Memory management commands

Memory management is crucial in operating systems as it handles the allocation, management, and optimization of the system's memory resources

1. Free:

- The `free` command displays the amount of free and used memory in the system, including physical and swap memory.
- It provides a snapshot of memory usage, showing total, used, free, shared, buffer/cache, and available memory.

#free (one time snapshot of memory)

#free –h (In human readable format)

#free –m (In Megabytes)

#free -g (In gigabytes)

#free –s <interval> (Continuously display memory usage updates at a specified interval (in seconds)

1. You are running a web server, and it's becoming unresponsive during high traffic. You suspect the server is running out of memory. How would you use the `free` command to confirm this?

->free –h

Output:

```
           total    used    free     shared    buff/cache    available

Mem:       16Gi     14Gi    200Mi    500Mi     2Gi           1Gi

Swap:      2Gi      1Gi     1Gi
```

Used memory (14Gi) is high, leaving only **200Mi free**.

Available memory (1Gi) is low, indicating that the system is nearing its memory capacity.

Swap (1Gi used) indicates that the system is already using some swap space, which can slow down the server as swapping is much slower than accessing RAM.

2. mkswap

The `mkswap` command is used to set up a swap area on a specified device or file.

Swap space is a portion of the hard drive used as virtual memory, allowing the system to handle more processes than it can fit in RAM alone.

```
->sudo mkswap /dev/sdX
->sudo mkswap /path/to/swapfile
```

You want to increase the swap space on your server because your application is consuming more memory than available. Describe how you would use the `mkswap` command to create a new swap file.

1. First, I would create a swap file using the dd command to allocate the required space (e.g., 2GB):

   ```
   ->sudo dd if=/dev/zero of=/swapfile bs=1M count=2048
   ```

2. Set the correct permissions for the swap file**:**

   ```
   ->sudo chmod 600 /swapfile
   ```

3. use the mkswap command to set up the swap file:

   ```
   ->sudo mkswap /swapfile
   ```

4. Finally, I would enable the swap file using:

   ```
   ->sudo swapon /swapfile
   ```

1. You have added a new disk to your server, and it has a partition /dev/sdb1 that you want to use for swap space instead of creating a swap file. How would you format this partition for swap and enable it?

```
->sudo mkswap /dev/sdb1
->sudo swapon /dev/sdb1
```

Verify that the swap partition is active

```
->swapon --show
->free –h
```

2. You are managing a busy production server that is running out of memory. The server already has 4 GB of swap, but it is completely used up. You need to add an additional 2 GB of swap without disrupting the server's operation.

Check existing swap:
->swapon --show
->free –h (Verify the current swap space is 4 GB and fully utilized)

Create a new 2 GB swap file:
->sudo dd if=/dev/zero of=/newswapfile bs=1M count=2048

sudo chmod 600 /newswapfile (set permission)

sudo mkswap /newswapfile (format the swap file)

sudo swapon /newswapfile (enable swap file)

3. swapon

The swapon command is used to enable devices and files for swapping, allowing the operating system to use them as swap space.

->sudo swapon /dev/sdX
->sudo swapon /path/to/swapfile

1. Your system is using an old 1 GB swap partition on /dev/sda3. You have added a new 4 GB swap partition on /dev/sdb2. How would you replace the old swap partition with the new one without affecting system performance?

Disable the old swap partition:

sudo swapoff /dev/sda3

Format the new partition /dev/sdb2 as swap:

sudo mkswap /dev/sdb2

Activate the new swap partition:

sudo swapon /dev/sdb2

2. You have two swap areas on your system, one on an SSD and another on an HDD. You want the system to use the SSD-based swap area first because it is faster. How would you adjust the priority of swap spaces using `mkswap` and other commands?

Check current swap areas:
->swapon –show

Set priority when activating the SSD swap space:
(For the SSD-based swap (assuming it's on /dev/nvme0n1p1))
->sudo swapon /dev/nvme0n1p1 --priority 100

Set a lower priority for the HDD swap space:
(For the HDD-based swap (assuming it's on /dev/sda2))
->sudo swapon /dev/sda2 --priority 10

4. vmstat

The vmstat (Virtual Memory Statistics) command in Linux is a powerful tool that provides information about processes, memory, paging, block I/O, traps, and CPU activity.

It helps system administrators monitor the system's performance and identify potential bottlenecks in real-time.

->vmstat

Output:

```
procs -----------memory---------- ---swap-- -----io---- -system-- ------cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 1  0      0  90400  30524 102524    0    0    20    10   50  102  2  1 95  2  0
```

procs:

- **r**: The number of processes waiting for CPU time (run queue).
- **b**: The number of processes in an uninterruptible sleep (typically I/O wait).

memory:

- **swpd**: Amount of swap memory in use.
- **free**: Amount of idle memory.
- **buff**: Memory used for buffers.
- **cache**: Memory used for page cache.

swap:

- **si**: Swap-in from disk (the amount of data being read from swap).
- **so**: Swap-out to disk (the amount of data being written to swap).

io:

- **bi**: Blocks received from a block device (read I/O).
- **bo**: Blocks sent to a block device (write I/O).

system:

- **in**: Number of interrupts per second.
- **cs**: Number of context switches per second.

cpu:

- **us**: Time spent running user processes.
- **sy**: Time spent running system (kernel) processes.
- **id**: Time spent idle.
- **wa**: Time spent waiting for I/O.
- **st**: Time stolen from a virtual machine (hypervisor-related

1. Your web server is running slowly, and users are complaining about performance issues. You run the vmstat command and observe high values in the us (user) and sy (system) columns under the CPU section. What does this indicate, and how would you investigate further?

```
->vmstat
```

High us value: Indicates that the CPU is spending a lot of time processing user applications.
High sy value: Indicates the system is spending significant time handling kernel tasks (e.g., I/O, system calls).

1. **You notice a high number of context switches (cs column) while running vmstat. What does this indicate, and how can you investigate further?**

```
->vmstat
```

A high number of context switches (cs) indicates that the CPU is frequently switching between processes. This can lead to performance degradation if excessive.

Check for many short-lived processes using top or ps.Frequent process creation and termination could be causing high context switches.

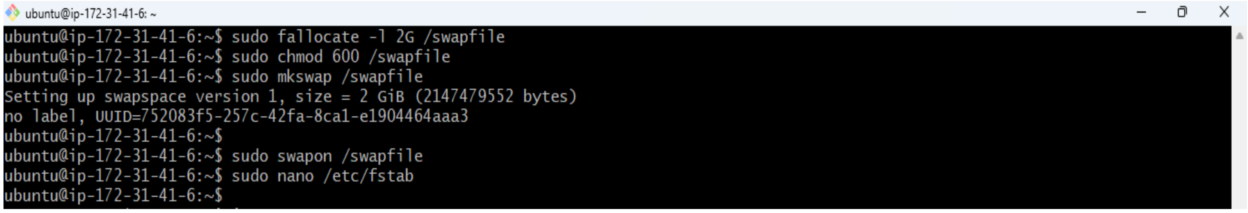
Use pidstat to monitor the number of context switches per process:

```
->pidstat -w 1
```

Assignment 3 - Memory Management Commands

1) Setup additional swap space in the system to solve low memory issue. The swap which you added should be available post reboot.

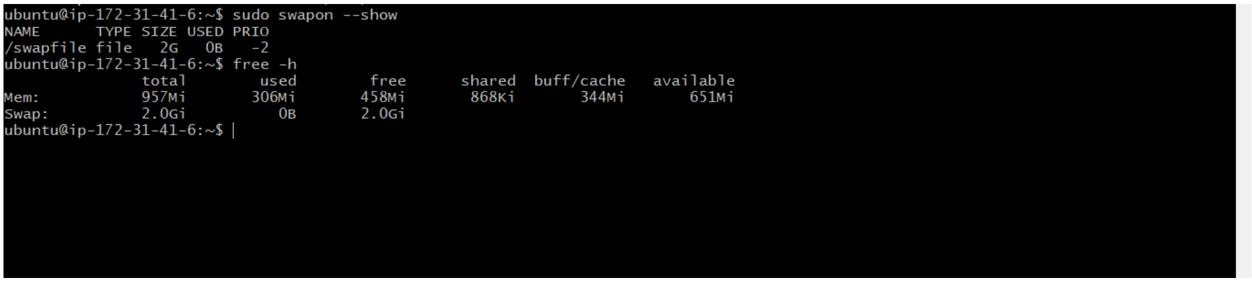
->sudo fallocate –l 2G /swapfile	(creating swap file)
->sudo chmod 600 /swapfile	(Giving permission of read, write to user on swap file)
->sudo mkswap /swapfile	(Formating swap file for swap space
->sudo swapon /swapfile	(Activating swap file)



```
ubuntu@ip-172-31-41-6:~$ sudo fallocate -l 2G /swapfile
ubuntu@ip-172-31-41-6:~$ sudo chmod 600 /swapfile
ubuntu@ip-172-31-41-6:~$ sudo mkswap /swapfile
Setting up swapspace version 1, size = 2 GiB (2147479552 bytes)
no label, UUID=752083f5-257c-42fa-8ca1-e1904464aaa3
ubuntu@ip-172-31-41-6:~$
ubuntu@ip-172-31-41-6:~$ sudo swapon /swapfile
ubuntu@ip-172-31-41-6:~$ sudo nano /etc/fstab
ubuntu@ip-172-31-41-6:~$
```

Verifying the swap space

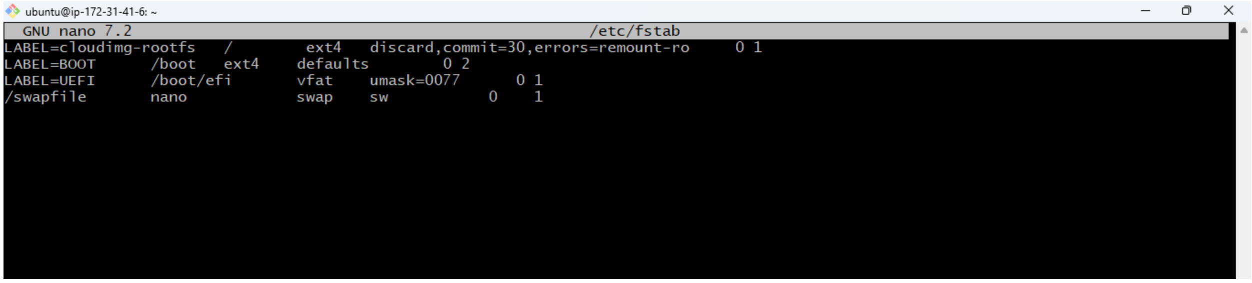
->free -h



```
ubuntu@ip-172-31-41-6:~$ sudo swapon --show
NAME      TYPE SIZE USED PRIO
/swapfile file   2G   0B   -2
ubuntu@ip-172-31-41-6:~$ free -h
               total        used        free      shared  buff/cache   available
Mem:           957Mi       306Mi       458Mi       868Ki       344Mi       651Mi
Swap:          2.0Gi          0B       2.0Gi
ubuntu@ip-172-31-41-6:~$ |
```

Making swap space available after reboot(permanent)

->/swapfile   none   swap   sw   0 0

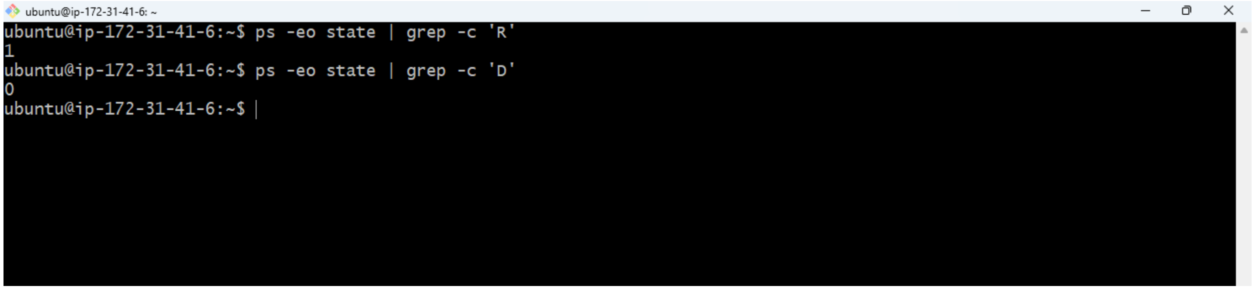


```
  GNU nano 7.2                                                    /etc/fstab
LABEL=cloudimg-rootfs   /        ext4   discard,commit=30,errors=remount-ro     0 1
LABEL=BOOT      /boot   ext4    defaults        0 2
LABEL=UEFI      /boot/efi       vfat    umask=0077      0 1
/swapfile       nano            swap    sw            0    1
```

2) Find out the number of process is in run queue and blocking queue

-> ps -eo state | grep -c 'R'   (Run queue)

-> ps -eo state | grep -c 'D'   (Blocking queue)



```
ubuntu@ip-172-31-41-6:~$ ps -eo state | grep -c 'R'
1
ubuntu@ip-172-31-41-6:~$ ps -eo state | grep -c 'D'
0
ubuntu@ip-172-31-41-6:~$ |
```