Package Management in Ubuntu (Debian-based Systems)

Category	Command	Description / Purpose
🆙 **Update System**	`bash sudo apt update && sudo apt upgrade -y && sudo apt full-upgrade -y && sudo apt autoremove -y`	Updates the list of available packages, upgrades all installed packages to the latest versions, performs full upgrade, and removes unnecessary dependencies automatically.
📦 **Install a Package**	`bash sudo apt install nginx -y`	Installs the specified package (here, **nginx**) without asking for confirmation.
❌ **Remove a Package**	`bash sudo apt remove nginx -y`	Removes the specified package but keeps configuration files.
🧹 **Completely Remove (Purge)**	`bash sudo apt purge nginx -y`	Removes the package **and** its configuration files.
🔍 **Search for a Package**	`bash apt search python3`	Searches repositories for packages matching a keyword (e.g., **python3**).
ℹ️ **Package Information**	`bash apt show nginx`	Displays detailed information about a package (version, dependencies, maintainer, etc.).
📋 **List Installed Packages**	```bash dpkg -l	grep nginx ```
💾 **Clean Package Cache**	`bash sudo apt clean && sudo apt autoclean`	Removes downloaded package files to free up space. `clean` removes all, while `autoclean` removes only outdated ones.
🧽 **Auto Remove Unused Packages**	`bash sudo apt autoremove -y`	Removes packages that were automatically installed but are no longer needed.
🧱 **Fix Broken Dependencies**	`bash sudo apt --fix-broken install`	Repairs broken package installations or missing dependencies.
💼 **List Upgradable Packages**	`bash apt list --upgradable`	Shows all packages that have newer versions available for upgrade.
📃 **Check Package Source / Repository**	`bash apt policy nginx`	Displays which repository the package is coming from and its priority.
🧩 **Reconfigure a Package**	`bash sudo dpkg-reconfigure package_name`	Reconfigures an already installed package interactively.
🔄 **Upgrade Ubuntu Release**	`bash sudo do-release-upgrade`	Upgrades Ubuntu to the next available distribution release.

Service Management Commands (systemd-based Systems)

Category	Ubuntu/Debian Command	CentOS/RHEL Command	Description / Purpose
▶️ Start Service	`sudo systemctl start nginx`	`sudo systemctl start nginx`	Starts the specified service immediately.
⬛ Stop Service	`sudo systemctl stop nginx`	`sudo systemctl stop nginx`	Stops the specified service immediately.
🔁 Restart Service	`sudo systemctl restart nginx`	`sudo systemctl restart nginx`	Restarts a running service (useful after configuration changes).
🔄 Reload Service (without restart)	`sudo systemctl reload nginx`	`sudo systemctl reload nginx`	Reloads the service configuration without fully restarting it (if supported).
🟢 Enable Service (Auto-start on Boot)	`sudo systemctl enable nginx`	`sudo systemctl enable nginx`	Enables the service to start automatically at system boot.
🔴 Disable Service (No Auto-start)	`sudo systemctl disable nginx`	`sudo systemctl disable nginx`	Disables auto-start at boot time.
📊 Check Service Status	`sudo systemctl status nginx`	`sudo systemctl status nginx`	Displays the current status, active state, logs, and PID of the service.
🔁 Reload Systemd Manager Configuration	`sudo systemctl daemon-reload`	`sudo systemctl daemon-reload`	Reloads systemd configurations (useful after editing service files).
📋 List All Service Unit Files	`systemctl list-unit-files --type=service`	`systemctl list-unit-files --type=service`	Lists all available services and their enable/disable states.

📜 **List Active (Running) Services**	```systemctl list-units --type=service --state=running```	```systemctl list-units --type=servic e --state=runni ng```	**Shows only active (currently running) services.**
🧩 **Check if Service is Enabled**	```systemctl is-enabled nginx```	```systemctl is-enabled nginx```	**Checks if a service is set to start automatically at boot.**
💼 **Check if Service is Active**	```systemctl is-active nginx```	```systemctl is-active nginx```	**Returns "active" or "inactive" for quick status checking.**
🖌️ **Mask a Service**	```sudo systemctl mask nginx```	```sudo systemctl mask nginx```	**Prevents a service from being started manually or automatically.**
🔓 **Unmask a Service**	```sudo systemctl unmask nginx```	```sudo systemctl unmask nginx```	**Reverses masking, allowing the service to start again.**
⚡ **View Recent Logs for a Service**	```sudo journalctl -u nginx```	```sudo journalctl -u nginx```	**Shows logs specific to the given service.**
🕐 **View Real-time Logs (Live Monitoring)**	```sudo journalctl -u nginx -f```	```sudo journalctl -u nginx -f```	**Streams live logs from the service in real-time.**

💾 File System Operations (Ubuntu/Debian & CentOS/RHEL)

Category	Command	Applies To	Description / Purpose
📊 Check Disk Usage	`df -h`	Both	Displays disk space usage for all mounted file systems in human-readable format.
📁 Check Directory Size	`du -sh /var/log`	Both	Shows the total size of the specified directory (`/var/log`).
🔍 Find Files by Name	`find /home -name "*.log"`	Both	Searches for all files with `.log` extension in the `/home` directory.
🕐 Find Recently Modified Files	`find /var -mtime -7`	Both	Finds files in `/var` that were modified within the last 7 days.
🔢 Count Files in a Directory	`` `find /dir -type f ``	wc -l`	Both
📄 Search Text in Files	`grep -r "error" /var/log/`	Both	Searches recursively for the word "error" in all files under `/var/log/`.
📡 Follow Live Log Output	`tail -f /var/log/syslog`	Both	Displays live updates from the log file (`syslog`). *(Use `/var/log/messages` in CentOS)*
📦 Create Archive (Compress Directory)	`tar -czf archive.tar.gz /dir`	Both	Creates a compressed `.tar.gz` archive of the specified directory.
📁 Extract Archive	`tar -xzf archive.tar.gz`	Both	Extracts files from a compressed `.tar.gz` archive.
🔍 View Contents of Archive	`tar -tzf archive.tar.gz`	Both	Lists all files contained inside an archive without extracting them.
🖌️ Delete Old Files	`find /var/log -type f -mtime +30 -delete`	Both	Deletes files older than 30 days in `/var/log`.
💽 Check File System Type	`df -Th`	Both	Displays mounted partitions along with their file system types (e.g., ext4, xfs).

| 🔒 Check File Permissions | `ls -lh /path/to/file` | Both | Displays detailed file information (including size, owner, and permissions). |

🌐 Networking Commands (Ubuntu/Debian & CentOS/RHEL)

Category	Command	Applies To	Description / Purpose
🌍 Show IP Addresses	`ip addr show` *(or)* `ip a`	Both	Displays all network interfaces and their assigned IP addresses.
⚙️ Show Interface Details	`ifconfig` *(requires* `net-tools` *package)*	Both	Shows network interfaces, IPs, MAC addresses, and packet stats.
🔌 Check Listening Ports	`ss -tulpn`	Both	Lists all active listening ports with associated services and PIDs.
📡 Test Network Connectivity	`ping google.com`	Both	Sends ICMP packets to test connectivity and packet loss.
🔍 DNS Lookup	`nslookup domain.com` *(or)* `dig domain.com`	Both	Performs a DNS resolution to find the IP address of a domain.
🗺️ Trace Network Route	`traceroute google.com`	Both	Traces the network path (hops) packets take to reach a host.
📥 Download File via HTTP/HTTPS	`wget https://example.com/file.zip`	Both	Downloads files from the internet using HTTP, HTTPS, or FTP.

🌐 HTTP Header Check (Web Response Test)	`curl -I http://domain.com`	Both	Fetches only HTTP headers to verify website response and status.
📊 Network Interface Statistics	`netstat -i` *(or)* `ip -s link`	Both	Displays interface statistics such as packets transmitted and errors.
🔁 Restart Network Service	`sudo systemctl restart networking` *(Ubuntu)* `sudo systemctl restart network` *(CentOS)*	Both	Restarts the network service (useful after configuration changes).
🧭 Show Routing Table	`ip route show`	Both	Displays current network routes and gateway settings.
🔒 Check Open Connections	`netstat -antp` *(or)* `ss -antp`	Both	Lists all established TCP connections and listening ports.
💬 Display Hostname and IP Info	`hostname -I` *(IPs)* `hostnamectl` *(system info)*	Both	Displays hostname and related network configuration details.
🖌 Flush DNS Cache	`sudo systemd-resolve --flush-caches` *(Ubuntu)* `sudo systemctl restart nscd` *(CentOS)*	Both	Clears cached DNS entries.
📡 Check Network Speed / Latency	`ping -c 10 8.8.8.8`	Both	Sends 10 pings to measure network speed and packet delay.

🔥 Firewall Management (Ubuntu/Debian & CentOS/RHEL)

Category	Ubuntu/Debian Command (UFW)	CentOS/RHEL Command (Firewalld)	Description / Purpose
🔍 **Check Firewall Status**	`sudo ufw status verbose`	`sudo firewall-cmd --state`	**Checks the current status of the firewall (active/inactive or running).**
🟢 **Enable Firewall**	`sudo ufw enable`	`sudo systemctl enable firewalld --now`	**Enables and starts the firewall service immediately.**
🔴 **Disable Firewall**	`sudo ufw disable`	`sudo systemctl disable firewalld --now`	**Disables and stops the firewall service.**
🧱 **Allow Port (e.g., SSH)**	`sudo ufw allow 22/tcp`	`sudo firewall-cmd --permanent --add-port=22/tcp sudo firewall-cmd --reload`	**Opens a specific TCP/UDP port permanently.**
🚫 **Deny Port (e.g., SSH)**	`sudo ufw deny 22/tcp`	`sudo firewall-cmd --permanent --remove-port=22/tcpsudo firewall-cmd --reload`	**Blocks access to a specific port.**
🌐 **Allow Service (by name)**	`sudo ufw allow 'Nginx Full'`	`sudo firewall-cmd --permanent --add-service=httpsudo firewall-cmd --reload`	**Allows predefined services (e.g., HTTP, HTTPS, SSH).**
🧱 **List All Rules / Allowed Ports**	`sudo ufw status numbered`	`sudo firewall-cmd --list-all`	**Displays all active firewall rules and allowed services.**
🖌️ **Delete a Specific Rule**	`sudo ufw delete <rule_number>`	`sudo firewall-cmd --permanent --remove-service=httpsudo`	**Deletes a specific rule from the firewall configuration.**

		`firewall-cmd --reload`	
🔄 **Reload Firewall Rules**	`sudo ufw reload`	`sudo firewall-cmd --reload`	**Reloads firewall configurations without restarting the service.**
🕐 **View Recent Blocked Logs**	`sudo less /var/log/ufw .log`	`sudo less /var/log/firewall d`	**Shows recent blocked or allowed connection logs.**
📋 **Show All Zones**	*N/A (UFW has no zones)*	`sudo firewall-cmd --list-all-zones`	**Displays all available network zones and their rules.**
🧩 **Assign Interface to Zone**	*N/A*	`sudo firewall-cmd --zone=public --change-interfac e=eth0 --permanent`	**Assigns a specific network interface to a firewall zone.**
💼 **Check Active Zones & Interfaces**	*N/A*	`sudo firewall-cmd --get-active-zone s`	**Shows which zones and interfaces are currently active.**

👤 User & Permission Management (Ubuntu/Debian & CentOS/RHEL)

Category	Ubuntu/Debian Command	CentOS/RHEL Command	Description / Purpose
👥 Add New User (with Home Directory)	`sudo useradd -m john`	`sudo useradd -m john`	Creates a new user named john with a home directory `/home/john`.
🔒 Set User Password	`sudo passwd john`	`sudo passwd john`	Sets or changes the password for the specified user.
🧩 Add User to Group	`sudo usermod -aG group john`	`sudo usermod -aG group john`	Adds a user to a specific group (without removing existing group memberships).
⚙️ Grant Sudo Access	`sudo usermod -aG sudo john`	`sudo usermod -aG wheel john`	Grants administrative (sudo) privileges to the user by adding them to the sudo/wheel group.
🗑️ Delete User (Keep Home Directory)	`sudo userdel john`	`sudo userdel john`	Deletes a user but retains their home directory and files.
🖌️ Delete User (Remove Home Directory)	`sudo userdel -r john`	`sudo userdel -r john`	Deletes the user and their home directory completely.
👀 List All Users	`cut -d: -f1 /etc/passwd`	`cut -d: -f1 /etc/passwd`	Lists all users on the system.
📄 List All Groups	`cut -d: -f1 /etc/group`	`cut -d: -f1 /etc/group`	Lists all existing groups.
🔒 Check User Groups	`groups john`	`groups john`	Displays all groups a user belongs to.
🧱 Change File Permissions	`chmod 755 script.sh`	`chmod 755 script.sh`	Sets read, write, and execute permissions for the owner, and read/execute for others.
🧍 Change File Owner	`sudo chown user:group file.txt`	`sudo chown user:group file.txt`	Changes the ownership of a file or directory.

⚡ **Make File Executable**	`chmod +x script.sh`	`chmod +x script.sh`	**Grants execute permission to the file owner (makes it runnable).**
🔁 **Switch User Account**	`su - username`	`su - username`	**Switches to another user account (requires password of the target user).**
🔄 **View Current User**	`whoami`	`whoami`	**Displays the name of the currently logged-in user.**
🗣 **Show Login History**	`last`	`last`	**Shows the history of user logins and reboots.**
🕵️ **Check Current Logged-in Users**	`who`	`who`	**Lists users currently logged into the system.**
💼 **Modify User Info (Full Name, etc.)**	`sudo chfn john`	`sudo chfn john`	**Updates a user's personal details such as full name or office info.**

⚙️ Process Management (Ubuntu/Debian & CentOS/RHEL)

Category	Ubuntu/Debian Command	CentOS/RHEL Command	Description / Purpose
📋 Show All Running Processes	`ps aux`	`ps aux`	Displays all running processes with detailed information (user, CPU, memory, PID, etc.).
🔍 Find Specific Process	`ps aux	grep nginx`	`ps aux	`ps aux`
🗨️ Interactive Process Monitor	`top`	`top`	Displays live CPU, memory usage, and process details interactively.
⚡ Enhanced Process Monitor	`htop` *(install if needed)*	`htop` *(install if needed)*	A user-friendly, colorized, interactive process viewer (better than `top`).
💀 Kill Process by PID	`kill -9 PID`	`kill -9 PID`	Forcefully terminates a process using its Process ID (PID).
❌ Kill Process by Name	`pkill nginx`	`pkill nginx`	Terminates all processes matching a specific name (e.g., nginx).
🌳 View Process Tree	`pstree`	`pstree`	Displays processes in a hierarchical tree structure.
📉 Show Top CPU-Consuming Processes	`ps -eo pid,comm,%cpu,%mem --sort=-%cpu	head`	`ps -eo pid,comm,%cpu,%mem --sort=-%cpu`
💾 Show Top Memory-Consuming Processes	`ps -eo pid,comm,%cpu,%mem --sort=-%mem	head`	`ps -eo pid,comm,%cpu,%mem --sort=-%mem`
⏸️ Stop (Suspend) a Foreground Job	`Ctrl + Z`	`Ctrl + Z`	Suspends a running foreground job and places it in the background.

▶️ Resume Job in Background	`bg %1`	`bg %1`	Resumes the specified suspended job to run in the background.
⏩ Bring Job to Foreground	`fg %1`	`fg %1`	Brings a background job back to the foreground.
📜 List Background Jobs	`jobs -l`	`jobs -l`	Displays all current background and suspended jobs.
🧩 Show Process for a Specific User	`ps -u username`	`ps -u username`	Lists all processes owned by a specific user.
💬 View System Resource Summary	`vmstat 1`	`vmstat 1`	Displays system resource usage every 1 second (CPU, memory, I/O).
🖌️ Kill All Processes by User	`sudo pkill -u username`	`sudo pkill -u username`	Terminates all processes owned by the specified user.

🖥️ System Information Commands

Category	Ubuntu/Debian Command	CentOS/RHEL Command	Description
OS Version	`cat /etc/os-release`	`cat /etc/os-release`	Show OS information
Kernel Version	`uname -a`	`uname -a`	Show kernel version
CPU Info	`lscpu`	`lscpu`	Show CPU information
Memory Info	`free -h`	`free -h`	Show memory usage
Uptime	`uptime`	`uptime`	Show system uptime and load
Hardware Info	`lshw`	`lshw`	Show detailed hardware information
USB Devices	`lsusb`	`lsusb`	List USB devices

| PCI Devices | `lspci` | `lspci` | List PCI devices |

🐳 Docker & Container Management Commands

Category	Ubuntu/Debian Command	CentOS/RHEL Command	Description
Start Docker	`sudo systemctl start docker`	`sudo systemctl start docker`	Start Docker service
Enable Docker	`sudo systemctl enable docker`	`sudo systemctl enable docker`	Enable Docker on boot
Run Container	`docker run -d -p 80:80 nginx`	`docker run -d -p 80:80 nginx`	Run container in background
List Containers	`docker ps -a`	`docker ps -a`	List all containers
Stop Container	`docker stop container_id`	`docker stop container_id`	Stop a running container
Container Logs	`docker logs container_id`	`docker logs container_id`	Show container logs
Remove Container	`docker rm container_id`	`docker rm container_id`	Remove a container
Docker Images	`docker images`	`docker images`	List all Docker images

❇️ Troubleshooting & Logs Management Commands

Category	Ubuntu/Debian Command	CentOS/RHEL Command	Description
System Logs	`journalctl -xe`	`journalctl -xe`	View detailed system logs
Log Files	`tail -f /var/log/syslog`	`tail -f /var/log/messages`	Follow system logs in real-time
Error Search	`grep -i error /var/log/syslog`	`grep -i error /var/log/messages`	Search for "error" in log files
Disk Check	`sudo fsck /dev/sda1`	`sudo fsck /dev/sda1`	Check and repair file system
Memory Test	`sudo memtester 1G 1`	`sudo memtester 1G 1`	Test system RAM for errors
Service Debug	`systemctl status service-name`	`systemctl status service-name`	Check status and logs of a service
Network Debug	`ping -c 4 google.com`	`ping -c 4 google.com`	Test network connectivity
Port Check	`telnet host port`	`telnet host port`	Check if a specific port is open