

SwapFaq - Community Help Wiki

 help.ubuntu.com/community/SwapFaq

Introduction

- **This FAQ is aimed at Linux novices.**
- People always wonder how much swap they should create at install time, or after installing they may think, "have I made a large enough swap? Should I reinstall with a larger swap?"
- This FAQ will tell you how much swap you need and how to add more swap after installation.
- You will be given very simple answers (to prevent losing too much time reading this FAQ) and some explanations that may help you form your own opinion.

What is swap?

Swap space is the area on a hard disk. It is a part of your machine's Virtual Memory, which is a combination of accessible physical memory (RAM) and the swap space. Swap holds memory pages that are temporarily inactive. Swap space is used when your operating system decides that it needs physical memory for active processes and the amount of available (unused) physical memory is insufficient. When this happens, inactive pages from the physical memory are then moved into the swap space, freeing up that physical memory for other uses. Note that the access time for swap is slower, depending on the speed of the hard drive. Do not consider it to be a complete replacement for the physical memory. Swap space can be a dedicated swap partition (recommended), a swap file, or a combination of swap partitions and swap file(s).

Why do I need swap?

Memory consuming programs Sometimes, a large program (like LibreOffice, Neverwinter Nights, or a video editor) make the entire system need extra memory. A significant number of the pages used by these large programs during its startup may only be used for initialization and then never used again. The system can swap out those pages and free the memory for other programs or even for the disk cache. In these cases, swap will be used to help the system handle any extra load.

Hibernation (suspend-to-disk) The hibernation feature (suspend-to-disk) writes out the contents of RAM to the swap partition before turning off the machine. Therefore, your swap partition should be at least as big as your RAM size. Although the latest versions of Ubuntu

don't support hibernation OOTB you may configure your system to allow Hibernation. In both alternatives (PM-UTILS or SYSTEMD) you may use a partition or a file.

Unforeseeable Circumstances Unforeseeable events can and will happen (a program going crazy, some action needing much more space than you thought, or any other unpredictable combination of events). In these cases, swap can give you an extra delay to figure out what happened, or to finish what you are working on.

Optimizing memory usage Since mechanical hard drives are considerably slower than RAM (SSD - Solid State Drive - storage is not as slow as physical drives, but still slower than RAM), when you need a file (be it a data file like a video, executables like Firefox, or libraries), the Linux kernel reads the file into RAM and keeps it there, so that the next time you need it, it is already in RAM and data access is much faster. The portions of RAM that accelerate disk read are called "cached memory." You will notice that they make a huge difference in terms of responsiveness. The Linux kernel automatically moves RAM reserved by programs--but not really used--into swap, so that it can serve the better purpose of extending cached memory.

Optimizing Swap performance Because swap space uses a disk device, this can cause performance issues in any system that uses swap space significantly because the system itself may also be using the same disk device at the same time that it is required for swap operations. One way to reduce this problem is to have swap space on a different physical drive so that the competition for that resource is either reduced or eliminated.

How much swap do I need?

For less than 1GB of physical memory (RAM), it's highly recommended that the swap space should, as a base minimum, be equal to the amount of RAM. Also, it's recommended that the swap space is maximum twice the amount of RAM depending upon the amount of hard disk space available for the system because of diminishing returns.

For more modern systems (>1GB), your swap space should be at a minimum be equal to your physical memory (RAM) size "if you use hibernation", otherwise you need a minimum of $\text{round}(\sqrt{\text{RAM}})$ and a maximum of twice the amount of RAM. The only downside to having more swap space than you will actually use, is the disk space you will be reserving for it.

The "diminishing returns" means that if you need more swap space than twice your RAM size, you'd better add more RAM as Hard Disk Drive (HDD) access is about 10^3 slower than RAM access, so something that would take 1 second, suddenly takes more than 15 minutes! And still more than a minute on a fast Solid State Drive (SSD)...

Example Scenarios

(last 3 columns denote swap space)

RAM	No hibernation	With Hibernation	Maximum
256MB	256MB	512MB	512MB
512MB	512MB	1024MB	1024MB
1024MB	1024MB	2048MB	2048MB

RAM	No hibernation	With Hibernation	Maximum
1GB	1GB	2GB	2GB
2GB	1GB	3GB	4GB
3GB	2GB	5GB	6GB
4GB	2GB	6GB	8GB
5GB	2GB	7GB	10GB
6GB	2GB	8GB	12GB
8GB	3GB	11GB	16GB
12GB	3GB	15GB	24GB
16GB	4GB	20GB	32GB
24GB	5GB	29GB	48GB
32GB	6GB	38GB	64GB
64GB	8GB	72GB	128GB
128GB	11GB	139GB	256GB
256GB	16GB	272GB	512GB
512GB	23GB	535GB	1TB
1TB	32GB	1056GB	2TB
2TB	46GB	2094GB	4TB
4TB	64GB	4160GB	8TB
8TB	91GB	8283GB	16TB

How do I add more swap?

Swap is generally associated with a swap partition, perhaps because the user is prompted to create a swap partition at the time of installation. In fact, any file can be used as a swapping device, be it a partition or a conventional file. Swap can be added by increasing the size of the swap partition or by adding a swap file. Keep in mind that when creating a swap file that it may not necessarily be using contiguous disk blocks (as a swap partition will), and this could have a negative impact on performance as disk access times may be longer, and the more your system uses swap, the worse it will be. The Linux kernel also accesses swap disk block IO directly bypassing all caching, metadata and filesystem code, so a swap file should have no ill effect on the stability of your base filesystem. Since kernel 2.6.29 the swap system has automatically supported TRIM capable devices like SSDs.

The advantages of a swap file are many, but it is problematic for using the default `swsusp` hibernation method for powerless sleep. The best solution for maintaining hibernate capability after adding RAM is to increase the size of the swap partition.

How do I add or modify a swap partition?

Process to Increase Size of Swap Partition and use it for Hibernation

- Creating the swap partition
- Activating the swap partition
- Making the new swap partition work for hibernate (optional)

Creating the swap partition

1. Boot to Ubuntu install CD and choose the option to run Ubuntu now
2. Go to system -> GParted Partition Editor
3. Delete the swap partition and, if there is nothing else in it, the extended partition that holds it. (If by some miracle you're able to resize your swap partition from here, I imagine your life will be a lot easier than mine.)
4. Decrease the size of your primary partition by the amount you want your new swap to be (I made mine 2x RAM + 500MB just to be safe). The easiest way to do this is to fill in the amount of space you want swap to be in the "free space following" field
5. In the free space that has now been created, choose new, type linux-swap and you can name the partition "swap" if you like
6. Hit the *Apply* button (should be a check mark) to write the changes to disk
7. When done, reboot back into Ubuntu

Activating the swap partition

(If your swap is on your primary hard drive, you don't need to do anything here.) Now you need to find what partition your swap is on and what its UUID is. UUID?! you say? Well that's the Universally Unique IDentifier for the partition so you can reference it even if it's on a different mount point from boot-to-boot due to adding disks, etc.

1. Pull up a terminal and run `gksu gparted &` and enter your root password. The & lets this process run while still giving you access to the command line.
2. Right-click on your swap partition and choose *Information*. You should see the ****Path**** and ****UUID**** listed there. Keep this open for further reference.

3. Run `gksu gedit /etc/fstab` & and look for the line that has `*swap*` in it. It should be the third column, separated by spaces or tabs. You can either use the path or the UUID to tell Linux where to find your swap partition. I recommend UUID because it'll stay constant even if you move the partition around or the disk somehow becomes `sdb` instead of `sda` or something like that. Make the appropriate edits and save the file. Your line should look something like this if you used UUID (with your UUID instead, of course):

- `UUID=41e86209-3802-424b-9a9d-d7683142dab7 none swap sw 0 0`
- or this if you used path: `/dev/sda2 none swap sw 0 0`

4. Save the file.
5. Enable the new swap partition with this command.

```
sudo swapon --all
```

OR

```
$ sudo swapon --all --verbose
swapon on /dev/sda2
swapon: /dev/sda2: found swap signature: version 1, page-size 4, same byte order
swapon: /dev/sda2: pagesize=4096, swaptsize=2147483648, devsize=2147483648
```

6. Confirm that the swap partition exists.

```
$ cat /proc/swaps
Filename                                Type      Size    Used    Priority
/dev/sda2                              partition 2097148 0       -1
```

7. Reboot to make sure the new swap gets activated properly at startup

Making the swap partition work for hibernate (optional)

'INFO: This will not work for 12.04, resume from hibernate work differently in 12.04.'

1. Pull up a Terminal again and run `cat /proc/swaps` and hopefully you see the path to your swap partition listed there. If not chances are something went wrong in the steps above. Here's my output:

```
Filename                                Type      Size    Used    Priority
/dev/sda2                              partition 2676732 73380   -1
```

1. `gksu gedit /etc/default/grub` & to pull up the boot loader configuration

2. Look for the line `GRUB_CMDLINE_LINUX=""` and make sure it looks like this (using your UUID of course) `GRUB_CMDLINE_LINUX="resume=UUID=41e86209-3802-424b-9a9d-d7683142dab7"` and save the file
3. `sudo update-grub` and wait for it to finish
4. `gksu gedit /etc/initramfs-tools/conf.d/resume` & and make sure its contents are `resume=UUID=41e86209-3802-424b-9a9d-d7683142dab7` (with your UUID of course in place of mine). Save the file!
5. `sudo update-initramfs -u`
6. Reboot!

Now you should be able to hibernate and resume!

How do I add a swap file?

Note: btrfs does not support swap files at the moment. See [man swapon](#), and [btrfs Faq](#)

Create the Swap File:

We will create a 1 GiB file (`/mnt/1GiB.swap`) to use as swap:

```
sudo fallocate -l 1g /mnt/1GiB.swap
```

`fallocate` size suffixes: g = Giga, m = Mega, etc. (See `man fallocate`).

If `fallocate` fails or it not available, you can use `dd`:

```
sudo dd if=/dev/zero of=/mnt/1GiB.swap bs=1024 count=1048576
```

We need to set the swap file permissions to `600` to prevent other users from being able to read potentially sensitive information from the swap file.

```
sudo chmod 600 /mnt/1GiB.swap
```

Format the file as swap:

```
sudo mkswap /mnt/1GiB.swap
```

Enable use of Swap File

```
sudo swapon /mnt/1GiB.swap
```

The additional swap is now available and verified with: `cat /proc/swaps`

Enable Swap File at Bootup

Add the swap file details to `/etc/fstab` so it will be available at bootup:

```
echo '/mnt/1GiB.swap swap swap defaults 0 0' | sudo tee -a /etc/fstab
```

Example of making a swap file

This is an example of making and using a swap file on a computer with no swap partition.

```
$ sudo fallocate -l 1g /mnt/1GiB.swap
$ sudo chmod 600 /mnt/1GiB.swap
$ sudo mkswap /mnt/1GiB.swap
Setting up swspace version 1, size = 1048576 kB
$ sudo swapon /mnt/1GiB.swap
$ cat /proc/swaps
Filename                                Type              Size    Used    Priority
/home/swapfile                         file              1048576 1048576 -1
$ echo '/mnt/4GiB.swap swap swap defaults 0 0' | sudo tee -a /etc/fstab
$ reboot
$ free -h
```

	total	used	free	shared	buff/cache	available
Mem:	15G	9.3G	454M	4.0G	5.8G	1.9G
Swap:	1.0G	1.0G	0B			

Disable and Remove a Swap File

Disable the swap file from the running system and then delete it:

```
sudo swapoff /mnt/1Gib.swap
sudo rm /mnt/1Gib.swap
```

Remove the swap file details from `fstab`:

```
gksudo gedit /etc/fstab
```

Removing the swap file line

```
/mnt/1GiB.swap swap swap defaults 0 0
```

What is swappiness and how do I change it?

The swappiness parameter controls the tendency of the kernel to move processes out of physical memory and onto the swap disk. Because disks are much slower than RAM, this can lead to slower response times for system and applications if processes are too

aggressively moved out of memory.

- swappiness can have a value of between 0 and 100
- swappiness=0 tells the kernel to avoid swapping processes out of physical memory for as long as possible
- swappiness=100 tells the kernel to aggressively swap processes out of physical memory and move them to swap cache

The default setting in Ubuntu is swappiness=60. Reducing the default value of swappiness will probably improve overall performance for a typical Ubuntu desktop installation. A value of swappiness=10 is recommended, but feel free to experiment. **Note:** Ubuntu server installations have different performance requirements to desktop systems, and the default value of 60 is likely more suitable.

To check the swappiness value

```
cat /proc/sys/vm/swappiness
```

To change the swappiness value A temporary change (lost on reboot) with a swappiness value of 10 can be made with

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

To make a change permanent, edit the configuration file with your favorite editor:

```
gksudo gedit /etc/sysctl.conf
```

Search for *vm.swappiness* and change its value as desired. If *vm.swappiness* does not exist, add it to the end of the file like so:

```
vm.swappiness=10
```

Save the file and reboot.

What is the priority of swap containers?

The Linux kernel assigns priorities to all swap containers. To see the priorities that the Linux Kernel assigns to all the swap containers use this command.

```
cat /proc/swaps
```

Priorities can be changed by using the swapon command or defined in /etc/fstab. Consult the manual page of swapon for more info

```
man swapon
```


Should I reinstall with more swap?

Definitely not. With the 2.6 kernel, "a swap file is just as fast as a swap partition."
([Wikipedia:Paging](#), [LKML](#)).

Why is my swap not being used?

My swap is not being used! When I issue the free command, it shows something like this:

```
tom@tom:~$ free
              total    used    free   shared  buffers   cached
Mem:      515980    448664    67316      0    17872    246348
-/+ buffers/cache:    184444    331536
Swap:      674688      0    674688
```

Note: This regards mainly swap on hard disk partitions, but it could help anyway. In these examples /dev/hda8 is considered as swap.

Swap may not be needed

Start many memory consuming applications (e.g. Gimp, web browsers, [LibreOffice](#) etc) and then issue the free command again. Is swap being used now?

Ubuntu Desktop uses Swap to Hibernate (PC off, no power needed, program states saved). If Hibernation is important to you, have more swap space than ram + swap overflow.

Is there a swap partition at all?

Use this command to see all partitions

```
sudo parted --list
```

You should be able to see something like this in the output

```
5    236GB  256GB  20.0GB  linux-swap(v1)
```

If not, you either need to [create a swapfile](#) or create a swap partition. To create a swap partition you can

- boot from your Ubuntu install CD, create a swap partition out of the free space on your hard disk and then interrupt your installation.
- use [Cfdisk](#).

Enabling a swap partition

In case you do have a swap partition, there are several ways of enabling it.

- Use the following command

```
cat /etc/fstab
```

- Ensure that there is a line link below. This enables swap on boot.

```
/dev/sdb5    none        swap    sw        0        0
```

- Then disable all swap, recreate it, then re-enable it with the following commands.

```
sudo swapoff -a
sudo /sbin/mkswap /dev/sdb5
sudo swapon -a
```

Empty Swap

Even if you have lots of RAM and even if you have a low swappiness value, it is possible that your computer swaps. This can hurt the multitasking performance of your desktop system.

You can use the following script to get the swap manually back into RAM:

- Place the script e.g. /usr/local/sbin:

```
gksudo gedit /usr/local/sbin/swap2ram.sh
```

- Copy-paste the script into the file:

```
#!/bin/sh

mem=$(LC_ALL=C free | awk 'Mem:/ {print $4}')
swap=$(LC_ALL=C free | awk '/Swap:/ {print $3}')

if [ $mem -lt $swap ]; then
    echo "ERROR: not enough RAM to write swap back, nothing done" >&2
    exit 1
fi

swapoff -a &&
swapon -a
```

- Save and close gedit
- Make the script executable:

```
sudo chmod +x /usr/local/sbin/swap2ram.sh
```

- Execute:

```
sudo /usr/local/sbin/swap2ram.sh
```

