Самостійна робота №1.

Виконав Вальдман Дмитро. ТК-31.

1) Підписати елементи наступного знімку екрану:

$ ls -l

total 86816

-rwxr-xr-x   1 manageme manageme    12349 Apr  3  2006 cg

-rw-r--r--   1 manageme manageme      774 Apr  3  2006 cg.cpp

drwxr-xr-x   2 manageme manageme     4096 Oct 13  2007 cgi-bin

-rw-r--r--   1 manageme manageme 88295680 Oct 14 20:55 error\_log

drwxr-x---  13 manageme mail         4096 Nov 26  2008 etc

drwx------  17 manageme manageme    65536 Oct  1 20:09 logs

drwxr-x---  92 manageme nobody       4096 Aug  7 13:32 public\_html

lrwxrwxrwx   1 root     root           11 Nov 29  2007 www -> public\_html

1 рядок – команда «отримати список елементів директорії (ls) у вигляді “довгого списку” (-1)»

2 рядок – кількість зайнятих блоків

По колонкам:

1 – права доступу

2 – кількість шляхів до файлу

3 – ім'я власника

4 – ім'я групи

5 – розмір у байтах

6 – дата та час останньої зміни

7 – назва файлу

2) Що означають літери «r, w, x» у першому стовпчику лістингу?

r – доступ на читання (**r**ead)

w – доступ на запис (**w**rite)

x – доступ на виконання (e**x**ecute)

s – suid-біт в трійці користувача; sgid-біт в трійці групи; при цьому x має бути встановлений

S – те саме, що s, тільки x не встановлений

Які особливості атрибуту "x" для каталогів?

Дозволяє доступ каталогів у поточному каталозі, робить його доступним через команду cd; даний атрибут "наслідується" від батьківських директорій: якщо будь-яка з батьківських директорій даної не має атрибуту "x", то дана директорія недоступна незалежно від того, які права на доступ у неї самої.

3) Які права доступу має кожний з наведених елементів каталогу?

cg – виконуваний файл з повними правами для власника, правом на читання та виконання для групи та інших

cg.cpp – невиконуваний файл з правами запису і читання для власника, правами читання для групи та інших

cgi-bin – директорія, повні права для власника, права на читання та виконання для групи та інших

error\_log – невиконуваний файл з правами на читання і запис для власника, тільки на читання для інших

etc – директорія з повними правами для власника, правом на читання та виконання для групи, відсутністю прав для інших

logs – директорія з повними правами для лише власника

public\_html – директорія з повними правами для власника, правом на читання та виконання для групи, відсутністю прав для інших

www – символьне посилання (soft link) на public\_html з повними правами для всіх

4) Що означало б, якби перша літера у правах доступу dr--r--r-- була:

- – файл

d – директорія

b – блочний пристрій

c – символьний пристрій

l – символьне посилання (soft link)

5) Якщо в назві файлу зустрічається "->" – це означає…

Що файл це посилання (soft link).

6) Що таке setuid, setgid? На що вони впливають? Як їх змінити?

Для виконуваних файлів це атрибути, що дозволяють користувачам, які запускають ці файли, одержати права власника (setuid) чи групи (setgid).

При застосуванні setgid до директорії нові файли та директорії, що створюються в ній, успадковують права власності її групи, а не права групи користувача, що створив файл.

Застосування setuid до директорій в більшості Unix-подібних системах, як правило, ні на що не впливає (ігнорується системою).

Ці команди впливають відповідно на можливість користувачів виконувати ті чи інші дії над файлами, права на які встановлені у власника чи групи.

Змінити:

• Для файлу:

chmod 2 file – для setuid

chmod 4 file – для setgid

Або:

chmod u+s file – для setuid

chmod g+s file – для setgid

• Для директорії:

chmod g+s file – для setgid

setuid ігнорується

7) Що таке SUID, SGID, sticky bit? На що вони впливають та як змінюють поведінку файлів, на які вони встановлені?

SUID (Set User ID) – параметр, що дозволяє виконати файл з правами його власника.

SGID (Set Group ID) – параметр, що дозволяє виконати файл з правами групи, яка має до нього доступ; для директорії – отримати права власності групи цієї директорії на створювані в ній файли та директорії.

Sticky bit – атрибут, що використовується для захисту файлів: дозволяє власнику каталогу (і тільки йому), на який цей атрибут встановлений, лише видаляти або перейменовувати файли, що знаходяться у цьому каталозі.

8) Які задачі вирішують та які параметри наступних команд?

• chown: зміна власника файлу

chown [-cfhvR] [--dereference] [--reference=rfile] user[:group] file

Параметри:

-c, --changes: детально описувати дію для кожного файлу, власник якого змінюється

-f, --silent, --quiet: не видавати повідомлення про помилки для файлів, чий власник не може бути змінений

-h, --no-dereference: опрацьовувати символьні посилання, а не файли, на які вони вказують. Дана опція доступна, лише якщо наявний системний виклик lchown

-R, --recursive: рекурсивна зміна власника каталогу та його вмісту

-v, --verbose: детальний опис дії (або її відсутності) для кожного файлу

--dereference: змінити власника файлу, на який вказує символьне посилання, замість самого символьного посилання.

--reference=rfile: змінити власника файла на такі самі, як у rfile.

• chgrp: зміна групи для файлу

chgrp [options]… GROUP FILE

Параметри:

Такі самі, як в chown

• umask: встановлення або отримання маски прав доступу для файлів. Зміна – побітове "І" між побітовим "НІ" аргумента та режимом повного доступу (777). Тобто, якщо аргумент позначити A, то матимемо: ~A & 777.

Параметри:

umask [-S] [mask]

-S: в символьній нотації

• chmod: зміна прав доступу на файл

chmod [options] mode[,mode] file1 [file2 …]

Параметри:

-R: рекурсивна зміна прав доступу для каталогів та їх вмісту

-f: не видавати повідомлення про помилки для файлів, чиї права доступу не можуть бути змінені

-v: детальний опис дії (або її відсутності) для кожного файлу