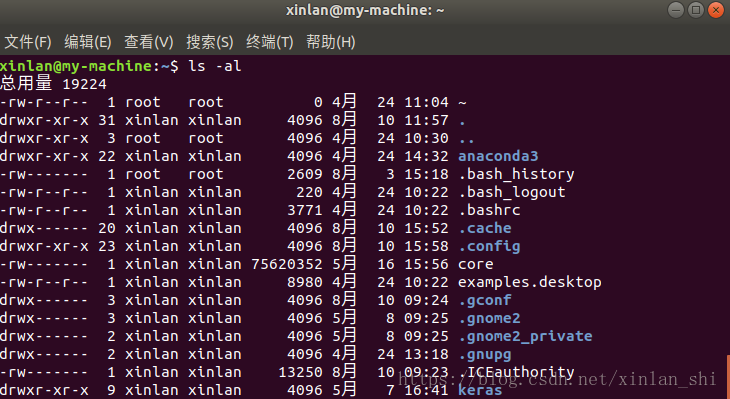
linux 文件的属性

在终端terminal输入“ls -al”命令就可以查看文件/目录的属性了



接下里分别讲述这7列所代表的意思

第一列：文件/目录的类型与权限

图中第一个字符出现了’d”-‘两种，’d’表示是目录，’-‘表示是文件，常用的还有’l’表示链接文件(linkfile)

图中的第二、三、四个字符为一组，第五、六、七为一组，第八、九、十为一组表示文件不同用户对文件的权限。图中出现了’r’(read，可读)，’w’(write，可写)， ‘x’(execute，可执行)，这三组每个位置都是固定‘可读可写可执行(rwx)’，若拥有对应权限则出现对应字符，若不具备对应权限则是出现’-‘(不具备对应位置的所有权)。第一组表示的是文件所有者的权限，第二组是文件所有者所在用户组的权限，第三组是其他人对这个文件所拥有的权限。

第二列：多少文件连接到此节点

也就是说文件下面有多少个文件

第三列：介绍文件的所有者

第四列：介绍文件所有者所在的用户组

第五列：介绍这个文件的大小，单位为：Ｂ

第六列：文件创建或者是最近修改的时间（月日）

第七列：文件的名称

在这里，强调一下文件/目录的可读（‘r’）、可执行（‘x’）权利可以开放给同用户组，其他人，但是可写（‘w’）权利一定要谨慎，保护好自己数据安全。

权限对文件：可读（‘r’）-读取内容，可写（‘w’）-编辑、修改内容，可执行（‘x’）-执行文件。

权限对目录：可读（‘r’）-读取目录下的文件名，可写（‘w’）-更改目录结构列表的权利(也就是新建、删除、重命名文件/目录)，（‘x’）-能否进入目录下(也就是能不能’cd’进去啦)

修改文件/目录的属性与权限

大部分文件创建的时候是使用默认的属性，修改文件/目录的属性与权限通常用到以下命令：‘chgrp(改变的文件的所属用户组)’、’chown(改变文件的所有者)’、’chmod(改变文件的权限)’

chgrp

chgrp [-R] dirname/filename

-R: 递归的修改目标目录下的所有文件与目录

chown

chown [-R] 帐号名称 (:组名) filename/dirname

-R: 递归的修改目标目录下的所有文件与目录

注意修改的账户名称必须是系统已存在的帐号，可以做到一起修改用户组

chmod

有两种修改方法

1)数字修改法

chmod [-R] xyz filename/dirname

-R: 递归的修改目标目录下的所有文件与目录

xyz 权限的数值属性

文章的上面讲到权限被分成了三组，每一组是独立的，’rwx’这三个位置出现了代表才有权限，这很像二进制呢，所以同样的’r’出现就是表示1\*2^2=4，’w’=1\*2^1=2，’e’=1\*2^0=1。如果对应位置是’-‘那就是0\*2^n=0。

因此将每一组分开看的话，我们用穷举方式列出所有的组合：

‘rwx’=7

‘rw-‘=6

‘r-x’=5

‘r–’=4

‘-wx’=3

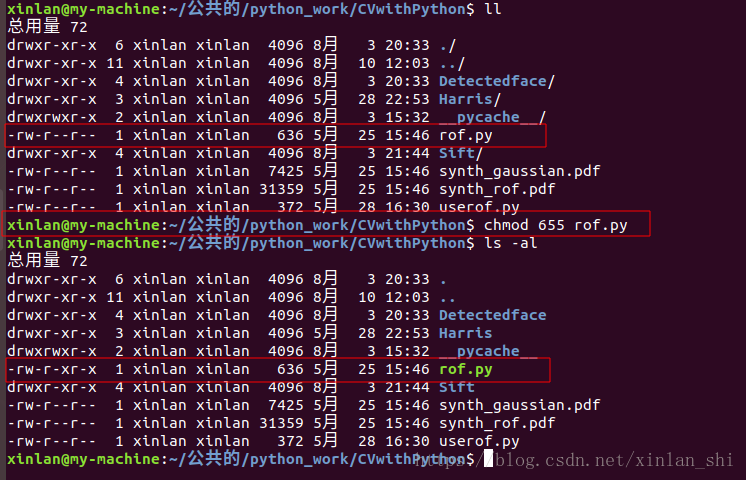
‘-w-‘=2

‘–x’=1

‘—’=0

不过有些是权限的设置不可能出现的了，哈哈，只是为了加强记忆学习。

接下来看一个例子，使用命令行在原来的基础上增加开放目标文件（rof.py文件）用户组、其他人的可执行权限(‘x’)

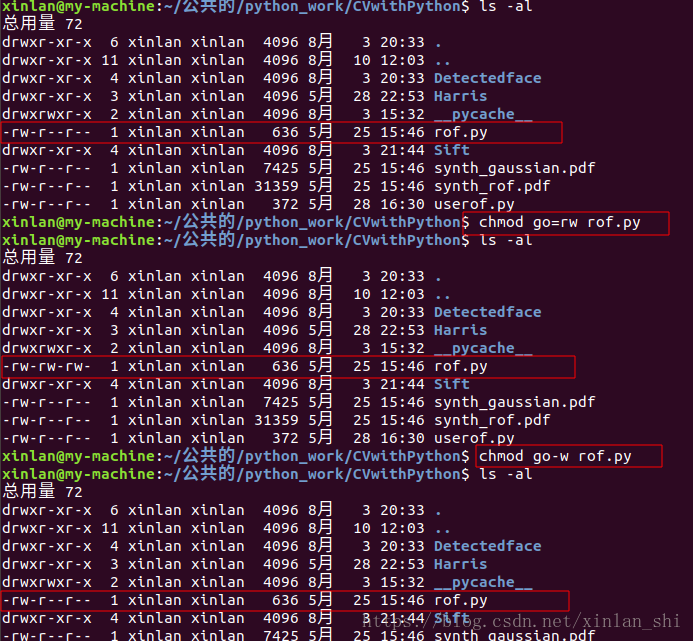


2)符号类型修改法

使用的u,g,o分别代表own,group,others三种身份，使用”=’权限符号’”（直接赋权的方式），”+’权限符号’”(在原来基础上添加指定的权限),”-‘权限符号’”(在原来基础上减去指定的权限)设置不同身份的权限

chmod [ugo][=+-][rwe] filename/dirname

还是使用一个例子更直观的表示一下



可以看到“=”有点类似数字类型改变文件权限的方法，直接开放对应的权限，给对应的用户。