

Increase swap

[Link](#)

Step 1 – Checking the System for Swap Information

Before we begin, we can check if the system already has some swap space available. It is possible to have multiple swap files or swap partitions, but generally one should be enough.

We can see if the system has any configured swap by typing:

```
sudo swapon --show
```

If you don't get back any output, this means your system does not have swap space available currently.

You can verify that there is no active swap using the `free` utility:

```
free -h
```

Step 2 – Checking Available Space on the Hard Drive Partition

Before we create our swap file, we'll check our current disk usage to make sure we have enough space. Do this by entering:

```
df -h
```

Output

Filesystem	Size	Used	Avail	Use%	Mounted on
udev	488M	0	488M	0%	/dev
tmpfs	100M	4.5M	96M	5%	/run

/dev/vda1	25G	989M	23G	5%	/
tmpfs	499M	0	499M	0%	/dev/shm
tmpfs	5.0M	0	5.0M	0%	/run/lock
tmpfs	499M	0	499M	0%	/sys/fs/cgroup
tmpfs	100M	0	100M	0%	/run/user/1001

The device with `/` in the `Mounted on` column is our disk in this case. We have plenty of space available in this example (only 1.4G used). Your usage will probably be different.

Although there are many opinions about the appropriate size of a swap space, it really depends on your personal preferences and your application requirements. Generally, an amount equal to or double the amount of RAM on your system is a good starting point. Another good rule of thumb is that anything over 4G of swap is probably unnecessary if you are just using it as a RAM fallback.

Step 3 – Creating a Swap File

Now that we know our available hard drive space, we can create a swap file on our filesystem. We will allocate a file of the swap size that we want called `swapfile` in our root (`/`) directory.

The best way of creating a swap file is with the `fallocate` program. This command instantly creates a file of the specified size.

Since the server in our example has 1G of RAM, we will create a 1G file in this guide. Adjust this to meet the needs of your own server:

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

We can verify that the correct amount of space was reserved by typing:

```
ls -lh /swapfile
```

Output

```
-rw-r--r-- 1 root root 1.0G May 29 17:34 /swapfile
```

Our file has been created with the correct amount of space set aside.

Step 4 – Enabling the Swap File

Now that we have a file of the correct size available, we need to actually turn this into swap space.

First, we need to lock down the permissions of the file so that only the users with **root** privileges can read the contents. This prevents normal users from being able to access the file, which would

have significant security implications.

Make the file only accessible to **root** by typing:

```
sudo chmod 600 /swapfile
```

Verify the permissions change by typing:

```
ls -lh /swapfile
```

Output

```
-rw----- 1 root root 1.0G May 29 17:34 /swapfile
```

As you can see, only the **root** user has the read and write flags enabled.

We can now mark the file as swap space by typing:

```
sudo mkswap /swapfile
```

Output

```
Setting up swappiness version 1, size = 1024 MiB (1073737728 bytes)
no label, UUID=b591444e-c12b-45a6-90fc-e8b24c67c006f
```

After marking the file, we can enable the swap file, allowing our system to start using it:

```
sudo swapon /swapfile
```

Verify that the swap is available by typing:

```
sudo swapon --show
```

Output

NAME	TYPE	SIZE	USED	PRI0
/swapfile	file	1024M	0B	-2

We can check the output of the `free` utility again to corroborate our findings:

```
free -h
```

Output

	total	used	free	shared	buff/cache	available
Mem:	990Mi	37Mi	860Mi	4.0Mi	92Mi	834Mi

Swap:	1.0Gi	0B	1.0Gi
-------	-------	----	-------

Our swap has been set up successfully and our operating system will begin to use it as necessary.

Step 5 – Making the Swap File Permanent

Our recent changes have enabled the swap file for the current session. However, if we reboot, the server will not retain the swap settings automatically. We can change this by adding the swap file to our `/etc/fstab` file.

Back up the `/etc/fstab` file in case anything goes wrong:

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

Add the swap file information to the end of your `/etc/fstab` file by typing:

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

Next we'll review some settings we can update to tune our swap space.

Revision #1

Created 29 April 2022 17:35:26 by Admin

Updated 29 April 2022 17:42:44 by Admin