

Administração de Sistemas

Orlando Sousa

Aula 8

Tarefas e detalhes da Pós-instalação do Linux

Resolução de problemas da instalação

- Ficheiros que permitem verificar a instalação e outros problemas:
 - **/root/install.log** – lista os pacotes instalados (pelo Anaconda - programa utilizado pela Red Hat para instalar o Linux)
 - **/root/anaconda-ks.cfg** – contém os comandos utilizados pelo Anaconda para instalar o Linux (pode servir como *template* para efectuar a **automatização** da instalação através do *Kickstart*)
 - **/var/log/messages** – ficheiro que contém a grande maioria dos “acontecimentos” no linux
 - **/var/log/dmesg** – contém a informação de *boot* (ex: informação de hardware, inicialização de processos, etc).
- Se o Servidor X não foi configurado correctamente, fazer CTRL-ALT-F2 para “voltar” ao modo texto
- Login gráfico e texto – geralmente utiliza o *runlevel* 5 para funcionar em modo gráfico e o *runlevel* 3 em modo texto. Para alterar o *runlevel*, usar o comando:
 - `init x` (onde x é o *runlevel*)

- **/bin** – contém executáveis. Os comandos apenas executados por alguns utilizadores devem ser colocados em `/usr/bin`
- **/boot** – contém todos os ficheiros necessários para efectuar o arranque (excepto os ficheiros de configuração).
Contém dados que são utilizados antes do kernel efectuar a inicialização do sistema através do que está definido em `/etc/inittab`
- **/dev** – contém todos os dispositivos (de blocos e de caracteres). Também contém os sockets e os pipes com nome. Em Linux todos os dispositivos são “vistos” como ficheiros.
- **/etc** – contém os ficheiros de configuração da máquina.
- **/home** – contém os directórios dos utilizadores
- **/lib** – contém bibliotecas (partilhadas) que são necessárias para o arranque do sistema ou para a utilização de certos comandos (localizados em `/bin`). As bibliotecas de suporte a `/usr` são instaladas em `/usr/bin`
- **/mnt** – contém *mounts* temporários para dispositivos de dados (ex: discos, drives de disquetes, CD-ROM, etc).
- **/opt** – contém todos os dados necessários para o suporte ao software adicionado no sistema original.
- **/proc** – contém um sistema de ficheiros virtual.
- **/root** – contém ficheiros de configuração para a conta “root” (administrador do sistema).
- **/sbin** – Contém executáveis que são apenas utilizados pelo superutilizador (o administrador do sistema)
- **/tmp** – é usado sempre que um programa necessita de escrever num ficheiro que será removido quando o programa terminar
- **/usr** – armazena dados partilháveis no modo “read only”
- **/var** – contém dados como logs, ficheiros de lock e dados específicos dos processos

Boot Loaders

- Software localizado no MBR (Master Boot Record) de um disco que permite “carregar” um ou mais sistemas operativos
- Depois de ligar o computador, a BIOS detecta todo o hardware básico, incluindo os discos. Posteriormente procura o *boot loader* no MBR do 1º disco. Nota: Nos pcs antigos, a BIOS apenas irá ler se este estiver nos primeiros 1024 cilindros do disco
- Se todos os discos forem IDE então o directório /boot deve estar num disco associado ao **primeiro** controlador IDE
- Se todos os discos forem SCSI, o directório /boot deve estar associado a disco com identidade SCSI 0 ou 1
- Se existirem discos IDE e SCSI então o directório /boot deve estar na primeira drive IDE ou na SCSI 0
- Nota: Quando o MBR já estiver “ocupado” por *outro boot loader*, então o 1º boot loader encontra o *boot loader* no primeiro sector da partição Linux

GRUB(GRand Unified Bootloader)

- Apresenta um menu para escolher o sistema operativo (no caso de existir mais do que um instalado)
 - É mais flexível que o LILO
 - Pode utilizar comandos
 - /etc/grub.conf – ficheiro de configuração
 - Se já instalou o LILO e quer mudar para GRUB, então deve editar o ficheiro /etc/grub.conf e depois executar “grub-install /dev/XYZ”, onde XYZ representa o disco com o directório /boot
- Nota: o ficheiro /etc/grub.conf está “ligado” ao ficheiro /boot/grub/grub.conf
- Comandos de edição do GRUB:
 - b – arranca o sistema operativo actual (o que está “escolhido”)
 - d – remove a linha actual
 - e – permite editar a linha actual
 - o – cria uma linha vazia
 - O – cria uma linha vazia acima da linha actual

Exemplo: grub.conf

```
# grub.conf generated by anaconda
# Note that you do not have to rerun grub after making changes to this file
# NOTICE: You do not have a /boot partition. This means that
# all kernel and initrd paths are relative to /, eg.
# root (hd0,0)
# kernel /boot/vmlinuz-version ro root=/dev/hda1
# initrd /boot/initrd-version.img
#boot=/dev/hda
default=0 A imagem por defeito será a primeira na lista (neste caso é “Red Hat Linux (2.4.7-10enterprise”)
timeout=10 Espera 10 segundos até “avançar”
splashimage=(hd0,0)/boot/grub/splash.xpm.gz É a imagem apresentada no menu de arranque
password --md5 $1$AÀnFvÁ6Q$6T7hhyN2k74Fizf29eOH70 Password utilizada para usar o GRUB. Tem de se
usar a opção -p. Sem a opção -md5, a password apareceria em texto claro!
title Red Hat Linux (2.4.7-10enterprise) Titulo que irá aparecer para escolher este sistema
root (hd0,0) Usa a primeira partição no primeiro disco como partição root
kernel /boot/vmlinuz-2.4.7-10enterprise ro root=/dev/hda1 Parte compactada do kernel.
initrd /boot/initrd-2.4.7-10enterprise.img Imagem utilizada (conjunto de ficheiros necessários durante o
processo de arranque do sistema)
title Red Hat Linux-up (2.4.7-10)
root (hd0,0)
kernel /boot/vmlinuz-2.4.7-10 ro root=/dev/hda1
initrd /boot/initrd-2.4.7-10.img
title Red Hat Linux-up (2.4.7-10)
root (hd1,0)
kernel /boot/vmlinuz-2.4.7-10 ro root=/dev/hda1
initrd /boot/initrd-2.4.7-10.img
```

A contagem dos discos inicia em 0, sendo o primeiro disco hd0

LILO (Linux LOader)

- É o *boot loader* tradicional. Cada vez é menos utilizado
- `/etc/lilo.conf` – ficheiro de configuração
- Se o ficheiro `/etc/lilo.conf` for alterado então as modificações têm que ser efectuadas através do comando “lilo”
- Exemplo: `lilo.conf`

prompt **Força o lilo boot a aparecer na consola**

`timeout=50` **Força o LILO a espera 5 segundos (50 décimas de segundo)**

`default=linux` **antes de passar para o sistema operativo por defeito, que é o linux**

`boot=/dev/hda` **Procura informação de boot no primeiro disco IDE**

`map=/boot/map` **Procura informação de mapeamento no primeiro disco IDE**

`install=/boot/boot.` **É o boot loader secundário**

`message=/boot/message` **Abre o ficheiro que é visto durante o processo de arranque**

`linear` **Usa endereços lineares para encontrar o directório /boot**

`image=/boot/vmlinuz-2.4.21-3` **É o kernel escolhido (compactado)**

`label=linux` **É o texto que aparece na escolha do sistema**

`initrd=/boot/initrd-2.4.21-3.img` **Local onde se encontra loader “secundário”**

`read-only` **Durante a instalação a RAM disk é iniciada no modo read-only. Depois do loader “secundário” ter terminado, a RAM disk é removida da memória, e a partição *root* real é então “montada” no modo read/write**

`append="hdc=ide-scsi root=LABEL=/"` **Adiciona um comando ao *kernel*. Este comando aponta para a etiqueta associada com o directório “raiz”**

`other=/dev/hda1` **É outra opção de boot, neste caso faz o arranque do Windows XP na primeira partição**

`label=WinXP`

`table=/dev/hda` **É o local da tabela de partições a ser usada, neste caso é a tabela do primeiro disco IDE**

- **Para criar uma disquete de arranque independente do *boot loader*, fazer:**
 - 1 # dd if=/boot/vmlinuz-2.4.7-10 of=/dev/fd0** (copia o kernel – neste caso `/boot/vmlinuz-2.4.7-10`, para a disquete - `/dev/fd0`)
 - 2 # rdev /dev/fd0 /dev/hda1** (informa o kernel da disquete qual é a partição *root*, através da utilização do comando *rdev*)

Init - O processo de arranque do Linux

- Depois do kernel ter sido lido, este inicia um programa (init), que inicia tudo o resto
- O comando **chkconfig** permite gerir serviços em vários *runlevels* (**redhat-config-services** é a versão gráfica)
- Existem seis *runlevels* (definidos em /etc/inittab)
- Cada *runlevel* está associado a um nível de funcionalidade
- Runlevels:
 - 0 – Halt
 - 1 – Single User mode
 - 2 – Multi-user, sem rede
 - 3 – Multi-user, com rede
 - 4 – não é utilizado
 - 5 – X11, o normal para fazer o login gráfico
 - 6 – Reboot (nunca coloque o initdefault no /etc/inittab para este valor!)

- O processo *init* executa */etc/rc.d/rc.sysinit* (este ficheiro efectua várias tarefas – ex: configuração da rede, mapeamento do teclado, *mount* de partições e hostnames, etc.)
- Depois o processo *init* determina o *runlevel* através da entrada *initdefault* no ficheiro ***/etc/inittab***
- Exemplo :
id:5:initdefault: **inicia no *runlevel* 5**
- Posteriormente, o processo *init*, inicia as *scripts* associadas ao *runlevel* escolhido (do directório ***/etc/rc.d***). Este directório contém os seguintes subdirectórios:
init.d , *rc0.d* , *rc1.d* , *rc2.d*, *rc3.d*, *rc4.d*, *rc5.d*, *rc6.d*

- Se o *runlevel* escolhido é o 5, então o processo *init* irá executar todas as *scripts* “kill” (ficheiros que começam por “K”) e “start” (ficheiros que começam por “S”) que existem em */etc/rc.d/rc5.d*. As *scripts* neste directório são uma ligação simbólica para *scripts* em */etc/rc.d/init.d*.
- O processo *init* executa todas as *scripts* K (que matam os processos não suportados no *runlevel* escolhido) e depois todas as *scripts* S (inicia processos associados ao *runlevel* escolhido)
- Também é possível executar essas *scripts* manualmente

Terminais virtuais

- O mesmo utilizador pode fazer o login várias vezes, permitindo deste modo que um (ou mais) utilizador(es) esteja em simultâneo a ver um ficheiro de ajuda numa consola, a compilar um ficheiro noutra, etc.
- Por defeito estão configurados 6 (pode adicionar mais 6)
- Pode “saltar” entre terminais virtuais adjacentes através das teclas ALT-seta direita e ALT-seta esquerda
- São configurados no ficheiro `/etc/inittab`

Exemplo: Entradas `/etc/inittab` para as primeiras 6 consolas

```
1:2345:respawn:/sbin/mingetty tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6
```

- Por defeito, a consola gráfica é a 7^a. Para “saltar” entre consolas, fazer CTRL-ALT-Fn (onde n é o número do terminal virtual!)

Partições

- A melhor forma para criar partições é a utilização do Disk Druid durante o processo de instalação. No entanto, como podem ser cometidos erros, como por ex. o esquecimento da criação de uma partição ou criar uma partição /home maior, é necessário conhecer os comandos que permitem manusear as partições
- O comando “**df**” apresenta o espaço livre e ocupado de todas as partições “montadas”

```
[root@Enterprise root]# df
Filesystem      1k-blocks    Used    Available    Use%    Mounted on
/dev/hda8        932833      502478    382162      57%     /
/dev/hda7        23300       2588     19509       12%     /boot
/dev/hda1       1052064     914784    137280      87%     /dosC
/dev/hda6       1052064     111648    940416      11%     /home
none            62828       0         62828      100%    /dev/shm
/dev/hdc        556054     556054     0          100%    /mnt/cdrom
[root@Enterprise root]#
```

- O comando “**mount**” (sem opções) permite ver o tipo de sistema de ficheiros de cada partição
- O comando “**fdisk**” permite criar, eliminar e alterar partições. Ver: **man fdisk**

Sistemas de ficheiros

- Tipos de sistemas de ficheiros
 - ext – o primeiro sistema de ficheiros do Linux
 - ext2(Second extended) – a base para o ext3 (que é o sistema de ficheiros usado pelo Red Hat)
 - swap – utilizado para partições de *swap*.
 - MS-DOS e VFAT – permitem aceder a sistemas de ficheiros formatados com o MS-DOS (todos antes do Windows 95). VFAT permite aceder a partições com Windows 9x/ NT/2000/XP/2003 formatadas com FAT16 ou FAT32
 - ISO 9660 – o sistema de ficheiros standard para os CD-ROMs
 - NTFS – sistema de ficheiros usado pelo Microsoft Windows NT/2000/XP/2003. Apenas funciona no modo de leitura
 - /proc – é o sistema de ficheiros virtual do Linux. É utilizado para disponibilizar informação ao kernel e o estado de dispositivos
 - /dev/pts – é a implementação em Linux do “Open Group’s Unix 98 PTY”
 - NFS – é o sistema de ficheiros de rede. É utilizado para partilhar ficheiros e impressoras entre computadores Linux e Unix
 - SMB – Server Message Block – é baseado nos protocolos da Microsoft e da IBM. Permite partilhar ficheiros e impressoras com os sistemas Microsoft Windows
 - ext3 – é o sistema de ficheiros utilizado pela Red Hat (é um ext2 com *journaling* – é mais eficiente durante o processo de verificação no arranque e mantém um *log* que pode ser utilizado para restaurar o sistema no caso de um *crash*)

/etc/fstab

- Ficheiro que contém informação relacionada com o *mount* de sistemas de ficheiros locais e remotos

LABEL	Ponto de mount	Sistema ficheiros	Opções mount	Valor dump (0 ou 1) 1- os dados são gravados no disco com o "dump"	Ordem pelo qual os sistemas de ficheiros são verificados no arranque
LABEL=/	/	ext3	defaults	1	1
LABEL=/boot	/boot	ext3	defaults	1	2
none /dev/pts	devpts	gid=5,mode=620		0	0
none /proc	proc	defaults		0	0
none /dev/shm	proc	tmpfs		0	0
/dev/hda3	swap	swap	defaults	0	0
LABEL=/usr	/usr	ext3	defaults	1	2
LABEL=/tmp	/tmp	ext3	defaults	1	2
LABEL=/var	/var	ext3	defaults	1	2
LABEL=/home	/home	ext3	defaults	1	2
/dev/cdrom	/mnt/cdrom	udf,iso9660	noauto,owner,kudzu,ro	0	0
/dev/floppy	/mnt/floppy	auto	noauto,owner,kudzu	0	0

- Consultar: **man mount**
- Efectuar o mount de drives de disquetes:
mount -t vfat /dev/fd0 /mnt/floppy (efectua o mount no directório /mnt/floppy usando o sistema de ficheiros VFAT)

Nota: para efectuar o *mount* até bastava: **mount /dev/fd0 /mnt /floppy**, já que o comando *mount* verifica automaticamente o sistema de ficheiros da disquete

Olhando novamente para o ficheiro */etc/fstab*, bastaria fazer um dos seguintes comandos:

mount /dev/fd0

mount /mnt/floppy

- Para efectuar o *mount* de um CD-ROM ISO 9660, fazer:

```
# mount -rt iso9660 /dev/cdrom /mnt/cdrom
```

O mesmo se aplica para esta situação. Basta executar um dos seguintes comandos:

```
# mount /dev/cdrom
```

```
# mount /mnt/cdrom
```

- O comando **umount** faz o *unmount*

```
# umount /mnt/floppy
```

```
# umount /mnt/cdrom
```

- Para utilizar a imagem de um cd sem ter de utilizar um CD-ROM, fazer:

```
# mkdir /mnt/imagem-iso
```

```
# mount -o loop caminho-do-ficheiro.iso /mnt/imagem-iso
```

Automounter

- O *daemon* **autofs** configura os directórios de *mount* à medida que são necessários (de uma forma temporária)
- O Red Hat configura os *automounts* no directório **/misc** (não funciona se existirem outros ficheiros nesse directório!)
- **/etc/auto.master** – contém comentários (que podem ser usados quando se tira o “#”) para o funcionamento do *automounter*

Exemplo: # /misc /etc/auto.misc --timeout=60 (se retirar o “#”, irá configurar o automount no directório /misc, apontando para /etc/auto.misc para os detalhes da configuração. Se o directório não for usado durante 60 segundos então é efectuado o unmount do directório)

- Pode configurar o *automount* para outros directórios. Um exemplo útil é configurar para o directório /home (/home /etc/auto.home --timeout=60).

- **/etc/auto.misc** – contém os comandos para efectuar o *automount*

cd -fstype=iso9660,ro,nosuid,nodev :/dev/cdrom

Esta linha está activa por defeito. Se tiver um cd no /dev/cdrom então pode aceder via /misc/cd. O *automounter* acede através do sistema de ficheiros iso9660. O mount está efectuado para ser “read only” (ro). Não é permitido configurar as permissões dos utilizadores (no suid). Não são usados dispositivos neste sistema de ficheiros (nodev)

#linux -ro,soft,intr ftp.example.org:/pub/linux

Esta linha (tem de tirar o comentário!) efectua um ponto de *mount* em /misc/linux para um directório partilhado /pub/linux (NTFS) no computador ftp.example.org

#boot -fstype=ext2 :/dev/hda1

Assume que o directório /boot está instalado na partição /dev/hda1. Assim não necessita efectuar o *mount* quando arranca o Linux. Em vez disso, este comando permite o *automount* com o comando **mount /misc/boot**

#floppy -fstype=auto :/dev/fd0

#floppy -fstype=ext2 :/dev/fd0

#e2floppy -fstype=ext2 :/dev/fd0

A primeira linha procura em /etc/filesystems o sistema de ficheiros que se enquadre com a disquete inserida. As duas últimas linhas assumem que a disquete está formatada com o sistema de ficheiros ext2

#jaz -fstype=ext2 :/dev/sdc1

Esta linha aponta para a primeira partição do terceiros disco SCSI. O comando “jaz” que será usado para a drive “Jazz” da lomega

#removable -fstype=ext2 :/dev/hdd

Esta linha assume que se quer aplicar o *automounter* ao disco IDE ligado como escravo no segundo controlador IDE. “removable” sugere que será utilizado para discos amovíveis.

- Para **activar** o *Automounter*, fazer (na linha de comandos!):
service autofs restart

Máquinas Virtuais

- VMWare Server – será utilizado nas aulas práticas
- No DEI, o VMWare server está em:
 - virtual.dei.isep.ipp.pt
 - Os postos acedem ao VMWare server através da “VMWare Server Console” que está em <\\virtual\clients>
 - **Ler o ficheiro \\virtual\clients\notas.txt**
 - Também estarão disponíveis *imagens* do Linux em <\\virtual\CDs>
 - Para obter informação detalhada do vmware, consultar www.vmware.com

Bibliografia

- The Linux System Administrator's Guide V0.9, L. Wirzenius, J. Oja, S. Stafford, A. Weeks, LDP, 2004
- Linux Systems Administration, Second Edition, Vicki Stanfield, Sybex, 2002
- Red Hat Linux Networking and System Administration, 3rd Ed., Collings and Wall, Wiley, 2005