

Como adicionar espaço de swap no Ubuntu 20.04

Link: <https://www.digitalocean.com/community/tutorials/how-to-add-swap-space-on-ubuntu-20-04-pt>

Introdução

Um das maneiras de se proteger contra erros de memória insuficiente em aplicativos é através da adição de um espaço de swap ao seu servidor. Neste guia, falaremos sobre como adicionar um arquivo swap a um servidor Ubuntu 20.04.

Aviso: embora o swap seja geralmente recomendado para sistemas que utilizam discos rígidos tradicionais, o uso do swap em SSDs pode causar problemas de degradação de hardware ao longo do tempo. Por este motivo, não recomendamos a habilitação do swap na DigitalOcean ou em qualquer outro provedor que utilize armazenamento SSD.

O que é o Swap?

O *Swap* é uma parcela do armazenamento do disco rígido que foi reservada para o sistema operacional com o objetivo de armazenar temporariamente dados que ele não consegue mais reter na RAM. Isso permite que você aumente a quantidade de informações que seu servidor consegue manter em sua memória de trabalho, com algumas advertências. O espaço de swap no disco rígido será usado principalmente quando já não houver espaço suficiente em RAM para manter os dados do aplicativo em uso.

As informações gravadas no disco ficarão significativamente mais lentas do que as informações mantidas em RAM, mas o sistema operacional preferirá manter os dados do aplicativo em memória e usar o swap para os dados mais antigos. De maneira geral, ter espaço de swap como uma alternativa para quando a RAM do seu sistema estiver esgotada pode ser uma boa estratégia de segurança contra exceções de memória insuficiente nos sistemas com armazenamento disponível que não seja SSD.

Passo 1 – Verificando o Sistema em Relação às Informações de Swap (troca)

Antes de começarmos, podemos verificar se o sistema já tem algum espaço de swap (troca) disponível. É possível ter vários arquivos de swap ou partições de swap, mas geralmente um deve ser o suficiente.

Podemos descobrir se o sistema tem algum swap configurado digitando:

```
1. sudo swapon --show
```

Se você não receber nenhum resultado, isso significa que seu sistema não tem espaço de swap disponível atualmente.

Você pode verificar se não existe um swap ativo usando o utilitário `free`:

```
1. free -h
```

Output

	total	used	free	shared	buff/cache	available
Mem:	981Mi	122Mi	647Mi	0.0Ki	211Mi	714Mi
Swap:	0B	0B	0B			

Como você pode ver na linha **Swap** do resultado, nenhum swap está ativo no sistema.

Passo 2 – Verificando o Espaço Disponível na Partição do Disco Rígido

Antes de criarmos nosso arquivo de swap, verificaremos o uso atual do disco para garantir que temos espaço suficiente. Faça isso digitando:

1. `df -h`

Output

Filesystem	Size	Used	Avail	Use%	Mounted on
udev	474M	0	474M	0%	/dev
tmpfs	99M	932K	98M	1%	/run
/dev/vda1	25G	1.4G	23G	7%	/
tmpfs	491M	0	491M	0%	/dev/shm
tmpfs	5.0M	0	5.0M	0%	/run/lock
tmpfs	491M	0	491M	0%	/sys/fs/cgroup
/dev/vda15	105M	3.9M	101M	4%	/boot/efi
/dev/loop0	55M	55M	0	100%	/snap/core18/1705
/dev/loop1	69M	69M	0	100%	/snap/lxd/14804
/dev/loop2	28M	28M	0	100%	/snap/snapd/7264
tmpfs	99M	0	99M	0%	/run/user/1000

O dispositivo com `/` na coluna `Mounted on` é o nosso disco neste caso. Temos bastante espaço disponível neste exemplo (apenas 1,4 GB usado). Seu uso provavelmente será diferente.

Apesar da divergência de opiniões quanto ao tamanho adequado de um espaço de swap, isso realmente dependerá de suas preferências pessoais e das exigências da sua aplicação. Geralmente, um espaço igual ou duas vezes o tamanho do espaço da RAM no seu sistema é um

bom ponto de partida. Outra boa regra de ouro é que qualquer coisa acima de 4 GB de swap é provavelmente desnecessária se você somente estiver usando-o como uma alternativa para a RAM.

Passo 3 – Criando um Arquivo de Swap

Agora que sabemos qual é o espaço disponível em nosso disco rígido, podemos criar um arquivo de swap no nosso sistema de arquivos. Alocaremos um arquivo do tamanho que queremos que o swap tenha chamado de `swapfile` em nosso diretório raiz (`/`).

A melhor maneira de criar um arquivo de swap é com o programa `fallocate`. Este comando cria instantaneamente um arquivo do tamanho especificado.

Uma vez que o servidor no nosso exemplo tem 1 GB de RAM, criaremos um arquivo de 1 GB neste guia. Ajuste isso para atender às necessidades do seu próprio servidor:

```
1. sudo fallocate -l 1G /swapfile
```

Podemos verificar se a quantidade correta de espaço foi reservada digitando:

```
1. ls -lh /swapfile
```

```
1. -rw-r--r-- 1 root root 1.0G Apr 25 11:14 /swapfile
```

Nosso arquivo foi criado com a quantidade correta do espaço reservado.

Passo 4 – Habilitando o Arquivo de

Swap

Agora que temos um arquivo do tamanho correto disponível, precisamos realmente transformar isso em espaço de swap.

Primeiro, precisamos bloquear as permissões do arquivo para que apenas os usuários com privilégios **root** possam ler o conteúdo. Isso impede que os usuários normais possam acessar o arquivo, o que teria implicações de segurança significativas.

Torne o arquivo acessível somente para **root** digitando:

```
1. sudo chmod 600 /swapfile
```

Verifique a alteração de permissões digitando:

```
1. ls -lh /swapfile
```

Output

```
-rw----- 1 root root 1.0G Apr 25 11:14 /swapfile
```

Como você pode ver, apenas o usuário **root** tem os sinalizadores de leitura e gravação habilitados.

Podemos agora marcar o arquivo como espaço de swap digitando:

```
1. sudo mkswap /swapfile
```

Output

```
Setting up swapspace version 1, size = 1024 MiB (1073737728 bytes)
no label, UUID=6e965805-2ab9-450f-aed6-577e74089dbf
```

Após marcar o arquivo, podemos habilitar o arquivo de swap, permitindo que nosso sistema comece a utilizá-lo:

```
1. sudo swapon /swapfile
```

Verifique se o swap está disponível digitando:

```
1. sudo swapon --show
```

Output

```
NAME      TYPE  SIZE USED PRIO
/swapfile file 1024M  0B   -2
```

Podemos verificar a saída do utilitário `free` novamente para corroborar nossos resultados:

```
1. free -h
```

Output

	total	used	free	shared	buff/cache	available
Mem:	981Mi	123Mi	644Mi	0.0Ki	213Mi	714Mi
Swap:	1.0Gi	0B	1.0Gi			

Nosso swap foi configurado com sucesso e nosso sistema operacional começará a usá-lo conforme necessário.

Passo 5 – Tornando o Arquivo de Swap Permanente

Nossas alterações recentes habilitaram o arquivo de swap para a sessão atual. No entanto, se reiniciarmos, o servidor não manterá as configurações de swap automaticamente. Podemos alterar isso adicionando o arquivo de swap ao nosso arquivo `/etc/fstab`.

Faça um backup do arquivo `/etc/fstab` para o caso de algo dar errado:

```
1. sudo cp /etc/fstab /etc/fstab.bak
```

Adicione a informação do arquivo de swap no final do seu arquivo `/etc/fstab` digitando:

```
1. echo '/swapfile none swap sw 0 0' | sudo tee -a /etc/fstab
```

Em seguida, avaliaremos algumas configurações que podemos atualizar para ajustar nosso espaço de swap.

Passo 6 – Ajustando as Configurações de Swap

Há algumas opções que você pode configurar que terão um impacto no desempenho do seu sistema quando estiver lidando com o swap.

Ajustando a propriedade Swappiness

O parâmetro `swappiness` configura a frequência com que o seu sistema transfere dados da RAM para o espaço de swap. Esse é um valor entre 0 e 100 que representa uma porcentagem.

Com valores próximos de zero, o kernel não irá transferir dados para o disco a menos que seja absolutamente necessário. Lembre-se, as interações com o arquivo de swap são “dispendiosas”, no sentido de que demoram mais que as interações com a RAM e podem causar uma redução significativa no desempenho. Dizer ao sistema para não depender tanto do swap irá geralmente tornar o seu sistema mais rápido.

Valores que estão mais próximos de 100 irão tentar colocar mais dados no swap em um esforço para manter mais espaço da RAM livre. Dependendo do perfil de memória de seus aplicativos ou do motivo pelo qual você está usando o seu servidor, isso pode ser melhor em alguns casos.

Podemos ver o valor atual do parâmetro `swappiness` digitando:

```
1. cat /proc/sys/vm/swappiness
```

Output

```
60
```

Para um desktop, um valor de `swappiness` de 60 não é um valor ruim. Para um servidor, você pode deixá-lo mais próximo de 0.

Podemos definir o parâmetro `swappiness` para um valor diferente usando o comando `sysctl`.

Por exemplo, para definir o valor do parâmetro `swappiness` em 10, poderíamos digitar:

```
1. sudo sysctl vm.swappiness=10
```

Output

```
vm.swappiness = 10
```

Este valor persistirá até a próxima reinicialização. Podemos definir este valor automaticamente na reinicialização, adicionando a linha no nosso arquivo `/etc/sysctl.conf`:

```
1. sudo nano /etc/sysctl.conf
```

No final, você pode adicionar:

`/etc/sysctl.conf`

```
vm.swappiness=10
```

Salve e feche o arquivo quando você terminar.

Ajustando a Configuração da Pressão por Cache

Outro valor relacionado que você pode querer modificar é o `vfs_cache_pressure`. Este ajuste configura o quanto o sistema escolherá para as informações cache dos objetos *inode* e *dentry* em detrimento de outros dados.

Basicamente, tratam-se de dados de acesso sobre o sistema de arquivos. De maneira geral, isso é difícil de consultar e, com frequência, muito solicitado. Assim, é algo muito bom que o seu sistema armazene dados em cache. Você pode ver o valor atual questionando o sistema de arquivos `proc` novamente:

```
1. cat /proc/sys/vm/vfs_cache_pressure
```

Output

```
100
```

Uma vez que ele está atualmente configurado, o nosso sistema remove as informações de inode do cache muito rapidamente. Podemos definir isso em um valor mais conservador como 50, digitando:

```
1. sudo sysctl vm.vfs_cache_pressure=50
```

Output

```
vm.vfs_cache_pressure = 50
```

Novamente, isso é apenas válido para a nossa sessão atual. Podemos alterar esse valor, adicionando-o ao nosso arquivo de configuração como fizemos com a nossa configuração do parâmetro swappiness:

```
1. sudo nano /etc/sysctl.conf
```

No final, adicione a linha que especifica o seu novo valor:

/etc/sysctl.conf

```
vm.vfs_cache_pressure=50
```

Salve e feche o arquivo quando você terminar.

Conclusão

Seguir as etapas deste guia lhe dará algum espaço para respirar em casos que de outra forma conduziram a exceções de falta de memória. O espaço de swap pode ser incrivelmente útil para evitar alguns desses problemas comuns.

Se você está encontrando erros de OOM (out of memory - falta de memória), ou se você descobrir que o seu sistema não consegue usar os aplicativos de que você precisa, a melhor solução é otimizar as configurações do seu aplicativo ou atualizar o seu servidor.

