

1.Distribution

LINUX: Redhat Centos Fedora Suse Opensuse Debian Ubuntu Gentoo

Archlinux slackware

UNIX: FreeBSD OpenBSD Solaris

2.device file

terminal: /dev/tty /dev/vt/n (:m) /dev/pts/n (:m)

disk/partition: /dev/hdxy /dev/sdxy /dev/srx /dev/mdxpy

/dev/vg_name/lv_namepx /dev/dsk/c#t#d#p#(controller target disk partition)

rpool/path (ZFS)

3.access system

① identity/permission

owner group others - r4 w2 x1

regular - ① read-only: r ② read and write: rw | execute: x

directory - ① list files only: r ② access files: rx ③ create and delete files: rwx

② SELinux

domain type

③ architecture

system process: identity/domain

file system: permission/type

4.process tree

init(root) -> daemon (stand_alone, xinetd_based) -> dependency

daemon type: listen by self or by xinetd, listen function configured in conf file

init adopt orphan process automatically

init reap zombie(address space release, process table left) child process automatically

process system call:

fork()&exec() - fork child process

exit()&SIGCLD - release address space, send signal

wait() - destroy zombie data structure

shell – fork(),exec(),wait() shell & - fork(),exec() system() – fork(),exec(),wait()

5.X window system

X client – xdm(gdm/kdm) -> xwm(gnome/kde) -> app(vmware)

X server – X / xming

Soft: xorg-x11, xorg-x11-server, gnome

6. inode/block

Inode table - inode num, access permission, uid, gid, access time, modify time, change time, size, link num, handler

Block - regular file: binary directory: list(name <-> inode) symbolic link: path

Hard link - different name -> one inode

Soft link - different name -> different inode, symbolic link record path

Mount/nfs - name -> another inode

7. process & thread

Thread creating: PID/TID=X, TGID=X --> PID/TID=Y, TGID=X

Process creating: PID/TID=X, TGID=X --> PID/TID=Z, TGID=Z

8. UNIX-like system

Application <- System lib <- System call | Kernel

System call trigger soft interruption to send parameters to kernel by registers, then kernel start corresponding threads service.