

Nickollas Carvalho
nickollas.wordpress.com

[TREINAMENTO LINUX]

Setembro de 2010

Sumário

[TREINAMENTO LINUX].....	1
[INTRODUÇÃO].....	8
[Introdução ao Software Livre]	8
[Licenças BSD, GPL, Artística].....	8
[Distribuições Linux].....	8
[Debian Linux].....	8
[CAPÍTULO 1 - INSTALAÇÃO].....	9
[Obtendo Debian Linux].....	9
[Instalação do Debian em modo texto].....	9
[RAID: Alta Disponibilidade em discos rígidos].....	29
[Sistema de Arquivos ext3].....	29
[CAPÍTULO 2 - FUNDAMENTOS].....	30
[Console].....	30
[Estrutura de Arquivos e Diretórios Linux]	30
A raiz /.....	30
O diretório /boot.....	30
O diretório /etc.....	30
O diretório /home.....	30
O diretório /root.....	30
O diretório /var.....	30
O diretório /proc.....	31
[Limpando a tela].....	31
[Sintaxe de comandos Linux].....	31
[Nomeclatura adotada para sintaxe dos comandos].....	31
[Manipulação de arquivos e diretórios].....	32
O comando ls.....	32
O comando cp.....	32
O comando du.....	32
O comando mv.....	32
O comando rm.....	32
O comando mkdir.....	33
O comando cd.....	33
[Imprimindo caracteres na saída padrão]	33
O comando echo.....	33
[Operadores da Shell].....	33
O operador &.....	33
O operador &&.....	34
[Entrada e Saída de dados].....	34
O operador >.....	34
O operador >>.....	34
O operador 	34
[Controle de fluxo].....	34
O comando more.....	34
[Porções específicas].....	35
O comando tail.....	35
O comando head.....	35

[Contagem].....	35
O comando wc.....	35
[Filtragem]	35
O comando grep.....	35
[Localização de arquivos e diretórios].....	36
O comando find.....	36
[Símbolos Curinga].....	36
O símbolo curinga: ^.....	36
O símbolo curinga: \$.....	36
O símbolo curinga: ?.....	36
O símbolo curinga: *.....	36
[Diferença entre arquivos].....	37
O comando diff.....	37
[Gerenciamento de memória RAM].....	37
O comando free.....	37
[Mostrar informações sobre o sistema].....	37
O comando uname.....	37
[Mostrar / ajustar a data do sistema].....	37
O comando date.....	37
[Mostrar por quanto tempo o computador está ligado].....	38
O comando uptime.....	38
[Tempo de execução de um programa].....	38
O comando time.....	38
[Conhecendo a Documentação]	38
O comando man.....	38
O atributo --help.....	38
[Editores de texto].....	38
O editor vim.....	38
[Configuração de rede].....	39
O arquivo /etc/udev/rules.d/*-persistent-net.rules.....	39
O comando ifconfig.....	39
O arquivo /etc/network/interfaces.....	39
O arquivo /etc/resolv.conf.....	39
O comando route.....	39
O arquivo /etc/hosts.....	40
[Gerenciamento de rede].....	40
O comando mii-tool.....	40
O comando ping.....	40
[Manipulando dispositivos de armazenamento].....	40
O comando dmesg.....	40
O comando mount.....	40
[Gerenciamento de partições].....	40
O comando df.....	40
O comando cfdisk.....	41
O comando mkfs.....	41
O arquivo /etc/fstab.....	41
[Gerenciamento de processos].....	41
O comando ps.....	41
O comando top.....	41
O comando kill.....	41
O comando killall5.....	42

[Prioridades dos processos].....	42
O comando nice.....	42
O comando renice.....	42
[CAPÍTULO 3 - ADMINISTRAÇÃO].....	43
[Gerenciamento de login].....	43
O comando w.....	43
O comando lastlog.....	43
[Permissões em arquivos e diretórios].....	43
O comando chmod.....	43
[Gerenciando usuários e grupos].....	43
O arquivo /etc/passwd.....	43
O arquivo /etc/shadow.....	43
O arquivo /etc/group.....	44
O diretório /etc/skel.....	44
O arquivo /etc/nologin.....	44
[Alterando dono de arquivos e diretórios]	44
O comando chown.....	44
[Alterando grupo de arquivos e diretórios].....	44
O comando chgrp.....	44
[Manipulando senhas de usuários]	44
O comando passwd.....	44
[Adicionando usuário].....	45
O comando useradd.....	45
Criando ambiente do usuário.....	45
[Alterando contas de usuário].....	45
O comando usermod.....	45
[Informações dos usuários].....	45
O comando id.....	45
[Removendo Usuário]	46
O comando userdel.....	46
[Atributos de arquivos e diretórios].....	46
O comando lsattr.....	46
O comando chattr.....	46
[Gerenciamento de pacotes com cálculo de dependência – apt].....	46
O comando apt-get.....	46
O comando apt-cache.....	47
O comando apt-cdrom.....	47
[Gerenciamento de pacotes - dpkg].....	47
O comando dpkg.....	47
O atributo --purge do comando dpkg.....	47
O comando dpkg-reconfigure.....	48
[Informações sobre arquivos].....	48
O atributo -S do comando dpkg.....	48
O comando file.....	48
[Atualização via Internet].....	48
A opção upgrade do comando apt-get.....	48
[Ativando syntax no vim].....	48
O arquivo /etc/vim/vimrc.....	48
[Empacotadores].....	49
O pacote bzip2.....	49
O comando tar.....	49

[Download de arquivos].....	49
O comando wget.....	49
[Compilação de programas].....	49
O comando ./.....	49
O arquivo configure.....	49
O comando make.....	50
A opção install do comando make.....	50
O comando gcc.....	50
[Gerenciamento de Hardware e Dispositivos].....	50
O comando lspci.....	50
[Gerenciamento de módulos do kernel].....	51
O diretório /lib/modules.....	51
O comando lsmod.....	51
O comando modprobe.....	51
O comando rmmod.....	51
O comando insmod.....	51
[Níveis de Execução].....	51
O comando runlevel.....	51
O comando init.....	51
O arquivo /etc/inittab.....	52
[Agendamento de tarefas].....	52
O cron.....	52
O comando crontab.....	52
[Agendando uma tarefa].....	52
Configurando o crontab.....	52
[Gerenciador de Boot GRUB].....	53
O arquivo /boot/grub/menu.lst.....	53
[Utilitários].....	53
O comando watch.....	53
O comando cal.....	53
O comando bc.....	53
[O kernel].....	53
[Compilando o kernel]	54
[Gerenciamento de log].....	55
O diretório /var/log/.....	55
O arquivo /var/log/syslog.....	55
O arquivo /var/log/messages.....	55
[CAPÍTULO 4 - REDES].....	56
O arquivo /etc/sysctl.conf.....	56
O diretório /proc/sys/net.....	56
O comando netstat.....	56
[Servidor SSH]	56
Apresentação.....	56
Instalação do servidor SSH.....	56
Configuração do servidor SSH.....	56
Instalação do cliente SSH.....	57
O comando ssh.....	57
[Servidor NIS]	57
Apresentação.....	57
Instalação do servidor NIS.....	57
Configuração do servidor NIS.....	57

Instalação do cliente NIS.....	58
Configuração do cliente NIS.....	58
[Servidor NFS]	59
Apresentação.....	59
Instalação do servidor NFS.....	59
Configuração do servidor NFS.....	59
Instalação do cliente NFS.....	60
Configuração do cliente NFS.....	60
[Servidor DNS]	60
Apresentação.....	60
Instalação.....	60
Configuração	60
[Servidor Web]	61
Apresentação.....	61
Instalação.....	62
Configuração de domínios virtuais no apache.....	62
[Suporte a PHP com MySQL no Servidor Web]	63
Apresentação.....	63
Instalação.....	63
Configuração.....	63
[Servidor FTP]	63
Apresentação.....	63
Instalação.....	63
Configuração.....	64
[Servidor Proxy]	64
Apresentação.....	64
Instalação.....	64
Configuração.....	65
O arquivo /var/log/squid3/access.log.....	66
[Auditoria no Servidor Proxy]	66
Apresentação.....	66
Instalação.....	66
Configuração.....	66
[Firewall]	66
Apresentação.....	66
O comando iptables.....	67
Módulos.....	67
Criando regras de firewall.....	67
O comando iptables-save.....	69
O comando iptables-restore.....	69
Fluxo de dados no iptables.....	70
[Utilitários de Rede].....	70
O comando mtr.....	70
O comando nmap.....	71
[O que é uma Shell?].....	72
[O que é um Script?]	72
[O que é Shell Script?].....	72
[Shell Script].....	72
Definindo o interpretador: #!.....	72
Variáveis.....	72
Tabela de variáveis fornecidas pela linguagem:.....	72

O operador = (igual).....	73
Diferença entre: ' (apóstrofo) e “ (aspas).....	73
O comando: \n.....	73
O comando if.....	74
Tabela de operadores do comando if.....	74
O comando else.....	75
O loop for.....	76
O comando sleep.....	77
O loop while.....	77
Tabela de operadores do loop while.....	78
[Script de Backup].....	78
Exemplo de Script.....	79

[INTRODUÇÃO]

[Introdução ao Software Livre]

Software livre é qualquer programa de computador que pode ser usado, copiado, estudado e redistribuído sem restrições, sem custos.

[Licenças BSD, GPL, Artística]

Licença de software é uma autorização do autor de um programa de computador concedida ao usuário deste software.

As **licenças BSD** (Berkeley Software Distribution) e **GPL** (General Public License) são as mais comuns para o software livre. As licenças BSD e a GPL diferem bastante no modo em que o código fonte pode ser usado.

A GPL requer que trabalhos derivados de software GPL sejam licenciados sob GPL. Já a licença BSD requer apenas o reconhecimento dos autores e outras pequenas restrições. Como resultado os códigos BSD podem ser utilizados em projetos livres com outras licenças como Linux (GPL) ou softwares proprietários como pilha IP do Windows e o Mac OS X.

A licença artística é utilizada por alguns softwares livres, como a implementação padrão do Perl.

[Distribuições Linux]

Uma Distribuição Linux é composta do núcleo Linux e um conjunto variável de software, dependendo de seus propósitos. Essa coleção de software livre e não-livre, é criada e mantida por indivíduos, grupos e organizações de todo o mundo. Indivíduos como Patrick Volkerding, companhias como a Red Hat, a SuSE, a Mandriva e a Canonical, bem como projetos de comunidades como o Debian ou o Gentoo, compilam softwares e fornecem a usuários diversos sistemas completos, prontos para instalação e uso.

[Debian Linux]

Debian é o nome de uma distribuição não comercial livre (gratuita e de código fonte aberto) de GNU/Linux (amplamente utilizada) onde um grupo de voluntários a mantêm. O Debian é especialmente conhecido pelo seu sistema de gestão de pacotes, chamado APT, que permite: atualizações relativamente fáceis a partir de versões realmente antigas; instalações quase sem esforço de novos pacotes e remoções limpas dos pacotes antigos. O Debian procura sempre manter os pacotes mais estáveis e antigos, garantindo estabilidade que é o grande foco para servidores.

[CAPÍTULO 1 - INSTALAÇÃO]

[Obtendo Debian Linux]

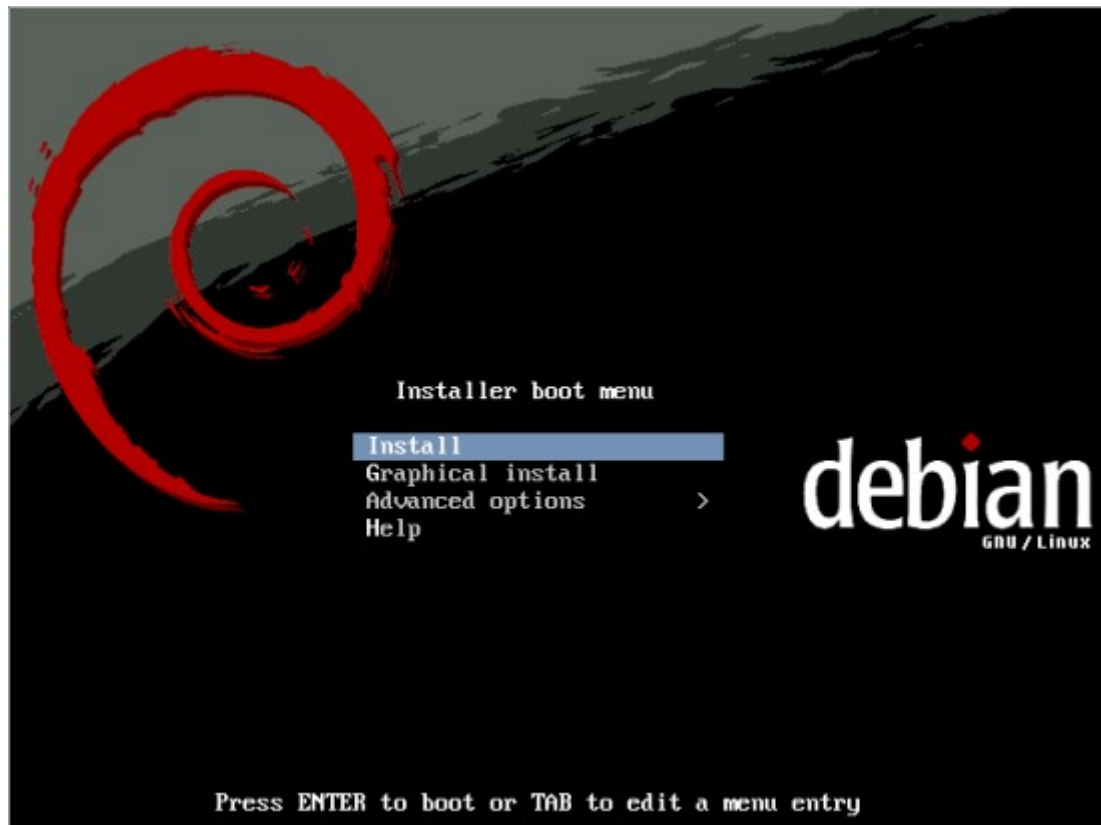
Obtenha a imagem .iso de instalação via rede do Debian em <http://debian.org/CD/netinst/> (Escolha sua arquitetura de processador em “businesscard images”).

Após fazer download, grave o CD e não se esqueça de configurar o BIOS para dar boot pelo CD.

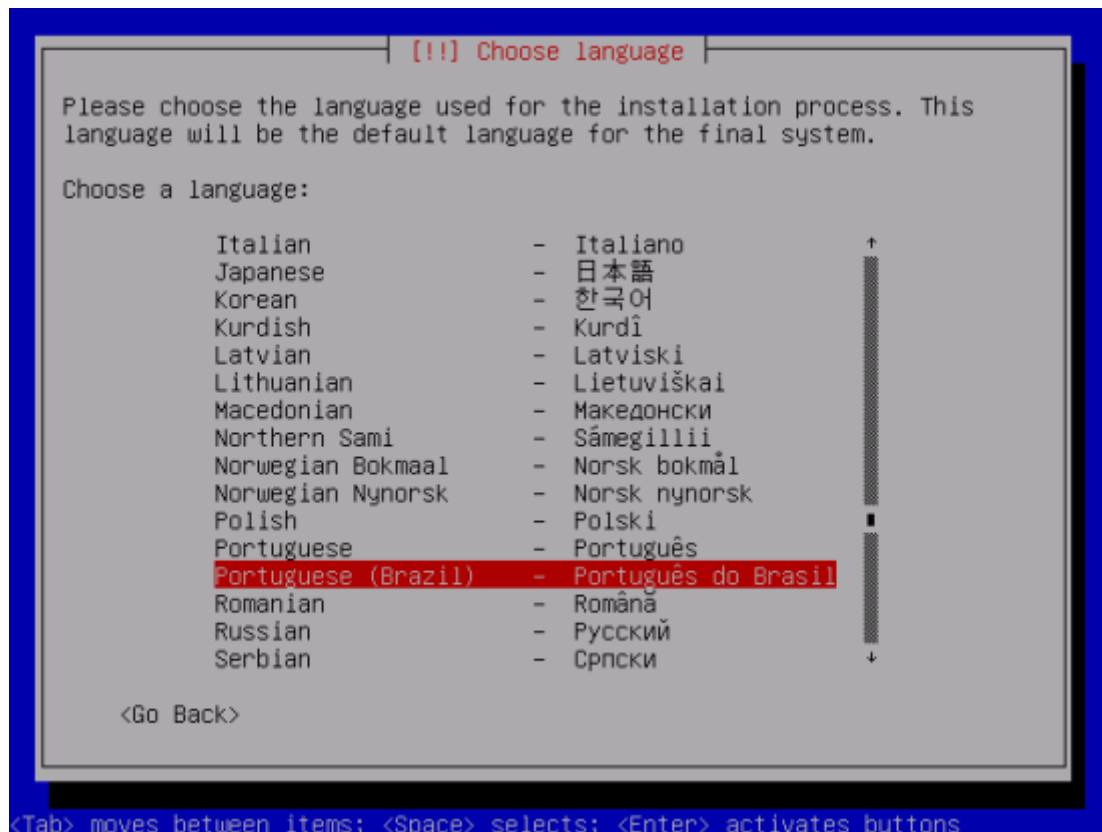
[Instalação do Debian em modo texto]

As sequência de telas abaixo irão lhe guiar para instalar o Debian em modo texto.

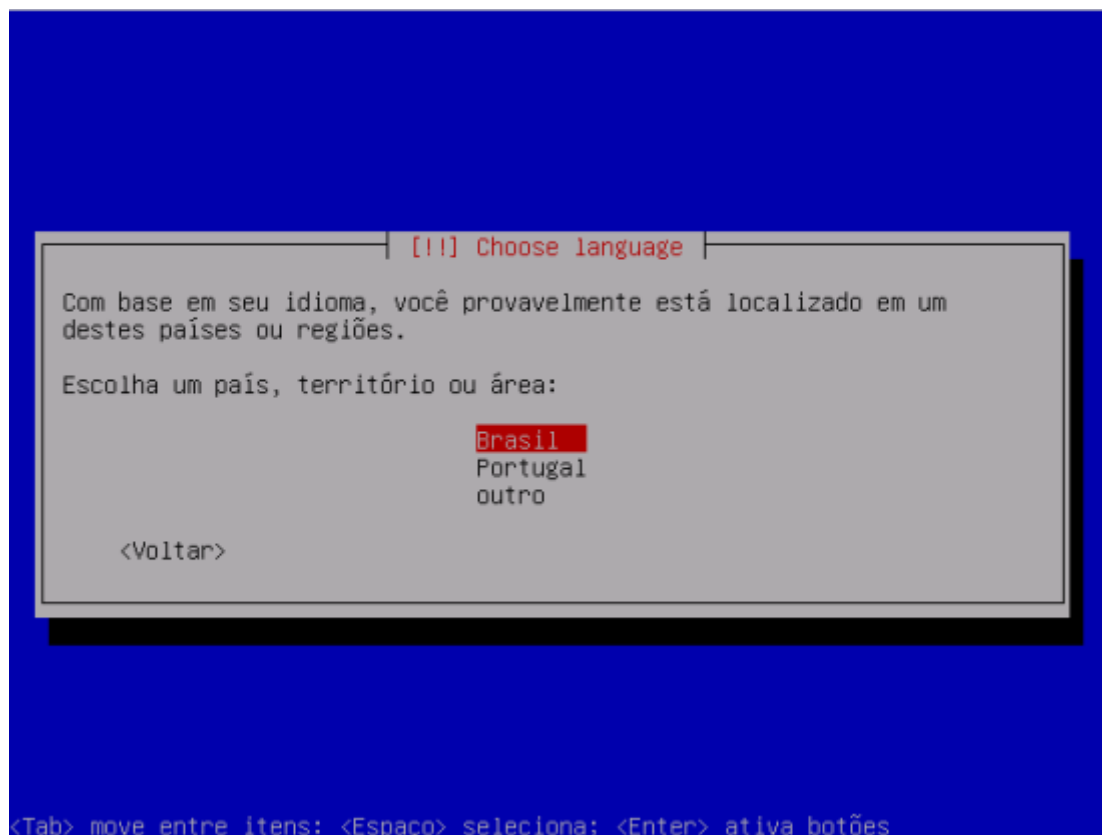
1 – Ao dar boot pelo CD a tela de opção de boot aparecerá. Escolha: Install



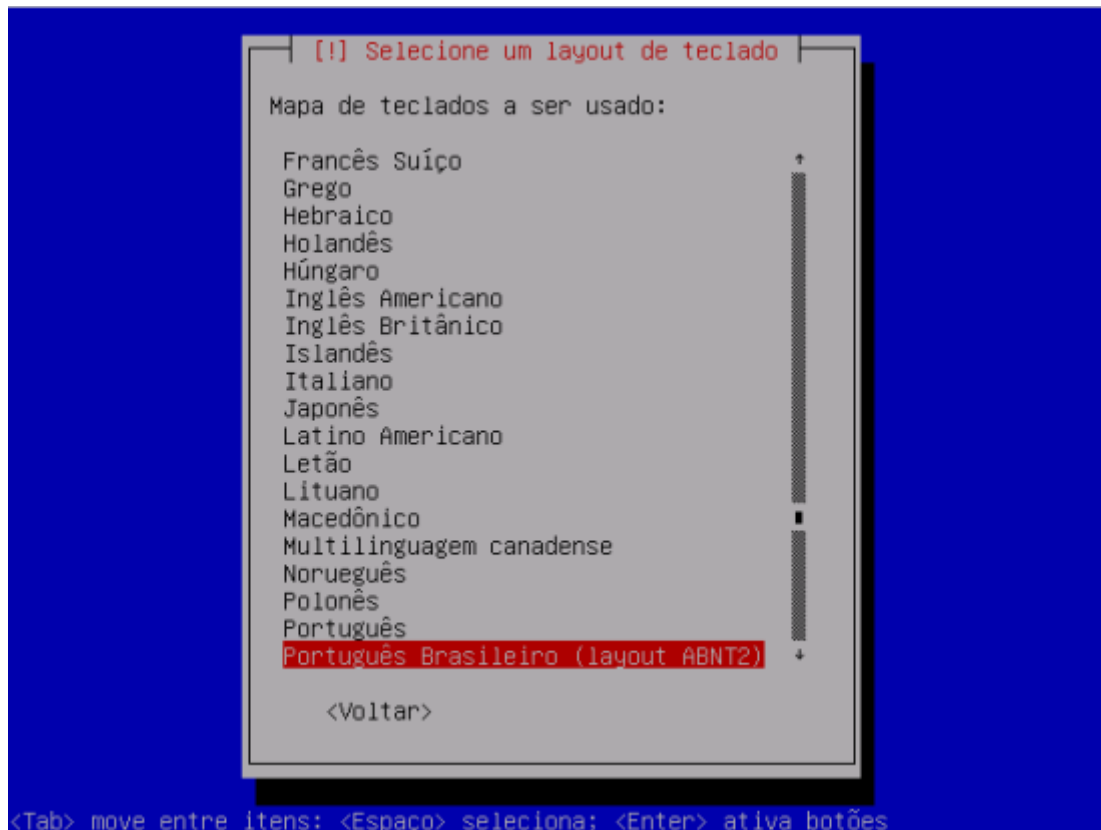
2 – Escolha o Idioma: Portuguese (Brazil) - Use a tecla TAB para navegar entre os menus



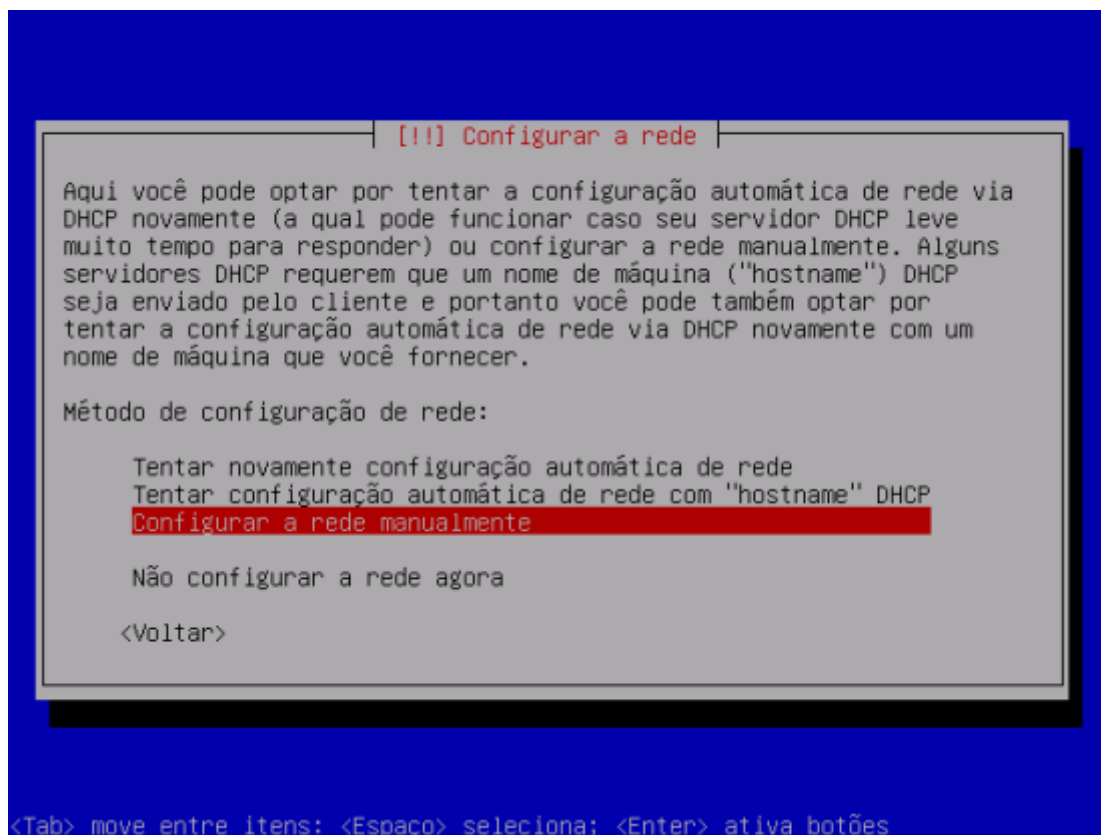
3 – Escolha o país: Brasil



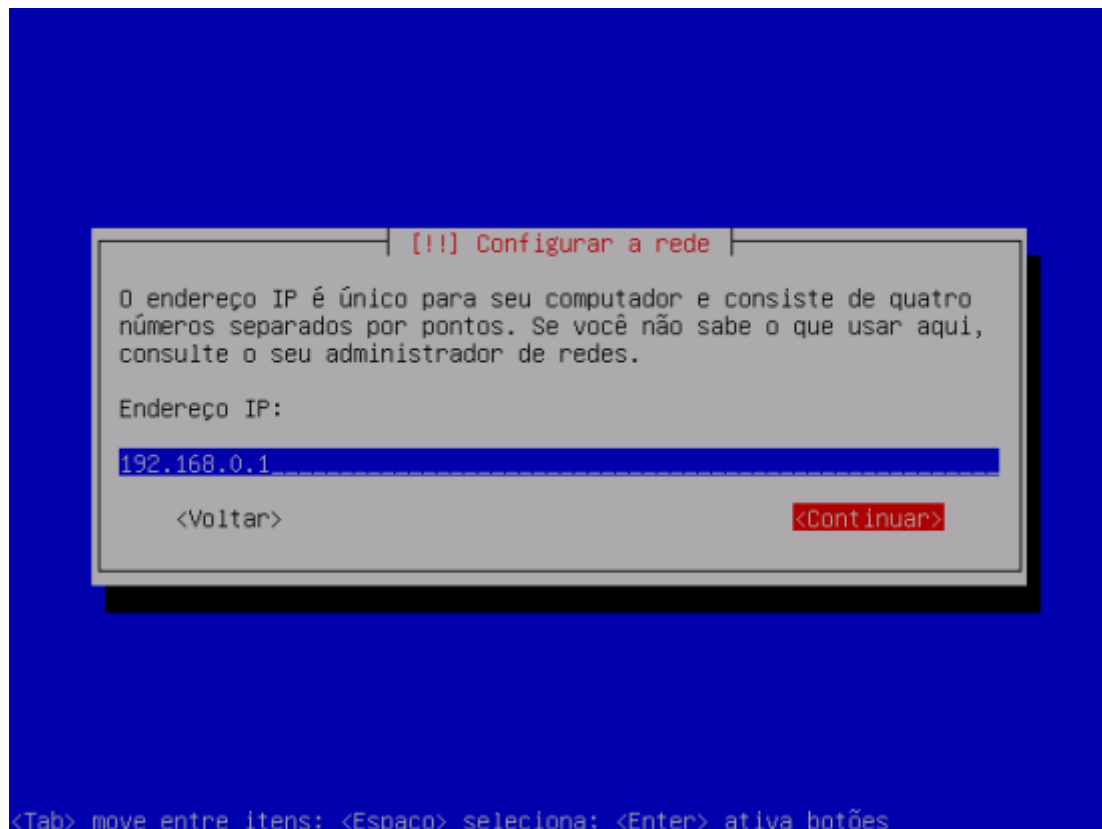
4 – Selecione o Layout do Teclado: Português Brasileiro (layout ABNT2)



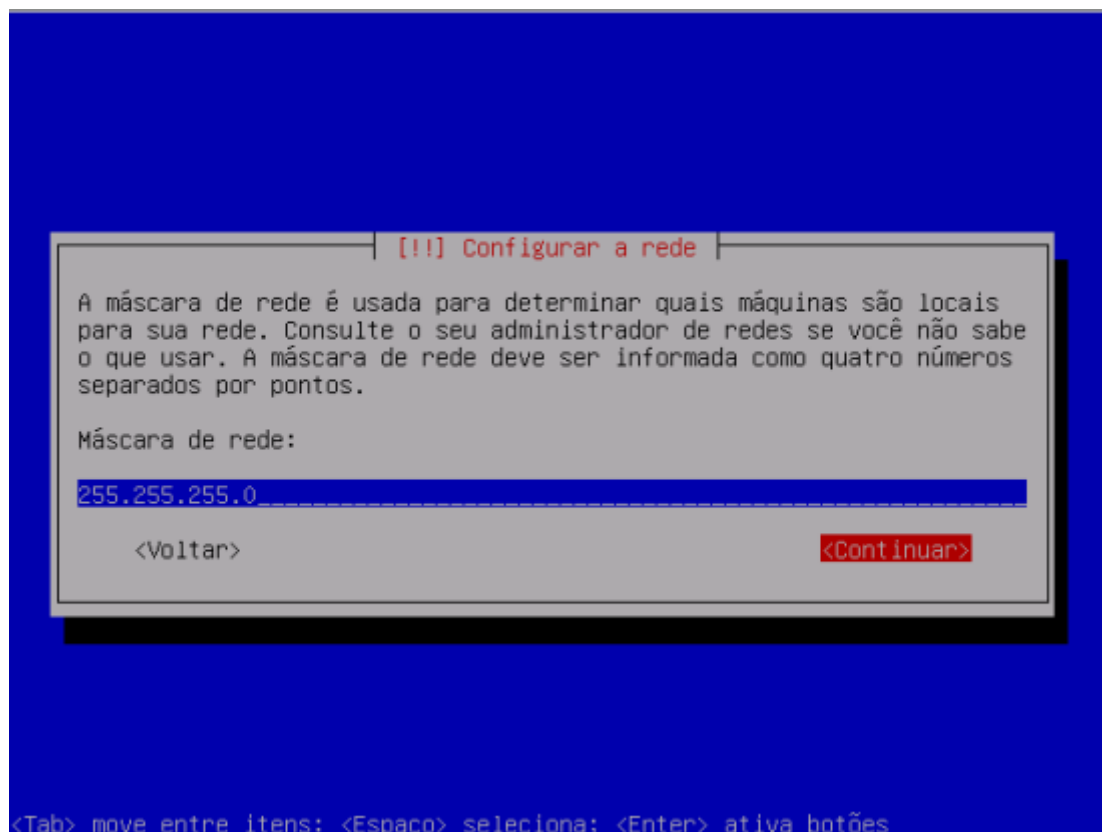
5 – Escolha: Configurar a rede manualmente



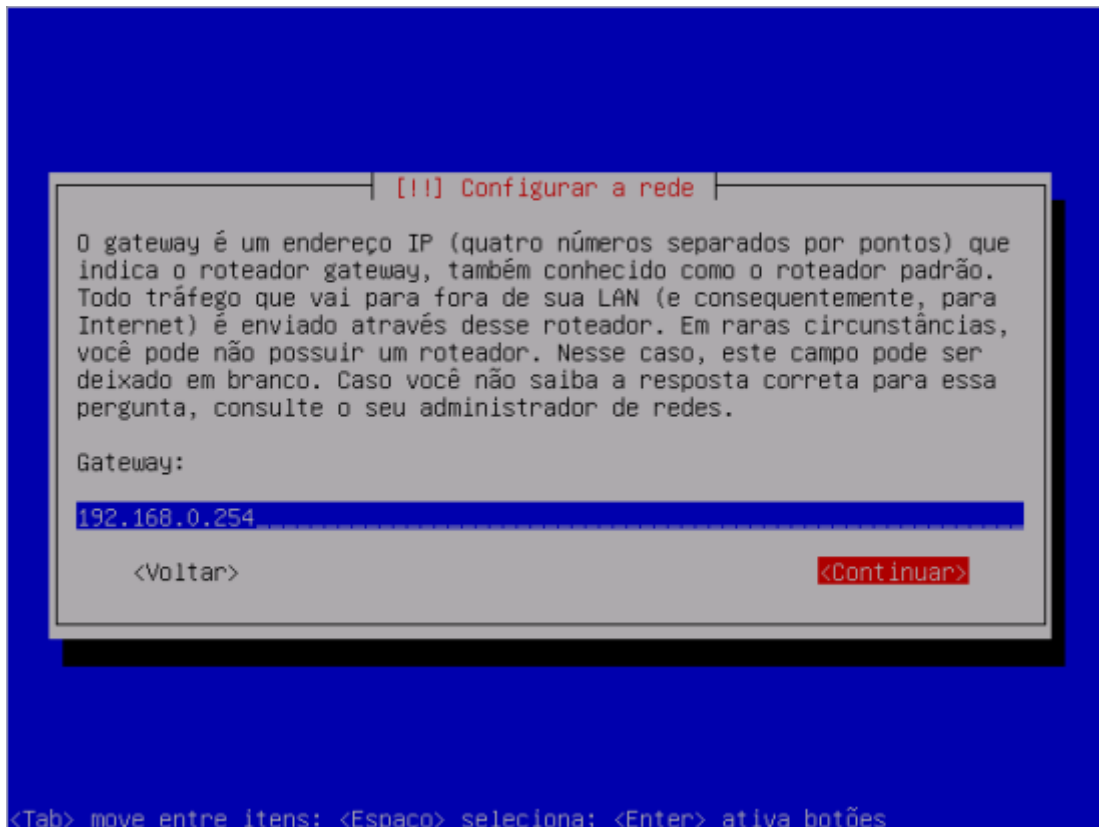
6 - Informe o endereço de IP



7 – Informe a Máscada de rede



8 – Informe o gateway



[!!] Configurar a rede

O gateway é um endereço IP (quatro números separados por pontos) que indica o roteador gateway, também conhecido como o roteador padrão. Todo tráfego que vai para fora de sua LAN (e conseqüentemente, para Internet) é enviado através desse roteador. Em raras circunstâncias, você pode não possuir um roteador. Nesse caso, este campo pode ser deixado em branco. Caso você não saiba a resposta correta para essa pergunta, consulte o seu administrador de redes.

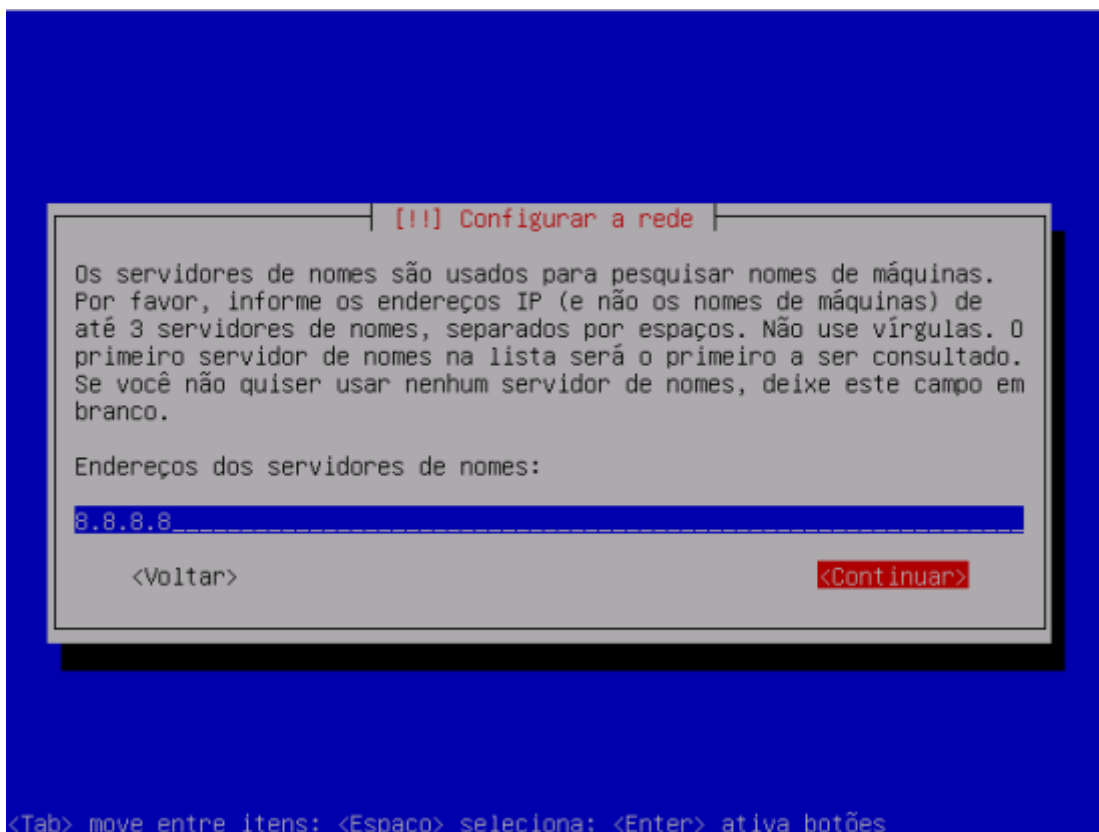
Gateway:

192.168.0.254

<Voltar> <Continuar>

<Tab> move entre itens; <Espaço> seleciona; <Enter> ativa botões

9 – Informe os endereços dos servidores de nomes (DNS)



[!!] Configurar a rede

Os servidores de nomes são usados para pesquisar nomes de máquinas. Por favor, informe os endereços IP (e não os nomes de máquinas) de até 3 servidores de nomes, separados por espaços. Não use vírgulas. O primeiro servidor de nomes na lista será o primeiro a ser consultado. Se você não quiser usar nenhum servidor de nomes, deixe este campo em branco.

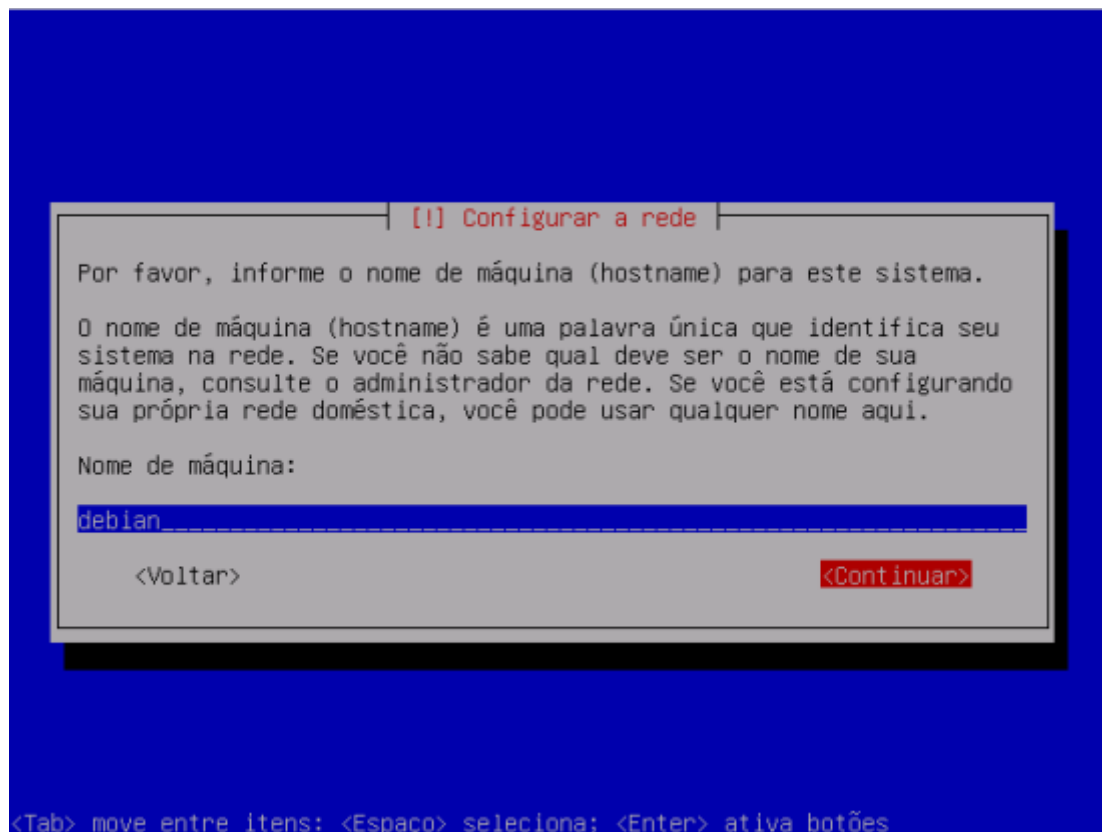
Endereços dos servidores de nomes:

8.8.8.8

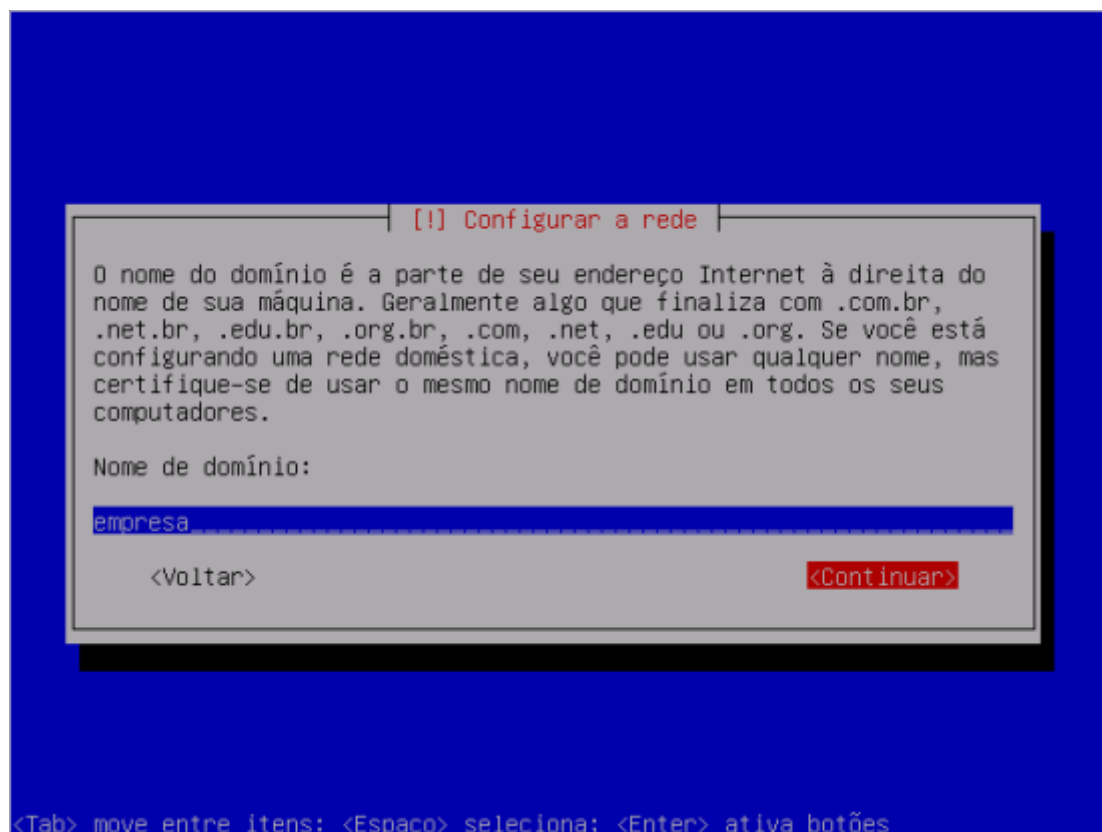
<Voltar> <Continuar>

<Tab> move entre itens; <Espaço> seleciona; <Enter> ativa botões

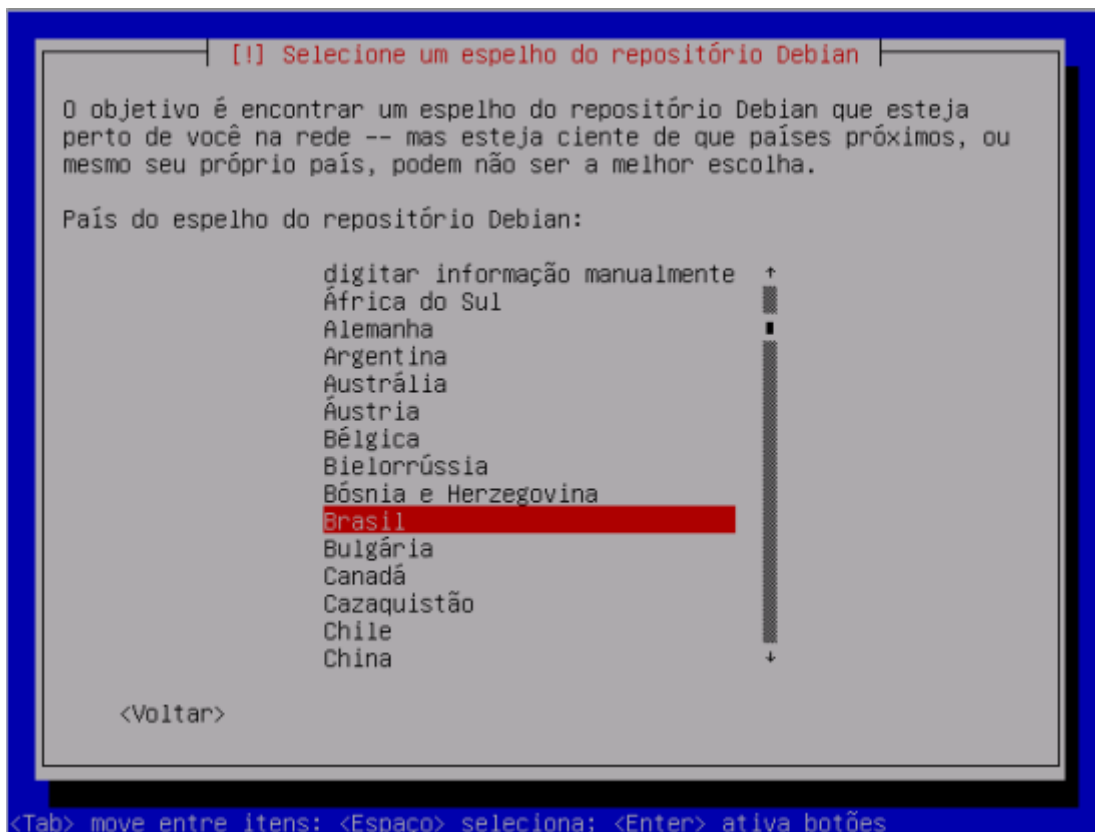
10 – Informe o Nome da máquina



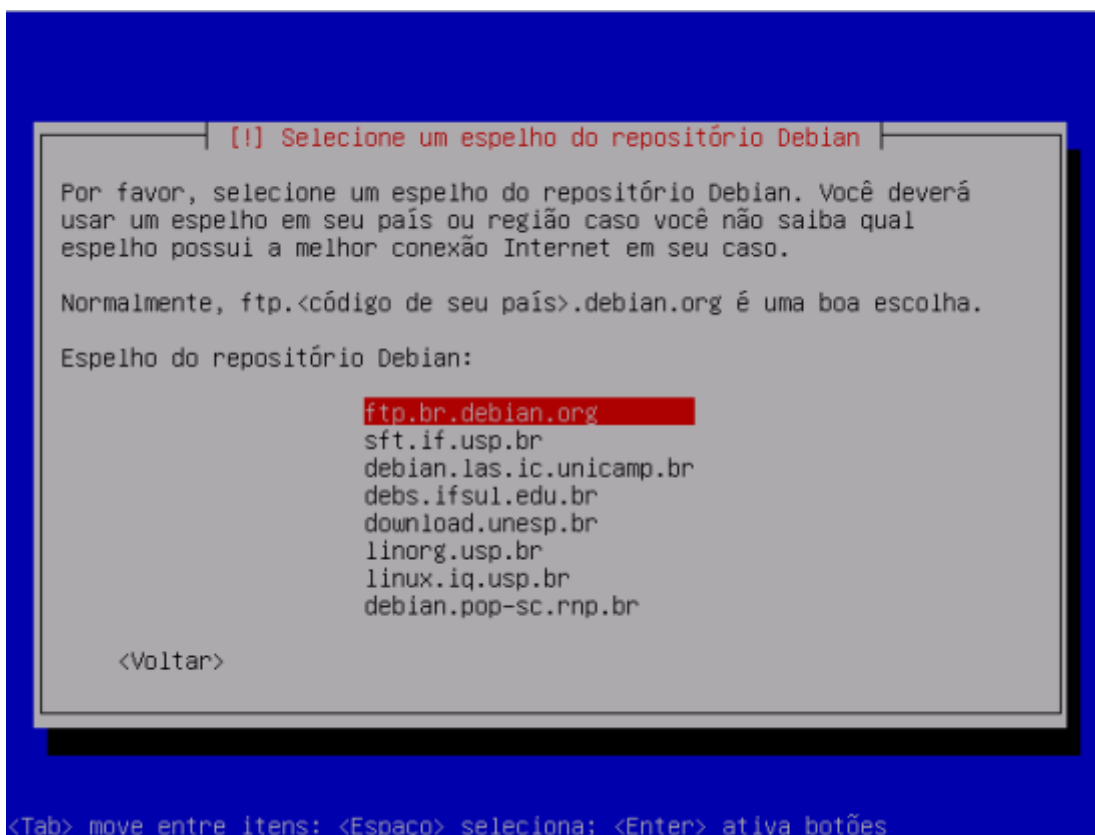
11 – Informe o Nome do domínio



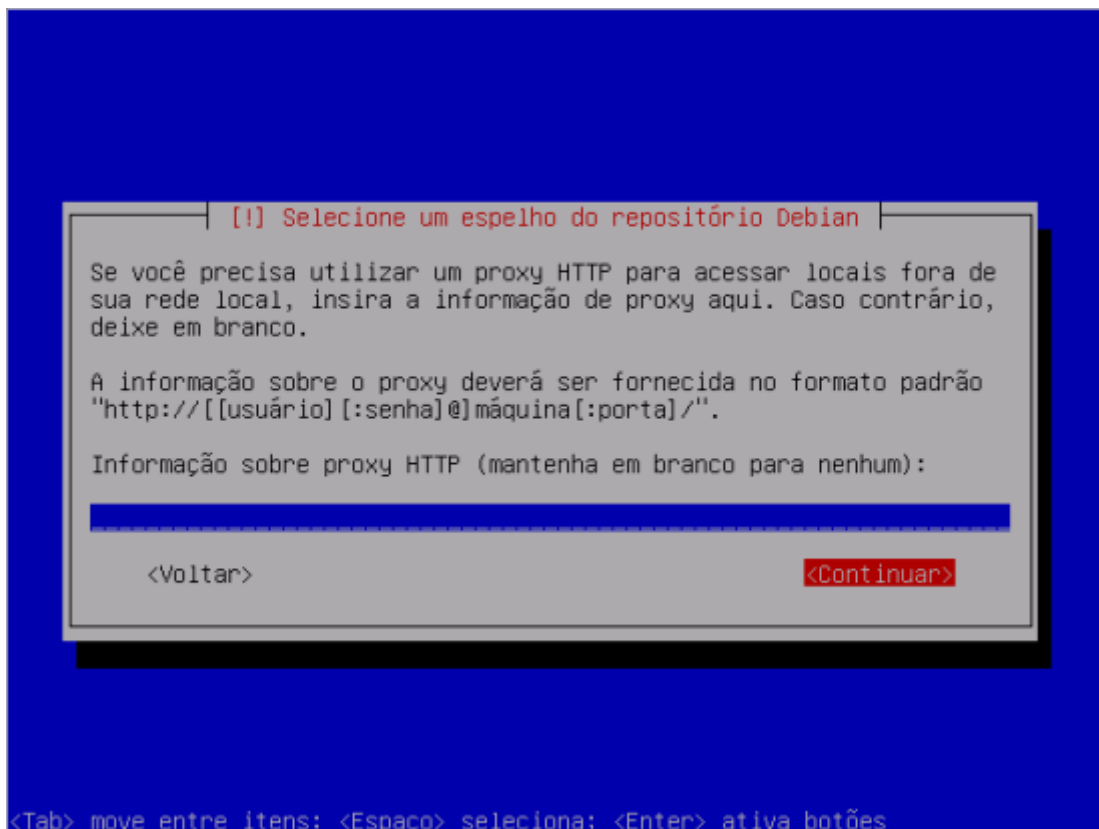
12 – Escolha o país do espelho do repositório Debian: Brasil



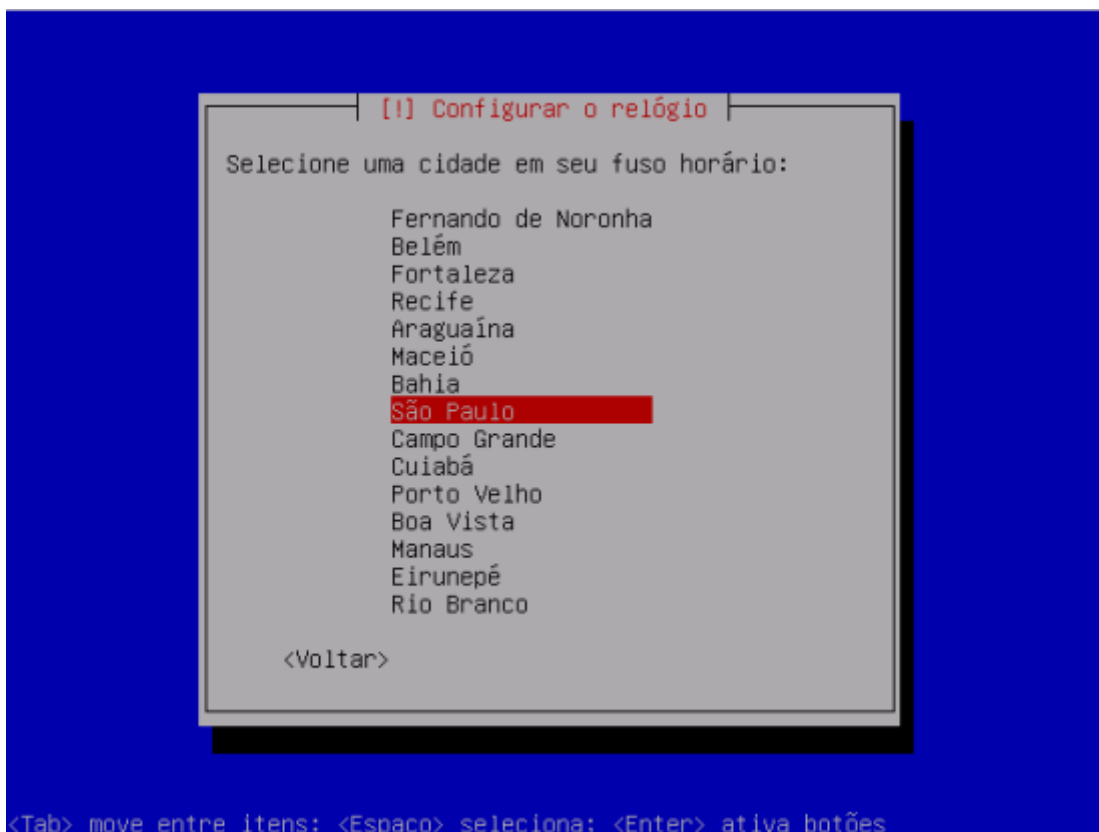
13 – Espelho do repositório Debian, escolha: ftp.br.debian.org



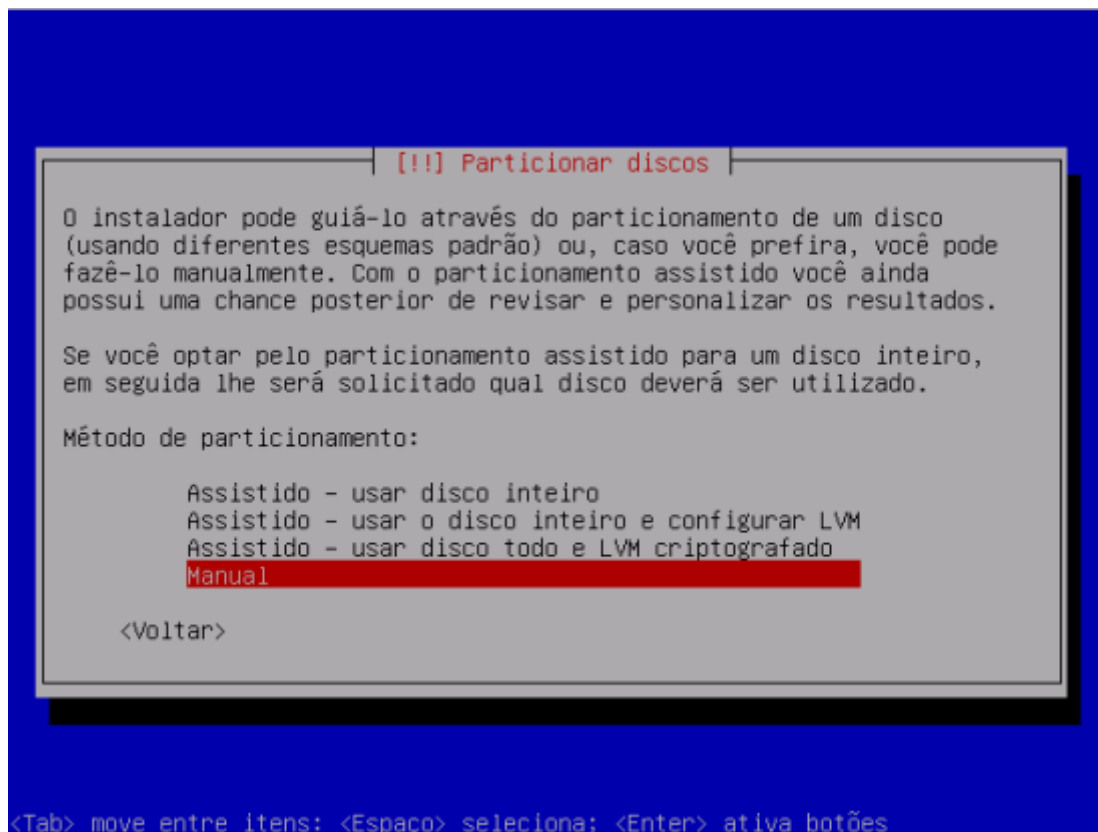
14 – Informação sobre proxy HTTP, deixe em branco.



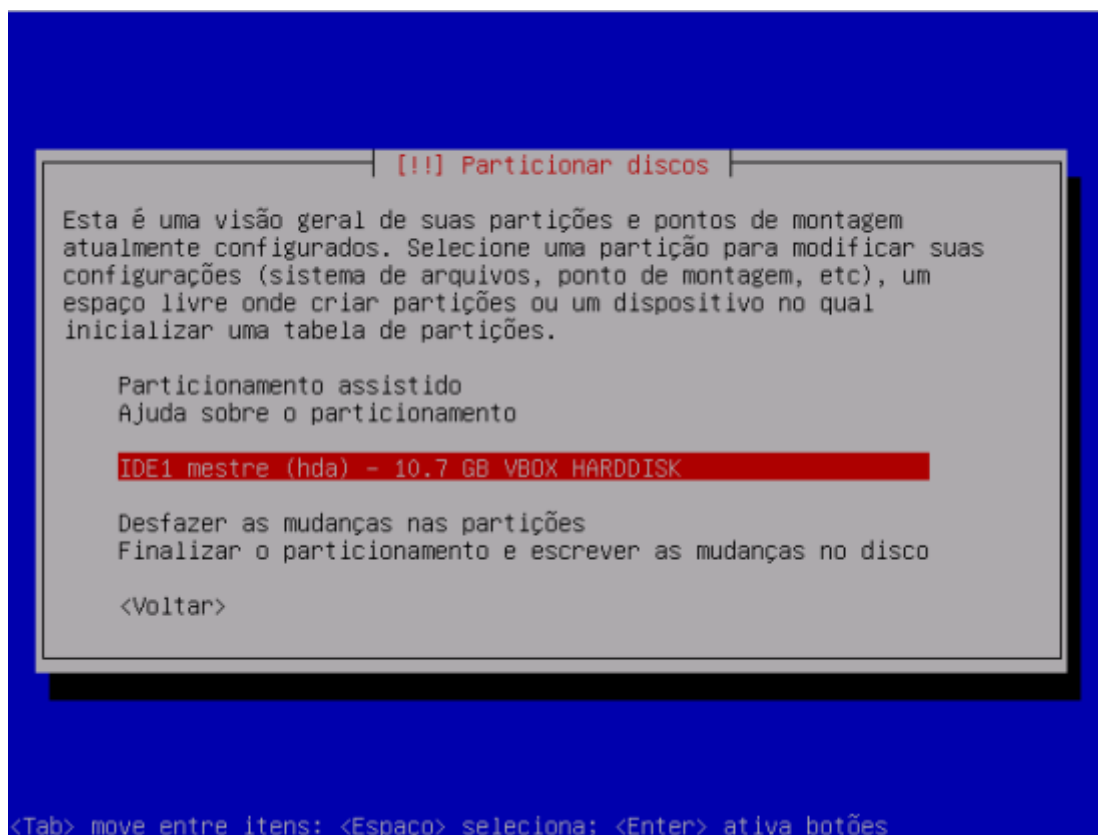
15 – Selecione uma cidade em seu fuso horário, escolha: São Paulo



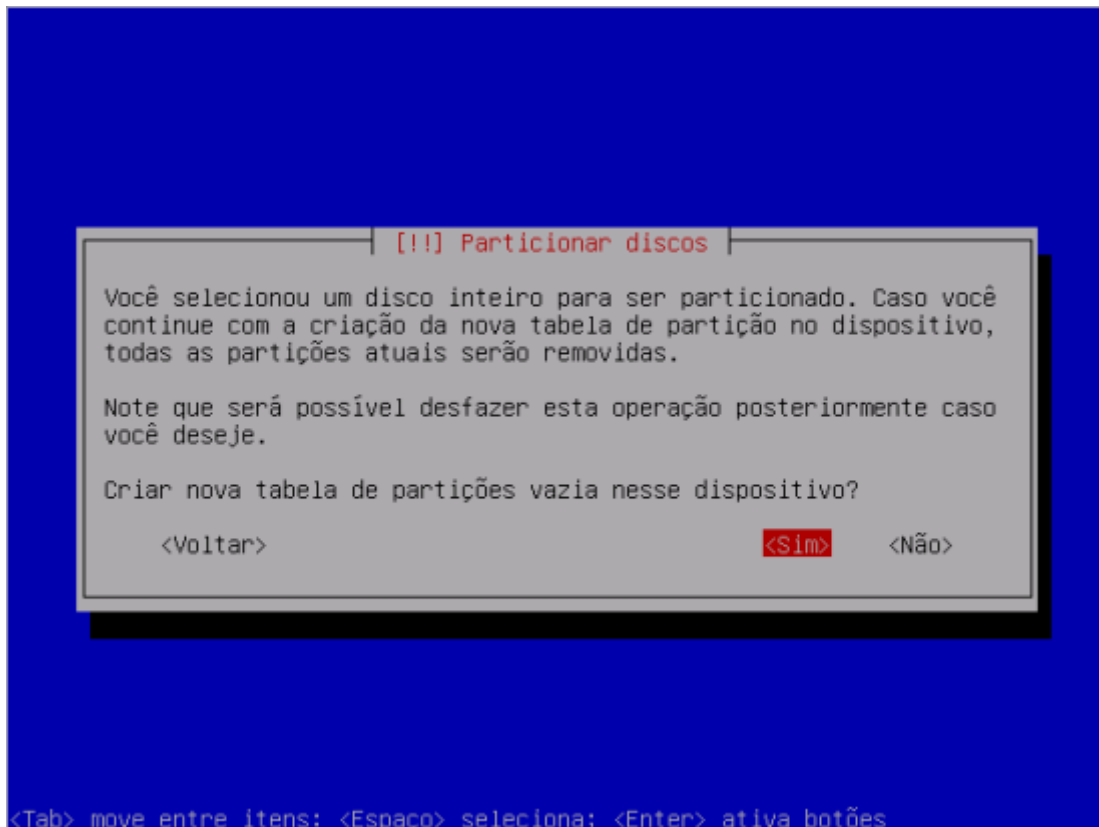
16 – Método de particionamento, escolha: Manual



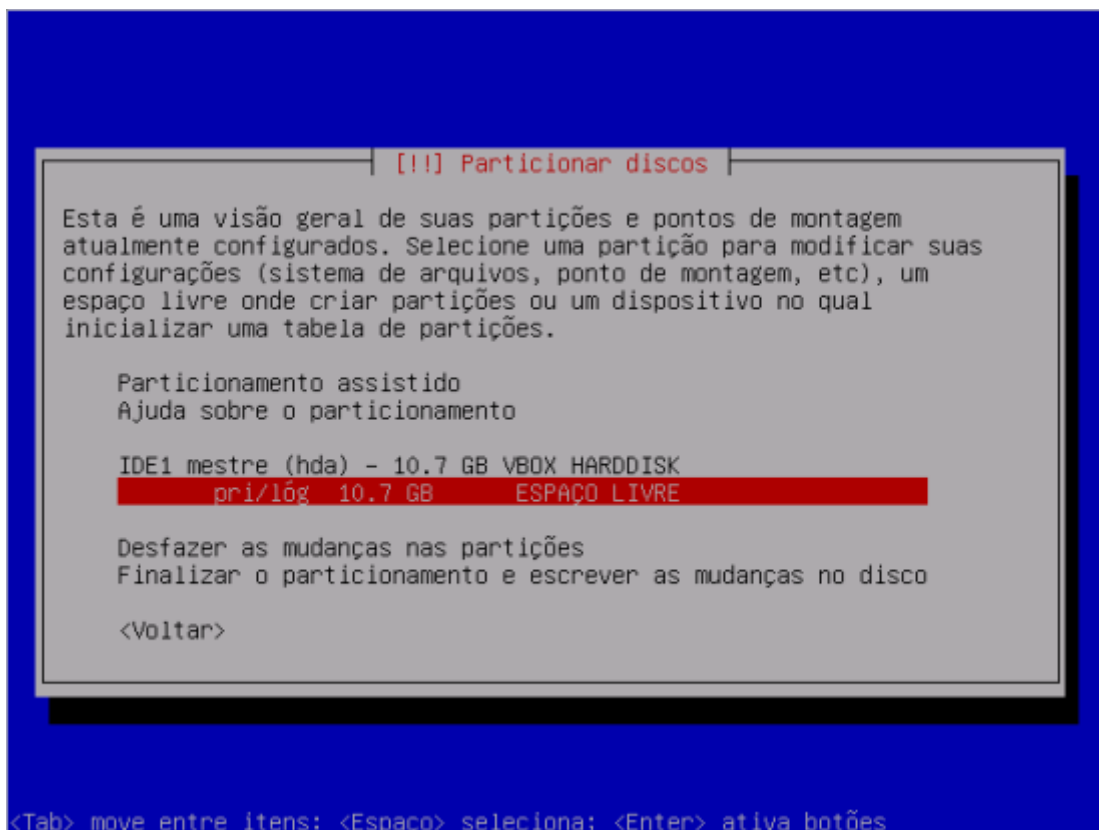
17 – Selecione em qual disco serão criadas as partições



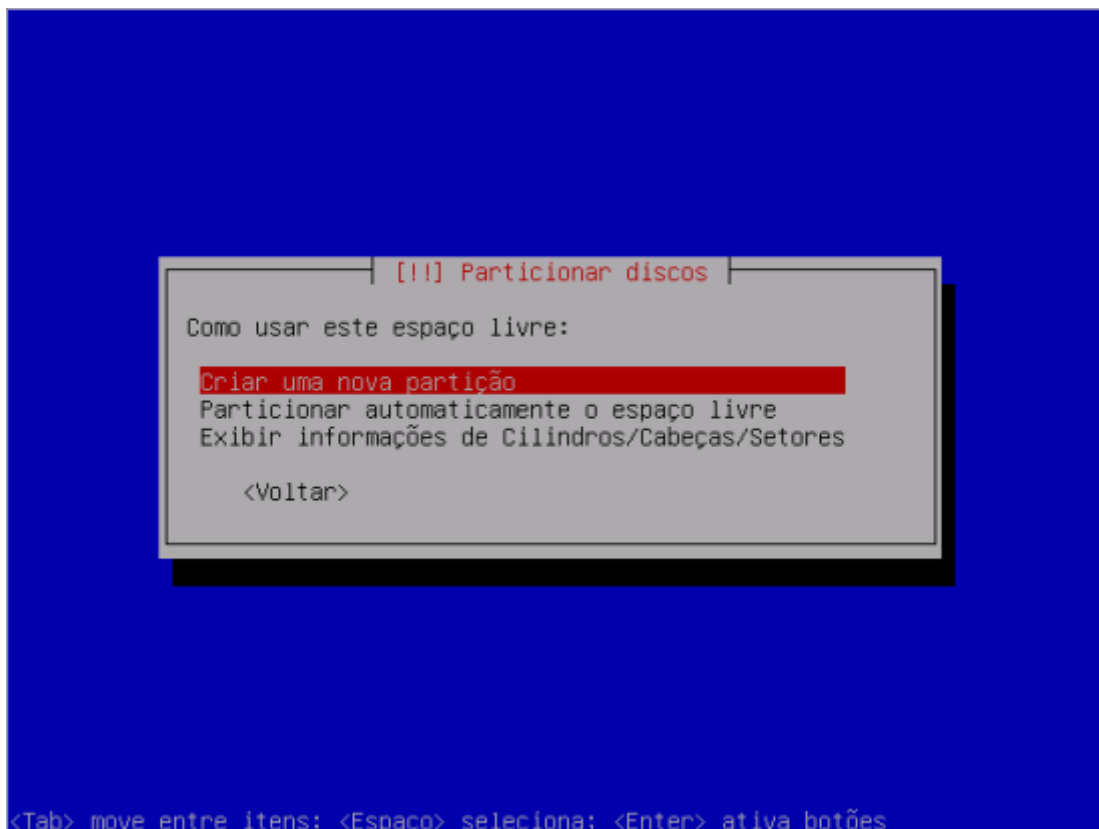
18 – Criar nova tabela de partição vazia nesse dispositivo, escolha: Sim



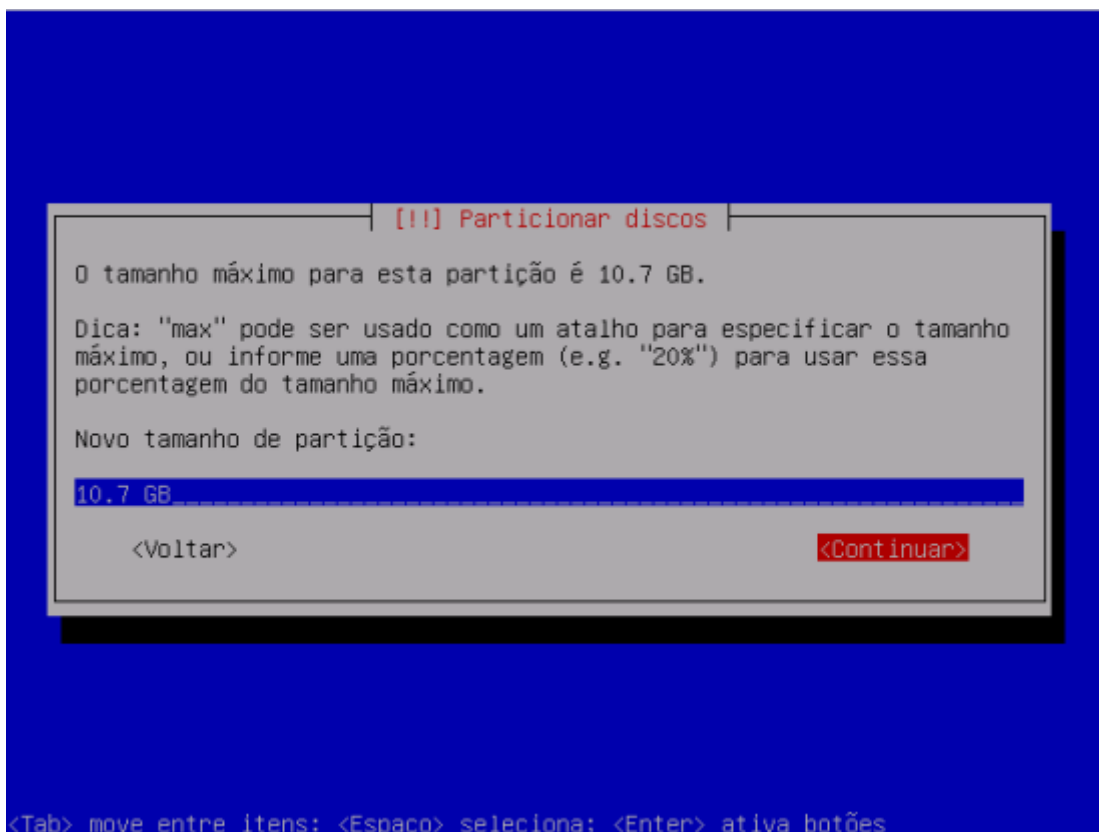
19 – Selecione o espaço livre no disco escolhido



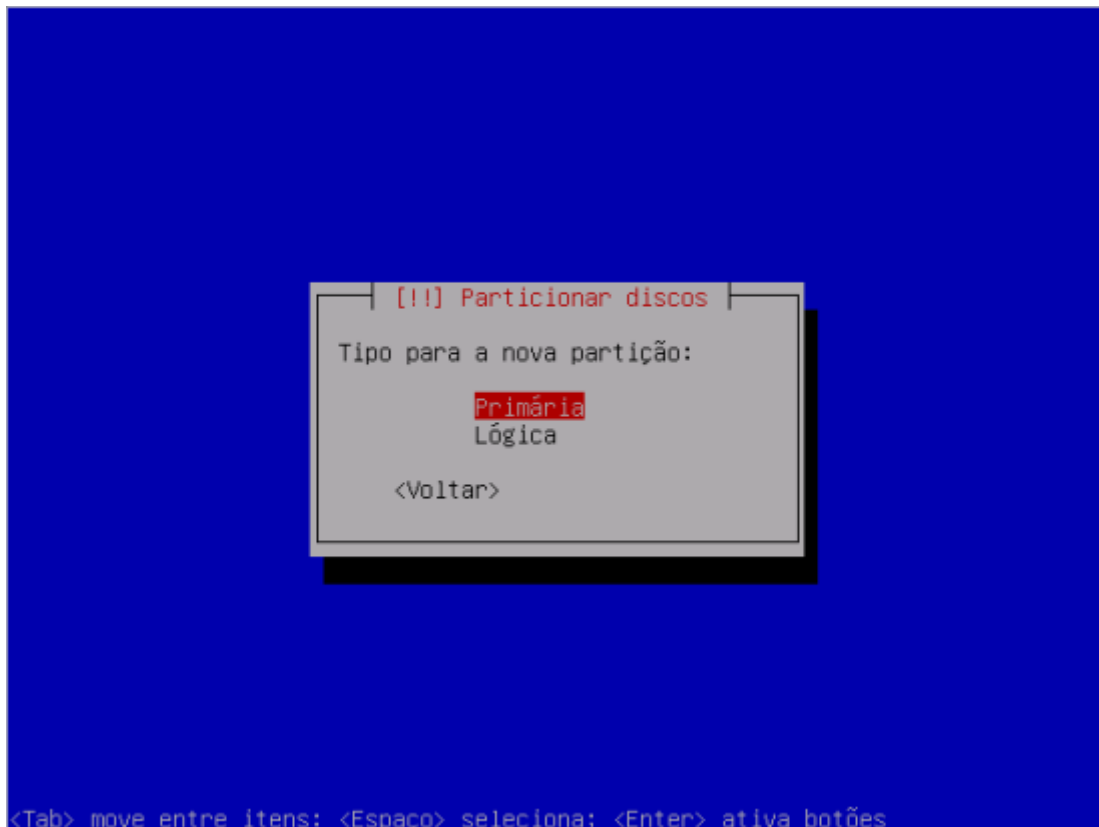
20 – Como usar este espaço livre, escolha: Criar uma nova partição



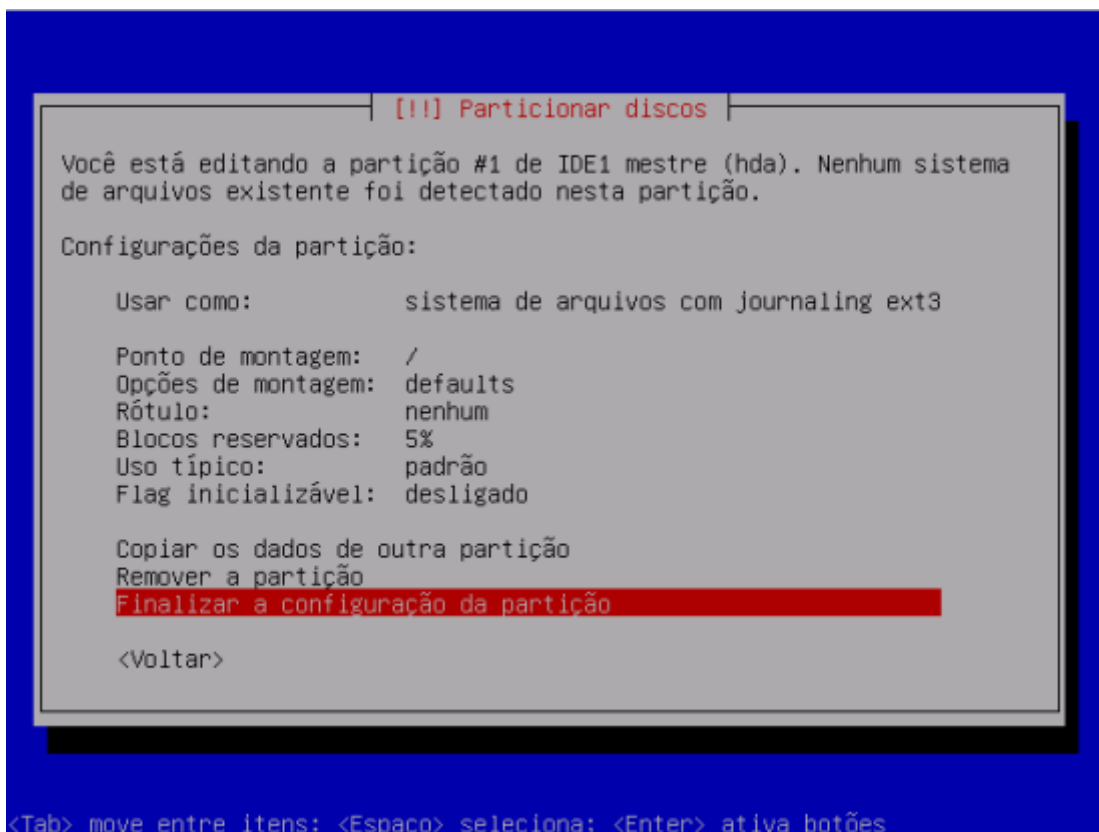
21 – Defina o tamanho da partição



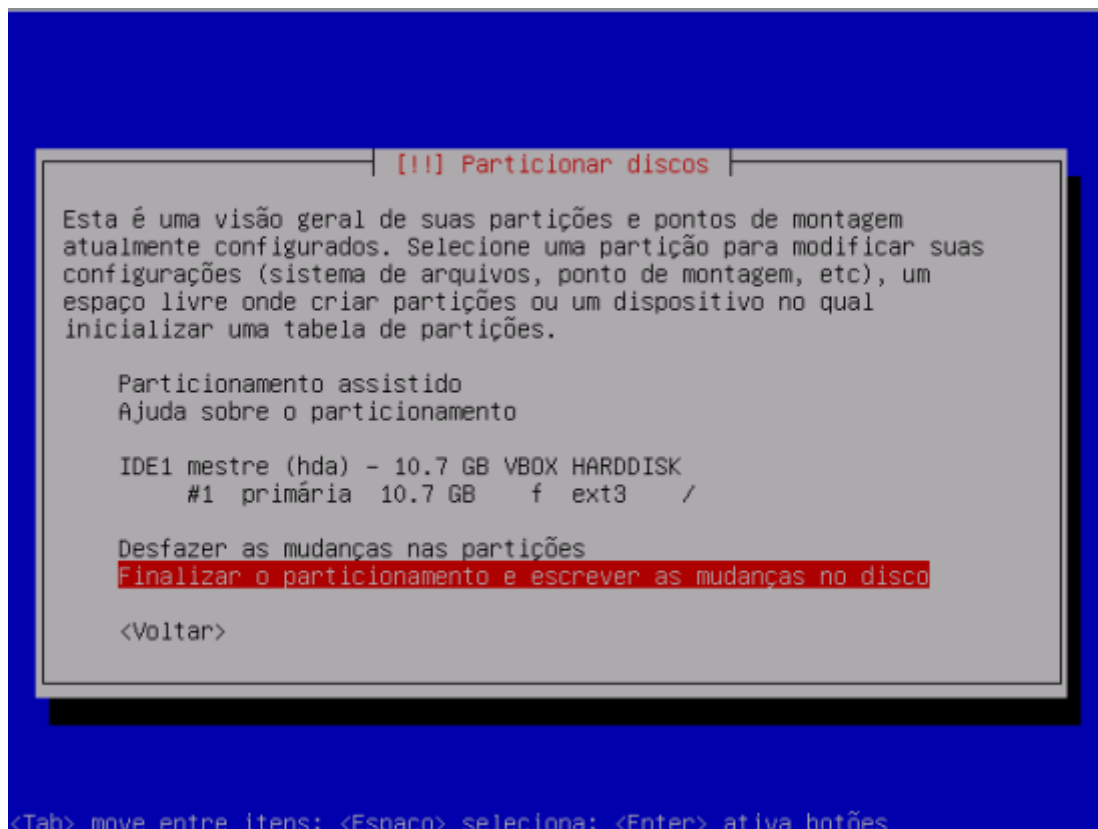
22 – Tipo para a nova partição, escolha: Primária



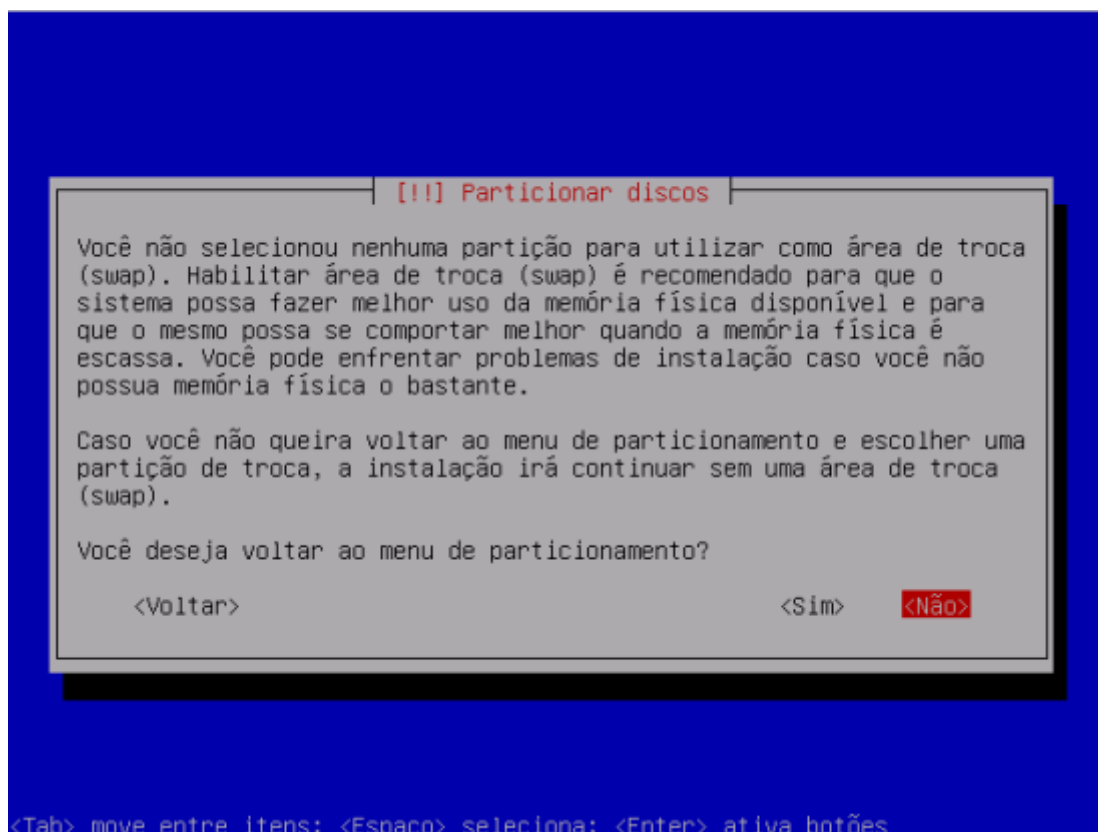
23 – Selecione: Finalizar a configuração da partição



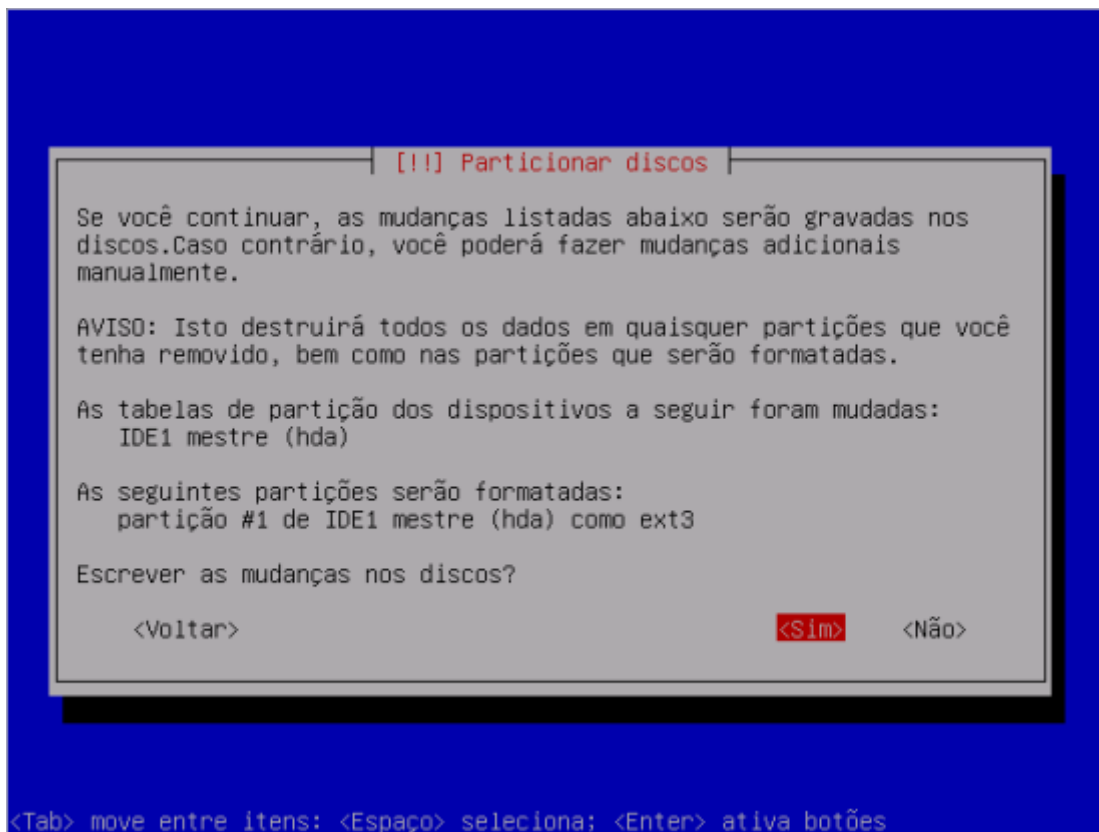
24 – Seleção: Finalizar o particionamento e escrever as mudanças no disco



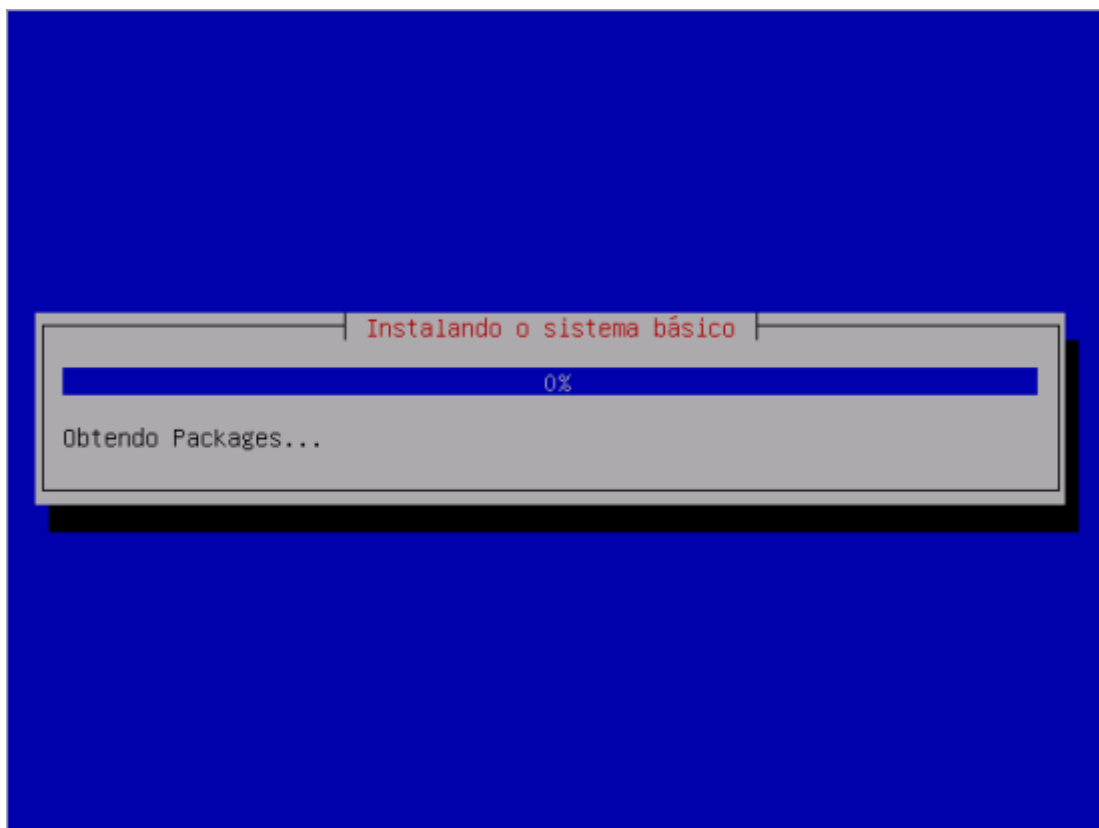
25 – Partição de área de troca swap, selecione: Não



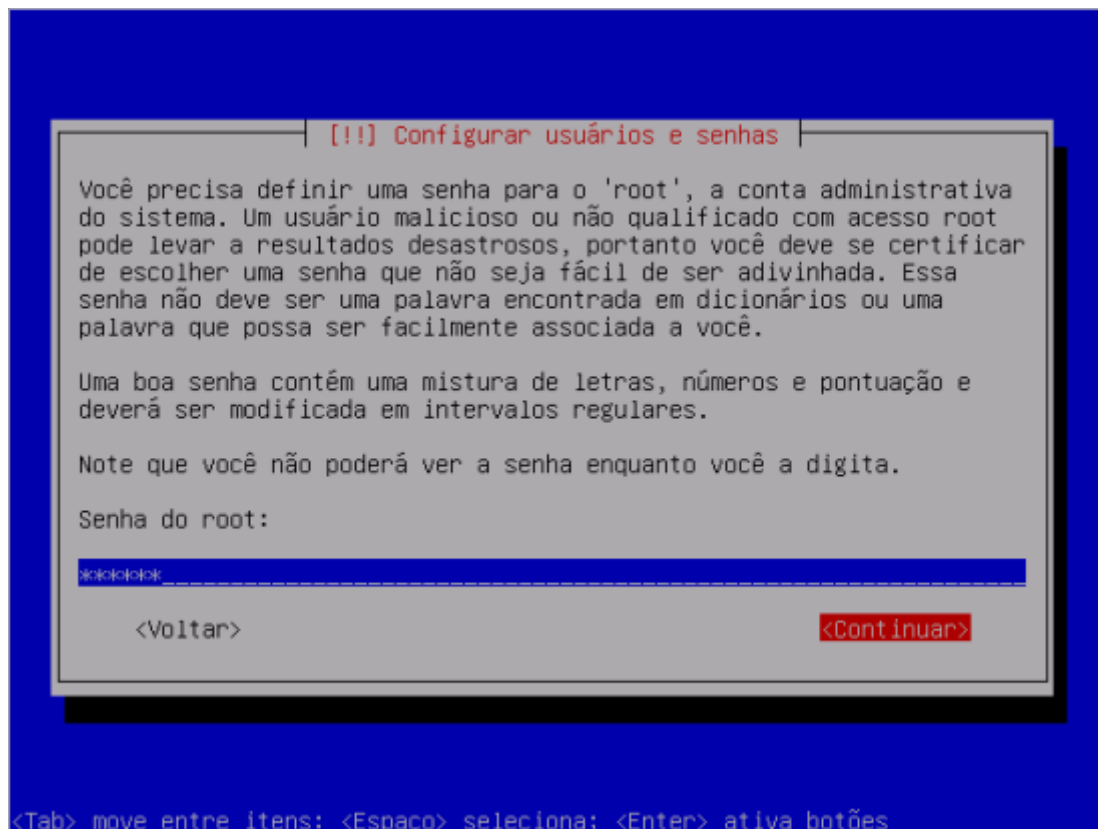
26 – Escrever as mudanças nos discos, escolha: Sim



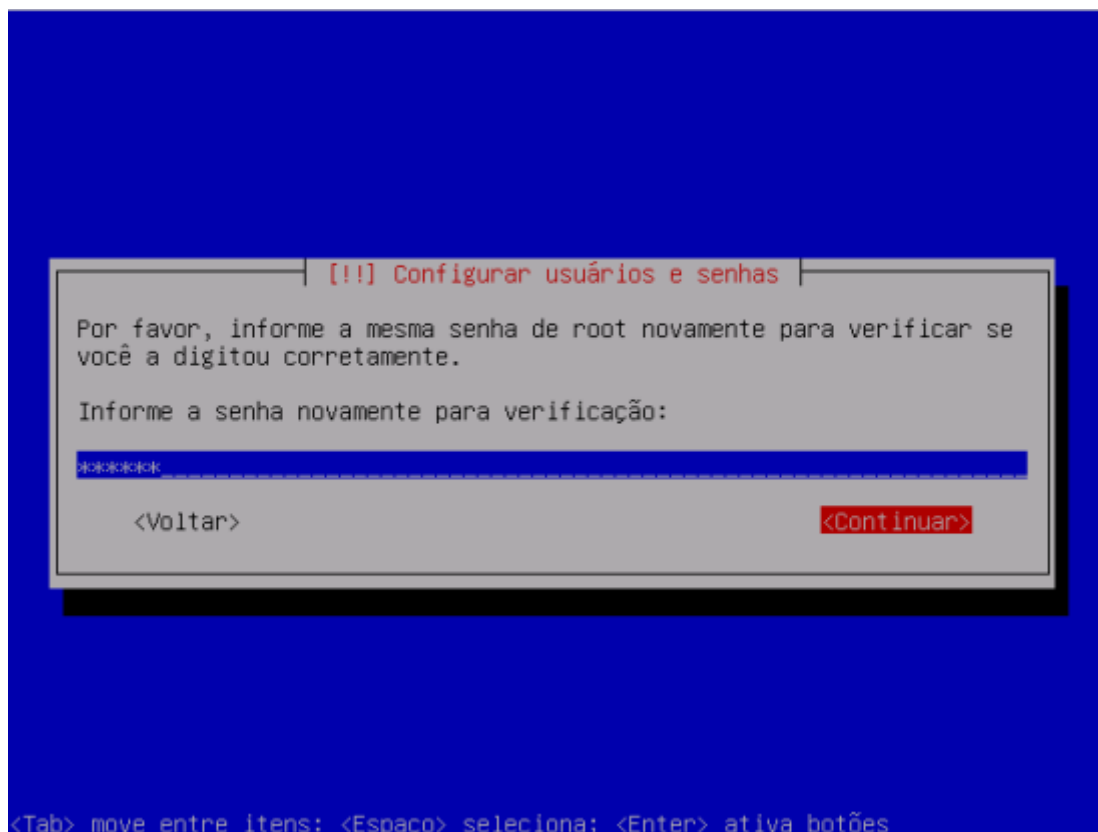
27 – Aguarde enquanto o sistema básico é instalado



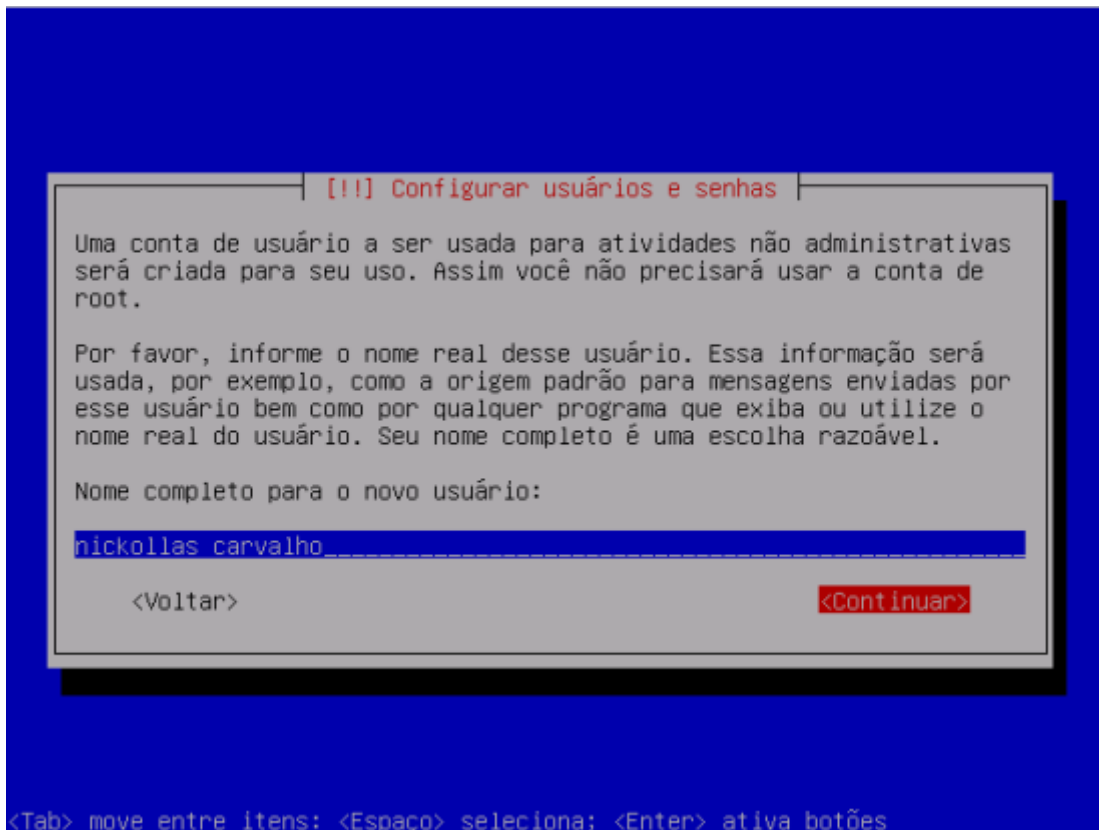
28 – Senha do root, informe a senha de sua escolha



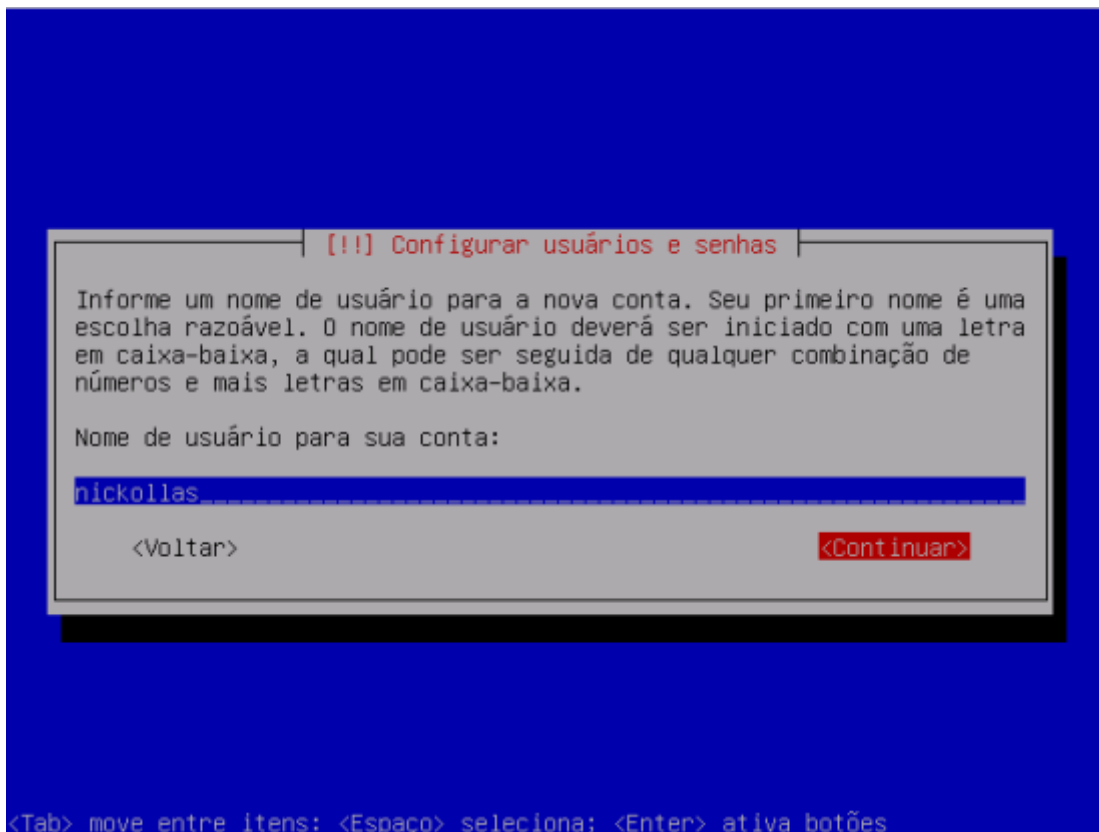
29 – Informe a senha novamente para verificação



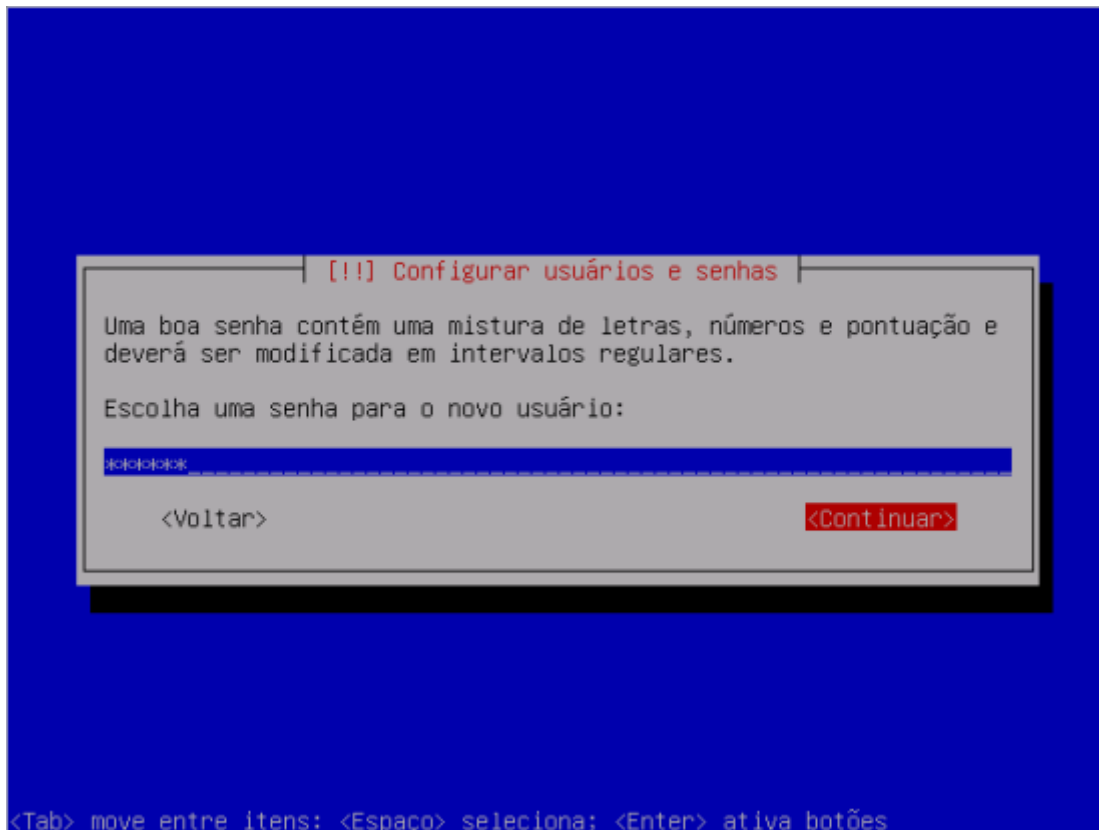
30 – Entre com o Nome completo para o novo usuário



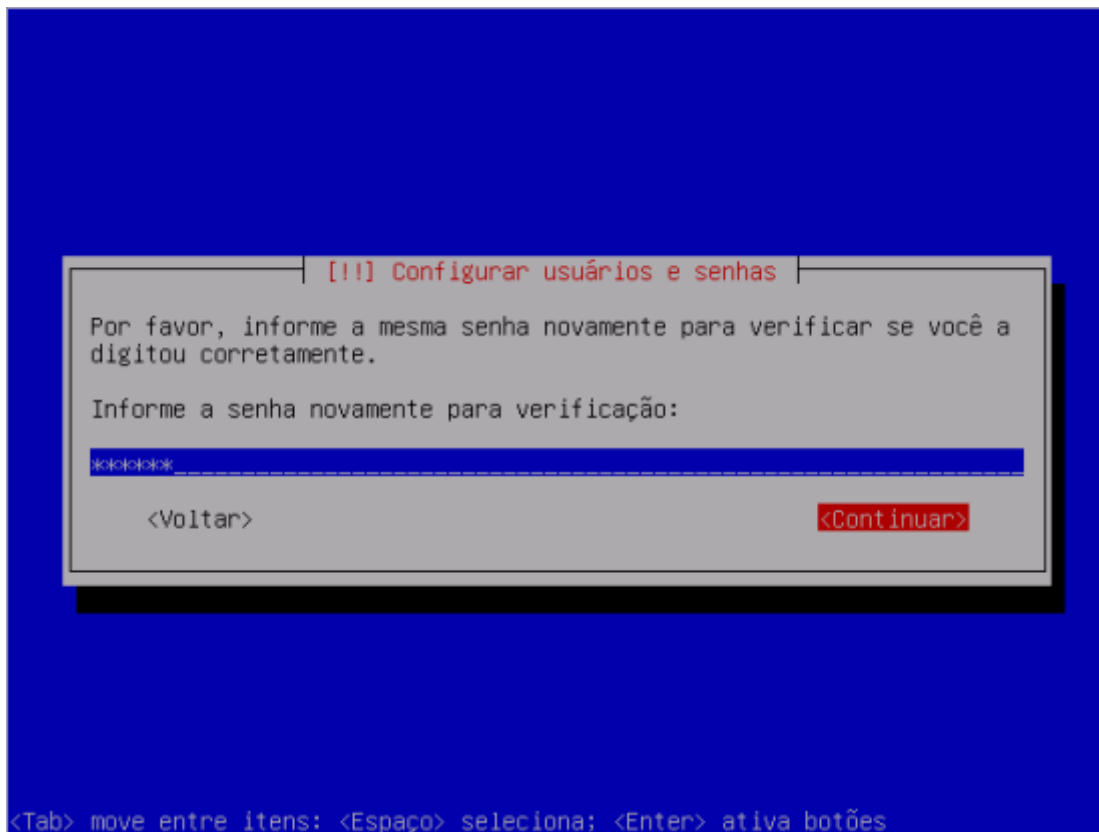
31 – Informe o nome de usuário para a nova conta



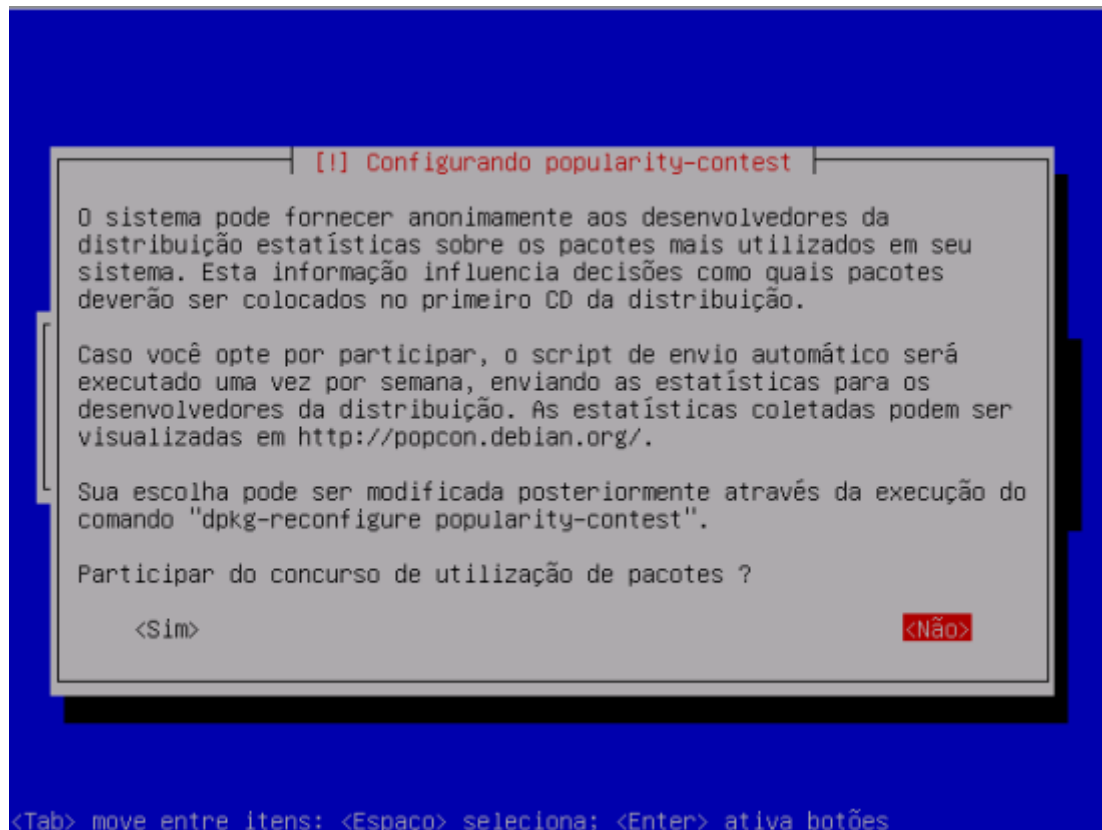
32 – Informe uma senha para o novo usuário



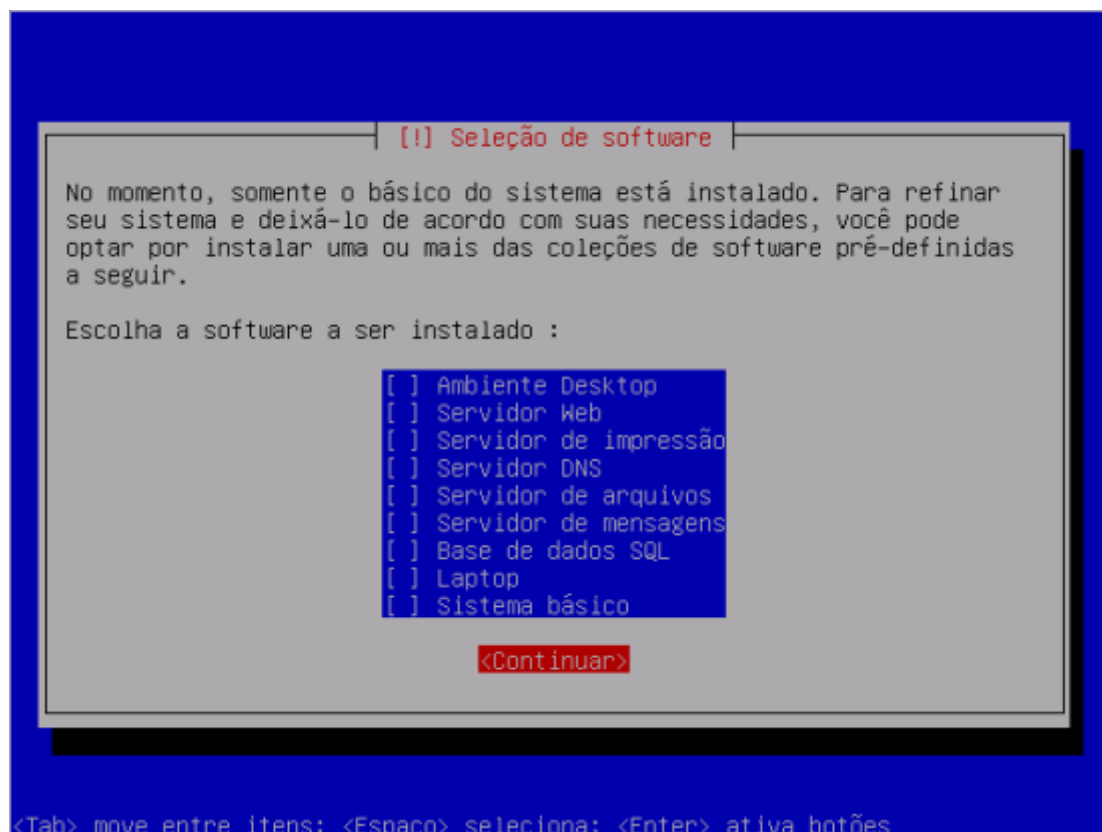
33 – Informe a senha novamente para verificação



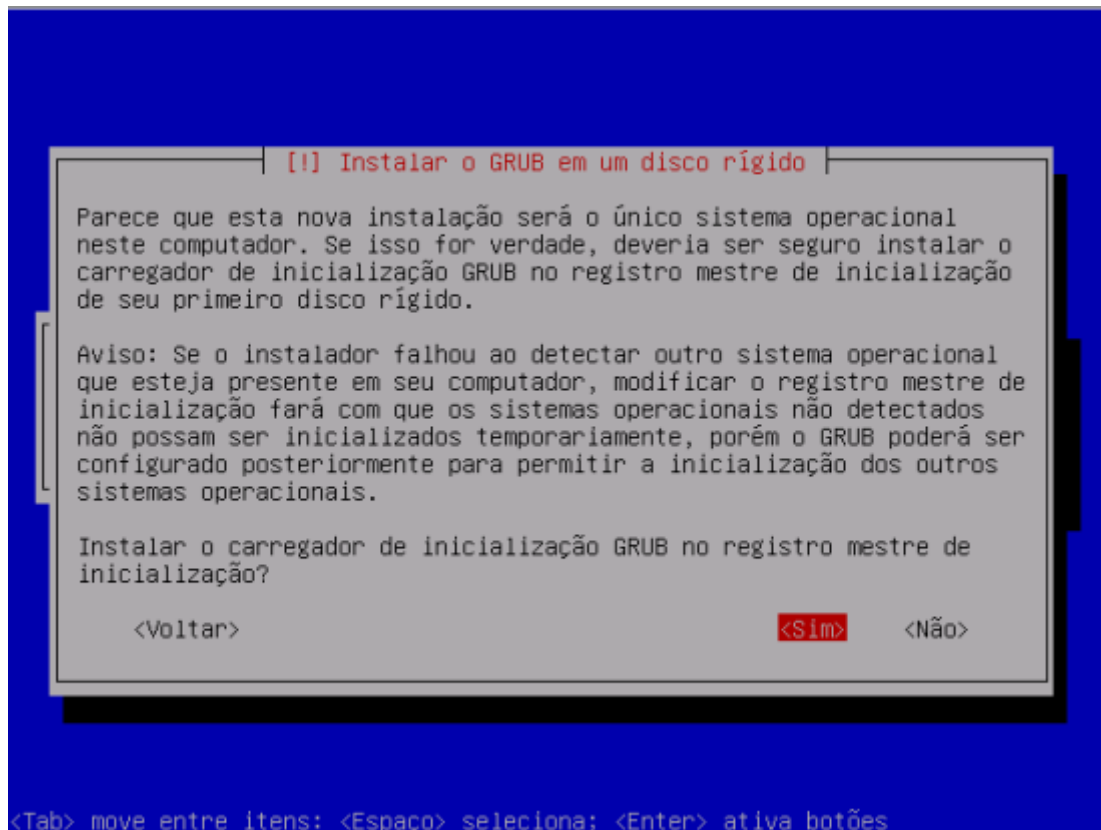
34 – Participar do concurso de utilização de pacotes, escolha: Não



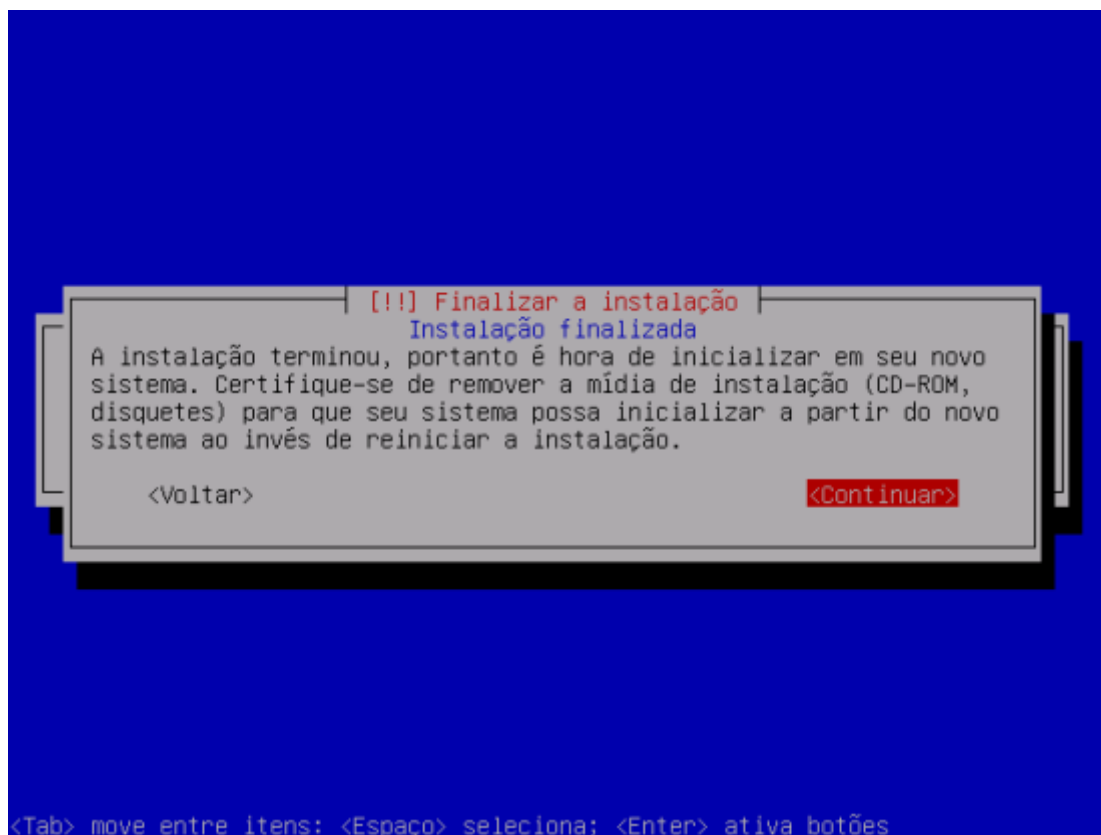
35 - Escolha o software a ser instalado, desmarque todas as opções e selecione: Continuar



36 – Instalar o carregador de inicialização GRUB no registro mestre de inicialização, escolha: Sim



37 – Instalação finalizada, Remova o CD e selecione: Continuar



38 – Assim que a máquina for reiniciada o gerenciador de boot abrirá. Pressione <Enter> ou aguarde

```
GNU GRUB  version 0.97  (639K lower / 261056K upper memory)

Debian GNU/Linux, kernel 2.6.26-2-686
Debian GNU/Linux, kernel 2.6.26-2-686 (single-user mode)

Use the ↑ and ↓ keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.

The highlighted entry will be booted automatically in 1 seconds.
```

39 – Esta é a console do linux, agora você pode efetuar o login.

```
done.
Setting the system clock.
Activating swap...done.
Checking root file system...fsck 1.41.3 (12-Oct-2008)
/dev/hda1: clean, 13868/655360 files, 177750/2620595 blocks
done.
Setting the system clock.
Cleaning up ifupdown...
Loading kernel modules...done.
Checking file systems...fsck 1.41.3 (12-Oct-2008)
done.
Setting kernel variables (/etc/sysctl.conf)...done.
Mounting local filesystems...done.
Activating swapfile swap...done.
Setting up networking...
Configuring network interfaces...done.
Setting console screen modes and fonts.
INIT: Entering runlevel: 2
Starting enhanced syslogd: rsyslogd.
Starting ACPI services...
Starting periodic command scheduler: crond.

Debian GNU/Linux 5.0 debian tty1
debian login: _
```

40 – Após efetuar o login você será levado ao shell (prompt de comando)

```
done.
Setting kernel variables (/etc/sysctl.conf)...done.
Mounting local filesystems...done.
Activating swapfile swap...done.
Setting up networking...
Configuring network interfaces...done.
Setting console screen modes and fonts.
INIT: Entering runlevel: 2
Starting enhanced syslogd: rsyslogd.
Starting ACPI services...
Starting periodic command scheduler: crond.

Debian GNU/Linux 5.0 debian tty1

debian login: root
Password:
Linux debian 2.6.26-2-686 #1 SMP Mon Aug 30 07:01:57 UTC 2010 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
debian:~# _
```

[RAID: Alta Disponibilidade em discos rígidos]

RAID seriam dois ou mais discos (HDs) trabalhando simultaneamente para um mesmo fim com a finalidade de obter backup, segurança e desempenho. Os níveis de RAID mais comuns são:

RAID 0 é uma simples concatenação de partições para criar uma grande partição virtual.

RAID 1 é o nível de RAID que implementa o espelhamento de discos. Para esta implementação são necessários no mínimo dois discos. Seu funcionamento é bem simples, todos os dados são gravados em dois discos simultaneamente se um disco falhar, os dados preservados no outro disco permitem a não descontinuidade da operação do sistema.

[Sistema de Arquivos ext3]

Ext3 (Third Extended file system) é um sistema de arquivos que acrescenta alguns recursos ao Ext2, dos quais o mais visível é o journaling que consiste em um registro (log ou journal) de transações cuja finalidade é recuperar o sistema em caso de desligamento não programado.

[CAPÍTULO 2 - FUNDAMENTOS]

[Console]

Console é o local onde os usuários efetuam o login. Por padrão você tem 6 consoles para efetuar o login. Estas consoles são acessíveis através das teclas ALT+F[1-6]

[Estrutura de Arquivos e Diretórios Linux]

A raiz /

Na raiz ficam todos os arquivos e diretórios do linux, os principais diretórios são: /boot, /etc, /home/, /root, /var, /proc.

O diretório /boot

Neste diretório ficam todos arquivos relacionados ao boot da máquina, isso inclui uma imagem compactada do kernel (vmlinuz-x.x.x) e um arquivo de configuração do gerenciador de boot (grub/menu.lst).

O diretório /etc

Este é o diretório mais importante do linux pois aqui se encontram todos os arquivos de configuração de todos os programas e dispositivos instalados no servidor. O diretório mais usado por um administrador de redes linux é o /etc.

O diretório /home

Em /home ficam os arquivos de todos os usuários do computador que não possuem privilégios de administrador.

O diretório /root

Este é o diretório do super-usuário do linux, o root. Somente ele pode executar tarefas que podem afetar a funcionalidade do sistema.

O diretório /var

Informações variáveis do sistema podem ser encontradas aqui e isso inclui os logs do sistema.

O diretório /proc

O diretório /proc é um sistema de arquivos virtual criado pelo kernel em memória ou seja ele só existe enquanto a máquina estiver ligada. Este diretório especial guarda todos os detalhes sobre o seu sistema Linux, incluindo o kernel, processos, e parâmetros de configuração.

[Limpando a tela]

Ao efetuar o login várias informações estarão na tela, você pode limpar a tela usando o atalho de teclado Crtl+L

[Sintaxe de comandos Linux]

A sintaxe básica de um comando no linux é:

comando [*atributos*] [*parâmetros*]

Os atributos e os parâmetros em alguns comandos são opcionais.

Exemplos:

[root]# ls	# sem atributos e sem parâmetros
[root]# ls -la	# com atributos, sem parâmetros
[root]# ls /	# com parâmetros, sem atributos
[root]# ls -la /	# com atributos e parâmetros

As opções de atributos de um comando pode ser vista acrescentando -help ao final de um comando.

Exemplo:

```
[ root ]# ls -help
```

[Nomeclatura adotada para sintaxe dos comandos]

A nomeclatura adotada para a sintaxe dos comandos foi:

[] - Entre chaves, campo obrigatório, **não pode** ser omitido

[/] - Entre chaves itálico, opção opcional, **pode** ser omitida

Exemplo:

Sintaxe: comando [*atributos*] [*arquivo*] # atributos opcionais, arquivo obrigatório

[Manipulação de arquivos e diretórios]

O comando ls

Lista arquivos e diretórios.

Sintaxe: ls [*atributos*] [*arquivo / diretório*]

Principais atributos: -a -l -h

Exemplo: ls arquivo

O comando cp

Copia arquivos / diretórios

Sintaxe: cp [*atributos*] [*arquivo / diretório*] [*arquivo / diretório*]

Principais atributos: -r -f

Exemplo: cp -f arquivo arquivo_backup
cp -rf diretorio diretorio_backup

faz uma copia de arquivo
faz uma cópia de diretorio

O comando du

Utilizado para: visualizar o tamanho de arquivos e diretórios

Sintaxe: du [*atributos*] [*arquivo / diretório*]

Principais atributos: -h

Exemplo: du -h arquivo

O comando mv

Move ou renomeia arquivo / diretorio

Sintaxe: mv [*arquivo / diretório*] [*arquivo / diretório*]

Exemplo: mv arquivo /tmp
mv arquivo arq

move arquivo para /tmp
renomeia arquivo para arq

O comando rm

Remove arquivo / diretório

Sintaxe: `rm [atributos] [arquivo / diretório]`

Principais atributos: `-r -f`

Exemplo: `rm -f arquivo`
`rm -rf diretório`

`# apaga arquivo`
`# apaga diretório`

O comando `mkdir`

Cria diretório

Sintaxe: `mkdir [atributos] [diretório]`

Principais atributos: `-p`

Exemplo: `mkdir diretório`

O comando `cd`

Entra em diretório

Sintaxe: `cd [opções][diretório]`

Principais opções: `-, ~`

Exemplo: `cd /etc`

[Imprimindo caracteres na saída padrão]

O comando `echo`

Imprime uma linha de texto na saída padrão

Sintaxe: `echo [atributos] [texto]`

Principais atributos: `-e`

Exemplos: `echo "meu texto"`
`echo -e "meu texto\n"`

`# imprime: meu texto`
`# imprime: meu texto + linha em branco`

[Operadores da Shell]

O operador `&`

Executa comandos em background

Sintaxe: [comando] [&]

Exemplo: find / -name casa &

O operador &&

Executa o comando em sequência.

Sintaxe: [comando] [&&] [comando]

Exemplo: ls && ls -a

[Entrada e Saída de dados]

O operador >

Pega a saída padrão e a armazena dentro de um arquivo apagando qualquer informação existente.

Exemplo: echo “conteúdo” > arquivo

O operador >>

Pega a saída padrão e a concatena dentro de um arquivo não apagando qualquer informação já existente.

Exemplo: echo “novo conteúdo” >> arquivo

O operador |

Retem o conteúdo da saída padrão e o repassa para o próximo comando

Exemplo: cat /etc/services | more

[Controle de fluxo]

O comando more

Pagina o conteúdo de arquivo

Sintaxe: more [arquivo]

Exemplo: more arquivo

[Porções específicas]

O comando tail

Mostra as últimas linhas de um arquivo

Sintaxe: tail [*atributos*] [arquivo]

Principais atributos: -f -n

Exemplo: tail /etc/services

O comando head

Mostra as primeiras linhas de um arquivo

Sintaxe: head [*atributos*] [arquivo]

Principais atributos: -n

Exemplo: head /etc/services

[Contagem]

O comando wc

Conta quantas linhas tem um arquivo

Sintaxe: wc [*atributos*] [arquivo]

Principais atributos: -l

Exemplo: wc -l arquivo

[Filtragem]

O comando grep

Filtra conteúdo da saída padrão / arquivo

Sintaxe: grep [*atributos*] [arquivo]

Principais atributos: -v

Exemplo: grep root /etc/passwd

[Localização de arquivos e diretórios]

O comando find

Localiza arquivos e diretórios no sistema

Sintaxe: find [diretório de busca] [atributos] [arquivo / diretório]

Principais atributos: -name, -iname

Exemplo: find / -name passwd

[Símbolos Curinga]

Símbolos curinga são operadores que oferecem maior poder ao tratamento de informações. Eles são utilizados em conjunto com comandos do sistema.

O símbolo curinga: ^

Casa com o começo da linha

Exemplo: grep ^s /etc/passwd

O símbolo curinga: \$

Casa com o fim da linha

Exemplo: grep false\$ /etc/passwd

O símbolo curinga: ?

Casa 1 vez com qualquer caracter

Exemplo: find / -name passwd?

O símbolo curinga: *

Casa 0 ou mais vezes com qualquer caracter

Exemplo: find / -name passwd*

[Diferença entre arquivos]

O comando diff

Mostra a diferença entre arquivos

Sintaxe: diff [arquivo] [arquivo]

Exemplo: diff arquivo1 arquivo2

[Gerenciamento de memória RAM]

O comando free

Mostra a quantidade de memória utilizada e livre contém no sistema.

Sintaxe: free [*atributos*]

Principais atributos: -m

Exemplo: free

[Mostrar informações sobre o sistema]

O comando uname

Mostra informações sobre o sistema

Sintaxe: uname [*atributos*]

Principais atributos: -a -r

Exemplo: uname -a

[Mostrar / ajustar a data do sistema]

O comando date

Mostra / ajusta a data sistema.

Sintaxe: date [*atributos*]

Principais atributos: -s

Exemplo: date -s "09/15/2010 19:11" # altera data para 15 de Setembro de 2010 às 19:11

[**Mostrar por quanto tempo o computador está ligado**]

O comando uptime

Mostra por quanto tempo o computador está ligado.

Exemplo: uptime

[**Tempo de execução de um programa**]

O comando time

Mostra o tempo de execução de um comando.

Sintaxe: time [comando]

Exemplo: time ls /etc

[**Conhecendo a Documentação**]

O comando man

Mostra a documentação / página de manual de um comando

Sintaxe: man [comando]

Exemplo: man ls

O atributo --help

Usado quando se quer ter uma ajuda rápida sobre o funcionamento de qualquer comando.

Sintaxe: [comando] [--help]

Exemplo: ls --help

[**Editores de texto**]

O editor vim

Editor arquivos

Sintaxe: vim [arquivo]

Principais comandos: insert, delete, dd, /, :[n], :%s/[caracteres]/g, :x, :q

Exemplo: vim arquivo

[Configuração de rede]

O arquivo /etc/udev/rules.d/*-persistent-net.rules

Contém informações físicas e lógicas de todas as placas de rede presentes na máquina

O comando ifconfig

Mostra informações e configura dispositivos de rede da máquina

Sintaxe: ifconfig [interface] [opções]

Principais opções: up, down

Exemplo: ifconfig eth1 172.16.0.1 netmask 255.255.255.0
ifconfig eth1 up

O arquivo /etc/network/interfaces

Contém informações sobre a configuração das placas de rede

Principais entradas: auto, iface, address, netmask, gateway

O arquivo /etc/resolv.conf

Mantém informações sobre os servidores de nomes (DNS) que são utilizado pelo máquina

Principais entradas: nameserver

O comando route

Configura o gateway padrão da máquina

Principais comandos: -n, add default gw, del

Exemplo: route add default gw 192.168.0.254

O arquivo /etc/hosts

Resolve nomes através das configurações inseridas

[Gerenciamento de rede]

O comando mii-tool

Mostra o status das interfaces de rede configuradas na máquina

Exemplo: mii-tool

O comando ping

Envia echo_requests para hosts de rede

Exemplo: ping 192.168.0.254

[Manipulando dispositivos de armazenamento]

O comando dmesg

Mostra o buffer do kernel

Exemplo: dmesg

O comando mount

Monta sistemas de arquivos

Sintaxe: mount [*atributos*] [*sistema de arquivos*]

Principais atributos: -t, -a

Exemplo: mount -t /dev/sdb1 /mnt

[Gerenciamento de partições]

O comando df

Mostra a quantidade de espaço em disco usada e livre

Sintaxe: df [*atributos*]

Principais atributos: -h -T

Exemplo: df -h

O comando cfdisk

Manipula partições em discos rígidos

O comando mkfs

Cria sistema de arquivos em uma partição

Sintaxe: mkfs [atributos] [partição]

Principais atributos: -t

Exemplo: mkfs -t ext3 /dev/hda5

O arquivo /etc/fstab

Contém informações estáticas sobre sistemas de arquivos montados

[Gerenciamento de processos]

O comando ps

Mostra informações sobre os processos correntes

Sintaxe: ps [opções]

Principais opções: ax

Exemplo: ps ax

O comando top

Mostra informações sobre os processos correntes em tempo real

Exemplo: top

O comando kill

Mata um processo através de seu pid

Sintaxe: kill [sinal] [pid]

Principais sinais: -9

Exemplo: kill -9 1379

O comando killall5

Mata um processo através de seu nome

Sintaxe: killall5 [sinal] [nome]

Principais sinais: -9

Exemplo: killall5 -9 bash

[Prioridades dos processos]

O comando nice

Define a prioridade de um processo

Sintaxe: nice [atributos] [prioridade] [comando]

Principais atributos: -n

Principais prioridades: -20

Exemplo: nice -n -20 ls

O comando renice

Redefine a prioridade de um processo

Sintaxe: renice [prioridade] [atributos] [pid]

Principais prioridades: -p

Exemplo: renice -20 -p 1317

[CAPÍTULO 3 - ADMINISTRAÇÃO]

[Gerenciamento de login]

O comando w

Mostra quem está logado no sistema

Exemplo: w

O comando lastlog

Informa em que data foi o último login dos usuários

Sintaxe: lastlog [*atributos*] [*usuário*]

Principais atributos: -u

Exemplo: lastlog

[Permissões em arquivos e diretórios]

O comando chmod

Altera as permissões de um arquivo ou diretório

Sintaxe: chmod [*modo*] [*arquivo / diretório*]

Modos: 4 2 1 (r w x) / ugoa

Exemplo: chmod 740 arquivo

[Gerenciando usuários e grupos]

O arquivo /etc/passwd

Contém informações dos usuários como: login, nome, id, diretório de arquivos, shell e outros

O arquivo /etc/shadow

Contém as senhas criptografadas de todos os usuários

O arquivo /etc/group

Contém informações sobre os grupos de cada usuário

O diretório /etc/skel

Contém os arquivos que compõem o estrutura básica do /home de um usuário

O arquivo /etc/nologin

Arquivo utilizado para dar manutenção no servidor, se criado nenhum usuário conseguirá fazer login

[Alterando dono de arquivos e diretórios]

O comando chown

Altera o dono e grupo de um arquivo / diretório

Sintaxe: `chown [atributos] [usuário] [.] [grupo] [arquivo / diretório]`

Principais atributos: -R

Exemplo: `chown daemon arquivo`
`chown -R daemon diretório`

[Alterando grupo de arquivos e diretórios]

O comando chgrp

Altera o grupo de um arquivo / diretório

Sintaxe: `chgrp [atributos] [grupo] [arquivo / diretório]`

Principais atributos: -R

Exemplo: `chgrp -R users diretório`

[Manipulando senhas de usuários]

O comando passwd

Define / altera a senha de um usuário

Sintaxe: `passwd [atributos] [usuário]`

Principais atributos: `-d`

Exemplo: `passwd root`

[Adicionando usuário]

O comando `useradd`

Adiciona um usuário no sistema

Sintaxe: `useradd [atributos] [usuário]`

Principais atributos: `-g -s`

Exemplo: `useradd aluno -g users -s /bin/false`

Criando ambiente do usuário

Depois de adicionar o usuário no sistema seu ambiente no sistema deve ser criado:

- Criar seu diretório pessoal: `mkdir /home/aluno`
- Copiar o skel para o home do usuário: `cp /etc/skel/* /home/aluno`
- Alterar o dono – diretório e arquivos de seu home : `chown -R aluno.users /home/aluno/`
- Definir uma senha para o usuário: `passwd aluno`

[Alterando contas de usuário]

O comando `usermod`

Altera conta de um usuário

Sintaxe: `usermod [atributos] [usuário]`

Principais atributos: `-G`

Exemplo: `usermod -G users,audio aluno`

[Informações dos usuários]

O comando `id`

Mostra a identidade do usuário

Sintaxe: id [*usuário*]

Exemplo: id aluno

[Removendo Usuário]

O comando userdel

Remove usuário do sistema

Sintaxe: userdel [*atributos*] [usuário]

Principais atributos: -r

Exemplo: userdel -r aluno

[Atributos de arquivos e diretórios]

O comando lsattr

Mostra os atributos de arquivos e diretórios

Sintaxe: lsattr [arquivo / diretório]

Exemplo: lsattr arquivo

O comando chattr

Altera atributos de arquivos e diretórios

Sintaxe: chattr [+| -] [atributos] [arquivo / diretório]

Principais atributos: i

Exemplo: chattr +i arquivo

[Gerenciamento de pacotes com cálculo de dependência – apt]

O comando apt-get

Instala e atualiza pacotes

Sintaxe: apt-get [opções] [pacote]

Principais opções: update, install, remove, clean, autoremove

Exemplo: apt-get update
apt-get install vim-full

O comando apt-cache

Pesquisa pacotes na base de dados do apt

Sintaxe: apt-cache [opções] [texto]

Principais opções: search

Exemplo: apt-cache search kernel

O comando apt-cdrom

Adiciona novo cd-rom na base de dados do apt

Sintaxe: apt-cdrom [opções]

Principais opções: add, remove

Exemplo: apt-cdrom add

[Gerenciamento de pacotes - dpkg]

O comando dpkg

Mostra informações, instala e remove pacotes

Sintaxe: dpkg [atributos] [*pacote*]

Principais atributos: -l, -L -i, -r

Exemplo: dpkg -l

O atributo --purge do comando dpkg

Remove todos os vestígios de um pacote removido

Sintaxe: dpkg --purge [pacote]

Exemplo: dpkg --purge nis

O comando dpkg-reconfigure

Reconfigura um pacote já instalado

Sintaxe: dpkg-reconfigure [pacote]

Exemplo: dpkg-reconfigure locales

[Informações sobre arquivos]

O atributo -S do comando dpkg

Diz a qual pacote pertence um arquivo

Exemplo: dpkg -S /bin/ls

O comando file

Informa qual o tipo do arquivo

Sintaxe: file [arquivo]

Exemplo: file /etc/passwd

[Atualização via Internet]

A opção upgrade do comando apt-get

Atualiza todo o sistema, instalando novas versões de pacotes

Exemplo: apt-get upgrade

[Ativando syntax no vim]

O arquivo /etc/vim/vimrc

Arquivo de configuração do vim

Principais entradas: syntax on

[Empacotadores]

O pacote bzip2

Compressor de arquivos, que pode ser utilizado em conjunto com o comando tar.

O comando tar

Arquiva, comprime e descomprime arquivos / diretórios

Sintaxe: tar [atributos] [-f] [arquivo] [arquivo / diretório]

Principais atributos: -x -c, -j, -v, -f

Exemplo: tar -cjvf arquivo.tar.bz2 diretorio

[Download de arquivos]

O comando wget

Faz download de arquivos na internet através de uma URL.

Sintaxe: wget [*atributos*] [URL]

Principais atributos: -c, -r

Exemplo: wget <http://nmap.org/dist/nmap-5.21.tar.bz2>

[Compilação de programas]

O comando ./

Executa script ou binário do diretório atual

Sintaxe: [./] [arquivo]

Exemplo: ./configure

O arquivo configure

Fornece opções de compilação para programas gerando arquivos Makefile. Se omitido, usará opções padrão.

Syntaxe: [./] [configure] [*atributos*]

Exemplo: `./configure --without-liblua`

O comando make

Compila automaticamente um programa escrito em C através de arquivos Makefile que informam como o código deve ser compilado.

Sintaxe: `make [opções]`

Principais opções: `install`

Exemplo: `make`
`make install`

A opção install do comando make

Configura o programa para rodar na máquina copiando / movendo binários e documentação para o PATH do sistema.

Exemplo: `make install`

O comando gcc

Compilador de programas escritos em C.

Sintaxe: `gcc [arquivo fonte] [atributos] [arquivo]`

Principais atributos: `-o`

Exemplo: `gcc wipe.c -o wipe`

[Gerenciamento de Hardware e Dispositivos]

O comando lspci

Lista todos dispositivos PCI instalados na máquina.

Sintaxe: `lspci [atributos]`

Principais atributos: `-v`

Exemplo: `lspci`

[Gerenciamento de módulos do kernel]

O diretório /lib/modules

Contém todos os módulos que foram compilados para as versões do kernel

O comando lsmod

Lista todos os módulos carregados no kernel

Exemplo: lsmod

O comando modprobe

Carrega um módulo presente em /lib/modules no kernel

Exemplo: modprobe ip_tables

O comando rmmod

Remove um módulo do kernel

Exemplo: rmmod ip_tables

O comando insmod

Carrega um módulo compilado pelo usuário no kernel

Sintaxe: insmod [módulo]

Exemplo: insmod modulo

[Níveis de Execução]

O comando runlevel

Mostra em qual nível o sistema está operando.

Exemplo: runlevel

O comando init

Define em qual nível o sistema será carregado

Sintaxe: init [runlevel]

Exemplo: init 5

O arquivo /etc/inittab

Contém configurações sobre níveis do sistema e sobre os terminais / consoles disponíveis para login

[Agendamento de tarefas]

O cron

Cron é um agendador de tarefas utilizado para executar comandos ou scripts para rodarem periodicamente durante um certo tempo ou data.

O comando crontab

Crontab é o comando que manipula / configura o cron.

Syntaxe: crontab [-u] [usuário] [atributos]

Principais atributos: -l -e -r

Exemplo: crontab -u root -e

[Agendando uma tarefa]

Configurando o crontab

As entradas de um arquivo crontab são:

Minuto	Hora	Dia_do_mes	Mes	Dia_da_semana	Comando
*	*	*	*	*	*

Exemplo:

```
# minuto hora dia_do_mes mes dia_da_semana comando
# Roda o script a cada 5 min
0-59/5 * * * * /usr/local/bin/scriptcron
```

[Gerenciador de Boot GRUB]

GRUB é um multi-gerenciador de boot de sistemas operacionais. É utilizado, normalmente, quando se deseja que um computador tenha dual booting, ou seja, que o usuário possa escolher ao iniciar a máquina, um sistema operacional (SO) dentre dois ou mais sistemas instalados.

O arquivo `/boot/grub/menu.lst`

Arquivo de configuração do GRUB

Principais entradas: `default`, `timeout`, `color`, `title`, `root`, `kernel`, `initrd`

[Utilitários]

O comando `watch`

Executa um comando infinitamente

Sintaxe: `watch [comando]`

Principais atributos: `-n`

Exemplo: `watch ls`

O comando `cal`

Calendário

Sintaxe: `cal [ano]`

Exemplo: `cal`

O comando `bc`

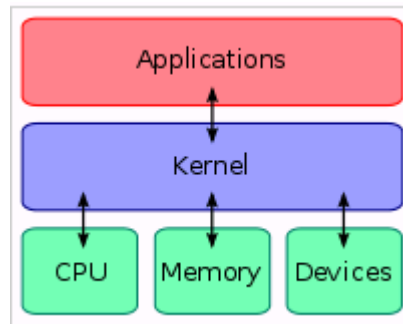
Calculadora

Principais comandos: `quit`

Exemplo: `bc`

[O kernel]

O kernel é o componente principal da maioria dos sistemas operacionais, é uma ponte entre o hardware e as aplicações. As responsabilidades do kernel incluem a gestão dos recursos do sistema como uso de CPU e memória além de gerenciar todo hardware instalado na máquina.



[Compilando o kernel]

1 - Instale os pacotes libncurses5-dev gcc libncurses5-dev make patch bzip2

```
apt-get install libncurses5-dev gcc libncurses5-dev make patch bzip2
```

2 - Baixe o código fonte do kernel em www.kernel.org

```
wget http://www.kernel.org/pub/linux/kernel/v2.6/linux-2.6.35.4.tar.bz2
```

3 - Descompacte o código fonte do kernel em "/usr/src"

```
tar -xjvf linux-2.6.17.8.tar.bz2 -C /usr/src
```

```
cd /usr/src/
```

4 - Crie um link simbólico do diretório que contém o código fonte para "linux"

```
ln -sf linux-2.6.17.8 linux
```

5 - Entre em "/usr/src/linux"

```
cd linux
```

6 - Copie o .config do kernel de produção para o diretório local

```
cp /boot/config-2.6.26-2-686 .config
```

7 - Selecione as opções de compilação do kernel

```
make menuconfig
```

* Marque as opções de compilação conforme sua preferência

ATENÇÃO: Cuidado na hora de selecionar como módulo <M> as opções do kernel. Nem tudo pode ser marcado como módulo. Algumas opções devem ser (built-in) incorporadas <*> no kernel, como

sistema de arquivos ext3.

8 - Entre com o comando "make" para compilar o kernel

make

9 - Instale os módulos que foram compilados (isso irá criar uma pasta com os módulos compilados em "/lib/modules/2.6.35.4")

make modules_install

10 - Copie a imagem gerada do kernel "vmlinuz-2.6.35.4" e "System.map-2.6.35.4" para "/boot"

make install

11 - Crie o "initrd.img" com os módulos do kernel compactados

mkinitramfs -o /boot/initrd.img-2.6.35.4 /lib/modules/2.6.35.4/

12 – Atualize o arquivo de configuração do GRUB para que ele reconheça o novo kernel

update-grub

13 - Testando: reinicie o máquina e selecione a opção do novo kernel

[Gerenciamento de log]

Log é o termo utilizado para descrever o processo de registro de eventos relevantes num sistema. Esse registro pode ser utilizado para restabelecer o estado original do sistema ou para que o administrador conheça o seu comportamento no passado. Um arquivo de log pode ser utilizado para auditoria e diagnóstico de problemas em sistemas.

Alguns sistemas operacionais disponibilizam um serviço de log chamado Syslog (descrito na RFC 3164), que filtra e registra as mensagens destinada ao log, livrando as aplicações do ônus de manter o seu sistema de log.

O diretório /var/log/

Armazena todos os logs do Linux, isso inclui logs de kernel, serviços e sistemas.

O arquivo /var/log/syslog

Principal arquivo de log em servidores Linux, esse é o arquivo de log central do servidor syslog

O arquivo /var/log/messages

Arquivo de log referênte a eventos do kernel

[CAPÍTULO 4 - REDES]

O arquivo /etc/sysctl.conf

Contém informações sobre configurações de variáveis do sistema.

Principais entradas: net.ipv4.ip_forward

O diretório /proc/sys/net

Neste diretório é possível alterar parâmetros do kernel relacionados a rede

O comando netstat

Lista conexões de rede ativas

Sintaxe: netstat [atributos]

Principais atributos: -n -l

Exemplo: netstat -nl

[Servidor SSH]

Apresentação

SSH (Secure Shell) é um programa de computador e um protocolo de rede que permite a conexão e execução de comandos em um host remoto. Possui as mesmas funcionalidades do TELNET, com a vantagem da conexão entre o cliente e o servidor ser criptografada.

Instalação do servidor SSH

Instale o pacote: openssh-server

```
apt-get install openssh-server
```

Configuração do servidor SSH

Principais entradas do arquivo de configuração /etc/ssh/sshd_config: Port, ListenAddress, PermitRootLogin, AllowUsers

1 – Adicione ou modifique as seguintes entradas no arquivo: `/etc/ssh/sshd_config`

`PermitRootLogin no`

`AllowUsers nickollas joao pedro`

2 – Reinicie o servidor ssh

`/etc/init.d/ssh restart`

Instalação do cliente SSH

Instale o pacote: `openssh-client`

`apt-get install openssh-client`

O comando ssh

Conecta a um servidor ssh

Sintaxe: `ssh [atributos] [usuário] [@] [host]`

Principais atributos: `-p`

Exemplo: `ssh nickollas@192.168.0.254`

[Servidor NIS]

Apresentação

NIS (Network Information Service) é um serviço de gerenciamento de login via rede que facilita a administração do ambiente Linux, pelo fato de manter uma base de dados centralizada. A base de dados NIS é criada a partir de tabelas (plain text database), tal como `/etc/passwd`, `/etc/shadow` e `/etc/group`.

Instalação do servidor NIS

Instale os pacotes: `portmap, nis`

`apt-get install portmap nis`

Configuração do servidor NIS

1 – Defina o domínio do servidor NIS

```
echo empresa > /etc/defaultdomain
```

2 – Altere a entrada do arquivo /etc/default/nis

```
NISSERVER=master
```

3 – Adicione a entrada no arquivo /etc/yp.conf

```
ypserver 127.0.0.1
```

4 – Modifique as entradas da coluna Security, “port” para “none” do arquivo /etc/ypserv.conf

```
*           : *       : shadow.byname   : none
*           : *       : passwd.adjunct.byname : none
```

5 – Crie a base de dados para consultas do servidor

```
cd /var/yp
```

```
make
```

6 – Reinicie os serviços portmap e nis

```
/etc/init.d/portmap restart
```

```
/etc/init.d/nis restart
```

7 – Testando a configuração do servidor NIS

```
yptest
```

Instalação do cliente NIS

Instale os pacotes: nis

```
apt-get install nis
```

Configuração do cliente NIS

1 – Defina o domínio do servidor NIS

```
echo empresa > /etc/defaultdomain
```

2 – Adicione a entrada para o servidor NIS no arquivo /etc/yp.conf

```
ypserver 192.168.0.1
```

3 – Adicione a entrada o arquivo /etc/passwd

```
echo '+:~:~:~:~:~:~:' >> /etc/passwd
```

4 – Adicione a entrada o arquivo /etc/shadow

```
echo '+:~:~:~:~:~:~:' >> /etc/shadow
```

5 – Adicione a entrada o arquivo /etc/group

```
echo '+:~:~:~:' >> /etc/group
```

6 – Reinicie os serviços portmap e nis

```
/etc/init.d/portmap restart
```

```
/etc/init.d/nis restart
```

7 – Testando a comunicação com servidor NIS

```
yptest
```

* Faça login numa console e veja se você cairá no shell

[Servidor NFS]

Apresentação

NFS (Network File System) é um sistema de arquivos distribuídos a fim de compartilhar arquivos e diretórios entre computadores conectados em rede, formando assim um diretório virtual.

Instalação do servidor NFS

Instale os pacotes: nfs-kernel-server, nfs-common, portmap

```
apt-get install nfs-kernel-server nfs-common portmap
```

Configuração do servidor NFS

1 – Adicione a entrada no arquivo de configuração do servidor nfs, /etc/exports

```
/home 192.168.0.0/255.255.255.0(rw,no_root_squash,async,no_subtree_check,insecure)
```

2 – Atualize a tabela de exportações do servidor NFS

```
exportfs -av
```

Instalação do cliente NFS

Instale os pacotes: nfs-common, portmap

```
apt-get install nfs-common portmap
```

Configuração do cliente NFS

1 – Adicione a entrada no arquivo /etc/fstab

```
192.168.0.1:/home /home nfs defaults 0 0
```

2 – Faça a máquina reler o arquivo fstab para montar a partição

```
mount -a
```

3 – Testando se o diretório foi montado

```
ls /home
```

[Servidor DNS]

Apresentação

DNS (Domain Name System) é um sistema de resoluções de nomes distribuído que pode operar em vários servidores.

BIND (Berkeley Internet Name Domain) é o servidor para o protocolo DNS mais utilizado na Internet, especialmente em sistemas do tipo Unix, onde ele pode ser considerado um padrão.

Instalação

Instale os pacotes bind9, dnsutils

```
apt-get install bind9 dnsutils
```

Configuração

1 - Adicione a entrada ao final do principal arquivo de configuração do bind, /etc/bind/named.conf

```
include "/etc/bind/dominios";
```

2 – Adicione a entrada do seu domínio no arquivo /etc/bind/dominios

```
zone "meudominio.com.br" {
    type master;
    file "/etc/bind/db.meudominio.com.br";
};
```

3 - Crie um arquivo de configuração para o domínio utilizando o modelo de /etc/bind/db.local

```
cp /etc/bind/db.local /etc/bind/db.meudominio.com.br
```

4 - Edite o arquivo do domínio conforme abaixo:

```
vi /etc/bind/db.meudominio.com.br
```



```
;
; BIND data file for meudominio.com.br
;
$TTL      604800
@         IN      SOA      ns.meudominio.com.br. root.meudominio.com.br. (
                        2          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
                        IN      NS       ns.meudominio.com.br.
                        IN      MX       0 mail.meudominio.com.br.
meudominio.com.br.    IN      A        192.168.0.1
ns                   IN      A        192.168.0.1
mail                 IN      A        192.168.0.1
www                  IN      CNAME     meudominio.com.br.
```

5 - Reinicie o bind

```
[root]# /etc/init.d/bind9 restart
```

6 - Configure a máquina para resolver nomes usando o IP do loopback

```
vi /etc/resolv.conf
```

```
nameserver 127.0.0.1
```

7 – Testando, fazendo consultas ao servidor de DNS

```
host -t ns meudominio.com.br
```

```
nslookup meudominio.com.br
```

[Servidor Web]

Apresentação

Um servidor web serve conteúdo, como páginas web, utilizando o protocolo HTTP, através da internet.

O **Apache** (ou Servidor HTTP Apache) é o servidor web mais utilizado no mundo.

Instalação

Instale os pacotes: apache2, lynx

```
apt-get install apache2 lynx
```

Configuração de domínios virtuais no apache

1 - Adicione a entrada no arquivo /etc/apache2/httpd.conf

```
ServerName localhost
```

2 – Crie o arquivo /etc/apache2/sites-available/meudominio.com.br com as seguintes entradas:

```
<VirtualHost *:80>
DocumentRoot    /var/www/virtual/meudominio.com.br/htdocs
ServerName      meudominio.com.br
ServerAlias     meudominio.com.br www.meudominio.com.br *.meudominio.com.br
ServerAdmin     suporte@meudominio.com.br
ErrorLog        /var/log/apache2/meudominio.com.br-error.log
TransferLog     /var/log/apache2/meudominio.br-access.log
</VirtualHost>
```

3 – Crie o diretório onde os arquivos do domínio ficarão armazenados

```
mkdir -p /var/www/virtual/meudominio.com.br/htdocs
```

4 – Crie o arquivo index.html com qualquer conteúdo dentro do diretório do domínio htdocs

```
echo "teste" > /var/www/virtual/meudominio.com.br/htdocs/index.html
```

5 – Habilitando o domínio virtual no apache

```
a2ensite meudominio.com.br
```

6 – Faça o apache reler seus arquivos de configuração

```
/etc/init.d/apache2 reload
```

7 – Testando o domínio virtual

```
lynx www.meudominio.com.br
```

Para remover seu domínio virtual use: a2dissite meudominio.com.br

[Suporte a PHP com MySQL no Servidor Web]

Apresentação

PHP é uma linguagem de programação interpretada, livre e muito utilizada para gerar conteúdo dinâmico na web.

MySQL é um bando de dados, que utiliza a linguagem SQL como interface. É atualmente um dos bancos de dados mais populares do mundo. Empresas como Google, NASA e Bradesco usam MySQL.

Instalação

Instale os pacotes: php5 libapache2-mod-php5 php5-mysql

Configuração

1 – Reinicie o servidor apache

```
/etc/init.d/apache2 restart
```

2 – Crie um arquivo de teste para o PHP

```
echo "<? phpinfo() ?>" > /var/www/index.php
```

3 – Testando se o PHP está habilitado no Apache

```
lynx localhost/index.php
```

[Servidor FTP]

Apresentação

Um Servidor **FTP** (File Transfer Protocol) disponibiliza á usuários um disco rígido ou arquivos sendo ou não necessária a autenticação desse usuário.

ProFTPD é o servidor FTP mais papoluar em ambientes Linux. ProFTPD usa apenas um arquivo de configuração, /etc/proftpd.conf. Seu arquivode configuração é muito semelhante ao arquivo de configuração do Apache.

Instalação

Instale o pacote proftpd, ftp

```
apt-get install proftpd ftp
```

Configuração

1 – Descomente ou altere as linhas abaixo do arquivo /etc/proftpd/proftpd.conf

```
ServerType standalone
```

```
DefaultRoot ~
```

```
RequireValidShell off
```

* Caso esteja usado NIS descomente a entrada:

```
PersistentPasswd off
```

2 - Reinicie o proftpd

```
/etc/init.d/proftpd restart
```

3 – Ative o acompanhamento de conexão para conexões FTP. Adicione as duas entradas abaixo no arquivo /etc/rc.local antes da entrada exit 0

```
modprobe ip_nat_ftp
```

```
modprobe ip_conntrack_ftp
```

4 – Execute o script /etc/rc.local

```
sh /etc/rc.local
```

5 - Testando a conexão com o servidor FTP

```
ftp localhost
```

[Servidor Proxy]

Apresentação

Proxy é um software que filtra e armazena conteúdo atendendo a requisições de clientes e a repassando para servidores. Um servidor proxy pode, opcionalmente, alterar a requisição do cliente ou a resposta do servidor e, algumas vezes, pode disponibilizar este recurso sem nem mesmo se conectar ao servidor.

Squid é um servidor proxy que suporta protocolos HTTP, HTTPS, FTP e outros. Ele reduz a latência de conexões e melhora o tempo de resposta fazendo cache de requisições frequentes de páginas web numa rede.

Instalação

Instale os pacotes squid3 lynx

```
apt-get install squid3 lynx
```

Configuração

1 – Adicione a entrada abaixo na primeira linha arquivo /etc/squid3/squid.conf

```
include /etc/squid3/squid-include.conf
```

2 – Altere as entradas (TAGs) do arquivo /etc/squid3/squid.conf conforme abaixo:

```
acl rede src 192.168.0.0/24
acl sites url_regex -i "/etc/squid3/sites.acl"
acl diretoria src "/etc/squid3/diretoria.acl"
acl downloads url_regex -i "/etc/squid3/downloads.acl"

http_access allow diretoria
http_access deny sites
http_access deny downloads
http_access allow rede

http_port 3128 transparent

cache_mem 32 MB

cache_dir ufs /var/spool/squid3 1024 16 256

maximum_object_size 100 MB

error_directory /usr/share/squid3/errors/Portuguese
```

3 – Adicione as palavras ou sites que deseja bloquear no arquivo /etc/squid3/sites.acl

```
echo "playboy" >> /etc/squid3/sites.acl
```

4 – Adicione as extensões dos arquivos que deseja bloquear no arquivo /etc/squid3/download.acl

```
echo ".zip" >> /etc/squid3/download.acl
```

5 – Adicione as máquinas do grupo diretória no arquivo /etc/squid3/diretoria.acl

```
echo "192.168.0.3" >> /etc/squid3/diretoria.acl
```

6 – Configure o servidor para navegar pelo proxy

```
export http_proxy="http://192.168.0.1:3128"
```

7 – Habilite o repasse de pacote no proxy

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

8 – Reinicie o Squid

```
killall -9 squid3 && /etc/init.d/squid3 start
```

9 - Testando o bloqueio do proxy

lynx www.playboy.com.br

lynx www.eicar.org/download/eicar_com.zip

O arquivo /var/log/squid3/access.log

Arquivo de log de acessos do squid

[Auditoria no Servidor Proxy]

Apresentação

Auditoria é um exame cuidadoso e sistemático das atividades desenvolvidas em determinada empresa ou setor, cujo objetivo é averiguar se elas estão de acordo com as disposições planejadas ou estabelecidas.

Sarg (Squid Analysis Report Generator) é uma ferramenta que permite ver "onde" seus usuários estão indo na Internet.

Instalação

Instale o pacote sarg

```
apt-get install sarg
```

Configuração

1 - Altere as entradas do arquivo /etc/squid/sarg.conf

```
language Portuguese
```

```
access_log /var/log/squid3/access.log
```

2 – Execute o sarg para gerar o relatório

```
sarg
```

3 – Visualizando o relatório

```
lynx localhost/squid-reports
```

[Firewall]

Apresentação

Netfilter é um módulo que fornece ao sistema operacional Linux as funções de firewall, NAT e log dos dados que trafegam por rede de computadores.

Iptables é o nome da ferramenta que permite a criação de regras de firewall e NATs. Apesar de, tecnicamente, o iptables ser apenas uma ferramenta que controla o módulo netfilter, o nome "iptables" é frequentemente utilizado como referência ao conjunto completo de funcionalidades do netfilter.

O comando iptables

O iptables é utilizado para manipular pacotes que chegam/saem do firewall.

Sintaxe: `iptables [-t] [tabela] [comando] [cadeia] [!] [parâmetros] [regra] [-j] [alvo]`

Principais tabelas: filter, nat, mangle

Principais comandos: -P -L -A -D -I -F -Z -N -X -n -v

Principais cadeias: INPUT, FORWARD, OUTPUT, PREROUTING, POSTROUTING

Principais parâmetros: -s -d -p -i -o

Principais alvos: ACCEPT, DROP, TOS, SNAT, DNAT, MASQUERADE, REDIRECT, LOG

Módulos

Uma regra do iptables é composta de vários módulos trabalhando em conjunto.

Principais módulos: multiport, recent, limit, icmp, comment, state, connlimit, limit, string, tos

Criando regras de firewall

1 - Setando a política padrão para DROP

```
iptables -P INPUT DROP
```

```
iptables -P FORWARD DROP
```

2 - Liberando acesso para a interface local

```
iptables -A INPUT -i lo -j ACCEPT
```

```
iptables -A OUTPUT -o lo -j ACCEPT
```

3 - Módulo recent, proteção contra "port scanner"

```
iptables -A INPUT -p tcp -m multiport --dports 1,5,10 -m recent --set --name DENY --rsource -j DROP
```

```
iptables -A INPUT -m recent --rcheck --name DENY --rsource -j DROP
```

4 - Liberando ping para máquina de firewall e para a rede

```
iptables -A INPUT -p icmp --icmp-type 0 -m limit --limit 1/sec -j ACCEPT
```

```
iptables -A FORWARD -p icmp --icmp-type 8 -m limit --limit 1/sec -j ACCEPT
```

```
iptables -A FORWARD -p icmp --icmp-type 0 -m limit --limit 1/sec -j ACCEPT
```

```
iptables -A OUTPUT -p icmp --icmp-type 8 -m limit --limit 1/sec -j ACCEPT
```

5 - Permitindo que firewall e rede estabeleçam conexão com outras máquinas

```
iptables -A INPUT -p ! icmp -m state --state ESTABLISHED -j ACCEPT
```

```
iptables -A FORWARD -s 192.168.0.0/24 -p ! icmp -m state --state NEW,ESTABLISHED -j ACCEPT
```

```
iptables -A FORWARD -d 192.168.0.0/24 -p ! icmp -m state --state ESTABLISHED -j ACCEPT
```

```
iptables -A OUTPUT -p ! icmp -m state --state NEW,ESTABLISHED -j ACCEPT
```

6 - Módulo multiport, liberando acesso a serviços específicos da máquina de firewall

```
iptables -A INPUT -p tcp -m multiport --dports 21,25,80,110 -m state --state NEW,ESTABLISHED -j ACCEPT
```

```
iptables -A INPUT -p tcp --dport 20 -m state --state RELATED,ESTABLISHED -j ACCEPT
```

```
iptables -A INPUT -p udp --dport 53 -m state --state NEW,ESTABLISHED -j ACCEPT
```

7 - Liberando acesso remoto por ssh com módulo recent

```
iptables -A INPUT -p tcp --dport 2221 -m recent --set --name SSH --rsource -j DROP
```

```
iptables -A INPUT -p tcp --dport 22 -m recent --rcheck --name SSH --rsource -j ACCEPT
```

8 - Trabalhando com NAT

```
iptables -t nat -A PREROUTING -d 200.157.231.195 -p tcp --dport 5900 -j DNAT --to-destination 192.168.0.253
```

```
iptables -t nat -A POSTROUTING -s 192.168.0.253 -p tcp --dport 5900 -j SNAT --to-source 200.157.231.195
```

```
iptables -A FORWARD -d 192.168.0.253 -p tcp --dport 5900 -m state --state NEW,ESTABLISHED -j ACCEPT
```

9 - Módulo comment, comentando uma regra

```
iptables -A INPUT -m comment --comment "Fazendo teste" -j ACCEPT
```

10 - Módulo connlimit, limitando número de conexões a um serviço por host

```
iptables -A INPUT -p tcp --dport 21 -m connlimit --connlimit-above 1 -j DROP
```

11 - Módulo iprange, liberando acesso a porta 80 para as máquinas 192.168.0.131-192.168.0.140

```
iptables -A FORWARD -p tcp --dport 80 -m iprange --src-range 192.168.0.131-192.168.0.140 -j ACCEPT
```

12 – Setando regra para proxy transparente

```
iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 3128
```

O comando iptables-save

Salva as regras de iptables carregadas no kernel para um arquivo

Sintaxe: iptables-save [>] [arquivo]

Exemplo: iptables-save > /etc/iptables.regras

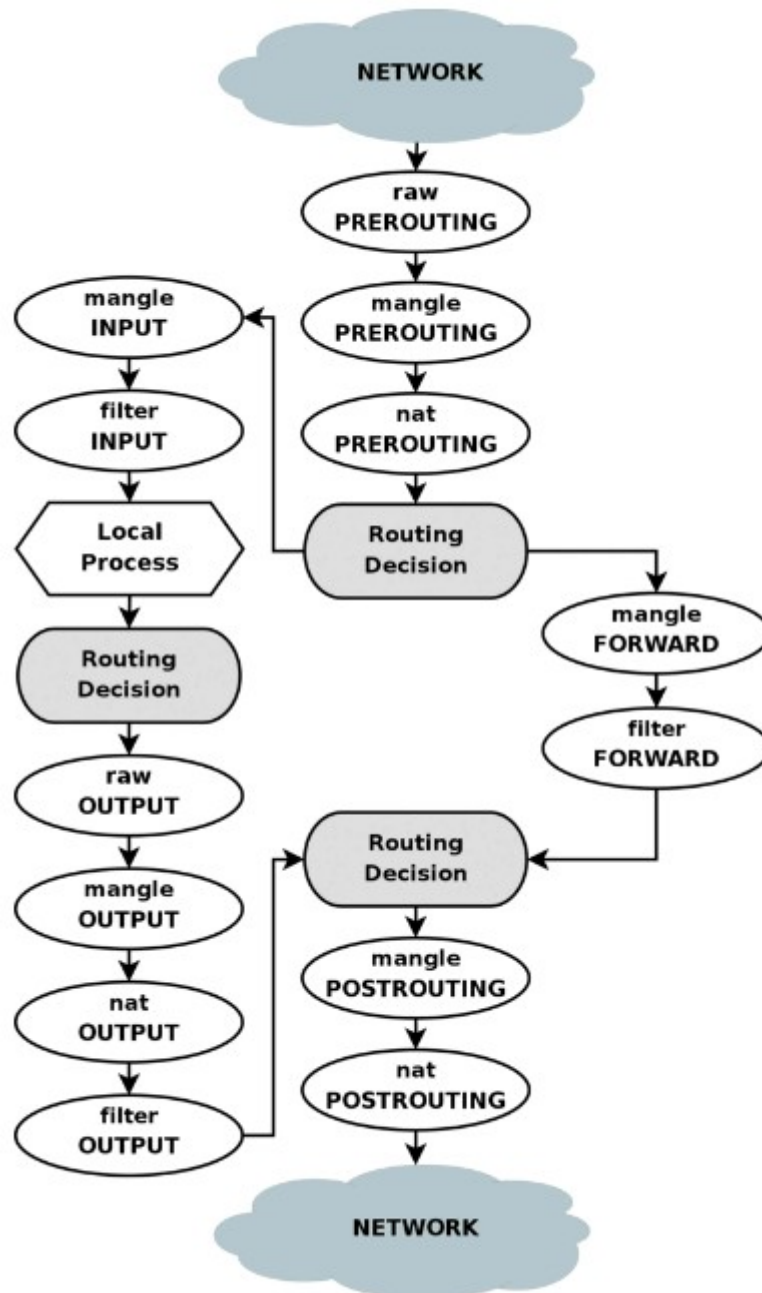
O comando iptables-restore

Carrega as regras de iptables num formato iptables-save para o kernel

Sintaxe: iptables-restore [<] [arquivo]

Exemplo: iptables-restore < /etc/iptables.regras

Fluxo de dados no iptables



[Utilitários de Rede]

O comando mtr

Ferramenta de diagnóstico de rede capaz de dizer em quais gateways há perdas de pacotes

Sintaxe: `mtr [host]`

Exemplo: `mtr www.uol.com.br`

O comando nmap

Scanner de portas para hosts remotos

Sintaxe: nmap [atributos] [host]

Principais atributos: -sS -O -p

Exemplo: nmap -sS -p 80-81 -A www.uol.com.br

[CAPÍTULO 5 - PROGRAMAÇÃO SHELL SCRIPT]

[O que é uma Shell?]

Shell é onde os comandos são interpretados ou executados. Na linha de comandos de um shell, podemos utilizar diversos comandos um após o outro, ou mesmo combiná-los numa mesma linha. Se colocarmos diversas linhas de comandos em um arquivo texto simples, teremos em mãos um Shell Script..

[O que é um Script?]

Script é uma descrição geral para qualquer programa de computador escrito em linguagem interpretada, não compilada. Outros exemplos de linguagens para scripts são o php, perl, python, javascript e muitos outros. Podemos então ter um script em perl, php, shell e assim em diante.

[O que é Shell Script?]

Shell script é uma Linguagem de programação interpretada usada em vários sistemas operacionais, com diferentes dialetos dependendo do interpretador de comandos utilizado. Um exemplo de interpretador de comandos é o bash, usado na grande maioria das distribuições Linux.

[Shell Script]

Definindo o interpretador: #!

O sinal #! na primeira linha do arquivo indica qual o interpretador de comandos será utilizado.

Exemplo:

```
#!/bin/bash
echo -e 'Olá Mundo!'
```

Imprime:

Olá Mundo!

Variáveis

Variáveis são lugares onde dados podem ser armazenados para serem utilizados futuramente num programa. Existem variáveis fornecidas pela linguagem e variáveis definidas pelo programador.

Tabela de variáveis fornecidas pela linguagem:

\$0	Nome do script que está sendo executado
\$1-\$9	Parâmetros passados à linha de comando
\$#	Número de parâmetros passados
\$?	Valor de retorno do último comando ou de todo o shell script. (o comando "exit 1" retorna o valor 1)
\$\$	Número do PID (Process ID)

O operador = (igual)

O operador = define o valor de uma variável trabalhando com dois operandos onde o valor do operando da direita é copiado para o operando da esquerda.

Exemplo:

```
#!/bin/bash
VAR='Olá Mundo!'
echo -e "$VAR"
```

Imprime:

Olá Mundo!

Diferença entre ' (apóstrofo) e “ (aspas)

Apóstrofos suprimem a interpolação de variáveis entre elas. Já as aspas fazem o contrário, interpolam qualquer variável entre elas.

Exemplo:

```
#!/bin/bash
VAR='Olá Mundo!'
echo -e '$VAR'
echo -e "$VAR"
```

Imprime:

\$VAR
Olá Mundo!

O comando: \n

O comando \n indica uma nova linha, é como se a tecla <Enter> tivesse sido pressionada.

```
#!/bin/bash
echo -e "Olá Mundo!\n"
```

Imprime:

Olá Mundo!

O comando if

O comando if é utilizado para fazer testes de condição. Os comandos do if são processados se a condição testada entre colchetes [] for positiva. Caso o teste falhe os comandos não serão processados.

Sitaxe do comando if:

```
if [ condição ] ; then
    [ comandos ]
fi
```

Exemplo:

```
#!/bin/bash
NOME='aluno'
if [ $NOME == aluno ]; then
    echo -e "Entrou no primeiro if..."
fi
if [ $NOME == xyz ]; then
    echo -e "Entrou no segundo if..."
fi
```

Imprime:

Entrou no primeiro if...

Tabela de operadores do comando if

==	Igual
!=	Diferente
-gt	Maior
-lt	Menor
-o	Ou
-d	Se for um diretório
-e	Se existir
-z	Se estiver vazio
-f	Se conter texto
-o	Se o usuário for o dono
-r	Se o arquivo pode ser lido
-w	Se o arquivo pode ser alterado
-x	Se o arquivo pode ser executado

O comando else

O comando else é utilizado em conjunto com o comando if. O comando else só é processado se o teste da condição if falhar.

Sitaxe do comando else:

```
if [ condição ] ; then
  [ comandos ]
else
  [ comandos ]
fi
```

Exemplo:

```
#!/bin/bash
NOME='aluno'
echo -e "\nPrimeiro teste:"
if [ $NOME == aluno ]; then
    echo -e "Entrou no if...\n"
else
    echo -e "Entrou no else..\n"
fi
echo -e "Segundo teste:"
if [ $NOME == xyz ]; then
    echo -e "Entrou no if...\n"
else
    echo -e "Entrou no else..\n"
fi
```

Imprime:

Primeiro teste:
Entrou no if...

Segundo teste:
Entrou no else..

O loop for

O loop for faz iterações numa lista de arquivos / diretórios

Sintaxe do loop for:

```
for variavel_de_iteração in lista
do
```

```
    comandos
```

```
done
```

Exemplo:

```
#!/bin/bash

for i in *
do

    echo $i
done
```

Imprime:

Todos os arquivos e diretórios presentes no diretório atual

O comando sleep

Atrasa a execução de um programa por alguns segundos

Sintaxe: sleep [segundos]

Exemplo:

```
#!/bin/bash

for i in *
do

    echo $i
    sleep 3
done
```

O loop while

O loop while é executado enquanto seu teste de condição for verdadeiro.

Sintaxe do loop while:

```
while [ condição ]
do
```

```
    comandos
```

```
done
```

Exemplo:

```
#!/bin/bash
i=0;
while [ $i -le 3 ]
do
    echo "$i"
    (( i++ ))
done
```

Imprime:

0
1
2
3

Tabela de operadores do loop while

-ge	Maior ou igual
-le	Menor ou igual

[Script de Backup]

Como automatizar o envio de backup para uma conta no gmail

1 - Instale o ssmtp e o mutt

```
[root]# apt-get install ssmtp mutt
```

2 - Configure o ssmtp

```
vi /etc/ssmtp/ssmtp.conf
```

```
hostname=localhost
rewriteDomain=gmail.com
AuthUser=nickollas@gmail.com
AuthPass=123456
AuthMethod=plain
FromLineOverride=NO
Mailhub=smtp.gmail.com:465
UseTLS=YES
```

3 - Crie o arquivo que conterá o corpo de email:

```
echo "a b c" > corpo_email
```

4 - Enviando email com anexo utilizando o mutt

```
[root]# mutt -s "Assunto" -i corpo_email -a /etc/iptables.regras -c
ajudantedepapainoel@gmail.com < /dev/null
```

Exemplo de Script

```
#!/bin/bash
#
# Script que envia backup para o gmail
#
EMAIL_TO=nickollas@gmail.com
SUBJECT='[ backup servidor ]'
BODY=corpo_email
ATTACH=/etc/iptables.regras
ATTACH2=/etc/init.d/rc.local

IP=`ifconfig ppp0 | grep inet`

# CRIANDO ARQUIVO DE CORPO DO EMAIL
echo > $BODY

# IP DO SERVIDOR
echo -e "Endereço IP do Servidor:" >> $BODY
echo -e $IP >> $BODY

# ENVIA E-MAIL
mutt -s "$SUBJECT" -i $BODY -a $ATTACH -a $ATTACH2 -a $ATTACH3 -c
$EMAIL_TO < /dev/null
```