



实践教程





【版权声明】

©2013-2025 腾讯云版权所有

本文档(含所有文字、数据、图片等内容)完整的著作权归腾讯云计算(北京)有限责任公司单独所有,未经腾讯云 事先明确书面许可,任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成 对腾讯云著作权的侵犯,腾讯云将依法采取措施追究法律责任。

【商标声明】

🕗 腾讯云

及其它腾讯云服务相关的商标均为腾讯云计算(北京)有限责任公司及其关联公司所有。本文档涉及的第三方主体的 商标,依法由权利人所有。未经腾讯云及有关权利人书面许可,任何主体不得以任何方式对前述商标进行使用、复 制、修改、传播、抄录等行为,否则将构成对腾讯云及有关权利人商标权的侵犯,腾讯云将依法采取措施追究法律责 任。

【服务声明】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况,部分产品、服务的内容可能不时有所调整。 您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定,除非双方另有约定,否则, 腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【联系我们】

我们致力于为您提供个性化的售前购买咨询服务,及相应的技术售后服务,任何问题请联系 4009100100或 95716。



云硬盘

文档目录

实践教程

如何衡量云硬盘的性能 多块弹性云硬盘构建 LVM 逻辑卷 MBR 分区云硬盘扩容至大于 2TB

实践教程 如何衡量云硬盘的性能

最近更新时间: 2025-02-20 10:30:03

操作前须知(重要)

- ☆ 警告:
 - 本文使用 FIO 作为测试工具,请不要在系统盘上进行 FIO 测试,避免损坏系统重要文件。
 - 为避免底层文件系统元数据损坏导致数据损坏,请不要在业务数据盘上进行测试。请使用测试机器未存 放业务数据的云硬盘进行压测,并提前创建快照保障您的数据安全。
 - 请确保 /etc/fstab 文件配置项中没有被测硬盘的挂载配置,否则将导致云服务器启动失败。

衡量指标

腾讯云提供的块存储设备根据类型的不同拥有不同的性能和价格,详细信息请参见 <mark>云硬盘类型</mark> 。由于不同应用程序 的工作负载不同,若未提供足够的 I/O 请求来充分利用云硬盘时,可能无法达到云硬盘的最大性能。 一般使用以下指标衡量云硬盘的性能:

- IOPS:每秒读/写次数,单位为次(计数)。存储设备的底层驱动类型决定了不同的 IOPS。
- 吞吐量:每秒的读写数据量,单位为MB/s。
- 时延: I/O 操作的发送时间到接收确认所经过的时间,单位为微秒。

测试工具

FIO 是测试磁盘性能的工具,用来对硬件进行压力测试和验证,本文以 FIO 为例。 使用 FIO 时,建议配合使用 libaio 的 I/O 引擎进行测试。请参见 工具安装 完成 FIO 和 libaio 的安装。

测试对象建议

- 建议在空闲的、未保存重要数据的硬盘上进行 FIO 测试,并在测试完后重新制作被测硬盘的文件系统。
- 测试硬盘性能时,建议直接测试裸数据盘(如 /dev/vdb)。
- 测试文件系统性能时,推荐指定具体文件测试(如 /data/file)。

工具安装

- 参见使用标准登录方式登录 Linux 实例(推荐)登录云服务器,本文以 CentOS 7.6 操作系统的云服务器为例。
- 2. 执行以下命令,查看云硬盘是否4KiB对齐。



fdisk -lu

如下图所示,若返回结果中的 Start 值能被8整除即是4KiB对齐。否则请完成4KiB对齐后再进行测试。

Device Boot	Start	End	Blocks	Id	System
/dev/vdb1	2048	20971519	10484736	83	Linux

3. 依次执行以下命令,安装测试工具 FIO 和 libaio。



安装完成后,请参见测试示例开始云硬盘性能测试。

测试示例

不同场景的测试公式基本一致,只有 rw、iodepth 和 bs(block size)三个参数的区别。例如,每个工作负载适 合最佳 iodepth 不同,取决于您的特定应用程序对于 IOPS 和延迟的敏感程度。 参数说明:

参数名	说明	取值样 例
bs	每次请求的块大小。取值包括4k、8k及16k等。	4k
ioengin e	I/O 引擎。推荐使用 Linux 的异步 I/O 引擎。	libaio
iodepth	请求的 I/O 队列深度。	1
direct	指定 direct 模式。 • True(1)表示指定 O_DIRECT 标识符,忽略 I/O 缓存,数据直写。 • False(0)表示不指定 O_DIRECT 标识符。 默认为 True(1)。	1
read	读写模式。取值包括顺序读(read)、顺序写(write)、随机读 (randread)、随机写(randwrite)、混合随机读写(randrw)和混合顺序 读写(rw,readwrite)。	read



time_b ased	指定采用时间模式。无需设置该参数值,只要 FIO 基于时间来运行。	N/A
runtime	指定测试时长,即 FIO 运行时长。	100s
refill_b uffers	FIO 将在每次提交时重新填充 I/O 缓冲区。默认设置是仅在初始时填充并重用该数 据。	N/A
norand ommap	在进行随机 I/O 时,FIO 将覆盖文件的每个块。若给出此参数,则将选择新的偏移 量而不查看 I/O 历史记录。	N/A
randre peat	随机序列是否可重复,True(1)表示随机序列可重复,False(0)表示随机序 列不可重复。默认为 True(1)。	0
group_ reporti ng	多个 job 并发时,打印整个 group 的统计值。	N/A
name	job 的名称。	fio− read
size	I/O 测试的寻址空间。	1G
filenam e	测试对象,即待测试的磁盘设备名称。	/dev/s db
numjob s	并发线程数,默认为1。当被测试硬盘性能较高时推荐加大numjobs数(如2或4 等)以增加压力。	1

常见用例如下:

bs = 4k iodepth = 1: 随机读/写测试,能反映硬盘的时延性能

△ 警告:

- 请不要在系统盘上进行 FIO 测试,避免损坏系统重要文件。
- 为避免底层文件系统元数据损坏导致数据损坏,请不要在业务数据盘上进行测试。请使用测试
 机器未存放业务数据的云硬盘进行压测,并提前创建快照保障您的数据安全。

▲ 注意:

执行命令前请务必做好快照。

执行以下命令,测试硬盘的随机写时延(此处测试对象是名为 /dev/vdb 的设备)。



fio -bs=4k -ioengine=libaio -iodepth=1 -numjobs=1 -direct=1 rw=randwrite -time_based -runtime=600 -refill_buffers -norandommap
-randrepeat=0 -group_reporting -name=fio-randwrite-lat -size=10G filename=/dev/vdb

执行以下命令,测试硬盘的随机读时延(此处测试对象是名为 /dev/vdb 的设备)。

fio -bs=4k -ioengine=libaio -iodepth=1 -numjobs=1 -direct=1 rw=randread -time_based -runtime=600 -refill_buffers -norandommap
-randrepeat=0 -group_reporting -name=fio-randread-lat -size=10G filename=/dev/vdb

执行以下命令,测试硬盘的随机混合读写时延(此处测试对象是名为 /dev/vdb 的设备)。

fio -bs=4k -ioengine=libaio -iodepth=1 -numjobs=1 -direct=1 rw=randrw -time_based -runtime=600 -refill_buffers -norandommap randrepeat=0 -group_reporting -name=fio-randread-lat -size=10G filename=/dev/vdb

示例:

登录云服务器后,执行以下命令,测试设备名为 /dev/vdc 的 SSD 云硬盘的随机混合读写时延性能。

fio -bs=4k -ioengine=libaio -iodepth=1 -numjobs=1 -direct=1 rw=randrw -time_based -runtime=100 -refill_buffers -norandommap randrepeat=0 -group_reporting -name=fio-randrw-lat -size=1G filename=/dev/vdc

测试结果如下图所示:

fio-read: (g=0): rw=randrw, bs=(R) 4096B-4096B, (W) 4096B-4096B, (T) 4096B-4096B, ioengine=libaio, iodepth
Starting 1 process
Jobs: 1 (f=1): [m(1)][100.0%][r=3411KiB/s,w=3603KiB/s][r=852,w=900 IOPS][eta 00m:00s]
fio-read: (groupid=0, jobs=1): err= 0: pid=2377: Thu Jun 13 18:23:47 2019
read: IOPS=880, BW=3523KiB/s (3607kB/s)(344MiB/100001msec)
slat (nsec): min=2905, max=62479, avg=5254.61, stdev=2075.46
clat (usec): min=205, max=6921, avg=463.65, stdev=259.48
lat (usec): min=209, max=6925, avg=469.13, stdev=259.56
clat percentiles (usec):
1.00th=[245], 5.00th=[269], 10.00th=[293], 20.00th=[375],
30.00th=[400], 40.00th=[416], 50.00th=[437], 60.00th=[457],
70.00th=[478], 80.00th=[498], 90.00th=[545], 95.00th=[619],
99.00th=[2057], 99.50th=[2376], 99.90th=[3294], 99.95th=[4015],
99.99th=[6259]
bw (KiB/s): min= 2168, max= 4024, per=100.00%, avg=3522.64, stdev=310.95, samples=200
iops : min= 542, max= 1006, avg=880.66, stdev=77.74, samples=200
write: IOPS=877, BW=3511KiB/s (3595kB/s)(343MiB/100001msec)
slat (nsec): min=2981, max=58808, avg=5377.71, stdev=2079.36
clat (usec): min=421, max=10492, avg=659.10, stdev=219.96
lat (usec): min=428, max=10496, avg=664.70, stdev=220.05
clat percentiles (usec):
1.00th=[490], 5.00th=[523], 10.00th=[545], 20.00th=[562],
30.00th=[578], 40.00th=[594], 50.00th=[611], 60.00th=[635],
70.00 th= $\begin{bmatrix} 660 \end{bmatrix}$, 80.00 th= $\begin{bmatrix} 693 \end{bmatrix}$, 90.00 th= $\begin{bmatrix} 783 \end{bmatrix}$, 95.00 th= $\begin{bmatrix} 914 \end{bmatrix}$,
99.00th=[1516], 99.50th=[1926], 99.90th=[3261], 99.95th=[3982],
99.99th=[5342]
bw (KiB/s): min= 2296, max= 4008, per=100.00%, avg=3510.84, stdev=305.46, samples=200
iops : min= 574, max= 1002, avg=877.71, stdev=76.36, samples=200
lat (usec) : 250=0.76%, 500=40.51%, 750=51.04%, 1000=5.03%
lat (msec) : 2=1.90%, 4=0.71%, 10=0.05%, 20=0.01%
cpu : usr=0.50%, sys=1.52%, ctx=175841, maif=0, minf=29
IO depths : 1=100.0%, 2=0.0%, 4=0.0%, 8=0.0%, 16=0.0%, 32=0.0%, >=64=0.0%
submit : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
complete : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%

bs = 128k iodepth = 32: 顺序读/写测试,能反映硬盘的吞吐性能

☆ 警告:

- 请不要在系统盘上进行 FIO 测试,避免损坏系统重要文件。
- 为避免底层文件系统元数据损坏导致数据损坏,请不要在业务数据盘上进行测试。请使用测试 机器未存放业务数据的云硬盘进行压测,并提前创建快照保障您的数据安全。

▲ 注意:

执行命令前请务必做好快照。

执行以下命令,测试硬盘的**顺序读吞吐带宽**(此处测试对象是名为 /dev/vdb 的设备)。

fio -bs=128k -ioengine=libaio -iodepth=32 -numjobs=1 -direct=1 rw=read -time_based -runtime=600 -refill_buffers -norandommap -



```
filename=/dev/vdb
```

示例:

登录云服务器后,执行以下命令,测试设备名为 /dev/vdc 的 SSD 云硬盘的顺序读吞吐性能。

fio -bs=128k -ioengine=libaio -iodepth=32 -numjobs=1 -direct=1 rw=read -time_based -runtime=600 -refill_buffers -norandommap randrepeat=0 -group_reporting -name=fio-read-throughput -size=10G
-filename=/dev/vdc

测试结果如下图所示:

fio-rw: (g=0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=32
fio-3.1
Starting 1 process
Jobs: 1 (f=1): [W(1)][100.0%][r=0KiB/s,w=260MiB/s][r=0,w=2082 IOPS][eta 00m:00s]
fio-rw: (groupid=0,_jobs=1):_err= 0: pid=2679: Thu Jun 13 18:27:32 2019
write: IOPS=2081, BW=260MiB/s (273MB/s) (25.4GiB/100045msec)
slat (nsec): min=2847, max=72524, avg=7739.21, stdev=3233.07
clat (usec): min=1033, max=250494, avg=15341.09, stdev=28854.07
lat (usec): min=1041, max=250503, avg=15349.03, stdev=28853.95
clat percentiles (usec):
1.00th=[1565], 5.00th=[1860], 10.00th=[2057], 20.00th=[2311],
30.00th=[2540], 40.00th=[2769], 50.00th=[2999], 60.00th=[3326],
70.00th=[3818], 80.00th=[5014], 90.00th=[82314], 95.00th=[84411],
99.00th=[86508], 99.50th=[86508], 99.90th=[87557], 99.95th=[88605],
99.99th=[90702]
bw (KiB/s): min=265708, max=319488, per=100.00%, avg=266498.36, stdev=3766.74, samples=200
iops : min= 2075, max= 2496, avg=2082.01, stdev=29.43, samples=200
lat (msec) : 2=8.46%, 4=64.09%, 10=11.90%, 20=0.18%, 50=0.01%
lat (msec) : 100=15.37%, 500=0.01%
cpu : usr=5.34%, sys=1.90%, ctx=63555, majf=0, minf=28
IO depths : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.1%, 32=100.0%, >=64=0.0%
submit : $0=0.0$ %, $4=100.0$ %, $8=0.0$ %, $16=0.0$ %, $32=0.0$ %, $64=0.0$ %, $5=64=0.0$ %
complete : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.1%, 64=0.0%
issued rwt: total=0,208238,0, short=0,0,0, dropped=0,0,0
latency : target=0, window=0, percentile=100.00%, depth=32
Run status group 0 (all jobs):
WRITE: bw=260MiB/s (273MB/s), 260MiB/s-260MiB/s (273MB/s-273MB/s), io=25.4GiB (27.3GB), run=100045-100045msec
Disk stats (read/write):
vdb: ios=42/207998, merge=0/0, ticks=21/3173469, in gueue=3174831, util=99.95%



执行以下命令,测试硬盘的随机混合读写 IOPS(此处测试对象是名为 /dev/vdb 的设备)。

fio -bs=4k -ioengine=libaio -iodepth=32 -numjobs=4 -direct=1 rw=randrw -time_based -runtime=600 -refill_buffers -norandommap randrepeat=0 -group_reporting -name=fio-randrw-iops -size=10G filename=/dev/vdb

示例:

登录云服务器后,执行以下命令,测试设备名为 /dev/vdc 的 SSD 云硬盘的随机读 IOPS 性能。

fio -bs=4k -ioengine=libaio -iodepth=32 -numjobs=4 -direct=1 rw=randread -time_based -runtime=600 -refill_buffers -norandommap



-randrepeat=0 -group_reporting -name=fio-randread-iops -size=10G -

filename=/dev/vdc

测试结果如下图所示:

<pre>[root@VM_16_21_centos ~]# fio -bs=4k -ioengine=libaio -iodepth=32 -direct=1 -rw=randread -time_based -runtime=300 -refill_buffers -norandommap -randrepeat=0 -group_reporting -name=fio-randreadsiz</pre>
e=100G -filename=/dev/vdc
fio-randread: (g=0): rw=randread, bs=(R) 4096B-4096B, (W) 4096B-4096B, (T) 4096B-4096B, ioengine=lib
aio, iodepth=32
fio-3.1
Starting 1 process
Jobs: 1 (f=1): [r(1)][100.0%][r=18.8MiB/s,w=0KiB/s][r=4804,w=0 IOPS][eta 00m:00s]
fio-randread: (groupid=0, jobs=1): err= 0: pid=2689: Tue Jul 16 13:39:33 2019
read: IOFS=4800, BW=18.8MiB/s (19.7MB/s) (5626MiB/300017msec)
<pre>slat (usec): min=2, max=3183, avg= 7.55, stdev=14.34</pre>
clat (usec): min=209, max=777668, avg=6656.04, stdev=45385.49
lat (usec): min=371, max=777673, avg=6664.08, stdev=45385.46
clat percentiles (usec):
1.00th=[635], 5.00th=[832], 10.00th=[938], 20.00th=[1090],
30.00th=[1237], 40.00th=[1352], 50.00th=[1467], 60.00th=[1582],
70.00th=[1713], 80.00th=[1991], 90.00th=[9372], 95.00th=[19006],
<pre>99.00th=[38536], 99.50th=[341836], 99.90th=[734004], 99.95th=[750781],</pre>
99.99th=[767558]
bw (KiB/s): min= 256, max=38424, per=100.00%, avg=19202.16, stdev=13626.10, samples=600
iops : min= 64, max= 9606, avg=4800.52, stdev=3406.52, samples=600
lat (usec) : 250=0.01%, 500=0.12%, 750=2.64%, 1000=10.86%
lat (msec) : 2=66.45%, 4=5.87%, 10=4.59%, 20=6.49%, 50=2.32%
lat (msec) : 100=0.01%, 500=0.36%, 750=0.26%, 1000=0.05%
cpu : usr=1.38%, sys=5.00%, ctx=233328, majf=0, minf=63
IO depths : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.1%, 32=100.0%, >=64=0.0%
submit : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
complete : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.1%, 64=0.0%, >=64=0.0%
issued rwt: total=1440148,0,0, short=0,0,0, dropped=0,0,0
latency : target=0, window=0, percentile=100.00%, depth=32
Run status group 0 (all jobs):
READ: bw=18.8MiB/s (19.7MB/s), 18.8MiB/s-18.8MiB/s (19.7MB/s-19.7MB/s), io=5626MiB (5899MB), run=
300017-300017msec
Disk stats (read/write):
Vdc: 103=1440052/0, merge=0/0, ticks=94/8008/0, in_queue=948521/, util=99.98%

多块弹性云硬盘构建 LVM 逻辑卷

最近更新时间: 2025-05-28 16:18:43

LVM 简介

逻辑卷管理(Logical Volume Manager,LVM)通过在硬盘和分区之上建立一个逻辑层,将磁盘或分区划分为 相同大小的 PE(Physical Extents)单元,不同的磁盘或分区可以划归到同一个卷组(VG,Volume Group),在 VG 上可以创建逻辑卷(LV,Logical Volume),在 LV 上可以创建文件系统。 相较于直接使用磁盘分区的方式,LVM 的优势在于弹性调整文件系统的容量:

- 文件系统不再受限于物理磁盘的大小,可以分布在多个磁盘中。
- 例如,您可以购买 3 块 4TB 的弹性云硬盘并使用 LVM 创建一个将近 12TB 的超大文件系统。
- 可以动态调整逻辑卷大小,不需要对磁盘重新分区。

当 LVM 卷组的空间无法满足您的需求时,您可以单独购买弹性云硬盘并挂载到相应的云服务器上,然后将其添加到 LVM 卷组中进行扩容操作。

构建 LVM

() 说明:

本示例基于 CentOS 8.1 操作系统,以使用 3 块弹性云硬盘通过 LVM 创建可动态调整大小的文件系统为例。如下图所示:

S 1	~			
[root@VM_63_126_centos /	~]# fdisk -l grep	vd gre	p -v vda	grep -v vdb
Disk /dev/vdc: 10.7 GB,	10737418240 bytes			
Disk /dev/vdd: 10.7 GB,	10737418240 bytes			
Disk /dev/vde: 10.7 GB,	10737418240 bytes			

步骤1 创建物理卷 PV

- 1. 以 root 用户 登录 Linux 云服务器。
- 2. 执行以下命令,创建一个物理卷(Physical Volume, PV)。





创建成功则如下图所示:

<pre>root@VM_63_126_centos ~]# pvcreate /dev/vdc /dev/</pre>	/vdd /dev/vde
Physical volume "/dev/vdc" successfully created	
Physical volume "/dev/vdd" successfully created	
Physical volume "/dev/vde" successfully created	

3. 执行以下命令,查看现在系统中的物理卷。

```
lvmdiskscan | grep LVM
```

[root@VM_63_	126_0	centos ~]# lvmdiskscan grep LVM
/dev/vdc	[10.00 GiB] LVM physical volume
/dev/vdd	[10.00 GiB] LVM physical volume
/dev/vde	[10.00 GiB] LVM physical volume
3 LVM phys	ical	volume whole disks
0 LVM phys	ical	volumes

步骤 2 创建卷组 VG

1. 执行以下命令,创建 VG。

vgcreate [-s <**指定**PE**大小**>] <**卷组名> <物理卷路径**>

本文以创建一个名为 "lvm_demo0" 的卷组并加入 vdc vdd 两个 PV 为例,则执行:

vgcreate lvm_demo0 /dev/vdc /dev/vdd

当提示 "Volume group "<卷组名>" successfully created"时,表示卷组创建成功。创建成功则如下 图所示:

[root@VM_63_126_centos ~]# vgcreate lvm_demo0 /dev/vdc /dev/vdd Volume group "lvm_demo0" successfully created

若想在卷组创建完成后继续加入 PV,可以执行以下命令向卷组中添加新的物理卷。

vgextend 卷组名 新物理卷路径

本示例中执行 vgextend lvm_demo0 /dev/vde 将 vde 也加入卷组中。

卷组创建完成后,可执行 vgs 、 vgdisplay 等命令查看当前系统中的卷组信息。如下图所示:

步骤 3 创建逻辑卷 LV



1. 执行以下命令,创建 LV。

lvcreate [-L **<逻辑卷大小**>][-n <mark><逻辑卷名称</mark>>] **<**VG**名称**>

本文以创建一个 8GB 的名为 "lv_0" 的逻辑卷为例,则执行:

```
lvcreate -L <u>8G -n lv_0 lvm_demo0</u>
```

创建成功则如下图所示:

```
[root@VM_63_126_centos ~]# lvcreate -L 8G -n lv_0 lvm_demo
Logical volume "lv_0" created
```

() 说明:

执行 pvs 命令,可查看到此时只有 /dev/vdc 被使用了8GB。如下图所示:

[root@VM_63	126_centos	s ~]#	pvs		
	PV	VG	Fmt	Attr	PSize	PFree
	/dev/vdc	lvm_demo	lvm2	a	10.00g	2.00g
	/dev/vdd	lvm demo	lvm2	a	10.00g	10.00g
	/dev/vde	lvm demo	lvm2	a	10.00g	10.00g
					2	2

步骤 4 创建并挂载文件系统

1. 执行以下命令,在创建好的逻辑卷上创建文件系统。

mkfs.ext3 /dev/lvm_demo0/lv_0

2. 执行以下命令,创建挂载节点目录 / vg0 。

mkdir /vg0

3. 执行以下命令, 挂载文件系统。

mount /dev/lvm_demo0/lv_0 /vg0

4. 可以通过以下命令查找 LVM 的挂载情况。

nount | grep lvm

挂载成功则如下图所示:



云硬盘

[root@VM_63_126_centos ~]# mount | grep lvm /dev/mapper/lvm_demo-lv_0 on /root/vg0 type ext3 (rw)

步骤 5 动态扩展逻辑卷及文件系统大小

▲ 注意:

仅当 VG 容量有剩余时,LV 容量可动态扩展。扩展 LV 容量后,需一并扩展创建在该 LV 上的文件系统的 大小。

1. 执行以下命令,扩展逻辑卷大小。

lvextend [-L +/- < 增减容量>] <逻辑卷路径>

本文以向逻辑卷 "lv_0" 扩展4GB容量为例,则执行:

lvextend -L +4G /dev/lvm_demo0/lv_0

扩展成功则如下图所示:

```
[root@VM_63_126_centos vg0]# lvextend -L +4G /dev/lvm_demo/lv_0
Size of logical volume lvm_demo/lv_0 changed from 8.00 GiB (2048 extents) to 12.00 GiB (3072 extents).
Logical volume lv_0 successfully resized
```

① 说明:
 执行 pvs 命令,可查看到此时 /dev/vdc 已被完全使用, /dev/vdd 被使用了2GB。如下图所示:
 [root@VM 63 126 centos vg0]# pvs

[100 c@m_00]	120_000000	, igo.	17 P V		
PV	VG	Fmt	Attr	PSize	PFree
/dev/vdc	lvm_demo	lvm2	a	10.00g	Θ
/dev/vdd	lvm_demo	lvm2	a	10.00g	7.99g
/dev/vde	lvm_demo	lvm2	a	10.00g	10.00g

2. 执行以下命令,扩展文件系统。

resize2fs /dev/lvm_demo0/lv_0



扩展成功则如下图所示:

扩展成功后,可执行以下命令,查看逻辑卷的容量是否变为12GB。

df -h

MBR 分区云硬盘扩容至大于 2TB

最近更新时间: 2025-06-12 11:02:32

操作场景

当您的云硬盘在 <mark>已有 MBR 分区并已创建文件系统</mark> 的情况下,扩容至大于 2TB,此时 MBR 分区下的文件系统已无 法扩容至大于 2TB,请参考本文将 MBR 分区形式转换为 GPT 分区形式。

注意事项

请谨慎操作,误操作可能会导致数据丢失或异常。请给对应云硬盘创建快照,完成数据备份。详情请参见 创建快 照。如出现误操作导致数据丢失,则可回滚快照进行数据恢复。

操作步骤

Linux实例

本示例以 CentOS 8.1 操作系统为例。

- 1. 登录云服务器,详情请参见 使用标准登录方式登录 Linux 实例(推荐)。
- 2. 执行如下命令,查看分区类型是否为 MBR。

若结果如下图所示(根据操作系统不同略有不同),则说明为 MBR 分区形式。



- 3. 安装 gdisk 工具。
 - CentOS 操作系统,执行如下命令。

yum install -y gdisk

○ Ubuntu 或 Debian 操作系统, 执行如下命令。

sudo apt-get install gdisk -y

4. 执行 MBR 转 GPT 操作。



执行如下 sgdisk 命令。注意: sgdisk 命令只能对设备进行操作如(/dev/vdb),不能对分区如(/dev/vdb1)操作,否则可能会损坏文件系统,操作前需确保已经做好快照备份 创建快照 。

sudo sgdisk -g **裸设备名 (如**/dev/vdb**)**

如果出现报错"Warning! Secondary partition table overlaps the last partition by 33 blocks! You will need to delete this partition or resize it in another utility"则表示扩容云硬 盘尾部剩余空间不足33个扇区,没有足够的空间用于 MBR 转换为 GPT,建议至少再扩容1G空间,参考 扩容云硬盘。

5. 确认修改 GPT 分区是否成功。

执行如下命令查看修改 GPT 分区是否成功。

fdisk -l

若结果如下图所示(根据操作系统不同略有不同),则说明已经修改为 GPT 分区。

Disk /dev/vdb: 2684.4 GB, 2684354560000 bytes, 5242880000 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/0 size (minimum/optimal): 262144 bytes / 262144 bytes
Disk label type: gpt
Disk identifier: 71ADC6FB-2232-4BA9-9622-1890138741D0
Start End Size Type Name

6. 扩展分区大小。

1

扩容文件系统前, 需把扩容云硬盘大小扩展到对应分区, 执行 growpart 命令(如下命令以 vdb1 扩容 为例)。

200G Linux filesyste Linux filesystem

growpart /dev/vdb 1

2048

419430399

7. 执行对应命令,扩容文件系统。

扩容 ext 文件系统

执行以下命令,扩容 ext 文件系统。

resize2fs /dev/**对应分区**

本文以分区路径以 /dev/vdb1 ,则执行:





xfs_growfs /dev/**对应分区**

本文以分区路径以 /dev/vdb1 ,则执行:

xfs_growfs_/dev/vdb1

8. 参见 设置开机自动挂载 操作,设置分区自动挂载。

Windows实例

本示例以 Windows Server 2025 操作系统为例,借用第三方免费版磁盘管理工具 傲梅分区助手 执行扩容。

1. 使用标准方式登录 Windows 实例。

打开**磁盘管理器**,当尝试进行磁盘分区扩容时,会发现**扩展卷**显示为灰色,无法直接扩容。

		连接成功。	如需粘贴命令,请点击 <u>这里</u> 。 您可	根据指引调整屏幕分辨率。	
8	一 磁盘管理			- o x	
回收站	文件(F) 操作(A) 查看(◆ ◆ ↓ 〒 ┃ 1 1 1	(V) 帮助(H) 严 🗙 🕝 🔒 📴 🖾			
Microsoft Edge	巻 布 = (C:) 简 = (磁盘 0 磁盘分区 1) 简 = 新加卷 (E:) 简	5局 <u> </u>	状态 容量 状态良好(49.90 GB 状态良好(100 MB 状态良好(2048.00 GB	可用空间 % 可用 37.77 GB 76 % 51 MB 51 % 2047.83 100 %	
PAGreen	 磁盘 0 基本 50.00 GB 联机 状态) MB NTFS 5良好 (系统, 活 大态良好 (启动, 页面	1文件, 故障转储, 主分区)		
PA_Green	- 磁盘 1 基本 32000.00 GB 联机 状态	□ 0卷 (E:) 48.00 GB NTFS 虎良好 (主分区)	29952.00 GB 未分配		
			打开(O) 资源管理器(E) 将分区标记为活动分区(M) 更改驱动器号和路径(C) 格式化(F)		
	■ 未分配 ■ 主分区		扩展卷(X)		
			》上 ^{编楼} (□) 添加镜像(A) 删除卷(D)		
			厪性(P)		
			帮助(H)		

2. 以管理员身份打开 Windows PowerShell,执行以下命令,检查是否为 MBR 分区。

Get-Disk | ft Number,Size,PartitionStyle

如果输出结果中 PartitionStyle 为 MBR,则表示为 MBR 分区。



- 3. 将磁盘格式由 MBR 转换为 GPT 格式。
 - 3.1 本示例使用绿色免费版的傲梅分区助手,下载并解压 PAGreen.7z 之后,进入 PA_Green 文件 夹,右击以管理员身份运行 PartAssist.exe,打开傲梅分区助手,选中要转换 GPT 格式的硬盘,依 次点击转换 —> 磁盘转换。

📤 傲梅分区助手 -	免费无损分区软	次件						☆ � ≡	_ 0	×
		→ 克隆	🔓 转换 🎤 清理	🔍 恢复	擦除	- ~ - 1	Vit -	IЩ		
名称	类型	容量 已使	磁盘转换			F F				
₩₩₩, 硬盘0	基本 MBR	50.00GB	磁盘转换 转换磁盘为GPT/M	IBR格式				HDD		
*:	NTFS	100.00 45.4	动态硬盘转换器		惩 统	是		<u>硬盘</u> 1		
С:	NTFS	49.90GB 12.3	■【= 将磁曲转换为动态(转换分区	遮描/基本磁盘		是		31.25 TB,基本 MBR		
ᡂ_ 硬盘1	基本 MBR	31.25TB	转換NTFS/FAT325							
E : 新	NTFS	2.00TB 167.	◎ F 将分区转换力NTFS ————————————————————————————————————	5分区/FA132分区		是	し、克隆硬銀			
*	未分配空间	29.25TB 0.00k	(B 逻辑	无			分 快速分	×		
								BR		
						-	面删除所有	有分区		
							▲ 擦除硬度	±		
硬盘0	*: C	: 0 00GB/75% 容闳					** 0 ====			
基本 MBR	NTF N	NTFS,系统,主	,				分区研	片整埋		
硬盘1 31.25TB	■:新加老 2.00TB(9.	■ 29.25TB(100%)	6 空闲)							
基本 MBR	NTFS,主	未分配空间								

3.2 单击**确定**。

转换GPT/MBR			?	×
MBR与GPT硬盘间的转换.				
当前选中的磁盘: 硬盘1	Ţ			
您确定要转换这个MBR硬盘到GPT硬	<u>唐</u> 曲吗?			
		确	定(<u>O</u>)	

3.3 单击**提交**。

傲梅分区助手 -	免费无损分区软件	ŧ					☆ � ≡ _ □
€提交	5 ¢	→ 克隆	🕴 转换	🦯 清理	🔍 恢复 🔤 擦除	-^	测试 二月
名称	类型 容	_跬 E	使用	类型	状态	F	
₩, 硬盘0	基本 MBR 50	0.00GB					
*:	NTFS 10	00.00 45	.45MB	ŧ	活动&系统	是	E:新加卷
С:	NTFS 49	9.90GB 12	.39GB	ŧ	引导	是	2.00 TB(99% 空闲),NTFS
●●● 硬盘1	基本 GPT 31	1.25TB					
E : 新	NTFS 2.	.00TB 16	7.79MB	GPT	无	是	
*	未分配空间 29	9.25TB 0.0	ООКВ	GPT	无		
						-	
1 1			_				
₩₩ 硬盘0	*: C:		_				
50.00GB	100 49.9	90GB(75% 空 EC 系体 主	(闲)				
		, 9, 20 20, <u>-</u>					
HOD							
硬盘1 31.25TB	E:新加巷 2.00TB(9	29.25TB(10	0% 空闲)				
基本 GPT	NTFS,主	未分配空间					

3.4 单击**执行 —> 是**。

	佳备就绪, 请确认 ⁻	下面的操作. 然后点击执行的	安钮让程序正式地工作.	
当削禦作致1, 忌禦作 转换MBR到GPT	-#X I	分区助车	×	
硬盘: 大小:	1 31.25TB	? 您现在就要执	行这些操作吗?	
		是(Y)	否(N)	
时间估算: 实现这些操作	乍可能需要花00:0	0:30时间或者更多.		
				执行(P)

3.5 单击确定。

操作进度					
N S R	当前操作数1,总操作数1				
	转换MBR到GPT		分区助手		×
	硬盘: 大小:	31.25TB	苏喜!所有	操作都已成功完成.	
				确定	
	当前操作进度				100%
	总进度				100%
	在完成所有操作后自动	关机(<u>S</u>).			
1) 操作期间请保持设备稳定	宝运行 <i>,避</i> 免中断或断电。				取消(<u>C</u>)

4. 磁盘分区扩容。

重新打开磁盘管理器 或者 在已打开的磁盘管理器界面,选中操作页签,选择**重新扫描磁盘**。

(/) 府川ム

📻 磁盘管	理								-	\times
文件(F)	操作(A)	查看(V)	帮助(H)							
<	刷新	i(F)		<u>,</u> 🖅						
卷	重新扫描磁盘(R)				文件系统	状态	容量	可用空间	% 可用	
= (C:)	创建	VHD			NTFS	状态良好 (49.90 GB	37.50 GB	75 %	
(磁盘 (附加	附加 VHD			NTFS	状态良好 (100 MB	51 MB	51 %	
➡ 新加卷	巻				NTFS	状态良好 (2048.00 GB	2047.83	100 %	
	帮助(H)									
- 磁盘	0									1
基本 50.00 GB 100 MB NTFS 联机 状态良好 (系统, 活				(C:) 49.90 状态良	GB NTFS 好 (启动, 页面)	文件, 故障转储,	主分区)			
- 磁盘	1		· · · · · · · · · · · · · · · · · · ·							
基本 32000.00 GB 联机 新加卷 (E:) 2048.00 GB NTFS 状态良好 (基本数据分) (26							
CD-ROM	ROM 0 1 (D:)									
无媒体										
■ 未分配	■ 主分区									

此时,再次进行分区扩容,会发现**扩展卷**选项已经可选了。



📅 磁盘管理							-		\times		
文件(F) 操作(A)	查看(V) 帮助(H)									
🗢 🄿 📰 <table-cell></table-cell>	TT 🗩 🗙 🗹	🔒 🔎 🛙	8								
卷	布局	类型	文件系统	状态	容量	可用空间	% 可用				
💻 (C:)	简单	基本	NTFS	状态良好 (49.90 GB	37.50 GB	75 %				
	1) 简单	基本	NTFS	状态良好 (100 MB	51 MB	51 %				
🛶 新加卷 (E:)	简单	基本	NTFS	状态良好 (2048.00 GB	2047.83	100 %				
	1										
➡ 磁盘 0 基本 50.00 GB 联机	磁盘 0 基本 基本 50.00 GB 50.00 GB 100 MB NTFS 状态良好 (系统, 活 49.90 GB NTFS 状态良好 (系统, 活 状态良好 (启动, 页面文件, 故障转储, 主分区)										
- 磁盘 1											
基本 32000.00 GB 联机	新加卷 (E:) 2048.00 GB N 状态良好 (基本	NTFS 《数据分区)		2995) 未分配	2.00 GB B						
				🚽 ग्रम	(O)						
				资源	管理器(E)						
				将分	区标记为活动分	<u>×</u> (M)					
无媒体				更改	驱动器号和路径(C)					
				格式	化(F)						
■ 未分配 ■ 主分区				扩展	卷(X)						
7				压缩	卷(H)						

下图示例已经将磁盘1从2TB扩容到了4TB。



									\sim			
文件(F) 操作(A) 查看(V) 帮助(H)												
卷	局	类型	文件系统	状态	容量	可用空间	% 可用					
💻 (C:) 简	单	基本	NTFS	状态良好 (. 49.90 GB	37.58 GB	75 %					
	单	基本	NTFS	状态良好 (. 100 MB	66 MB	66 %					
🚍 新加卷 (E:) 简	前单	基本	NTFS	状态良好 (. 4096.00 GB	4095.76	100 %					
-												
■ 磁盘 0												
50.00 GB 100												
联机状态	な良好 (系统,	活状态良	好 (启动, 页面文	(件,故障转储	¥, 主分区)							
		T										
「「「「「「「」」」	n# (E·)	///////////////////////////////////////		//////					_			
32000.00 GB 409	96.00 GB NT	FS		2	7904.00 GB							
联机 状态	悠良好 (基本数	数据分区)		#	卡分配							
CD-ROM (D:)												
无媒体												
		Ν										
		13										
■ 未分配 ■ 主分区												