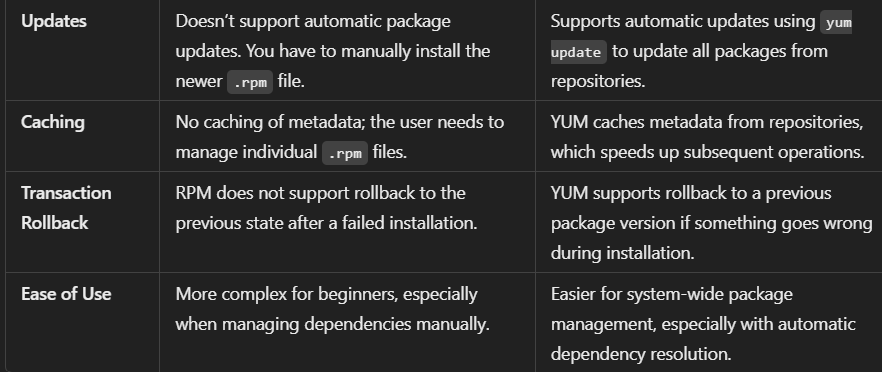
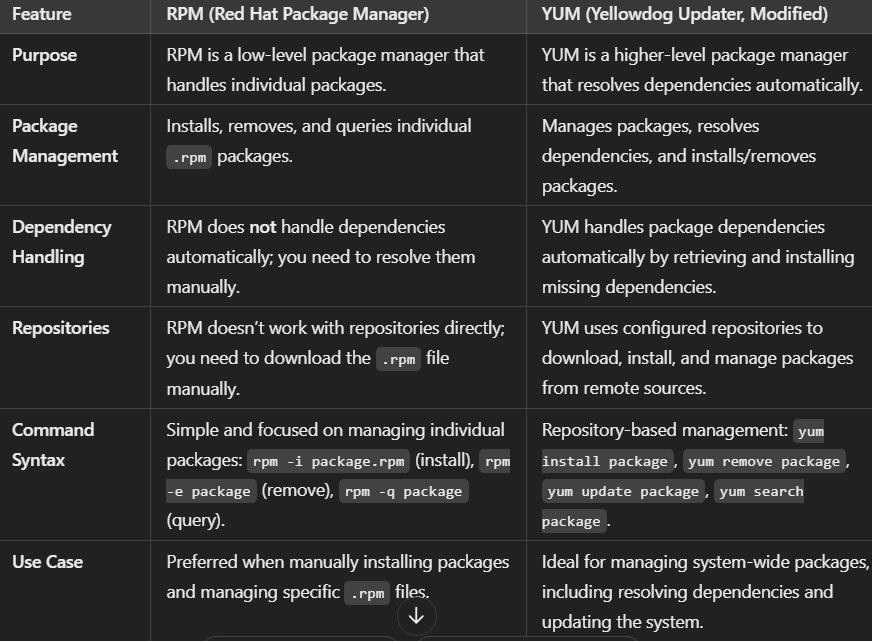
yum

1. configure local yum repository in Redhat Linux? repository file name should be local-data. Repo ?Also configure client repository using ftp ?

1. Difference between rpm and yum?



1. write command to install package?

This command are widely used to install packages

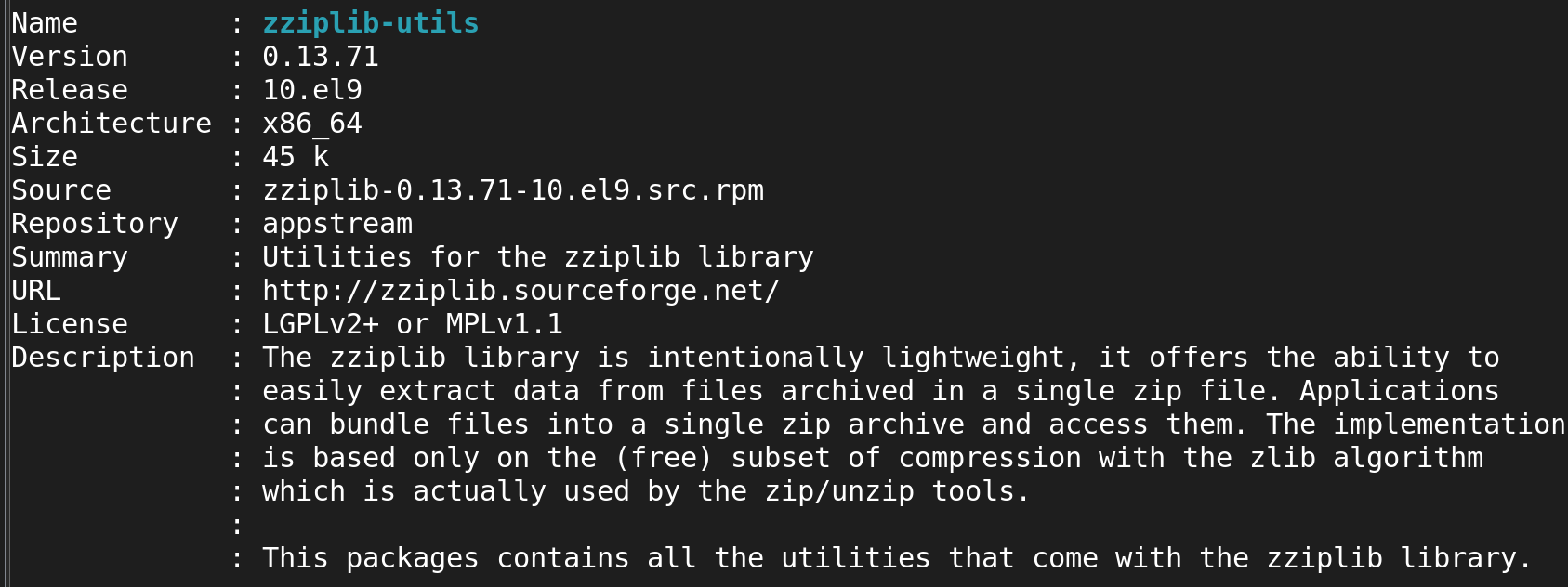
**# yum install package\_name**

**# rpm -i package\_name.rpm**

**# dnf install package\_name**

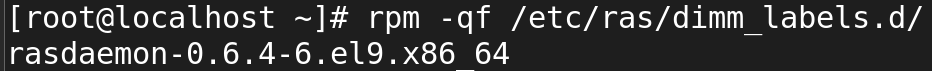
1. How to get the package information using yum?

yum info package\_name



1. How to find files belongs to which rpm package?

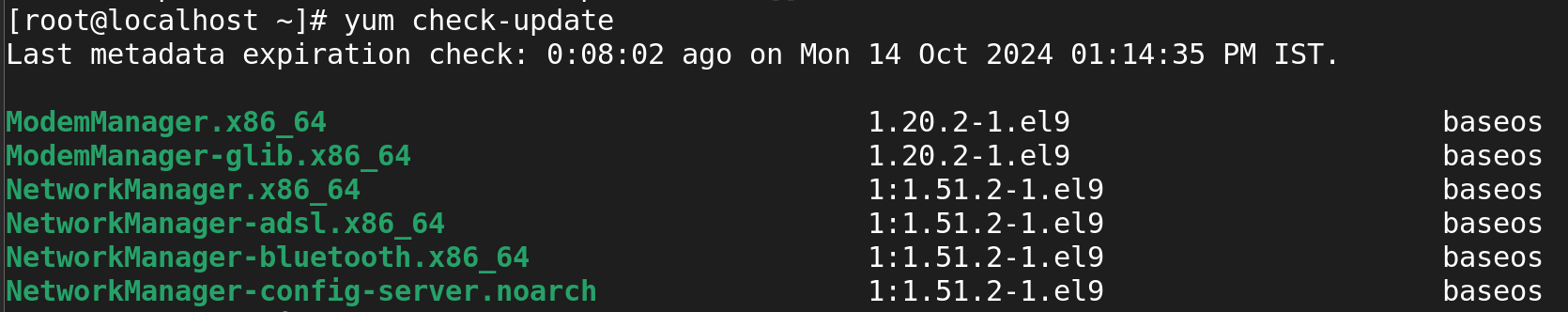
**# rpm -qf /path/to/file**



1. How to check the updates for yum repository?

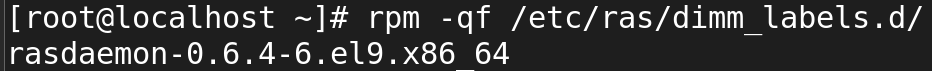
# yum update

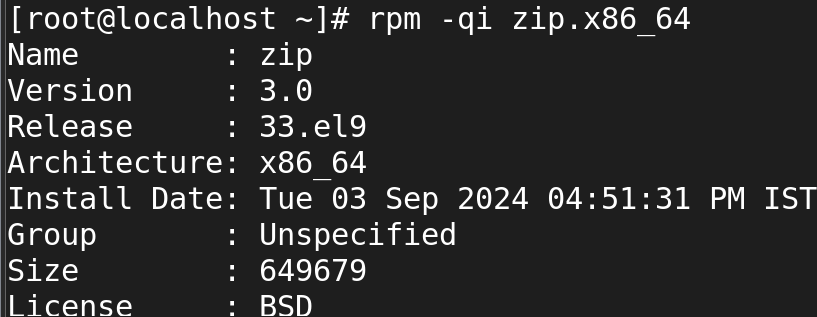
# yum update package\_name

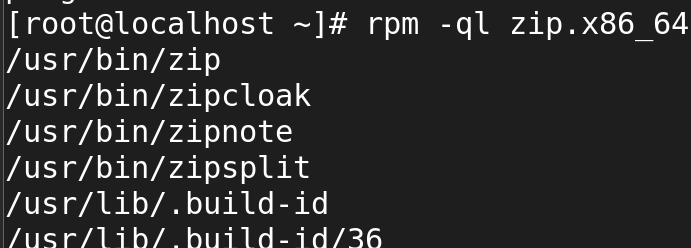


1. How to know the particular files belongs to which package?

**# rpm -qf /path/to/file**



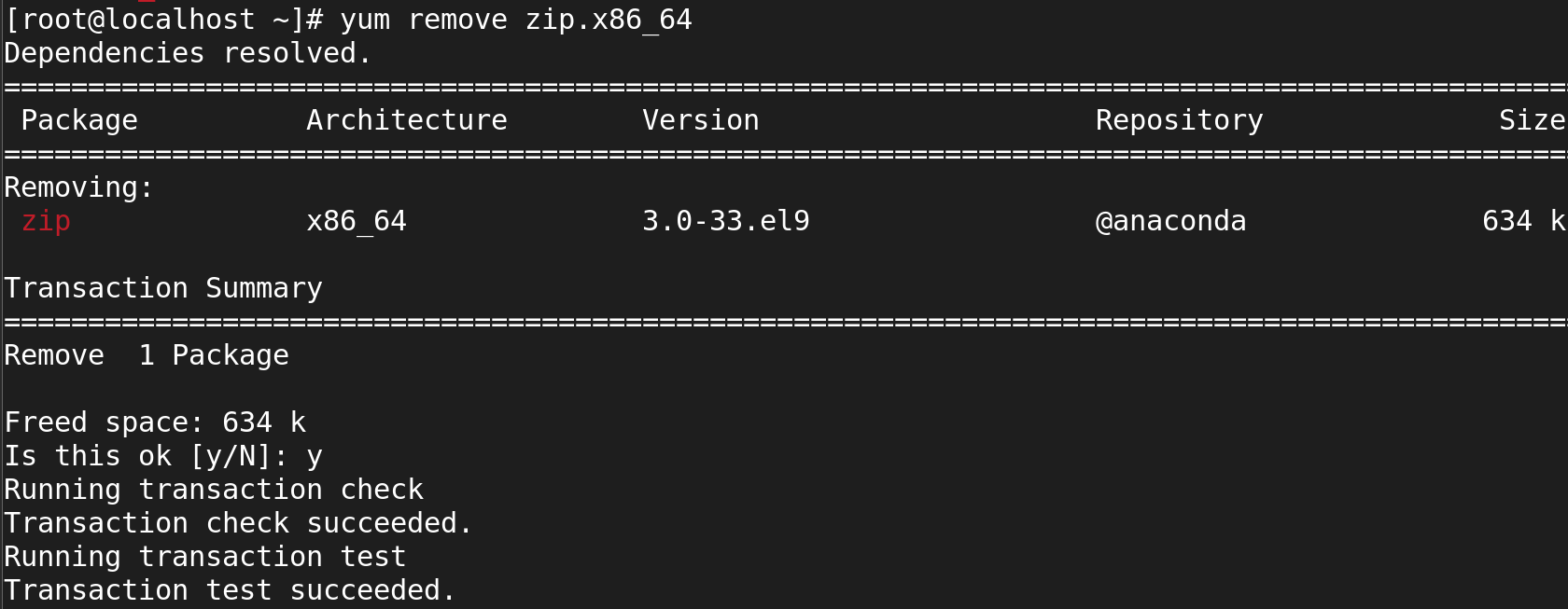




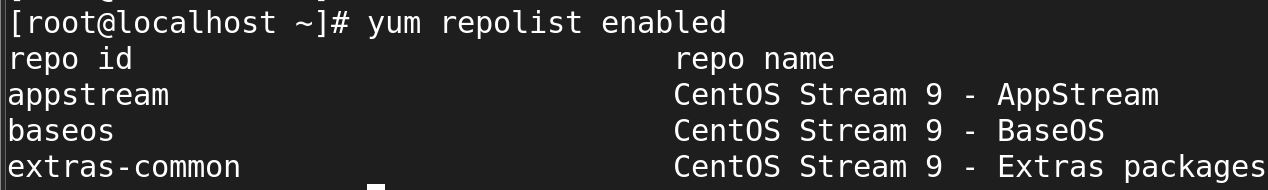
1. How to remove the package using yum?

**yum remove package\_name**

**yum remove package1 package2 package3**

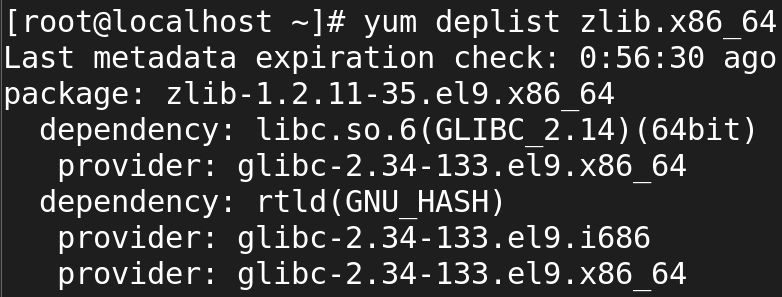


1. How to list the enabled repositories?



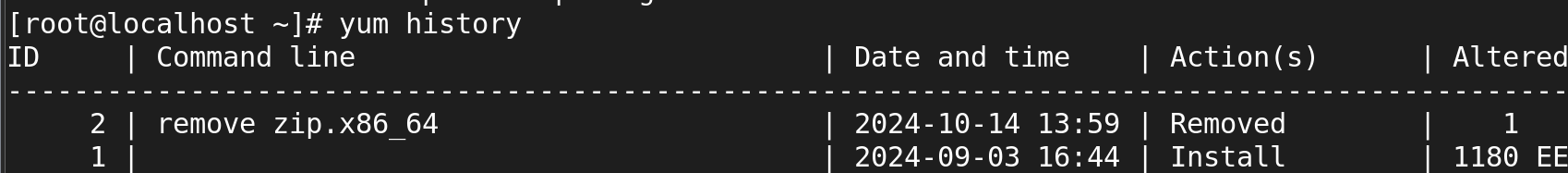
1. How to get the list of dependencies for specific package?

yum deplist package\_name



1. How to display a list of recently installed software?

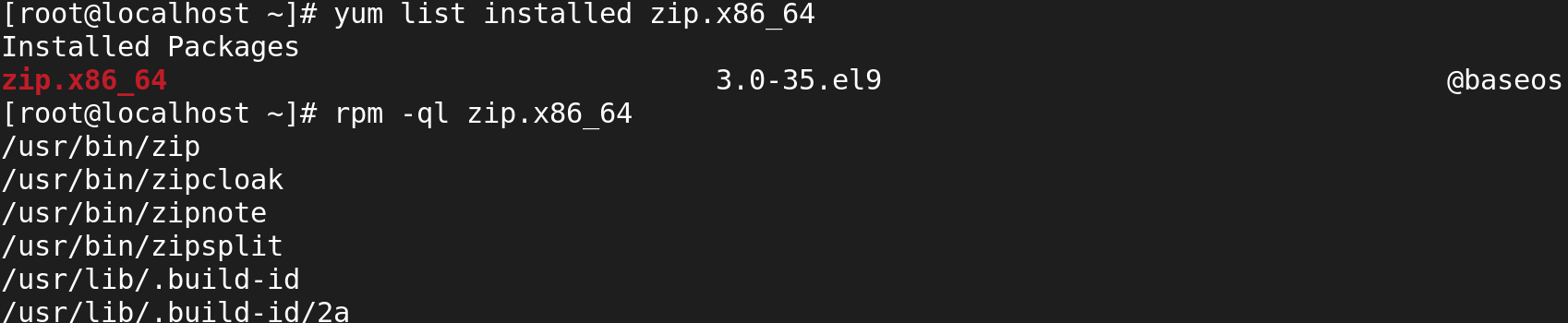
**# yum history**



1. How to display the list of installed software package’s files?

rpm -ql <package\_name>

yum list installed <package\_name>



1. How to install specific software package using rpm command on Redhat Linux?

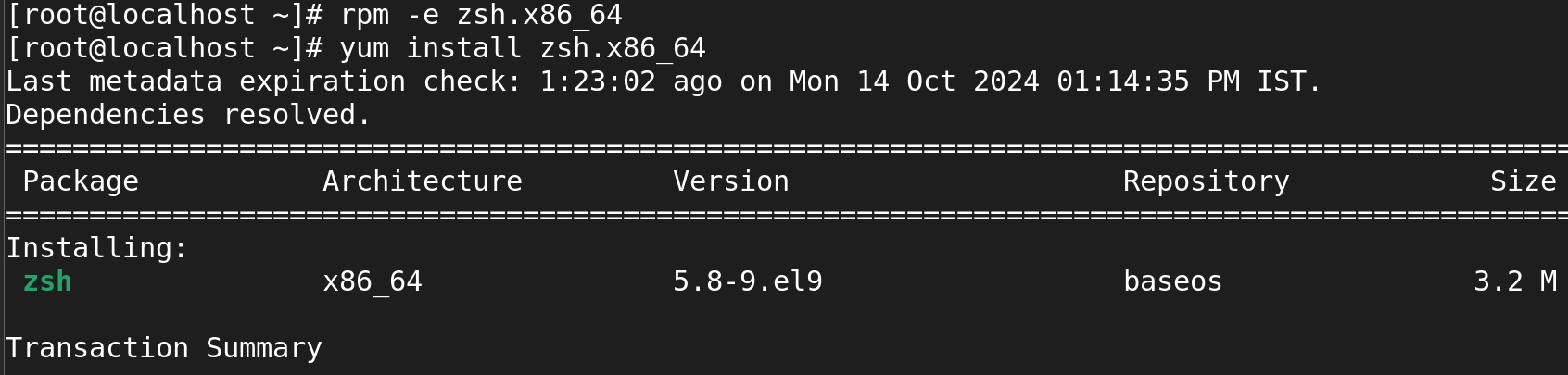
**rpm -ivh <package\_file.rpm>**

1. How to remove the software package from on Redhat Linux?

**rpm -e <package\_name>**

**yum remove <package\_name>**

**dnf remove <package\_name>**



1. what is the repository file location?

**Primary Location:**

* /etc/yum.repos.d/

This directory contains .repo files that define repositories for the yum package manager. Each repository can have its own configuration file.

1. write yum configuration file?

**Create a new repository file** in the /etc/yum.repos.d/ directory. For example, you can create a file named local.repo.

**sudo vi /etc/yum.repos.d/local.repo**

**Inside the configfile**

[local-repo]name=Local Repository

baseurl=file:///mnt/repo/

enabled=1

gpgcheck=0

**LVM**

1. Is it possible to increase the logical volume online?

Yes, it is possible to increase a logical volume (LV) online in Linux, especially if you are using Logical Volume Management (LVM). This feature allows you to resize the logical volume while it is mounted and in use, without the need for downtime

1. how to remove LVM completely from the host?

Removing LVM (Logical Volume Management) completely from a host involves several steps, including unmounting filesystems, removing logical volumes, volume groups, and physical volumes.

1. how to reduce lvm file system?

1. In MBR how many numbers of partition can create?

In MBR (Master Boot Record) partitioning scheme, you can create:

Up to 4 primary partitions: This is the maximum number of primary partitions you can have in an MBR partitioning scheme.

Alternatively, if you need more than four partitions.

you can create:

3 primary partitions + 1 extended partition: The extended partition can then be subdivided into multiple logical partitions. This allows you to create a total of up to 128 logical partitions within the extended partition, depending on the operating system.

1. In GPT how many numbers of partition can create?

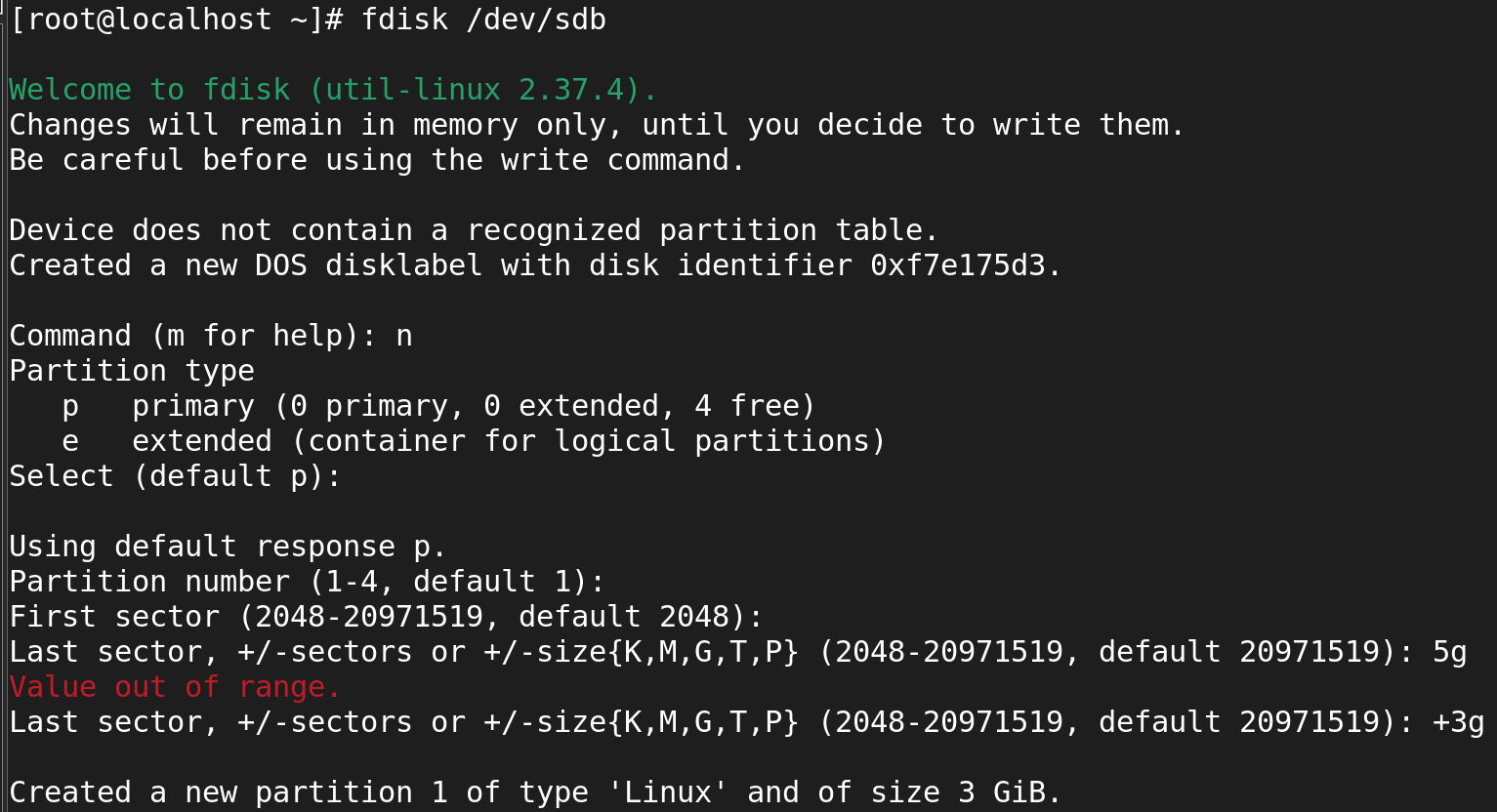
In the GPT (GUID Partition Table) partitioning scheme, you can create:

* **Up to 128 primary partitions**: GPT allows for a much larger number of partitions compared to MBR, and the default limit is 128 partitions.

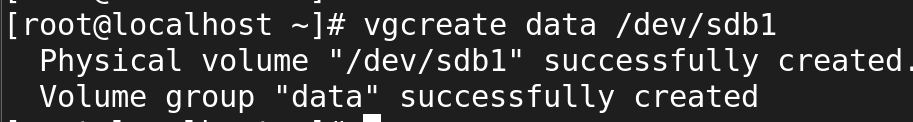
· **Maximum Size**: Up to 9.4 ZB

· **Redundancy and Reliability**: Better compared to MBR

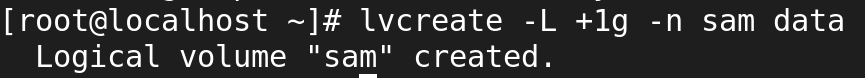
1. what are the commands to create partition?



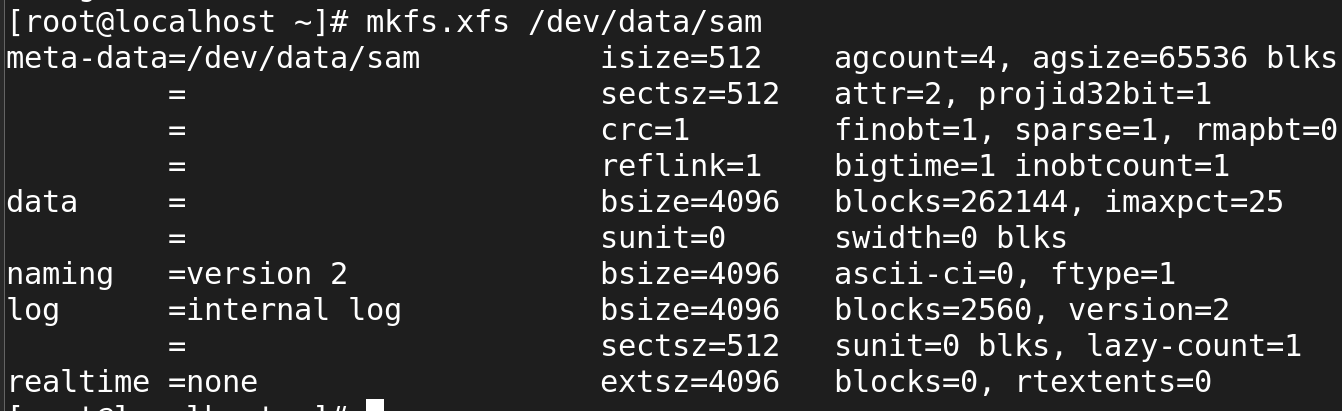
1. what is the command to create volumegroup?

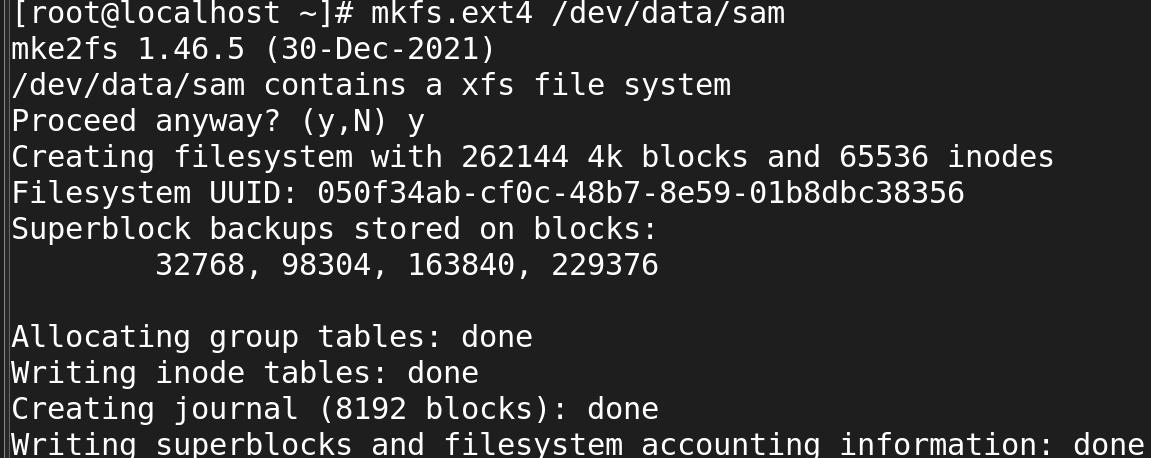


1. what is the command to create logical volume?



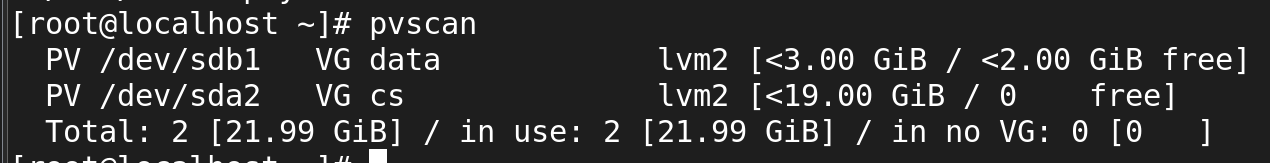
1. how to create ext4 and xfs file system?



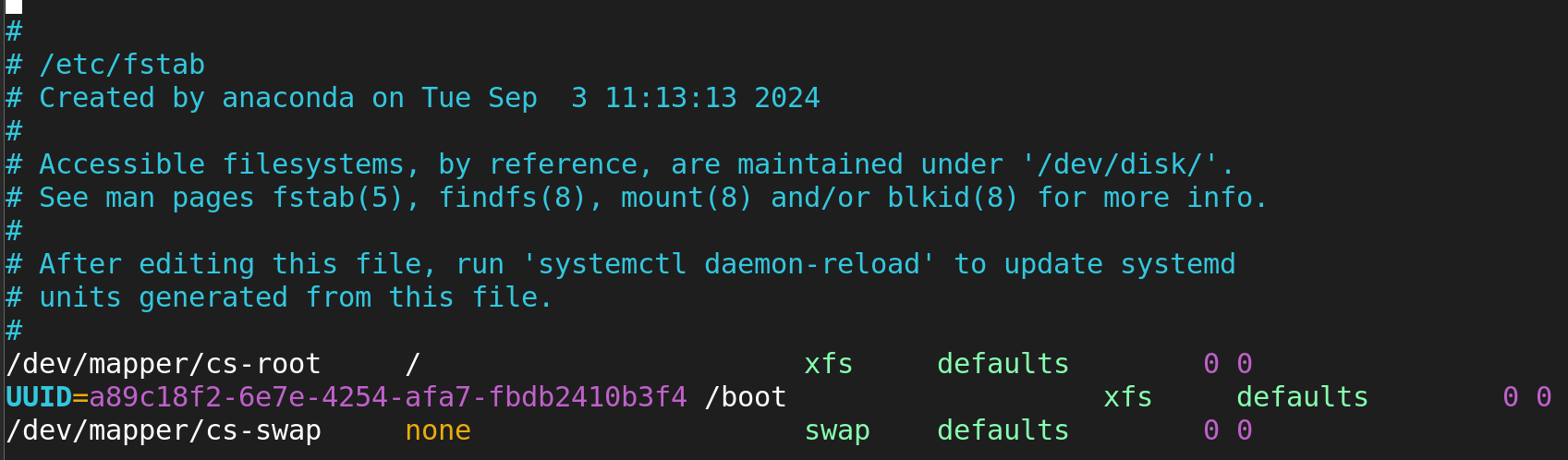


1. how to scan physical volume?



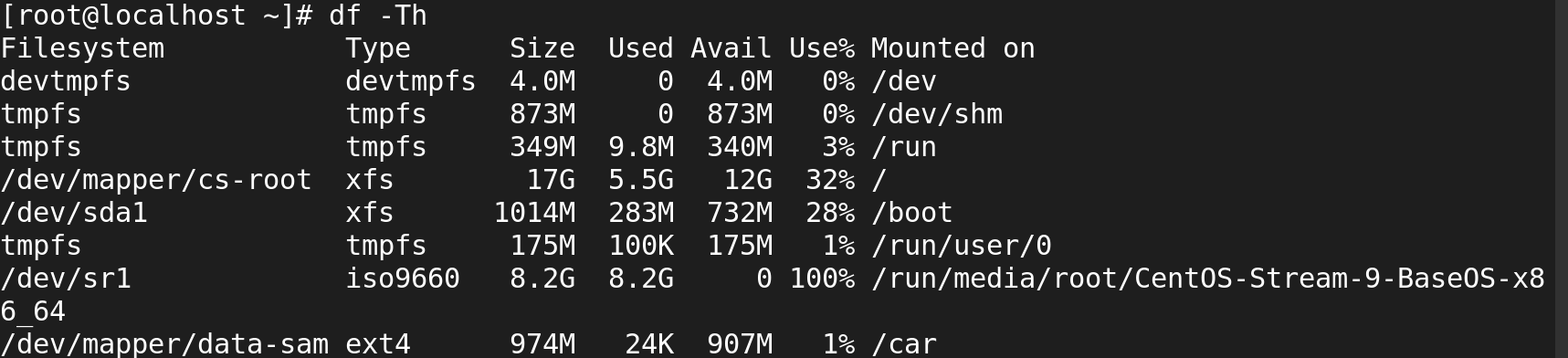


1. what are the field in /etc/fstab?



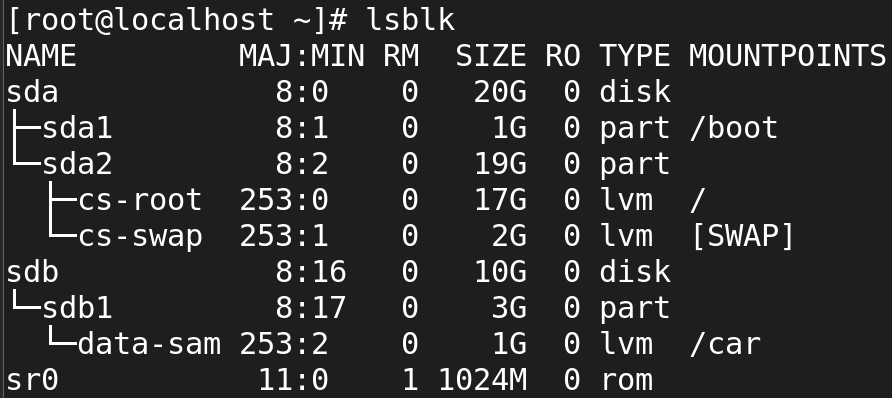
1. what is the command see the mounted file system?

**# df -Th**



1. what is the command get disk details?

**Using** lsblk:



1. How to see the detailed volume group, physical and **logical volume information?**

**By using command like**

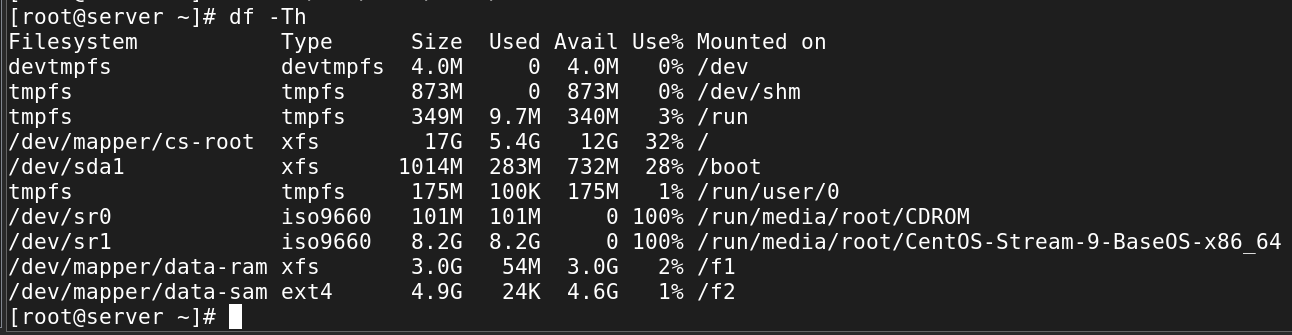
**Lvs**

**Pvs**

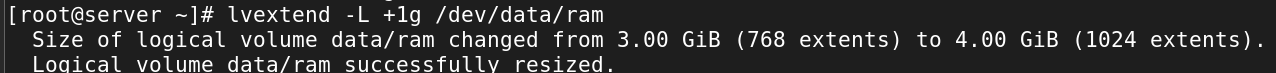
**vgs**

1. Attach LUN/disk 3GB, create lvm based filesystem(xfs), mountpoint should be /dev\_data

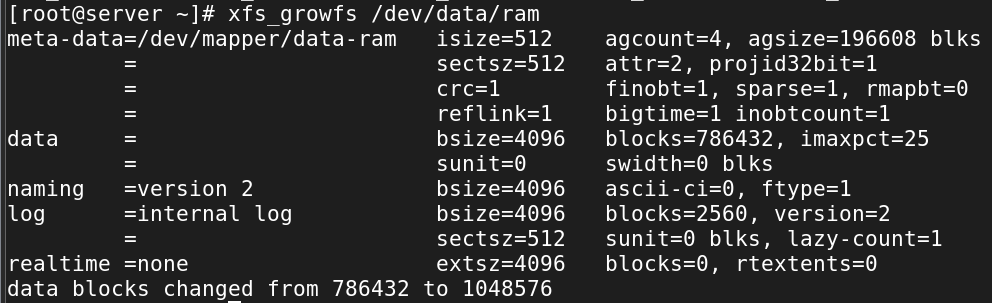
* df -h

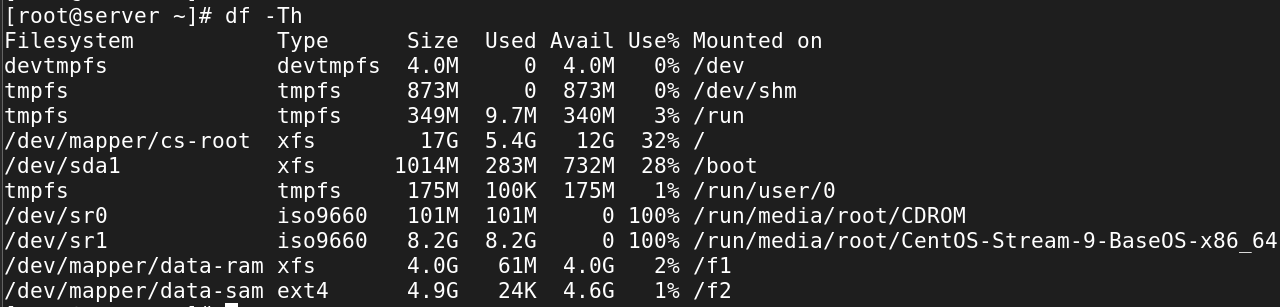


1. Extend the above /dev\_data filesystem 1GB final file system size should be 4GB?

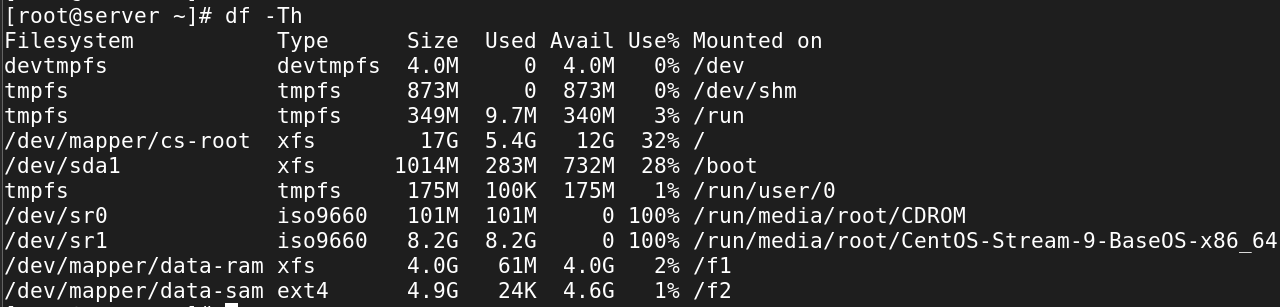


For extended lvg in ram to show in df -Th we should grow file system.





1. reduce above /dev\_data filesystem 1GB?

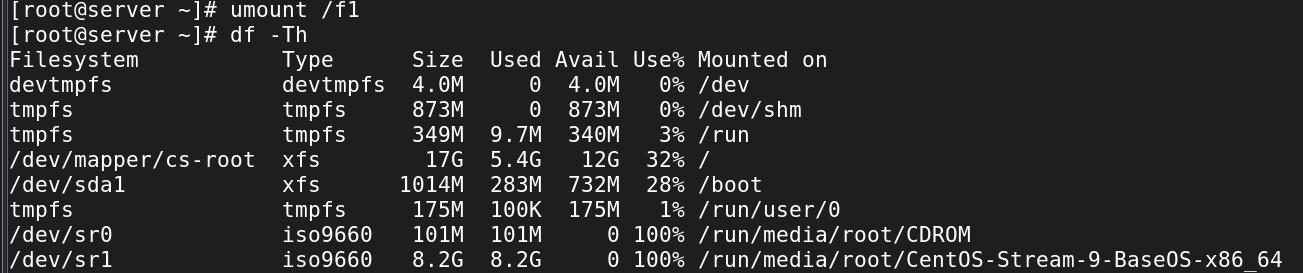


**Remove LVM:**

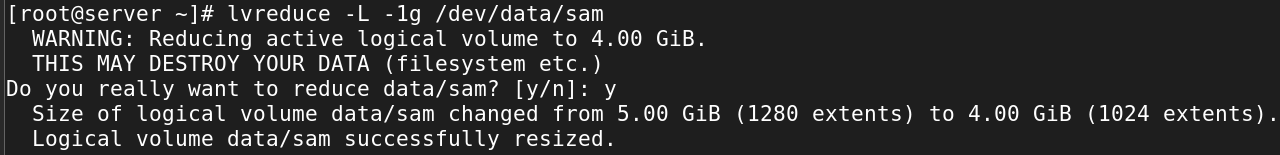
Remove fstab entry first

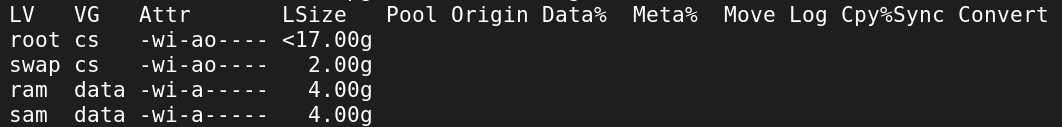


**Unmount the directory:**

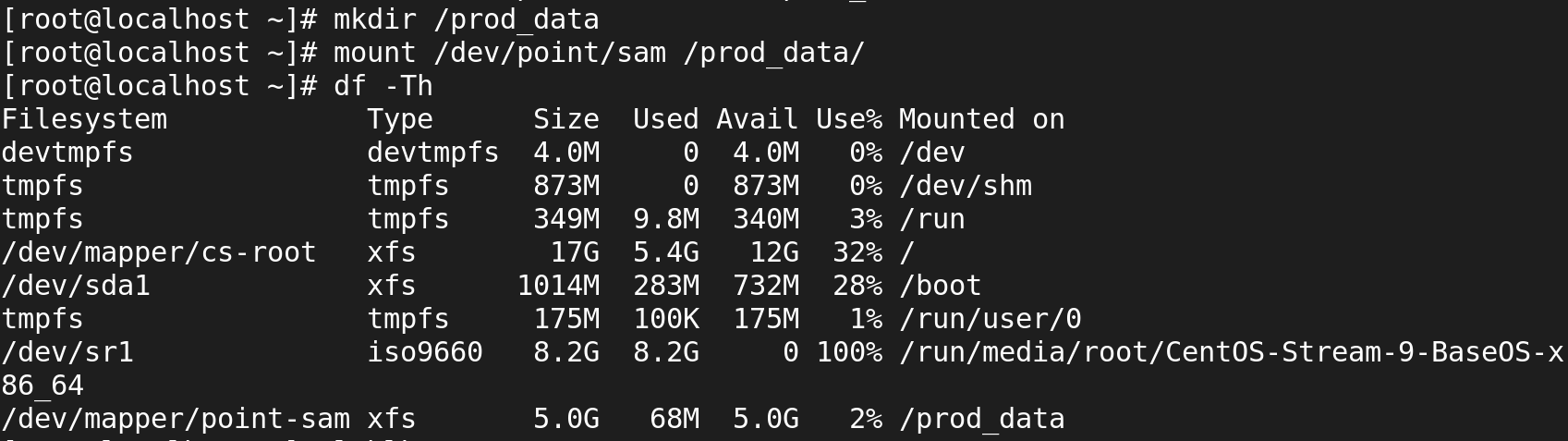


To reduce the logical volume:

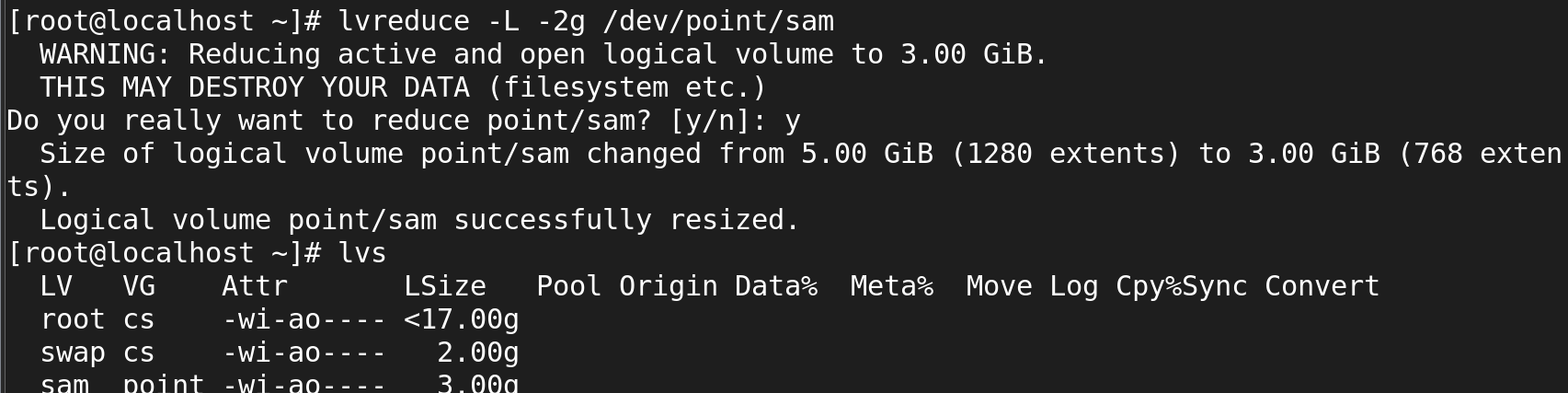




1. Attach LUN/disk 6GB, create lvm based filesystem(ext4), mountpoint should be /prod\_data

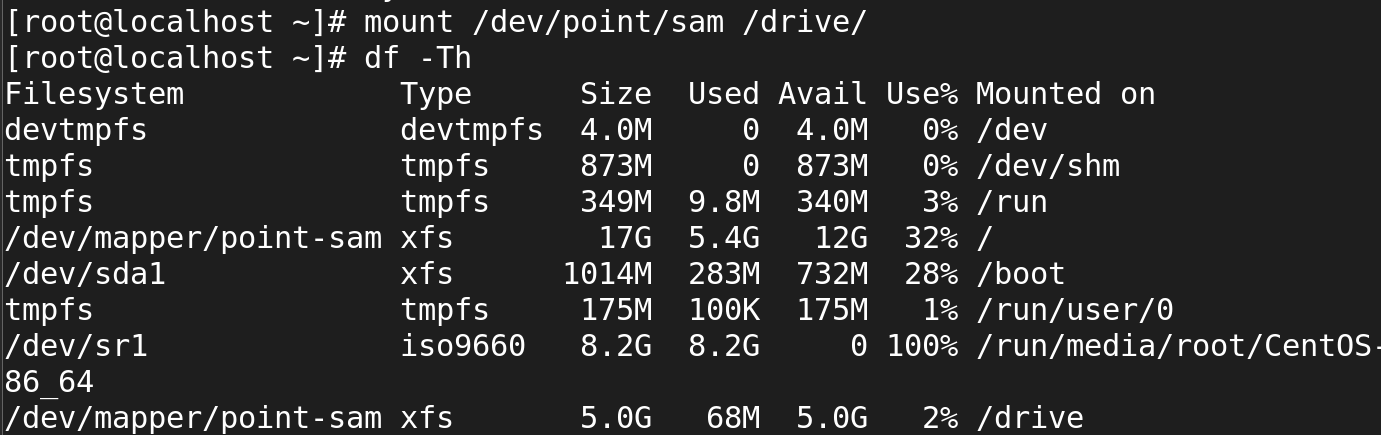


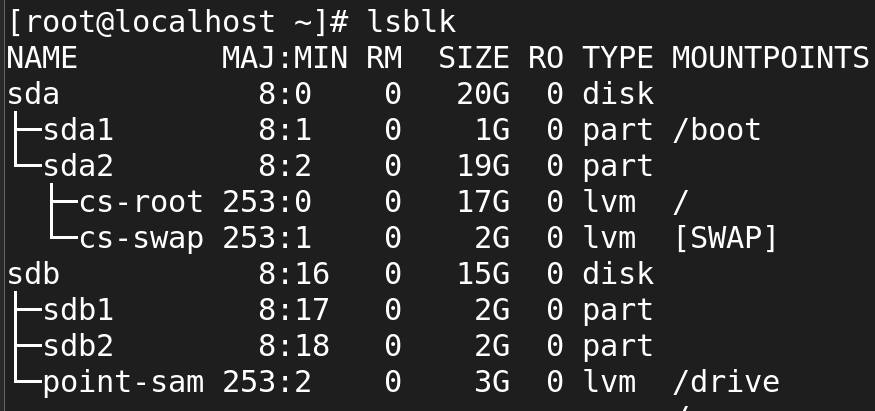
1. reduce above lvm filesystem 2GB and final size of filesystem should be 4G?



1. is it possible to reduce the xfs filesystem?

No, it is not possible to reduce an XFS filesystem directly. XFS does not support shrinking the filesystem.

1. Attach disk 4GB, create partition-based filesystem, mountpoint should be /drive 
2. Attach disk 6GB, create partitions 3GB & 3GB respectively and create filesystem



1. Attach disk 3GB, create raw disk-based filesystem, mount point should be /device
2. What are the steps involved to create the logical volume from scratch?

**sudo fdisk /dev/sdb**

**sudo pvcreate /dev/sdb1**

**sudo vgcreate my\_volume\_group /dev/sdb1**

**sudo lvcreate -L 10G -n my\_logical\_volume my\_volume\_group**

**sudo mkfs.ext4 /dev/my\_volume\_group/my\_logical\_volume**

**sudo mkdir /mnt/my\_mount\_point**

**sudo mount /dev/my\_volume\_group/my\_logical\_volume /mnt/my\_mount\_point**

**sudo nano /etc/fstab**

1. what is swap?

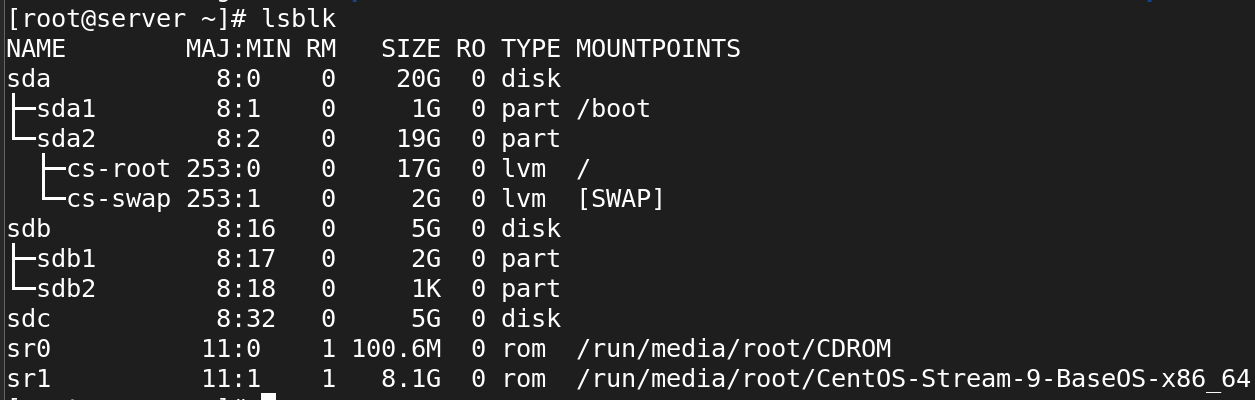
**Swap** in Linux is a space on the disk used as virtual memory when the system's physical RAM (Random Access Memory) is fully utilized. If the system's RAM is running low or is completely exhausted, inactive pages in memory are swapped out to this disk space, allowing the system to continue running without running out of memory.

**Purpose of Swap:**

* It acts as an overflow for when physical memory (RAM) is insufficient.
* It helps in preventing out-of-memory errors.

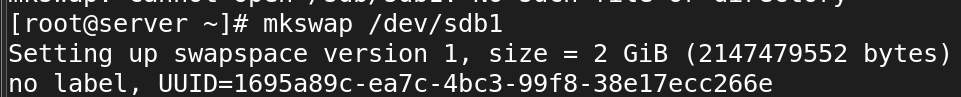
1. create 2Gb swap filesystem and enable swap?

* lsblk



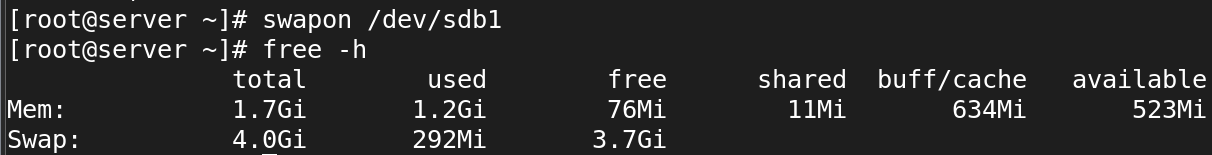
**Make SWAP:**

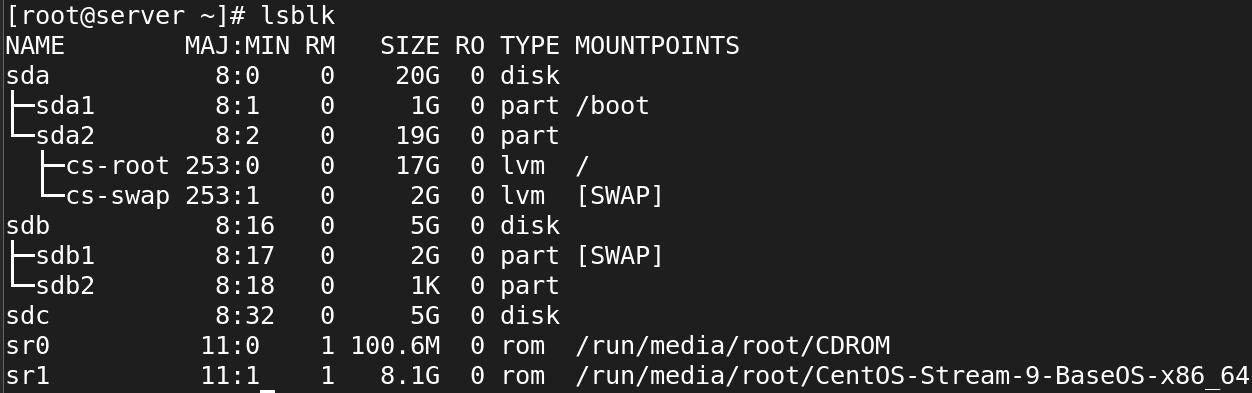
mkswap /dev/sdb1



**On the SWAP:**

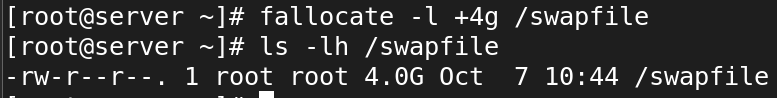
* swapon /dev/sdb1





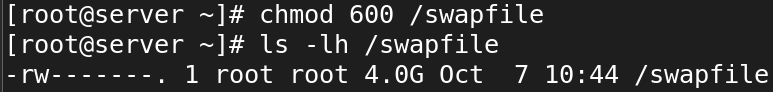
1. create 2GB swap file and enable swap

fallocate -l +4G /swapfile



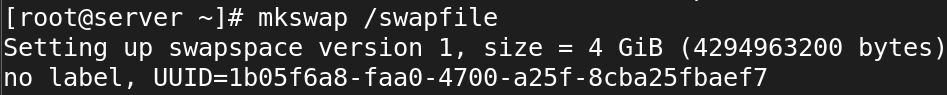
1. For Swapping the file we should change 644 to 600 Permission:

* chmod 600 /swapfile



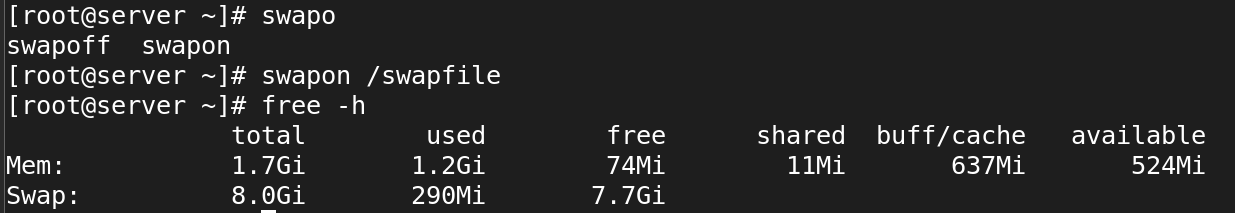
1. Make SWAP:

* mkswap /swapfile



1. On the SWAP:

* swapon /swapfile



**SSH**

* what is ssh?

**SSH (Secure Shell)** is a **network protocol** used to securely connect to remote machines and systems over an insecure network. It enables encrypted communication between the client and the server, ensuring that data (such as commands, files, and credentials) is transmitted securely.

* port number of SSH?
* default port 🡪 22
* Is it possible to change ssh port number?

Yes, it is possible to change the SSH port number. By default, SSH listens on port **22**, but you can configure it to use a different port for added security or to avoid conflicts with other services.

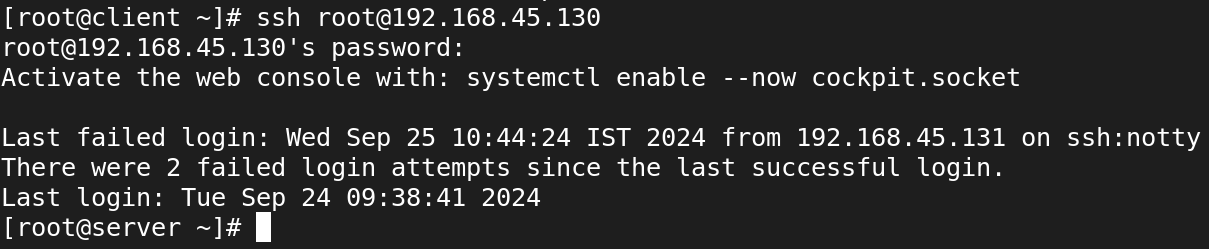
The configuration file is usually located at /etc/ssh/sshd\_config.

#Port 22

Port 2222

Write command to login remote machine in terminal?

**ssh root@192.168.45.130**



* Write command to login local machine in terminal?

To log in to the local machine using the terminal, you can simply use the ssh command with the local username and the localhost or 127.0.0.1 address (the loopback address that refers to the local machine).

ssh username@localhost

ssh username@127.0.0.1

* ssh configuration file location?

**SSH Client Configuration**

* **Location**: ~/.ssh/config
* **Description**: This file contains configuration settings for the SSH client, allowing users to customize their SSH connections. You can specify options such as host aliases, user names, ports, and identity files for specific remote hosts.

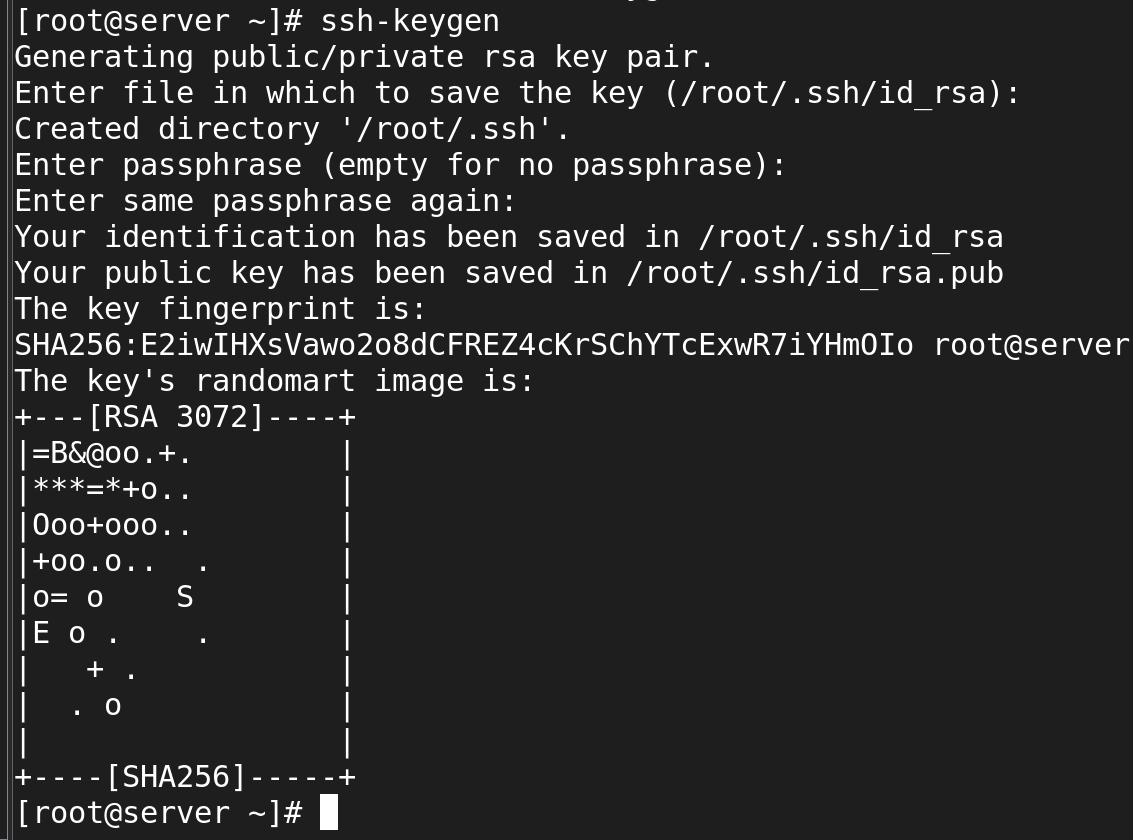
**SSH Server Configuration**

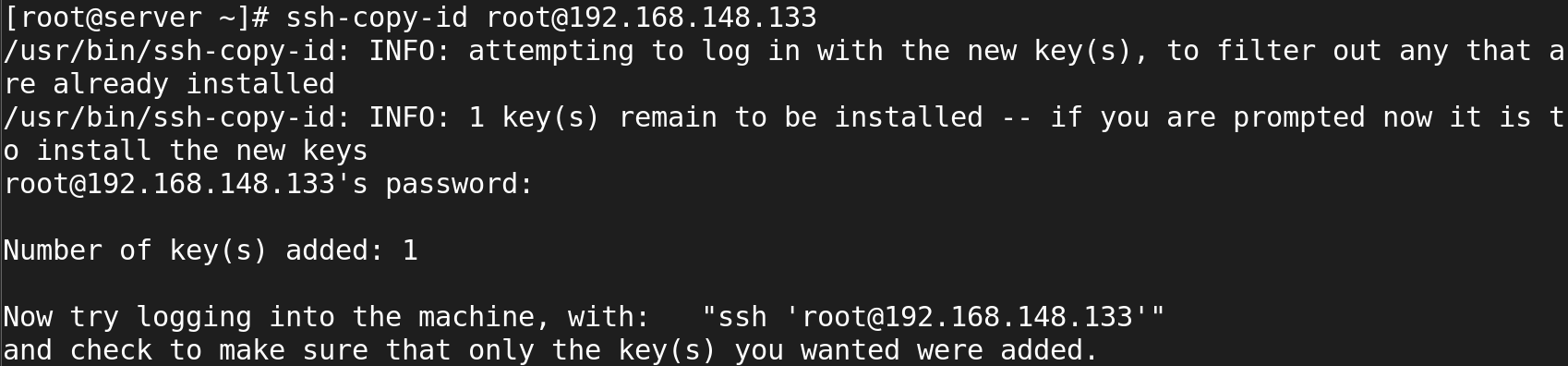
* **Location**: /etc/ssh/sshd\_config
* **Description**: This file contains the configuration settings for the SSH server (sshd). It controls how the SSH server behaves and manages security settings, authentication methods, and more.
* configure SSH key based authentication? write path of key file location? server1-user:admin1 to server2-user:admin1

* what is the command to generate the ssh key?

**Syntax:**

* ssh-keygen



* what is the command to copy the keys to remote machine? 
* when you generate the key what are files will be create?

**Private Key**

* **File Name**: id\_rsa (or another name if specified)
* **Location**: ~/.ssh/id\_rsa
* **Description**: This file contains the private key, which should be kept secure and never shared with anyone. It is used to authenticate your identity when connecting to an SSH server. The private key must be protected from unauthorized access.

**Public Key**

* **File Name**: id\_rsa.pub (or another name with .pub extension if specified)
* **Location**: ~/.ssh/id\_rsa.pub
* **Description**: This file contains the public key, which can be shared with remote SSH servers to allow authentication. You will copy the contents of this file to the authorized keys file on the remote server to enable key-based authentication.
* what is the ssh daemon name?
* **sshd** is the server-side component of the SSH protocol, responsible for handling incoming SSH connections, managing user authentication, and providing secure access to the system over a network.
* It listens for connection requests on a specified port (default is port **22**) and initiates secure communication sessions with SSH clients.
* write ssh service-related commands which you know?

sudo systemctl start sshd

sudo systemctl stop sshd

sudo systemctl restart sshd

sudo systemctl status sshd

sudo systemctl reload sshd

sudo systemctl enable sshd

sudo systemctl disable sshd

* Disable ssh root login?

sudo vi /etc/ssh/sshd\_config

PermitRootLogin no

sudo systemctl restart ssh

* How to disable password-based authentication?

To disable password-based authentication for SSH and enforce key-based authentication, follow these steps:

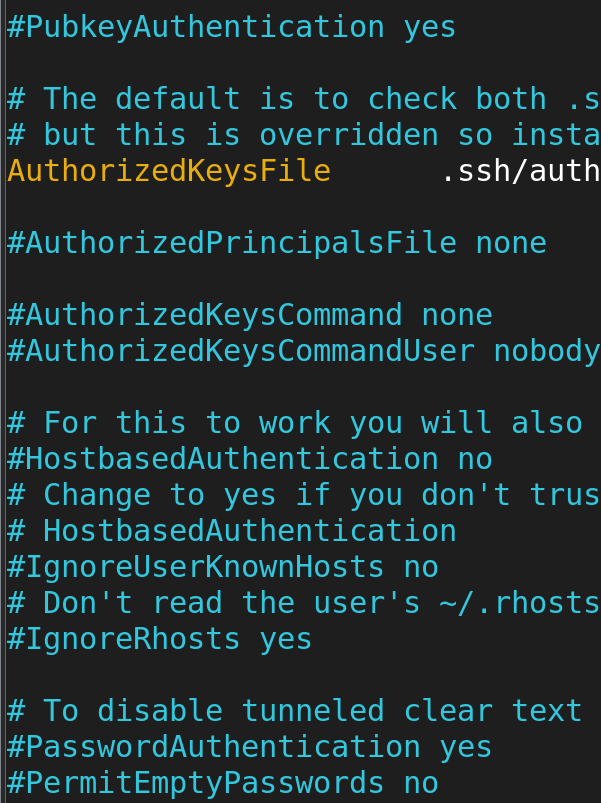
vi /etc/ssh/sshd\_config

#PasswordAuthentication yes

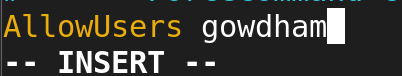
PasswordAuthentication no

PubkeyAuthentication yes

sudo systemctl restart sshd



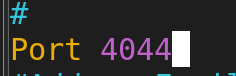
* Allow ssh access to particular user? (user1, user2, user3)

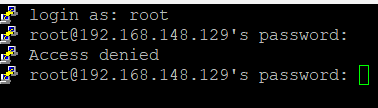


sudo systemctl restart sshd

ssh user1@your\_server\_ip

* Change ssh port number 4044 and try to ssh login.





* How to start, stop, restart, check the status of ssh service?

### **Start the SSH Service**

* To start the SSH service (sshd)
* systemctl start sshd



### **Stop the SSH Service**

* To stop the SSH service
* systemctl stop sshd



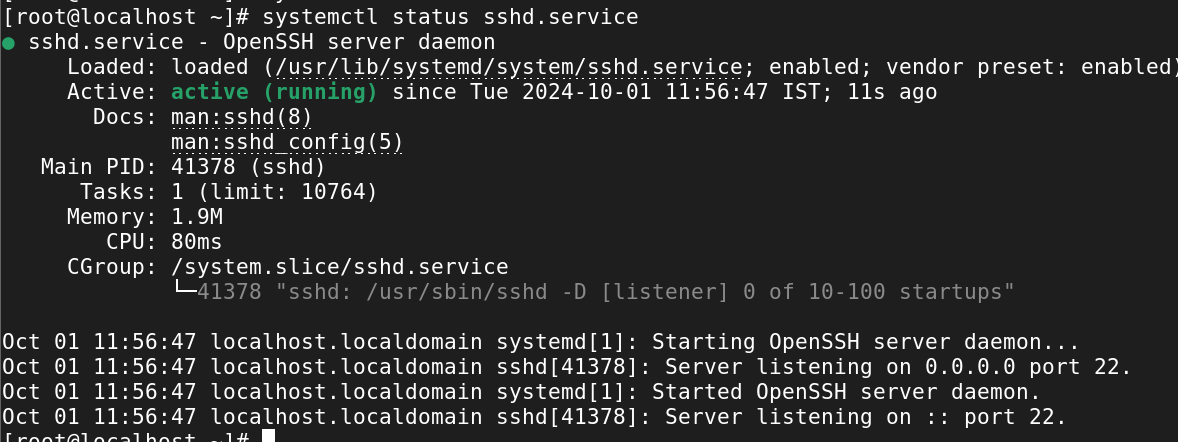
### **Restart the SSH Service**

* To restart the SSH service, which is useful after making configuration changes
* systemctl restart sshd



### **Check the Status of the SSH Service**

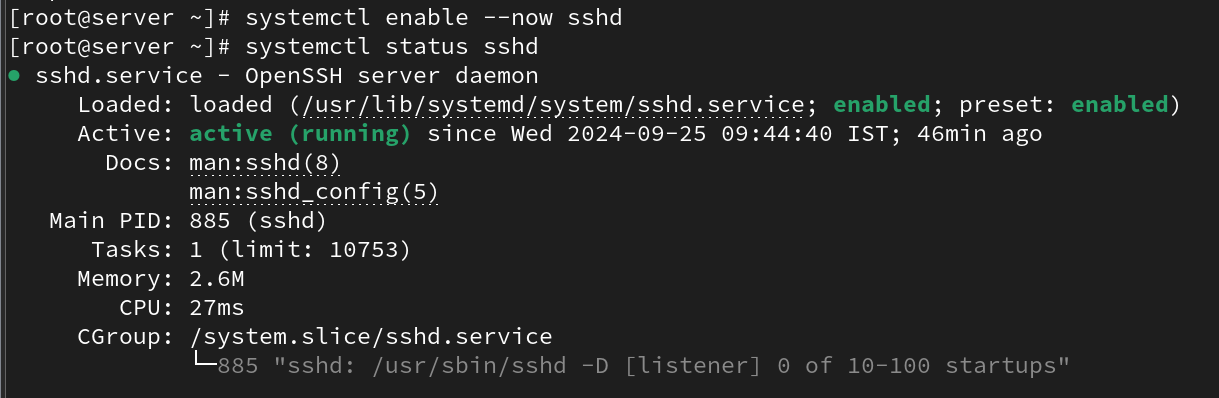
* To check the current status of the SSH service, use:
* systemctl status sshd



* How to start the ssh service permanently?

Syntax:

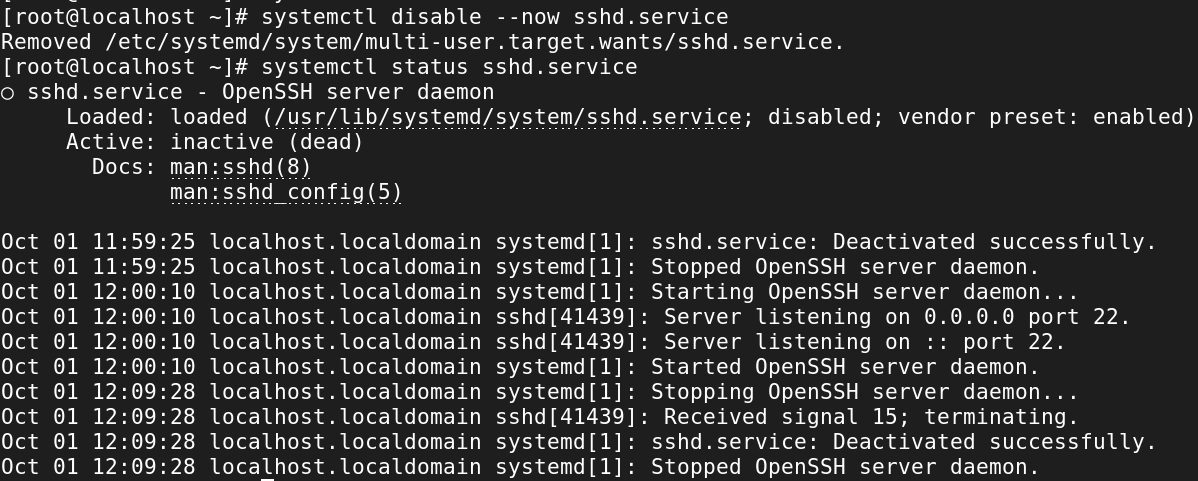
* systemctl enable --now sshd



* How to stop the ssh service permanently?

Syntax:

* systemctl disable --now sshd



Cron

=======

1. what is cronjob?

* The cron tool allows Linux users to execute commands or scripts at a specific date and time.
* It is very useful to perform regular tasks such as regular backups, daily scans, /tmp directory cleanup, and restarting the system at a given time.
* cron executes scheduled jobs automatically in the backend at a specific time.
* Each user can have their own ctrontab, and this can be found it in “/var/spool”.
* cron jobs can be allowed or disallowed for any users by adding a user in the “cron.allow” and “cron.deny” file.

1. what are the fields in cronjob?

**Star Represents:**

* Minute — (0–59)
* Hour — (0–23)
* Day of the month — (1–31)
* Month — (1–12)
* Day of the week — (0–7) (Sunday is represented by both 0 and 7)

1. how to edit cronjob?

Open the crontab file for editing:

# crontab -e

Add or modify the cron jobs in the following format:

# \* \* \* \* \* command\_to\_run

Example (runs a script every day at 3 AM):

# 0 3 \* \* \* /path/to/your/script.sh

1. How to list cronjob?

To list the cron jobs for the current user in Linux, you can use the following command:

**# crontab -l**

This will display all scheduled cron jobs for the current user.

1. how to remove cronjob?

If you want to remove all cron jobs for the current user, you can run:

**# crontab -r**

This will completely remove the crontab for the current user.

1. how to edit, list, remove other users cronjob?

· **Edit**: sudo crontab -u user1 -e

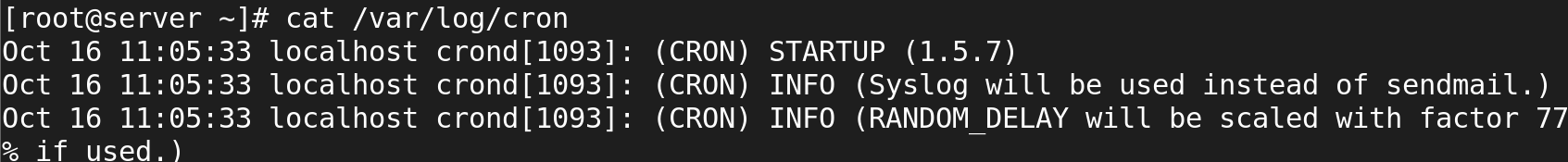
· **List**: sudo crontab -u user1 -l

· **Remove all cron jobs**: sudo crontab -u user1 -r

· **Remove a specific cron job**: sudo crontab -u user1 -e, then delete the specific line.

1. what is the cron log file?

# cat /var/log/cron



1. schedule cronjob at yearly once?

0 0 1 1 \* /path/to/script.sh

* **0 0**: At midnight (00:00)
* **1 1**: On January 1st (the first day of the first month)
* **\***: Every year

This will ensure your script runs at midnight on January 1st each year.

1. schedule cronjob at monthly once?

0 0 1 \* \* /path/to/script.sh

* **0 0**: At midnight (00:00)
* **1**: On the 1st day of the month
* **\***: Every month

This will make the script run once at the beginning of every month.

1. schedule cronjob at daily once?

0 0 \* \* \* /path/to/script.sh

* **0 0**: At midnight (00:00)
* **\***: Every day of the month, every month, and every day of the week

This will run the script once daily at the specified time.

1. schedule cronjob at hourly once?

0 \* \* \* \* /path/to/script.sh

* **0**: At the beginning of the hour
* **\***: Every hour, every day, every month

This will run the script at the start of every hour.

1. schedule the cronjob at machine reboot?

# @reboot /usr/local/bin/myscript.sh

1. schedule task every weekday (Monday to Friday) at 10pm.

# 0 22 \* \* 1-5 your-command-here

· 0: The minute when the command will run (0 minutes past the hour).

· 22: The hour when the command will run (22:00, or 10 PM).

· \*: Every day of the month.

· \*: Every month.

· 1-5: Days of the week (1 is Monday and 5 is Friday, so this covers weekdays).

1. Perform a backup at midnight every Tuesday.

0 0 \* \* 2 your-backup-command-here

### Explanation of the Cron Syntax:

* 0: The minute when the command will run (0 minutes past the hour).
* 0: The hour when the command will run (0:00, or midnight).
* \*: Every day of the month.
* \*: Every month.
* 2: Day of the week (2 represents Tuesday).

1. Perform monitoring every minute during January, February, and May.

\* \* \* 1,2,5 \* your-monitoring-command-here

### Explanation of the Cron Syntax:

* \*: Every minute.
* \*: Every hour.
* \*: Every day of the month.
* 1,2,5: Months (January, February, and May).
* \*: Every day of the week.

1. Clear the cache every 10 minutes at 5am, starting from 5:10am.

\*/10 5 \* \* \* your-cache-clear-command-here

### Explanation of the Cron Syntax:

* \*/10: Every 10 minutes.
* 5: At 5 AM.
* \*: Every day of the month.
* \*: Every month.
* \*: Every day of the week.

1. Make the task run quarterly on the first day of the month at 8am.
2. Create a backup every hour.

# 0 \* \* \* \* /path/to/backup\_script.sh

· 0: At minute 0.

· \*: Every hour.

· \*: Every day of the month.

· \*: Every month.

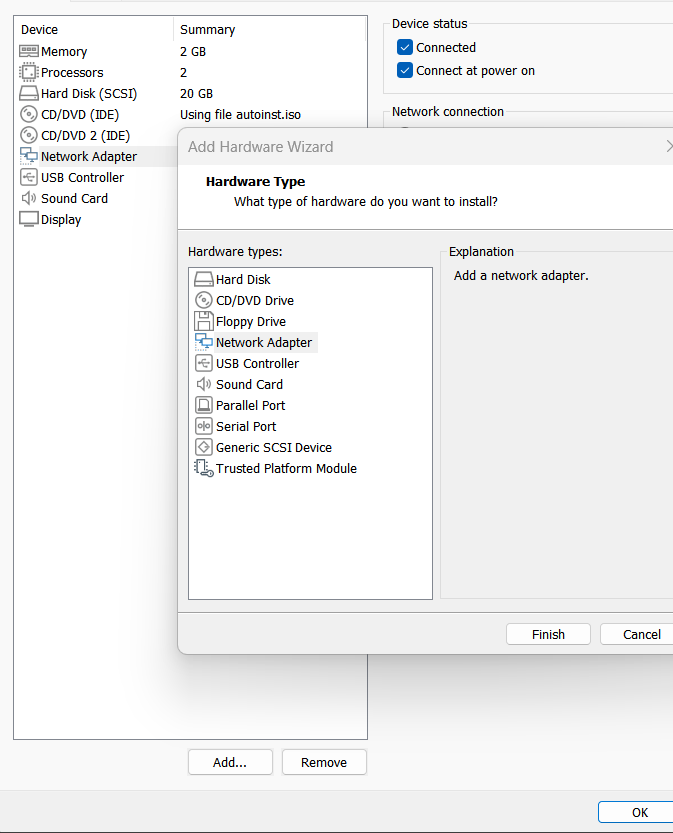
· \*: Every day of the week.

1. Include multiple tasks on a single cron job. Useful for scheduling multiple tasks to run at the same time.
2. Clear cache every time you turn on the system.

**Network**

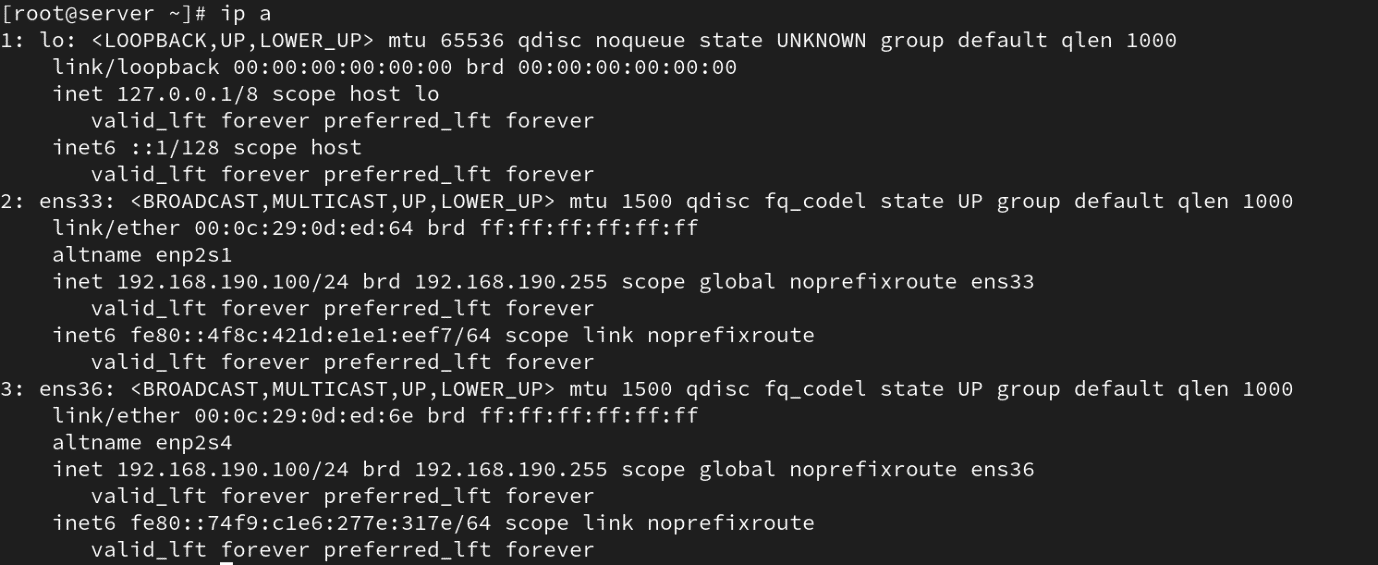
========

* Add additional ethernet card and assign static ip



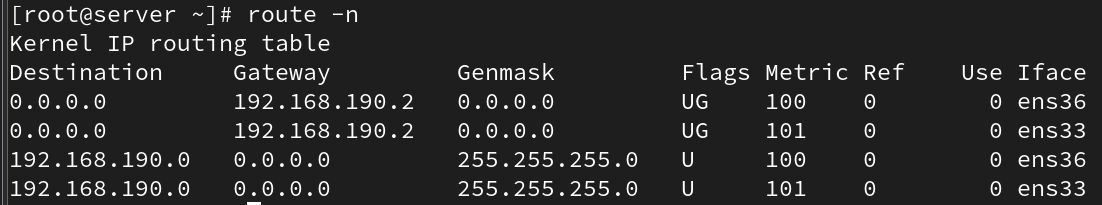
* Add additional ethernet card and assign dynamic ip

#ifconfig or #ip a



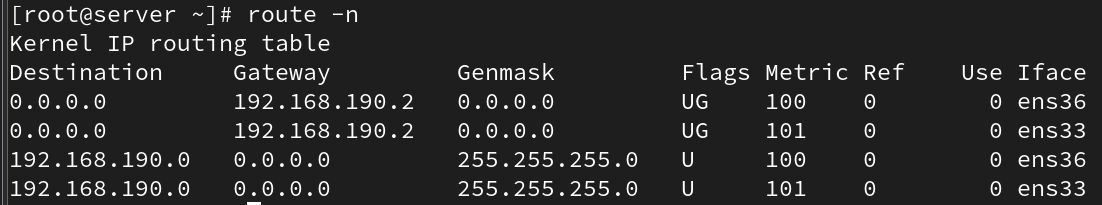
* how to check the server IP also show the current IP details?

# route -n

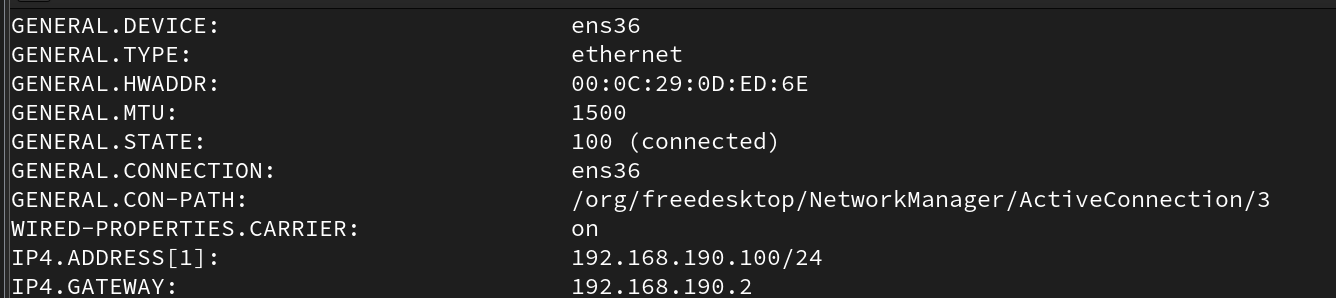


* how to check the server routes also show current route details?

# route -n



* add two ethernet card ans assignee one ip?



* assign two ip address for one Ethernet card?



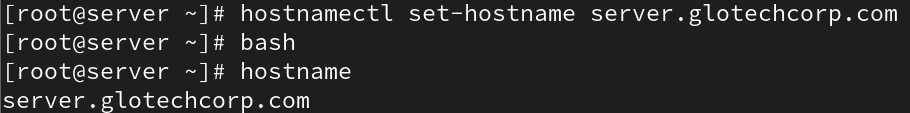
* how to check the server hostname

#hostname



* change server hostname as server.glotechcorp.com permanently

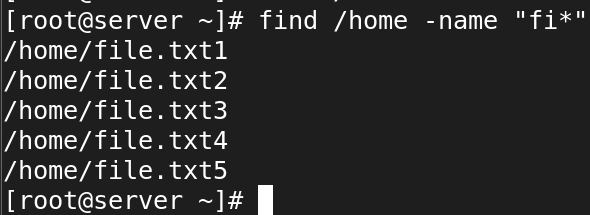
#hostnamectl set-hostname server.glotechcorp.com



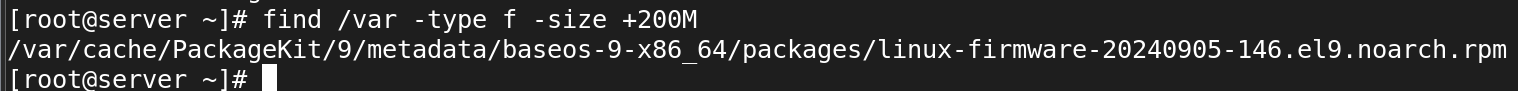
**Find**

====

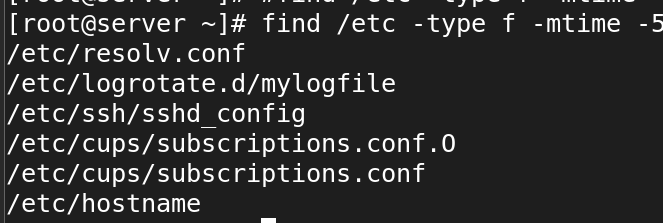
* How would you use the `find` command to search for all files with the `.txt` extension in the `/home` directory?



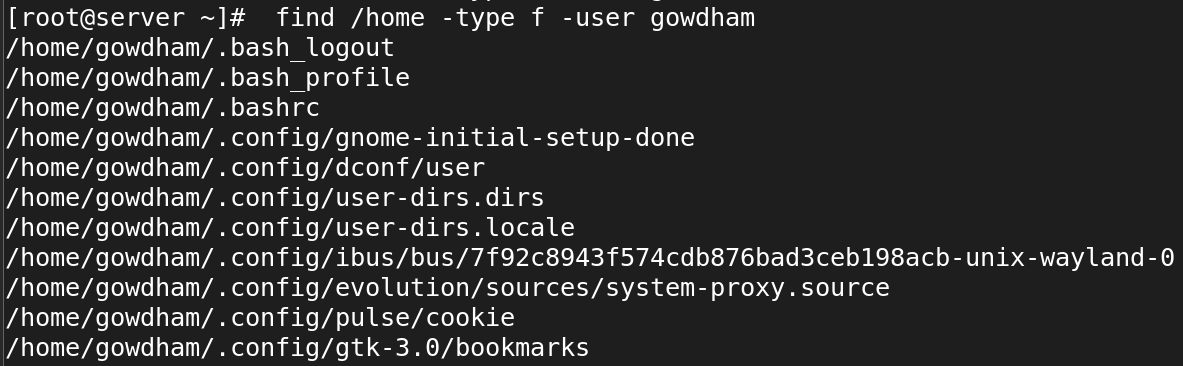
* Write a command using `find` to search for files larger than 200MB in the `/var` directory.



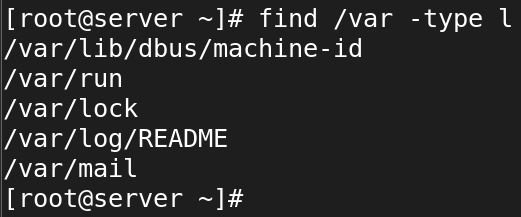
* Use `find` to locate all files in the `/etc` directory that were modified within the last 7 days.



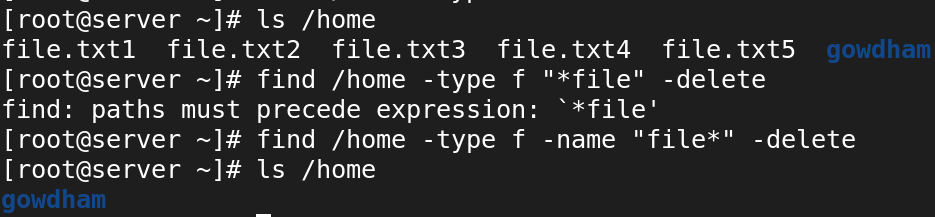
* How can you use `find` to search for files owned by a specific user, for example, `user1`, in the `/home` directory?



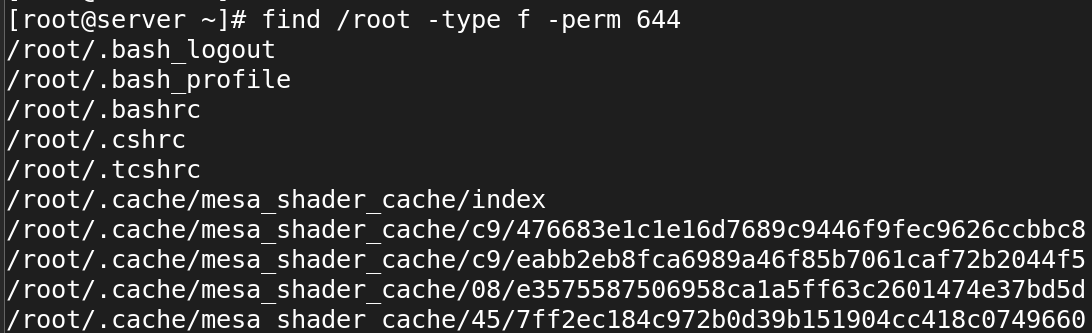
* Find all symbolic links within the `/usr` directory using the `find` command



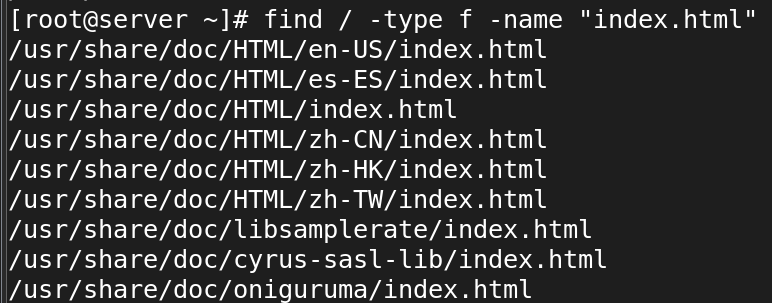
* Write a `find` command to delete all `file` files in the `/home` directory.



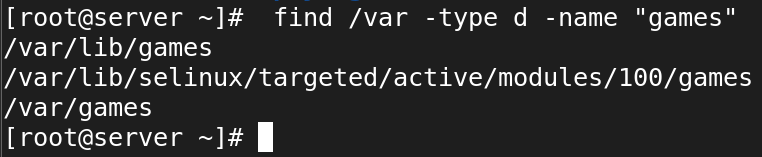
* How would you use the `find` command to locate files with the permissions `644` within `/etc`?



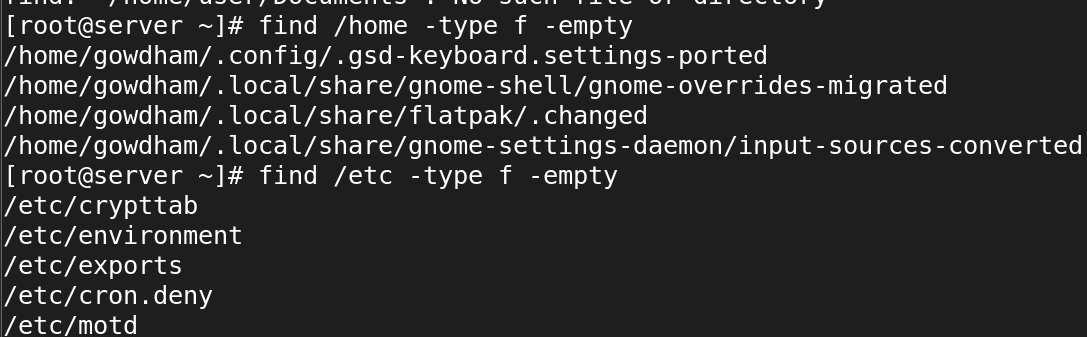
* Use the `find` command to locate all files named `index.html` starting from the root directory (`/`).



* Write a command using `find` to search for directories (not files) named `backup` within the `/var` directory.



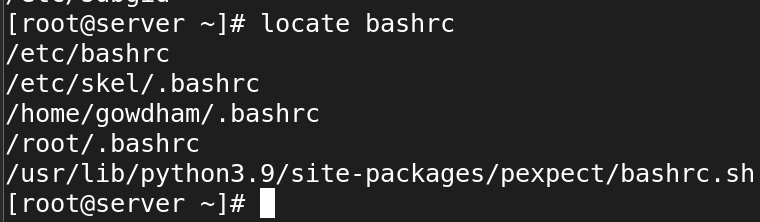
* How would you find all empty files within the `/home/user/Documents` directory using `find`?



**Locate**

======

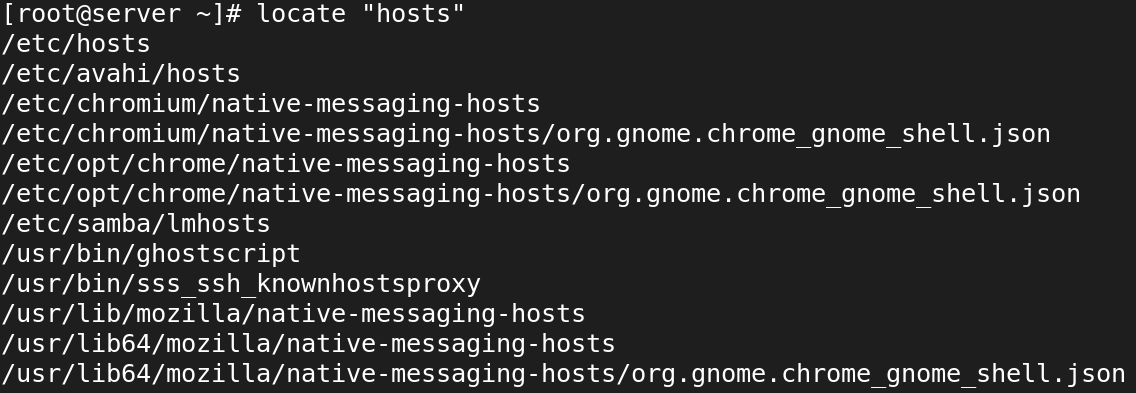
* 1. How would you use the `locate` command to find all files containing the name `bashrc`?



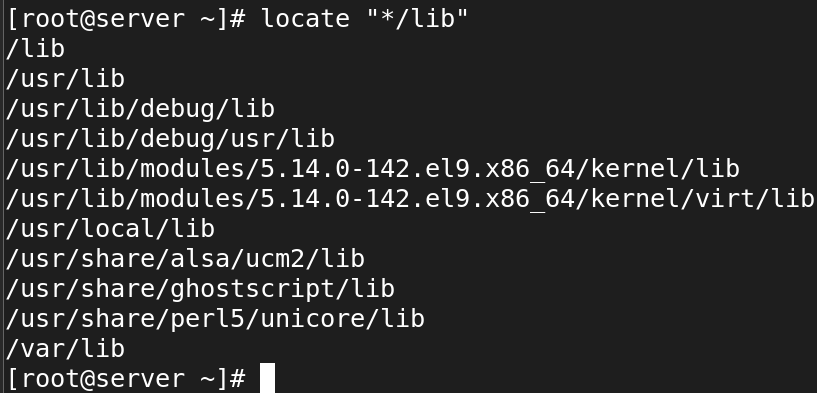
* 2. Write a `locate` command to search for files with the `.conf` extension on your system.



* 3. How can you use `locate` to find all occurrences of a file named `hosts` in the filesystem?

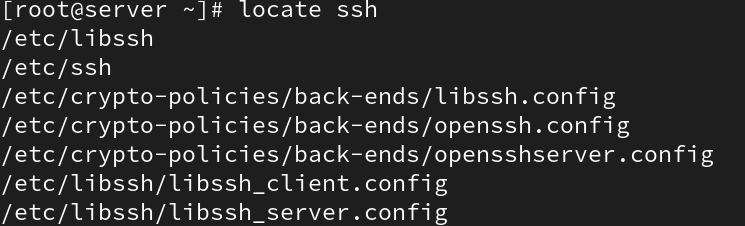


* 4. Use the `locate` command to find directories containing the name `bin` in their path.

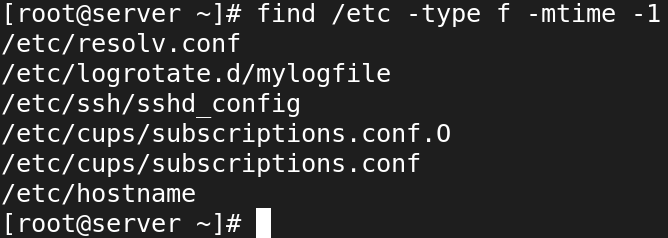


* 5. Write a `locate` command to find files related to the program `ssh`.

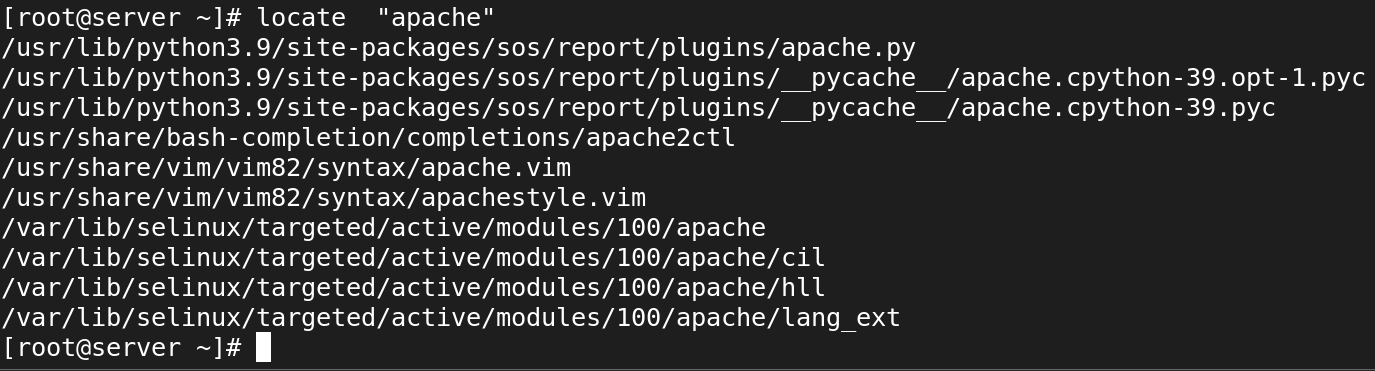
#locate ssh



* 6. How would you use `locate` to search for files that were updated in the last 24 hours?

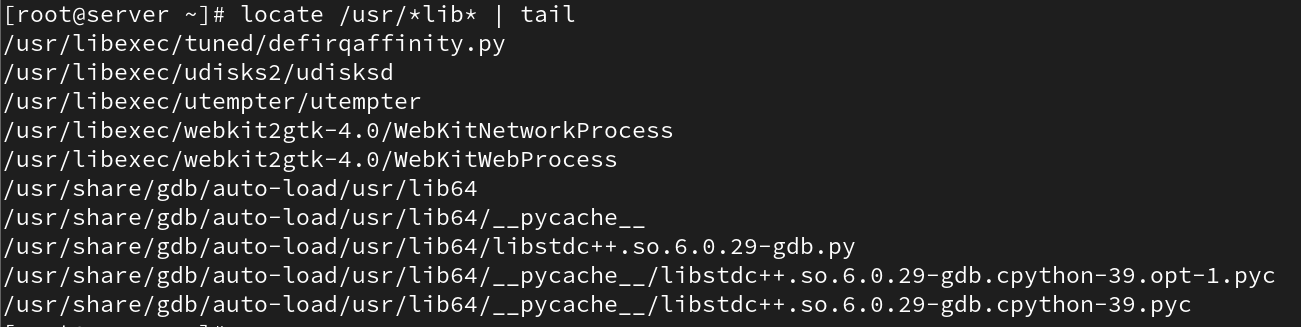


* 7. Use the `locate` command to find all files and directories containing the word `apache`.



* 8. Write a `locate` command to search for all files in the `/usr` directory that contain the string `lib`.

# locate /usr/\*lib\*



* 9. How can you update the `locate` database manually before performing a search?

#updatedb



* 10. Write a `locate` command to find all files with the `.jpg` extension.

# locate "\*.jpg"

