LVM Snapshot 简介

2014/9/3
renyl

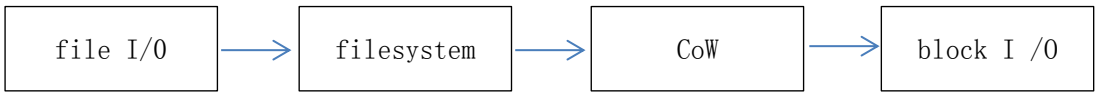
1 基本原理

1) Logical Volume Manager (LVM)提供了对任意一个 Logical Volume(LV)做“快照”(snapshot)的功能，以此来获得一个分区一致性备份。

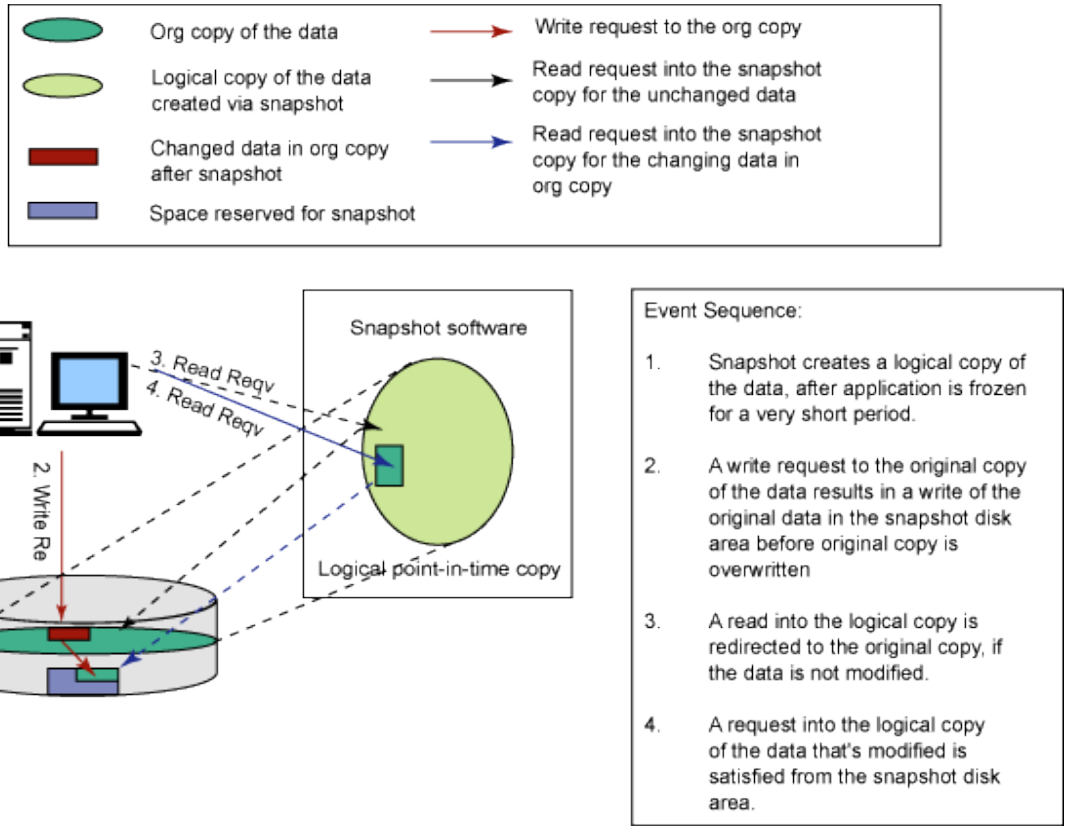
2) LVM 的 snapshot 是通过“写时复制”(copy on write) 的方法实现：
   a) 当一个 snapshot 创建的时候，仅拷贝原始卷里数据的元数据(meta-data)。创建的时候，并不会有数据的物理拷贝，因此 snapshot 的创建几乎是实时的。
   b) 当原始卷上有写操作执行时，snapshot 跟踪原始卷块的改变，这个时候原始卷上将要改变的数据在改变之前被拷贝到 snapshot 预留的空间里，因此这个原理的实现叫做写时复制(copy-on-write)。

3) 在写操作写入块之前，CoW 会将原始数据移动到 snapshot 空间里，这样就保证了所有的数据在 snapshot 创建时保持一致。而对于 snapshot 的读操作，如果是读取数据块是没有修改过的，那么会将读操作直接重定向到原始卷上，如果是要读取已经修改过的块，那么就读取拷贝到 snapshot 中的块。

4) 这样，通常的文件 I/O 流程有一个改变，那就是在文件系统和设备驱动之间增加了一个 cow 层，变成了下面这个样子：



下图描述了 COW 的实现原理：



说明：

1) 采取 CoW 实现方式时，snapshot 空间的大小并不需要和原始卷一样大，其大小仅仅只需要考虑，从 shapshot 创建到释放这段时间内，估计块的改变量有多大。
2) 如果 snapshot 的空间记录满了原始卷块变换的信息，那么这个 snapshot 立刻被释放，从而无法使用，从而导致这个 snapshot 无效。
3) 因此，一定要 snapshot 的生命周期里，做完需要做得事情。否则，当原始卷的改变量大于 snapshot 空间大小时，就无法恢复到分区原始状态了。

2 LVM 基本命令

2.1 修改分区 System ID

```
[root@localhost /]# parted  /dev/sde
GNU Parted 3.1
Using /dev/sde
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) p
Model: LSI RAID 5/6 SAS 6G (scsi)
Disk /dev/sde: 146GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type     File system  Flags
 1      512B    20.0GB  20.0GB  primary  xfs          lvm
 2      20.0GB  40.0GB  20.0GB  primary  xfs

(parted) set 2 lvm
New state?  [on]/off? on
(parted) p
Model: LSI RAID 5/6 SAS 6G (scsi)
Disk /dev/sde: 146GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type     File system  Flags
 1      512B    20.0GB  20.0GB  primary  xfs          lvm
 2      20.0GB  40.0GB  20.0GB  primary  xfs          lvm

(parted) q
Information: You may need to update /etc/fstab.

[root@localhost /]#
```

2.2 建立 PV（Physical Volume）

```
[root@localhost /]# pvcreate /dev/sde1
WARNING: xfs signature detected on /dev/sde1 at offset 0. Wipe it? [y/n] y
  Wiping xfs signature on /dev/sde1.
  Physical volume "/dev/sde1" successfully created
[root@localhost /]# pvcreate /dev/sde2
WARNING: xfs signature detected on /dev/sde2 at offset 0. Wipe it? [y/n] y
  Wiping xfs signature on /dev/sde2.
  Physical volume "/dev/sde2" successfully created
```

2.3 建立 VG（Volume Group）

```
[root@localhost ~]# vgcreate  my_vg   /dev/sde1  /dev/sde2
  Volume group "my_vg" successfully created
[root@localhost ~]#
```

2.4 建立 LV（Logical Volume）

```
[root@localhost ~]# lvcreate -L 10GB -n my_lv my_vg
  Logical volume "my_lv" created
[root@localhost ~]#
```

注：

1）-L：后面接容量，容量的单位可以是 KB、MB、GB 等。

2）-n：后面接 LV 的名称。

2.5 格式化分区

```
[root@localhost ~]# mkfs.xfs  /dev/my_vg/my_lv
meta-data=/dev/my_vg/my_lv       isize=256    agcount=4, agsize=655360 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=0
data     =                       bsize=4096   blocks=2621440, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=0
log      =internal log           bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
[root@localhost /]# mount /dev/my_vg/my_lv  /mnt/
[root@localhost /]# df -hT
Filesystem              Type      Size  Used Avail Use% Mounted on
/dev/mapper/vgrhel-root xfs        20G  5.7G   14G  30% /
devtmpfs                devtmpfs   16G     0   16G   0% /dev
tmpfs                   tmpfs      16G   80K   16G   1% /dev/shm
tmpfs                   tmpfs      16G  9.3M   16G   1% /run
tmpfs                   tmpfs      16G     0   16G   0% /sys/fs/cgroup
/dev/sda1               xfs       509M  121M  388M  24% /boot
/dev/mapper/my_vg-my_lv xfs        10G   33M   10G   1% /mnt
```

2.6 扩展 VG 和 LV 大小

```
[root@localhost /]# parted  /dev/sde
GNU Parted 3.1
Using /dev/sde
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) p
Model: LSI RAID 5/6 SAS 6G (scsi)
```

```
Disk /dev/sde: 146GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type     File system  Flags
 1      512B    20.0GB  20.0GB  primary               lvm
 2      20.0GB  40.0GB  20.0GB  primary               lvm
 3      40.0GB  60.0GB  20.0GB  primary  xfs          lvm
[root@localhost /]# umount /mnt/
[root@localhost /]# pvcreate  /dev/sde3
WARNING: xfs signature detected on /dev/sde3 at offset 0. Wipe it? [y/n] y
  Wiping xfs signature on /dev/sde3.
  Physical volume "/dev/sde3" successfully created
[root@localhost /]# vgextend my_vg  /dev/sde3
    Volume group "my_vg" successfully extended
[root@localhost /]# vgdisplay
    --- Volume group ---
  VG Name               my_vg
  System ID
  Format                lvm2
  Metadata Areas        3
  Metadata Sequence No  10
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                1
  Open LV               0
  Max PV                0
  Cur PV                3
  Act PV                3
  VG Size               55.88 GiB
  PE Size               4.00 MiB
  Total PE              14304
  Alloc PE / Size       2560 / 10.00 GiB
  Free  PE / Size       11744 / 45.88 GiB
  VG UUID               YecA1m-RiT2-cq5n-ownL-Ej2i-2eSe-J1RfAg
[root@localhost /]# lvextend  -L +10GB /dev/my_vg/my_lv
    Extending logical volume my_lv to 20.00 GiB
    Logical volume my_lv successfully resized
[root@localhost /]# mount /dev/my_vg/my_lv  /mnt/
[root@localhost /]# df -hT
Filesystem              Type      Size  Used Avail Use% Mounted on
/dev/mapper/vgrhel-root xfs        20G  5.7G   14G  30% /
devtmpfs                devtmpfs   16G     0   16G   0% /dev
tmpfs                   tmpfs      16G   80K   16G   1% /dev/shm
tmpfs                   tmpfs      16G  9.3M   16G   1% /run
tmpfs                   tmpfs      16G     0   16G   0% /sys/fs/cgroup
/dev/sda1               xfs       509M  121M  388M  24% /boot
/dev/mapper/my_vg-my_lv xfs        10G   33M   10G   1% /mnt
[root@localhost /]# xfs_growfs   /mnt/
```

```
meta-data=/dev/mapper/my_vg-my_lv isize=256    agcount=4, agsize=655360 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=0
data     =                       bsize=4096   blocks=2621440, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=0
log      =internal               bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
data blocks changed from 2621440 to 5242880
[root@localhost /]# df -hT
Filesystem              Type      Size  Used Avail Use% Mounted on
/dev/mapper/vgrhel-root xfs        20G  5.7G   14G  30% /
devtmpfs                devtmpfs   16G     0   16G   0% /dev
tmpfs                   tmpfs      16G     0   16G   0% /sys/fs/cgroup
/dev/sda1               xfs       509M  121M  388M  24% /boot
/dev/mapper/my_vg-my_lv xfs        20G   33M   20G   1% /mnt
[root@localhost /]#
```

2.7 LVM 删除

```
[root@localhost /]# umount /mnt/
[root@localhost /]# lvremove /dev/my_vg/my_lv
Do you really want to remove active logical volume my_lv? [y/n]: y
  Logical volume "my_lv" successfully removed
[root@localhost /]# vgchange  -a n my_vg    //让这个 vg 不具有 Active 的标志。
  0 logical volume(s) in volume group "my_vg" now active
[root@localhost /]# vgremove  my_vg
  Volume group "my_vg" successfully removed
[root@localhost /]# pvremove  /dev/sde1
  Labels on physical volume "/dev/sde1" successfully wiped
[root@localhost /]# pvremove  /dev/sde2
  Labels on physical volume "/dev/sde2" successfully wiped
[root@localhost /]# pvremove  /dev/sde3
  Labels on physical volume "/dev/sde3" successfully wiped
[root@localhost /]#
```

2.8 相关命令

任务	PV	VG	LV
搜索（scan）	pvscan	vgscan	lvscan
建立（create）	pvcreate	vgcreate	lvcreate
列出（display）	pvdisplay	vgdisplay	lvdisplay
增加（extend）	-	vgextend	lvextend
减少（reduce）	-	vgreduce	lvreduce
删除（remove）	pvremove	vgremove	lvremove
改变容量（resize）	-	lvresize	-

3 LVM Snaphost

3.1 Backup

```
[root@localhost /]# lvcreate -L 5GB -s -n my_snapshot  /dev/my_vg/my_lv
  Logical volume "my_snapshot" created
[root@localhost /]#
```

3.2 Restore

```
[root@localhost /]#lvconvert  --merge  /dev/my_vg/my_snapshot
  Logical volume my_vg/my_lv contains a filesystem in use.
  Can't merge over open origin volume.
  Merging of snapshot my_snapshot will start next activation.
[root@localhost /]#reboot   //重启后生效
```

4 注意事项

1) 当 Snapshot 的空间记录满了原始卷块变换的信息，那么这个 Snapshot 将立刻被释放，从而导致无法使用这个 Snapshot。因此，在建立 Snapshot 时，需要预估计原始卷块需要做多大的修改量。

2) 在建立 Snapshot 之前确保被备份的文件都在磁盘上，因此需要 umount 这个分区或者执行命令“echo 3 > /proc/sys/vm/drop_caches”。

3) 在使用 Snapshot 进行恢复时，如果原始卷块被 umount 的话，恢复立刻生效。如果原始卷块正在 mount 被使用中，那么系统重启后将生效。

4) 系统的 boot 分区不能使用 LVM 进行管理。

5 附录

自动备份与恢复脚本：

```
[root@localhost renyl]# cat lvm_snapshot.sh
#!/bin/bash

#Program:
#  backup and restore with filesystem snapshots
#
#Histroy:
#  renyl  2014/7/1  0.1version
#
#  renyl  2014/7/2  0.2version
#     add:auto backup after restore snapshot


help()
{
   set +x
   echo "Parameter is wrong."
   echo "Usage: $0  -check"
   echo "Usage: $0  -backup  <size>  <snapshot_name>  <full_backup_lv>"
   echo "Usage: $0  -restore <full_snapshot_name>"
   echo "Example: $0 -backup 5GB renyl_snap /dev/vgsnap/lvroot"
   echo "Example: $0 -restore /dev/vgsnap/renyl_snap"
   exit 1
}

option=$1

case ${option} in

   "-check")
      vgdisplay
      lvdisplay
      exit 0
      ;;

   "-backup")
      :
      ;;

   "-restore")
      full_snapshot_name=$2

      if [ -e "${full_snapshot_name}" ];then

         snap_path_line=`lvdisplay  | grep -n  "${full_snapshot_name}" | awk
'BEGIN {FS=":"}; {print $1}'`
```

```
        backup_vg_line=`expr ${snap_path_line} + 2`    #magin number
        backup_lv_line=`expr ${snap_path_line} + 6`    #magic number
        snap_size_line=`expr ${snap_path_line} + 11`   #magin number
        snap_lv_line=`expr ${snap_path_line} + 1`      #magin number

        backup_vg_name=`lvdisplay   | sed -n "${backup_vg_line}p" | awk
'{print $3}'``  #magic number
        backup_lv_name=`lvdisplay   | sed -n "${backup_lv_line}p" | awk
'{print $7}'``  #magic number
        snapshot_size=`lvdisplay    | sed -n "${snap_size_line}p" | awk
'{print $3}''`   #magic number
        snapshot_name=`lvdisplay  | sed -n "${snap_lv_line}p" | awk '{print
$3}'``     #magic number

        PWD=`pwd`
echo  "${PWD}/lvm_snapshot.sh  -backup  ${snapshot_size}  ${snapshot_name}
/dev/${backup_vg_name}/${backup_lv_name}"  >>/etc/rc.d/rc.local

        chmod +x /etc/rc.d/rc.local

        lvconvert  --merge  ${full_snapshot_name}

        echo "-----------------------------------------------"
        echo  "Warning: It will be effective after system reboot."

     else
        echo "File ${full_snapshot_name} is not exsit!"
        echo "-----------------------------------------------"
        help
     fi

     exit 0
     ;;

  *)
     help
     ;;
esac

parameter_num=$#

if [ ${parameter_num} -ne  "4" ];then

   echo "Run backup need  4 parameter."
   echo "-----------------------------------------------"
   help
fi

snap_size=$2
snap_name=$3
full_backup_lv=$4
```

```
lvcreate  -L ${snap_size}  -s -n ${snap_name} ${full_backup_lv}

remove_line=`cat /etc/rc.d/rc.local | grep -n "lvm_snapshot.sh" | awk 'BEGIN
{FS=":"}; {print $1}'`

if [ -n "${remove_line}" ];then

sed -i "${remove_line}d" /etc/rc.d/rc.local

fi
```