**ACL权限**

**1，ACL权限简介与开启**

ACL权限是为了解决所有者，所属组，其他人三个身份权限不足的问题。    如一个临时用户，试听者。

**查看分区ACL权限是否开启**

dumpe2fs 命令是查询指定分区详细文件系统信息的命令

    选项：-h    仅显示超级块中信息，而不显示磁盘块组的详细信息

 eg:#dumpe2fs -h /dev/sda5     **查看根分区是否支持ACL权限,使用df -h 命令查看到 /分区（根分区）的文件系统是/dev/sda5**

Default mount options: user\_xattr acl        默认支持

**临时开启分区ACL权限**

#mount -o remount,acl /    重新挂载根分区，并挂载加入acl权限

**永久开启分区ACL权限**

#vi /etc/fstab

UUID=8ed6d7bf-062f-4dab-a82f-636f1edda577 / ext4 defaults,acl        1 1         加入acl

#mount -o remount     /

重新挂载文件系统或重启系统，使修改生效

**2，查看与设定ACL权限**

**查看ACl命令**

#getfacle 文件名    查看acl权限

**设定acl权限的命令**

**#setfacl  选项  文件名**

    选项：

    -m    设定acl权限

    -x     删除指定的acl权限

    -b    删除所有的acl权限

    -d    设定默认acl权限

    -k    删除默认acl权限

    -R    递归设定acl权限

**给用户设定acl权限**

**setfacl -m u:st:rx   /project/**给用户st赋予r-x权限，使用“u:用户名:权限" 格式

具体操作：

先创建两个用户，再创建一个用户组，把这两个用户加入到用户组中（gpasswd -a 用户  组），修改文件的所有者，所属组，修改文件权限，

**给用户组设定ACL权限**

**setfacl -m g:tgoup2:rwx   /project/**给用户组tgroup分配acl权限，使用“g:用户组名:权限" 格式

**3，最大有效权限和删除ACR权限**

mask是用来指定最大有效权限的。如果给用户授予了ACL权限，是需要和mask的权限“相与”才能得到用户的真正权限。

修改最大有效权限

#setfacl -m  m:rx  文件名        设定mask权限为r-x，使用“m:权限”格式

删除acl权限

#setfacl -x u:用户名    文件名    删除指定用户的acl权限

#setfacl -x  g:组名   文件名    删除指定用户组的acl权限

#setfacl  -b   文件名    删除文件的所有的acl权限

**4，默认的ACL权限和递归ACL权限**

**递归acl权限**

递归是父目录在设定acl权限时，所有的子文件和子目录也会拥有相同的acl权限。

#setfacl -m u:用户名:权限 -R 文件名（目录）

**默认ACL权限**

默认acl权限的作用是如果给父目录设定了默认acl权限，那么父目录中所有新建的子文件都会继承父目录的acl权限。

#setfacl -m d:u:用户名:权限   文件名（目录）

**文件特殊权限   （SetUID，SetGID,Sticky BIT）**

-----------------------------

**SetUID的功能**

只有可以执行的二进制程序才能设定SUID权限，

命令执行者要对该程序拥有x（执行）权限

命令执行者在执行该程序时获得该程序文件属主的身份

setuid权限只在该程序执行过程中有效，也就是说身份改变只在程序执行过程中有效

SetUID：

思考：为什么普通用户可以更改密码？

SetUID的定义：当一个可执行程序具有SetUID权限，用户执行这个程序时，将以这个程序所有者的身份执行。

范例：1、将touch命令授予SetUID权限

          2、当vi命令被授予SetUID权限

          3、查找SetUID程序：

                find / -perm -4000 -o -perm -2000

SetUID=4 所有者

**设置和取消setuid权限**

chmod   u+s   u-s

chmod  4755  755

**SetGID针对文件的作用**

只有可以执行的二进制程序才能设定SGID权限，

命令执行者要对该程序拥有x（执行）权限

命令执行者在执行该程序时,组身份升级为该程序文件的属组

setgid权限只在该程序执行过程中有效，也就是说组身份改变只在程序执行过程中有效

**SetGID针对目录的作用**

普通用户必须对此目录拥有r和x权限，才能进入此目录

普通用户在此目录中的有效组会变成此目录的属组

若普通用户对此目录拥有w权限时，新建的文件的默认属组是这个目录的属组

setgid权限只在该程序执行过程中有效，也就是说组身份改变只在程序执行过程中有效

locate命令设置了setGID权限

SetGID=2 所属组

**设置和取消setgid权限**

chomd  g+s   ug+s   g-s

chomd  2755  6755   755

一旦设置了suid或者guid，一个 s 将出现在 x的位置上。记住：在设置 s u i d或g u i d的同时，相应的

执行权限位必须要被设置。

eg:drwsrwsrwx     设置了suid和guid的目录权限  权限为6777

**Sticky BIT**

**SBIT粘着位作用**

粘着位只对目录有效

普通用户对该目录拥有w和x权限，即普通用户可以在此目录拥有写入权限

如果没有粘着位，因为普通用户拥有w权限，所有可以删除此目录下所有文件，包括其他用户建立的文件。一旦赋予了粘着位，除了root可以删除所有文件，普通用户就算拥有w权限，也只能删除自己建立的文件，但是不能删除其他用户建立的文件。

**/tmp目录设置了粘着位权限**

粘着位=1 其他人 （针对目录设置）

t,如果一个权限为777的目录具有粘着位，每个用户都可以在该目录下创建文件，但是只能删除自己是所有的文件

**设置和取消粘着位**

chmod   o+t   o-t     目录名

chmod  1777   777    目录名

eg:drwsrwsrwt    目录权限为 7777

**3，文件系统属性chattr权限**

#chattr [+-=] [选项] 文件或目录名

+：增加权限    -：删除权限    =：等于某权限

选项：

i:如果对文件设置i属性，那么不允许对文件进行删除，改名，也不能添加和修改数据；如果对目录设置i属性，那么只能修改目录下文件的数据，但不允许建立和删除文件。

a:如果对文件设置a属性，那么只能在文件中增加数据（只能以>>方式追加，不能使用vi编辑），但是不能删除也不能修改数据；如果对目录设置a属性，那么只允许在目录中建立和修改文件（只能修改文件内容，不能修改文件名），但是不允许删除。

查看文件系统属性

#lsattr    选项    文件名

选项：

    -a:显示所有文件和目录

    -d:若目标是目录，仅列出目录本身的属性，而不是子文件的