

# Administração de Sistemas

Orlando Sousa

## **Aula 10**

Gestão de quotas  
Automatização de tarefas  
Backups

# Gestão de quotas

- São utilizadas para limitar o espaço em disco ocupado por utilizadores e/ou grupos
- Evita que um pequeno número de utilizadores ocupe grande parte do espaço em disco
- Existem duas formas de configurar quotas para utilizadores:
  - Através do nº de *inodes*
  - Através do nº de blocos (ex: blocks de 1KB)  
Como cada ficheiro necessita de um *inode*, pode limitar a utilização do espaço em disco através do número de ficheiros ou através do espaço absoluto.
- Pode definir diferentes configurações de quotas para diferentes sistemas de ficheiros (Ex: pode definir diferentes quotas para o directório /home e para o /tmp desde que estejam *montadas* em partições diferentes).
- Para verificar se a configuração de quotas está activa, deve consultar a configuração do kernel (a configuração básica está em /boot (ex: grep CONFIG\_QUOTA /boot/config-versão-kernel ). O valor deve ser **“CONFIG\_QUOTA=y”**
- Para verificar a instalação do “pacote” quota, fazer: **rpm -qi quota** e **rpm -ql quota**
- A *script* /etc/rc.sysinit inclui comandos para iniciar o serviço de quotas. Executa o comando **quotacheck** e o **quotaon** (para activar gestão de quotas em todos os sistemas de ficheiros indicados em /etc/fstab)

# Activação de quotas

- Gestão de quotas de utilizadores e grupos no ficheiro `/etc/fstab` (para o fazer é necessário acrescentar a *keyword* **usrquota** e/ou **grpquota** no campo “Options”)

Exemplo de activação da gestão de quotas para o directório `/home`:

Device	Mount point	Filesys	Options	dump	Fsck
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
<b>/dev/hdd1</b>	<b>/home</b>	<b>ext3</b>	<b>exec,dev,suid,rw,usrquota,grpquota</b>	<b>1</b>	<b>2</b>
/dev/cdrom	/mnt/cdrom	udf,iso9660	noauto,owner,kudzu,ro	0	0
/dev/floppy	/mnt/floppy	auto	noauto,owner,kudzu	0	0

- Para a gestão de quotas são necessários os ficheiros:
  - **aquota.user** – para a gestão de quotas de utilizadores
  - **aquota.group** – para a gestão de quotas de grupos

Para activar gestão de quotas no directório `/home`, fazer:

```
# mount -o remount /home
```

```
# quotacheck -avgm (nota: man quotacheck)
```

Nota: Pode verificar a criação dos ficheiros através do comando: `ls -l /home/aquota.*`

# edquota

- **edquota** – comando que permite configurar as quotas dos utilizadores (edita o ficheiro `aquota.user` e `aquota.group` usando o `vi`)

**Exemplo:** alterar a quota para o utilizador `asist06`

```
# edquota -u asist06
```

Disk quotas for user asist06 (uid 510):

Filesystem	Blocks	soft	hard	inodes	soft	hard
<b>/dev/hdd1</b>	<b>52</b>	<b>0</b>	<b>0</b>	<b>13</b>	<b>0</b>	<b>0</b>

Grace period before enforcing soft limits for users:

Time units may be: days, hours, minutes, or seconds

Filesystem	Block Grace period	Inode Grace period
<b>/dev/hdd1</b>	<b>7 days</b>	<b>7 days</b>

Estas linhas indicam o seguinte: O sistema de ficheiros com gestão de quotas está *montado* na partição `hdd1`, o utilizador está a ocupar 52 blocos (e actualmente o *limite soft* e *hard* estão a **zero**, ou seja, **não tem limite**), Está a ocupar 13 *inodes*, O período de tolerância (grace period) é de 7 dias para o número de blocos e de inodes

# Gestão do espaço em disco

- **Limite Soft**- é o espaço máximo que um utilizador pode ter numa partição. Se tiver um período de tolerância (nº de dias que o utilizador pode ultrapassar o limite) então funcionará como um aviso, sendo notificado de violação do espaço permitido.
- **Limite Hard** – no caso de ter um período de tolerância é necessário configurar este valor. **É o espaço máximo que o utilizador pode usar**. Se não existir período de tolerância então o espaço máximo será definido no “Limite Soft”
- Para definir o período de tolerância para todos os utilizadores fazer: **edquota -t**
- Pode usar os dados de configuração de um utilizador já configurado na gestão de quotas, na aplicação da gestão de quotas a novos utilizadores (-p utilizador\_configurado argumentos):  
**# edquota -up assist06 aluno1 aluno2 prof1**
- Também pode configurar a gestão de quotas por grupos (existentes em /etc/group):  
**# edquota -g alunos**
- **Automatização das definições:**
  - Tem de utilizar o comando quotacheck de uma forma regular. Pode utilizar a automatização de tarefas. O exemplo seguinte executa o comando **quotacheck** às 4 horas todas as sextas-feiras:  
**0 4 \* \* 6 /sbin/quotacheck -avug**
  - Também pode utilizar o comando edquota para aplicar quotas a todos os utilizadores do sistema. O exemplo seguinte aplica as definições das quotas configuradas no utilizador *asit06* a todos os utilizadores:  
**edquota -p assist06 `awk -F: '\$3 > 499 {print \$1}' /etc/passwd`**
- **Relatórios de gestão de quotas**
  - Útil para o administrador verificar quem está a usar mais espaço em disco.
  - Pode gerar relatórios por utilizador, grupo, ou todos utilizadores numa dada partição
  - Para ver informação da gestão de quotas para todos, usar: **repquota -a**
  - Para ver informação apenas de um utilizador, fazer: **quota -uv utilizador**

# Automatização de tarefas - cron

- **cron** – permite executar tarefas a um determinado instante
  - Utiliza o *daemon* `crond`. Este serviço verifica o directório `/var/spool/cron` para tarefas (por utilizador). Também verifica as tarefas escalonadas no computador em `/etc/crontab` e no directório `/etc/cron.d`
  - Cada linha do ficheiro **crontab** pode ter uma linha em branco, comentários (`#`), variáveis ou um comando
  - quando se executa um comando, as acções da *shell* são baseadas nas variáveis ambiente.
  - Estrutura de uma linha no `crontab`:  
#minute, hour, day of month (1-31), month, day of week(0-7), commando

Nota: 0-7 são ambos domingo

Exemplo:

**1 5 3 4 \* ls** (nota: um '\*' significa que executa em todas unidades desse campo – ex: um \* nos minutos significa que o comando será executado todos os minutos). Este exemplo executa o comando 'ls' todos os dias 3 de Abril às 05h:01m

Notas:

- Também pode utilizar intervalo inicial- intervalo final. Ex: Se utilizar 7-9 no campo das horas, significa que o comando será executado às 7h, 8h e 9h.
- Pode também utilizar valor1, valor2, ..., valorn. Ex: Se utilizar 0,5,10,15,20,25,30,35,40,45,50,45,50,55 no campo minutos, o comando será executado de cinco em cinco minutos
- **crontab -l** – apresenta a lista de tarefas escalonadas para o utilizador actual
- **crontab -u utilizador** – permite ao administrador (root) alterar o escalonamento de tarefas de um utilizador
- **crontab -r** – remove entradas
- **crontab -e** – permite editar – por defeito, utiliza o *vi*

Nota: Ver: `man crontab`

# Automatização de tarefas - at

- **at** – comando que permite executar uma tarefa (apenas **uma vez**) num determinada hora:minutos:segundos
    - As tarefas escalonadas estão em `/var/spool/at`
    - Estrutura da utilização: **at instante**
    - **instante** pode ser:
      - Minutos (ex: `at now +10 minutes` – inicia o comando daqui a dez minutos)
      - Hora (ex: `at now + 2 hours` – executa o comando daqui a duas horas)
      - Dias (ex: `at now + 1 day` )
      - Semanas (ex: `at now + 1 week` )
      - Normal (ex: `at 3:00 12/13/06` – executa o comando no dia 13 de Dezembro de 2006 às 3h00 )
    - Exemplo de utilização (executa daqui a uma hora a script *notas* e armazena os resultados no ficheiro *resultados* )
      - # **at now + 1 hour**
      - at> **/home/asist06/notas > /home/michael/resultados**
      - at> **Ctrl-D**
    - a
  - **atq** – permite verificar o estado das tarefas escalonadas
  - **atrm** – permite remover uma tarefa escalonada (**atrm número\_da\_tarefa**)
  - **SEGURANÇA**
    - Pode configurar quem tem acesso e quem não tem na utilização do escalonamento de tarefas. Para isso existem os ficheiros:
      - **/etc/cron.allow**
      - **/etc/cron.deny**
- Notas:
- Se os dois ficheiros não existirem então todos os utilizadores podem escalonar tarefas
  - Se existir o ficheiro `/etc/cron.allow` então apenas esse poderão usar o cron
  - Se apenas existe o ficheiro `/etc/cron.deny` então apenas esses utilizadores não podem utilizar o cron.
- Para configurar o acesso ao comando **at** deverá configurar os ficheiros **/etc/at.allow** e/ou **/etc/at.deny**

# Backups

- **Estratégias**

- **Total** – faz sempre a cópia de todos os ficheiros
- **Incremental** – apenas copia os ficheiros alterados desde o último backup (total ou incremental) - deve ser utilizado em conjunto com o “Total”. Diminui o tempo de backup. O tempo de restauro é superior, já que necessita do backup total + as cópias incrementais até ao dia da “falha”
- **Diferencial** – Copia os ficheiros alterados desde o último backup “Total” – deve ser utilizado em conjunto com o “Total”. Facilita a tarefa de restauro, já que são necessárias apenas duas cópias (Total + Diferencial)

- **Comandos**

- **Tapes:**

- **mt** – permite aceder e manipular uma tape

- Exemplos

- # mt -f /dev/nftape reten

- # mt -f /dev/nftape erase

- # mt -f /dev/nftape fsf 2

- # mt -f /dev/nftape eof

- **rmt** – comando utilizado para manipular uma tape localizada num servidor remoto

- **CD-R e CD-RW** (Os dados devem ser armazenados numa *imagem* e depois *escritos* num CD):

- **mkisofs** – cria uma *imagem* ISO-9660 para ser *escrita* pelo comando `cdrecord`

- Exemplo

- # mkisofs -r -J -o imagem\_cd.iso /source/directory

- Nota: pode utilizar o *loopback* para verificar se a imagem está correcta:

- # mount -t iso9660 -o loop imagem\_cd.iso /mnt/tmp

- **cdrecord** – permite *escrever* a imagem num CD-R

- **Exemplo:**

- # cdrecord -v speed=2 dev=0,6,0 cd\_image.iso



# Comandos: *dump*

- **dump** – verifica o sistema de ficheiros e analisa o que tem de ser copiado em função do nível. É baseado em i-nodes

`dump [-level] [-b blocksize] [-B records] [-f file] [-u] directório`

**Exemplo:** backup total ao domingo + backup incremental nos outros dias. O exemplo assume que se quer efectuar o backup de /, /usr e /var para uma tape SCSI em /dev/nst0

Comandos para o backup total

```
# dump 0uf /dev/nst0 /  
# dump 0uf /dev/nst0 /usr  
# dump 0uf /dev/nst0 /var
```

Comandos para o backup incremental

```
# dump 5uf /dev/nst0 /  
# dump 5uf /dev/nst0 /usr  
# dump 5uf /dev/nst0 /var
```

Nota: Para que os backups fossem sempre efectuados teria que os colocar no **cron**

- **dump remoto**

Exemplo:

```
# dump 0uf servidor:/dev/nst0 /usr
```

Uma forma mais eficiente:

```
# dump 0uf - /usr | rsh servidor dd of=/dev/rmt0
```

O servidor de backups tem de estar a usar rshd (ou in.rshd)

Para melhorar a segurança, usar:

```
# ssh client_system "dump 0uf - /usr" | dd of=/dev/rmt0
```

# Comandos: *restore*

- **restore** – comando que permite extraír ficheiros do backup. Pode utilizar o modo interactivo (que permite escolher os ficheiros a restaurar) ou então restaurar todos os ficheiros

**Exemplo 1:** restaura /usr/bin do backup efectuado com o *dump*:

```
# restore -i -f /dev/nst0
```

Dará origem ao seguinte:

```
restore> ls (na prompt do restore pode executar vários comandos)
```

```
::
```

```
local/
```

Para restaurar o directório bin temos de o adicionar à lista de itens a restaurar e depois extraír:

```
restore> add bin
```

```
restore> extract
```

Se tiver vários arquivos de backup numa única tape então é necessário especificar qual quer utilizar (usando a opção *-s*):

```
# restore -i -s 2 -f /dev/nrst0
```

**Exemplo 2:**

Para ver um determinado ficheiro, fazer (neste exemplo /local/bin/ntpd):

```
# restore -t -f /dev/nst0 local/bin/ntpd
```

Para ver uma lista com todos os ficheiros da tape, fazer:

```
# restore -t -f /dev/nst0
```

Para escolher os ficheiros a restaurar, fazer (neste exemplo é /usr/local/bin):

```
# restore -x /usr/local/bin -f /dev/nst0
```

- **restore remoto**

```
# restore rf servidor:/dev/nst0
```

# Comandos: tar

- Permite extrair ou *juntar* ficheiros

Um exemplo típico para efectuar o backup é:

```
# tar --create --verbose --gzip --one-file-system --same-permissions  
--file /dev/st0 / /home /usr/local
```

Pode fazer o mesmo usando abreviaturas:

```
# tar cvzlpf /dev/st0 / /home /usr/local (este comando efectua o backup de /, /home e /usr/local )
```

Para efectuar um backup incremental seria necessário usar:

```
# tar cvzlpf /dev/st0 --listed-incremental /root/increments / /home /usr/local
```

(Este exemplo efectua o backup incremental no ficheiro /root/increments e usa o conteúdo desse ficheiro para verificar quais os ficheiros em que irá fazer o backup)

Nota: para utilizar vários ficheiros de backup numa única tape, pode utilizar o comando **mt**

```
# tar cvzf archive.tgz /usr/local/bin (compacta os ficheiros no ficheiro archive.tgz. Para descompactar teria de usar: tar tvzf archive.tgz)
```

Nota: para usar caminhos absolutos é necessário acrescentar a opção “P”

Para efectuar o tar numa máquina remota, colocar antes do nome do ficheiro o nome da máquina:

```
# tar cvzf nome_da_máquina:/dev/st0 /usr/local/bin
```

```
# tar xvzf nome_máquina:/dev/st0
```

# Comandos: cpio

- Tenta juntar as vantagens do *dump* e do *tar*

## Exemplos:

Para armazenar os ficheiros do directório actual para um ficheiro arquivo.cpio, fazer:

```
# find . | cpio -o > arquivo.cpio
```

Para restaurar fazer:

```
# cpio -i < arquivo.cpio
```

Para obter uma lista dos ficheiros que estão no arquivo, fazer:

```
# cpio -t < arquivo.cpio
```

## Exemplos de funcionamento com uma tape:

```
# find . | cpio -o > /dev/st0
```

```
# cpio -i < /dev/st0
```

```
# cpio -t < /dev/st0
```

## Exemplos de funcionamento com um computador remoto:

```
# find . | cpio -o -F nome_máquina:/dev/st0
```

## Outras ferramentas de backup

- **AMANDA** (Advanced Maryland Automatic Network Disk Archiver)- é uma aplicação cliente/servidor . Permite efectuar o backup de drives de máquinas de uma rede local para um único servidor de backups
- **TAPER**
- **BRU**
- **Outras:**

<http://www.linux.org/apps/all/Administration/Backup.html>

## 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