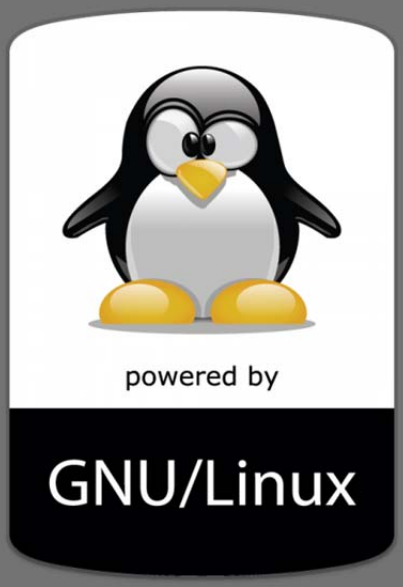
2015

# توسعه فضای ذخیره‌سازی دیسک با LVM

# (ویرایش سوم)

نویسنده حسام الدین توحید




skywan13@chmail.ir

skywan13@chmail.ir
**نویسنده : حسام الدین توحید**

## مقدمه مولف :

آنچه پیش رو دارید ویرایش **سوم** مقاله توسعه فضای ذخیره‌سازی دیسک با LVM است که به صورت رایگان و تحت لیسانس GNU GPLv3 به علاقه مندان لینوکس هدیه می گردد . در تهیه این مقاله از سر فصل های درسی گفته شده در دوره هایLPIc2 و RHCE استفاده شده ولازم می دانم از **مهندس مهدوی فر** و **مهندس جعفر عقربی** به خاطر راهنمایی های مفیدشان و همچنین **مرکز آموزشهای پیشرفته دانشگاه شریف (لایتک)** تشکر کافی را داشته باشم. این مطالب با نگاهی کاربردی و بدون پرداختن به بحث های تئوریک و بر اساس توزیع CentOS سری 6 گردآوری و عرضه شده که امید است این مطلب بتواند باعث ارتقاء دانش فنی کاربران لینوکس و متخصصین IT شود. زکات علم نشر آن است.

موفق باشید

**حسام الدین توحید**

اردیبهشت 1395

skywan13@chmail.ir

# فهرست


- مقدمه‌ای بر LVM 4
- مزایای LVM 6
- ساختمان LVM 6
- شروع به کار با LVM 7
- راه‌اندازی LVM در محیط CLI 9


skywan13@chmail.ir

# مقدمه‌ای بر LVM

LVM برگرفته از Logical Volume Management یک تکنیک اختصاص فضای هارد دیسک برای استفاده در مواردی است که به خاطر کمبود فضای ذخیره‌سازی تعداد هارد دیسک‌های یک سیستم افزایش پیدا می‌کند و می‌تواند قابلیت‌های بسیار زیادی در حوزه پارتیشن بندی هارد دیسک‌ها به ما بدهد. ازLVM می‌توان به عنوان یک لایه ظریف نرم افزاری در بالای چندین هارد دیسک و پارتیشن یاد کرد که این قابلیت را به admin می‌دهد تا بتواند به راحتی پارتیشن‌ها را بصورت متمرکز ، توزیع شده یا ترکیبی بر روی چندین هارد دیسک ایجاد، resize و مدیریت کرده و یک فایل سیستم را بصورت همزمان بر روی چندین پارتیشن در دسترس قرار دهد بدون اینکه کاربر متوجه تعویض هارد دیسک ویا هر تغییری شود.LVM امکان، ایجاد Snapshot Backup یا ترکیب چندین دیسک برای اجرا شدن تحت یک پارتیشن واحد و بسیاری دیگر از امکانات را فراهم می‌کند. و همچنین به شما امکاناتی را ارائه می دهد که در حالت عادی فایل سیستم قادر به انجام چنین کاری نمی‌باشد، برای مثال Ext3 نمی‌تواند از قابلیت‌های live snapshot استفاده کند اما با استفاده ازLVM شما می‌توانید از Logical Volume های خود بدون نیاز به اینکه Unmount شوند Backup یا Snapshot بگیرید. به طور کلی LVM انعطاف پذیری بهتری در مدیریت دیسک ها و پارتیشن ها ارائه می‌کند. بدون وجود LVM، تغییر اندازه یا resize کردن یک پارتیشن کاری دشوار است و ممکن است اطلاعات پارتیشن از بین برود و یا باعث Downtime و از دسترس خارج شدن سیستم شود ولی با استفاده از تکنولوژی به راحتی می‌توان این کار را انجام داد.

مهمترین نکته این است که در چنین شرایطی سیستم عامل و کاربر به هیچ عنوان متوجه این تغییرات نخواهند شد، لذا می توان براحتی با استفاده از تکنیک LVM فضای پارتیشن‌ها را کم یا زیاد کرد و یا آنها را به جای دیگری انتقال داد بدون اینکه کوچکترین تداخلی در کارکرد سیستم عامل پیش بیاید. از LVM به عنوان یک تکنیک مجازی سازی فضای ذخیره سازی یا **Storage Virtualization** نیز نام برده می‌شود. LVM در همان لایه ای کار می‌کند که درایورها کار می‌کنند بنابراین تا حدود زیادی درگیری با سیستم عامل ندارد و این امکان را به کرنل سیستم عامل می دهد که پارتیشن بندی هارد دیسک های موجود بر روی سیستم وابستگی به ساختار و لایه بندی دیسک های سخت‌افزاری روی سیستم نداشته باشند و بتوانند بصورت مستقل از دیسک عمل کنند.

با LVM دیسک ها و پارتیشن ها می توانند شامل دیسک ها و پارتیشن های متعددی در غالب یک دستگاه واحد باشند که در غالب مکانیزم‌های Volume Group (دیسک ها) وLogical Volume (پارتیشن ها) نشان داده شده و ماهیت اصلی آنها از دید سیستم عامل پنهان می‌ماند. چونکه Volume Group ها و Logical Volume ها وابسته به هارد دیسک‌ها نیستند و واقعا بصورت فیزیکی بر روی هارد درایو قرار نگرفته‌اند به راحتی می‌توان بدون کوچکترین وقفه در سیستم دیسک‌ها و پارتیشن‌ها را مدیریت کرد. ویژگی دیگر LVM که در سیستم فایلی مانند EXT3 وجود ندارد این است LVM قادر به ایجاد Snapshot Backup **از** Logical Volume بدون Unmountکردن سیستم فایل است.



skywan13@chmail.ir

# مزایای استفاده از LVM

**برای سیستم های کوچک :** زمانی که شما در سیستم خانگی خودتان با مشکل کم بودن فضا مواجه می‌شوید و برای مثال شاخه home شما پر می‌شود می توانید به راحتی یک هارد دیسک جدید تهیه کنید و فضای جدید را به راحتی به پارتیشن home خود اضافه کنید. بدون آنکه نیاز به نصب مجدد سیستم عامل داشته باشید.

**برای سیستم های بزرگ:** برای سیستم های بزرگ مدیریت دیسکها می تواند کار بسیار زمان بری باشد. با کمک LVM admin سیستم می‌تواند تنها زمانی که به فضای بیشتری نیاز داشت یک دیسک جدید به سیستم اضافه کرده و آن را به فضای قبلی اضافه کند.

اما به طور کلی مزایای زیر را می‌توان برای LVM برشمرد:

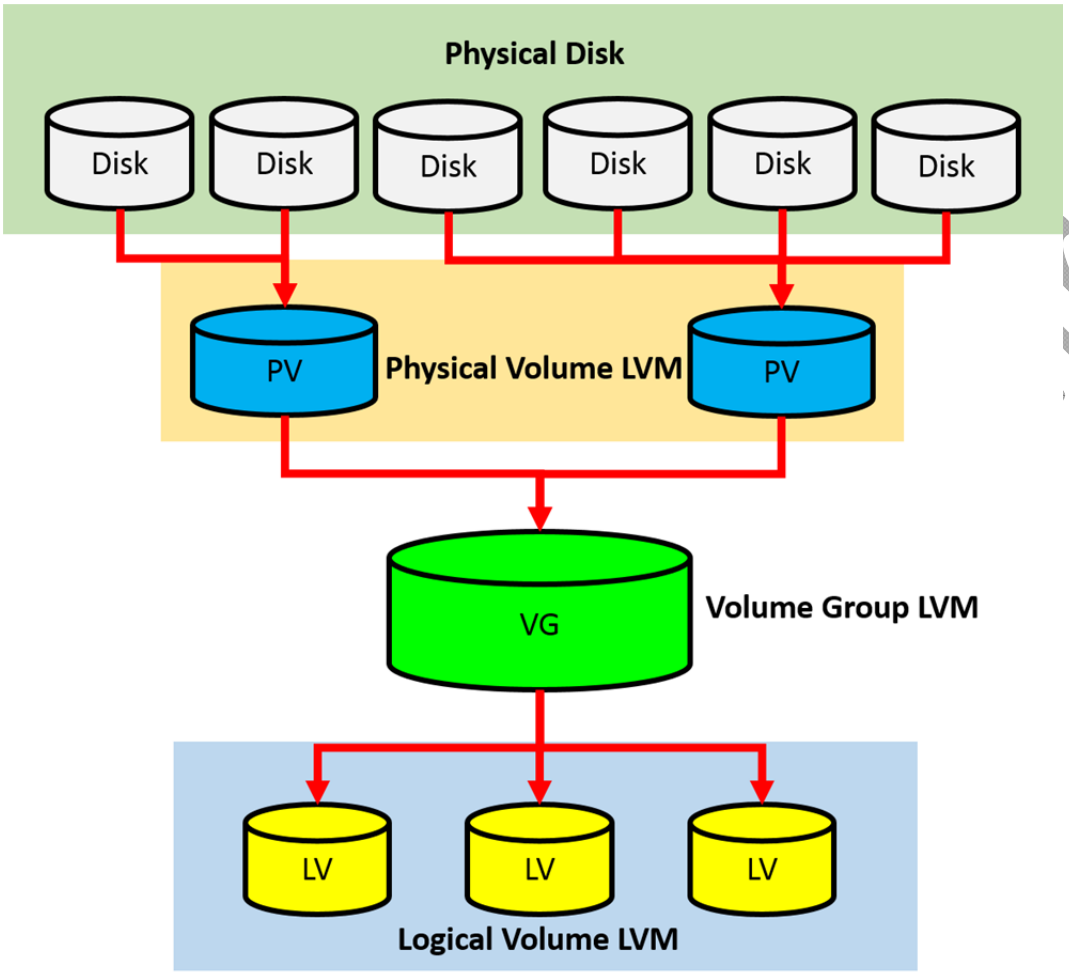
1. می‌توانید از هر تعداد هارد دیسکی که بر روی سیستم وجود دارد در قالب یک هارد دیسک استفاده کنید.
2. مدیریت دیسک‌ها از نظر سیستم عامل ساده تر می‌شود زیرا سیستم عامل نمی‌تواند لایه پایینی را مشاهده کند.
3. فضای یک Logical Volume می‌تواند بین چندین دیسک تقسیم شود.
4. می‌توانید Logical Volume هایی با ظرفیت کم ایجاد کنید و بعدها به فضای آن در صورت نیاز اضافه کنید.
5. می‌توانید فضای پارتیشن های خود را به سادگی تغییر دهید بدون اینکه کاربر و سیستم عامل درگیر شوند.
6. اضافه، حذف و جایگزین کردن دیسک های فیزیکی بسیار ساده تر می‌شود و سرویس‌دهی قطع نمی‌شود.
7. قابلیت استفاده از Backup با استفاده از Snapshot ها فراهم می‌شود.
8. انتقال آنلاین دیسک‌ها و محتویات آنها بدون Restart کردن سرویس ها امکانپذیر می‌شود.

# ساختمان LVM

در ساختار پارتیشن بندی و مدیریت دیسک‌های سنتی که در قدیم استفاده می‌شد ، سیستم عامل به دنبال هارد دیسک‌های نصب شده بر روی دستگاه می‌گشت و بعد از شناسایی کردن هارد دیسک‌ها به سراغ شناسایی پارتیشن‌ها و فضاهای موجود بر روی آنها می‌رفت. بنابراین اگر هارد دیسک اول را به سیستم معرفی می‌کردید طبیعتا به شکل /dev/sda/ و هارد دیسک دوم به شکل /dev/sdb/ و ..... به همین شکل شناسایی می‌شد و سیستم عامل بعد از شناسایی دیسک در داخل همان هارد به دنبال پارتیشن های /dev/sda1/ و /dev/sda2/ و ... می‌گشت اما با استفاده از LVM دیگر سیستم عامل به ساختار دیسک‌ها وابسته نیست و دیسک‌ها و پارتیشن‌ها می‌توانند شامل چندین دیسک و چندین پارتیشن باشند که در قالب یک پارتیشن یا فایل سیستم از یک دیسک دیده می‌شوند. اضافه کردن فضای یک هارد جدید به سیستم باید بدون downtime انجام گیرد. برای کار با LVM باید با بخشهای مختلف ساختمان آن آشنا شوید که در اینجا آنها را معرفی می‌کنم.



skywan13@chmail.ir



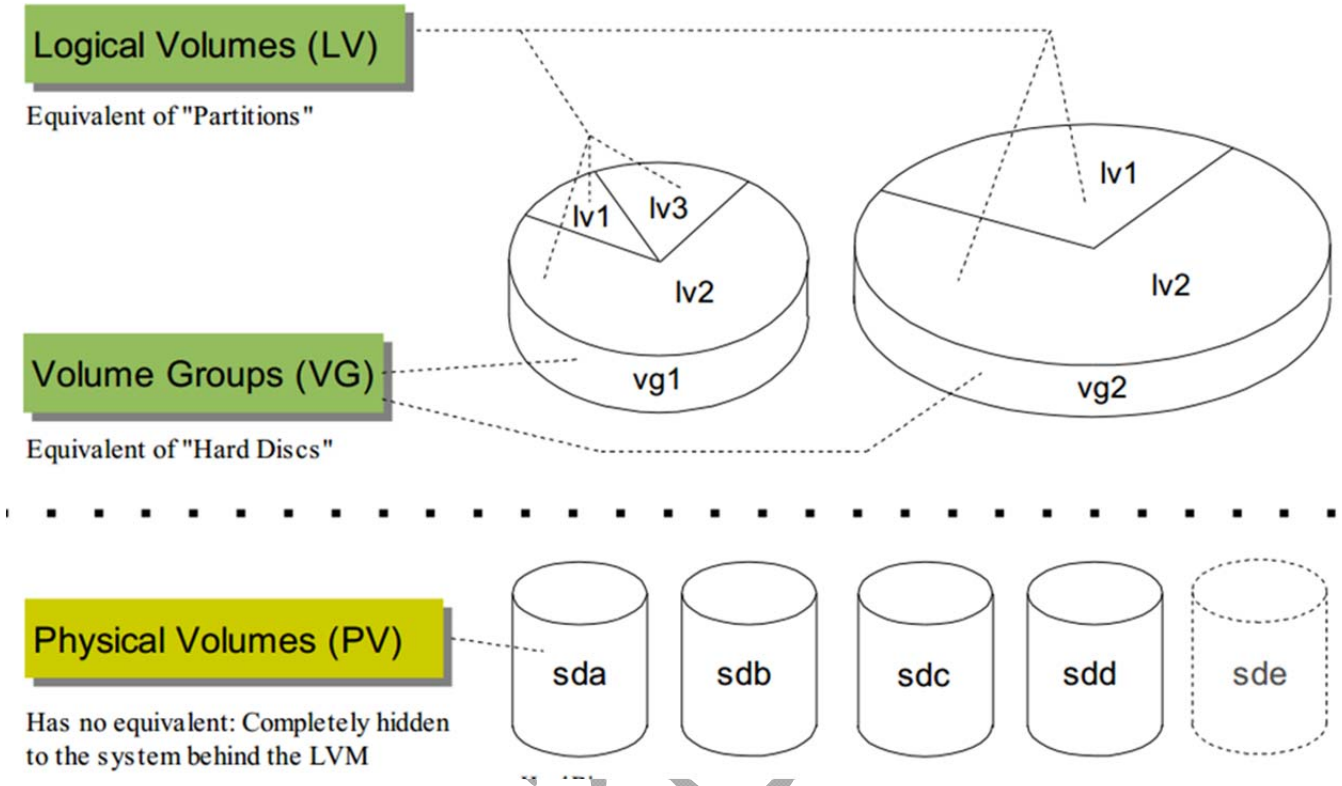

Physical Volumes برابر با دیسک‌هایی هستند مانند دیسک sda که فضای ذخیره سازی Logical Volume ها را فراهم می کنند. Logical Volume ها برابر با پارتیشن‌ها هستند که سیستم فایل بر روی آنها سوار می‌شود. کل یک دیسک می تواند به صورت یک پارتیشن باشد یا اینکه به چندین پارتیشن مجزا تقسیم شود.

هر Volume Group مجموعه ای از Physical Volumesها می‌باشد که در عموم سیستم‌ها تنها به یک Volume Group نیاز است که شامل تمامی Physical Volumes های موجود است.

**نکته:** دایرکتوری boot نباید عضو LVM باشد زیرا Bootloader نمی‌تواند آنرا بخواند پس اگر دایرکتوری ریشه، به صورت LVM بود، می بایست boot را در پارتیشنی جدا از دایرکتوری / و غیرLVM قرار بدهید.

Volume Group می‌تواند به چندین Logical Volume تقسیم شود که به نقاط اتصالی مانند دایرکتوری / و یا دایرکتوری home/ و غیره اختصاص داده شده می‌شوند. وقتی که فضای یک پارتیشن پر می‌شود، فضای اضافی را می‌توان از یک Volume Group به پارتیشن مربوطه اختصاص داد تا فضای پارتیشن افزایش یابد. زمانی که یک دیسک جدید به

سیستم اضافه می‌شود، می‌تواند به Volume Group اضافه شود لذا پارتیشن‌هایی که logical Volume هستند می‌توانند افزایش حجم یابند.




## Physical Volume (PV):

اولین لایه بعد از هارد دیسک‌های فیزیکی، PV است. به پارتیشن‌هایی که مستقیما بر روی هارد دیسک ایجاد می‌شوند و می‌توانید Volume Group ها را در آنها ایجاد کنید، PV گفته می‌شود. در واقع PVها اولین اطلاعات مدیریتی هستند که بر روی هارد دیسک فیزیکی نوشته می‌شوند تا سیستم قادر به شناسایی آنها باشد. برای درک بهتر فقط کافیست بدانید که PVها بلوک‌های بزرگی هستند که برای مدیریت اولیه هارد دیسک‌ها ایجاد می‌شوند و معمولا شامل همه فضای دیسک سخت می‌شوند. از PV به عنوان نگهدارنده PEها استفاده می‌شود.

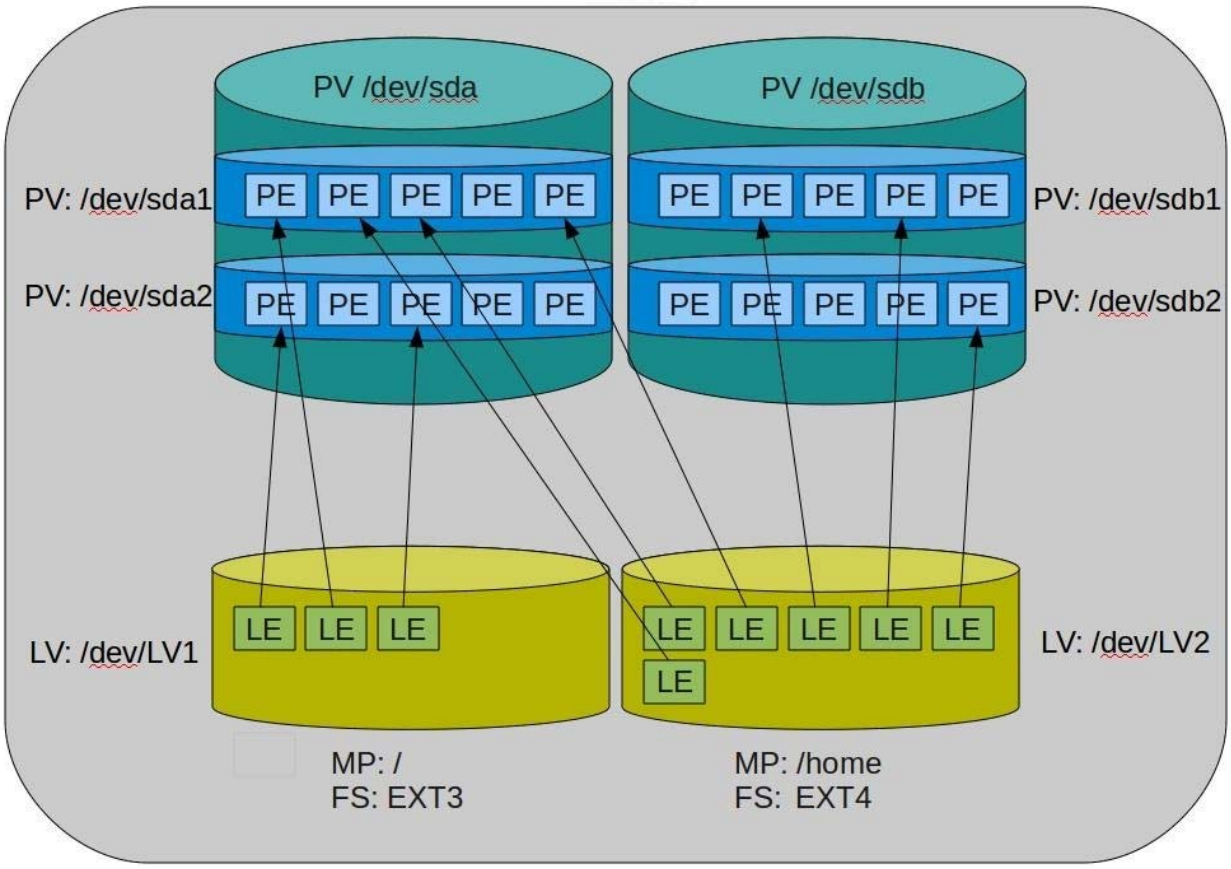
## Volume Group (VG):

VG بالاترین سطح ظاهری است که به وسیله LVM استفاده می‌شود. VG مجموعه‌ای از LV ها و PV ها را در یک واحد مدیریتی جمع می‌کند. یک VG شامل مجموعه‌ای از Physical Volume ها است که در قالب یک دیسک تمایش داده می‌شوند یا بهتر بگوییم در قالب یک Storage Volume نمایش داده می‌شوند.

skywan13@chmail.ir

**Logical Volume (LV):** این قسمت در واقع همان پارتیشن‌هایی هستند که سیستم می‌شناسد. یک Virtual یا Logical Partition که بر روی یک Volume Group قرار گرفته است می‌تواند شامل چندین Physical Extents باشد. در واقع LV نتیجه کاری LVM است که قرار است یوزرها از آن استفاده کنند. LVها محل ذخیره‌سازی داده‌های موجود در سیستم می‌باشند.

**Physical Extent (PE) :**
کوچکترین اندازه یا فضایی است که می‌توان از Physical Volume گرفت و به Logical Volume اختصاص داد PE می‌باشد. پیشفرض این فضا 4 مگابایت است و برای درک بهتر موضوع کافیست آن را قسمتی از دیسک فرض کنید که می توان به هر پارتیشنی اختصاص داد.



# شروع کار با LVM

اولین کاری که باید انجام دهیم Initialize کردن پارتیشن‌هاست. قبل از این کارها باید توسط دستوراتی مانند fdisk و یا دستورات مشابه هارد یا پارتیشن مورد نظر را پیکربندی کرده و آنها را جزء پارتیشن‌های LVM ای قرار داده باشیم. ساخت PV توسط دستور pvcreate انجام می گیرد. *این دستور یک توصیفگر VG در اول دیسک ایجاد می کند.*

## ساخت PV

**# pvcreate /dev/sda5**

## ساخت VG

خوب حالا می توانیم یک VG بسازیم.

**# vgcreate my_volume_group /dev/sda5**

my_volume_group : یک نام دلخواه است که باید به VG داده شود.

## اضافه کردن یک PV به VG

در صورتی که بخواهید یک PV دیگر را به VG اضافه کنید می‌توانید به شکل زیر عمل نمایید:

**# vgextend my_volume_group /dev/sdb6**

## ساخت LV

برای ساختن یک LV به ظرفیت ۱۰گیگ به صورت زیر عمل کنید:

**# lvcraete -L 10G my_volume_group –name my_logical_volume**

اگر بخواهید یک LV بسازید که تمام VG را در بر داشته باشد از vgdisplay استفاده کنید تا مجموع PE های موجود را ببینید سپس دستور lvcreate را اجرا کنید:

**# vgdisplay | grep “Total PE”**
**Total PE 3576**

در اینجا، ۳۵۷۶ عدد PE در این VG وجود دارد. برای ساخت LV که تمام این فضا را شامل شود از lvcreate به شکل زیر می‌توان استفاده کرد:

**# lvcreate -l 3576 my_volume_group –name my_logical_volume**

دقت کنید در اینجا از حرف کوچک **l** برای مقدار دهی استفاده کردیم.

## ساخت سیستم فایل

اکنون LV آماده است و شما می‌توانید با آن به صورت یک پارتیشن معمولی رفتار کرده و آن را فرمت کنید:

**# mkfs.ext3 /dev/my_volume_group/my_logical_volume**

سپس آن را mount و از آن استفاده نمایید. در صورتی که می خواهید در هنگام راه اندازی سیستم به صورت خودکار mount شود خط مربوط به آن را به فایل fstab اضافه کنید.

skywan13@chmail.ir

## توسعه یک LV

در صورتی که یکPV به VG اضافه کردید، یا در VG فعلی فضای خالی در اختیار دارید می‌توانید LV را توسعه دهید. برای توسعه LV به دو صورت می‌توان عمل کرد:

**#lvextend -L12G /dev/my_volume_group/my_logical_volume**

دستور بالا حجم LV را به G۱۲ افزایش می دهد.

**#lvextend -L+1G /dev/my_volume_group/my_logical_volume**

دستور بالا یک گیگابایت به my_logical_volume اضافه می کند.

بعد از آنکه LV را توسعه دادید شما باید سیستم فایل را به اندازه‌ایی که با آن مطابقت داشته باشد افزایش دهید. با کمک دستور resize2fs می‌توانید این کار را انجام دهید. بهتر است قبل از اجرای resize2fs سیستم فایل چک شود:

**#e2fsck -f /dev/my_volume_group/my_logical_volume**
**#resize2fs /dev/my_volume_group/my_logical_volume**

## پاک کردن LVM

برای پاک کردن LVM بر عکس مسیر ساخت عمل می‌کنیم. ابتدا باید LV پاک شود، قبل از هر کاری باید آن را umount کرد و سپس با کمک دستور زیر آن را remove کنید:

**#lvremove /dev/my_volume_group/my_logical_volume**

بعد از آن نوبت به پاک کردن VG می‌رسد:

**#vgremove my_volume_group**

و در آخر پاک کردن PV :

**#pvremove /dev/sdb1**

پیاده سازی LVM چندان هم ساده نیست و پیچیدگی‌هایی که در پیاده سازی این مفهوم وجود دارد باعث می‌شود استفاده از آن راحت نباشد و این به عنوان یک نقطه ضعف برای LVM در نظر گرفته می‌شود .

جهت درک بهتر مطالب گفته شده ، در ادامه ساخت LVM صورت کامندی و بر اساس چند سناریو کوچک آموزش داده می شود.

# راه اندازی LVM به کمک محیط Shell

در این سناریو ما سیستمی داریم که علاوه بر اینکه یک هارد دومی دارد مقداری از هارد اول آن استفاده شده و مقداری از آن باقی مانده است. ما می‌خواهیم ابتدا از باقی مانده آن یک پارتیشن لینوکسی بسازیم لذا ابتدا با دستور fdisk –l یک آمار از موجودی ظرفیت هارد دیسک می‌گیریم:



skywan13@chmail.ir

```
[root@localhost ~]# fdisk -l

Disk /dev/sda: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1          38      305203+  83  Linux
/dev/sda2              39         430     3148740   83  Linux
/dev/sda3             431         688     2072385   83  Linux
/dev/sda4             689        1305     4956052+   5  Extended
/dev/sda5             689         819     1052226   82  Linux swap / Solaris

Disk /dev/sdb: 16.1 GB, 16106127360 bytes
255 heads, 63 sectors/track, 1958 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
[root@localhost ~]# _
```

سپس با دستور fdisk وارد فضای پارتیشن بندی شده و یک پرینت از پارتیشن‌های هارد می‌بینیم.

```
[root@localhost ~]# fdisk /dev/sda

The number of cylinders for this disk is set to 1305.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): p

Disk /dev/sda: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1          38      305203+  83  Linux
/dev/sda2              39         430     3148740   83  Linux
/dev/sda3             431         688     2072385   83  Linux
/dev/sda4             689        1305     4956052+   5  Extended
/dev/sda5             689         819     1052226   82  Linux swap / Solaris

Command (m for help): _
```

در اینجا به ما اعلام می‌کند از سیلندر 820 تا 1305 هارد خالی است لذا ما اولین سیلندر را 821 قرار داده و در خط بعد ظرفیتی که می‌خواهیم پارتیشن مذکور داشته باشد را وارد می‌کنیم.

```
Command (m for help): n
First cylinder (820-1305, default 820): 821
Last cylinder or +size or +sizeM or +sizeK (821-1305, default 1305): +2G

Command (m for help): _
```

همانطور که در شکل زیر می‌بینید پارتیشن جدید با نام sda6 ساخته شد.

```
Command (m for help): p

Disk /dev/sda: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1          38      305203+  83  Linux
/dev/sda2              39         430     3148740   83  Linux
/dev/sda3             431         688     2072385   83  Linux
/dev/sda4             689        1305     4956052+   5  Extended
/dev/sda5             689         819     1052226   82  Linux swap / Solaris
/dev/sda6             821        1064     1959930   83  Linux

Command (m for help): _
```

پارتیشن جدید از نوع لینوکسی می‌باشد که باید آن را به یک پارتیشن LVM ای تبدیل کنیم.

```
Command (m for help): t
Partition number (1-6): 6
Hex code (type L to list codes): _
```

و سپس با تایپ دستور **l** لیست پارتیشنهایی که می‌تواند برای ما بسازد را مشاهده کرده و نوع دلخواه را انتخاب می‌کنیم.

skywan13@gmail.ir

```
 0  Empty           1e  Hidden W95 FAT1 80  Old Minix       bf  Solaris
 1  FAT12           24  NEC DOS         81  Minix / old Lin c1  DRDOS/sec (FAT-
 2  XENIX root      39  Plan 9          82  Linux swap / So c4  DRDOS/sec (FAT-
 3  XENIX usr       3c  PartitionMagic  83  Linux           c6  DRDOS/sec (FAT-
 4  FAT16 <32M      40  Venix 80286     84  OS/2 hidden C:  c7  Syrinx
 5  Extended        41  PPC PReP Boot   85  Linux extended  da  Non-FS data
 6  FAT16           42  SFS             86  NTFS volume set db  CP/M / CTOS / .
 7  HPFS/NTFS       4d  QNX4.x          87  NTFS volume set de  Dell Utility
 8  AIX             4e  QNX4.x 2nd part 88  Linux plaintext df  BootIt
 9  AIX bootable    4f  QNX4.x 3rd part 8e  Linux LVM       e1  DOS access
 a  OS/2 Boot Manag 50  OnTrack DM      93  Amoeba          e3  DOS R/O
 b  W95 FAT32       51  OnTrack DM6 Aux 94  Amoeba BBT      e4  SpeedStor
 c  W95 FAT32 (LBA) 52  CP/M            9f  BSD/OS          eb  BeOS fs
 e  W95 FAT16 (LBA) 53  OnTrack DM6 Aux a0  IBM Thinkpad hi ee  EFI GPT
 f  W95 Ext'd (LBA) 54  OnTrackDM6      a5  FreeBSD         ef  EFI (FAT-12/16/
10  OPUS            55  EZ-Drive        a6  OpenBSD         f0  Linux/PA-RISC b
11  Hidden FAT12    56  Golden Bow      a7  NeXTSTEP        f1  SpeedStor
12  Compaq diagnost 5c  Priam Edisk     a8  Darwin UFS      f4  SpeedStor
14  Hidden FAT16 <3 61  SpeedStor       a9  NetBSD          f2  DOS secondary
16  Hidden FAT16    63  GNU HURD or Sys ab  Darwin boot     fb  VMware VMFS
17  Hidden HPFS/NTF 64  Novell Netware  b7  BSDI fs         fc  VMware VMKCORE
18  AST SmartSleep  65  Novell Netware  b8  BSDI swap       fd  Linux raid auto
1b  Hidden W95 FAT3 70  DiskSecure Mult bb  Boot Wizard hid fe  LANstep
1c  Hidden W95 FAT3 75  PC/IX           be  Solaris boot    ff  BBT
Hex code (type L to list codes): _
```

با وارد کردن گزینه 8e نوع پارتیشن مزکور به LVM تغییر پیدا می کند.

```
Hex code (type L to list codes): 8e
Changed system type of partition 6 to 8e (Linux LVM)

Command (m for help): p

Disk /dev/sda: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1          38      305203+  83  Linux
/dev/sda2              39         430     3148740   83  Linux
/dev/sda3             431         688     2072385   83  Linux
/dev/sda4             689        1305     4956052+   5  Extended
/dev/sda5             689         819     1052226   82  Linux swap / Solaris
/dev/sda6             821        1064     1959930   8e  Linux LVM

Command (m for help): _
```

سپس جهت اعمال تغییرات از گزینه w تایپ کنید. اما همانطور که می‌بینید پیام سیستم این مفهوم را می‌رساند که باید سیستم ریست شود تا جدول پارتیشن سیستم بروز گردد. این کار در سرور امکان پذیر نیست و ممکن است باعث صدماتی شود .

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or resource
 busy.
The kernel still uses the old table.
The new table will be used at the next reboot.
Syncing disks.
[root@localhost ~]# _
```

لذا برای حل این گونه مشکلات شرکت ردهت دستور partpeobe را ارائه کرد. اگر بخواهیم کل جدول پارتیشن سیستم به روز شود از دستور اول استفاده و اگر بخواهیم فقط تغییرات صورت گرفته در جدول پارتیشن سیستم درج شود از دستور دوم استفاده می کنیم.

```
[root@localhost ~]# man partprobe
[root@localhost ~]# partprobe /dev/sda
[root@localhost ~]# partprobe /dev/sda6
[root@localhost ~]#
```

همانطور که می بینید یک sdb هم وجود دارد که جهت اجرای سناریو باید آن را هم پارتیشن‌بندی کنیم.

```
[root@localhost ~]# fdisk -l

Disk /dev/sda: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1          38      305203+  83  Linux
/dev/sda2              39         430     3148740   83  Linux
/dev/sda3             431         688     2072385   83  Linux
/dev/sda4             689        1305     4956052+   5  Extended
/dev/sda5             689         819     1052226   82  Linux swap / Solaris
/dev/sda6             821        1064     1959930   8e  Linux LVM

Disk /dev/sdb: 16.1 GB, 16106127360 bytes
255 heads, 63 sectors/track, 1958 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
```

```
[root@localhost ~]# fdisk /dev/sdb

The number of cylinders for this disk is set to 1958.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-1958, default 1):
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-1958, default 1958): +2G

Command (m for help): p

Disk /dev/sdb: 16.1 GB, 16106127360 bytes
255 heads, 63 sectors/track, 1958 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1               1         244     1959898+  83  Linux

Command (m for help):
```

همانطور که میبینید پارتیشن ساخته شده لینوکسی است و باید به LVM تغییر type پیدا کند.

```
Command (m for help): t
Selected partition 1
Hex code (type L to list codes):
```

در اینجا کد تغییر type را وارد کرده تغییرات را write می کنیم.

skywan13@

```
Hex code (type L to list codes): 8e
Changed system type of partition 1 to 8e (Linux LVM)

Command (m for help): p

Disk /dev/sdb: 16.1 GB, 16106127360 bytes
255 heads, 63 sectors/track, 1958 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1               1         244     1959898+  8e  Linux LVM

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
[root@localhost ~]#
```

در انتها جدول پارتیشن را بروز کرده و دوباره از پارتیشن های موجود جهت اطمینان از تغییر لیست می گیریم.

```
[root@localhost ~]# partprobe /dev/sdb1
[root@localhost ~]# fdisk -l

Disk /dev/sda: 10.7 GB, 10737418240 bytes
255 heads, 63 sectors/track, 1305 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1          38      305203+  83  Linux
/dev/sda2              39         430     3148740   83  Linux
/dev/sda3             431         688     2072385   83  Linux
/dev/sda4             689        1305     4956052+   5  Extended
/dev/sda5             689         819     1052226   82  Linux swap / Solaris
/dev/sda6             821        1064     1959930   8e  Linux LVM

Disk /dev/sdb: 16.1 GB, 16106127360 bytes
255 heads, 63 sectors/track, 1958 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1               1         244     1959898+  8e  Linux LVM
```

تا الان ما فقط دو پارتیشن با نوع lvm ساخته ایم. برای ادامه کار باید با آنها یک PV تشکیل دهیم. لازم به ذکر است این *پارتیشن ها بر روی هارد های جداگانه قرار دارد.*

```
[root@localhost ~]# pvdisplay
  /dev/hdc: open failed: No medium found
[root@localhost ~]# pvcreate /dev/sda6 /dev/sdb1
  Physical volume "/dev/sda6" successfully created
  Physical volume "/dev/sdb1" successfully created
[root@localhost ~]#
```

بعد از ساخت PV، جهت اطمینان از صحت ساخت با دستور pvdisplay یک آمار از سیستم می گیریم :

```
[root@localhost ~]# pvdisplay
  /dev/hdc: open failed: No medium found
  "/dev/sda6" is a new physical volume of "1.87 GB"
  --- NEW Physical volume ---
  PV Name               /dev/sda6
  VG Name
  PV Size               1.87 GB
  Allocatable           NO
  PE Size (KByte)       0
  Total PE              0
  Free PE               0
  Allocated PE          0
  PV UUID               kBFkGt-seno-jfVU-5hF6-yXXb-a3IN-2aTT3c

  "/dev/sdb1" is a new physical volume of "1.87 GB"
  --- NEW Physical volume ---
  PV Name               /dev/sdb1
  VG Name
  PV Size               1.87 GB
  Allocatable           NO
  PE Size (KByte)       0
  Total PE              0
  Free PE               0
  Allocated PE          0
  PV UUID               1ES0eL-h3LS-YWZo-xlYU-PHct-W2DU-c7ogPC
```

حال باید PV های درست شده را در یک گروه قرار دهیم که به این گروه VG گفته می‌شود. یک VG مجموع PV هاست

```
[root@localhost ~]# vgdisplay
  /dev/hdc: open failed: No medium found
[root@localhost ~]# vgcreate testlvm /dev/sda6 /dev/sdb1
  /dev/hdc: open failed: No medium found
  Volume group "testlvm" successfully created
[root@localhost ~]# echo $?
0
[root@localhost ~]#
```

سپس از محتویات VG یک لیست میگیریم:

```
[root@localhost ~]# vgdisplay
  /dev/hdc: open failed: No medium found
  --- Volume group ---
  VG Name               testlvm
  System ID
  Format                lvm2
  Metadata Areas        2
  Metadata Sequence No  1
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                0
  Open LV               0
  Max PV                0
  Cur PV                2
  Act PV                2
  VG Size               3.73 GB
  PE Size               4.00 MB
  Total PE              956
  Alloc PE / Size       0 / 0
  Free  PE / Size       956 / 3.73 GB
  VG UUID               ysD7D2-lgaK-T51R-bPH6-geC3-D6ax-nCzLoY
```

بعد از این مرحله باید از VG یک LV بسازیم که یا به یک دارکتوری ما اضافه شود و یا خودش یک دایرکتوری مجزا باشد.

L- : با این آپشن حجم Lv را مشخص می کنیم.

testlvm : نام VG را وارد می کنیم.(این نام فرضی بوده و هر چیزی می تواند باشد)

-n datalvm : با این آپشن نام LV را مشخص می کنیم.

```
[root@localhost ~]# lvcreate -L 3G testlvm -n datalvm
  /dev/hdc: open failed: No medium found
  Logical volume "datalvm" created
```

sky wan13@c

```
[root@localhost ~]# lvdisplay
  /dev/hdc: open failed: No medium found
  --- Logical volume ---
  LV Name                /dev/testlvm/datalvm
  VG Name                testlvm
  LV UUID                4FpK8m-43TG-x48k-wi2d-d2M5-yWlm-hXRRLZ
  LV Write Access        read/write
  LV Status              available
  # open                 0
  LV Size                3.00 GB
  Current LE             768
  Segments               2
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     256
  Block device           253:0
```

و با این دستور هم لیست LV ها را مشاهده می کنیم.

```
[root@localhost ~]# lvdisplay
  /dev/hdc: open failed: No medium found
  --- Logical volume ---
  LV Name                /dev/testlvm/datalvm
  VG Name                testlvm
  LV UUID                4FpK8m-43TG-x48k-wi2d-d2M5-yWlm-hXRRLZ
  LV Write Access        read/write
  LV Status              available
  # open                 0
  LV Size                3.00 GB
  Current LE             768
  Segments               2
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     256
  Block device           253:0
```

LV ای که ساخته‌ایم هنوز خام است و باید برای آن فایل سیستم انتخاب کرده و آن را فورمت کنیم. برای این کار از دستور زیر بهره می‌بریم.

```
[root@localhost ~]# mkfs.ext3 /dev/testlvm/datalvm
mke2fs 1.39 (29-May-2006)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
393216 inodes, 786432 blocks
39321 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=805306368
24 block groups
32768 blocks per group, 32768 fragments per group
16384 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912

Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

در انتها هم باید این فضای ساخته شده را mount کنیم.

```
[root@localhost ~]# mount /dev/testlvm/datalvm /DATA/
[root@localhost ~]# df -ah
Filesystem            Size  Used Avail Use% Mounted on
/dev/sda3             2.0G  874M  986M  47% /
proc                     0     0     0    -  /proc
sysfs                    0     0     0    -  /sys
devpts                   0     0     0    -  /dev/pts
/dev/sda2             3.0G   69M  2.7G   3% /home
/dev/sda1             289M   16M  258M   6% /boot
tmpfs                 252M     0  252M   0% /dev/shm
none                     0     0     0    -  /proc/sys/fs/binfmt_misc
sunrpc                   0     0     0    -  /var/lib/nfs/rpc_pipefs
/dev/mapper/testlvm-datalvm
                      3.0G   69M  2.8G   3% /DATA
```

حال فرض کنید که هارد سیستم پر شده و می خواهیم یک PV را به VG موجود اضافه کنیم تا مشکل کمبود ظرفیت بر طرف شود. لذا از هاردی که قبلا به سیستم اضافه شده یک مقداری را طبق روش زیر جدا می کنیم:

```
[root@localhost DATA]# fdisk /dev/sdb

The number of cylinders for this disk is set to 1958.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 2
First cylinder (245-1958, default 245):
Using default value 245
Last cylinder or +size or +sizeM or +sizeK (245-1958, default 1958): +5
G
Command (m for help): t
Partition number (1-4): 8e
Value out of range.
Partition number (1-4): 2
Hex code (type L to list codes): 8e
Changed system type of partition 2 to 8e (Linux LVM)

Command (m for help): p

Disk /dev/sdb: 16.1 GB, 16106127360 bytes
255 heads, 63 sectors/track, 1958 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1               1         244     1959898+  8e  Linux LVM
/dev/sdb2             245         853     4891792+  8e  Linux LVM

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
```

سپس برای اعمال تغییرات سیستم عامل را مجبور می کنیم تا جدول پارتیشن های سیستم را بخواند:

```
[root@localhost DATA]# partprobe /dev/sdb
[root@localhost DATA]# partprobe /dev/sdb2
```

حالا باید با دستور pvcreate از پارتیشن ساخته شده یک PV بسازیم.

```
[root@localhost DATA]# pvcreate /dev/sdb2
  Physical volume "/dev/sdb2" successfully created
```

و در این قسمت باید PV ساخته شده را به VG موجود اضافه کنیم .

```
[root@localhost DATA]# vgextend testlvm /dev/sdb2
  /dev/hdc: open failed: No medium found
  Volume group "testlvm" successfully extended
[root@localhost DATA]#
[root@localhost DATA]# vgdisplay
  /dev/hdc: open failed: No medium found
  --- Volume group ---
  VG Name               testlvm
  System ID
  Format                lvm2
  Metadata Areas        3
  Metadata Sequence No  3
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                1
  Open LV               1
  Max PV                0
  Cur PV                3
  Act PV                3
  VG Size               8.40 GB
  PE Size               4.00 MB
  Total PE              2150
  Alloc PE / Size       768 / 3.00 GB
  Free  PE / Size       1382 / 5.40 GB
```

و برای جدا کردن یک PV از VG طبق روش زیر عمل کنید:

```
[root@localhost DATA]# vgreduce testlvm /dev/sdb2
  /dev/hdc: open failed: No medium found
  Removed "/dev/sdb2" from volume group "testlvm"
[root@localhost DATA]#
[root@localhost DATA]# vgdisplay
  /dev/hdc: open failed: No medium found
  --- Volume group ---
  VG Name               testlvm
  System ID
  Format                lvm2
  Metadata Areas        2
  Metadata Sequence No  4
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                1
  Open LV               1
  Max PV                0
  Cur PV                2
  Act PV                2
  VG Size               3.73 GB
  PE Size               4.00 MB
  Total PE              956
  Alloc PE / Size       768 / 3.00 GB
  Free  PE / Size       188 / 752.00 MB
  VG UUID               ysD7D2-lgaK-T51R-
```

حال فرض کنید حجم یک پارتیشن کم است و ما می‌خواهیم یک PV به VG اضافه کرده تا بتوانیم حجم LV را افزایش دهیم.

```
[root@localhost DATA]# vgextend testlvm /dev/sdb2
  /dev/hdc: open failed: No medium found
  Volume group "testlvm" successfully extended
[root@localhost DATA]# vgdisplay
  /dev/hdc: open failed: No medium found
  --- Volume group ---
  VG Name               testlvm
  System ID
  Format                lvm2
  Metadata Areas        3
  Metadata Sequence No  5
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                1
  Open LV               1
  Max PV                0
  Cur PV                3
  Act PV                3
  VG Size               8.40 GB
  PE Size               4.00 MB
  Total PE              2150
  Alloc PE / Size       768 / 3.00 GB
  Free  PE / Size       1382 / 5.40 GB
```

ما می خواهیم 5گیگ به حجم LV اضافه شود

```
[root@localhost DATA]# lvdisplay
  /dev/hdc: open failed: No medium found
  --- Logical volume ---
  LV Name                /dev/testlvm/datalvm
  VG Name                testlvm
  LV UUID                4FpK8m-43TG-x48k-wi2d-d2M5-yWlm-hXRRLZ
  LV Write Access        read/write
  LV Status              available
  # open                 1
  LV Size                3.00 GB
  Current LE             768
  Segments               2
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     256
  Block device           253:0
```

دستور **lvresize** هم می‌تواند حجم را کم کند و هم می‌تواند آن را افزایش دهد. در اینجا عدد 8 بدین معنا است که ظرفیت LV به 8 گیگ برسد. اگر 5G+ هم قرار می‌دادیم همین کار را انجام می‌داد. آدرسی که در اینجا وارد می‌کنیم آدرس LV است که قبلا ایجاد کرده‌ایم. در اینجا بدون مشکل خاصی عمل resize با موفقیت انجام شد ولی ممکن است در بعضی مواقع پیغام Busy با ما نشان داده شود. علت آن هم این است که دایرکتوری مورد نظر در حال ارائه سرویس به کاربران و یا سیستم می‌باشد که در چنین مواقعی باید ابتدا دایرکتوری مورد نظر را umount کرده و بعد آن را resize کنیم.

```
[root@localhost DATA]# lvresize -L 8G /dev/mapper/testlvm-datalvm
  /dev/hdc: open failed: No medium found
  Extending logical volume datalvm to 8.00 GB
  Logical volume datalvm successfully resized
[root@localhost DATA]#
```

ما عمل resize را انجام دادیم ولی همانطور که می بینید افزایش حجمی به ما نشان نمی‌دهد.

```
[root@localhost DATA]# df -ah
Filesystem            Size  Used Avail Use% Mounted on
/dev/sda3             2.0G  874M  986M  47% /
proc                     0     0     0   -  /proc
sysfs                    0     0     0   -  /sys
devpts                   0     0     0   -  /dev/pts
/dev/sda2             3.0G   69M  2.7G   3% /home
/dev/sda1             289M   16M  258M   6% /boot
tmpfs                 252M     0  252M   0% /dev/shm
none                     0     0     0   -  /proc/sys/fs/binfmt_misc
sunrpc                   0     0     0   -  /var/lib/nfs/rpc_pipefs
/dev/mapper/testlvm-datalvm
                      3.0G   69M  2.8G   3% /DATA
[root@localhost DATA]#
```

چون ما قبلا روی LV مورد نظر یک فایل سیستم ایجاد کرده ایم که خود آن بلاک وآینود درست می کند. حجم با موفقیت تغییر کرده ولی مقدار صحیح را به ما نشان نمی‌دهد ، ما باید به طریقی به آن بفهمانیم که سایز تغییر کرده است. لذا از دستور زیر برای قابل استفاده شدن فضای مورد نظر استفاده می کنیم :

**-p** : این گزینه درصد resize را به ما نشان می دهد.

**-f** : این گزینه هم به صورت force کار درخواستی را انجام می دهد.

```
[root@localhost DATA]# resize2fs -p -f /dev/mapper/testlvm-datalvm
resize2fs 1.39 (29-May-2006)
Filesystem at /dev/mapper/testlvm-datalvm is mounted on /DATA; on-line
resizing required
Performing an on-line resize of /dev/mapper/testlvm-datalvm to 2097152
(4k) blocks.
The filesystem on /dev/mapper/testlvm-datalvm is now 2097152 blocks lon
g.
```

با وارد کردن دستور df -ah می توانید تغییر حجم را مشاهده کنید.

```
[root@localhost DATA]# df -ah
Filesystem            Size  Used Avail Use% Mounted on
/dev/sda3             2.0G  874M  986M  47% /
proc                     0     0     0   -  /proc
sysfs                    0     0     0   -  /sys
devpts                   0     0     0   -  /dev/pts
/dev/sda2             3.0G   69M  2.7G   3% /home
/dev/sda1             289M   16M  258M   6% /boot
tmpfs                 252M     0  252M   0% /dev/shm
none                     0     0     0   -  /proc/sys/fs/binfmt_misc
sunrpc                   0     0     0   -  /var/lib/nfs/rpc_pipefs
/dev/mapper/testlvm-datalvm
                      7.9G   71M  7.5G   1% /DATA
[root@localhost DATA]#
```

امید است با اجرای این سناریو توانسته باشم مفاهیم LVM را به طور ساده بیان کنم.

skywan13@chmail.ir