Практична робота № 11. Розмежування прав доступу до файлів та каталогів

Операційні системи сімейства UNIX традиційно розраховані на багато користувачів системи. Щоб почати працювати, користувач повинен увійти в систему, увівши з вільного термінала своє реєстраційне ім'я і пароль. Реєстрацію нових користувачів зазвичай виконує адміністратор системи. Основними мінімальними даними, необхідними для реєстрації користувача в системі, є:

– ім'я користувача;
– назва групи, до якої відноситься користувач;
– пароль.

Системний файл реєстрації користувачів

В UNIX-системах реєстрація користувачів ведеться у файлі **`/etc/passwd`**. Вміст цього файлу є послідовністю текстових рядків. Кожен рядок відповідає одному зареєстрованому в системі користувачеві і містить 7 полів, розділених символами двокрапки. Це такі поля:

1) реєстраційне ім'я користувача;
2) зашифрований пароль;
3) значення UID (ідентифікатор користувача);
4) значення GID (ідентифікатор основної групи, до якої відноситься користувач);
5) коментар (може містити розширену інформацію про користувача, наприклад, ім'я, посаду, телефони та ін.);
6) домашній каталог;
7) командна оболонка користувача.

Приклад рядка з файлу `/etc/passwd` (зауважте, що поле коментаря в цьому випадку відсутнє):

`john:*:1004:101::/home/john:/bin/bash`

Файл ***`/etc/passwd`*** повинен бути доступним для читання всім користувачам, оскільки до нього повинні звертатися багато програм, що запускаються від імені рядового користувача (наприклад, щоб дізнатися відповідність UID реєстраційного імені). Але доступність для читання всіх зашифрованих паролів серйозно зменшує безпеку системи, тому що сучасні обчислювальні потужності дозволяють порівняно швидко підбирати паролі (особливо невдало обрані деякими користувачами).

Тому часто використовується схема тіньових паролів (shadow passwords), за якої поле пароля в ***/etc/passwd*** ігнорується, а реальний пароль береться з іншого файлу (наприклад, ***/etc/shadow***), доступного для читання тільки привілейованому користувачеві. Файл тіньових паролів часто містить й іншу важливу інформацію: термін, протягом якого допускається використання незмінного пароля, дата останньої зміни пароля та ін. У разі використання тіньових паролів друге поле в ***/etc/passwd*** зазвичай містить символ зірочки або будь-який інший довільний символ. Порожнім поле пароля в ***/etc/passwd*** залишати не можна, оскільки в цьому випадку система може порахувати, що цьому користувачеві пароль не потрібний.

Файл реєстрації груп користувачів

Інформація про групи, які відомі системі, міститься у файлі ***/etc/group***. Подібно файлу реєстрації користувачів, інформація в ***/etc/group*** представляє собою набір рядків, по одній для кожної зареєстрованої групи користувачів. Кожен рядок містить чотири поля, розділених двокрапкою:

1) реєстраційне ім'я групи;
2) пароль групи (зазвичай це поле пусте, оскільки групам зазвичай не призначають паролі);
3) значення GID, що відповідає цій групі;
4) розділений комами список користувачів, що входять у групу (може бути порожнім).

Зауважимо, що порожній список користувачів у записі ***/etc/group*** не означає, що в цій групі немає жодного користувача, бо GID основної групи користувача визначається у файлі ***/etc/passwd***.

Визначення ідентифікаторів користувачів і груп

Щоб визначити UID користувача, GID і ім'я його основної групи, а також список інших груп, до якого включений користувач, можна використовувати команду ***id***.

```
$ id [<користувач>]
```

У разі її використання без аргументів, команда виведе інформацію про поточного користувача. Якщо ж вказати як аргумент ім'я зареєстрованого користувача, результат виконання команди буде відповідати зазначеному користувачу.

Окремим випадком команди ***id*** є команда ***groups***, яка видає список імен всіх груп, в яких розташований поточний або вказаний користувач.

```
$ groups [<користувач>]
```

Перегляд і інтерпретація прав доступу до файлів

У UNIX базові права доступу до файлів включають 3 складові:

- дозвіл на читання (позначається літерою «***r***», від слова ***Read***);
- дозвіл на запис (літера «***w***», від слова ***Write***);
- дозвіл на виконання (літера «***x***», від слова ***eXecute***).

Нижче у таблиці наведені дії, які дозволяють виконувать відповідні дозволи для файлів і каталогів.

Дозвіл	*Дії з файлами*	*Дії з каталогами*
читання	читати вміст файлів	переглядати перелік імен файлів у каталозі (наприклад, командою ***ls***)
запис	змінювати вміст файлу	створювати у каталозі нові файли і каталоги або видаляти їх
виконання	запускати файли на виконання (як програми в машинному коді і командні файли)	переходити в цей каталог (тому для каталогів право виконання часто називають правом пошуку)

Якщо на файлі стоїть атрибут виконання, то незалежно від його імені він вважається програмою, яку можна запустити (на відміну від DOS або Windows, в UNIX можливість виконання файлу не залежить від розширення імені файлу, такого як .exe).

Відзначимо, що для каталогів біти читання і виконання (***r*** і ***x***) частіше використовуються в парі, тобто або присутні обидва, або відсутні.

В атрибутах доступу до файлів перераховані типи прав доступу можуть бути надані для 3 класів користувачів:

- власника (у кожного файлу в UNIX є один власник);
- групи (з кожним файлом пов'язана група користувачів цього файлу);
- всіх інших користувачів.

Набір прав доступу для конкретних файлів можна переглянути за допомогою команди ***ls -l***. Наприклад:

```
$ ls -l tmp/
drwxrwxr-x 10 john users 1024 Jan 13 newdir
-rw-r----- 1 john users 173727 Jan 13 23:48
archive-0113.zip
```

У цьому прикладі видно, що власником файлів є користувач ***john***, а групою власників є група ***users***. Набір літер і прочерків у лівій частині визначає тип файлу (перший символ) і права доступу до файлу (решта дев'ять символів).

У наведеному прикладі перший запис відноситься до каталогу (перша літера ***d***) і демонструє права доступу ***rwxrwxr-x***. Другий запис відноситься до звичайного файлу (прочерк на місці першого символу) і показує права ***rw-r-----***. Дев'ять символів прав доступу визначають можливість читання (***r***), записи (***w***) і виконання (***x***) для власника файлу (перші 3 символи), групи власника (наступні 3 символи) і всіх інших (останні 3 символи). Прочерки означають відсутність відповідних прав. Отже, в наведеному прикладі:

– ***john*** і всі користувачі групи ***users*** можуть переглядати і змінювати вміст каталогу ***newdir***, а також переходити в нього, а інші користувачі можуть читати і переходити в цей каталог, але не можуть створювати або видаляти в ньому нові файли;

– ***john*** може читати і змінювати файл ***archive-0113.zip***, користувачі групи users можуть тільки читати вміст цього файлу, а всі інші не мають до нього ніяких прав доступу.

Крім символьного подання прав доступу, часто використовується цифрова форма. У цифровому поданні права доступу складаються з 3 вісімкових цифр, кожна з яких визначає набір з трьох бітів повноважень ***rwx***. Щоб перевести права доступу з символьного представлення в числове, слід:

– представити набір прав в двійковому вигляді (наприклад, ***110100000*** для набору прав ***rw-r-----***);

– перевести отримане двійкове число в вісімкову систему числення (наприклад, вісімковим представленням двійкового числа ***110100000*** буде ***640***).

Права доступу так само можна у числовій формі шляхом підсумовування вісімкових значень окремих бітів прав доступу:

– ***400*** – власник має право на читання;
– ***200*** – власник має право на запис;
– ***100*** – власник має право на виконання;
– ***040*** – група має право на читання;
– ***020*** − група має право на запис;
– ***010*** – група має право на виконання;
– ***004*** – інші мають право на читання;
– ***002*** – інші мають право на запис;
– ***001*** – інші мають право на виконання.

Можна помітити, що для прав доступу ***rw-r-----*** отримаємо: ***400 + 200 + 040 = 640***.

Зміна власників файлів

Власником файлу стає користувач, який створив цей файл. Групою власника за замовчуванням стає основна група реєстрації користувача. Для зміни власників призначена стандартна команда ***chown*** (change owner). Однак у сучасних системах власника файлів може змінювати тільки привілейований користувач (root). У звичайного користувача існує можливість зміни тільки групи власників, і то лише в межах тих груп, в які входить сам користувач. Для зміни групи власників зручно використовувати команду ***chgrp*** (change group). Наприклад, щоб зробити групою власників каталогу ***newdir*** групу ***students***, можна ввести:

```
$ chgrp student newdir
```

Існує можливість рекурсивної зміни власників для всіх файлів і підкаталогів заданого каталогу. Для цього слід використовувати ключ ***-R***, наприклад:

```
$ chgrp -R student newdir
```

Зміна прав доступу до файлів

Змінити права доступу до файлу може або його власник, або привілейований користувач (***root***). Робиться це командою ***chmod*** (Change mode). Існує два формати використання цієї команди: з використанням символьного і числового представлення прав доступу.

Використання числового представлення дозволяє однією командою змінити повний набір прав доступу, наприклад:

```
$ chmod 770 newdir
```

Ця команда встановить права доступу в числове значення ***770***, тобто ***rwxrwx---***, що дасть повні права власнику і групі власника, і ніяких прав всім іншим.

Використання символьного представлення прав доступу в команді ***chmod*** може здатися дещо складнішим, але дозволяє маніпулювати окремими бітами прав доступу. Наприклад, щоб зняти біт запису для групи власника каталогу ***newdir***, достатньо ввести:

```
$ chmod g-w newdir
```

Умовний синтаксис цієї команди такий:

```
$ chmod {u, g, o, a} {+, -, =} {r, w, x} <файл ...>
```

Як аргумент команда приймає вказівку класів користувачів:

– «***u***» – власник-користувач (***user***);
– «***g***» – власник-група (***group***);

– «***o***» – інші користувачі (***others***);
– «***a***» – всі перераховані вище групи разом (***all***).

Операцію, яку необхідно зробити з правами доступу:
– «+» – додати;
– «-» – прибрати;
– «=» – привласнити.

Права доступу («***r***», «***w***», «***x***») можна призначити каталогам і файлам.

Як і у команді ***chgrp***, у ***chmod*** може використовуватися ключ ***-R***, що дозволяє рекурсивно обробляти вміст підкаталогів.

Встановлення права доступу за замовчуванням

Очевидно, що під час створення нових файлів і каталогів вони вже будуть мати певний набір прав доступу. Ці права доступу, що встановлюються за замовчуванням, визначаються значенням маски прав доступу, яка встановлюється командою ***umask***. У результаті введення цієї команди без аргументів вона виведе поточне значення маски, у разі використання вісімкового числа як аргумент буде встановлено нове значення.

Маска прав доступу визначає, які права мають бути видалені з повного набору прав, тобто маска прав доступу є в деякому роді зворотним значенням прав доступу. Наприклад, маска ***022*** призведе до скидання бітів запису для групи власника та інших користувачів. Зауважимо, що для звичайних файлів (НЕ каталогів) всі біти виконання (***x***) в правах за замовчуванням будуть скинуті незалежно від поточної маски.

Приклад, що демонструє ефект команди umask:

```
$ umask
002
$ mkdir dir1
$ ls -l
drwxrwxr-x 2 john users 1024 Apr 21 7:29 dir1
$ umask 072
$ umask
072
$ mkdir dir2
$ ls -l
drwxrwxr-x 2 john users 1024 Apr 21 7:29 dir1
drwx --- r-x 2 john users 1024 Apr 21 7:30 dir2
```

Завдання

1. Знайти запис у файлі **`/etc/passwd`**, що відповідає вашому реєстраційному імені.
2. Визначити свій UID, дізнатися, до яких груп належить ваше реєстраційне ім'я, пояснити результати виконання команд **`id`**, **`groups`**.
3. Визначити межі файлового простору, де система дозволяє створювати власні файли і каталоги.
4. Перевірити, чи можливе втручання в приватний файловий простір іншого користувача системи:
   – який входить до тієї самої групи, що й ви;
   – який входить до будь-якої іншої групи.
5. Дізнатися, які права доступу мають новостворювані файли і каталоги (тобто створити новий файл і новий каталог і переглянути для них права доступу).
6. Визначити значення umask, за якого створювані файли і каталоги будуть недоступні для читання, запису і виконання всім, окрім власника.
7. Зробити свій домашній каталог видимим для всіх користувачів групи, до якої ви належите.
8. Створити в домашньому каталозі підкаталог **`tmp`**, файли в якому зможе створювати, видаляти і перейменовувати будь-хто, що входить до групи вашої групи, при цьому вміст цього підкаталогу не має бути видимим всім іншим користувачам системи.

Контрольні питання

1. Як кодуються в атрибутах файлу і каталогу права доступу? Які формати записи прав бувають?
2. Кто може змінювати права доступу до файлів?
3. Які команди для зміни символьних кодів прав доступу ви знаєте? Перерахуйте і розкажіть про призначення кожної з команд.
4. У чому різниця в застосуванні команд chmod і umask?
5. Які команди обробки файлів дозволяють (або забороняють) права на читання, запис і виконання?
6. Які команди обробки каталогів дозволяють (або забороняють) ці ж права?
7. Що означає право на виконання стосовно каталогу?
8. Які правами треба володіти, щоб видалити файл або каталог?
9. Яке символьне значення запису прав доступу відповідає вісімковому значенню **`641`**?
10. Яке вісімкове значення запису прав доступу відповідає символьному значенню **`rw-r------`**?