
SO2 Session 25 - Lab 03 (Special files and intro to Character device drivers)
shell : ls /dev - l
=> crw-rw---- (c : character, permission)
=> drw-rw---- (d : directory, some)
brw-rw---- (b : block)
role of Linux os : redirection (from the user space to the kernel space) of the system call (open) to the specific driver
we have two options of device files :
1) character
2) block
In transfer process we use character or block by size of data
With Linux : for any type of block device always this application can work with (read, write, ...)
and for any type of system file always this application work with the corresponding system calls
/proc all in the ram => all the file of system that be in the ram : pseudo-filesystem
FS : file system :
RAM : /proc, /tmp, /sys
Block : rootfs
Net : netfs
Majors and Minors
example : HDD has 4 partition
1) it must have one identifier
2) Driver must create 4 devices (1 parson = one device)
3) all the parson has the some major (owns Driver)
4) different minor
Choosing identifier for new device :
1) Static but it must be not seem to be used
2) Dynamic
Create a device type file
sudo mknod /dev/teechleef c 42 0
=> ls -l /dev/teechleef
crw-r--r-- 1 root root 42, 0 May 17 16:58 /dev/teechleef
Data structure for a character device
Struct cdev
file_operations : pointer to function
Example : read value Temperature
/dev/x → Driver → Temperate
1) Driver : function read ( )
call in file_operation
2) to connect the buffer to the user space
we use copy_to_user( )
3 important Structure to any file of file in Linux :
struct file_operation (to manipulation files)
struct file (definition of files permission)
struct inode (Hardware low level)