

Cloud Block Storage

Tutoriais Práticas

Product Documentation



[Declaração de direitos autorais]

Direitos autorais ©2013–2025 Tencent Cloud. Todos os direitos reservados.

Os direitos autorais deste documento pertencem exclusivamente à Tencent Cloud. Sem a autorização prévia por escrito da Tencent Cloud, nenhuma entidade pode copiar, modificar, plagiar ou disseminar todo ou parte do conteúdo deste documento, sob qualquer forma.

[Declaração de marca registrada]



e outras marcas registradas relacionadas aos Serviços da Tencent Cloud são de propriedade das entidades relevantes de empresas sob o Grupo Tencent. Além disso, as marcas registradas de entidades terceirizadas envolvidas neste documento são de propriedade dos detentores de direitos, de acordo com a lei.

[Declaração de serviço]

Este documento tem como objetivo fornecer aos clientes uma visão geral de todos ou parte dos produtos e serviços da Tencent Cloud no momento. O conteúdo de alguns produtos e serviços pode ser ajustado. Os tipos de produtos, serviços e os padrões de serviço da Tencent Cloud que você adquirir deverão ser acordados pelo contrato comercial entre você e a Tencent Cloud. Salvo acordo em contrário entre ambas as partes, a Tencent Cloud não assume nenhuma promessa ou garantia, expressa ou implícita, com relação ao conteúdo deste documento.

Contents

Tutoriais Práticas

Medida do desempenho de discos em nuvem

Criação de volumes lógicos LVM com vários discos em nuvem elásticos

Tutoriais Práticas

Medida do desempenho de discos em nuvem

Last updated: 2023-12-26 11:17:00

Métricas

Os dispositivos do CBS do Tencent Cloud têm desempenho e preços diferentes conforme o tipo. Para obter mais informações, consulte [Tipos de discos em nuvem](#). Como aplicativos diferentes têm cargas de trabalho diferentes, se a quantidade de solicitações de E/S for baixa, o disco em nuvem poderá não apresentar o desempenho máximo.

As métricas a seguir costumam ser usadas para medir o desempenho de um disco em nuvem:

- IOPS: contagem de leitura/gravação por segundo. O IOPS varia de acordo com o tipo de unidade subjacente do dispositivo de armazenamento.
- Taxa de transferência: volume de dados lidos/gravados por segundo, com unidade em MB/s.
- Latência: tempo decorrido desde o envio de uma operação de E/S até o recebimento da confirmação (em segundos).

Ferramenta de teste

A FIO é uma ferramenta para teste de desempenho do disco. Ela é usada para realizar teste de estresse e verificação no hardware. Este documento usa a FIO como exemplo.

Nós recomendamos que você use a FIO junto com o mecanismo de E/S do libaio para realizar o teste. Instale a FIO e o libaio consultando [Instalação de ferramentas](#).

ⓘ Nota:

- Para evitar danos em arquivos importantes do sistema, não realize o teste FIO no disco do sistema.
- Para evitar corrompimento dos dados causado pelo corrompimento de metadados do sistema de arquivos de base, não realize o teste em um disco de dados empresarial.
- Confira se o arquivo `/etc/fstab` NÃO contém a configuração de montagem do disco a ser testado. Caso contrário, poderá não ser possível inicializar o CVM.

Objetos de teste recomendados

- Nós recomendamos que você realize o teste FIO em discos vazios que não armazenam dados importantes e recrie o sistema de arquivos após a conclusão do teste.

- Ao testar o desempenho do disco, recomendamos que você teste diretamente discos de dados brutos (como `/dev/vdb`).
- Ao testar o desempenho do sistema de arquivos, recomendamos que você especifique o teste do arquivo (como `/data/file`).

Instalação de ferramentas

1. Faça login no CVM conforme indicado em [Faça login em instância do Linux usando o método de login padrão](#). Aqui, use como exemplo o CVM que está executando o sistema operacional CentOS 7.6.
2. Execute o seguinte comando para verificar se o disco em nuvem está alinhado com 4KiB.

```
fdisk -lu
```

Conforme mostrado abaixo, se o valor inicial na saída do comando for divisível por 8, significa que o disco está alinhado com 4KiB. Caso contrário, realize o alinhamento com 4KiB antes do teste.

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

3. Execute os comandos a seguir em sequência para instalar as ferramentas de teste, FIO e libaio.

```
yum install libaio -y
```

```
yum install libaio-devel -y
```

```
yum install fio -y
```

Após concluído, inicie o teste de desempenho do disco em nuvem conforme indicado no exemplo de teste abaixo.

Exemplo de teste

As fórmulas de teste para diferentes cenários são basicamente as mesmas, exceto os parâmetros `rw`, `iodepth` e `bs` (tamanho do bloco). Por exemplo, o `iodepth` ideal para cada carga de trabalho é diferente, já que depende da sensibilidade do seu aplicativo ao IOPS e à latência.

Descrição do parâmetro:

Parâmetro	Observações	Valor de amostra
-----------	-------------	------------------

bs	Tamanho do bloco por solicitação. Valores válidos incluem 4k, 8k e 16k.	4k
ioengine	Mecanismo de E/S. Nós recomendamos que você use o mecanismo de E/S assíncrono do Linux.	libaio
iodepth	Profundidade enfileirada de uma solicitação de E/S.	1
direct	<p>Especifica o modo direto.</p> <ul style="list-style-type: none"> • True (1) indica que o identificador O_DIRECT é especificado, o cache de E/S será ignorado e os dados serão gravados diretamente. • False (0) indica que o identificador O_DIRECT não é especificado. <p>O padrão é True (1).</p>	1
rw	Modo de leitura e gravação. Valores válidos incluem read, write, randread, randwrite, randrw e rw, readwrite.	read
time_based	Especifica que o modo de tempo está sendo usado. Contanto que a FIO seja executada com base no tempo, não é necessário definir esse parâmetro.	N/A
runtime	Especifica a duração do teste, que é o tempo de execução da FIO.	600
refill_buffers	A FIO vai preencher novamente o buffer de E/S a cada envio. A configuração padrão é preencher o buffer de E/S apenas no início e reutilizar os dados.	N/A
norandommap	Ao realizar operações de E/S aleatórias, a FIO sobrescreve cada bloco do arquivo. Se este parâmetro estiver definido, uma nova compensação será selecionada sem visualizar o histórico de E/S.	N/A
randrepeat	Especifica se a sequência aleatória é repetível. True (1) indica que a sequência aleatória é repetível. False (0) indica que a sequência aleatória não é repetível. O valor padrão é True (1).	0
group_reporting	Quando vários jobs são simultâneos, as estatísticas do group todo são impressas.	N/A
name	Nome do job.	fio-read
size	Espaço de endereço do teste de E/S.	100 GB
filename	Objeto de teste, que é o nome do disco a ser testado.	/dev/sdb

Alguns casos de uso comuns são:

- **bs=4k iodepth=1:** teste de leitura/gravação aleatória, que pode refletir o desempenho de latência de um disco.

Execute o comando a seguir para testar a latência de leitura aleatória do disco:

```
fio -bs=4k -ioengine=libaio -iodepth=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
```

Execute o comando a seguir para testar a latência de gravação aleatória do disco:

```
fio -bs=4k -ioengine=libaio -iodepth=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
```

Execute o comando a seguir para testar o desempenho de latência de leitura e gravação aleatória híbrida de um disco em nuvem SSD:

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

A figura a seguir mostra a saída do comando:

```
fio-read: (g=0): rw=randrw, bs=(R) 4096B-4096B, (W) 4096B-4096B, (T) 4096B-4096B, ioengine=libaio, iodepth=1
fio-3.1
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],
      | 70.00th=[ 660], 80.00th=[ 693], 90.00th=[ 783], 95.00th=[ 914],
      | 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, majf=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: teste de leitura/gravação sequencial, que pode refletir o desempenho da taxa de transferência do disco.

Execute o comando a seguir para testar a largura de banda da taxa de transferência de leitura sequencial:

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

Execute o comando a seguir para testar a largura de banda da taxa de transferência de gravação sequencial:

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


Execute o comando a seguir para testar o desempenho da taxa de transferência de leitura sequencial de um disco em nuvem SSD:

```
fio --bs=128k --ioengine=libaio --iodepth=32 --direct=1 --rw=read --
time_based --runtime=100 --refill_buffers --norandommap --randrepeat=0
--group_reporting --name=fio-rw --size=1G --filename=/dev/vdb
```

A figura a seguir mostra a saída do comando:

```
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%, >=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 rwts: 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 queue=3174831, util=99.95%
```

- bs=4k iodepth=32: teste de leitura/gravação aleatória, que pode refletir o desempenho de IOPS do disco rígido.

Execute o comando a seguir para testar o IOPS de leitura aleatória do disco:

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

Execute o comando a seguir para testar o IOPS de gravação aleatória do disco:

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

Teste o desempenho do IOPS de leitura aleatória do disco em nuvem SSD. Isso é mostrado na figura a seguir:

```
[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-randread --siz
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: IOPS=4800, BW=18.8MiB/s (19.7MB/s) (5626MiB/300017msec)
    slat (usec): min=2, max=3183, avg= 7.55, stdev=14.34
    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],
      | 99.00th=[ 38536], 99.50th=[341836], 99.90th=[734004], 99.95th=[750781],
      | 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 rwts: 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: ios=1440052/0, merge=0/0, ticks=9478008/0, in_queue=9485217, util=99.98%
```

Criação de volumes lógicos LVM com vários discos em nuvem elásticos

Last updated: 2023-12-26 11:17:30

Introdução ao LVM

O Gerenciador de volume lógico (LVM, na sigla em inglês) divide seus discos ou partições em unidades de extensões físicas (PEs, na sigla em inglês) com o mesmo tamanho ao criar uma camada lógica sobre os discos e as partições. Ele combina diferentes discos e partições no mesmo grupo de volume (VG, na sigla em inglês), para que você possa criar volumes lógicos (LVs, na sigla em inglês) no VG, e os sistemas de arquivos nos LVs.

Em comparação com o uso de partições de disco diretamente, o LVM permite o escalonamento elástico dos seus sistemas de arquivos.

- O sistema de arquivos não está mais limitado pelo tamanho do disco físico. Na verdade, ele pode ser distribuído em vários discos:

Por exemplo, você pode adquirir 3 discos em nuvem elásticos de 4 TB e usar o LVM para criar um sistema de arquivos maior com até 12 TB.

- Você pode redimensionar os LVs de forma dinâmica em vez de reparticionar seus discos.

Quando a capacidade do VG do LVM não consegue atender às suas necessidades, você pode adquirir um disco em nuvem elástico à parte, montá-lo na sua instância do CVM e escalar seu VG adicionando o disco em nuvem a ele.

Criação do LVM

Nota:

Os exemplos a seguir usam 3 discos em nuvem elásticos para criar um sistema de arquivos redimensionável de forma dinâmica por meio do LVM:

```
[root@VM_63_126_centos ~]# fdisk -l | grep vd | grep -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
```

Etapa 1: criar um volume físico (PV)

1. [Faça login no seu CVM do Linux](#) como usuário raiz.
2. Execute o seguinte comando para criar os PVs:

```
pvcreate <disk path 1> ... <disk path N>
```

Use `/dev/vdc` , `/dev/vdd` e `/dev/vde` como exemplo e execute:

```
pvcreate /dev/vdc /dev/vdd /dev/vde
```

A figura a seguir mostra a saída do comando quando a criação obtém êxito:

```
[root@VM_63_126_centos ~]# pvcreate /dev/vdc /dev/vdd /dev/vde
Physical volume "/dev/vdc" successfully created
Physical volume "/dev/vdd" successfully created
Physical volume "/dev/vde" successfully created
```

3. Veja os volumes físicos no sistema atual com o comando:

```
lvmdiskscan | grep LVM
```

```
[root@VM_63_126_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 physical volume whole disks
0 LVM physical volumes
```

Etapa 2: criar um grupo de volume (VG)

Execute o seguinte comando para criar um grupo de volume:

```
vgcreate [-s <specified PE size>] <VG name> <PV path>.
```

Considere que você queira criar um VG chamado “lvm_demo0” e execute:

```
vgcreate lvm_demo0 /dev/vdc /dev/vdd
```

A figura a seguir mostra a saída do comando quando a criação obtém êxito:

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

Quando a mensagem "Volume group successfully created (Grupo de volume criado com êxito)" for exibida, significa que seu VG foi criado com êxito.

- Após a criação, você pode adicionar um novo PV ao VG com o comando:

```
vgextend VG name New PV path
```

A figura a seguir mostra a saída do comando quando a operação obtém êxito:

```
[root@VM_63_126_centos ~]# vgextend lvm_demo0 /dev/vdf
Volume group "lvm_demo0" successfully extended
[root@VM_63_126_centos ~]#
```

- Após criado, você pode executar `vgs`, `vgdisplay` ou outros comandos para exibir os VGs no sistema atual, conforme mostrado abaixo:

```
[root@VM_63_126_centos ~]# vgs
VG      #PV #LV #SN Attr   VSize  VFree
lvm_demo 3   0   0 wz--n- 29.99g 29.99g
```

Etapa 3: criar um volume lógico (LV)

Execute o seguinte comando para criar um LV:

```
lvcreate [-L <LV size>] [-n <LV name>] <VG name>.
```

Considere que você queira criar um volume lógico de 8 GB chamado "lv_0" e execute:

```
lvcreate -L 8G -n lv_0 lvm_demo0.
```

A figura a seguir mostra a saída do comando quando a criação obtém êxito:

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

Nota:

Execute `pvs` e você verá que os 8 GB vêm apenas de `/dev/vdc`, conforme mostrado abaixo:

```
[root@VM_63_126_centos ~]# 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
```

Etapa 4: criar e montar um sistema de arquivos

1. Crie um sistema de arquivos em um LV existente usando o comando:

```
mkfs.ext3 /dev/lvm_demo0/lv_0.
```

2. Monte o sistema de arquivos com o comando:

```
mount /dev/lvm_demo0/lv_0 vg0/.
```

A figura a seguir mostra a saída do comando quando a montagem obtém êxito:

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

Etapa 5: realizar escalonamento dinâmico no seu volume lógico e no sistema de arquivos

⚠ Atenção:

LVs podem ser escalonados de forma dinâmica apenas quando a capacidade do VG não está totalmente utilizada. Ao escalonar a capacidade de um LV, você também precisa escalonar o sistema de arquivos criado nesse LV.

1. Escalone um LV executando este comando:

```
lvextend [-L +/- <capacity amount>] <LV path>.
```

Considere que você queira aumentar a capacidade do LV chamado “lv_0” em 4 GB, então execute:

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

A figura a seguir mostra a saída do comando quando o escalonamento obtém êxito:

```
[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
```

ⓘ Nota:

Execute `pvs` e você verá que `/dev/vdc` foi totalmente usado, e 2 GB foram usados de `/dev/vdd`, conforme mostrado abaixo:

```
[root@VM_63_126_centos vg0]# pvs
PV          VG      Fmt Attr PSize  PFree
/dev/vdc    lvm_demo lvm2 a-- 10.00g  0
/dev/vdd    lvm_demo lvm2 a-- 10.00g  7.99g
/dev/vde    lvm_demo lvm2 a-- 10.00g 10.00g
```

2. Escalone o sistema de arquivos com o comando

```
resize2fs /dev/lvm_demo0/lv_0.
```

A figura a seguir mostra a saída do comando quando o escalonamento obtém êxito:

```
[root@VM_63_126_centos vg0]# resize2fs /dev/lvm_demo/lv_0
resize2fs 1.41.12 (17-May-2010)
Filesystem at /dev/lvm_demo/lv_0 is mounted on /root/vg0; on-line resizing required
old_desc_blocks = 1, new_desc_blocks = 1
Performing an on-line resize of /dev/lvm_demo/lv_0 to 3145728 (4k) blocks.
The filesystem on /dev/lvm_demo/lv_0 is now 3145728 blocks long.

[root@VM_63_126_centos vg0]# df -h
Filesystem                Size      Used Avail Use% Mounted on
/dev/vda1                  7.9G    1019M   6.5G  14% /
/dev/mapper/lvm_demo-lv_0  12G      549M    11G   5% /root/vg0
```

Após a conclusão, você pode executar o comando a seguir para ver se a capacidade do seu LV mudou para 12 GB:

```
df -h
```