**Practical Number : 8**

**AIM :** Memory Management

i). Use dmesg to find the total amount of memory in your computer.

ii). Use free to display memory usage in kilobytes (then in megabytes).

iii). On a virtual machine, create a swap partition (you might need an extra virtual disk for this).

iv). Add a 20 megabyte swap file to the system.

v). Put all swap spaces in /etc/fstab and activate them. Test with a reboot that they are mounted.

vi). Use free to verify usage of current swap.

**Theory :**

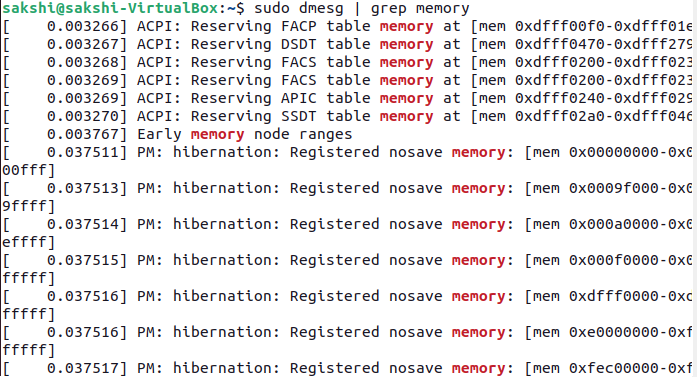
1. Linux memory management subsystem is responsible, as the name implies, for managing the memory in the system. This includes implementation of virtual memory and demand paging, memory allocation both for kernel internal structures and user space programs, mapping of files into processes address space and many other cool things.
2. In Linux, the physical memory is called memory. When the physical memory fills up, Linux intelligently moves the less frequently accessed data from the memory to a specific part of the disk (HDD or SSD). This part of the disk is called swap.
3. When there is no available free physical memory, some less frequently access data is moved to the swap. This frees up the physical memory and thus saves the system from crashing.

1.Use dmesg to find the total amount of memory in your computer :

dmesg command shows you the last status messages reported by your OS kernel, and since every boot procedure includes scanning the hardware and confirming the devices and resources recognized by the kernel, you can see some basic information by using dmesg.

Syntax: ~$ dmesg | grep Memory

For our purpose, we need to filter out the memory status:

****

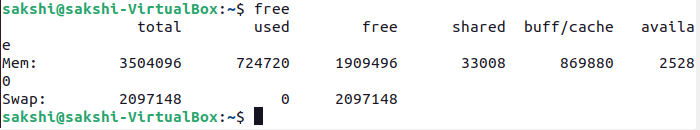
2. use to display memory usage in kilobytes (then in megabytes)

While using LINUX there might come a situation when you are willing to install a new application (big in size) and you wish to know for the amount of free memory available on your system. In LINUX, there exists a command line utility for this and that is free command which displays the total amount of free space available along with the amount of memory used and swap memory in the system, and also the buffers used by the kernel. free is a command that displays the total memory usage information of the system. free is shipped with almost all the Linux distribution by default.

syntax: ~$ free [OPTIONS]

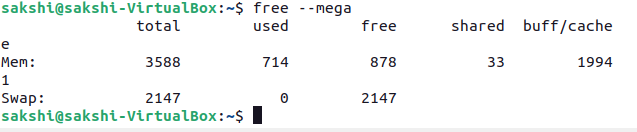
OPTIONS : refers to the options compatible with free command.

You can check memory usage with the free command as follows:



If we want to display the memory and swap usage information in megabytes, then run the free command with the –mega option as follows:

Syntax: $ free --mega



3. On a virtual machine, create a swap partition:

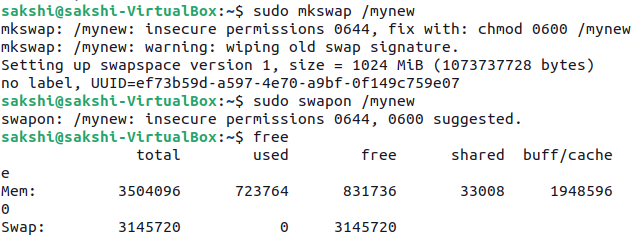
Linux provides for two types of swap space. By default, most Linux installations create a swap partition, but it is also possible to use a specially configured file as a swap file. A swap partition is just what its name implies—a standard disk partition that is designated as swap space by the mkswap command.

A swap file can be used if there is no free disk space in which to create a new swap partition or space in a volume group where a logical volume can be created for swap space. This is just a regular file that is created and preallocated to a specified size. Then the mkswap command is run to configure it as swap space. I don’t recommend using a file for swap space unless absolutely necessary.

Syntax: $ sudo dd if=/dev/zero of=/swapfile bs=1024 count=2097152;

$ sudo chmod 600 /mynew(file nmae)

$ sudo mkswap /(file\_name) mynew

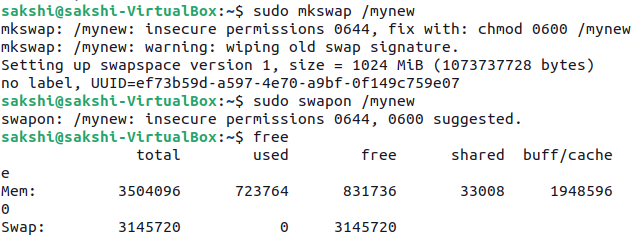


4. Add a 1024 megabyte swap file to the system :

Swap is a space on a disk that is used when the amount of physical RAM memory is full. When a Linux system runs out of RAM, inactive pages are moved from the RAM to the swap space.

Swap space can take the form of either a dedicated swap partition or a swap file. Typically, when running Ubuntu on a virtual machine, a swap partition is not present, and the only option is to create a swap file. Syntax: $ sudo swapon /(file name) mynew ---- to set file in swaping portion.

Syntax: $ sudo swapon /(file name) mynew ---- to set file in swaping portion.

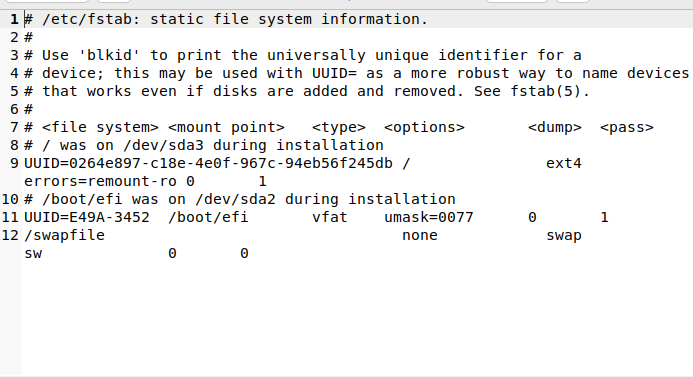


5. Put all swap spaces in /etc/fstab and activate them:

One of the easiest way of increasing the responsiveness of your server and guarding against out-of-memory errors in applications is to add some swap space.

To add all swap spaces in fstab we uesd following command: $ sudo gedit(compiler) /etc/fstab





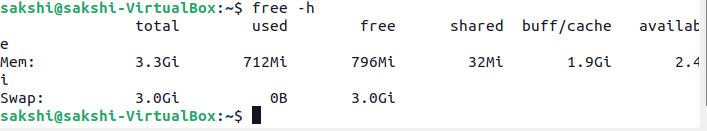
6.Use free to verify usage of current swap:

You can check the total, used and free swap space with the help of free command. The result will be displayed in kilobytes. The command looks like the following.

Syntax :

$ free

If you want to check the swap space in human-readable format with the closest possible unit, use the -h switch as follows.



**Conclusion:** I study about memory management, dmesg command, free command and file swapping in this practical.