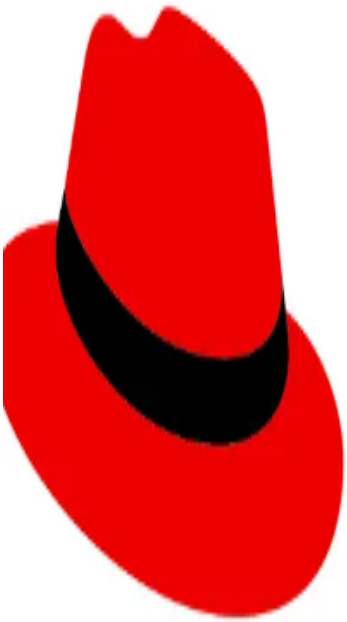
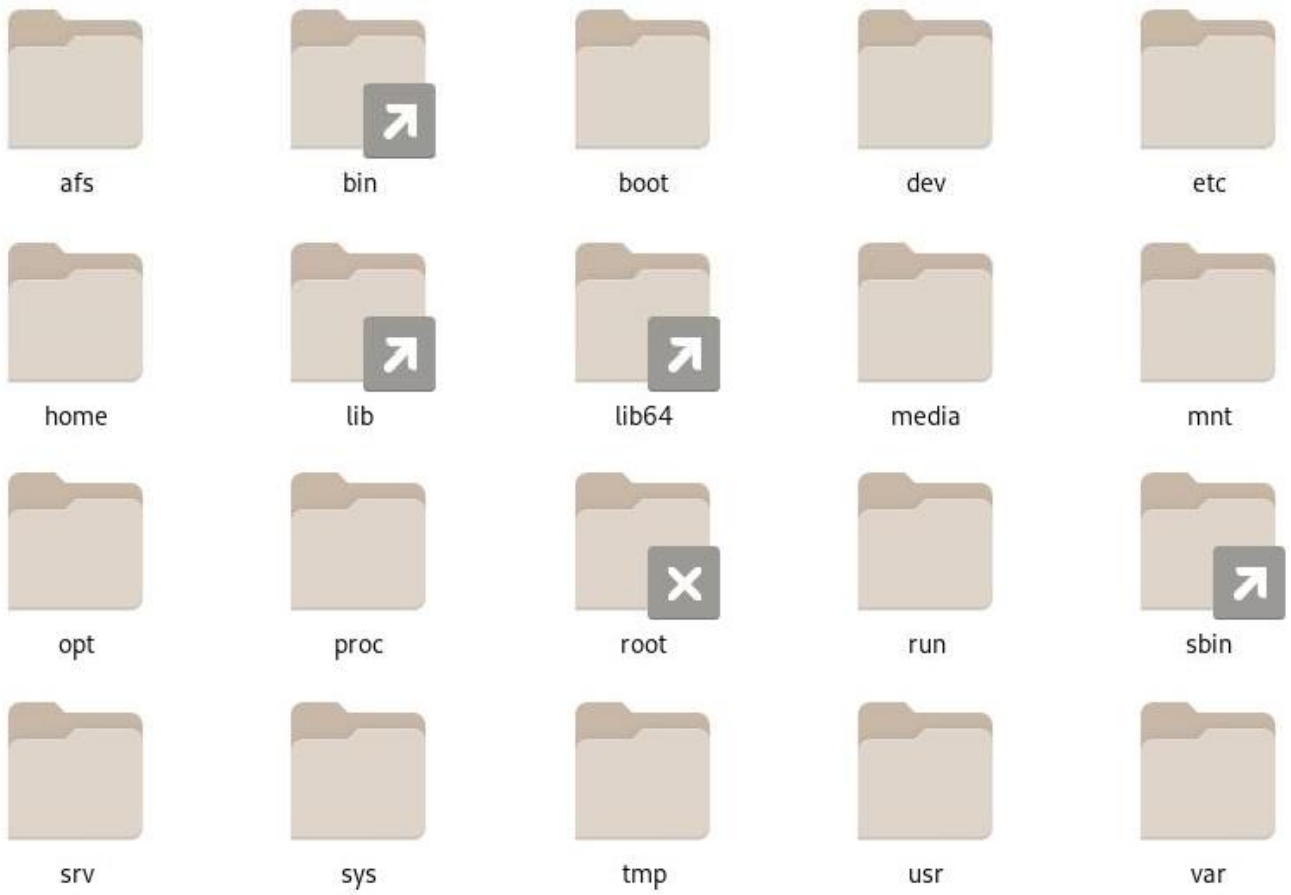


Red Hat Enterprise Linux 9

RHCSA – SA 1 LAB BOOK

Chapter 3 - Managing Files from the Command Line Specifying Files by Name Matching File Names with Shell Expansions

The File-System Hierarchy



/: Root of the file system hierarchy, all further directories are subdirectories of /.

/afs: (Andrew file system) Sharing file & directory across the network.

/bin: binary executable files of commands.

/boot: Files required to boot the system (boot loader, kernel Image, initramfs).

/dev: Files related to devices (Hard Disk, Terminal file).

/etc: All configuration files.

/home: Home directories of all Normal users.

/lib and /lib64: All library files 32 bit libraries are stored in /lib and 64 bit

libraries in /lib64.

/media: mount point for removable devices up to RHEL6. Not used in RHEl9

/mnt: mount point for shared drives and partitions.

/opt: all 3rd party application data.

/proc: all processes data, cpuinfo, meminfo, partition table and etc.

/root: home directory of root user

/run: all running services data

/sbin: all binary executable by root user

/srv: to store any particular data

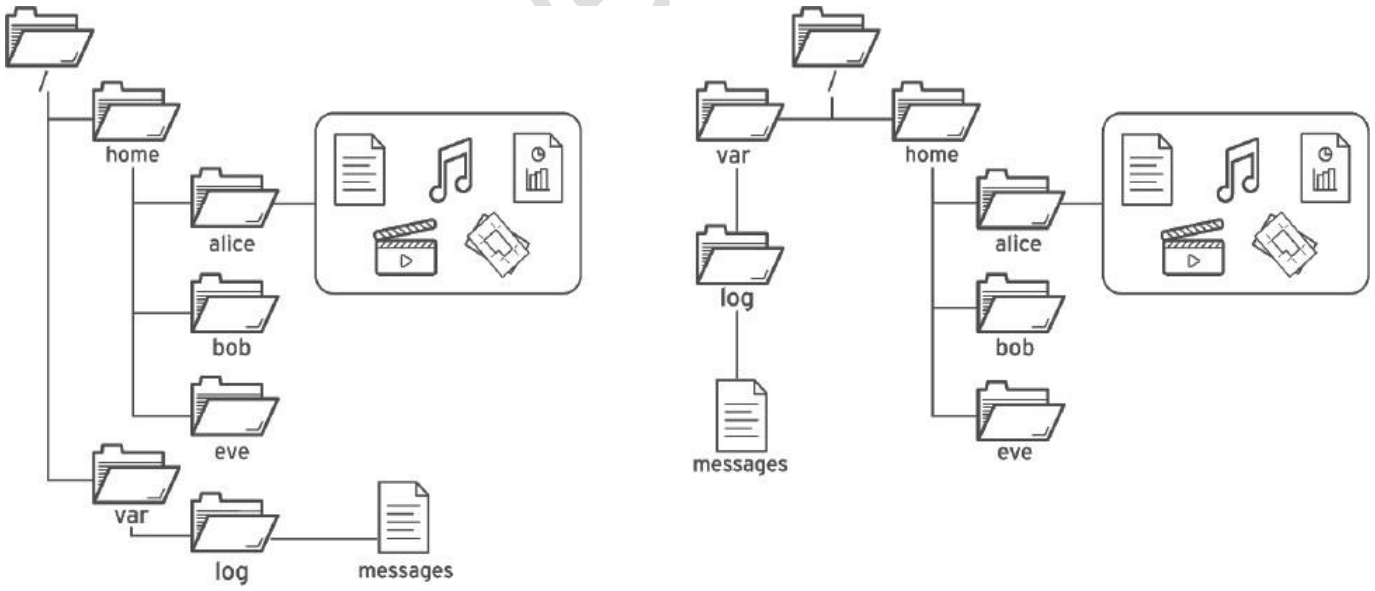
/sys: system data

/usr: It is 2nd level file system hierarchy all documentation, configuration, library and binary files all sharable data.

/tmp: temporary files.

/var: variable data.

Absolute paths and relative paths



Types of Path used to create Directory

Absolute path: Path which starts with /. It will traverse each and every directory in order to reach destination.

OR

The full path to a file or directory.

[root@serverX~] # mkdir /tmp/test

Relative path: Path related to current working directory

OR

Start from current working directory.

[root@serverX~] # cd /tmp

[root@serverX temp] # mkdir test

Managing Links between Files

It is possible to create multiple names that point to the same file.

There are two ways:

1) **Hard Link**:
   - Every file starts with a single hard link, from its initial name to the data on the file system.
     [user@host ~]$ ls -l newfile.txt
     -rw-r--r--. 1 user user 0 Mar 11 19:19 newfile.txt
   - Hard Link is a new name of file pointing to same data.
   - To check number of hard links available to a file use "ls -l" command
   - Use the ln command to create a new hard link (another name) that points to an existing file.
   - **Command: [user@host ~]$** ln newfile.txt /tmp/newfile-hlink2.txt
   - All hard links that reference the same file will have the same link count, access permissions, user and group ownerships, time stamps, and file content.
   - To find out whether two files are hard links of each other, one way is to use the - i option with the ls command to list the files inode number

- Hard link files have same inode numbers
  Command: [user@host ~]$ ls -il newfile.txt /tmp/newfile-hlink2.txt
  8924107 -rw-rw-r--. 2 user user 12 Mar 11 19:19 newfile.txt
  8924107 -rw-rw-r--. 2 user user 12 Mar 11 19:19 /tmp/newfile-hlink2.txt
- Even if the original file gets deleted, the contents of the file are still available as long as at least one hard link exists.

Limitations:

- Hard links can only be used with regular files.
- In command cannot be used to create a hard link to a directory or special file.
- Hard links can only be used if both files are on the same file system.

2) Soft Link:

- The ln -s command creates a soft link, which is also called a "symbolic link."
- A soft link is not a regular file, but a special type of file that points to an existing file or directory.

Advantages:

- They can link two files on different file systems.
- They can point to a directory or special file, not just a regular file.
- ln - s -: command is used to create a new soft link for the existing file.
- **Command:**
  [user@host ~]$ ln -s /home/user/newfile-link2.txt /tmp/newfile-symlink.txt
  [user@host ~]$ ls -l newfile-link2.txt /tmp/newfile-symlink.txt
- When the original regular file gets deleted, the soft link will still point to the file but the target is gone.
- A soft link pointing to a missing file is called a "dangling soft link."

END