Red Hat

Administrator Commands

---

1. System Monitoring and Troubleshooting

Command	Description
`top`, `htop`	Real-time process monitoring
`ps -aux --forest`	View processes in a tree-like format
`systemctl status xx.service`	Check status of a service
`journalctl -xe`	View detailed logs for troubleshooting
`journalctl –since "2 hours ago"`	Filter logs from last 2 hours
`sar -u 1 5`	CPU usage every 5 seconds
`dstat -cdngy`	Live stats for CPU, disk, network, and memory
`iotop`	Monitor I/O usage by process
`strace -p <pid>`	Debug issues by tracing system calls for a process
`vmstat 1`	Real-time system performance metrics

---

2. Package Management and Repositories

Command	Description
`dnf update`	Update all packages
`dnf install <package>`	Install a package
`dnf list installed`	List all installed packages
`dnf check-update`	Check for available updates
`dnf remove <package>`	Remove a package
`rpm -qa`	Search for installed packages

`rpm -V <package>`	Verify package integrity
`rpm -U <package>`	Upgrade a package
`rpm -e <package>`	Remove a package
`yum repolist --disabled`	List disabled repositories
`yum-config-manager --enable <repo>`	Enable a specific repository
`yum repolist all`	List all repositories

---

3. User and Permissions

Command	Description
`id <user>`	Display user ID and group ID
`usermod -aG <group> <user>`	Add a user to a group
`chage -l <user>`	List account expiration details
`getfacl <file>`	Get file ACL permissions
`setfacl -m u:<username>:rw <file>`	Modify file ACL permissions
`useradd -m -d /home/<user> -s /bin/bash <user>`	Create a new user
`groupadd <group>`	Create a new group
`usermod -s /sbin/nologin <user>`	Change a user's shell to `nologin`
`passwd -e <user>`	Expire a user's password
`chage -E YYYY-MM-DD <user>`	Set expiration date for user
`chage -m 7 -M 90 -W 14 <user>`	Minimum 7 days, max 90 days, 14-day warning
`gpasswd -a <user> <group>`	Add user to group
`gpasswd -d <user> <group>`	Remove user from group
`groups <user>`	Show groups for a user

`chown <user>:<group> <file>`	Change file ownership
`visudo`	Edit the sudoers file

---

4. SUID vs SGID vs STICKY BIT

Command	Description
`chmod 4xxx <file>`	Set SUID on a file
`chmod 2xxx <file>`	Set SGID on a file
`chmod 1xxx <file>`	Set Sticky Bit on a file

---

5. Networking commands

Command	Description
`nmcli connection show`	Show network connections
`nmcli connection modify "xxxx" ipv4.addresses "192.168.1.100/24" ipv4.gateway "192.168.1.1" ipv4.dns "8.8.8.8 8.8.4.4" ipv4.method manual`	Configure a static IP
`nmcli connection up`	Bring up a network connection
`nmtui`	Text-based network manager
`ip link`	Show network interfaces
`ping -c 5 <host>`	Send 5 ICMP packets to a host
`traceroute <host>`	Trace the route to a host
`firewall-cmd --list-all`	Show all firewall rules

`firewall-cmd --add-port=<port>/tcp --permanent`	Add a port to the firewall
`firewall-cmd reload`	Reload firewall rules
`lsof -i -P`	List open network connections
`ss -ntulp`	Show listening ports and connections
`tcpdump -i eth0`	Capture packets on `eth0`
`nc -zv 192.168.1.10 443`	Test if a port is open
`wget <url>`	Download a file using HTTP/HTTPS
`curl <url>`	Fetch content from a URL

---

6. SELinux and Security

Command	Description
`sestatus`	Display SELinux status
`getenforce`	Get SELinux enforcement mode
`setenforce 1`	Enable SELinux
`setenforce 0`	Disable SELinux
`chcon -t <type> <file>`	Change SELinux context of a file
`restorecon -Rv <dir>`	Restore SELinux context recursively
`chcon --reference=/path/ref /path/to/dest`	Copy SELinux context from one file to another
`openssl req -new -x509 -keyout server.key -out server.crt -days 365`	Generate self-signed SSL certificate
`ssh-keygen -t ed25519`	Generate SSH key with ed25519
`ssh-copy-id user@host`	Copy SSH key to a remote host
`openssl passwd -6`	Generate hashed password

7. Important log files

File	Description
`/var/log/messages`	General system messages
`/var/log/secure`	Authentication and authorization logs
`/var/log/cron`	Logs for scheduled tasks (cron jobs)
`/var/log/dmesg`	Kernel ring buffer messages
`/var/log/boot.log`	Boot process logs
`/var/log/httpd`	Apache logs
`/var/log/yum.log`	Yum/DNF logs
`/var/log/journal/`	Persistent systemd journal logs
`/var/log/kern.log`	Kernel-specific logs

---

8. Useful commands

Command	Description
`rsync -avz /source /destination`	Synchronize files between source and destination
`scp file user@host:/destination`	Securely copy a file to a remote host
`tar -czvf backup.tar.gz /dir`	Compress a directory to a tarball

---

9. Subscription-manager

Command	Description
`subscription-manager register --org <org_id> --activationkey <ak>`	Register the system with Red Hat

`subscription-manager release --show`	Show current release version
`subscription-manager release --set=8.10`	Set the release version to 8.10
`subscription-manager unregister`	Unregister the system
`subscription-manager repos --list`	List available repositories
`subscription-manager repos --enable=<repo>`	Enable a specific repository

---

10. Storage Management

Command	Description
`lsblk`	List block devices
`mkfs.ext4 /dev/sdx1`	Format partition as ext4
`mount /dev/sdx1 /mnt`	Mount partition
`mount -a`	Mount all filesystems from `/etc/fstab`
`umount /mnt`	Unmount a partition
`pvcreate /dev/sdx1`	Create a physical volume for LVM
`vgcreate my_vg /dev/sdx1`	Create a volume group
`lvcreate -L 10G -n my_lv my_vg`	Create a logical volume of 10GB
`vgextend my_vg /dev/sdx1`	Add a physical volume to a volume group
`lvextend -r -L +2G /dev/mapper/my_vg-my_lv`	Extend a logical volume by 2GB
`lvextend -r -l +100%FREE /dev/my_vg-my_lv`	Extend a logical volume to use all free space

---

11. Scheduled Tasks

Command	Description

`crontab -e`	Edit crontab for the current user
`crontab -l`	List the current user's crontab
`crontab -l -u <user>`	List crontab for a specific user
`cat /etc/crontab`	Check syntax

---

12. Ansible Commands

Command	Description
`ansible-playbook -i inventory_file playbook_file --limit='host1,host2,!host3'`	Run ansible playbook to inventory hosts, limit execution by using `--limit.`
`!host`	Exclude host from playbook
`--check`	Run ansible playbook in check mode, no changes will be performed on the hosts
`ansible-playbook -i 'host1,host2,' playbook_file`	Run ansible playbook to specified hosts, no need of inventory file
`ansible-vault create file.yml`	Create a new file encrypted with Ansible Vault.
`ansible-vault encrypt file.yml`	Encrypt an existing file with Ansible Vault.
`ansible-vault decrypt file.yml`	Decrypt a file encrypted with Ansible Vau
`ansible all -v -i inventory -b -m shell -a 'reboot' --limit='server1'`	Execute command from shell with root permissions