

MOVING AROUND

Command	What it does
<code>pwd</code>	Show the folder you are in
<code>ls -la</code>	List all files, including hidden ones, with details
<code>cd /var/log</code>	Go to a folder
<code>cd ..</code>	Go up one folder
<code>cd ~</code>	Go to your home folder
<code>cd -</code>	Go back to the previous folder

FILES AND FOLDERS

Command	What it does
<code>mkdir -p a/b/c</code>	Make folders, including any missing parents
<code>touch notes.txt</code>	Create an empty file (or update its timestamp)
<code>cp -r src/ dst/</code>	Copy a folder and everything in it
<code>mv old.txt new.txt</code>	Move or rename
<code>rm file.txt</code>	Delete a file
<code>rm -r folder/</code>	Delete a folder and everything in it
<code>ln -s /path/target link</code>	Make a symbolic link
<code>find / -name "*.conf"</code>	Find files by name
<code>find . -size +100M</code>	Find files bigger than 100 MB

READING FILES

Command	What it does
<code>cat file</code>	Print a whole file
<code>less file</code>	Scroll through a file (q to quit)
<code>head -n 20 file</code>	First 20 lines
<code>tail -n 20 file</code>	Last 20 lines
<code>tail -f /var/log/syslog</code>	Follow a log as it grows
<code>wc -l file</code>	Count lines
<code>diff a b</code>	Show the differences between two files

SEARCHING AND FILTERING TEXT

Command	What it does
<code>grep "error" file</code>	Lines that contain a word
<code>grep -ri "error" /etc</code>	Search a folder, ignoring case
<code>grep -v "#" file</code>	Lines that do NOT match
<code>sort file uniq -c</code>	Count repeated lines
<code>cut -d: -f1 /etc/passwd</code>	Take field 1, split on colons
<code>awk '{print \$1}' file</code>	Print the first column
<code>sed 's/old/new/g' file</code>	Replace text

PERMISSIONS

Command	Meaning
<code>755</code>	<code>rwxr-xr-x</code>
<code>644</code>	<code>rw-r--r--</code>
<code>700</code>	<code>rwX-----</code>
<code>600</code>	<code>rw-----</code>
<code>chmod u+x f</code>	Add execute for owner
<code>chown u:g f</code>	Change owner and group
<code>SUID/SGID/sticky</code>	4 / 2 / 1 in front: <code>chmod 4755</code>

USERS AND GROUPS

Command	What it does
<code>whoami</code>	Show your user name
<code>id</code>	Show your user ID and groups
<code>sudo command</code>	Run one command as root
<code>su - alice</code>	Switch to another user with their login environment
<code>useradd -m alice</code>	Create a user with a home folder
<code>passwd alice</code>	Set a password
<code>usermod -aG sudo alice</code>	Add a user to a group (keep the -a)
<code>/etc/passwd</code>	User accounts
<code>/etc/shadow</code>	Password hashes (root only)

PROCESSES

Command	What it does
<code>ps aux</code>	Every running process
<code>top</code>	Live view of CPU and memory
<code>kill 1234</code>	Ask process 1234 to stop (SIGTERM, 15)
<code>kill -9 1234</code>	Force it to stop (SIGKILL, 9)
<code>pkill nginx</code>	Stop processes by name
<code>command &</code>	Run in the background
<code>jobs</code>	List background jobs
<code>fg %1</code>	Bring job 1 back to the front
<code>nohup command &</code>	Keep running after you log out

SERVICES AND LOGS (SYSTEMD)

Command	What it does
<code>systemctl status sshd</code>	Is the service running?
<code>systemctl start sshd</code>	Start it now
<code>systemctl restart sshd</code>	Restart it
<code>systemctl enable sshd</code>	Start it at every boot
<code>journalctl -u sshd</code>	Logs for one service
<code>journalctl -f</code>	Follow the system log live

DISKS AND SYSTEM

Command	What it does
<code>df -h</code>	Free space on each disk
<code>du -sh folder/</code>	Size of a folder
<code>free -h</code>	Memory use
<code>lsblk</code>	List disks and partitions
<code>mount /dev/sdb1 /mnt</code>	Mount a partition
<code>uname -a</code>	Kernel version
<code>uptime</code>	How long the system has been up, and load

NETWORKING

Command	What it does
<code>ip a</code>	IP addresses (replaces ifconfig)
<code>ip r</code>	Routing table
<code>ss -tulpn</code>	Open ports and the program on each (replaces netstat)
<code>ping -c 4 8.8.8.8</code>	Test reachability with 4 pings
<code>traceroute host</code>	Show each hop to a host
<code>dig example.com</code>	DNS lookup
<code>curl -I https://example.com</code>	Fetch only the HTTP headers
<code>ssh user@host</code>	Log in to another machine
<code>scp file user@host:/tmp/</code>	Copy a file over SSH

PACKAGES

Debian/Ubuntu	Red Hat/Fedora
<code>apt update</code>	<code>dnf check-update</code>
<code>apt install nginx</code>	<code>dnf install nginx</code>
<code>apt remove nginx</code>	<code>dnf remove nginx</code>
<code>dpkg -l</code>	<code>rpm -qa</code>

ARCHIVES

Command	What it does
<code>tar -czvf site.tar.gz site/</code>	Create a gzip archive
<code>tar -xzvf site.tar.gz</code>	Extract it
<code>tar -tzvf site.tar.gz</code>	List what is inside
<code>zip -r site.zip site/</code>	Create a zip
<code>unzip site.zip</code>	Extract a zip

REDIRECTION AND PIPES

Symbol	What it does
<code>cmd > file</code>	Write output to a file (overwrites)
<code>cmd >> file</code>	Add output to the end of a file
<code>cmd 2> err.log</code>	Send errors to a file
<code>cmd > all.log 2>&1</code>	Send output and errors to one file
<code>cmd1 cmd2</code>	Pass output to the next command
<code>cmd tee file</code>	Show output and save it
<code>cmd1 && cmd2</code>	Run cmd2 only if cmd1 worked
<code>cmd1 cmd2</code>	Run cmd2 only if cmd1 failed

FOLDERS WORTH KNOWING

Path	What lives there
<code>/etc</code>	System configuration files
<code>/var/log</code>	Logs
<code>/home</code>	Users' home folders
<code>/root</code>	The root user's home folder
<code>/tmp</code>	Temporary files
<code>/usr/bin</code>	Most programs
<code>/proc</code>	Live kernel and process information
<code>/dev</code>	Devices, such as disks
<code>/etc/fstab</code>	Disks to mount at boot
<code>/etc/hosts</code>	Local name to IP lookups

EXAM TRAPS

- `>` overwrites a file; `>>` adds to it.
- Plain `kill` sends SIGTERM (15), not SIGKILL (9).
- `usermod -G` without `-a` replaces the user's groups.
- Password hashes live in `/etc/shadow`, not `/etc/passwd`.
- `su -` loads the target user's environment; plain `su` keeps yours.
- `systemctl enable` does not start a service now. It starts it at the next boot.