizações de segurança existentes no mirror selecionado. Depois disso cairemos no menu de instalação novamente.
Finalizar a configuração do sistema básico	Selecionar a opção	Com isso, encerra-se a instalação do sistema. poderemos voltar ao menu de instalação, se for o caso, com o comando base-config.

AJUSTES PÓS-INSTALAÇÃO

CONFIGURAÇÃO DO APT

A configuração do APT se dá com base em dois arquivos:

- /etc/apt/apt.conf: contém as configurações globais do APT. A mais importante é a configuração do Proxy.
- /etc/apt/sources.list: contém a relação dos repositórios, locais ou remotos. Será usado apenas se for usado um proxy para acesso à internet.

Obs.: Caso não tenha sido criado o arquivo o arquivo /etc/apt.conf durante a instalação do sistema, poderemos criá-lo com o comando base-config – configuração do APT.

Para inserir um repositório, bastará executar o comando: #apt-setup.

Caso haja necessidade de excluir algum repositório, bastará editar o arquivo /etc/apt/sources.list, e excluir ou comentar com “#” as linhas dos referidos repositórios.

Obs.: O arquivo apt.conf não admite o caractere # para comentários, utilize ' (aspas simples) para comentários neste arquivo.

Não é interessante configurar mais de três repositórios, pois eles são redundantes. Vários repositórios só irá fazer perder tempo.

ATUALIZAÇÃO DO SISTEMA

Logo após a instalação é importante fazer uma atualização do sistema, com os comandos:

apt-get update

apt-get upgrade

Alguns pacotes, contendo programas essenciais para o correto funcionamento do sistema, deverão ser instalados. São eles:

- mc: O mcedit é o editor de texto mais amigável para o ambiente Shell.
- less: filtro que permite a leitura de arquivos longos diretamente na tela.
- apmd: responsável pela gerência de energia do computador. Responsável pelo desligamento para máquinas ATX.
- rcconf: permite escolher quais serviços irão ao ar juntamente com o Linux.

Para instalar esses programas, execute os comandos:
apt-get install mc
apt-get install less
apt-get install apmd
apt-get install rcconf

DEFINIÇÃO DE ALIASES

O alias é um recurso de atalho para simplificar um comando. Tem por objetivo facilitar a administração e aumentar a segurança.

Vamos definir algumas aliases para uma operação segura no sistema. Edite o arquivo /etc/profile e insira as seguintes linhas no final desse arquivo:

alias ls=”ls - - color”
alias df=”df -h”
alias cds=”cd /etc/init.d;ls”

Obs.: Para atualizar as mudanças no arquivo deve ser feito logout e logar-se novamente.

Em algumas versões deverá ser instalado o pacote alias.

ESTABELECIMENTO DE UM MTA

O Debian instala o Exim como MTA (Mail Transfer Agent), servidor responsável por enviar e-mail de usuário ou de outras máquinas). No entanto, o Exim não vem configurado, por default, para o envio de e-mails local. Assim sendo deveremos fazer a instalação de um MTA mais funcional para realizar a tarefa. Utilizaremos para o Sendmail. A instalação deve ser feita com o comando:
#apt-get install sendmail

O Sendmail já vem configurado para enviar mensagens localmente, sem precisar fazer qualquer configuração, a partir de agora o usuário root começará a receber mensagens periódicas do sistema.

SERVIÇOS NA INICIALIZAÇÃO DO SISTEMA

Utilize o comando #rcconf para selecionar os serviços que irão rodar na inicialização do Linux. Inicialmente ficarão habilitados os seguintes serviços: apmd, cron, inetd, klogd, makedev, sysklogd, sendmail.

INICIANDO OU PARANDO DAEMONS COM O COMANDO SERVICE

Para usar o comando service para dar start, stop, reload ou restart em um daemon teremos que copiar o script service de uma distribuição que utilize (conectiva).

Copiar o script service para o diretótio /sbin.

AJUSTE DA RESOLUÇÃO LOCAL DE NOMES

É importante observar o conteúdo do arquivo /etc/hosts, para que ele esteja coerente com a sua rede.

Máquinas que utilizam endereços IPs fixos deverão o arquivo deverá estar da seguinte forma:

127.0.0.1 localhost.localdomain localhost
ip_do_host nome_do_host.domio nome_do_host

Exemplo:

127.0.0.1 localhost.localdomain localhost
ip_do_host micro01.minharede.com.br micro01

CRIAÇÃO DO /etc/init.d/rc.local

Assim como ocorre no MS-DOS, para executar comandos durante a inicialização editamos o arquivo autoexec.bat, podemos fazer o mesmo no Linux.

Para isso, serão utilizados os arquivos /etc/init.d/rc.local e etc/profile. A diferença entre eles é que o rc.local é executado imediatamente após o sistema ir ao ar e, o segundo após o login do usuário.

Por default o Debian não possui um arquivo /etc/init.d/rc.local, este arquivo serve para executar comandos e rotinas logo após o boot do sistema. para criá-lo siga a seqüência:

- Dentro de /etc/init.d execute os comandos abaixo;
- # echo '#!/bin/bash' > rc.local
- # chmod 500 rc.local permissão xr para dono
- # update-rc.d rc.local defaults 99 adiciona o serviço rc.local aos runleavels obedecendo aos valores default e utilizando a prioridade 99 de execução.

EDITOR PADRÃO

Para que o editor de textos padrão do shel seja o mcedit, deveremos adicionar a seguinte linha no fim do arquivo /etc/profile:

export EDITOR=mcedit

ATUALIZAÇÃO DO KERNEL POR APT

Para saber a versão do kernel utilizado utilize o comando:

#cat /etc/proc/version

Para localizar um novo kernel com o apt utilize o comando:

#apt-cache search kernel-image

Ele irá retornar algumas linhas. para instalação de um kernel, deve-se observar os prefixos das imagens:

i386 Processadores Intel 386,486 e compatíveis.

i586tsc	Processadores Intel Pentium MMX, AMD K5, Cyrix 6x86, Cyrix III
i686	Processadores Intel Pentium Pro, P II, P III, P4
k6	Processadores AMD K6, K6 II, K6 III.
k7	Processadores AMD Duron, Athlon, Sempron.
k8	Processadores AMD 64 bits,
smp	Para máquinas multiprocessadas (mais de um processador).

Instale o Kernel com o seguinte comando:

apt-get install <kernel escolhido>

Após a instalação, bastará reiniciar a máquina e o grub mostrará o kernel antigo e o novo kernel.

INSTALAÇÃO EM NOTEBOOKS

Após instalação em um notebook, não deixe de consultar o endereço: http://www.linux-on-laptops.com. Neste site, serão mostrados todos os ajustes finos necessários para cada marca de notebook.

CORES NO VI

Para deixar o editor de textos vi com cores siga os seguintes passos:

- Instale o vim (vi improved)

apt-get install vim

- Edite o arquivo /etc/vim/vimrc e descomente a linha “syntax on retirando o “.

GERENCIAMENTO DE PACOTES NO DEBIAN

A mais conhecida forma de gerenciar pacotes no Debian é o APT (Advanced Package Tool) e o DPKG (Debian PacKaGe). Os comandos descritos a seguir deverão ser dados pelo usuário root.

# apt-get update	Atualiza a lista de pacotes disponíveis
# apt-get upgrade	Atualiza o sistema
# apt-get dist-upgrade	Atualiza o sistema. Faz migração de versão se houver.
# apt-get install pacote	Instala o pacote em questão.
# apt-get source pacote	Faz dowloads das fontes relativos ao pacote.
# apt-get remove pacote	Desinstala o pacote, deixando os arquivos de configuração.
# apt-cache searche expressão	Procura por pacotes que contem a expressão.
# apt-cache show pacote	Mostra, em detalhes, dados sobre o pacote.
# apt-cache depends pacote	Mostra as dependências dos pacotes.
# apt-cache stats	Mostra estatísticas do banco de dados do pacote.
# apt-cache pkgnames	Mostra uma relação de pacotes disponíveis.

GERÊNCIADOR DE BOOT (GRUB)

Os gerenciadores de boot servem para permitir que vários sistemas operacionais coexistam no mesmo computador. O GRUB denomina da seguinte forma os dispositivos IDE:

hd0	HD Master da 1ª IDE (/dev/hda).
hd1	HD Slave da 1ª IDE (/dev/hdb).
hd2	HD Master da 2ª IDE (/dev/hdc).

hd3 HD Slave da 2ª IDE (/dev/hdd).

O arquivo de configuração do GRUB é /boot/grub/menu.lst.
A resolução no ambiente Shell pode ser definida pelo GRUB na linha kernel do arquivo de configuração /boot/grub/menu.lst.

```
kernel          /boot/vmlinux-2.4.27-2-386   root=dev/hda6  ro vga=771
```

O argumento “vga=771” define a resolução a ser adotada no Shell, no caso 771 refere-se a resolução 800x600x256cores. Essa configuração é feita de acordo com a seguinte tabela:

		Resoluções			
		640x480	800x600	1024x768	1280x1024
Cores	256	769	771	773	775
	32.768	784	787	790	793
	65.536	785	788	791	794
	16.777.16	786	789	792	795

Opções de configuração do GRUB:

Opção	***Função***
default x	Refere-se ao bloco que por default, dará boot. Caso o usuário não selecione nenhuma opção dentro do tempo estabelecido pela opção timeout. O x refere-se ao bloco que irá iniciar por default.
timeout x	Estabelece que o usuário terá x segundos para decidir qual sistema operacional dará partida na máquina.
color a/b c/d	Define as cores do menu.
passwd xxx	O GRUB permite a edição de parâmetros de boot diretamente na tela menu durante o boot, Caso esta linha seja acrescida, a senha xxx terá que ser digitada para que a edição seja permitida.
title	Além de iniciar um bloco define o título, referente ao sistema operacional, que será apresentado no menu de boot.
root	Designa o dispositivo e partição que contém o diretório /boot.
kernel	Contém a localização do kernel dentro da partição designada por root.
initrd	Designa o arquivo que contém a imagem a ser inicializada no boot.
boot	Permite a edição do bloco na tela de menu, durante o boot.

A imagem de fundo do GRUB pode ser trocada, para isso a imagem deve ter extensão .xpm, 14 cores no máximo e 640x480 de tamanho. Podemos

converter uma imagem jpg com o comando abaixo:
#convert imagem.jpg -colors 14 -geometry 640x480! imagem.xpm
#gzip -9 imagem.xpm

Edite o arquivo /boot/grub/menu.lst e altere a linha:
splashimage=(hd0,3)/boot/grub/splash.xpm.gz
para:
splashimage=(hd0,3)/boot/grub/imagem.xpm.gz

ALTERNÂNCIA DE FONTES APT

Caso deseje buscar um pacote testing ou unstable, edite o arquivo /etc/apt/sources.list e altere, na fonte, a palavra stable para testing ou unstable (depende do local onde está o pacote). Rode o comando:

#apt-get update
#apt-get install pacote

Retorne o arquivo /etc/apt/sources.list a situação anterior e rode o comando:
#apt-get update
Obs.: Não rode o comando #apt-get upgrade para não danificar o sistema.

Para transformar uma versão testing em stable, altere, em etc/apt/sources.list, todos os testing para stable e rode a seqüência de comandos:

apt-get update
apt-get dist-upgrade

Finalmente, para atualizar, de stable para stable, quando uma nova versão for lançada, rode os comandos:

apt-get update
apt-get dist upgrade

BUSCA DE PACOTES APT INDEPENDENTES

Para busca de pacotes APT não oficiais, produções independentes, gerados a partir de um repositório não oficial, é possível fazer busca em http://apt-get.org.

DESCOBRINDO EM QUAL PACOTE APT ESTÁ UM DETERMINADO ARQUIVO

É possível descobrir em qual pacote APT um arquivo específico foi inserido. para isso bastará fazer uma pesquisa em http://www.debian.org/distrib/packages. o formulário com atenção para selecionar as opções corretas.

DPKG (Debian PacKaGe)

Para desinstalação completa de pacotes deve ser usado o comando dpkg. A seguir são mostrados alguns comandos:

dpkg -P pacote Desinstala o pacote e todos os arquivos de configuração.
dpkg -L pacote Mostra todos os arquivos instalados pelo pacote e a localização dos mesmos.
dpkg -l Mostra todos os pacotes instalados e desinstalados.
dpkg -i pacote Instala um pacote .deb
dpkg -reconfigure pacote Reconfigura um pacote que foi configurado durante a instalação.

A reconfiguração de um pacote pode ser feito em 4 níveis

- critical
- high
- medium
- low

O DPKG considera esses níveis para fazer perguntas durante o processo. Em critical ele tentará não fazer perguntas. Em low, ele perguntará tudo o que puder. Geralmente medium será uma boa escolha.
#dpkg-reconfigure -p *nível pacote*

Para a desinstalação completa de um pacote sem que ocorra a desinstalação de pacotes dependentes, aplicar a seqüência:
apt-get remove pacote
dpkg -P pacote

AMBIENTE SHELL

O ambiente shell é a famosa tela preta. Por default, o Debian e a maioria das distribuições utiliza o Bash (Bourne-Again Shell).

REALIZANDO LOGIN NO SHELL

Na tela do shell ao inicializarmos o sistema aparecerrá a distribuição, a sua versão, o nome atribuido à maquina e o terminal no qual operamos. Na linha seguinte, será mostrado o pedido de login.

No prompt de login, digite root ou qualquer outro usuário cadastrado e digite a sua respectiva senha.

O prompt do sistema poderá terminar com um caracter cerquilha “ # ” ou com um cifrão “ $ ”. A cerquilha só é utilizada com o root e o cifrão com qualquer outro usuário.
Para desfazer o login, digite # logout ou exit.

EDIÇÃO DE TEXTOS NO SHELL

Existem vários editores de texto para Linux em modo shell. Dentre eles destacam-se o vi, e o mcedit.

OPERAÇÃO BÁSICA COM O VI

Para criar um arquivo e iniciar sua edição, no prompt do sistema, digite:
#vi nome_do_arquivo

COMANDOS BÁSICOS COM O VI

i Entra no modo de edição

Esc	Sai do modo de edição
Esc + dd	Remove a linha inteira
/palavra	Procura pela palavra
Esc + :q!	Retorna para o shell sem salvar
Esc + :wq!	Salva e retorna para o shell
Esc + x!	Salva e força a saída

OPERAÇÃO BÁSICA COM O MCEDIT

Para criar um arquivo e iniciar sua edição, no prompt do sistema, digite:
#mcedit nome_do_arquivo

É importante observar quando editar um arquivo com o mcedit, inserir um <Enter> no final do arquivo para caracterizar o final do arquivo e evitar erros de configuração ou funcionamento no sistema.

OPERAÇÃO MULTIUSUÁRIO

O linux é um sistema operacional multiusuário, pois permite que vários usuários diferentes o utilizem ao mesmo tempo. Essa utilização poderá ser local ou remota. Será local quando os usuários estiverem diretamente conectados à maquina. Será remoto quando os usuários estiverem operando em rede.

Os terminais locais podem ser acessados com as seguintes combinações de teclas:
ALT + F1, F2, F3, F4, F5, F6, F7 (terminal gráfico).

BARRA COMUM E BARRA INVERTIDA

A barra invertida (\) é pouco utilizada, utiliza-se para indicar que um comando irá continuar na próxima linha, e para proteger um caractere que possui significado especial. Veja o exemplo abaixo:

```
# free \
-m
```

CARACTERES MAIÚSCULOS E MINÚSCULOS

O linux é sensitivo, ou seja, diferencia caracteres maiúsculos de minúsculos, por isso ha necessidade de tomar cuidado na hora de digitar um comando ou arquivo.

NOMES DE ARQUIVOS E DIRETÓRIOS

Ao contrário do Windows, os arquivos e diretórios do Linux não têm um formato específico, não existe as extensões .doc, .exe para definir o tipo de um arquivo.

Apenas alguns arquivos gerados pelo OpenOffice terão extensões outros não. Abaixo temos algumas extensões usadas no Linux.

- **txt** - O .txt indica que o conteúdo é um arquivo texto.
- script**.sh** - Arquivo de Script (interpretado por /bin/sh).
- system**.log** - Registro de algum programa no sistema.
- arquivo**.gz** - Arquivo compactado pelo utilitário gzip.
- index**.html** - Página de Internet (formato Hypertexto).

No Linux, os arquivos e diretório iniciados com um ponto ”.” serão ocultos.

Ex: .profile.

DIRETÓRIOS BÁSICOS DO SISTEMA

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

DIRETÓRIO	***CONTEÚDO***
boot	Contém o Kernel e os arquivos que controlam a inicialização do sistema.
bin	Arquivos executáveis (binários) que podem ser acessados por qualquer usuário.
sbin	Similar ao /bin, no entanto são destinados á administração e manutenção do sistema. Alguns só podem ser executados pelo root.
root	Diretório local do super usuário (root).
home	Diretórios locais (home) dos usuários.
usr	Guarda dados compartilhados, antigo /home do Unix.
proc	Diretório virtual, não existe no HD, apenas no kernel. Permite acessar informações sobre a máquina e sobre os processos.
dev	Contém arquivos que servem de ligação com os dispositivos de hardware.
etc	Arquivos de configuração do sistema da máquina local com arquivos diversos para a administração do sistema.
lib	Contém os módulos do Kernel. Arquivos das bibliotecas compartilhadas usados com freqüência.
tmp	Arquivos temporários gerados por alguns utilitários.
mnt	Ponto de montagem de partição temporária.
var	Informação variável, como logs, spool de impressoras, caixas postais em servidores de e-mails.
/media	Trata-se de um ponto de montagem de mídias removíveis.

CADASTRAMENTO DE USUÁRIO

Para cadastrarmos um usuário, utilizamos o comando adduser. Com este comando será criado automaticamente um diretório para o mesmo em /home, esse diretório irá conter as configurações do usuário.

adduser teste - Este comando já solicita a senha e cria o diretório home do usuário, cria os arquivos /etc/passwd, /etc/shadow e /etc/group.

Digite a senha do usuário e confirme a mesma.

useradd teste - similar ao comando adduser, porém não pede a senha (tem que implementar com o comando passwd teste, e não cria o diretório home do usuário.

ALTERAÇÃO DE SENHAS DE USUÁRIOS

Para alterar a senha de qualquer usuário, inclusive a do root, bastará digitar:

passwd login_do_usuário
passwd login_do_usuário -l Bloqueia um usuário.
passwd -u login_do usuário Desbloqueia um usuário.

REMOÇÃO DE USUÁRIOS

Para remover um usuário do sistema há duas opções:

userdel login_ do_usuário remove o usuário mas mantém seus diretórios.
#userdel -r login_ do_usuário remove completamente o usuário, sem deixar rastos.

RETIRANDO O SHELL DO USUÁRIO

Essa medida vai fazer com que o usuário utilize apenas serviços como mail e páginas, mas não possa logar no servidor utilizando serviços que precisem de um shell (telnet, ftp, login local, ssh, etc).

#chsh login_do_usuário -s /

Para devolver o shell ao usuário, digite:

#chsh login_do_usuário -s /bin/bash

COMANDOS BÁSICOS DO SISTEMA

LIGAR DESLIGAR REINICIAR

#shutdown -h now	Desligar o Linux
#halt -p	Desligar o Linux
#init 0	Desligar o Linux
#shutdown -r now	Reiniciar o Linux
#reboot	Reiniciar o Linux
#init 6	Reiniciar o Linux
Ctr + Alt + Del	Reiniciar o Linux

COMANDOS DE AJUDA

man

Comando de ajuda mais utilizado nos Unix e nos Linux.

#man <expressão>

info

Similar ao man.

whatis

Mostra resumidamente para que serve um determinado comando.

whatis ls

apropos

Equivale ao man -k <expressão>

ALIASES

alias

Gera um atalho para um comando.

alias Mostra os alias já definidos.

#mount -t vfat /dev/fd0 /mnt/floppy Podemos simplificar este comando com.

#alias floppy=' mount -t vfat /dev/fd0 /mnt/floppy' Agora basta digitar floppy

#floppy

unalias

desfaz um alias.

EXECUÇÃO EM SEGUNDO PLANO

&

Faz com que um processo rode em segundo plano sem utilizar o ambiente Shell, deixando-o livre para outras utilizações. Para iso usamos o caracter “&” no final do comando.

#updatedb & - Cria o banco de dados a ser utilizado pelo comando locate.

fg

Traz para primeiro plano um processo que está rodando em segundo plano.

#fg updatedb

GERÊNCIA DE ARQUIVOS E DIRETÓRIOS

ls

Mostra os arquivos e diretórios existentes no disco e as suas propriedades. Análogo ao dir do MS-DOS.

Exemplo: # ls /etc

ls -a Mostra arquivos ocultos.

ls -l Mostra detalhes, como permissões de acesso, tamanho e data de gravação.

ls –color Mostra conteúdo de diretórios com detalhes coloridos.

ls -h Mostra o tamanho dos arquivos, utilizando notações humanos.

ls -R Implementa recursividade.

ls -S Mostra arquivos em ordem de tamanho.

ls -t Mostra os arquivos e diretórios ordenados pela data e hora de criação ou última modificação.

ls -u Mostra arquivos e diretórios ordenados pela data e hora do último acesso.

rm

remove diretórios, vazios ou não, e arquivos.

#rm -rf diretório remove diretórios cheios sem pedir confirmação.

cp

copia arquivos e diretórios.

cd /usr/teste /etc - Copia o arquivo teste, presente no diretório /usr, para o diretório /etc.

mv

Move arquivos ou diretórios. Também utilizado para renomear arquivos ou diretórios.

find

Procura por arquivos e diretórios.

#find / -name nome_arquivo

locate

Assim como o comando find, ele procura arquivos.

updatedb

Cria o banco de dados a ser utilizado pelo comando locate.

which

Mostra a localização de um arquivo executável no sistema.

#which reboot

chmod

Altera permissões de leitura, escrita e execução de um arquivo ou diretório.

chown

Altera o proprietário e o grupo de um determinado arquivo.

ln

Cria links (atalhos) para arquivos ou diretórios. O link pode ser do tipo hard e simbólico.

Hard é uma cópia perfeita de arquivo que funciona como um espelho. quando o arquivo é alterado, o espelho também é alterado; simbólico é um atalho para um arquivo ou diretório.

ln /etc/profile/tmp - cria um hard link do arquivo /etc/profile dentro de /tmp.

ln -s /etc/profile/tmp - cria um link simbólico do diretório /etc/profile dentro de /tmp, será um atalho.

diff

Mostra as diferenças entre os conteúdos de dois arquivos texto. Também compara dois diretórios.

mkdir

Cria diretórios.

#mkdir /mala - Cria o diretório mala na raiz do sistema.

#mkdir mala - Cria o diretório mala dentro do diretório no qual estiver operando.

rmdir

Remove diretórios vazios.

rmdir /tmp/teste - Remove o diretório vazio teste, que esta dentro de /usr

cd

Executa a navegação em diretórios. Muda o diretório atual.

#cd /usr - vai para o diretório /usr/bin

pwd

Mostra caminho (path) do diretório atual.

cat

Concatena (junta) arquivos. Se direcionado a um único arquivo, mostra o conteúdo do mesmo.

-n - Numera as linhas

-b - Não numera as linhas em branco

-s - Não exibe mais de uma linha em branco, seqüencialmente.

cat arq1 arq2 - Concatena o conteúdo de arq1 e arq2 e mostra na tela.

du

Mostra o espaço em disco, ocupado por um diretório.

tail

Mostra as últimas linhas de um arquivo. Por default mostra as 10 (dez) últimas linhas.

head

Mostra as primeiras linhas de um arquivo. Por default mostra as 10 (dez) primeiras linhas.

md5sum

Calcula o hash MD5 de arquivos. Hash é um algoritmo matemático que, quando aplicado a um arquivo, analisa a sua seqüência de bits e, em função dessa seqüência, retorna uma cadeia, de tamanho fixo, de números hexadecimais. A operação é irreversível porque a partir de um dado poderemos obter um dado a partir de um hash. No entanto, não poderemos obter um dado a partir de um hash, ou seja, é uma operação de ida sem volta.

md5sum /etc/profile

sha1sum

O SHA1 é outro tipo de hash, calcula o hash SHA1 de arquivos, o seu resultado tem 40 caracteres.

sha1sum /tmp/teste.

dd

Cria uma imagem a partir de dados existentes em uma mídia ou converte uma imagem para uma mídia.

dd if=/dev/cdrom of=teste.iso Cria uma imagem a partir de um CD-ROM.

dd if=/dev/fd0 of=disquete.img Cria imagem com o nome disquete.img a partir de um conteúdo de um disquete.

dd if=/media/floppy0 of=floppy_image Cria uma imagem de um disquete para o arquivo floppy_image no diretório atual.

dd if=floppy_image of=/media/floppy0 O arquivo floppy_image é copiado para o disquete.

mkisofs

Gera uma imagem ISO a partir do conteúdo de um diretório.

cdrecord

Grava imagem ISO em um CD-R, CD-RW,DVD-R.

./<nome do programa>

Utilizado para rodar um executável no diretório atual.

touch

Atualiza data e hora de um arquivo. Tambem cria arquivos vazios.

#touch texto

GERÊNCIA DE CADASTRO

adduser

Adiciona usuários ao sistema. solicita dados e senha do usuário, cria o diretório home do usuário.

useradd

similar adduser, não cria diretório home. Após ser criado poderá ser atribuída uma senha com o comando passwd.

userdel

Exclui usuários do sistema.

passwd

cadastra ou altera a senha de um usuário. Para bloquear o acesso do usuário utilize o comando:

passwd usuário - Cadastra ou altera a senha do usuário.
passwd -l usuário - Bloqueia usuários
passwd -u usuário - Desbloqueia usuários

GERÊNCIA DE USUÁRIOS

su

Substitute user. Troca de usuário corrente.

su teste - Muda para o usuário teste

who

Mostra os usuários que estão conectados no momento, o terminal, a data e a hora de conexão.

whoami

Exibe o nome do usuário que esta conectado.

GERÊNCIA DE MEMÓRIA

free

Mostra os espaços livres e ocupados em memória DRAM e Swap.

GERÊNCIA DE PROCESSAMENTO

ps

Mostra os processos que estão sendo executados.

ps a - Mostra os processos que rodam em todos os terminais. A expressão tty designa terminal local, enquanto que pts designa terminal remoto.

#ps ax - Mostra todos os processos roam do nos terminais e os que independem de terminais.

kill

Envia um sinal para matar um processo em execução.

O killall é bem similar ao kill, com a diferença de que no killall usamos o nome do processo ao invés do PID, assim se tivermos vários processos abertos ao mesmo tempo e com o mesmo nome, killall vai matar todos de uma vez, rodam do em diferentes terminais.

#kill <PID>

#killall <nome do processo>

GERÊNCIA DE MEMÓRIA E PROCESSAMENTO

top

Mostra , em uma interface interativa, a utilização de recursos de CPU e memória por parte dos processos. Similar ao ps.

GERÊNCIA DE SISTEMA

clear

Limpa tela.

set

Mostra as variáveis de ambiente do usuário logado. São diferentes para cada usuário.

df

Mostra os espaços livres e ocupados em disco.
df -h - Mostra a utilização do disco em notações humanas

durep

Instale com # apt-get install durep
Mostra graficamente utilização do disco.

last

Mostra os últimos logins e logouts de usuários, além de reinicialização e desligamentos.

history

Mostra os comandos emitidos pelo usuário naquele terminal.

dmesg

Mostra toda a rotina de inicialização do sistema. Útil para encontrar erros ocorridos na inicialização.

arch

Mostra a arquitetura de processamento da máquina.

Date

Mostra ou altera a data e a hora atual do sistema operacional.
#date mmddaaaa - Altera a data do sistema.

tzsetup

Permite alterar o fuso horário do sistema. Pode-se usar também os comandos:
zconfig e # tzselect

logout

Sai do sistema. O comando exit faria a mesma coisa.

reset

Utilizado para estabelecer o terminal quando o mesmo ficar desfigurado em virtude da leitura de arquivos binários ou operações com resultado não esperado
Exemplo: #cat /bin/arch
#reset

GERÊNCIA DE HARDWARE

lspci

Mostra todos os dispositivos pci presentes na máquina.
cat/proc/cpuinfo
lê o arquivo /proc/cpuinfo, que contem todos os dados sobre o processamento da máquina.

DIVERSOS

cal

Calendário on-line.
#cal -3 - Exibe o calendário do mês atual, o anterior e o posterior.
#cal 2005 - Exibe o calendário do ano de 2005.
#cal -3 - Exibe o calendário do mês atual, o anterior e o posterior.

FILTROS BÁSICOS E PIPES

PIPES

Simbolizado pelo caracter barra (|), é uma implementação que permite que os resultados de um comando seja passado para outro comando. Exemplo:
cat /etc/profile |tail - O resultado será as últimas dez linhas do arquivo.

more
Usado para exibir informações em várias páginas de vídeo. Geralmente é utilizado após um pipe.
dmesg|more - Mostra telas da seqüência de inicialização.
less
Funciona como o more, no entanto permite as setas para navegar para cima e para baixo.
grep
Procura por um texto dentro de um arquivo.
#grep palavra arquivo
#dpkg -l | grep mozilla
>
Lê-se desvio ou redirecionamento. Redireciona alguma saída de dados para um arquivo ou dispositivo. Exemplos:
ls > z.txt - executa um comando ls desvia a saída para um arquivo z.txt, este arquivo será criado, automaticamente dentro do diretório no qual se opera.
>>
Lê-se desvio para o final, desvia para o fim do arquivo, sem apagar o conteúdo anterior. Exemplo:
echo texto >>texto2.txt - Envia o a palavra texto para o final do arquivo texto1.txt.
2>
Lê-se desvio de erro ou redirecionamento de erro. Redireciona o resultado anormal de alguma ação para um arquivo.
#cat /etc/xxx 2> /tmp/erro.txt
#startx 2> /tmp/erro_statx.txt
2>>
Lê-se desvio de erro para o final. Redireciona uma saída de erro para o final de um arquivo. Não sobrescreve um arquivo já existente.

MONTAGEM E UTILIZAÇÃO DE DISPOSITIVOS

Muitos parâmetros podem ser omitidos durante a montagem e formatação de dispositivos. Isso porque os parâmetro a mais utilizados encontram-se na tabela de file systems em /etc/fstab.
No UNIX, diferente do MS-DOS e Windows, eles são montados dentro de um diretório. Para montarmos um dispositivo, utilizamos o comando mount.
Para saber quais unidades de disco ou partições estão montados, basta emitir o comando # mount
Nunca retire uma media antes de desmontá-la.

MONTANDO UM FLOPPY DISK

mount -t vfat / dev/fd0 /media/floppy

Para desmontar um disquete utilize:
#umont /media/floppy

MONTANDO UM CD-ROM
#mount /mnt/cdrom
para desmontar
#umount /dev/cdrom ou #eject

MONTANDO UMA PENDRIVE
mount /dev/sda1 /mnt/pendrive

MONTANDO PARTIÇÃO FAT32 OU NTFS
mount -t vfat /dev/hda1 /mnt/windows
Deverá existir o diretório /mnt/windows
Para desmontar:
Para uma partição NTFS, utilize, utilize -t ntfs. O Linux só consegue trabalhar com NTFS no modelo read-only.
umount /mnt/windows

MONTANDO IMAGEM
mount -o loop nome_da_imagem.iso /mnt/imagem

FORMATANDO UM DISQUETE (fazer como root)
Para formatar o disquete é necessário que ele esteja desmontado. Vamos formatá-lo logicamente com o comando fdformat e logicamente com mkfs (gera o file system), utilize o comando:
fdformat /dev/fd0   e depois
mkfs.vfat /dev/fd0

FORMATANDO UMA PARTIÇÃO DO HD
mkfs.reserfs /dev/hda2

FSTAB

O arquivo /etc/fstab permite que as partições do sistema sejam montadas facilmente especificando somente o dispositivo ou o ponto de montagem. Este arquivo contém parâmetros sobre as partições que são lidos pelo comando mount. Cada linha deste arquivo contém a partição que desejamos montar, o ponto de montagem, o sistema de arquivos usados pela partição e outras opções.

Sistema_de_arquivos	Ponto_de_montagem	Tipo	Opções	dump	Ordem
/dev/hda2	/	reiserfs	notail	0	1
/dev/hdd	/media/cdrom0	iso9660	ro/user/noauto	0	0
/dev/fd0	/media/floppy0	auto	ro/user/noauto	0	0

Onde:
Sistema de arquivos
Partição que deseja montar.
Ponto de montagem
Diretório do Linux onde a partição montada será acessada.
Tipo
Tipo de ssistema de arquivos usado na partição que será montada (vfat para partições fat32, ntfs para partições ntfs, ext3 para partições ext3 do Linux, reiserfs para partições reiserfs do Linux, iso9660 para CD-ROM.
Opções
Especifica as opções usadas com o sistema de arquivos:
defaults – Utiliza valores padrões de montagens.
auto – Permite a montagem em série com o comando #mount -a
noauto – não monta o sistema e arquivos durante o boot, ideal para mídias.
ro – monta como somente leitura.
user - Permite que usuários montem o file system. (não recomendável).
nouser – Não permite que um usuário comum monte o file system.
Ordem
Define a ordem em que os sistemas de arquivos serão verificados na inicialização. Se usar 0, o sistema de arquivos não será verificado. O sistema de arquivos que deverá ser verificado primeiro é o raiz "/".

Após configurar o /etc/fstab, basta digitar o comando:
#mount /media/cdrom0 para montar o CD-ROM
Não é necessário especificar o sistema.

RECUPERAÇÃO DE DESASTRES

RECUPERAÇÃO DO GRUB E MBR

Após instalar o Linux é importante fazer o backup do setor de boot do HD (MBR).
Cria a imagem da seguinte maneira:
dd if=/dev/hda of=mbr.img bs=512 count=
O comando acima irá gerar o arquivo mbr.img, que conterá o GRUB e a tabela de partições do HD. A imagem terá um tamanho de aproximadamente 512 bytes, que pode ser armazenado em um disquete.
O /dev/hda deve ser a partição onde está gravado o GRUB.

Para restaurar o MBR dê o boot com o Kurumin e digite o comando abaixo:
dd if=mbr.img of=dev/hda bs=512 count=1

Podemos fazer o GRUB funcionar a partir de um disquete, para isso crie a imagem no disquete com o comando:
#dd if=/dev/hda of=/dev/fd0 bs=512 count=1
Obs.: O disquete não poderá estar montado na hora da criação.

Outra forma de recuperar o GRUB é dar o boot com o CD do Kurumin ou Knoppix, depois monte a a partição que contém o /boot (geralmente é a partição /) em /mnt. Se o seu /boot estiver em /dev/hda3 digite o comando:
#mount /dev/hda3 /mnt

Se a partição hda3 contém o diretório /boot, deduz que ela é responsável pelo boot do sistema. Para o GRUB, essa é a partição hd0,2. No caso, o primeiro dispositivo IDE (hd0) e a terceira partição (primeira partição é zero).

Entre no prompt do GRUB com o seguinte comando:
#grub

Entre com os comandos:
grub> root (hda,2)
grub> setup (hd0)
grub> quit

Essa sequência de comandos informou ao GRUB qual é a partição que contém o diretório /boot e jogou os dados referentes ao boot no MBR do primeiro HD.

Reinicie a máquina e teste.

PERDA DE SENHA DO ROOT

Caso ocorra a perda de senha do root faça o seguinte:

- No momento da inicialização do computador, selecione no menu do GRUB a opção que dará boot no linux e não tecle ENTER.
- Digite a letra “e”.
- Selecione a linha que começa com kernel.
- Digite a letra “e” novamente.
- Escreva no fim da linha: init=/bin/bash
- Digite ENTER e, depois, a letra “b”.

Após o sistema inicializar,digite no prompt:
mount -o remount, rw /
mount /dev/hda/usr

paswd root - Digite a nova senha de root e reinicialize o sistema.

Caso não funcione tente:
useradd -u0 -d/root -g root root -s/bin/bash

CORROMPIMENTO DE FILE SYSTEM

É possível que haja corrompimento de file system e a máquina não reinicialize corretamente. Nesse caso, será necessário fazer uma checagem de file system. A melhor forma de fazer isto é dando um boot via Kurumin e, em seguida, executar o comando fsck contra as partições (ou reiserfsck, se o filesystem for ReiserFS).

Para ver a tabela de partições do HD, digite:
#fdisk -l

Para checar o /dev/hda2, do tipo ReiserFS, digite:
#reiserfs /dev/hda2 ou
#fsck.reiserfs /dev/hda2
Obs.:As partições não podem estar montadas por isso usamos um sistema externo de boot (no caso o Kurumin)

NÍVEIS DE OPERAÇÃO

As maioria das distribuições Linux funcionam com sete runleavels:
0 é o halt responsável por desligar o sistema
1 utilizado para manutenção do sistema, não ha rede,serviços,monousuários.
2 multiusuário sem rede
3 similar ao nível 2 com rede
4 reservado para uso local
5 exclusivo para o ambiente gráfico
6 utilizado para reinicializar o sistema

O Debian segue uma lógica diferente:
0 halt
1 modo monousuário
2 a 5 multiusuário
6 reboot

Cada runleavel pode ser alterado com o comando init x, onde x é o runleavel desejado.
runleavelmostra o runleavel atual

PERMISSÕES DE ACESSO E EXECUÇÃO

No Linux, cada arquivo/diretorio tem um dono, um grupo e permissões de escrita, leitura e execução.
Execute o comando a seguir:
cd /etc
ls -l

Possibilidades	*Ligações fortes*	*Dono do Arquivo*	*Grupo do arquivo*	*tamanho*	*Data/hora*	*Nome*

Possibilidades	*Ligações fortes*	*Dono do Arquivo*	*Grupo do arquivo*	*tamanho*	*Data/hora*	*Nome*
-rw-r—r--	*1*	root	root	848	2003-03-05 13:01	ksysguard

Para alterarmos as permissões de escrita (w), execução (x) e leitura (r), devemos utilizar o comando chmod, com a seguinte sintaxe:
#chmod dg o <arquivo ou diretório ou link>
Onde “d”, “g” e “o” representam as permissões para o dono do arquivo (exceto o root, pois esse tem acesso a qualquer arquivo/diretorio), o dono e outros.

R	*W*	*X*
4	2	1

chmod 750 /etc/ppp Dá permissão de rwx ao dono, rx ao grupo e nenhuma para outros. Tudo em relação ao diretório /etc/ppp

SUID

É um recurso que faz com que qualquer usuário, ao executar um determinado arquivo, tenha os mesmos direitos do dono do arquivo. Geralmente essa técnica é usada para arquivos pertencentes ao root.
ls -l /sbin/halt
-rwxr-xr-x 1 root root 10208 2005-01-04 20:43/sbin/halt

O comando halt é utilizado para desligar o computador, execute como um usuário comum o seguinte comando:
/sbin/halt -p

Retornará uma mensagem que somente o root pode executá-lo. Isso ocorre porque somente o root tem permissão para executá-lo.

Para conceder a permissão de execução para qualquer usuário basta acrescentar o número 4 na frente da permissão original:
chmod 4777 /sbin/halt

Verificando os detalhes do arquivo /sbin/halt teremos:
ls -l /sbin/halt
-rwsxr-xr-x 1 root root 10208 2005-01-04 20:43/sbin/halt

Observe que, no lugar do x do dono apareceu um s. Esse s é de SUID. Isso quer dizer que quem executar tal arquivo, terá os mesmos privilégios de root sobre ele. Agora então teremos sucesso na execução do comando /sbin/halt -p.

SGID

O SGID (Set Group ID), é idêntico ao SUID. No entanto, é voltado para o grupo do arquivo. Qualquer usuário que executar um arquivo com SGID terá os mesmos direitos do grupo do arquivo. O SGID é atribuído com o número 2.
chmod 2777 /sbin/halt
Obs.: Para atribuir SUID e SGID ao mesmo tempo, utilize o número 6 (4+2).

GRUPOS

Para criação de grupos utilize o comando:
addgroup labcta

Execute o comando abaixo para verificar o grupo criado:
cat /etc/group

Para inserir usuários dentro de um grupo editaremos o arquivo /etc/group, e digitamos o nome do usuário em seu respectivo grupo:
lab2cta:x:1002:aluno1,aluno2

Para verificar o grupo de um usuário logado use o comando, se estiver logado, deve fazer logout e logar-se novamente:
#groups ou #id

Para excluir um grupo use o comando:
#groupdel lab2cta
Obs.:Não podemos remover um grupo primário de um usuário, primeiro remova o usuário.

INICIALIZANDO E PARANDO DAEMONS

Existem métodos que permitem que os serviços sejam inicializados, parados ou atualizados, sem a necessidade de reiniciar todo o sistema.

A maioria dos daemons que selecionamos no rcconf encontram-se nos diretórios /etc/init.d/.
para executá-los usamos os comandos:
/etc/init.d/apache2 stop - Para o apache2
/etc/init.d/apache2 start - Inicia o apache2
/etc/init.d/apache2 restart - Para e em seguida reativa o serviço
/etc/init.d/apache2 reload - Os arquivos de configuração são relidos sem parar o serviço.

Chamamos os daemons que rodam independentes, presentes em /etc/init.d, de standalone; e daemons inetd aqueles controlados pelo sistema inet.d ou xinetd

LOGS DE SISTEMA

Os logs de sistema ficam em /var/log.

- auth.log: mostra todos os logins realizados nos terminais ou remotamente por ftp, telnet, ssh, etc.
- daemon.log: armazena os dados gerais da atividade de daemons e de rede.
- dmesg: contém os eventos de inicialização do sistema.
- mail.info e mail.log: mostram atividades de smtp, pop3, imap, antivírus.
- kern.log: mostra as mensagens emitidas pelo kernel.
- syslog: outra boa fonte para verificar atividades daemons em geral em rede.

INSTALAÇÃO DO AMBIENTE GRÁFICO

INSTALAÇÃO DO KDE

Para instalar o Kde, em português, execute:
apt-get install kde
#apt-get install kde-i18n-ptbr.
O arquivo para refazer a configuração é o /etc/X11/XF86Config-4. As seguir veremos as principais linhas de configuração desse arquivo:

- Tipo de mouse

Durante a instalação é perguntado qual a porta o mouse está conectado. Utilize o esquema abaixo:

mouse serial	/dev/ttySx
mouse PS/2	/dev/psaux
mouse USB	/dev/imput/mice
mouse PS/2 genérico sem wheel	PS/2
mouse PS/2 genérico com wheel	ImPS/2

Linhas de configuração do mouse no arquivo XF86Config-4:

```
Section "InputDevice"
        Identifier    "Generic Mouse"
        Driver              "mouse"
        Option              "SendCoreEvents"       "true"
        Option              "Device"           "/dev/input/mice"
        Option              "Protocol"         "ImPS/2"
        Option              "Emulate3Buttons"      "true"
EndSection
```

- Configuração de cores

A linha DefaultDepth define a quantidade de cores da tela de acordo com a tabela abaixo:

Depth	Cores
1	2 cores
4	16 cores
8	256 cores
15	32.768 cores
16	65.536 cores
24	16.777.216 cores

O valor Depth deve ser inserido na linha DefaultDepth da Section “Screen”.

```
Section "Screen"
        Identifier    "Default Screen"
        Device              "Trident Microsystems CyberBlade/i7"
        Monitor             " Digital 14 in. Color Monitor (FR-PCXCV-C*)"
        DefaultDepth    16
        SubSection "Display"
```

Acresça na sub-seção “Display” referente ao Depth escolhido os valores: “1024x768” “800x600”.

```
        SubSection "Display"
                Depth               16
                Modes               "1024x768" "800x600" "640x480"
        EndSubSection
```

- Ajuste frequência do monitor

No caso de haver “fliquer” (tela piscando) ou não conseguir colocar a interface gráfica na resolução de vídeo pretendida, será necessário ajustar as freqüências horizontal e vertical do monitor.

Ajuste as freqüências na seção “monitor”, nas linhas “HorizSync” e “VertRefresh”, o primeiro valor representa a menor freqüência possível e o segundo valor representa a maior freqüência possível, é nesse último valor que devemos atuar. Vá testando até descobrir o maior valor que o monitor agüenta sem perder o sincronismo. Faça primeiro com a sincronia horizontal (HorizSync), que poderá variar de 28-33 a 28-85, depois com a vertical (VertRefresh) que poderá variar de 43-42 a 43-150.

```
Section "Monitor"
        Identifier    " Digital 14 in. Color Monitor (FR-PCXCV-C*)"
        HorizSync         30.0-54.0
        VertRefresh       50.0-90.0
        Option            "DPMS"
EndSection
```

Após alterar a configuração desse arquivo restart ele com:

```
# / etc / init.d / xdm restart   ou
# / etc / init.d / gdm restart
```

ALTERNÂNCIA DE TERMINAIS

A partir de um ambiente gráfico, a alternância de terminais deve ser feito com as teclas Ctrl+Alt+Fx, aonde x vai de 1 a 8.

CONFIGURAÇÃO DA REDE

A maioria das placas de rede PCI são reconhecidas automaticamente pelo Debian.

Para verificar as configurações de rede digite:

```
# ifconfig
```

A configuração dos adaptadores de rede é feita no arquivo : /etc/networking/interfaces.

Este arquivo vai ficar da seguinte forma:

```
auto eth0
iface eth0 inet static
```

address 10.0.1.1
netmask 255.255.0.0
network 10.0.0.0
broadcast 10.255.0.0
gateway 10.0.3.5

Após configurar o adaptador de rede, é necessário reinicializar a rede:

```
# /etc/init.d/networking restart
```

ALIAS DE IP

Para permitir estabelecer mais de um endereço IP para a mesma placa de rede, teremos que renomear a interface.

Usa eth0 para interface original, eth0:0 para a primeira alias da eth0; eth0:1 para a segunda alias da eth0. O arquivo vai ficar da seguinte forma:

```
auto eth0
iface eth0 inet static
        address     10.0.1.1
        netmask     255.255.0.0
        network     10.0.0.0
        broadcast   10.255.0.0
        gateway     10.0.3.5

auto eth0
iface eth0 inet static
        address     127.1.1.15
        netmask     255.255.0.0
        network     127.0.0.0
        broadcast   101.25.15.0
```

Não pode haver mais de um default gatway na mesma máquina.

Os endereços dos servidores DNS poderão ser configurados em /etc/resolv.conf .

COMANDOS EM REDES TCP/IP

ping

Comando utilizado para saber se um pacote está chegando no seu destino.

```
# ping 10.0.0.1          - Verifica se há conexão com a máquina.
```

arp

```
# arp -a 10.0.0.1        - Mostra o MAC das máquinas que tiveram comunicação com o nosso host.
# arping 10.0.0.1        - Mostra o MAC do adaptador que está sendo pingado.
```

ifconfig
Mostra as configurações dos adaptadores de rede local.
ifconfig
#ifconfig eth0 10.0.0.5 netmask 255.255.255.0 up - Habilita a rede.
#ifconfig eth0:0 10.0.0.6 up - Habilita um IP virtual.
#ifconfig eth0 down - Desabilita interface de rede.

route
Edita a tabela de roteamento de rede.
#route – Mostra a tabela de roteamento.
#route add default gw ip_do_gateway - Estabelece uma rota default

wget
Utilizado para fazer dowloads de arquivos e diretórios em modo texto.
#wget endereço_www_ou_ftp

lynx
Utilizado para navegar em modo texto.

ssh
Modo mais seguro e atual de fazer acesso remoto. Similar ao telnet, porém os dados trafegam criptografados. O login sera estabelecido como root.
#ssh 10.0.0.5
Para fazer login com usuários diferentes utilize o caracter @:
#ssh aluno@10.0.0.5

scp
Usado para fazer transferência de arquivos, de forma encriptada, entre hosts.
#scp arquivo1 root@ip_máquina_destino:/home/aluno
O comando acima irá transferir o arquivo arquivo1 da máquina local para o diretório /home/aluno de uma outra máquina.
#scp root@ip_máquina_destino:/home/aluno/arquivo2
O comando acima fará a transferência do arquivo arquivo2 de uma máquina remota para a máquina local.

who
Mostra quem está atualmente conectado no computador.
#who -H Mostra o cabeçalho das colunas.
#who -i Mostra o tempo que o usuário está parado em Horas:Minutos.

whoami
Mostra o nome que usou para se conectar ao sistema.

dnsdomainname
Mostra o nome do domínio de seu sistema.

hostname
Mostra ou muda o nome de seu computador na rede.

users
Mostra os nomes de usuários usando atualmente o sistema.

AGENDAMENTO DE TAREFAS

CRON

O CRON permite agendar tarefas no Linux, a serem realizadas periodicamente pelo sistema, utilizando como parâmetros mês, dia do mês, dia da semana, hora, minuto, ação a ser realizada.
Para editar o CRON digite:
crontab -e - Edita crontab
crontab -l - Mostra o crontab atual
crontab -r - Remove todo o crontab
Para criar uma terefa basta inserir uma linha com a seguinte sintaxe:
minuto hora dia_do_mês mês dia_da _semana /path/comando

Exemplo:
15 23 * * * /sbin/reboot
- Reinicia o sistema às 23:15hs, todos os meses dias e ano.

15 23 2 * 0 /sbin/reboot
- Reinicia o sistema às 23:15hs, todos os dias 2 do mês, em todos os meses, desde que seja domingo.

15 23 2 * * /sbin/reboot
- Reinicia o sistema às 23:15, todos os dias 2 de todos os meses e ano.

0 0,12 * * 1-3 /sbin/reboot
- Reinicia o sistema às 00:000 e às 12:00, e segunda à quarta-feira

0 0,12 * * 1-3,5 /sbin/reboot
- Reinicia o sistema às 00:000 e às 12:00, e segunda à quarta-feira e sexta-feira.

*/10 * * * * /sbin/reboot
- A cada 10 minutos o sistema irá reiniciar.

O CRON não precisa ser parado e reinicializado para ler o crontab. Após a edição do crontab, as novas tarefas já estão valendo.

Ele deve ser marcado no rcconf.

INSTALAÇÃO DO WEBMIN

O webmin é um programa para administração do sistema através browser. Através de uma interface web é possível configurar o sistema e servidores. Funciona local ou remotamente.

Para o webmin rodar o apache deverá estar instalado e rodando.

Para instalar o webmin siga o comando abaixo:

apt-get install webmin
para testar abra o browser e digite ip_da_máquina:10000 ou 127.0.0.0:10000.

MÓDULOS DO WEBMIN

Para procurar módulos disponíveis para o webmin digite:
apt-cache search webmin
Instale o módulo desejado com o seguinte comando:
apt-get install nome_do_módulo

CRIAÇÃO DO REPOSITÓRIO DEBIAN

Com a criação de um repositório, não haverá necessidade de buscar os pacotes necessários para instalação em um servidor externo, uma rede interna poderá baixar os pacotes de um servidor dentro de uma rede local. Isso significa um menor tempo na instalação de pacotes.

A seguir é descrito um processo para montagem de um repositório:

- Crie no diretório raiz o diretório /debian.
- Crie o diretório /ISO dentro do diretório /debian.
- Baixe as imagens de um mirror (http://linorg.usp.br/iso/debian/3.1_r0a/), baixe as imagens iso-dvd que são apenas 2(duas).
- Copie as imagens para o diretório /debian.
- Monte as imagens no diretório /debian/iso.

mount - o loop debian-31r0a-i386-binary-1.iso /debian/iso
mount - o loop debian-31r0a-i386-binary-2.iso /debian/iso

Para as máquinas acessarem o repositório insira no sources.list o caminho do repositório. Exemplo:
deb http://ip_do_repositório/debian/iso1/ Stable main
deb http://ip_do_repositório/debian/iso2/ Stable main

Importante: no arquivo /etc/apt/apt.conf, a linha de configuração do proxy de ser comentada com ' (aspas simples).

ANOTAÇÕES:

SERVIDOR WWW

1. INTRODUÇÃO

O servidor web é um programa responsável por disponibilizar páginas, fotos, ou qualquer outro tipo de objeto ao navegador do cliente.

O Apache é um servidor Web extremamente configurável, robusto e de alta performance desenvolvido por uma equipe de voluntários (conhecida como Apache Group) buscando criar um servidor web com muitas características e com código fonte disponível gratuitamente via Internet.

A primeira versão oficial do Apache foi a 0.6.2, lançada em Abril de 1995.

2. CARACTERÍSTICAS

Abaixo estão algumas características que fazem esse servidor web o preferido entre os administradores de sistemas:

5. Possui suporte a scripts cgi usando linguagens como *Perl, PHP, Shell Script, ASP, etc*.
6. Suporte a autorização de acesso podendo ser especificadas restrições de acesso separadamente para cada endereço/arquivo/diretório acessado no servidor.
7. Autenticação requerendo um nome de usuário e senha válidos para acesso a alguma página/sub-diretório/arquivo (suportando criptografia via Crypto e MD5).
8. Negociação de conteúdo, permitindo a exibição da página Web no idioma requisitado pelo Cliente Navegador.
9. Suporte a tipos mime.
10. Personalização de logs.
11. Mensagens de erro.
12. Suporte a virtual hosting (é possível servir 2 ou mais páginas com endereços/ portas diferentes através do mesmo processo ou usar mais de um processo para controlar mais de um endereço).
13. Suporte a IP virtual hosting.
14. Suporte a name virtual hosting.
15. Suporte a servidor Proxy ftp e http, com limite de acesso, caching (todas flexivelmente configuráveis).
16. Suporte a proxy e redirecionamentos baseados em URLs para endereços Internos.
17. Suporte a criptografia via SSL,Certificados digitais

18. Módulos DSO (Dynamic Shared Objects) permitem adicionar/remover funcionalidades e recursos sem necessidade de recompilação do programa.

- **VERIFICAR INSTALAÇÃO**
  # dpkg -l | grep apache2

4. INSTALAÇÃO
 # apt-get install apache2

5. DESINSTALAÇÃO
 # apt-get remove apache2
 # dpkg -P apache2

6. VERIFICAR O FUNCIONAMENTO

Abrir o browser e digitar o IP da máquina (ou nome_da_máquina ou 127.0.0.1 ou localhost)

7. CONFIGURAÇÃO DO SERVIÇO

Os arquivos responsáveis pela configuração do servidor apache estão localizados em:
/etc/apache2:

- **apache2.conf**
  Arquivo de configuração principal.
- **conf.d**
  Diretório para adicionar diretrizes indicadas em apache2.conf.

- **http.conf**
  Arquivo vazio

- **magic**
  Carrega dados mágicos para para o módulo mine. Não ha necessidade de mexer nesse arquivo.

- **mods-available**
  Este diretório contém uma série de arquivos de configuração necessários para carregar e utilizar módulos do Apache.

- **mods-enabled**
  Este diretório contém links de mods-available para permitir a utilização de módulos do apache.

- **ports.conf**
  Arquivo de configuração da porta utilizada pelo apache.

- **sites-available**
  Contém o arquivo "default' utilizado para configuração de hosts virtuais.

- **sites-enable**
  Similar a função do mods-enable, contém links para arquivo de configuração em mods-available que deve ser utilizado.

O diretório default para armazenar as páginas html é o **/var/www**. O diretório onde serão armazenadas as páginas html deverão ter a permissão 755.

Arquivos de log criados pelo Apache

O servidor apache2 grava seus arquivos de log geralmente em /var/log/apache2, tanto os seus nomes como conteúdo podem ser personalizados nos arquivos de configuração . Mesmo assim, os arquivos de logs encontrados na instalação padrão do Apache2 são os seguintes:

- **access.log** - Registra detalhes sobre o acesso as páginas do servidor apache2.
- **error.log** - Registra detalhes saber erros de acesso as páginas ou erros internos do servidor.

8. INICIALIZAÇÃO DO APACHE2

#/etc/init.d/apache2 start	iniciar o servidor apache2
# /etc/init.d/apache 2 stop	parar o serviço apache
# /etc/init.d/apache2 restart	Reinicia o serviço apache após uma pausa de 5s.

- **ARQUIVO /etc/apache2/apache2.conf**

ServerRoot "/etc/apache2"

É o caminho do diretório onde irão ficar os arquivos de configuração. Pode ser mudado se necessário.

PidFile /var/run/apache2.pid

Arquivo onde fica armazenado o PID (número que identifica o processo)

Timeout 300.

Tempo máximo (em segundos) que o servidor esperará, mantendo uma conexão aberta com o cliente. Se o limite for excedido, ele terá de criar uma nova conexão com o mesmo.

KeepAlive On

Define se vai permitir ou não conexões persistentes (mais que uma requisição por conexão). Mude para "Off" para desativar.

MaxKeepAliveRequests 100

Diretamente associado a opção anterior. Determina o número máximo de requisições que serão permitidas durante uma conexão persistente. Mude para 0 para permitir uma quantidade ilimitada. É recomendamos deixar este número alto, para obter a máxima performance

ErrorLog /var/log/apache2/error.log

Arquivo onde serão gravados os logs relacionados a erros do servidor.

Include /etc/apache2/ports.conf

Arquivo de configuração da porta.

Alias /icons/ "/usr/share/apache2/icons/"

Alias para diretórios de imagens.

DirectoryIndex index.html index.cgi index.pl index.php index.xhtml

Nome padrão para procura de páginas html no diretório.

LanguagePriority en da nl et fr de el it ja ko no pl pt pt-br ltz ca es sv tw

Permite definir a prioridade para a exibição de documentos caso nenhum documento confira durante a negociação de conteúdo. Para fazer isto, especifique os idiomas em ordem de preferência de exibição de idiomas.

IMPORTANTE: Após qualquer mudança nos arquivos e configuração do Apache2, o serviço deverá ser reinicializado.

PRÁTICA

- Mudar a página default do apache para o idioma pt-br
- Abrir o arquivo apache2.conf e colocar pt-br no inicio da linha:
  LanguagePriority pt-br en da nl et fr de el it ja ko no pl pt ltz ca es sv tw

- Criar uma página em html em /var/www para acessar como página principal de um servidor

Abrir o arquivo /etc/sites-available/default
Comentar a linha:
#RedirectMatch ^/$ /default_html/

- Criar a página de teste index.html em /var/www
- Reinicializar o servidor
- Acessar a pagina digitando o ip da máquina ou host em um browser.

- Criação de diretório virtual de um servidor
- Criar um direttório com o nome virtual em /var/www
- Criar uma página index.html ou deixar de forma que apareça uma lista de arquivos e diretórios.
- Acessar o através browser, digitando: host_da_máquina/virtual.

- Criação de um host virtual baseado em nome
- Criar o diretório /HostVirtual

mkdir /HostVirtual
chmod 755 /HostVirtual

- Criar em /HostVirtual o arquivo index.html
- Criar o host spider2 em /etc/localhost
- Abrir o arquivo /etc/apache2/sites-available/default e inserir no final

```
NameVirtualHost 10.0.0.1
     <VirtualHost spider>
     ServerName spider.localdomain
    ServerAdmin admin@site.com.br
     DocumentRoot /var/www
     </VirtualHost>

    <VirtualHost spider2>
     ServerName spider2.localdomain
     ServerAdmin admin@site.com.br
     DocumentRoot /HostVirtual
    </VirtualHost>
```

- Abrir as páginas “spider.localdomain” e “spider2.localdomain” em um browser

- Criação de host virtual baseado em IP
- Criar IPs virtuais para servidor www.
  ifconfig eth0:0 10.0.0.10 netmask 255.255.252.0 up

Dê um ping no IP criado para confirmar sua criação

- Criar o diretório /var/www/virtual
  # mkdir /var/www/virtual

chmod 755 /var/www/virtual

- Criar em /var/www/virtual o arquivo index.html
- Criar o host virtual para o ip 10.0.0.10 em /etc/localhost
- Abrir o arquivo /etc/apache2/sites-available/default e inserir no final

```
NameVirtualHost virtual
<VirtualHost 10.0.0.10>
        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/virtual/
        <Directory />
                Options FollowSymLinks
                AllowOverride None
        </Directory>
</VirtualHost>
```

- Testar a configuração, abrindo a páginas “virtual.localdomain” e pelo ip“10.0.0.10” em um browser

- Restrição de acesso à pagina por usuário e senha
- Usando como exemplo a página index.html em /var/www
- Criar em /var/www o arquivo .htaccess com o seguinte conteúdo:

```
AuthName "Acesso Restrito à Usuários"
AuthType Basic
AuthUserFile /var/www/acesso
require valid-user
```

Onde:

- AuthName: O nome que aparece como mensagem de Login. Pode usar algo como "Entre com Login e Senha", ou coisa deste tipo.

15. AuthType: Tipo de autenticação. Atualmente o Basic é o tipo mais comum. Existe também o "Digest", mas ainda não é muito utilizado e suportado pelos clientes.
16. AuthUserFile: Onde está o arquivo de usuários e senhas que agente criou.
17. require valid-user: O que o Apache precisa para validar o acesso. Neste caso a gente indicou que precisa de um usuário válido para acessar a página, ou seja, alguém que digitou um usuário e senha e bateu com o que está no arquivo de senhas. Pode-se restringir para apenas alguns usuários do arquivo de senhas. Por exemplo, se eu quisesse restringir apenas para o usuário eitch e sakura, ao invés de "require valid-user", ficaria "require user eitch sakura".

- No diretório /var/www criar os usuários alfa e bravo para acesso à página:

htpasswd -c acesso alfa (o -c cria o diretório acesso)
htpasswd acesso bravo

SERVIDOR DE NOMES - BIND -

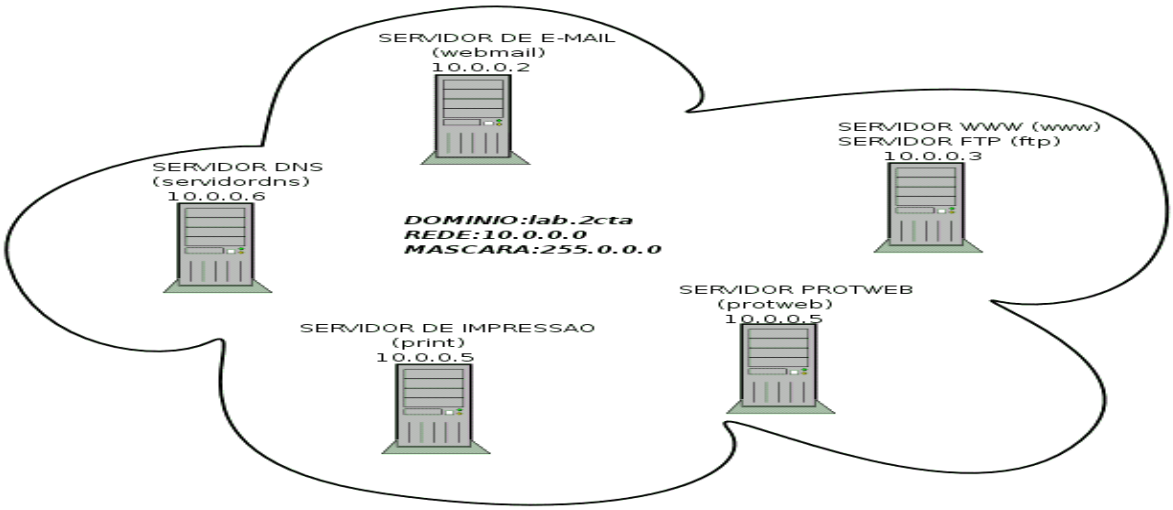
1. DEFINIÇÕES

DNS ou Domain Name System - consiste num serviço, onde são armazenadas ligações entre endereço IPs e domínios. Quando se pede ao seu navegador, cliente de email,

cliente de ftp, ou qualquer outro aplicativo para chamar um determinado domínio, automaticamente ele ascende ao servidor DNS configurado, e encontra o respectivo endereço IP da máquina que fornece o serviço requisitado e, assim, torna-se possível utilizar determinados serviços usando nomes em oposição a endereços IP.

- **BIND** (Berkeley Internet Name Domain) é o servidor de DNS mais usado na Internet, Foi criado originalmente por Paul Vixie em 1988 ao trabalhar para o DEC.

As configurações mostradas no decorrer desta nota de aula estão baseados no diagrama de rede abaixo representado:



- **PASSOS PARA UTILIZAÇÃO DO DNS**
- Verificar instalação do pacote bind

#dpkg - l | grep bind

- Instalar o bacote bind

 # apt-get install bind

- Tornar o sevidor cliente dele mesmo

incluir o ip do servidor no arquivo /etc/resolv.conf

- Configurar o servidor
- Colocar no ar o daemon do DNS (bind)

3. O QUE DECLARAR NO DNS?

Um servidor de nomes deve ser configurado de modo a não fornecer dados excessivos, ao ponto de auxiliar um ataque de rede. Assim, devemos declarar no DNS:

- Todos os hosts que necessitam ser chamados por um nome;
- As impressoras de rede;

Não devem constar no DNS:

- Os roteadores e firewall;
- As máquinas clientes de rede, a não ser que haja uma necessidade disso;
- Outras máquinas que não precisarem ser chamadas pelo nome.

4. LINUX COMO CLIENTE DNS

Para tornarmos o Linux um cliente DNS, devemos editar o arquivo /etc/resolv.conf. Nesse arquivo são estabelecidos o dominio local e os servidores de nomes da rede. Configure-o da seguinte forma:

search lab.2cta
nameserver ip_servidor_DNS_primário
nameserver ip_servidor_DNS_secundário

A linha search designa o dominio local e é utilizada para complememntos de nomes de hosts. poderemos ter de uma a três linhas namesever, designamdo os servidores de nomes.

5. LINUX COMO SERVIDOR DNS PRIMÁRIO

Um servidor DNS simples realiza os seguintes trabalhos básicos:

- Cache em memória RAM das URLs consultadas;
- Resolução direta (ao entrar um nome, será devolvido o seu respectivo IP). É configurada para os dominios da máquina, incluindo a rede local (127.0.0.0).

6. ARQUIVOS DE CONFIGURAÇÃO DO BIND

Considerando a rede 10.0.0.0, cujo dominio é lab.2cta, para configurar o servidor DNS, iremos utilizar os seguintes arquivos:

- /etc/bind/named.boot – conterá os dados essenciais relativos à configuração do DNS, indicando o diretório de configuração e os arquivos de configuração.
- /etc/bind/named.conf – arquivo de configuração principal. Será gerado a partir do /etc/bind/named.boot.
- /etc/bind/db.root – contém a localização dos servidores DNS raiz do mundo. É útil quando o servidor DNS está trabalhando na internet. Não deverá ser alterado. Em algumas distribuições chama-se named.ca.
- /etc/bind/db.local – responsável pela resolução reversa direta do dominio localhost.
- /etc/bind/db.127- responsável pela resolução reversa da rede 127.0.0.0.
- /etc/bind/db.lab.2cta – responsável pela resolução direta do dominio ao qual o DNS pertence.
- /etc/bind/db.10 – responsável pela resolução reversa da rede a qual o DNS pertence.

OBS.: Os nomes acima citados (db.root, db.local, db.127, db.lab.2cta e db.10) representam uma padronização que pode ser alterada a qualquer momento, bastando declarar isso no arquivo /etc/named.boot e, em consequencia, em /etc/named.conf.

7. CONFIGURAÇÃO DO SERVIDOR DNS PRIMÁRIO

7.1. Arquivo /etc/bind/named.boot

O arquivo /etc/bind/named.boot é um dos principais arquivos do DNS. É ele quem determina os arquivos que irão conter as configurações.

```
directory                       /etc/bind
cache           .               db.root
primary         localhost       db.local
primary         127.in-addr.arpa db.127
primary         lab.2cta        db.lab.2cta
primary         10.in-addr.arpa db.10
xfrnets         none
```

Foram os seguintes parâmetros utilizados:

- *directory /etc/bind*

Determina que o diretório de trabalho para todos os arquivos especificados será o /etc/bind.

- *cache . db.root*

Especifica o arquivo que conhece os DNS de nível superior (raízes). O ponto no meio especifica os DNS raizes e não deve ser esquecido.

- *primary localhost db.local*

Determina que a resolução direta para localhost será feita pelo arquivo db.local.

- *primary 127.in-addr.arpa db.127*

Determina que a resolução reversa para a rede 127.0.0.0 será feita pelo arquivo db.127.

OBS.: O mapeamento reverso é obtido com os octetos que representam a rede, sem os zeros completadores, escritos de trás para frente, juntamente com a expressão in-addr.arpa, que representa reverso. Exemplos:
Rede: 10.0.0.0
Mascara: 255.0.0.0
Mapeamento:10.in-addr.arpa

Rede: 192.20.5.0
Mascara: 255.255.255.0
Mapeamento:5.20.192.in-addr.arpa

- primary lab.2cta db.lab.2cta

Determina que a resolução direta para o dominio lab.2cta será feito pelo arquivo db.lab.2cta.

primary 10.in-addr.arpa db.10

Determina que a resolução reversa para a rede 10.0.0.0 será feita pelo arquivo db.10.

- xfrnets none

Essa linha bloqueia as requisições de passagem de tabelas realizadas por um DNS secundário. Isso faz parte da segurança.

A linha:
xfrnets 10.0.0.19 10.0.0.25
permitirá que os hosts listados atuem como DNS secundários.

OBS.: O arquivo named.boot, muito utilizado antigamente, já não existe mais. Em substituição, foi criado o named.conf. Mas como muitos já estavam acostumados com o named.boot e ele é bem mais fácil de ser escrito, foi desenvolvida uma rotina em linguagem perl, para a criação do named.conf a partir do named.boot. É o named-bootconf. Para gerar o named.conf, vá para /etc/bind, crie o arquivo named.boot e execute a seguinte linha de comando:

 # named-bootconf < named.boot > named.conf

Como resultado será gerado o /etc/bind/named.conf baseado no named.boot.

7.2. Arquivo /etc/bind/named.conf

Não é difícil entender o named.conf. Se quiser, voce poderá fazê-lo diretamente, sem a conversão do named.boot.

```
options {
        directory "/etc/bind";
        allow-transfer {
                none;
        };
};

zone "." {
        type hint;
        file "db.root";
};

zone "localhost" {
        type master;
        file "db.local";
};

zone "127.in-addr.arpa" {
        type master;
        file "db.127";
};

zone "lab.2cta" {
        type master;
        file "db.lab.2cta";
};

zone "10.in-addr.arpa" {
        type master;
        file "db.10";
};
```

7.3. Arquivo etc/bind/db.local

A resolução direta do dominio localhost segue a seguinte estrutura:

```
$TTL    1D
@           IN      SOA     servidordns.lab.2cta.  araujo.lab.2cta. (
                            20051101      ; Serial
                            3H            ; Refresh
                            15M           ; Retry
                            1W            ; Expire
                            1D )          ; minimum
```

IN NS servidordns.lab.2cta.
localhost. IN A 127.0.0.1

Na configuração acima, uitlizamos algumas notações que analizaremos agora.
Para comentários nos arquivos de configuração de DNS, utiliza-se o ponto e vírgula (;).

- TTL – Time To Live
  É o tempo máximo, em segundos, que as informações prestadas poderão ser armazenadas em cache para serem utilizadas por quem as solicitou. Depois desse tempo, o solicitante deverá consultar novamente o DNS. Foi utilizada o alor 1D, equivalente a 1 dia.

- @ - Zona definida em /etc/bind/named.conf
  É um substituto para o nome completo para o nome da zona, Sempre que aparecer, irá se referir à zona a ser configurada. A zona a foi definida em /etc/bind/named.conf.

- IN – internet
  Utilizado antes de dos registros deDNS. Os registros representam as informações básicas a serem prestadas.

- IN SOA – Start Of Autthorithy
  É um registro que indica o inicio da configuração de uma zona. O SOA compreende o cabeçalho da configuração. Esse cabeçalho é composto pelos seguintes ítens:
  Uma linha inicial, do tipo:
  @ IN SOA servidordns.lab.2cta. araujo.lab.2cta. (
  Onde lemos: dominio (zona) específicado em /etc/bind/named.conf (caractere @); inicie a configuração da citada zona (IN SOA), cujo DNS principal será a máquina servidordns.lab.2cta.; o endereço eletronico de quem configurou o DNS é araujo@lab.2cta (o caractere @ deve ser substituido por um ponto); abre parenteses para configurar dados de sincronização.

Alguns dados para sincronização, assim discriminados:

```
20051101  ; Serial
 3H       ; Refresh
15M       ; Retry
1W        ; Expire
1D )      ; minimum
```

- Serial: representa a versão da configuração. Geralmente, utiliza-se a data, no formato YYYYMMDD, uma vez que ela nunca mais se repetirá e será útil no controle. Deve ser incrementado cada vez que que houver alteração no arquivo;
- Refresh: é o intervalo de tempo que o DNS secundário deverá levar para comparar suas informações com as do servidor. Foi utiliado o valor 3H, equivalente a 3horas.
- Retry: caso o servidor primário esteja for a d o ar, este campo informa quanto tempo o DNS secundário deverá aguardar antes de un novo contato. Nesse caso 15 minutos.
- Expire: se ainda não houver comunicação, o DNS secundário responderá às consultas de DNS, no lugar do primário, por um tempo x definido por Expire. No caso foi definido 1W, referente à uma semana. Depois disso, o secundário para de responder.

- Minimum: representa o tempo que os clientes devem manter as informações recebidas em cache. Funciona como o $TTL. Depois disso, fecha-se o parenteses e esta pronto o cabeçalho.

● IN NS – Name Server
Registro que indica quais máquinas máquinas serão as servidoras DNS da zona configurada em IN SOA. Atente para a existencia de um ponto no final do nome.

● IN A – Address
Registro que correlaciona nomes de máquinas com endereços IP.
A utiliza o formato:
<nome da máquina> IN A <endereço IP>
O nome pode ser inserido de duas formas diferentes:
- Somente o nome sem o dominio. Nesse caso, O DNS server irá inserir, automáticamente o dominio:

servidor IN A 10.5.0.2

Nesse caso o próprio DNS irá compor o domínio, resultante em:
servidor.lab.2cta
- Nome completo com um ponto no final. Fica assim:

servidor.lab.2cta. IN A 10.5.0.2

O ponto final é muito importanete, pois evita a composição de dominio. Se não colocarmos, o DNS irá compor o dominio, resultando em:
servidor.lab.2cta.lab.2cta

Cabe ainda ressaltar que podemos inserir mais de um nome para o mesmo IP, fazendo com que a máquina responda pelos dois nomes. Exemplo:

```
servidor.lab.2cta   IN  A   10.0.0.3
www.lab.2cta        IN  A   10.0.0.3
ftp.lab.2cta        IN  A   10.0.0.3
```

● IN CNAME – canonical name
Usado para criar aliases em relação às definições IN A.
Exemplo:

```
servidor  IN  A          10.0.0.3
www       IN  CNAME      servidor
ftp       IN  CNAME      servidor
```

Assim, servidor. lab.2cta é a máquina 10.0.0.3. Os nomes www.lab.2cta e ftp.lab.2cta referem-se ao host servidor.lab.2cta, ou seja, 10.0.0.3.

OBS.: Nem sempre esse processo funciona, é recomendável colocar várias entradas IN A.

● IN MX – Mail Exchange
Determina quem serão os servidores SMTP (envio de mensagens) do dominio. As entradas devem conter números definindo a prioridade de utilização. Quanto menor o número, mais prioridade.
Exemplo:

```
IN   MX   1   servidor.lab.2cta.
IN   MX   2   servidor2.lab.2cta.
```

OBS.:Lembre-se de usar nome completo, tem que haver o ponto no final. Todos os nomes existntes deverão ter a respectiva linha IN A, definindo o IP.

- IN PTR – Point To Reverse

Utilizado apenas no mapeamento reverso. Já haviamos definido anteriormente que o mapeamento reverso seria obtido no aequivo named.boot, com os octetos que representam a rede, sem os zeros completadores, escritos de trás para frente, juntamente com a excessão in-addr.arpa. No arquivo db.10 colocaremos o que sobrou do IP, ou seja, os hosts, só que invertidos.
Exemplo:
Rede: 10.0.0.0
Máscara:255.0.0.0
Host a ser mapeado: 10.20.50.15
Entrada: 15.50.20

Um exemplo de entrada seria:

17.20.50 IN PTR teste.lab.2cta.
17.20.51

OBS.: No exemplo anterior, o arquivo /etc/bind/named.conf vai possuir na área de configuração da zona reversa para esta rede a seguinte linha:
“10.in-addr.arpa”.

Rede: 192.20.5.0
Máscara:255.255.255.0
Host a ser mapeado: 192.20.5.72
Entrada: 72

Um exemplo de entrada seria:

72 IN PTR teste.lab.2cta.

OBS.: No exemplo acima, o arquivo /etc/bind/named.conf vai possuir na área de configuração da zona reversa para esta rede a seguinte linha:
“192.20.5.in-addr.arpa”.

OBS.: Para cada entrada IN A existente em qualquer arquvo, deve possuir uma entrada PTR correspondente.

7.4. Arquivo /etc/bind/db.127

Responsável pela resolução reversa da rede 127.0.0.0. Segue a seguinte estrutura:

```
$TTL    1D
@       IN      SOA     servidordns.lab.2cta. araujo.lab.2cta. (
                        20051101    ; Serial
                        3H          ; Refresh
                        15M         ; Retry
                        1W          ; Expire
                        1D )        ; minimum

        IN      NS      servidordns.lab.2cta.
1.0.0   IN      PTR     localhost.
```

7.5. Arquivo /etc/bind/db.lab.2cta

Responsável pela resolução direta da rede 10.0.0.0. Segue a seguinte estrutura:

```
$TTL    1D
@       IN      SOA     servidordns.lab.2cta. araujo.lab.2cta. (
                        20051101    ; Serial
                        3H          ; Refresh
```

```
                    15M         ; Retry
                    1W          ; Expire
                    1D )        ; minimum
        IN      NS      servidordns.lab.2cta.

; SERVIDORES SMTP IN  MX
        IN      MX      2       webmail.lab.2cta.

; SERVIDOR DNS E MAIL – IN A
servidordns.lab.2cta.           IN   A  10.0.0.1
webmail.lab.2cta.       IN   A  10.0.0.2

; OUTRAS MÁQUINAS
ftp.lab.2cta.           IN   A  10.0.0.3
www.lab.2cta.           IN   A  10.0.0.3
protweb.lab.2cta.       IN   A  10.0.0.4
print.lab.2cta.                 IN   A  10.0.0.5
```

A técnica de construção desse arquivo se resume em citar o nome do servidor de nomes (IN NS), o seu IP (IN A), o nome do servidor mail (IN MX), o IP do servidor mail, caso ainda não tenha sido feito (IN A) e os aliases das máquinas de interesse (IN CNAME).

Ao invés de fazer várias entradas IN A para um mesmo endeeço IP, podemos fazer a entrada canônica com o registro CNAME.

Exemplo:

```
ftp.lab.2cta.           IN   A  10.0.04
www.lab.2cta.           IN   A  10.0.0.4
```

Poderia ser:

```
ftp.lab.2cta.           IN   A  10.0.0.4
www.lab.2cta.           IN   CNAME   ftp.lab.2cta.
```

7.6. Arquivo /etc/bind/db.10

Responsável pela resolução reversa da rede 10.0.0.0. Segue a seguinte estrutura:

```
$TTL    1D
@       IN      SOA     servidordns.lab.2cta. araujo.lab.2cta. (
                        20051101  ; Serial
                        3H      ; Refresh
                        15M     ; Retry
                        1W      ; Expire
                        1D )    ; minimum

        IN      NS      servidordns.lab.2cta.

        IN      MX      1       webmail.lab.2cta.

1.0.0.  IN   PTR        servidordns.lab.2cta.
2.0.0   IN   PTR        webmail.lab.2cta.
2.0.0   IN   PTR        ftp.lab.2cta.
3.0.0   IN   PTR        www.lab.2cta.
4.0.0   IN   PTR         protweb.lab.2cta
5.0.0   IN   PTR         print.lab.2cta
```

OBS.: Lembre-se que cada entrada IN A, no db.lab.2cta deve ter um IN PTR correspondente em db.10. Os registros IN CNAME devem ser desprezados.

8. COLOCANDO O DNS NO AR

Para que o DNS inicialize com o sistema, habilite-o no rcconf.
Para ativá-lo depois de configurá-lo, execute o comando:
#/etc/init.d/bind restart

9. TESTANDO O DNS

Para testar o DNS utilizar o comando nslookup, ele e comando dig vem no pacote dnsutils para isso deverá ser instalado, execute:

 # apt-get install dns utils

#nslookup www.lab.2cta

O resultado deverá ser:
Server: ip-do_servidor_dns
Address: ip-do_servidor_dns #53
Name:www.lab.2cta
Addess:10.0.0.3
Execute:
nslookup 10.0.0.3
O resultado deverá ser:
Server: ip_do_servidor_dns
Address: ip-do_servidor_dns #53
3.0.0.10. in.addr.arpa name = www.rede.com.br.
3.0.0.10. in.addr.arpa name = ftp.rede.com.br.

Caso haja qualquer resultado diferente, revise as configurações.
O comando nslookup pode ser substituiídos pelos comandos dig.
dig micro100.lab.2cta
dig -x 10.0.0.3

Caso tenha problemas, leia os logs /var/log/syslog e /var/daemon.log. Procure por bind.

10. CONFIGURANDO DNS SLAVE

Vamos tornar um DNS já existente escravo do nosso DNS, ou seja, o nosso DNS vai receber as informações do db.rede do DNS slave, para isso siga os seguintes passos:

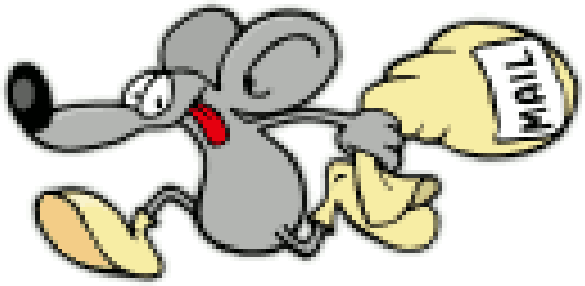
- Edite o arquivo /etc/bind/named.conf
- Insira no final do arquivo as linhas:

zone "nome_do_dominio_ do_DNS_Slave" {
type slave;
file "db.slave";
masters {IP_do_DNS_Slave};

- Restart o servidor DNS;
- Verifique o arquivo gerado /etc/bind/db.mestre.

SERVIDOR

DE CORREIO



POSTFIX

TÓPICOS

- Infra-estrutura de e-mail
- Instalação do POSTFIX
- Configuração Básica
- Configuração de envio e recebimento
- Bloqueio através do Cabeçalho e Corpo
- Restrição de envio por usuários

- Outros comandos
- Demonstração

1. INFRA-ESTRUTURA DE E-MAIL

Os principais elementos desse trajeto são:

- **MUA – Mail User Agent**: o programa que o usuário acessa para compor seu e-mail. Exemplo: Outlook, Netscape, Kmail, etc.

- **MTA – Mail Transport Agent**: recebe o e-mail do MUA e o envia a outros MTA, para que seja entregue ao destinatário. O principal MTA em UNIX é o sendmail, mas existem também o qmail , postfix e outros.

- **MDA – Mail Delivery Agent:** recebe o e-mail do MTA e o deposita na caixa de correio do usuário. O MDA default do Linux é o procmail, mas existem diversos outros.

- **MAA – Mail Access Agent**: permite ao MUA o acesso aos emails que estão na caixa de correio do usuário. Na prática, esta função é exercida pelos servidores POP3 e/ou IMAP.

Os principais protocolos em uso são:

- **SMTP – Simple Mail Transfer Protocol**: é o protocolo responsável pelo transporte dos emails entre MTAs, e do MUA ao MTA.

- **POP3 – Post-Office Protocol v3**: usado pelo MUA para acessar as mensagens armazenadas no servidor.

– Usa a porta 110/TCP ou 995/TCP (versão segura POP3S).
– Considera apenas uma pasta no servidor (INBOX).
- Por default descarrega as mensagens do servidor no cliente.

- **IMAP - Internet Message Access Protocol**: idem, porém mais versátil que o POP3

– Usa a porta 143/TCP ou 993/TCP (versão segura IMAPS)
– Pode manter diversas pastas no servidor, além da INBOX.
– Por default mantém as mensagens no servidor.
– Pode movimentar mensagens em ambas as direções
(entre pastas no cliente e no servidor)

2. INSTALAÇÃO DO POSTFIX

- VERIFICAR INSTALAÇÃO

```
# dpkg -l | grep postfix
```

- INSTALAÇÃO POSTFIX

```
# apt-get install postfix
```

- INSTALAÇÃO POP3

```
# apt-get install qpopper
```

- Verificar no arquivo /etc/inet.d.conf se a linha do pop3 está descomentada.

CONFIGURAÇÃO BÁSICA DO POSTFIX

Os arquivos de configuração do postfix encontram-se em: /etc/postfix

- **master.cf**
  Gerencia número de processos e serviços.
- **main.cf**
  Arquivo de configuração principal do postfix.

/etc/aliases
  Arquivo onde são armazenados os aliases.

- **/etc/mailname**
  Arquivo onde deve estar configurado com host.localdomain.
- **/var/spool/nome_usuário**
  Arquivo onde serão arquivadas as mensagens.

Ferramenta de configuração do Postfix

postconf	(Mostra todas os parâmetros)
postconf -n	(Mostra os parâmetros não padrão)
postconf -e parametro=valor	(Adiciona no final do arquivo main.cf)

3. Principais parâmetros do arquivo de configuração /etc/postfix/main.cf

queue_directory = /var/spool/postfix

- Diretório de Fila (spool de mensagens)

command_directory = /usr/sbin

- Diretório de Comandos

daemon_directory = /usr/lib/postfix

- Diretório de Daemon

mail_owner = postfix

- Usuário do Postfix

myhostname = servidor.meudominio.com.br

- Hostname do servidor

mydomain = meudominio.com.br

- Dominio do servidor

myorigin = $mydomain

- Qual o nome completo após o @ do e-mail

inet_interfaces = all

Qual interface responde pelo Postfix

mydestination = $myhostname, localhost.$mydomain, $mydomain

- Destinos válidos

mynetworks_style = subnet

Confia somente no host (Class / Subnet / Host)

mydomain = meudominio.com.br

- Domínio do servidor

unknown_local_recipient_reject_code = 500

- Resposta para usuários não encontrados

mynetworks = 127.0.0.0/8, 192.168.0.0/24

- Rede que serão liberadas para Relay

#mynetworks = $config_directory/mynetworks

alias_maps = hash:/etc/aliases

- Arquivos com alias

home_mailbox = Mailbox

Formato da Caixa de E-mail (Mailbox / Maildir)

mail_spool_directory = /var/spool/mail

- Diretório de Armazenamento de E-mails

mailbox_size_limit = 51200000

-Tamanho da caixa do usuário (50 Megas)

message_size_limit = 10240000

- Tamanho máximo da mensagem (10 Megas)

smtpd_banner = $myhostname - Mail Server

- Banner do servidor SMTP

debug_peer_level = 2

- Nível de debug

debugger_command =

- Parâmetros para o debug

PATH=/bin:/usr/bin:/usr/local/bin:/usr/X11R6/bin

xxgdb $daemon_directory/$process_name

$process_id & sleep 5

sendmail_path = /usr/sbin/sendmail

- Caminho do Sendmail

newaliases_path = /usr/bin/newaliases

- Caminho do Newaliases

mailq_path = /usr/bin/mailq

- Caminho do Mailq

setgid_group = postdrop

- Grupo do Postfix

manpage_directory = /usr/local/man

- Diretório do Manual

4. CONFIGURAÇÃO DE ENVIO E RECEBIMENTO

Esses parâmetros visam melhorar a segurança do servidor e combater o Spam:

smtpd_recipient_limit = 100
- Número máximo de destinatários no mesmo e-mail

strict_rfc821_envelopes = yes
- Respeita RFC 821 - MAIL FROM e RCPT TO

smtpd_helo_required = yes
- Ativo checagem de helo

disable_vrfy_command = yes
- Desabilita VRFY

maps_rbl_domains = relays.ordb.org, list.dsbl.org,
dun.dnsrbl.net, spam.dnsrbl.net
- Listas de RBL
Obs.: Utilizar com cuidado as listas, pois algumas bloqueiam e-mails do Brasil. Mais informações em: http://www.dnsstuff.com

smtpd_client_restrictions =
- Restricão do cliente - Após o aceite da conexao SMTP
Checa conteúdo do CLIENT_ACCESS
check_client_access hash:/etc/postfix/client_access,
Permite "mynetwork"
permit_mynetworks,
Permite conteudo do ACCESS
hash:/etc/postfix/access,
Quando não há entrada PTR do IP
reject_unknown_client,
Bloqueio comando para forçar entrega
reject_unauth_pipelining,
Bloqueia IP's listados em RBL
reject_rbl_client maps_rbl_domains

smtpd_helo_restrictions =
- Restricão durante comando HELO/EHLO
Permite "mynetwork"
permit_mynetworks,
Quando não é informado o hostname
reject_invalid_hostname,
Quando não existe entrada DNS A ou MX
reject_unknown_hostname,
Quando o hostname não apresenta hostname válido
reject_non_fqdn_hostname,
Bloqueio comando para forçar entrega
reject_unauth_pipelining,
Bloqueia IP's listados em RBL

reject_rbl_client maps_rbl_domains

smtpd_sender_restrictions =
- Restricão aplicada no MAIL FROM
Permite "mynetwork"
permit_mynetworks,
Permite conteudo do ACCESS
check_sender_access hash:/etc/postfix/access,
Bloqueio quando não existe entrada DNS A ou MX
reject_unknown_sender_domain,
Quando o hostname não apresenta hostname válido
reject_non_fqdn_sender,
Bloqueio comando para forçar entrega.
reject_unauth_pipelining

smtpd_recipient_restrictions =
- Restricão aplicada no RCPT TO
Permite "mynetwork"
permit_mynetworks,
Permite conteúdo do ACCESS
check_sender_access hash:/etc/postfix/access,
Bloqueia quando não existe entrada DNS A ou MX
reject_unknown_recipient_domain,
Quando o hostname não apresenta hostname válido
reject_non_fqdn_recipient,
Bloqueio comando para forçar entrega
reject_unauth_pipelining

Arquivo /etc/postfix/access
joaozinho@123.com.br E-MAIL REJEITADO
123.com.br DOMINIO REJEITADO
/^postmaster@/ OK
/^abuse@/ OK

Arquivo /etc/postfix/client_access
200.200.200.200 RELAY
100.100.100 554 SPAMMER NETWORK
150.100.100.100 554 SPAMMER HOST
dsl.telesp.net.br 554 SPAMMER NETWORK

5. BLOQUEIO ATRAVÉS DO CABEÇALHO E CORPO
- Bloqueio por Assunto
Adicione a linha abaixo no main.cf
header_checks = pcre:/etc/postfix/header_checks
mime_header_checks = $header_checks
nested_header_checks = $header_checks

Conteúdo do /etc/postfix/header_checks
/^Subject: Trabalhe em casa/ REJECT SPAMMER
/^To: joao@trabalheemcasa.com.br/ REJECT
*/^Subject:.*V.agr.\?*./ REJECT Email rejeitado*
*/^Subject:.*FIQUE RICO.*./ REJECT Email rejeitado*

/^Content-(Type|Disposition):.(file)?name=.*\.(com|lnk|*
bat|scr|chm|hlp|hta|reg|shs|vbe|vbs|wsf|wsh|pif)/
REJECT Email rejeitado, devido a um arquivo .${3} em
anexo

Bloqueio por Conteúdo
Adicione a linha abaixo no main.cf
body_checks = pcre:/etc/postfix/body_checks
 # Verifica os 50 K inicais
body_checks_size_limit = 51200
Conteúdo do /etc/postfix/body_checks
/^Content-(Type|Disposition):.(file)?name=.*\.(com|lnk|*
bat|scr|chm|hlp|hta|reg|shs|vbe|vbs|wsf|wsh|pif|exe)/
REJECT Email rejeitado, devido a um arquivo .$ {3} em
anexo
*/^.*decidaservencedor.kit.net*/ REJECT Spammer.*
/^RSLxwtYBDB6FCv8ybBcS0zp9VU5of3K4BXuwyehTM
0RI9IrSjVuwP94xfn0wgOjouKWzGXHVk3qg$/ DISCARD
VIRUS(sobig.f)
/^(UEsDBAoAAAAAA(.....KJx\+eAFgAAABYAA|...Nz|
K4)|ApIAUCZKAEAD\/bJpmiwQBPQl6AEAS
85pmm7ZH8gqwAO4sKimaZqmoJiQiICapmmaeHBoYFh
QzWCf)/ DISCARD VIRUS (W32/Mydoom@MM)

Algumas opções para o arquivo /etc/postfix/body_checks
REJECT [texto opcional]
- Rejeita a mensagem e retorno erro para o remetente
OK
- Aceita a mensagem
IGNORE
- Ignora a mensagem sem reportar mensagem para o remetente
DISCARD [texto opcional]
Ignora a mensagem

6. RESTRIÇÃO DE ENVIO POR USUÁRIO
As vezes é necessário bloquear o envio de e-mail de determinados usuários e para isso fazemos:
/etc/postfix/main.cf:
smtpd_recipient_restrictions = hash:/etc/postfix/usuarios_restritos

... outros parâmetros ...

/etc/postfix/main.cf:
smtpd_restriction_classes = dominios_restritos
dominios_restritos =
check_sender_access hash:/etc/postfix/insiders, reject
/etc/postfix/usuarios_restritos:
usuario1@meudominio.com.br dominios_restritos
usuario2@meudominio.com.br dominios_restritos
/etc/postfix/dominios_restritos:
dominio1.com.br OK

dominio2.com.br OK
dominio3.com.br OK

Depois de criar os arquivos é necessário rodar os comandos:
postmap /etc/postfix/usuarios_restritos
postmap /etc/postfix/dominios_restritos
postfix reload

7. Outros comandos
always_bcc = email@meudominio.com.br
- Todos os e-mails que chegam irão para e-mail abaixo

bounce_size_limit = 50000
Tamanho da mensagem de erro
header_size_limit = 102400
- Tamanho máximo do HEADER aceito

smtp_destination_concurrency_limit = 20
- Entrega de e-mails para mesmo destino

default_destination_concurrency_limit = 20
- Entrega de e-mails para mesmo destino - remoto

default_destination_recipient_limit = 50
- Entrega de e-mails para mesmo destino - local

fast_flush_refresh_time = 12h
- Tempo de reenvio de mensagem em fila

fast_flush_purge_time = 1d
- Tempo de deleção de mensagem em fila

maximal_queue_lifetime = 240m
- Tempo de mensagem em fila

As variavéis de tempo válidas são:
s -> segundos (seconds)
m -> minutos (minutes)
h -> horas (hours)
d -> dias (days)
w -> semanas (week)

8. DEMONSTRAÇÃO - CONFIGURAÇÃO DO SERVIDOR DE CORREIO
- Abrir o arquivo /etc/postfix/main.cf

smtpd_banner = $myhostname ESMTP $mail_name (Debian/GNU)
- Mensagem de resposta do servidor.
- Mudar para: smtp_banner = BEM VINDO AO SERVIDOR DE CORREIO $mail_name

 # appending .domain is the MUA's job.
append_dot_mydomain = no

- Caso o servidor envie mensagens como: usuario@servidor.lab.2cta.servidor.lab.teste, deixe no.
- Testar mudando para “yes” e enviar usando mail from:araujo@gmail, para ver a diferença.

 # Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h
- Descomentando a linha acima irá gerar um atraso no envio do correio

myhostname = servidor.lab.2cta
- nome do servidor e dominio

alias_maps = hash:/etc/aliases
- arquivo onde são armazenadps os aliases.

alias_database = hash:/etc/aliases
myorigin = /etc/mailname
- arquivo onde deve estar configurado com host.localdomain

mydestination = localhost.localdomain, localhost.localdomain, , localhost,servidor.lab.2cta
- Inserir o nome do servidor.

mynetworks =10.1.12.0/24, 127.0.0.0/8
- Considerando uma rede 10.1.12.0 com mascara de rede 255.255.252.0, o valor apos a “/” é a quantidade de “1” dos octetos da máscara de rede vezes a quantidade de octetos que corresponde à rede.

mailbox_size_limit = 0
- Tamanho da caixa de mensagem em MB (caixa com 10 MB - 10000 x 1024=10240000)
recipient_delimiter = +

inet_interfaces = all

- Interface de rede que irá responder o correio

CAIXA DE CORREIO

As mensagens para os usuários serão arquivadas no arquivo: /var/spool/nome_usuario

CRIAÇÃO DE ALIASES

A criação de aliases, irá nos auxiliar quando queremos que um usuário receba mensagens com outro nome. O arquivo responsavel por criar aliases para um usuário é o /etc/aliases; a sintaxe deste aruivo deve seguir a seguinte estrutura:

ALIAS	USUÁRIO_EXISTENTE
webmaster:	root
www:	root
ftp:	root

O exemplo acima fará com que as mensagens enviadas para

webmaster@servidor.lab.2cta, www@servidor.lab.2cta, ftp@servidor.lab.2cta, serão recebidas pelo usuário root@servidor.lab.2cta.

Toda vez que este arquivo for modificado deverá ser executado o comando abaixo:

```
# newaliases          ou
# postalias
```

TESTE DE ENVIO E RECEBIMENTO DE MENSAGENS PELO POSTFIX

Para testar o funcionamento do servidor, em um terminal, acessar o servidor de e-mail através telnet:

```
# telnet ip_do_servidor 25
ehlo server          (teste de resposta do servidor, deverá retornar o nome do servidor)
quit                 (desconectar do telnet)
```

Para testar envio de mensagem, em um terminal, acessar o servidor de e-mail através telnet:

```
# telnet ip_do_servidor 25              ou
# telnet nome_do_servidor 25   (tem que estar no DNS)
mail from: usuario@qualquer.provedor      (e-mail do remetente)
rcpt to: aluno@servidor.lab.2cta          (e-mail do destinatário)
data                                      (inicio da mensagem)
Teste de mensagem                         (conteúdo da mensagem)
.                                    (ponto final, indicando o final da mensagem)
quit                                 (desconectar do telnet)
```

Para testar o recebimento da mensagem, em um terminal, acessar o servidor de e-mail através telnet:

```
# telnet ip_do_servidor 110             ou
# telnet nome_do_servidor 110 (tem que estar no DNS)
list                          (lista as mensagens)
retr x                        (mostra a mensagem número x)
dele x                        (apaga a mensagem número x)
quit                          (desconectar do telnet)
```

Testar envio e recebimento de mensagens pelo mozilla_thunderbird

apt-get install mozilla-thunderbird
apt-get install mozilla-thunderbird-locale-pt-br

- Configurar o thunderbird

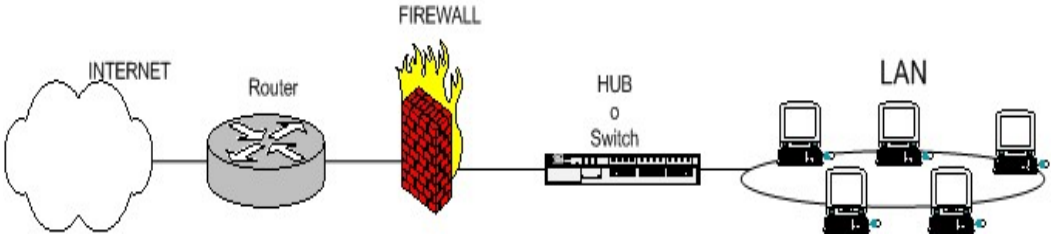
SERVIDOR IPTABLES

FIREWALL – IPTABLES

SUMARIO

1. INTRODUÇÃO

Um firewall é um dispositivo que tem como função primária filtrar o tráfico de dados entre redes, ou entre um computador e uma rede.



Estamos concentrados em proteger as unidades da internet e da EBNET. Os vírus podem vir pelas duas redes.

Há duas maneiras de se implementar um Firewall:

Política default de aceitar: todo o tráfico que passa pelo firewall é aceito, apenas fica bloqueado no firewall o que for explicitado nele.

Política default de rejeitar: todo o tráfico que passa pelo firewall é rejeitado, apenas passa pelo firewall o que for explicitado nele.

O que é o iptables?

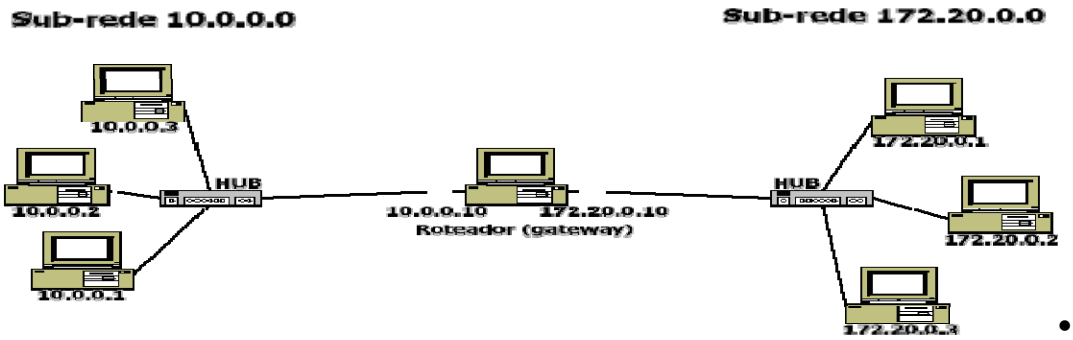
É um sistema de firewall vinculado ao kernel do Linux. Funciona diferente dos serviços.

2. IPFORWARD

O IP FORWARD é um tipo de roteamento. É estabelecido quando colocamos uma máquina entre duas ou mais sub-redes diferentes e há a livre passagem de pacotes entre elas, quando necessário. É importante ressaltar que o roteamento só irá funcionar quando for feito entre SUB-REDES DIFERENTES. Não se pode colocar um roteador entre duas sub-redes iguais.

O roteamento também é útil para diminuir o tráfego na rede como um todo, pois só deixa o pacote mudar de sub-rede se isso for realmente necessário.

Para fazer o IP FORWARD, o micro roteador deve possuir uma placa de rede em cada sub-rede. Também deveremos informar em cada máquina quem será o micro responsável pelo roteamento. O nome técnico desse micro é gateway.



- Inserir um micro com duas placas de rede entre as duas sub-redes, configurando cada placa de acordo com cada sub-rede;
- Pefinir, em cada máquina, de cada sub-rede, quem é o seu gateway;
- Ativar o IP FORWARD via kernel.

O estabelecimento de IP FORWARD entre mais de duas sub-redes segue o mesmo princípio, bastando acrescer quantas placas de rede forem necessárias no gateway. Também é possível utilizar placas de fax-modem sozinhas ou em conjunto com placas de rede.

- Definindo o gateway

Tarefa feita na configuração de rede do Windows.

- Ativando o roteamento via kernel

O roteamento via kernel será ativado com o comando:
#echo 1 > /proc/sys/net/ipv4/ip_forward

Esse roteamento será perdido se a rede (e, em conseqüência, a máquina) for reinicializada (#/etc/rc.d/init.d/network restart).

Poderíamos inserir a regra no fim do arquivo /etc/rc.d/rc.local, para que a mesma seja ativada a cada reinicialização do sistema. No entanto, um reinício da rede mataria o roteamento novamente.

Uma forma de deixar a regra de roteamento permanentemente ativada, resistindo a qualquer tipo de reinicialização, seria a alteração do arquivo /etc/sysctl.conf:
net.ipv4.ip_forward = 1

Vamos considerar firewall como sendo o efetivo controle do roteamento. Então, vamos configurar uma máquina capaz de tomar decisões em relação ao tráfego de rede. Podemos citar como firewall, as máquinas que executam os seguintes serviços:

- roteamento controlado por regras de análise de cabeçalho IP (filtro de pacotes);
- roteamento mascarado controlado por regras de análise de cabeçalho IP (filtro de pacotes mascarado ou firewall de mascaramento);
- roteamento controlado por regras de análise de conteúdo de pacotes (filtro de

conteúdo);
- roteamento mascarado controlado por regras de análise de URL (proxy).

Analisaremos o filtro de pacotes existente no Linux. Ele verifica apenas o cabeçalho de cada pacote, definindo o que ocorrerá com tais pacotes. Basicamente, só entende endereço IP, máscara de sub-rede, portas e tipos de protocolos. Não analisa o conteúdo do pacote e nem trata as "palavras" da URL.
Todas as expressões firewall, quando utilizadas daqui por diante, referir-se-ão ao filtro de pacotes do Linux.

A filtragem de pacotes é uma atividade interna do kernel.

O FILTRO DE PACOTES do Linux funciona mediante regras estabelecidas. Todos os pacotes entram no kernel para serem analisados. As CHAINS (correntes) são as situações possíveis dentro do kernel. Quando um pacote entra no kernel, este verifica o destino do pacote e decide qual chain irá tratar do pacote. Isso se chama roteamento interno. Os tipos de chains irão depender da tabela que estaremos utilizando no momento. Existem 3 tabelas possíveis:

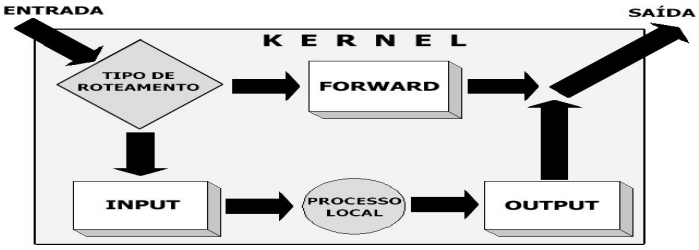
- filter: é a tabela default. Quando não especificarmos a tabela, a filter será utilizada. Refere-se às atividades normais de tráfego de dados, sem a ocorrência de NAT. Admite as chains INPUT, OUTPUT e FORWARD.
- nat: utilizada quando há NAT. Exemplo: passagem de dados de uma rede privada para a Internet. Admite as chains PREROUTING, OUTPUT e POSTROUTING.
- mangle: basicamente, trabalha com marcação de pacotes e QoS. Neste tutorial, não trataremos dessa tabela.

O iptables, diferentemente dos filtros anteriores, é um firewall stateful, ou seja, trabalha com os estados das conexões. Os anteriores eram stateless.

3. TABELAS

Tabela Filter

Vejamos o funcionamento da tabela filter (default) e as suas respectivas chains:



São três, as possíveis chains:

- **INPUT**: utilizada quando o destino final é a própria máquina firewall;
- **OUTPUT**: qualquer pacote gerado na máquina firewall e que deva sair para a rede será tratado pela chain OUTPUT;

FORWARD: qualquer pacote que atravessa o firewall, oriundo de uma máquina e direcionado a outra, será tratado pela chain FORWARD.

Regras de firewall

As regras (rules) de firewall, geralmente, são compostas assim:
#iptables [-t tabela] [opção] [chain] [dados] -j [ação]
Exemplo:
#iptables -A FORWARD -d 192.168.1.1 -j DROP

A linha acima determina que todos os pacotes destinados à máquina 192.168.1.1 devem ser descartados.

No caso:
tabela: filter (é a default)
opção: -A

chain: FORWARD
dados: -d 192.168.1.1
ação: DROP

Existem outras possibilidades que fogem à sintaxe mostrada anteriormente. É o caso do comando #iptables -L, que mostra as regras em vigor.

Análise de regras com a tabela filter

Opções

As principais opções são:

-P --> Policy (política). Altera a política da chain. A política inicial de cada chain é ACCEPT. Isso faz com que o firewall, inicialmente, aceite qualquer INPUT, OUTPUT ou FORWARD. A política pode ser alterada para DROP, que irá negar o serviço da chain, até que uma opção -A entre em vigor. O -P não aceita REJECT ou LOG. Exemplos:
#iptables -P FORWARD DROP
#iptables -P INPUT ACCEPT

-A --> Append (anexar). Acresce uma nova regra à chain. Tem prioridade sobre o -P. Geralmente, como buscamos segurança máxima, colocamos todas as chains em política DROP, com o -P e, depois, abrimos o que é necessário com o -A. Exemplos:
#iptables -A OUTPUT -d 172.20.5.10 -j ACCEPT
#iptables -A FORWARD -s 10.0.0.1 -j DROP
#iptables -A FORWARD -d www.chat.com.br -j DROP

-D --> Delete (apagar). Apaga uma regra. A regra deve ser escrita novamente, trocando-se a opção para -D. Exemplos:
Para apagar as regras anteriores, usa-se:
#iptables -D OUTPUT -d 172.20.5.10 -j ACCEPT
#iptables -D FORWARD -s 10.0.0.1 -j DROP
#iptables -D FORWARD -d www.chat.com.br -j DROP
Também é possível apagar a regra pelo seu número de ordem. Pode-se utilizar o -L para verificar o número de ordem. Verificado esse número, basta citar a chain e o número de ordem. Exemplo:
#iptables -D FORWARD 4
Isso deleta a regra número 4 de forward.

-L --> List (listar). Lista as regras existentes. Exemplos:
#iptables -L
#iptables -L FORWARD

-F --> Flush (esvaziar). Remove todas as regras existentes. No entanto, não altera a política (-P). Exemplos:
#iptables -F
#iptables -F FORWARD

Chains

As chains já são conhecidas:

INPUT --> Refere-se a todos os pacotes destinados à máquina firewall.
OUTPUT --> Refere-se a todos os pacotes gerados na máquina firewall.
FORWARD --> Refere-se a todos os pacotes oriundos de uma máquina e destinados a outra. São pacotes que atravessam a máquina firewall, mas não são destinados a ela.

Dados

Os elementos mais comuns para se gerar dados são os seguintes:

-**s** --> Source (origem). Estabelece a origem do pacote. Geralmente é uma combinação do endereço IP com a máscara de sub-rede, separados por uma barra. Exemplo:
-s 172.20.0.0/255.255.0.0

No caso, vimos a sub-rede 172.20.0.0. Para hosts, a máscara sempre será 255.255.255.255. Exemplo:

-s 172.20.5.10/255.255.255.255
Agora vimos o host 172.20.5.10. Ainda no caso de hosts, a máscara pode ser omitida. Caso iss
-s 172.20.5.10

Isso corresponde ao host 172.20.5.10. Há um recurso para simplificar a utilização da máscara de sub-rede. Basta utilizar a quantidade de bits 1 existentes na máscara. Assim, a máscara 255.255.0.0 vira 16. A utilização fica assim:
-s 172.20.0.0/16
Outra possibilidade é a designação de hosts pelo nome. Exemplo:
-s www.chat.com.br
Para especificar qualquer origem, utilize a rota default, ou seja, 0.0.0.0/0.0.0.0, também admitindo 0/0.

d --> Destination (destino). Estabelece o destino do pacote. Funciona exatamente como o -s, incluindo a sintaxe.
-p --> Protocol (protocolo). Especifica o protocolo a ser filtrado. O protocolo IP pode ser especificado pelo seu número (vide /etc/protocols) ou pelo nome. Os protocolos mais utilizados são udp, tcp e icmp.
-i --> In-Interface (interface de entrada). Especifica a interface de entrada. As interfaces existentes podem ser vistas com o comando #ifconfig. O -i não pode ser utilizado com a chain OUTPUT. Exemplo:
-i ppp0
O sinal + pode ser utilizado para simbolizar várias interfaces. Exemplo:
-i eth+
eth+ refere-se à eth0, eth1, eth2 etc
-o --> Out-Interface (interface de saída). Especifica a interface de saída. Similar a -i, inclusive nas flexibilidades. O -o não pode ser utilizado com a chain INPUT.
! --> Exclusão. Utilizado com -s, -d, -p, -i, -o e outros, para excluir o argumento. Exemplo:
-s ! 10.0.0.1
Isso refere-se a qualquer endereço de entrada, exceto o 10.0.0.1.
-p ! tcp
Todos os protocolos, exceto o TCP.
--sport --> Source Port. Porta de origem. Só funciona com as opções -p udp e -p tcp. Exemplo:
-p tcp --sport 80
Refere-se à porta 80 sobre protocolo TCP.
--dport --> Destination Port. Porta de destino. Só funciona com as opções -p udp e -p tcp. Similar a --sport.

Ações
As principais ações são:
ACCEPT --> Aceitar. Permite a passagem do pacote.
DROP --> Abandonar. Não permite a passagem do pacote, descartando-o. Não avisa a origem sobre o ocorrido.
REJECT --> Igual ao DROP, mas avisa a origem sobre o ocorrido (envia pacote icmp unreachable).
LOG --> Cria um log referente à regra, em /var/log/messages. Usar antes de outras ações.

Exemplos comentados de regras de firewall (tabela filter)

#iptables -L
Lista todas as regras existentes.

#iptables -F

Apaga todas as regras sem alterar a política.

#iptables -P FORWARD DROP
Estabelece uma política de proibição inicial de passagem de pacotes entre sub-redes.

#iptables -A FORWARD -j DROP
Todos os pacotes oriundos de qualquer sub-rede e destinados a qualquer sub-rede deverão ser descartados.

#iptables -A FORWARD -j ACCEPT
Todos os pacotes oriundos de qualquer sub-rede e destinados a qualquer sub-rede deverão ser aceitos.

#iptables -A FORWARD -s 10.0.0.0/8 -d www.chat.com.br -j DROP
Os pacotes oriundos da sub-rede 10.0.0.0 (máscara 255.0.0.0) e destinados aos hosts cujos endereços IP respondem pelo nome www.chat.com.br deverão ser descartados. Note que se a máquina possuir domínios virtuais, todos esses serão bloqueados.

#iptables -A FORWARD -s 10.0.0.0/8 -d www.chat.com.br -j REJECT
Os pacotes oriundos da sub-rede 10.0.0.0 (máscara 255.0.0.0) e destinados aos hosts cujos endereços IP respondem pelo nome www.chat.com.br deverão ser descartados. Deverá ser enviado um ICMP avisando à origem.

#iptables -A FORWARD -d www.chat.com.br -j DROP
Os pacotes oriundos de qualquer lugar e destinados aos hosts cujos endereços IP respondem pelo nome www.chat.com.br deverão ser descartados.

#iptables -A FORWARD -d 10.0.0.0/8 -s www.chat.com.br -j DROP
Os pacotes destinados à sub-rede 10.0.0.0 (máscara 255.0.0.0) e oriundos aos hosts cujos endereços IP respondem pelo nome www.chat.com.br deverão ser descartados.

#iptables -A FORWARD -s www.chat.com.br -j DROP
Os pacotes oriundos aos hosts cujos endereços IP respondem pelo nome www.chat.com.br e destinados a qualquer lugar deverão ser descartados.

#iptables -A FORWARD -s 200.221.20.0/24 -j DROP
Os pacotes oriundos da sub-rede 200.221.20.0 (máscara 255.255.255.0) e destinados a qualquer lugar deverão ser descartados.

#iptables -A FORWARD -s 10.0.0.5 -p icmp -j DROP
Os pacotes icmp oriundos do host 10.0.0.5 e destinados a qualquer lugar deverão ser descartados.

#iptables -A FORWARD -i eth0 -j ACCEPT
Os pacotes que entrarem pela interface eth0 serão aceitos.

#iptables -A FORWARD -i ! eth0 -j ACCEPT
Os pacotes que entrarem por qualquer interface, exceto a eth0, serão aceitos.

#iptables -A FORWARD -s 10.0.0.5 -p tcp --sport 80 -j LOG
O tráfego de pacotes TCP oriundos da porta 80 do host 10.0.0.5 e destinados a qualquer lugar deverá ser gravado em log. No caso, /var/log/messages.

#iptables -A FORWARD -p tcp --dport 25 -j ACCEPT
Os pacotes TCP destinados à porta 25 de qualquer host deverão ser aceitos.

Observações importantes

Impasses e ordem de processamento

As regras serão interpretadas na ordem em que aparecerem. Sempre que um pacote se adequar a uma regra, tal regra processará o pacote e a sequência iptables será finalizada naquele instante, sem que as regras seguintes atuem. Isso não se aplicará às regras terminadas com -j LOG. Nesse caso, a regra com -j LOG irá atuar, se for o caso, e permitirá o prosseguimento da seqüencia.

Conclusão: se houver impasse entre regras, sempre valerá a primeira.
#iptables -A FORWARD -p icmp -j DROP
#iptables -A FORWARD -p icmp -j ACCEPT
Valerá:
#iptables -A FORWARD -p icmp -j DROP
Já entre as regras:
#iptables -A FORWARD -p icmp -j ACCEPT
#iptables -A FORWARD -p icmp -j DROP
Valerá:
#iptables -A FORWARD -p icmp -j ACCEPT

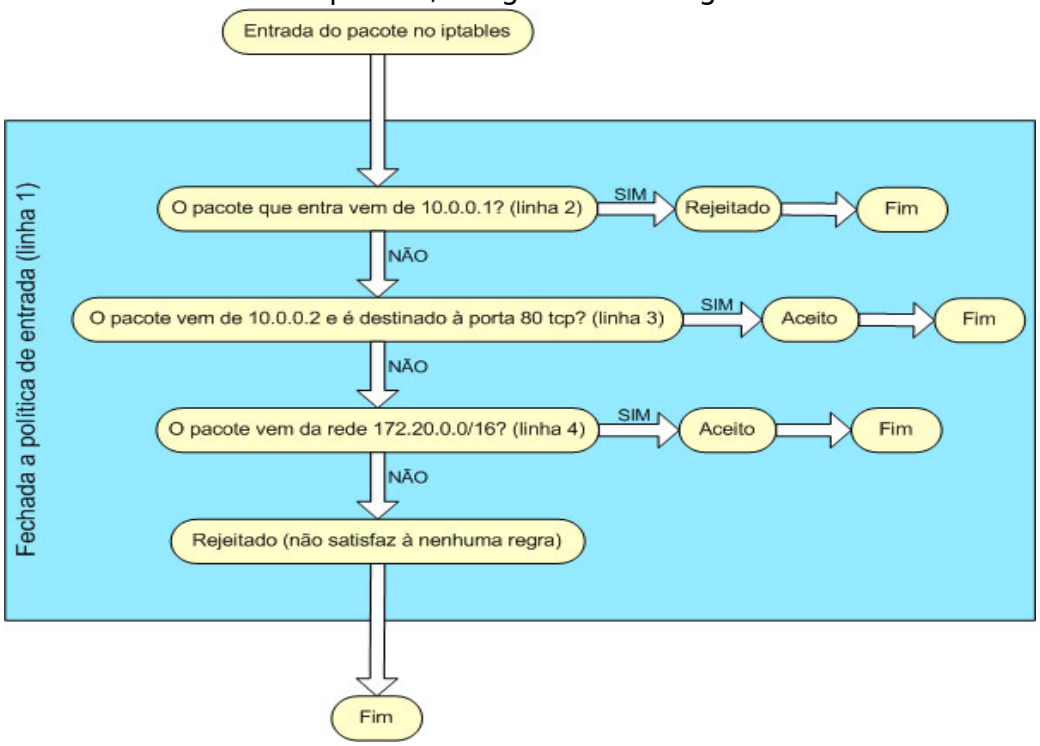
Em resumo:

ACCEPT --> Pára de processar regras para o pacote atual;
DROP --> Pára de processar regras para o pacote atual;
REJECT --> Pára de processar regras para o pacote atual;
LOG --> Continua a processar regras para o pacote atual;

Vamos ver um exemplo. As regras serão as seguintes:

```
iptables -P INPUT DROP
iptables -A INPUT -s 10.0.0.1 -j DROP
iptables -A INPUT -s 10.0.0.2 -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -s 172.20.0.0/16 -j ACCEPT
```

Analisando-se o fluxo de um pacote, chegamos ao diagrama:



O Retorno

Ao se fazer determinadas regras, devemos prever o retorno. Assim, digamos que exista a seguinte situação:
#iptables -P FORWARD DROP
#iptables -A FORWARD -s 10.0.0.0/8 -d 172.20.0.0/16 -j ACCEPT

Com as regras anteriores, fechamos todo o FORWARD e depois abrimos da sub-rede 10.0.0.0 para a sub-rede 172.20.0.0. No entanto, não tornamos possível a resposta da sub-rede 172.20.0.0 para a sub-rede 10.0.0.0. O correto, então, seria:
#iptables -P FORWARD DROP
#iptables -A FORWARD -s 10.0.0.0/8 -d 172.20.0.0/16 -j ACCEPT
#iptables -A FORWARD -d 10.0.0.0/8 -s 172.20.0.0/16 -j ACCEPT

IP FORWARD

Caso haja o envolvimento de mais de uma sub-rede, será necessário que o IP FORWARD seja ativado para que o iptables funcione corretamente. O IP FORWARD, via kernel, pode ser ativado pelo comando:
#echo 1 > /proc/sys/net/ipv4/ip_forward

Cabe lembrar que a reinicialização do daemon de rede fará com que o roteamento seja perdido. Uma forma de deixar a regra de roteamento permanentemente ativada, resistindo a qualquer tipo de reinicialização, seria a alteração do arquivo /etc/sysctl.conf:
net.ipv4.ip_forward = 1

Extensões

As extensões permitem filtragens especiais, principalmente contra ataques de hackers. Os exemplos abaixo mostram como controlar os pings que atravessam o firewall:
Contra Ping
#iptables -A FORWARD -p icmp --icmp-type echo-request -j DROP
Contra Ping of Death
#iptables -A FORWARD -p icmp --icmp-type echo-request -m limit --limit 1/s -j ACCEPT#iptables -A FORWARD -p icmp --icmp-type echo-request -j DROP

É lógico que as regras anteriores podem ser utilizadas com INPUT.
Mais proteção

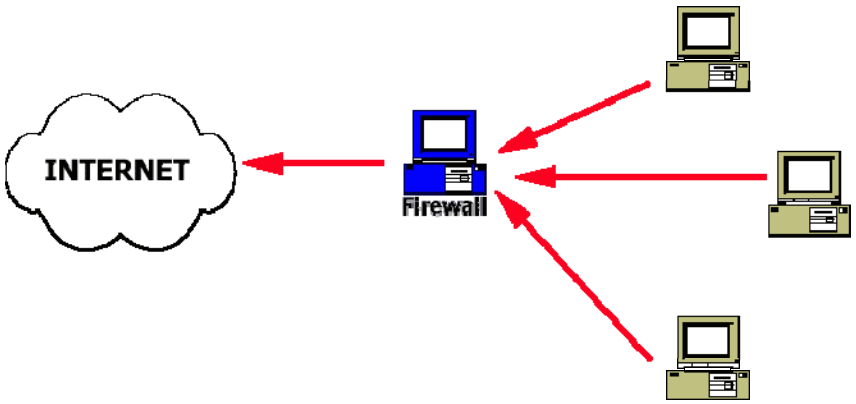
Existe, ainda, uma regra muito importante que pode ser utilizada como segurança. É a proteção contra pacotes danificados, suspeitos ou mal formados.
#iptables -A FORWARD -m unclean -j DROP
Também pode ser utilizado com INPUT.

Network Address Translator - NAT (tabela nat)

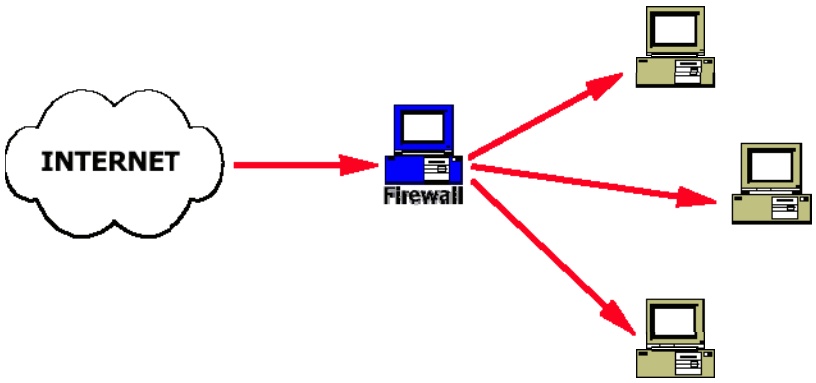
Existem vários recursos que utilizam NAT. Os mais conhecidos são:
Mascaramento (masquerading)
Redirecionamento de portas (port forwarding ou PAT)
Redirecionamento de servidores (forwarding)
Proxy transparente (transparent proxy)
Balanceamento de carga (load balance)

Mascaramento

O mascaramento é uma forma de fazer NAT (Network Address Translation). Com isso, é possível fazer uma rede privada navegar na Internet. A rede solicita os dados para a máquina que faz o mascaramento. Essa busca tais dados na Internet...



...e os entrega aos solicitantes:



O único endereço IP que irá circular na Internet será o do firewall.

O mascaramento também possui um esquema de funcionamento. Como haverá trocas de endereços, deveremos utilizar a tabela NAT para fazer isso.

Redirecionamento de portas

O redirecionamento de portas ocorre quando desejamos alterar a porta de destino de uma requisição. Exemplo: tudo que for destinado à porta 23 de qualquer máquina, quando passar pela máquina firewall, será redirecionado para a porta 10000 de outro servidor.

Redirecionamento de servidores

Todos os pacotes destinados a um servidor serão redirecionados para outro servidor.

Proxy transparente

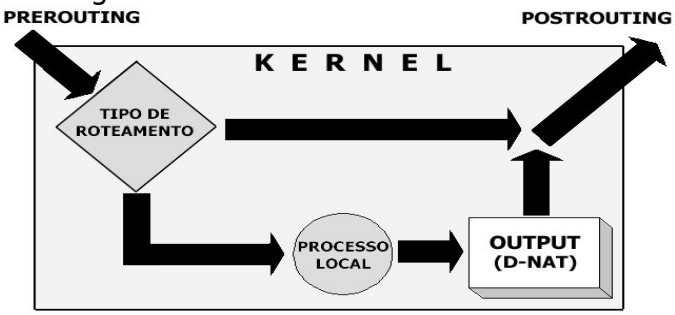
É a técnica que força o uso de um servidor proxy na rede.

Balanceamento de carga

O balanceamento de carga (load balance) é uma técnica utilizada para distribuir carga entre servidores sincronizados. O load balance é o ato de distribuir os clientes aos servidores mais desocupados. Esse trabalho também pode ser feito por servidores DNS.

A tabela NAT

A tabela NAT funciona da seguinte forma:



O NAT é dividido em:
SNAT: aplica-se quando desejamos alterar o endereço de origem do pacote. Somente a chain POSTROUTING faz SNAT. O mascaramento é um exemplo de SNAT.
DNAT: aplica-se quando desejamos alterar o endereço de destino do pacote. As chains PREROUTING e OUTPUT fazem DNAT. O redirecionamento de porta, o redirecionamento de servidor, o load balance e o proxy transparente são exemplos de DNAT.

Regras de NAT

Para fazer o mascaramento, deveremos, antes, carregar o módulo de NAT:
#modprobe iptable_nat
As regras mais utilizadas, além da maioria dos recursos descritos para uso com a tabela filter, contêm o seguinte:

Chains
Existem as seguintes chains:
PREROUTING: utilizada para analisar pacotes que estão entrando no kernel para sofrerem NAT. O PREROUTING pode fazer ações de NAT com o endereço de destino do pacote. Isso é conhecido como DNAT (Destination NAT);
POSTROUTING: utilizada para analisar pacotes que estão saindo do kernel, após sofrerem NAT. O POSTROUTING pode fazer ações de NAT com o endereço de origem do pacote. Isso é conhecido como SNAT (Source NAT);
OUTPUT: utilizada para analisar pacotes que são gerados na própria máquina e que irão sofrer NAT. O OUTPUT pode fazer ações de NAT com o endereço de destino do pacote. Também é DNAT.

Opções
-A --> Append (anexar).
-D --> Deletar.
Dados
-t --> Table (tabela). Estabelece a tabela a ser utilizada. A tabela default, por omissão, é filter. Para o mascaramento ou NAT será nat. Exemplo:
#iptables -t nat -A ...
--to --> utilizado para definir IP e porta de destino, após um DNAT, ou de origem, após um SNAT. Deve ser utilizado após uma ação (-j ação). Assim:
-j DNAT --to 10.0.0.2
-j DNAT --to 10.0.0.2:80
-j SNAT --to 172.20.0.2
--dport --> assim como -d define um host de destino, --dport define uma porta de destino. Deve ser utilizado antes de uma ação (-j ação). Antes de --dport, deve ser especificado um protocolo (-p). Exemplo:
-d 172.20.0.1 -p tcp --dport 80 -j DNAT --to 10.0.0.2
--sport --> assim como -s define um host de origem, --sport define uma porta de origem. Deve ser utilizado antes de uma ação (-j ação).
--to-port --> define uma porta de destino, após um REDIRECT.
Obs: A maioria dos dados básicos apresentados para a tabela filter continuam valendo. Exemplo: -p servirá para definir um protocolo de rede; -d define um host de destino.

Ações
SNAT --> Utilizado com POSTROUTING para fazer ações de mascaramento da origem.
DNAT --> Utilizado com PREROUTING e OUTPUT para fazer ações de redirecionamento de portas e servidores, balanceamento de carga e proxy transparente. Caso a porta de destino não seja especificada, valerá a porta de origem. No firewall, a porta que será redirecionada não pode existir ou estar ocupada por um daemon.

MASQUERADE --> Faz mascaramento na saída de dados.
REDIRECT --> Redireciona uma requisição para uma porta local do firewall.

#iptables -t nat -L
Mostra as regras de NAT ativas.

#iptables -t nat -F
Apaga todas as regras de NAT existentes.

#iptables -t nat -A POSTROUTING -o ppp0 -j MASQUERADE
Todos os pacotes que saírem pela interface ppp0 (modem) serão mascarados. Isso dá um nível de segurança elevado à rede que está atrás da ppp0. É uma boa regra para navegação na Internet. Note que esse tipo de mascaramento não usa SNAT.

#iptables -t nat -A POSTROUTING -d 0/0 -j MASQUERADE
Tem o mesmo efeito da regra anterior. No entanto, parece ser menos segura, pois estabelece que qualquer pacote destinado a qualquer outra rede, diferente da interna, será mascarado. A regra anterior refere-se aos pacotes que saem por determinada interface. A opção -d 0/0 poderia ser -d 0.0.0.0/0 também. É uma outra regra para navegação na Internet.

#iptables -t nat -A PREROUTING -p tcp -d 10.0.0.2 --dport 80 -j DNAT --to 172.20.0.1
Redireciona todos os pacotes destinados à porta 80 da máquina 10.0.0.2 para a máquina 172.20.0.1. Esse tipo de regra exige a especificação do protocolo. Como não foi especificada uma porta de destino, a porta 80 será mantida como destino.

#iptables -t nat -A OUTPUT -p tcp -d 10.0.0.10 -j DNAT --to 10.0.0.1
Qualquer pacote TCP, originado na máquina firewall, destinado a qualquer porta da máquina 10.0.0.10, será desviado para a máquina 10.0.0.1 .

#iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to 200.20.0.1
Essa regra faz com que todos os pacotes que irão sair pela interface eth0 tenham o seu endereço de origem alterado para 200.20.0.1 .

#iptables -t nat -A PREROUTING -i eth0 -j DNAT --to 172.20.0.1
Todos os pacotes que entrarem pela eth0 serão enviados para a máquina 172.20.0.1

#iptables -t nat -A PREROUTING -i eth0 -j DNAT --to 172.20.0.1-172.20.0.3
Aqui haverá o load balance. Todos os pacotes que entrarem pela eth0 serão distribuídos entre as máquinas 172.20.0.1 , 172.20.0.2 e 172.20.0.3

#iptables -t nat -A PREROUTING -s 10.0.0.0/8 -p tcp --dport 80 -j REDIRECT --to-port 3128
Todos os pacotes TCP que vierem da rede 10.0.0.0, com máscara 255.0.0.0, destinados à porta 80 de qualquer host, não sairão; serão redirecionados para a porta 3128 do firewall. Isso é o passo necessário para fazer um proxy transparente. O proxy utilizado deverá aceitar esse tipo de recurso. No caso, o Squid, que aceita transparência, deverá estar instalado na máquina firewall, servindo na porta 3128.

#iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -o eth1 -j SNAT 200.20.5.0/24
Uma situação interessante: todos os pacotes que saírem da rede 192.168.1.0 serão transformados em 200.20.5.0 .

Execução do mascaramento destinado à Internet

Por ser uma atividade perigosa, o acesso à Internet deve ser feito com um máximo grau de segurança. Assim, vejamos as regras básicas para permitir que uma rede privada navegue com um IP válido.

Primeiro exemplo: uma rede na Internet

Vamos permitir que a rede 10.0.0.0 navegue na Internet. A máquina firewall (gateway) será a 10.0.0.1. Regras:

```
#echo 1 > /proc/sys/net/ipv4/ip_forward
#modprobe iptable_nat
#iptables -t nat -A POSTROUTING -s 10.0.0.0/8 -o ppp0 -j MASQUERADE
```

O procedimento é totalmente seguro, pois discrimina uma origem, que só poderá sair pela ppp0, de forma mascarada. Hoje em dia, o carregamento do módulo iptable_nat, na maioria das vezes, se dá automaticamente, dispensando a segunda linha.

Segundo exemplo: alguns hosts na Internet

Vamos permitir que alguns hosts, no caso, o 10.0.0.10, o 10.0.0.20 e o 10.5.2.41, naveguem na Internet. A máquina firewall (gateway) será a 10.0.0.1. Regras:

```
#echo 1 > /proc/sys/net/ipv4/ip_forward
#iptables -t nat -A POSTROUTING -s 10.0.0.10 -o ppp0 -j MASQUERADE
#iptables -t nat -A POSTROUTING -s 10.0.0.20 -o ppp0 -j MASQUERADE
#iptables -t nat -A POSTROUTING -s 10.5.2.41 -o ppp0 -j MASQUERADE
```

Tabelas Filter e NAT atuando em conjunto

As tabelas filter e nat podem atuar em conjunto, funcionando em paralelo. Há de se ter cuidado pois, como já disse, elas atuam em paralelo, como duas pilhas que serão executadas ao mesmo tempo. Assim sendo, se tivermos as regras:

```
#iptables -t nat -A POSTROUTING -s 10.0.0.0/8 -o eth0 -j MASQUERADE
#iptables -A FORWARD -j DROP
```

Apesar da primeira (tabela nat) possibilitar a navegação mascarada da rede 10.0.0.0 na Internet, essa navegação não ocorrerá, pois a segunda regra (tabela filter) irá barrar o forward entre as redes.

- **SALVANDO E RECUPERANDO REGRAS**

As regras do iptables devem ser salvas de algum modo, ou se perderão quando a máquina for desligada. As formas para salvar as regras são:

- Criar um script e coloca-lo para iniciar no rc.local

Criar o arquivo /etc/init.d/firewall com as regras do firewall

```
#!/bin/sh
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP
```

Tornar o arquivo executável.

```
#chmod 777 firewall
```

No arquivo rc.local inserir no final a linha

```
/etc/init.d/firewall start
```

Testar a configuração reinicializando o computador.

- Outra forma de fazer isso é com o comando iptables-save.

```
#iptables-save >/etc/firewall
```

Para carregar as regras utiliza-se o comando iptables-restore.

```
#iptables-restore </etc/firewall
```

5. A SEGURANÇA NO FIREWALL

O sistema de firewall deve ser protegido para que o restante da rede também tenha segurança. Assim, algumas regras básicas devem ser observadas:

- Feche a máquina firewall, de modo que todas os pacotes destinados diretamente a ela sejam descartados:

```
#iptables -P INPUT DROP
```

- Em seguida, aos poucos, abra o que for necessário. Cuidado, pois muitas vezes o firewall precisará de vários acessos abertos. Por exemplo: se uma máquina firewall isolado também for proxy, a mesma será servidora da intranet e cliente da Internet, necessitando assim das portas superiores a 1023 abertas.
- Atualize sempre o firewall e o kernel;
- NUNCA rode qualquer serviço, principalmente os remotos, como telnet e ftp, na máquina firewall, quando se tratar de firewall isolado;
- Se tiver que administrar remotamente um firewall, utilize ssh. Nesse caso, o ssh não deverá permitir o login como root;
- Nunca cadastre qualquer usuário na máquina firewall, caso se trate de firewall isolado, a não ser os que irão administrar por ssh;
- Utilize TCP Wrappers totalmente fechado (ALL:ALL em /etc/hosts.deny) na máquina de firewall isolado; abra o ssh (em /etc/hosts.allow) apenas para os clientes que forem fazer administração remota;
- Anule as respostas a ICMP 8 (echo reply) no firewall isolado, para evitar identificação da topologia na rede e ataques de Ping of Death. A melhor forma de se fazer isso é atuando sobre regras do kernel, com o comando:

```
#echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_all
```

- Não insira referências ao firewall no DNS;
- Não deixe o firewall isolado com cara de firewall. Dê um nome descaracterizado para ele;

Faça log de ações suspeitas que estiverem ocorrendo na rede;

- Teste, teste, teste novamente.
- Não insira referências ao firewall no DNS;
- Não deixe o firewall isolado com cara de firewall. Dê um nome descaracterizado para ele;
- Faça log de ações suspeitas que estiverem ocorrendo na rede;
- Teste, teste, teste novamente.