5.4.4. Creating Thinly-Provisioned Logical Volumes

7-9 minutes

As of the Red Hat Enterprise Linux 6.4 release, logical volumes can be thinly provisioned. This allows you to create logical volumes that are larger than the available extents. Using thin provisioning, you can manage a storage pool of free space, known as a thin pool, which can be allocated to an arbitrary number of devices when needed by applications. You can then create devices that can be bound to the thin pool for later allocation when an application actually writes to the logical volume. The thin pool can be expanded dynamically when needed for cost-effective allocation of storage space.

This section provides an overview of the basic commands you use to create and grow thinly-provisioned logical volumes. For detailed information on LVM thin provisioning as well as information on using the LVM commands and utilities with thinly-provisioned logical volumes, see the `lvmthin`(7) man page.

Thin volumes are not supported across the nodes in a cluster. The thin pool and all its thin volumes must be exclusively activated on only one cluster node.

To create a thin volume, you perform the following tasks:

1. Create a volume group with the `vgcreate` command.
2. Create a thin pool with the `lvcreate` command.
3. Create a thin volume in the thin pool with the `lvcreate` command.

You can use the `-T` (or `--thin`) option of the `lvcreate` command to create either a thin pool or a thin volume. You can also use `-T` option of the `lvcreate` command to create both a thin pool and a thin volume in that pool at the same time with a single command.

The following command uses the `-T` option of the `lvcreate` command to create a thin pool named `mythinpool` that is in the volume group `vg001` and that is 100M in size. Note that since you are creating a pool of physical space, you must specify the size of the pool. The `-T` option of the `lvcreate` command does not take an argument; it deduces what type of device is to be created from the other options the command specifies.

```
#
 lvcreate -L 100M -T vg001/mythinpool

  Rounding up size to full physical extent 4.00 MiB
  Logical volume "mythinpool" created
#
 lvs

  LV            VG     Attr     LSize   Pool Origin Data%  Move Log Copy% Convert
  my mythinpool vg001  twi-a-tz 100.00m               0.00
```

Copied!

The following command uses the `-T` option of the `lvcreate` command to create a thin volume named `thinvolume` in the thin pool `vg001/mythinpool`. Note that in this case you are specifying virtual size, and that you are specifying a virtual size for the volume that is greater than the pool that contains it.

```
#
 lvcreate -V1G -T vg001/mythinpool -n thinvolume
```

```
  Logical volume "thinvolume" created
#
 lvs

  LV          VG       Attr     LSize   Pool       Origin Data%  Move Log Copy%
Convert
  mythinpool  vg001    twi-a-tz 100.00m                     0.00
  thinvolume  vg001    Vwi-a-tz   1.00g mythinpool          0.00
```

Copied!

The following command uses the `-T` option of the `lvcreate` command to create a thin pool and a thin volume in that pool by specifying both a size and a virtual size argument for the `lvcreate` command. This command creates a thin pool named `mythinpool` in the volume group `vg001` and it also creates a thin volume named `thinvolume` in that pool.

```
#
 lvcreate -L 100M -T vg001/mythinpool -V1G -n thinvolume

  Rounding up size to full physical extent 4.00 MiB
  Logical volume "thinvolume" created
#
 lvs

  LV           VG       Attr     LSize   Pool     Origin Data%  Move Log Copy%
Convert
  mythinpool   vg001    twi-a-tz 100.00m                   0.00
  thinvolume   vg001    Vwi-a-tz   1.00g mythinpool        0.00
```

Copied!

You can also create a thin pool by specifying the `--thinpool` parameter of the `lvcreate` command. Unlike the `-T` option, the `--thinpool` parameter requires an argument, which is the name of the thin pool logical volume that you are creating. The following example specifies the `--thinpool` parameter of the `lvcreate` command to create a thin pool named `mythinpool` that is in the volume group `vg001` and that is 100M in size:

```
#
 lvcreate -L 100M --thinpool mythinpool vg001

  Rounding up size to full physical extent 4.00 MiB
  Logical volume "mythinpool" created
#
 lvs

  LV          VG     Attr     LSize   Pool Origin Data%  Move Log Copy% Convert
  mythinpool  vg001  twi-a-tz 100.00m               0.00
```

Copied!

Striping is supported for pool creation. The following command creates a 100M thin pool named `pool` in volume group `vg001` with two 64 kB stripes and a chunk size of 256 kB. It also creates a 1T thin volume, `vg00/thin_lv`.

```
#
 lvcreate -i 2 -I 64 -c 256 -L100M -T vg00/pool -V 1T --name thin_lv
```

Copied!

You can extend the size of a thin volume with the `lvextend` command. You cannot, however, reduce the size of a thin pool.

The following command resizes an existing thin pool that is 100M in size by extending it another 100M.

```
#
 lvextend -L+100M vg001/mythinpool

  Extending logical volume mythinpool to 200.00 MiB
  Logical volume mythinpool successfully resized
#
 lvs

  LV           VG       Attr     LSize   Pool     Origin Data%  Move Log Copy%
Convert
  mythinpool   vg001    twi-a-tz 200.00m                    0.00
  thinvolume   vg001    Vwi-a-tz   1.00g mythinpool          0.00
```

Copied!

As with other types of logical volumes, you can rename the volume with the `lvrename`, you can remove the volume with the `lvremove`, and you can display information about the volume with the `lvs` and `lvdisplay` commands.

By default, the `lvcreate` command sets the size of the thin pool's metadata logical volume according to the formula (Pool_LV_size / Pool_LV_chunk_size * 64). You cannot currently resize the metadata volume, however, so if you expect significant growth of the size of thin pool at a later time you should increase this value with the `--poolmetadatasize` parameter of the `lvcreate` command. The supported value for the thin pool's metadata logical volume is in the range between 2MiB and 16GiB.

You can use the `--thinpool` parameter of the `lvconvert` command to convert an existing logical volume to a thin pool volume. When you convert an existing logical volume to a thin pool volume, you must use the `--poolmetadata` parameter in conjunction with the `--thinpool` parameter of the `lvconvert` to convert an existing logical volume to the thin pool volume's metadata volume.

Converting a logical volume to a thin pool volume or a thin pool metadata volume destroys the content of the logical volume, since in this case the `lvconvert` does not preserve the content of the devices but instead overwrites the content.

The following example converts the existing logical volume `lv1` in volume group `vg001` to a thin pool volume and converts the existing logical volume `lv2` in volume group `vg001` to the metadata volume for that thin pool volume.

```
#
 lvconvert --thinpool vg001/lv1 --poolmetadata vg001/lv2

  Converted vg001/lv1 to thin pool.
```

Copied!