Combinando Software-Raid e Logical Volume Manager

Jogue seus dados

Quando o espaço no disco rígido acaba, o sonho de qualquer usuário ou administrador de sistemas é adicionar outro disco e continuar a trabalhar. O Logical Volume Manager (LVM) torna este sonho realidade: ele concatena diversos dispositivos de bloco em um único dispositivo de armazenamento homogêneo, que pode ser facilmente modificado.

POR CARSTEN WIESE

Apesar do aumento constante da capacidade dos discos rígidos, o problema do armazenamento de dados continua a ser uma questão complicada. Além disso, o aumento da quantidade de dados a armazenar também cresce a cada dia. Em sistemas distribuídos, especialmente, nos quais os dados das aplicações e dos usuários são armazenados em diferentes discos e partições, é muito freqüente haver problemas de espaço. Adicionar novos discos rígidos ao sistema não é realmente uma solução para o problema: em busca de espaço, não é incomum que dados acabem por ser freqüentemente transferidos de um lugar para outro.

Uma solução viável são os chamados sistemas Raid, que concatenam diversos discos rígidos em um único dispositivo lógico. O artigo [1] descreve como reconfigurar um sistema Linux para utilizar Raid5 via software sem que haja necessidade de reinstalação do sistema. A utilização de sistemas Raid, entretanto, traz consigo uma desvantagem: o sistema fica limitado ao tamanho do Raid. Dessa forma, colocar as áreas de usuários em outros volumes ou adicionar discos rígidos com maior capacidade de armazenamento exigiria necessariamente um sistema Raid ou discos rígidos extra isolados (i.e., sem redundância), o que leva de novo ao mencionado problema de distribuição de dados.

Com o Logical Volume Manager (LVM) o administrador ganha a flexibilidade que faltava para, por exemplo, aumentar a capacidade total de armazenagem ou remover volumes mesmo com o sistema em operação. O que o LVM tem de tão especial é exatamente a capacidade de aumentar ou diminuir a quantidade de memória disponível com o sistema funcionando - desde que o sistema de arquivos também o permita.

Raid por software e Logical Volume Manager formam assim uma excelente combinação de alta disponibilidade e escalabilidade. E como "quitute" adicional temos a função de Snapshot do LVM, da qual falaremos mais adiante.

Muitas alternativas, difícil decisão

Atualmente existem duas excelentes implementações de gerenciador de volumes no Linux: de um lado temos o Logical Volume Manager (LVM), desenvolvido desde 1997 por Heinz Mauelshagen, cuja concepção é baseada no LVM do sistema HP-UX, e do outro temos o Enterprise Volume Management System (EVMS) [2] da IBM, colocado sob a GPL em 2002.

O desenvolvimento de dois diferentes gerenciadores de volumes foi motivo de algumas discussões na comunidade Linux. Muitos questionavam se não seria melhor unir os dois projetos. Para o kernel 2.6, Linus Torvalds decidiu adotar o LVM em sua segunda versão (LVM2).

Universos paralelos

A IBM, entretanto, continua a desenvolver o EVMS. O quadro "Viagem pelo EVMS" oferece uma breve introdução desta poderosa ferramenta, que é mais do que um simples gerenciador de volumes para Linux. Este artigo, todavia, tratará mais especificamente do LVM, que normalmente já está disponível no kernel 2.4 das atuais distribuições Linux.

O kernel 2.4 padrão utiliza ainda a versão 1.x do LVM. Para atualizar o sistema

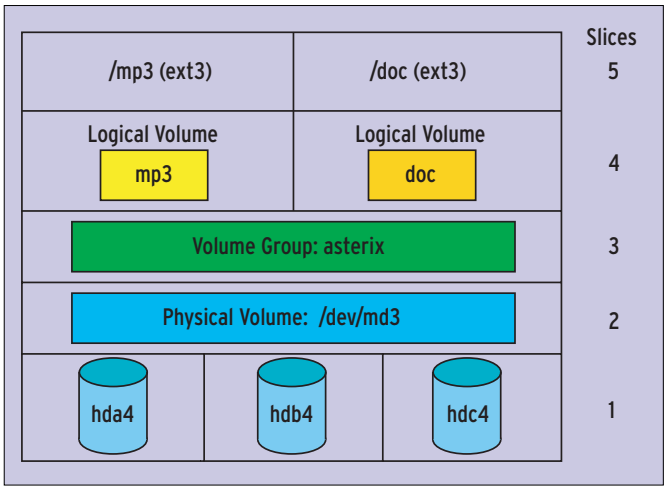

Figura 1:O grupo de volumes concatena um ou mais volumes físicos em uma unidade de dados lógica. Os volumes lógicos utilizam essa unidade de dados e são simples dispositivos de bloco (block devices).

para o LVM da versão 2.00.20 temos que aplicar a ele as alterações fornecidas pelo pacote disponível em [3]. Esse pacote contém, além dos patches para o kernel 2.4, também a versão adequada dos programas de gerenciamento do LVM, bem como as alterações necessárias do mapeador de dispositivos (aka "device mapper") do sistema.

No que tange a gerenciadores de volume, o SuSE Linux 9.1 oferece a maior variedade de escolha entre as distribuições Linux, incluindo o LVM, LVM2 e o EVMS. O Fedora Linux inclui o LVM e LVM2 e o Red Hat dispõe atualmente apenas do LVM em suas versões Professional Workstation e Enterprise.

Os exemplos que mostraremos a seguir funcionam com a versão 1.x do LVM. As novidades da versão 2 estão descritas no quadro "Novidades no LVM2".

LVM em Raid por Software

O sistema de trabalho do LVM é dividido em três partes: o volume físico (PV), o grupo de volumes (VG) e o volume lógico (LV). A Figura 1 esclarece a terminologia e ilustra as camadas de abstração do sistema.

A base ilustrada na camada 1 representa os discos rígidos reais - no nosso caso três discos IDE. Sobre ela, na segunda camada, é constituído o chamado "Multiple Device" (*/dev/md3*) do Raid por software - podemos dizer, entretanto, que ele é um volume físico (PV), uma vez que o Raid por software emula um dispositivo físico. No caso do Raid por hardware, a controladora Raid combina os discos rígidos ilustrados na camada 1 em um único dispositivo físico de armazenagem (dispositivo de bloco).

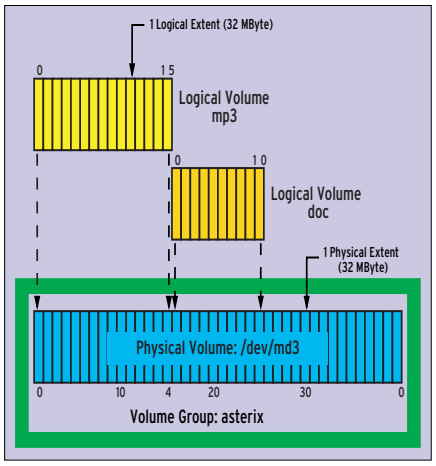

Figura 2: Os blocos lógicos têm o mesmo tamanho de seus blocos físicos correspondentes. Em modo linear, os volumes lógicos ficam diretamente um atrás do outro no volume físico.

No início de cada volume físico (PV) encontra-se, no primeiro bloco físico do volume (PE), a área de descrição do grupo de volumes (VGDA). Ela contém meta-dados da configuração do LVM, é dividida em quatro setores e pode ser comparada à tabela de partição de um disco rígido comum. No diretório */etc/lvmconf* é gravada uma cópia da VGDA.

O grupo de volumes "asterix" na camada 3 combina um ou mais volumes físicos em um único - neste exemplo apenas */dev/md3*. É como se fosse criada uma nova unidade de armazenagem que se estendesse por todos os volumes físicos. O grupo de volumes é totalmente flexível, podendo ser facilmente expandido com volumes físicos adicionais.

Na quarta camada estão os volumes lógicos: no exemplo são eles *mp3* e *doc*. Cada volume lógico (LV) é um dispositivo de blocos separado, com o nome do volume lógico correspondente, no nosso exemplo */dev/asterix/mp3* e */dev/asterix/doc*, cujo tamanho pode ser modificado posteriormente. Em ambos os volumes deve ser gerado um sistema de arquivos, que será montado no sistema como de costume (camada 5).

A unidade indivisível...

Vamos agora "dissecar" as camadas dois, três e quatro. A menor unidade de armazenamento físico de um sistema LVM é o bloco físico (PE), citado anteriormente. Cada volume físico, quando o grupo de volumes é criado, é dividido em blocos físicos de mesmo tamanho, por padrão 32 MBytes. No total, um volume físico pode conter até 65535 blocos físicos, o que resulta em um tamanho máximo de 2 TBytes. Cada bloco físico recebe uma identificação numérica (ID) que começa do 0 em cada volume físico - desse modo, cada bloco físico dentro de um volume físico tem seu próprio ID.

Analogamente, cada volume lógico (LV) é dividido em blocos lógicos (LE). O tamanho dos blocos lógicos e físicos é sempre o mesmo; um bloco lógico tem assim, por padrão, 32 MBytes de tamanho. Como no caso dos blocos físicos, os blocos lógicos são numerados a partir do 0 e estão conectados logicamente ao seu bloco físico correspondente, de modo que podemos dizer que cada bloco lógico (LE) está mapeado exatamente contra um único bloco físico (PE). A Figura 2 mostra como os volumes lógicos *mp3* e *doc* estão mapeados no volume físico */dev/md3* do grupo *asterix*.

Quando um programa acessa um determinado byte do volume lógico *doc*, primeiramente o ID do bloco lógico no qual os dados se encontram é calculado. Por meio da correlação dos blocos lógicos com os físicos, o LVM identifica o ID do bloco físico apropriado e acessa a posição correspondente em */dev/md3* - não se esqueça que *doc* está mapeado no volume físico */dev/md3*.

Fatiado

Há dois modos de distribuição para os dados nos volumes físicos: linear e em fatias ("striping"). Normalmente, o LVM utiliza o modo linear, adequado para grupos de volume que consistem apenas de um volume físico e um disco rígido ou sistema Raid. Os blocos lógicos são associados de modo linear e crescente aos blocos físicos.

Para grupos de volume que se estendam por vários volumes físicos, a opção de distribuição em fatias torna-se interessante. Blocos lógicos vizinhos são, neste caso, associados a blocos físicos de dois volumes físicos diferentes: no caso do acesso a dados que estejam armazenados em blocos lógicos vizinhos, tais dados vêm de diferentes discos rígidos, o que em determinados casos pode aumentar a velocidade de transferência. A principal desvantagem é que esse tipo de volume lógico não pode ser expandido posteriormente. Por isso, na prática, o modo linear é quase o único a ser utilizado. O comando

```
lvcreate -L300 -nmp3 asterix
```

produz um volume lógico em modo linear de 320 MByte chamado *mp3* dentro do grupo de volumes *asterix*. Se desejarmos utilizar o modo em fatias, temos que adicionar o parâmetro *-i*, bem como o número de fatias - por exemplo *-i2*, no caso de dois volumes físicos.

O volume lógico *mp3* do exemplo acima deveria ter 300 MByte. No entanto, como um bloco lógico tem 32 MByte, a partição foi expandida automaticamente para o próximo múltiplo de 32 MByte.

Ferramentas

O pacote LVM contém uma gama completa de ferramentas de administração, com as quais as diferentes camadas descritas anteriormente podem ser criadas e manipuladas. Os programas *vgcreate*, *vgdisplay*, *vgchange* e *vgremove*, por exemplo, são responsáveis pela manipulação de grupos de volumes.

Para fazer um backup e restaurar os arquivos de configuração do grupo de volumes temos os utilitários *vgcfgbackup* e *vgcfgrestore*; *vgreduce* e *vgextend* diminuem ou expandem um grupo de volumes. Para transferir um grupo de volumes de um computador para outro temos *vgexport* e *vgimport*. Para dividir ou concatenar vários grupos de volumes utiliza-se *vgsplit* e *vgmerge*; *vgscan* procura por grupos de volumes perdidos em dispositivos de blocos e *vgrename* modifica seus nomes.

Para as duas outras camadas de abstração - volumes físicos e lógicos - há também ferramentas semelhantes, cujos nomes só se diferenciam dos citados anteriormente pelas duas primeiras letras: *pv* para volumes físicos e *lv* para volumes lógicos.

Excursão pelo EVMS

O Enterprise Volume Management System (EVMS) da IBM fornece uma estrutura básica para todos as variações do gerenciamento de volumes. Sua arquitetura consiste em um modelo baseado em "plugins", no qual se podem adicionar módulos isoladamente, em forma de extensões. Ele é compatível com LVM, integra Raid por software ("Multiple Devices") e oferece suporte para os sistemas de arquivo atuais. A resetorização de blocos defeituosos ("Bad Block Relocation" - BBR) e o suporte para cluster também não são novidades para o sistema. A Figura 5 ilustra os módulos do kernel isoladamente, bem como os plugins que acompanham o kernel padrão do SuSE Linux 9.0.

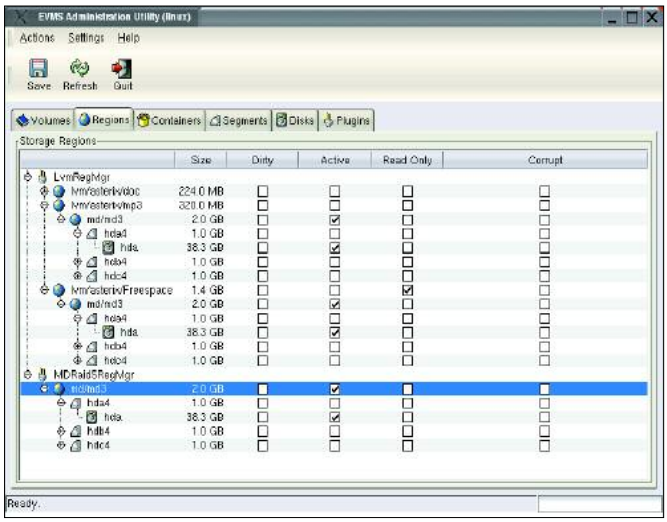

Figura 6:O EVMS pode ser administrado via interface gráfica ou em modo texto. A nomenclatura utilizada é diferente da usada pelo LVM.

Além da interface gráfica de administração do EVMS (Figura 6) há também uma interface em modo texto. A terminologia do EVMS é um pouco diferente da do LVM - volumes físicos são chamados de segmentos, grupos de volumes são conhecidos por containers e os volumes lógicos são chamados de regiões. A documentação disponível na página do projeto na Internet [2] facilita a familiarização com o sistema.

Planejamento

O volume físico é um array RAID nível 5 por software, composto por três discos rígidos IDE, conforme mostrado na Figura 1. Os dados de */mp3* e */doc* devem ser armazenados no volume RAID e o uso do LVM é indicado caso seja necessária uma futuras expansão no espaço de armazenamento.

Instruções sobre a configuração de um Raid5 por software a partir de três discos rígidos são encontradas em [1]. Nos próximos exemplos partimos do pressuposto de que o sistema Raid */dev/md3* já esteja funcionando e inicializado.

No kernel do Linux na categoria *Multiple Device Support* os itens *RAID support* e *Logical Volume Manager (LVM) support* devem estar ativados (veja Figura 3). No caso do kernel 2.4 "vanilla" precisamos ainda aplicar as alterações mencionadas no início deste artigo para atualizar o LVM para a versão 2.

Caso o suporte a LVM não esteja compilado no kernel e sim disponível na

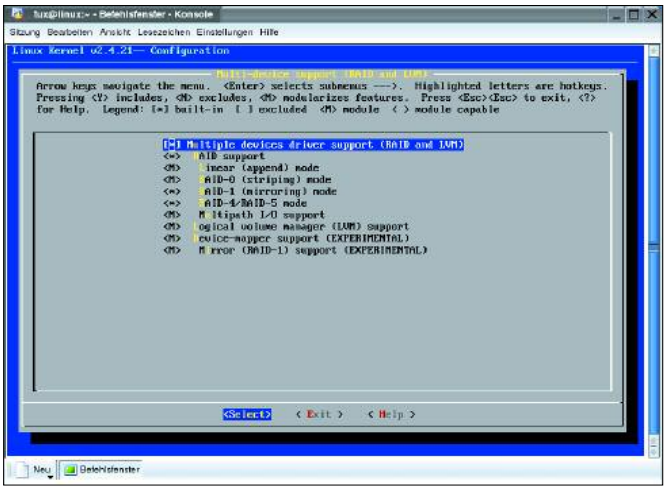

Figura 3: O kernel 2.4 ainda contém a versão 1 do Logical Volume Manager (LVM). Para utilizar o LVM2 ele precisa ser alterado com os "patches" disponíveis em [3].

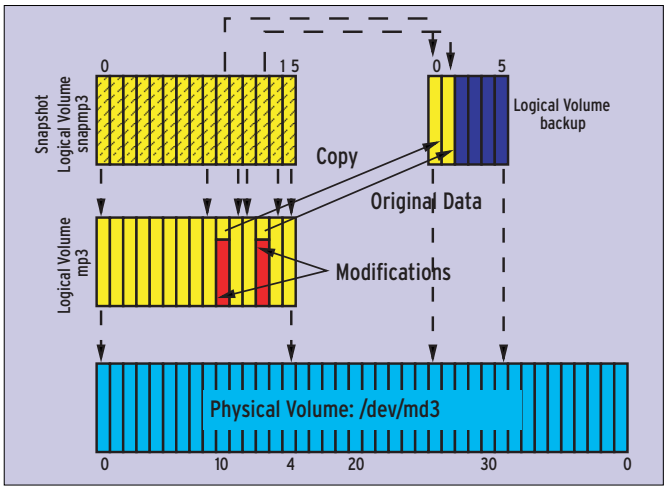

Figura 4: Um snapshot é um volume lógico apenas de leitura, que garante a consistência dos dados em um grupo de volumes para backup, embora o volume lógico original permaneça alterável.

O que há de novo no LVM2

As três principais modificações ocorridas no LVM2 são o mapeador de dispositivos (*Device Mapper*), o novo formato dos metadados e o arquivo de configuração *lvm.conf*. O novo mapeador de dispositivos permite criar um novo dispositivo de blocos em um dispositivo já existente. O formato dos metadados do LVM2 é mais estável e estruturado de maneira mais eficiente.

No novo arquivo de configuração (*lvm.conf*) é possível incluir parâmetros para cada dispositivo isoladamente, bem como ajustar o backup dos metadados e o nível de *logging* (registro) do sistema. O LVM2 é retro-compatível com o LVM1. Para converter os metadados do grupo de volumes *asterix* ao novo formato, use o comando *vgconvert -M2 asterix*.

forma de um módulo, precisamos colocá-lo na RAM Disk inicial do sistema (*initrd*). Isso pode ser feito pelo comando *lvmcreate_initrd*.

Criação dos volumes

O volume físico é o Raid5 por software */dev/md3*. O comando *pvcreate /dev/md3* prepara o dispositivo de blocos da maneira adequada. Sua configuração pode ser verificada através do comando *pvdisplay /dev/md3*.

O grupo de volumes *asterix* é criado pelo comando *vgcreate asterix /dev/md3*. No caso de vários volumes físicos, adicione o nome do dispositivo de blocos correspondente ao comando.

A quarta camada, com os volumes lógicos *mp3* e *doc*, é criada pelo comando *lvcreate -L300 -nmp3 asterix* e *lvcreate -L200 -ndoc asterix*. Com isso, o tamanho real dos volumes lógicos é de 320 MByte para o primeiro e 224 MByte para o segundo - lembre-se que o LVM utiliza somente blocos lógicos com tamanhos múltiplos de 32 MByte, como já dissemos anteriormente.

No diretório */dev/asterix* devem existir agora três novos dispositivos: o dispositivo de caracteres ("character device") *group* e os dois dispositivos de blocos *mp3* e *doc*. Os sistemas de arquivos são criados da mesma forma que nos dispositivos de blocos comuns, com os comandos *mkfs.ext3 /dev/asterix/mp3* e *mkfs.ext3 /dev/asterix/doc*.

A montagem dos volumes lógicos no sistema de arquivos também não se diferencia em nada daquela dos dispositivos de blocos comuns. Entretanto, antes de ser montados, os programas *vgscan* e *vgchange -ay* precisam ser executados. Além disso, durante o processo de desligamento do sistema, deve-se rodar o programa *vgchange -an*. De preferência, esses comandos devem ser adicionados aos scritps de inicialização e desligamento do sistema: */etc/init.d/boot* e */etc/init.d/halt*, de acordo com a distribuição Linux utilizada.

Quando o espaço do volume lógico *mp3* acabar, basta usar o comando *lvextend -L 500M /dev/asterix/mp3* para expandi-lo para 500 MByte. Em seguida, o sistema de arquivos tem que ser ajustado ao novo tamanho do volume lógico, utilizando - no caso do sistema de arquivos Ext3, por exemplo - o comando *resize2fs /dev/asterix/mp3*.

Snapshots

A função de snapshot do LVM é especialmente útil para efetuar backups de um grupo de volumes. Ela permite que se crie, a qualquer momento, um "clone" do volume lógico que não pode ser modificado diretamente. O snapshot assim criado é uma cópia "congelada" (*frozen image*) do volume lógico original, que deve ser montado apenas para leitura no sistema de arquivos durante o processo de backup. O volume lógico original permanece alterável e modificações efetuadas em qualquer de seus blocos lógicos ocasionam automaticamente a cópia do bloco lógico original para o snapshot (Figura 4).

A quantidade de memória designada para o snapshot deve ser suficiente para absorver pelo menos a mesma quantidade de dados escrita no volume lógico original durante o backup. De preferência, pode-se criar um grupo de volumes à parte, por exemplo *operator*, em um disco rígido extra. Se a quantidade de dados a serem gravados no snapshot superar o tamanho indicado durante sua criação, ou se o computador for reinicializado, o snapshot desaparece automaticamente – os dados escritos nele até então são copiados de volta para o volume lógico original.

Instruções detalhadas para a realização de backups com snapshots podem ser encontradas no tutorial de LVM [4], no capítulo 11.4.

Riscos do "fatiamento"

Dois tutoriais ([5] e [6]) descrevem como instalar um sistema de arquivos raiz em um Raid por software com LVM. Isso faz sentido em alguns casos, mas aumenta em muito o trabalho do administrador e, em caso de falha em algum dos componentes do sistema de armazenagem, as coisas podem se complicar. É importante verificar em que casos o uso do LVM faz sentido e é necessário.

Outra ponto que deve ser muito bem analisado é o emprego do modo em fatias (striping), pois ele nem sempre é sinônimo de melhor desempenho – al´m de ter a desvantagem de engessar o tamanho do volume lógico, que não pode mais ser alterado.

A falha de um disco rígido é compensada pelo sistema Raid, mas ela pode tomar proporções catastróficas se um volume lógico no modo em fatias estiver distribuído em dois ou mais discos rígidos. Neste caso o conjunto de dados inteiro torna-se inconsistente, enquanto que para volumes lógicos em modo linear apenas os dados do volume lógico afetado pela falha são atingidos por ela. ■

Legenda

LE	Bloco Lógico
LV	Volume Lógico
LVM	Logical Volume Manager
PE	Bloco Físico
PV	Volume Físico
VG	Grupo de Volumes
VGDA	Descritor do Grupo de Volumes

INFORMAÇÕES

[1] Carsten Wiese, "Volks-Raid": Linux-Magazin 07/2003, página 50; *http://www.linux-magazin.de/Artikel/ausgabe/2003/07/volksraid/volksraid.html]*

[2] Enterprise Volume Management System (EVMS): *http://evms.sf.net*

[3] Patch e mapeador de dispositivos para o LVM2: *http://sources.redhat.com/pub/lvm2/*

[4] Tutorial de LVM: *http://tldp.org/HOWTO/LVM-HOWTO/*

[5] Tutorial de implementação de um sistema raiz com LVM e Raid por software: *http://www.midhgard.it/docs/lvm/html/index.html*

[6] Tutorial de implementação de um sistema raiz em Raid por software: *http://karaolides.com/computing/HOWTO/lvmraid*