



Российский разработчик и поставщик
решений на основе программного обеспечения
с открытым исходным кодом

Управление доступом к файлам

Индексный дескриптор файла и классы пользователей

В индексном дескрипторе файла (inode) содержатся:

- идентификатор пользователя (UID);
- идентификатор группы (GID);
- права доступа к файлу, под которые выделены 12 бит.

Права доступа к файлам могут назначаться для трёх классов пользователей:

владелец файла — пользователь, чей UID указан в inode файла;

группа — владелец файла (группа владельца) — пользователи, входящие в группу, чей GID указан в inode файла;

все остальные - пользователи, у которых UID не совпадает с UID из inode и которые не входят в группу — владелец файла.

Права доступа к файлам и каталогам

Файл:

r (Read) — чтение содержимого файла

w (Write) — изменение содержимого файла

x (eXecute) — выполнение файла

Каталог:

r (Read) — список файлов (ls имя_каталога)

w (Write) — создание/удаление файла в каталоге
(touch имя_каталога / имя_файла)

x (eXecute) — права доступа к файлам в каталоге (cd имя_каталога)

Специальные биты защиты

suid (SetUID) — процесс выполняется от имени владельца файла (программы)

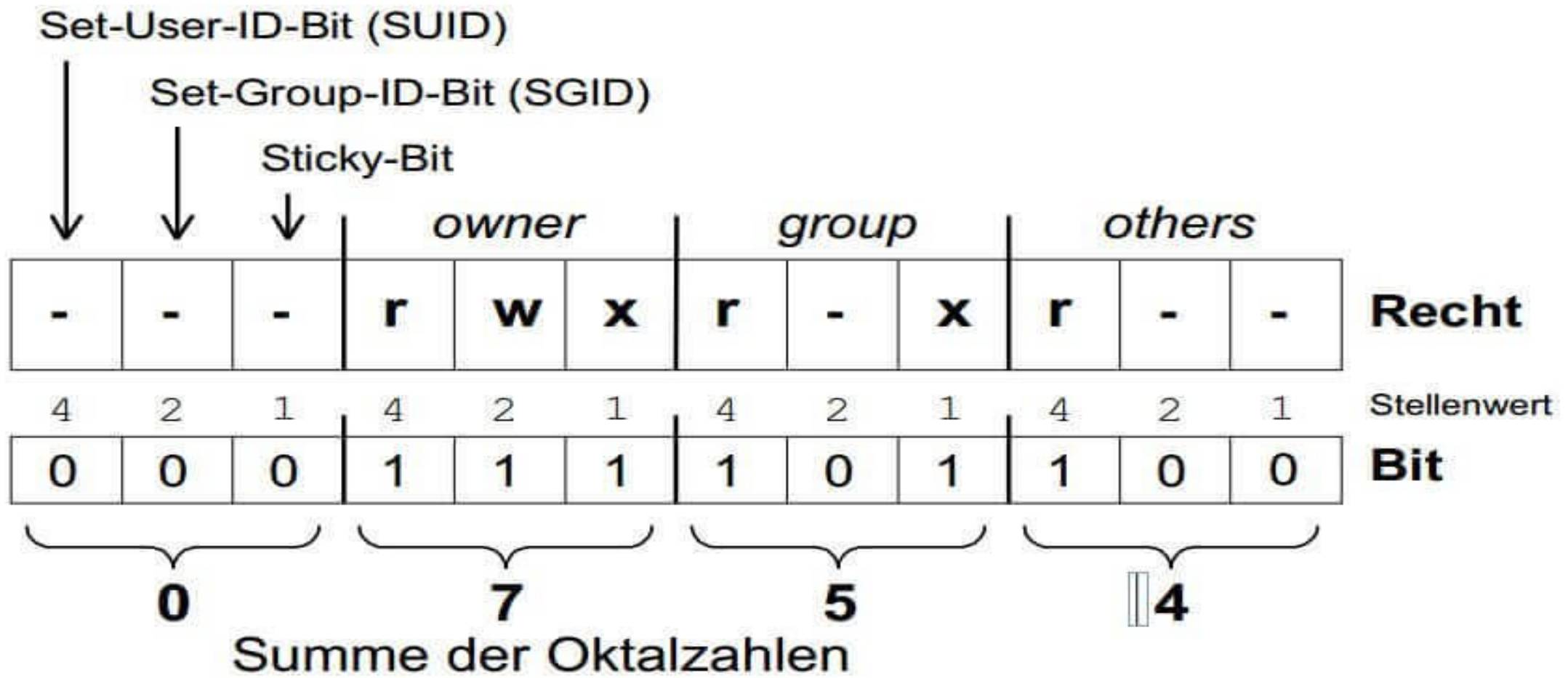
sgid (SetGID) — процесс, при доступе к файлам, использует группу-владельца программы

sticky bit ("липкий" бит) — устанавливается на общие каталоги (все могут создавать файлы, а удалять — только владелец файла)

Права доступа

ОСТ	BIN	Mask	Права на файл	Права на каталог
0	000	---	отсутствие прав	отсутствие прав
1	001	--x	права на выполнение	доступ к файлам и их атрибутам
2	010	-w-	права на запись	отсутствие прав
3	011	-wx	права на запись и выполнение	все, кроме доступа к именам файлов
4	100	r--	права на чтение	только чтение имен файлов
5	101	r-x	права на чтение и выполнение	чтение имен файлов и доступ к файлам и их атрибутам
6	110	rw-	права на чтение и запись	только чтение имен файлов
7	111	rwX	полные права	все права

Маска прав доступа



Управление правами доступа

Утилиты командной строки:

- chown
- chgrp
- chmod
- umask

Менеджеры файлов:

- Caja
- mc (Midnight Commander)

Управление списками прав доступа (ACL)

- ACL для пользователя:
user:имя_пользователя:права
или сокращенная форма
u:имя_пользователя:права (например, rwx)
- ACL для группы:
group:имя_группы:права
или сокращенная форма
g:имя_группы:права
- Маска, определяющая максимальные права для ACL:
mask::права
- На каталог, кроме обычного ACL, можно установить ACL по умолчанию (default). ACL по умолчанию будет автоматически назначаться на все новые файлы в этом каталоге

Управление списками прав доступа (ACL)

ACL атрибуты дополняют обычные атрибуты, которые связаны со всеми inode в файловой системе.

Часто они используются для предоставления дополнительных возможностей средствами файловой системы.

Проверить поддерживаются ли расширенные атрибуты файловой системой:

```
tune2fs -l /dev/mapper/ro-root | grep "Default mount options:"
```

```
Default mount options: user_xattr acl
```

Управление списками прав доступа (ACL)

Просмотр **ACL** для файловых объектов

getfacl [<опции>] <имя_файла_или_каталога>

В листинге будет отображаться знак «+», если **установлены ACL**:

ls -l

-rw--w-r--+ 17 ivanov ivanov 30744 мар 08 15:39 Открытка.jpg

ACL представляют собой пары

имя:значение

Управление списками прав доступа (ACL)

Установка ACL выполняется командой **setfacl**

setfacl <опции> <ключ> <список правил> <объект>

Часто используемые ключи:

-m - модифицирует указанные ACL

-x - удаляет указанные ACL

setfacl -m u:dmitry:rw myfile.txt – Добавляем права пользователю

setfacl -m g:test:r myfile.txt – Добавляем права группе

setfacl -m m:rx myfile.txt – Меняе эффективную маску в rx:

setfacl -m o:- myfile.txt – Убирает все права у «остальных»

setfacl -x g:test myfile.txt – Убираем права группе

Специальные атрибуты файлов и каталогов

Для задания других атрибутов, отличных от стандартных, существует команда **chattr**, позволяющая устанавливать специальные атрибуты на файлы и каталоги. Для просмотра установленных специальных атрибутов используется утилита **lsattr**.

Некоторые атрибуты:

a — файл может быть открыт только в режиме добавления

A — не обновлять время последнего доступа

e — использовать при записи файла экстенды (непрерывные блоки)

i — сделать неизменяемым

s — безопасное удаление с последующей перезаписью нулями

Атрибуты A, s и u может пользователь, все остальные только root

Специальные атрибуты файлов и каталогов

Просмотр специальных атрибутов

lsattr [<опции>] [<имена_файлов_или_каталогов>]

-a — вывод всех объектов, включая скрытые, в указанной папке

-d — вывод информации о каталоге, а не его содержимом

Если файл или каталог не указан, то выводится информация о содержимом текущей папки.

chattr [<опции>] <+/-/=><атрибуты> <имя_файла>

Опции:

-R — рекурсивная обработка каталога

-f — принудительное выполнение, игнорируя ошибки

Внеядерные атрибуты файлов и каталогов

Просмотр и назначение внеядерных атрибутов

setfattr <опции> <имя файла>

-n — задаёт имя атрибута, который необходимо установить

-v — задаёт значение атрибута

-x — удаляет атрибут с указанным именем

getfattr <опции> <имя файла>

-n — задаёт имя атрибута, который необходимо считать

-d — считывает значения всех атрибутов файла

-m — задаёт шаблон имён атрибутов, которые необходимо считать

setfattr -n user.filetype -v text FileName – устанавливаем

getfattr -d FileName – просматриваем

getfattr -m . -d 111 — просмотр всех внеядерных атрибутов

Практическая работа

1. Создайте в РЕД ОС пользователей (user1 и user2) и задайте их пароли. Зарегистрируйтесь в первой консоли как user1.
2. С помощью Ctrl+Alt+F2 (Alt+F2) откройте второй текстовый терминал и зарегистрируйтесь как user2.
3. Просмотрите список основных каталогов в / и укажите, каких прав доступа вам не хватает для входа в каждый из каталогов
4. Создайте два новых временных каталога `mkdir -m 777 /home/temp1` и `mkdir -m 1777 /home/temp2`. Проверьте права доступа к каталогам /home/user1 и /home/user2
5. Задайте права доступа к файлам "по умолчанию". Для этого установите `umask 022`. Поясните назначение `umask`
6. С помощью команды `find` с правами администратора найдите в корневом каталоге файлы имеющие атрибуты SUID (`find / -type f -perm -4000`); файлы, которые разрешено модифицировать всем (`find / -type f -perm -2`); файлы, не имеющие владельца (`find / -nouser`); объясните, какой интерес могут представлять для администратора указанные категории файлов?
7. С помощью конфигурации `pwquality` установить следующую парольную политику: минимальная длина пароля — 8 символов, в нём должны быть минимум 2 цифры и 1 буква в верхнем регистре
8. Добавить каталогу /home/user2 командой `setfacl` права `rw` для пользователя user1
9. Вывести командой `getfacl` списки доступа на этот каталог. Вывод перенаправить в файл `acl.log`
10. Установить бит запрещения изменения на файл `/home/user2/.bash_history`



Спасибо за внимание!

**www.red-soft.ru
redos@red-soft.ru**

