Ubuntu Cli Cheatsheet

DavDev's Ubuntu CLI cheat sheet

System

System information

`uname -a` : Displays all system information. `hostnamectl` : Shows current hostname and related details. `lscpu` : Lists CPU architecture information. `timedatectl status` : Shows system time.

`ls` : Lists files and directories. `touch <filename>` : Creates an empty file or updates the last accessed date. `cp <source> <destination>` : Copies files from source to destination. `mv <source> <destination>` : Moves files or renames them. `rm <filename>` : Deletes a file.

System monitoring and management

top : Displays real-time system processes. htop : An interactive process viewer (needs installation). df -h : Shows disk usage in a human-readable format. free -m : Displays free and used memory in MB. kill : Terminates a process.

Files

File management

Directory navigation

`pwd` : Displays the current directory path. `cd <directory>` : Changes the current directory. `mkdir <dirname>` : Creates a new directory.

File permissions and ownership

Running commands

`[command] &` : Runs command in the background. `jobs` : Displays background commands. `fg <command number>` : Brings command to the foreground.

`chmod [who][+/-][permissions] <file>` : Changes file permissions. `chmod u+x <file>` : Makes a file executable by its owner. `chown [user]:[group] <file>` : Changes file owner and group.

Searching and finding

Service management

`sudo systemctl start <service>`: Starts a service. `sudo systemctl stop <service>` : Stops a service `sudo systemctl status <service>` : Checks the status of a service. `sudo systemctl reload <service>` : Reloads a service's configuration without interrupting its operation. `journalctl -f` : Follows the journal, showing new log messages in real time. `journalctl -u <unit_name>` : Displays logs for a specific systemd unit.

Cron jobs and scheduling

`crontab -e` : Edits cron jobs for the current user. `crontab -l` : Lists cron jobs for the current user.

`find [directory] -name <search_pattern>` : Finds files and directories. `grep <search_pattern> <file>` : Searches for a pattern in files.

Archiving and compression

tar -czvf <name.tar.gz> files : Compresses files into a tar.gz archive. tar -xvf <name.tar.[gz|bz|xz]> [destination] : Extracts a compressed tar archive.

Text editing and processing

`nano [file]` : Opens a file in the Nano text editor. `cat <file>` : Displays the contents of a file. `less <file>` : Displays the paginated content of a file. `head <file>` : Shows the first few lines of a file. `tail <file>` : Shows the last few lines of a file. `awk ‘{print}’ [file]` : Prints every line in a file.

Packages

Package management (APT)

`sudo apt install <package>` : Installs a package. `sudo apt install -f -reinstall <package>` : Reinstalls a broken package. `apt search <package>` : Searches for APT packages. `apt-cache policy <package>` : Lists available package versions. `sudo apt update` : Updates package lists. `sudo apt upgrade` : Upgrades all upgradable packages. `sudo apt remove <package>` : Removes a package. `sudo apt purge <package>` : Removes a package and all its configuration files.

`ip addr show` : Displays network interfaces and IP addresses. `ip -s link` : Shows network statistics. `ss -l` : Shows listening sockets. `ping <host>` : Pings a host and outputs results.

Netplan configuration (read more at netplan.io)

`cat /etc/netplan/*.yaml` : Displays the current Netplan configuration. `sudo netplan try` : Tests a new configuration for a set period of time. `sudo netplan apply` : Applies the current Netplan configuration.

Package management (Snap)

`snap find <package>` : Search for Snap packages. `sudo snap install <snap_name>` : Installs a Snap package. `sudo snap remove <snap_name>` : Removes a Snap package. `sudo snap refresh` : Updates all installed Snap packages. `snap list` : Lists all installed Snap packages. `snap info <snap_name>` : Displays information about a Snap package.

Firewall management

`sudo ufw status` : Displays the status of the firewall. `sudo ufw enable` : Enables the firewall. `sudo ufw disable` : Disables the firewall. `sudo ufw allow <port/service>` : Allows traffic on a specific port or service. `sudo ufw deny <port/service>` : Denies traffic on a specific port or service. `sudo ufw delete allow/deny <port/service>` : Deletes an existing rule.

SSH and remote access

`ssh <user@host>` : Connects to a remote host via SSH. `scp <source> <user@host>:<destination>` : Securely copies files between hosts.

Users & groups

User management

`w` : Shows which users are logged in. `sudo adduser <username>` : Creates a new user. `sudo deluser <username>` : Deletes a user. `sudo passwd <username>` : Sets or changes the password for a user. `su <username>` : Switches user. `sudo passwd -l <username>` : Locks a user account. `sudo passwd -u <username>` : Unlocks a user password. `Sudo change <username>` : Sets user password expiration date.

Firewall management

`sudo ufw status` : Displays the status of the firewall. `sudo ufw enable` : Enables the firewall. `sudo ufw disable` : Disables the firewall. `sudo ufw allow <port/service>` : Allows traffic on a specific port or service. `sudo ufw deny <port/service>` : Denies traffic on a specific port or service. `sudo ufw delete allow/deny <port/service>` : Deletes an existing rule.

Group management

`id [username]` : Displays user and group IDs. `groups [username]` : Shows the groups a user belongs to. `sudo addgroup <groupname>` : Creates a new group. `sudo delgroup <groupname>` : Deletes a group.

Ubuntu LXD

LXD is a modern, secure and powerful tool that provides a unified experience for running and managing containers or virtual machines. Visit https ://canonical.com/lxd for more information.

`lxd init` : initializes LXD before first use

Creating instances

`lxc init ubuntu:22.04 <container name>` : Creates a lxc system container (without starting it). `lxc launch ubuntu:24.04 <container name>` : Creates and starts a lxc system container. `lxc launch ubuntu:22.04 <vm name> --vm` : Creates and starts a virtual machine.

Managing instances

`lxc list` : Lists instances. `lxc info <instance>` : Shows status information about an instance. `lxc start <instance>` : Starts an instance. `lxc stop <instance> [--force]` : Stops an instance. `lxc delete <instance> [--force|--interactive]` : Deletes an instance.

Accessing instances

`lxc exec <instance> -- <command>` : Runs a command inside an instance. `lxc exec <instance> -- bash` : Gets shell access to an instance (if bash is installed). `lxc console <instance> [flags]` : Gets console access to an instance. `lxc file pull <instance>/<instance_filepath> <local_filepath>` : Pulls a file from an instance. `lxc file pull <local_filepath> <instance>/<instance_filepath>` : Pushes a file to an instance.

Using projects

`lxc project create <project> [--config <option>]` : Creates a project. `lxc project set <project> <option>` : Configures a project. `lxc project switch <project>` : Switches to a project.

Pipewire

`systemctl --user restart pipewire.service` : Restarts the Pipewire service.