System Management

CompTIA Linux+

KNOWLEDGE CHEATSHEET

LINUX OPERATING LAYERS

Hardware Devices → **Kernel** Storage & network → **systemd** Services → **Workloads** Apps & containers

01 System Management 32%

Filesystem layout and hardware

- **/etc** holds configuration; **/var** holds changing data and logs; **/home** holds user homes; **/opt** holds add-on software. **/boot** holds kernel and boot artifacts; **/proc** and **/sys** expose kernel information.
- **/bin** and **/usr/bin** contain user commands; **/sbin** and **/usr/sbin** contain system administration tools. Location alone does not restrict execution; permissions and privilege checks do. Some distributions merge these paths.
- **uname -r** reports the running kernel; **uname -m** reports architecture. **/etc/os-release** identifies the distribution. **lscpu**, **lspci**, **lsusb**, **lsscsi** and **lshw** inventory CPU, buses and devices.
- **lsmod** lists loaded modules; **modinfo** shows module metadata. **modprobe** loads a module with dependencies; **insmod** inserts a specified module file. **dmesg** reveals kernel device and driver messages.
- **ln -s TARGET LINK** creates a symbolic link that names another path; it can cross filesystems and can become dangling. A **hard link** names the same inode and normally must stay within one filesystem.

Boot chain and recovery tools

- **BIOS or UEFI → bootloader → kernel and initramfs → root filesystem → init/systemd** is the startup chain. The bootloader selects and loads a kernel; systemd starts userspace services.
- **GRUB** can select another installed kernel or temporarily edit kernel arguments. On legacy BIOS/MBR systems, **grub-install /dev/sdX** targets the whole boot disk. UEFI instead uses an EFI System Partition, commonly mounted at **/boot/efi**.
- **/etc/default/grub** supplies settings for generated GRUB configuration. Regenerate configuration with the distribution's GRUB tooling after persistent changes; do not assume a single output path fits every distribution.
- **vmlinuz** is a kernel image; **initramfs** supplies early userspace and drivers needed to reach root storage. **dracut** or **update-initramfs** rebuilds it on applicable systems. Kernel packages can restore missing kernel files.
- **systemd.unit=rescue.target** requests maintenance mode; **emergency.target** provides a more minimal environment. Broken kernel or bootloader files may require rescue media, mounting the installed system and a **chroot** repair.

System Management

CompTIA Linux+

KNOWLEDGE CHEATSHEET

LINUX OPERATING LAYERS

Hardware Devices → **Kernel** Storage & network → **systemd** Services → **Workloads** Apps & containers

Service lifecycle and targets

- **systemctl status UNIT** shows state and recent messages. **start/stop** changes current operation; **enable/disable** changes boot activation. **enable --now** enables and starts; **disable --now** disables and stops.
- **mask** blocks activation by linking a unit to **/dev/null**; it does not stop an already running service unless combined with **--now**. **unmask** removes that block; starting the service remains a separate action.
- **daemon-reload** rereads unit definitions after edits. **reload UNIT** asks a supporting application to reread its configuration; **restart UNIT** stops and starts it. These operations serve different purposes.
- **multi-user.target** provides normal non-GUI operation; **graphical.target** adds the graphical login environment. **isolate** changes the current target; **set-default** changes the boot target; **get-default** displays it. Traditional runlevels 3 and 5 commonly map to these targets.

Unit definitions and dependencies

- Put local units and overrides under **/etc/systemd/system**; use **systemctl edit UNIT** for a drop-in. **[Unit]** describes dependencies, **[Service]** defines execution, and **[Install]** defines enablement relationships.
- **ExecStart=** names the service command. **User=** and **Group=** select its identity; **EnvironmentFile=** loads settings. **Type=oneshot** suits a finite task; **Restart=on-failure** requests recovery after eligible failures.
- **After=database.service** orders startup when both units participate; **Requires=database.service** pulls in a stronger dependency. Use both when the application needs the database started first. Ordering alone does not guarantee application readiness.
- **Wants=** is a weaker dependency; **WantedBy=multi-user.target** enables association with that target. **RequiresMountsFor=/path** adds mount requirements and ordering for a path. **TimeoutStartSec=** controls startup timeout.

Packages, repositories and source builds

- **dnf install/update/remove PACKAGE** and legacy **yum** manage RPM systems with dependency resolution. **rpm -qa** lists installed packages, **rpm -qf /path** finds a file's owning package, **rpm -ql PACKAGE** lists files, and **rpm -V PACKAGE** checks installed file metadata.
- **rpm -i FILE.rpm** installs a local RPM; **rpm -Uvh FILE.rpm** upgrades or installs; **rpm -e PACKAGE** erases. Repository definitions commonly live in **/etc/yum.repos.d/*.repo**; review signatures and repository trust.
- **apt update** refreshes package indexes; **apt upgrade** installs available upgrades. **apt install ./app.deb** can resolve dependencies; **dpkg -i app.deb** installs a local package without automatically fetching dependencies. **dpkg -l** and **snap list** list their installed packages.
- APT repository configuration uses **/etc/apt/sources.list** and files under **sources.list.d**. **dnf history** shows transactions; **history undo** attempts reversal subject to package availability. **.rpmnew** preserves a new packaged config beside a local one; **.rpmsave** preserves an older config.

System Management

CompTIA Linux+

KNOWLEDGE CHEATSHEET

LINUX OPERATING LAYERS

Hardware Devices → **Kernel** Storage & network → **systemd** Services → **Workloads** Apps & containers

- A common source build sequence is **./configure** → **make** → **make install**; actual build instructions take precedence. Kernel modules must match the target kernel's build environment. **rpmdb --rebuilddb** rebuilds an RPM database; it does not reinstall damaged package files.
- YUM/DNF **exclude=PATTERN** configuration can exclude matching packages from ordinary transactions; repository and command options can affect the result. Inspect the proposed transaction before applying an update. **wget URL** or **curl -O URL** downloads a remote file.

Disks, partitions and filesystems

- **lsblk -f** shows block-device hierarchy and filesystem identifiers. **fdisk -l** or **parted -l** lists partition tables; **blkid** reports UUIDs and types. **partprobe** asks the kernel to reread partition tables after changes.
- **MBR** supports four primary partition entries, with an extended partition providing logical partitions. **GPT** supports more partitions and larger disks. A partition such as **/dev/sdb1** differs from its parent disk **/dev/sdb**.
- **mkfs.ext4 DEVICE** and **mkfs.xfs DEVICE** create filesystems and overwrite existing filesystem structures. **mount DEVICE /mountpoint** attaches one; **umount /mountpoint** detaches it. Create the mountpoint directory first.
- **ext4** can grow and can shrink offline with appropriate tools; **XFS** grows while mounted using **xfs_growfs /mountpoint** and cannot be shrunk. **fstrim** informs supported SSD storage which filesystem blocks are unused.

Persistent mounts and volume growth

- **/etc/fstab** fields are **source, mountpoint, type, options, dump, fsck pass**. Example: **UUID=… /data ext4 defaults 0 2**. UUIDs and labels avoid reliance on disk enumeration; cloned filesystems may need unique identifiers.
- **findmnt** inspects mounted filesystems; **mount -a** attempts configured mounts and exposes many fstab errors before reboot. **ro/rw**, **noexec**, **nosuid** and **nodev** affect access and execution independently of ordinary file modes.
- **LVM: physical volume** → **volume group** → **logical volume** → **filesystem**. **pvs**, **vgs** and **lvs** inspect each layer; **pvdisplay**, **vgdisplay** and **lvdisplay** give details. **lvs VG** lists logical volumes in a group.
- Add storage with **pvcreate /dev/sdX**, then **vgextend VG /dev/sdX**. Grow a logical volume with **lvextend -r -L +10G /dev/VG/LV** or **lvextend -r -l +100%FREE /dev/VG/LV**; **-r** also requests filesystem resizing through supported tools.
- When an existing disk grows, resize the partition if present, then the PV with **pvresize**, then the LV and filesystem. **vgchange -ay VG** or **lvchange -ay /dev/VG/LV** activates volumes; activation does not repair missing storage.

System Management

CompTIA Linux+

KNOWLEDGE CHEATSHEET

LINUX OPERATING LAYERS

Hardware Devices → **Kernel** Storage & network → **systemd** Services → **Workloads** Apps & containers

RAID, backups and archives

- **RAID 0** stripes without redundancy; **RAID 1** mirrors; **RAID 5** tolerates one failed member; **RAID 6** tolerates two; **RAID 10** combines mirrors and stripes. RAID 10 survival depends on which mirror members fail. RAID does not replace backups.
- **dd if=SOURCE of=DESTINATION** copies raw blocks. Reversing these arguments destroys the wrong data; verify device names and destination capacity. Use a quiescent source or a consistent snapshot for dependable disk images.
- **rsync -a -e ssh /data/ host:/backup/data/** copies directory contents with archive metadata handling. Without the source trailing slash, the directory itself is copied. Preview **--delete** with **--dry-run** because it removes destination extras.
- **tar -czf backup.tar.gz DIR** creates a compressed archive; **tar -xzf** extracts; **tar -tf** lists. Compression flags include **-z** gzip, **-j** bzip2 and **-J** xz. Inspect paths before extraction.
- **gzip -c FILE > FILE.gz** retains the original; **gunzip** decompresses. **unzip** handles ZIP archives. **cpio -o** creates an archive from a filename list; **cpio -i** extracts. Use null-delimited filenames when supported to handle unusual names.

Interfaces, addresses and routes

- **ip link show dev eth0** reports link state; **ip addr show dev eth0** reports assigned addresses and prefixes. **ip -s link** shows counters; **ethtool eth0** checks link negotiation. Administrative **UP** differs from detected carrier.
- **ip addr add 192.0.2.10/24 dev eth0** adds a runtime address; **ip link set eth0 up** enables the interface. IPv4 prefix lengths range from **/0 to /32**. **/24** corresponds to **255.255.255.0**; **/32** identifies one host.
- **ip route** shows routing; **ip route get DESTINATION** shows the selected path. **ip route add 10.0.6.0/24 via GATEWAY dev eth0** adds a runtime route. The default route handles destinations without a more specific match.
- Runtime **ip** changes usually disappear after reboot. Persist settings through the active network manager, such as **nmcli connection modify NAME +ipv4.routes "PREFIX GATEWAY"**, then apply the profile. Legacy ifcfg files and systemd-networkd use different configuration formats.

Names, time and network services

- **/etc/hosts** provides static name mappings; **/etc/nsswitch.conf** controls lookup order. **/etc/resolv.conf** supplies resolver settings, often generated by a manager. **getent hosts NAME** tests the system lookup path; **dig** queries DNS directly.
- **dig @SERVER example.org MX** asks a chosen resolver for mail exchangers; **dig -x ADDRESS** queries reverse PTR records. **A/AAAA** map names to IPv4/IPv6, **CNAME** aliases names, **NS** identifies nameservers and **MX** identifies mail exchangers.

System Management

CompTIA Linux+

KNOWLEDGE CHEATSHEET

LINUX OPERATING LAYERS

Hardware Devices → **Kernel** Storage & network → **systemd** Services → **Workloads** Apps & containers

- **SOA** carries zone authority metadata; **TXT** carries text; **SRV** describes service location; **RRSIG** contains a DNSSEC signature. Multiple address records can spread requests, but DNS round robin alone does not remove unhealthy servers.
- **hostnamectl set-hostname NAME** changes the system hostname; it does not create DNS records. **timedatectl set-timezone REGION/CITY** changes the time zone. NTP synchronization through **chronyd** or **ntpd** is separate from time-zone selection.

Virtual machines and provisioning

- **KVM** uses Linux kernel virtualization with userspace tools such as QEMU and libvirt. Check CPU virtualization features, firmware enablement and loaded modules. **modprobe kvm** handles module dependencies; vendor modules such as **kvm_intel** or **kvm_amd** provide hardware support.
- **Nested virtualization** requires hypervisor exposure of virtualization extensions to the guest. Merely installing KVM inside a VM does not establish that support. **virsh** manages libvirt domains; **virt-install** creates guests.
- **cloud-init** applies instance metadata and user data during provisioning, including users, keys, packages and startup commands. Treat user data as configuration with access controls; avoid embedding reusable secrets.
- **VMs** have guest kernels and virtual hardware; **containers** share the host kernel and isolate processes with namespaces and resource controls. Cloning a host requires reviewing machine identity, SSH host keys, addresses and filesystem identifiers.

Container images and runtime

- **docker pull IMAGE:TAG** retrieves an image; **build -t NAME:TAG .** builds from a context; **tag OLD NEW** adds a reference; **push** uploads. A new tag does not change image contents or remove the old tag.
- A **Dockerfile** commonly begins with **FROM**; **WORKDIR** selects a working directory; **COPY** adds files; **RUN** executes build steps. **ENTRYPOINT ["/startScript.sh"]** defines the executable; **CMD** can supply default arguments overridden at runtime.
- **docker run -d --name web -p 8080:80 IMAGE** creates a detached container and maps **host:container** ports. **EXPOSE** documents a container port; it does not publish it on the host. Podman provides comparable commands for many tasks.
- **-v /host/data:/container/data** binds persistent host data; named volumes also persist independently of container replacement. **--restart=always** sets a restart policy; host startup still needs the container runtime available. Keep credentials outside image layers.

Container inspection and orchestration

- **docker ps** lists running containers; **ps -a** includes stopped ones. **inspect** displays configuration, mounts, port mappings and startup details; **logs** reads captured stdout/stderr; **stats** reports resource usage.
- **docker exec -it NAME /bin/sh** enters a running container with a new process. **docker cp NAME:/path ./file** copies out a file. **run** creates a new container; it does not enter an existing one.

System Management / Security

CompTIA Linux+

KNOWLEDGE CHEATSHEET

ACCESS DECISION LAYERS

- **Identity** — Users & groups
- **Permissions** — Modes & ACLs
- **Policy** — SELinux & sudo
- **Network** — SSH & firewall

- **docker rm** removes containers; **rmi** removes image references. **container prune** removes stopped containers; **image prune -a** removes images unused by containers; **network prune** removes unused eligible networks. Review scope before cleanup.
- **Kubernetes** schedules and reconciles workloads across nodes. A **Pod** groups containers; a **Deployment** manages replicas; a **Service** provides stable discovery; **ConfigMaps/Secrets** supply configuration; **PersistentVolumeClaims** request persistent storage.
- A **sidecar** supports an application, such as shipping logs; an **ambassador** proxies external communication. A **service mesh** can provide service traffic policies, mutual TLS and telemetry. High availability also requires replicas, healthy nodes and suitable storage.

02 Security 21%

Accounts, groups and login policy

- **/etc/passwd** stores account identity, UID/GID, home and shell; **/etc/shadow** stores protected password hashes and aging data. **/etc/group** stores local groups. **getent passwd** includes accounts supplied by configured directory services; **id USER** shows group membership.
- **useradd -m -d /home/project USER** creates a home at a selected path; **-b** selects a base path. **usermod -aG GROUP USER** appends supplementary membership; **-G** without **-a** replaces it. Existing sessions need new credentials, usually by logging in again.
- **/etc/skel** supplies new-home defaults; **/etc/default/useradd** and **/etc/login.defs** control applicable creation defaults. **usermod -s /sbin/nologin USER** blocks an interactive login shell. Some services also validate shells against **/etc/shells**.
- **passwd USER** changes a local password; **passwd -S USER** reports status. **chage -l USER** shows aging; **chage -d 0 USER** requires a password change; **useradd -e YYYY-MM-DD USER** sets account expiration; **chage -E -1 USER** removes that expiration.
- **passwd -l** locks password authentication; it does not necessarily block SSH keys or all other access. PAM failure locks are separate: inspect/reset with **faillock** or legacy **pam_tally2** as configured. Check shell, expiration and directory-backed identity before changing credentials.

PAM and centralized authentication

- **PAM** stacks authentication, account, password and session modules under **/etc/pam.d**. Control flags such as **required**, **requisite** and **sufficient** change stack evaluation; the module's presence alone does not describe the complete policy.
- **pam_nologin** can deny non-root logins while **/etc/nologin** exists. **pam_access** applies access policy; configured MFA modules add factors. **MFA** combines independent factor types; **SSO** reduces repeated sign-ins across services.

Security

CompTIA Linux+

KNOWLEDGE CHEATSHEET

ACCESS DECISION LAYERS

- **Identity** — Users & groups
- **Permissions** — Modes & ACLs
- **Policy** — SELinux & sudo
- **Network** — SSH & firewall

- **LDAP** centralizes directory users and groups; **SSSD** can integrate directory identity and authentication on Linux. **Kerberos** supplies ticket-based authentication. Name lookup, credential validation and authorization remain separate checks.
- **LDAPS** or LDAP with **StartTLS** protects directory traffic. Validate the certificate's trust chain and hostname, and check time validity. Reachable TCP alone does not establish successful TLS authentication.

Privilege delegation and hardening

- **sudo** runs an authorized command as another identity; **sudo -u USER COMMAND** selects a user. **su - USER** starts a login environment and changes to that user's home; plain **su USER** preserves more of the invoking environment.
- Edit sudo policy with **visudo** and validate with **visudo -c**. **/etc/sudoers.d** supports modular policy when included. **%GROUP** denotes a group; **USER ALL=(root) /absolute/command** grants a specific root command.
- Membership in **wheel** or **sudo** grants privilege only if policy authorizes it. Prefer exact commands and controlled arguments; **NOPASSWD:** removes a password prompt for matching commands and does not independently grant extra commands.
- Remove unnecessary packages, disable unused services, patch supported software and run daemons under dedicated accounts. A **GRUB password** restricts unauthorized boot-entry editing; full-disk encryption and physical controls address broader offline access risks.

Mode bits and shared directories

- **r=4, w=2, x=1** apply separately to owner, group and others. **chmod 750 script.sh** grants owner rwx, group r-x and others none. **chown USER:GROUP FILE** changes ownership; **chgrp GROUP FILE** changes the group.
- Directory **r** lists names, **x** permits traversal and **w+x** permits entry creation/removal. Deleting or renaming a file chiefly depends on its parent directory, rather than the file's write permission. Every parent in a path needs traversal permission.
- **umask** removes permissions from the creation mode: normally **0666 & ~mask** for files and **0777 & ~mask** for directories. **0027** gives typical defaults **0640/0750**; **0077** gives **0600/0700**. It does not change existing files or add execute bits.
- **SUID (4000)** runs a supported executable with its owner's effective UID; **SGID (2000)** similarly selects its group. SGID on a directory makes new entries inherit its group. It does not itself grant group write permission.
- The **sticky bit (1000)** restricts deletion/renaming in a writable directory to the file owner, directory owner or privileged process. **chmod 1777 DIR** combines it with broad directory access. It does not stop someone writing an otherwise writable file.

Security

CompTIA Linux+

KNOWLEDGE CHEATSHEET

ACCESS DECISION LAYERS

- **Identity** — Users & groups
- **Permissions** — Modes & ACLs
- **Policy** — SELinux & sudo
- **Network** — SSH & firewall

ACLs and extended attributes

- **getfacl PATH** displays access and default ACLs; **setfacl -m u:USER:rwx DIR** or **-m g:GROUP:rw FILE** adds named access without replacing the owning group. Directories usually need **x** for useful access.
- The **ACL mask** caps effective rights of named users and group-class entries. A **default ACL** on a directory supplies inherited ACLs for new children; it does not grant access to the parent itself or retroactively alter existing children.
- **setfacl -x u:USER FILE** removes one named entry; **setfacl -b FILE** removes extended access ACL entries. Check the displayed effective permissions and mask when an apparently permissive ACL still fails.
- **lsattr** shows filesystem attributes. **chattr +i FILE** makes it immutable until an authorized process clears **-i**; **+a** permits append-only writes on supporting filesystems. Neither replaces backup or access control, and privileged users may clear the attributes.

SSH authentication and trust

- **ssh-keygen -t rsa -b 4096** generates a 4096-bit RSA pair; a passphrase protects the private key. **ssh-copy-id -i ~/.ssh/id_rsa.pub USER@HOST** installs the public key. The private key stays on the client; the server verifies proof of possession.
- **~/.ssh/authorized_keys** authorizes incoming user keys; **~/.ssh/known_hosts** records trusted server identities. **/etc/ssh/ssh_host_*_key** files are server host private keys. These files solve different trust problems.
- Typical secure permissions are **0700** for **~/.ssh**, **0600** for private keys and **authorized_keys**, with correct ownership and no unsafe parent-directory writes. On SELinux systems, restore appropriate labels with **restorecon** when needed.
- **ssh -i KEY -p PORT USER@HOST** chooses identity and port. **ssh-agent/ssh-add** hold unlocked identities for reuse; an agent passphrase prompt differs from a server password prompt. **ssh -v** exposes key selection, signature compatibility and authentication failures.

Security

CompTIA Linux+

KNOWLEDGE CHEATSHEET

ACCESS DECISION LAYERS

- **Identity** — Users & groups
- **Permissions** — Modes & ACLs
- **Policy** — SELinux & sudo
- **Network** — SSH & firewall

SSH server, clients and forwarding

- **/etc/ssh/sshd_config** configures the server; **/etc/ssh/ssh_config** and **~/.ssh/config** configure clients. A client **Host** stanza can set **HostName**, **User**, **Port** and **IdentityFile** for a named destination.
- **PermitRootLogin no** denies direct root SSH login; **prohibit-password** permits supported non-password methods for root. To require user keys, enable public-key authentication and disable both password and keyboard-interactive alternatives as appropriate to the PAM policy.
- Validate server syntax with **sshd -t** before applying changes. Check service reload support, listening sockets, firewall rules and SELinux port labels. Preserve an existing administrative session while testing a new connection.
- **ssh -L 8080:localhost:80 USER@SERVER** exposes a local listener forwarded to the server's loopback port 80. **ssh -R 8000:localhost:8000 USER@SERVER** creates a remote listener forwarded to the client's local port 8000, subject to server policy.
- **ssh -X USER@SERVER APPLICATION** enables supported X11 forwarding; the SSH session sets **DISPLAY**. **-x** disables it. **scp** and **sftp** transfer files over SSH; file transfer alone does not install an authentication key in the right location.

Firewalls and packet translation

- **iptables** filter chains are **INPUT** for local delivery, **OUTPUT** for locally generated traffic and **FORWARD** for routed traffic. Rules are ordered; an earlier matching verdict can prevent a later rule from taking effect.
- **iptables -A INPUT -p tcp --dport 443 -j ACCEPT** permits matching inbound traffic; **-s ADDRESS** limits the source. **DROP** silently discards; **REJECT** sends an error. **-L -n -v** lists rules/counters; **-D** deletes; **-F** flushes rules without resetting default policies.
- Stateful rules use **-m conntrack --ctstate ESTABLISHED,RELATED** to recognize tracked flows. In the **nat** table, **DNAT** commonly changes destinations in **PREROUTING**; **SNAT/MASQUERADE** commonly changes source addresses in **POSTROUTING**.
- **sysctl -w net.ipv4.ip_forward=1** enables IPv4 routing now. Persist **net.ipv4.ip_forward=1** in **/etc/sysctl.d/*.conf** and apply with **sysctl --system**. Routing also needs valid routes and a permitting FORWARD policy; NAT serves a separate purpose.

Security

CompTIA Linux+

KNOWLEDGE CHEATSHEET

ACCESS DECISION LAYERS

- **Identity** — Users & groups
- **Permissions** — Modes & ACLs
- **Policy** — SELinux & sudo
- **Network** — SSH & firewall

Managed firewall policies

- **firewall-cmd --get-active-zones** identifies active interface/source bindings. **--zone=public --list-all** shows a zone's configuration; **--list-services** lists allowed services; **--get-services** lists available service definitions, not current allowances.
- **firewall-cmd --zone=public --add-port=2222/tcp** changes runtime rules. Add **--permanent** to edit persistent configuration; **--reload** applies persistent rules and replaces unsaved runtime settings. **--runtime-to-permanent** saves the current runtime configuration.
- **--add-service=ssh** uses the service definition, normally TCP 22; a custom SSH port needs the matching explicit port rule or service definition. **--query-service=http** checks the selected zone. Enabling firewalld at boot is separate from saving its rules.
- **ufw status** displays policy; **ufw allow 22/tcp** allows SSH; **ufw allow from 10.10.0.0/24 to any port 22 proto tcp** limits sources. **ufw delete allow 22/tcp** removes the corresponding rule. DNS needs UDP and TCP 53; NTP commonly uses UDP 123.

SELinux and data protection

- **SELinux** adds mandatory access control to ordinary permissions. **getenforce/sestatus** report state; **setenforce 0/1** changes permissive/enforcing mode at runtime. **/etc/selinux/config** controls boot configuration; permissive mode logs denials without enforcing them.
- Inspect contexts with **ls -Z** and denials with **ausearch -m AVC**. **semanage fcontext** defines persistent path-label mappings; **restorecon** applies them. **chcon** changes a current label that later relabeling can replace.
- **semanage port -a -t ssh_port_t -p tcp 2222** authorizes an otherwise unassigned port type; existing mappings may need modification. **setsebool -P httpd_can_network_connect_db on** permits the supported web-to-database behavior persistently. Correct the matching policy cause rather than broadly disabling enforcement.
- **LUKS** provides a standard encrypted-volume format over **dm-crypt**. For a new volume: **cryptsetup luksFormat DEVICE** → **cryptsetup open DEVICE NAME** → **mkfs on /dev/mapper/NAME** → **mount**. Formatting destroys prior data; back up before conversion. **/etc/crypttab** supports boot-time unlocking.
- **sha256sum FILE** computes an integrity digest; compare it with a trusted expected digest. **gpg --verify** can validate a signature and signer relationship when the key is trusted. Hashing is not encryption; **TLS** protects data in transit and LUKS protects stored blocks.

Scripting & Automation

CompTIA Linux+

KNOWLEDGE CHEATSHEET

REPEATABLE DELIVERY

Define Scripts & IaC → **Version** Git → **Validate** Tests & plans → **Apply** Jobs & pipelines

03 Scripting, Containers and Automation 19%

Shell execution and environment

- A script's **shebang**, such as **#!/bin/bash**, selects its interpreter when executed directly. It needs execution permission and traversable parent directories; an interpreter can read a script without its execute bit, but other access controls still apply.
- **$?** holds the last exit status: **0** conventionally means success; nonzero means failure. **$1** is the first positional argument, **$#** counts arguments and **"$@"** preserves each argument. Quote expansions such as **"$filename"** to prevent unintended splitting and globbing.
- **export NAME=value** passes a variable to child processes. **PATH** is a colon-separated command search list; append with **export PATH="$PATH:/opt/app/bin"**. **source ~/.bashrc** applies definitions in the current shell, including aliases.
- **~/.bashrc** usually configures interactive non-login Bash shells; login profiles and **/etc/profile** serve login environments. Scheduled jobs may have a minimal environment, so use explicit paths and required variable definitions.
- **>** overwrites stdout, **>>** appends, **2>** redirects stderr, and **2>&1** duplicates the current stdout target. **COMMAND >log 2>&1** captures both streams; **|** pipes stdout and **tee** writes output to a file while displaying it.

Conditions, loops and safe filenames

- **if [-f "$file"]; then ...; fi** tests for a regular file; **-d** tests a directory; **-e** tests existence; **-r/-w/-x** test access. Bracket syntax needs spaces; use **-eq/-gt/-lt** for integer comparisons.
- **for**, **while** and **case** express repetition and selection. **while IFS= read -r line; do ...; done < list** reads lines without losing backslashes or splitting whitespace. Use an inner loop only when each item truly needs every combination.
- **find /var/log -maxdepth 1 -type f -name '*.log'** searches only that directory. **-mtime +180** matches files whose age in rounded-down 24-hour periods exceeds 180; **-size +100M** tests size; **-xdev** stays on one filesystem.
- **find ... -exec COMMAND {} +** passes matched names safely in batches. **-print0 | xargs -0 COMMAND** handles embedded spaces and newlines. Preview matches before deletion; ordinary whitespace-delimited **xargs** can misinterpret filenames.

Text processing and editing

- **grep -i** ignores case, **-l** lists matching filenames, **-n** shows line numbers and **-v** inverts the match. **grep -E 'pattern1|pattern2'** supports extended alternation; shell globs and regular expressions have different syntax.
- **awk -F: '$3 > 500 {print $1}' /etc/passwd** filters a numeric field; **cut -d: -f1,3** selects fields. Simple delimiter tools do not fully parse CSV containing quoted delimiters or embedded newlines.

Scripting & Automation

CompTIA Linux+

KNOWLEDGE CHEATSHEET

REPEATABLE DELIVERY

Define Scripts & IaC → **Version** Git → **Validate** Tests & plans → **Apply** Jobs & pipelines

- **sort | uniq -c** counts adjacent equal lines after sorting. **sort -n** uses numeric order. **tr '[:lower:]' '[:upper:]' < input > output** transforms case; use a different output path to avoid truncating the input.
- **sed 's/OLD/NEW/g' FILE** substitutes text; **sed -i.bak** can edit while retaining a backup. Escape regex metacharacters, such as dots in literal IP addresses. In **vi**, **i** inserts, **A** appends at line end, **Esc** returns to command mode and **:wq** saves and exits.
- **head -n N** reads the beginning; **tail -n 20** reads the last 20 lines; **tail -f** follows growth and **tail -F** can follow a replaced filename. **file PATH** identifies content type; **cp** copies, **mv** moves, **mv -i** prompts before overwrite and **mv -n** skips overwriting.

Scheduling and repeatable jobs

- A user crontab has **minute hour day-of-month month day-of-week command**. **30 1 * * * /path/backup.sh** runs daily at 01:30. System crontabs and **/etc/cron.d** entries add a username before the command.
- **0 9 * * 6,0 root /root/script.sh** is a system-crontab weekend job. **0 */2 * 5 0-2 COMMAND** runs every two hours Sunday through Tuesday in May. Commas list values, hyphens define ranges and slashes define steps.
- **crontab -e/-l** edits/lists a user's schedule. **cron.allow/cron.deny** can restrict use according to the implementation. For a one-time job, **at -f /path/script.sh 21:00** submits the file to **atd**; **atq/atrm** list/remove jobs.
- A systemd **.timer** activates a service. **OnCalendar=*-*-* 00:00:00** schedules midnight daily; **OnUnitActiveSec=2h** uses an elapsed interval. **Persistent=true** can catch up missed calendar activations; **RandomizedDelaySec=** adds delay.
- After timer edits, run **systemctl daemon-reload**, enable/start the timer and inspect **systemctl list-timers** plus the service journal. **systemd-analyze calendar** checks calendar expressions. Running the service manually does not prove the timer schedule is correct.

Git workflow and sensitive files

- **git clone URL** creates a local repository copy. **git fetch** refreshes remote-tracking information without integrating it; **git pull** fetches and integrates according to configuration. **git branch -a** shows local and remote-tracking branches.
- **git checkout -b feature** or **git switch -c feature** creates and selects a branch. **git add FILE → git commit -m "message" → git push** stages, records and publishes changes. A push publishes the selected ref; it does not automatically merge a feature branch into main.

Scripting & Automation / Troubleshooting

CompTIA Linux+

KNOWLEDGE CHEATSHEET

REPEATABLE DELIVERY

Define Scripts & IaC → **Version** Git → **Validate** Tests & plans → **Apply** Jobs & pipelines

- **git status**, **git diff** and **git log** inspect state, changes and history. **git merge** integrates branches; **git rebase** reapplies commits on a new base; **git tag** labels a commit. Review and test changes before merging through a pull request.
- **.gitignore** excludes matching untracked files, such as state files or local configuration. It does not untrack committed files; **git rm --cached FILE** removes a tracked entry while retaining the working file. Exposed secrets require rotation and appropriate history cleanup.

Infrastructure and configuration as code

- **Infrastructure as code** describes reproducible infrastructure in version-controlled definitions. **Terraform** provisions resources through providers: **init** prepares a workspace, **plan** previews changes and **apply** executes them. Protect state because it may contain sensitive data.
- **Ansible** uses inventory, modules and YAML playbooks, commonly over SSH without a target agent. Desired-state tasks aim for **idempotence**: repeated runs should not create unnecessary changes. **Puppet** uses manifests and **Chef** uses recipes for configuration management.
- **YAML** is indentation-based serialization with mappings and lists; use spaces consistently. **JSON** uses objects, arrays, quoted keys and explicit delimiters. **HCL** is Terraform's configuration syntax. These data formats are distinct from shell programs.
- **CI/CD** automates build, test and delivery/deployment steps; **Jenkins** is a common pipeline tool. Keep repeatable definitions in source control, validate changes before rollout and inject environment-specific settings through controlled configuration.

04 Troubleshooting 28%

Evidence, logs and systematic diagnosis

- **Identify scope → gather evidence → form and test a hypothesis → implement a controlled fix → verify and document**. Correlate timestamps, recent changes and affected users before altering a system. A reboot may hide evidence without resolving the cause.
- **journalctl -b** shows the current boot; **-b -1** shows the previous retained boot; **-u UNIT** filters a unit; **-k** filters kernel messages; **-f** follows events. **journalctl -xb** adds explanatory catalog text for the current boot.

Troubleshooting

CompTIA Linux+

KNOWLEDGE CHEATSHEET

DIAGNOSE WITH EVIDENCE

Scope Symptoms → **Observe** Logs & metrics → **Test** Hypothesis → **Verify** Lasting fix

- **/etc/systemd/journald.conf** configures journal storage. **Storage=persistent** uses durable journal storage; **volatile** uses **/run/log/journal**. **auto** persists when **/var/log/journal** exists. Apply supported configuration changes by restarting journald.
- **journalctl --rotate** archives active journals; **--vacuum-time=7d** or **--vacuum-size=SIZE** removes eligible archived files. **logrotate** manages conventional log files. **/var/log/secure** or **auth.log**, **last**, **lastlog** and **loginctl list-sessions** provide different authentication/session evidence.

Processes, signals and job control

- **ps -ef**, **ps aux**, **top** and **htop** inspect processes. **top -u USER** narrows by account; **pstree -p** shows parent relationships; **lsof -p PID** shows opened files; **strace -p PID** traces system calls when permissions allow.
- **SIGTERM (15)** requests graceful termination; **SIGKILL (9)** forces termination without process cleanup; **SIGHUP (1)** often requests configuration reload, depending on the application. **kill** targets PIDs; **pkill -f** matches full command lines using a regex.
- A **Z/defunct** process has exited but its parent has not reaped its status. It cannot be fixed by killing the zombie again; diagnose and safely restart or repair the parent. A process in uninterruptible **D** state may remain until kernel I/O completes.
- **Ctrl+Z** suspends a foreground job; **bg** resumes it in the background; **fg** returns it to the foreground; **jobs** lists shell jobs. **COMMAND &** starts in background; **nohup**, **disown**, **screen** or **tmux** address different session-lifetime needs.
- **nice -n 10 COMMAND** starts with lower scheduling priority; **renice -n 15 -p PID** changes a running process. Nice values range **-20 to 19**; lower values mean higher priority and normally require privilege. Niceness changes do not cure a memory leak or add CPU capacity.

CPU, memory and kernel tuning

- **Load average** measures runnable and uninterruptible tasks over 1, 5 and 15 minutes; compare with CPU count and wait metrics. A high load alone does not prove CPU saturation. **top** distinguishes user, system, idle, I/O wait and hypervisor **steal** time.
- **free -h** shows memory and swap; focus on **available** memory as well as cache. **vmstat 1** shows run queues, paging and CPU states; sustained **si/so** suggests active swapping. High steal time points toward contention for guest CPU time on the host.
- **journalctl -k** or kernel logs can show the **OOM killer** terminating a process. **/proc/PID/oom_score** exposes its selection score. Inspect memory growth, limits and workload demand; more swap can absorb pressure but cannot guarantee process survival.

Troubleshooting

CompTIA Linux+

KNOWLEDGE CHEATSHEET

DIAGNOSE WITH EVIDENCE

Scope Symptoms → **Observe** Logs & metrics → **Test** Hypothesis → **Verify** Lasting fix

- **mkswap** initializes designated swap storage; **swapon** activates it and **swapon --show** lists active swap. Protect a swap file with restrictive permissions and use filesystem-supported allocation. Persistent activation requires an appropriate fstab entry; **swapon -a** activates configured swap.
- **sysctl -w KEY=value** changes a supported runtime parameter; **sysctl -p** loads the default sysctl configuration, while **--system** loads standard configuration directories. **vm.swappiness**, **vm.nr_hugepages** and overcommit settings tune different memory behavior; none creates physical RAM. **tuned/tuned-adm** apply and select tuning profiles.

Capacity, I/O and filesystem failures

- **df -h** measures filesystem space; **df -i** measures inode use; **du -ah /path** locates allocated usage. Inodes track metadata and data-block references, while directory entries hold names. Many small files can exhaust inodes despite free data blocks.
- If **df** reports space that **du** cannot find, inspect deleted but open files with **lsof +L1**, hidden contents beneath mountpoints and filesystem reservations. Remove only understood data; release an open file through the owning application when possible.
- **iostat -xz 1** helps distinguish IOPS, throughput, latency and queue pressure. **iowait** signals CPU time idle with pending I/O, rather than CPU computation. Check device errors, workload patterns and storage limits before adding CPU resources.
- **Read-only filesystem** errors can follow a protective remount after corruption or I/O faults. Check **findmnt**, kernel logs, device health and mount options. Stop writers and unmount before offline **fsck.ext4** or **xfs_repair**; root filesystems may require rescue media.
- Repair with the tool matching the actual filesystem. **mkfs** creates a filesystem and is not a repair. Investigate failed disks and missing LVM PVs before rebuilding; mirrored volumes require replacement and restoration of redundancy rather than recreating the data volume.

Network and name-resolution failures

- Work through **link** → **address/prefix** → **neighbor** → **route** → **DNS** → **listener** → **firewall/application**. **ip neigh** inspects local neighbor resolution; **ip route get DESTINATION** checks route choice. A responding ping proves neither TCP availability nor application health.
- **ss -lntp** lists TCP listeners with process details where permitted; **ss -lnup** checks UDP. **127.0.0.1** accepts local connections only; **0.0.0.0** listens on IPv4 local addresses. **nc -vz HOST PORT** tests a TCP connection; **curl** exercises HTTP(S).
- **NXDOMAIN** means the queried DNS name does not exist in that response; a timeout means no timely reply. Compare **dig @SERVER NAME**, **getent hosts NAME**, hosts-file entries and NSS order. Persist resolver changes through their owning network manager.

Troubleshooting

CompTIA Linux+

KNOWLEDGE CHEATSHEET

DIAGNOSE WITH EVIDENCE

Scope Symptoms → **Observe** Logs & metrics → **Test** Hypothesis → **Verify** Lasting fix

- **tcpdump -ni eth0 host 192.0.2.10** captures matching traffic. **mtr -T -P 443 HOST** traces with TCP probes; apparent loss at one intermediate hop can reflect probe rate limiting when later hops respond normally. **nmap -Pn** skips host discovery and tests ports despite blocked discovery probes.
- Inspect interface errors/drops, MTU, negotiated speed and duplex, routing metrics and firewall counters. A refused connection suggests a rejection or absent listener; a timeout may indicate filtering or loss. **No route to host** can also be produced by an ICMP rejection, so inspect evidence before changing routes.

Boot, service and access failures

- For emergency mode after a mount failure, inspect **journalctl -xb**, **/etc/fstab**, UUIDs, device availability and filesystem health. Correct the failing requirement; defer a genuinely optional mount deliberately. Validate mounts before attempting normal boot again.
- For a failed service, inspect **systemctl status**, **journalctl -u UNIT**, unit overrides, startup commands, service identity, dependencies, timeouts and occupied ports. **Enabled** does not mean **running**; **active** does not prove the application is ready.
- For **permission denied**, check path spelling, ownership, directory traversal, mode bits, ACL mask, attributes, mount flags and SELinux denials. **noexec** can block direct execution even when the script has **x** permission; an immutable attribute can block metadata changes despite elevated privilege.
- **Too many open files** points toward descriptor limits or a leak; inspect **ulimit -n**, **/proc/PID/limits** and **lsof**. PAM **limits.conf/limits.d** and systemd **LimitNOFILE=** affect different execution contexts. Changing a shell limit does not automatically update an existing daemon.
- For SSH failures, distinguish network refusal/timeout from authentication failure. Inspect client **ssh -v**, server logs, account state, key location/ownership, allowed algorithms and SELinux labels. Validate the fix with a fresh connection and confirm it survives the intended reboot or reload.