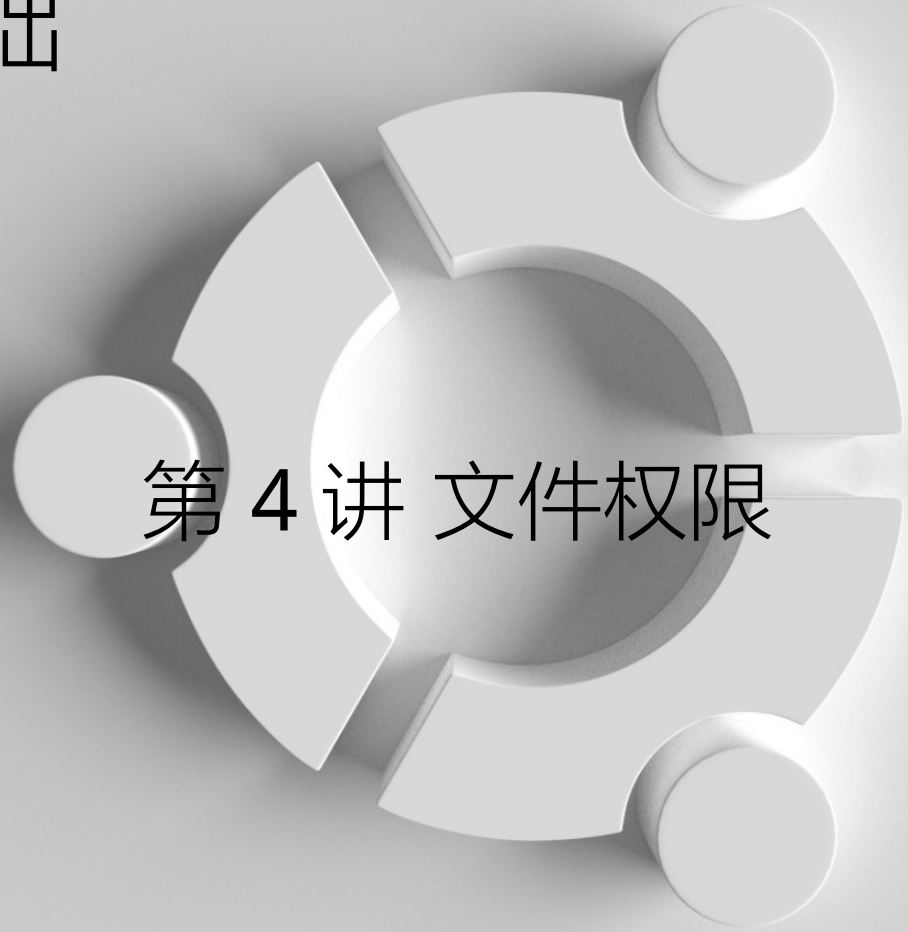
Linux 基础



第 4 讲 文件权限

Linux 继承 Unix 的设计思想：一切皆文件。

Linux 通过文件的形式对接驱动、设备、内存数据等。

一切皆文件

- 目录是一种比较特殊的文件，其内容是记录其他文件、目录的名称等信息。

- 和普通文件不同的是，目录内容永远不会真正为空，至少包括 . 和 .. 两个目录。

从 ls 开始

- ls 可以显示文件的详细信息，包括权限、所有者、创建时间、大小、文件名、文件类型等。
- 可以通过 ls -l 查看当前目录下的文件信息：

```
drwxr-xr-x  2 wy wy 4096 2月   15 19:54 gowork
-rw-rw-r--  1 wy wy    0 2月   21 10:00 http.go
drwxr-xr-x  5 wy wy 4096 2月   15 19:13 linuxinit
lrwxrwxrwx  1 wy wy    6 2月   18 18:01 music -> 音乐
```

类型和权限　硬链接数　用户 用户组　文件大小　修改时间　文件名

文件类型

- ls 使用 10 个字符表示权限和类型，第一个字符表示类型。
  - - 表示普通文件
  - d 表示目录
  - c 表示字符设备
  - b 表示块设备
  - p 表示管道
  - s 表示套接字文件
  - l 表示符号链接（软链接）

`ls -l` 显示的文件权限

- 文件类型后 9 个字符表示权限， 3 个一组，分别表示文件所属用户、用户组、其他用户对应的权限。
- 按照顺序， `r` 表示可读， `w` 表示可写， `x` 表示可执行， `-` 表示没有此权限。
-
- 示例： `rwxr-xr--` 表示用户具备**可读可写可执行**的权限，而用户组具备**可读可执行**的权限，其他用户仅可读。

文件权限与标志位

- r：可读； w：可写； x：可执行。
- 八进制表示： r:100; w:010; x:001

用户	用户组	其他用户
rwx	r-x	r-x
111	101	101
八进制表示： 0755		

文件的默认权限

- 创建文件时，不指定权限，会自动分配一个默认权限。
- 默认权限是通过权限掩码生成的。
- 运行 `umask` 查看当前的权限掩码。
- 运行 `umask 022` 设置权限掩码。

权限掩码

- 使用 touch 可以创建空文件：

  touch [文件名]

- 如果权限掩码为 0022，则文件的权限为： 0644

- 计算方式： 0777 按位减去掩码对应的位，如果是文件再去掉可执行权限。

更改文件权限

- 命令 chmod 用于设置文件权限，示例：

chmod 755 bin/pse	rwxr-xr-x
chmod +x bin/pse	添加可执行权限，所属用户与用户组具备可执行权限
chmod -w bin/pse	去掉写权限，用户，用户组，其他用户都会去掉写权限
chmod u=rwx,g=rx,o=r bin/pse	相当于chmod 754 bin/pse

更改文件所属用户和用户组

- 命令 chown 用于修改文件所属用户和组：

命令	说明
`chown oklinux:oklinux hd1`	更改文件 hd1 所属用户和组为 oklinux
`chown :helo hd1`	更改文件 hd1 所属用户组为 helo
`chown oklinux:  hd1`	更改文件 hd1 所属用户为 oklinux
`chown oklinux:oklinux tmp/ -R`	目录 tmp 中所有文件所属用户和组改为 oklinux

目录的可执行权限

- 目录如果没有可执行权限则无法访问。

- 如果某一目录只允许用户自己访问，则可以去掉所属组和其他用户的可执行权限。

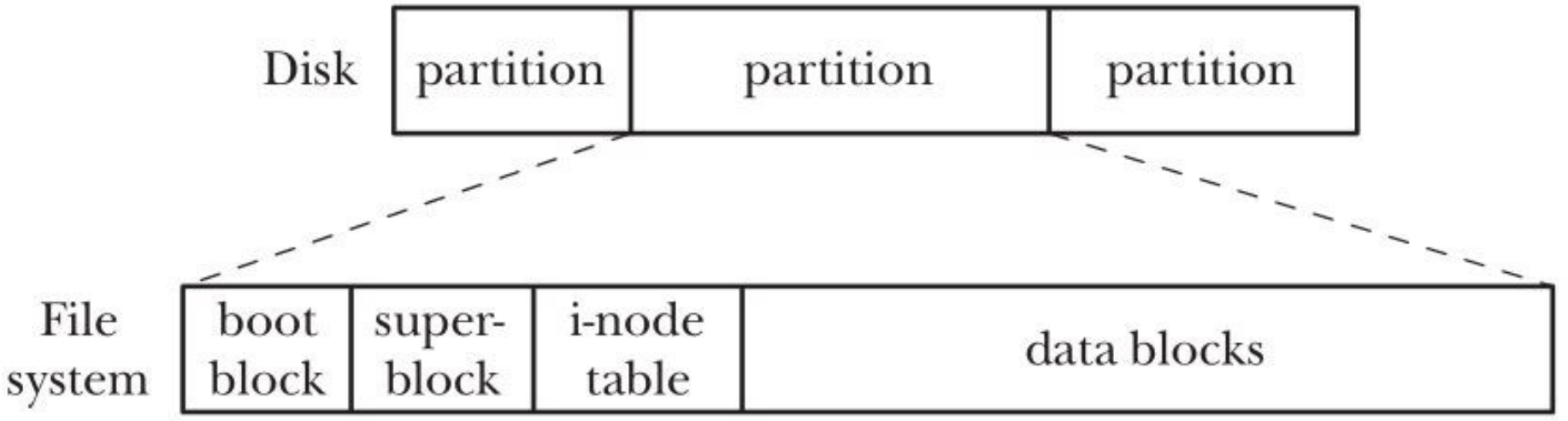
硬盘分区、 i-node 、软链接和硬链接

硬盘分区概要

- 存储设备往往被分割成块，称为扇区。一个扇区大小 512 字节。
- 每一个分区有包括以下几个区域：

  `Boot, Super, i-node table, Data`

硬盘分区概要


Disk
partition
partition
partition
File system
boot block
super-block
i-node table
data blocks

硬盘分区概要

- Boot block：如果系统安装在此分区，则此区域存储启动相关的程序等信息。

- Super block：存储文件系统的类型，i-node table 大小等信息。

硬盘分区概要

- I-node table：文件的入口，每个文件都会有一个 I 节点（i-node），记录了文件的存储扇区、权限、所有者等信息。后面讲到权限操作等命令其实都在改变 I 节点的数据。
- Data blocks：实际存储数据的区域。

查看文件 i-node 号和权限

- 运行 `ls -li` 可以查看文件的 i-node 编号以及权限。

- 运行 `ls -i` 可以查看文件的 i-node 号。

目录记录的信息

- 当在目录中创建文件时，实际会创建一个 i-node ，并在目录中添加一条文件名映射到 i-node 编号的记录。

- 用户要根据文件名找到文件，而系统则根据 i-node 编号。

硬链接

- 硬链接没有创建新文件，只是多了一个别名。指向同一个 i-node 号。

- 创建硬链接： ln [目标文件] [硬链接名]
- i-node 在一个分区下是唯一的，不同分区会出现重复，所以硬链接不能跨分区。

硬链接

- 硬链接本质上是增加了 i-node 引用计数。

- 使用 rm 删除文件，实际上删除的是硬链接计数，如果计数为 0，则 i-node 标记的存储区域被释放掉，才是真正的删除文件。

符号链接

- 也称为软链接，类似于 Windows 下的快捷方式。
- 创建软链接： `ln -s [ 目标文件 ] [ 链接名 ]`
- 创建符号链接实际上是创建了新的文件，文件类型标记为符号链接，存储内容是目标文件的路径。

符号链接

- 创建符号链接可以跨分区，可以跨文件系统。

- 符号链接比较灵活，实际使用非常广泛。

符号链接

- 删除目标文件，不会删除符号链接，如果再以相同的名称创建新的文件，则符号链接继续指向该文件。
- 删除符号链接不会删除目标文件。
- 符号链接的权限是目标文件的权限。