

Lab 3. System plikowy cd., lista kontroli dostępu ACL (Access Control List).

1. Ustawienia maski.

Standardowo w systemie Linux - jeżeli maska jest wyłączona - uprawnienia są równe:

- dla tworzonego pliku **rw-rw-rw-**
- dla tworzonego katalogu **rwxrwxrwx**

Plikom i katalogom tworzonym w systemie linux nadawane są prawa dostępu zależne od maski, którą ustawia się poleceniem **umask**. Jest to polecenie wewnętrzne (wbudowane) powłoki. Polecenie **umask** określa, które prawa mają być zamaskowane. Np. polecenie:

```
$ umask 077
```

sprawi, że nowoutworzone pliki i katalogi nie będą miały żadnych praw na poziomie grupy oraz reszty użytkowników systemu:

```
$mkdir katalog
```

```
$touch plik
```

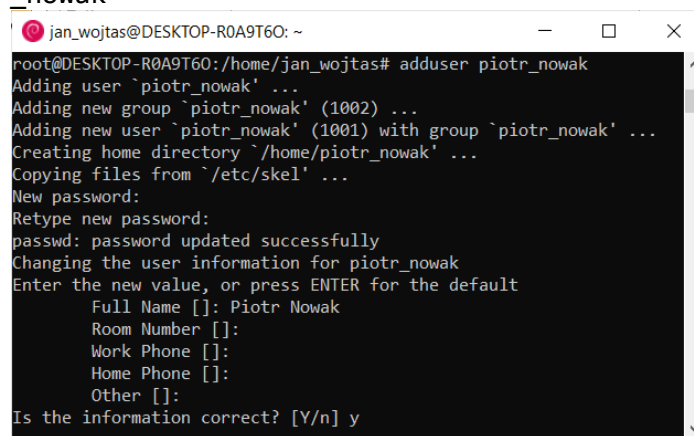
Poniższe polecenie pokazuje prawa do pliku plik i katalogu katalog – zostały one nadane zgodnie z maską 077:

```
$ls -ld katalog plik
drwx----- 2 anka anka 4096 2009-03-24 17:44 katalog
-rw----- 1 anka anka 0 2009-03-24 17:44 plik
```

2. Użytkownik i grupa podstawowa użytkownika – wybrane polecenia.

Tworzenie kont użytkowników (z przypisaniem do domyślnej grupy):

```
adduser piotr_nowak
```

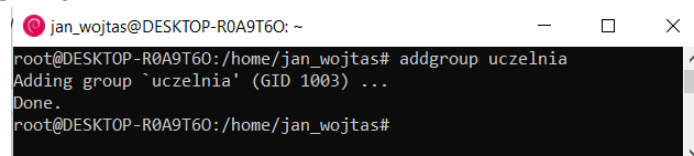


Wpis w /etc/passwd (lub w bazie):

```
piotr_nowak:x:1001:1002:Piotr Nowak,,,:/home/piotr_nowak:/bin/bash
```

Tworzenie grup:

```
addgroup uczelnia
```



Wpis w /etc/group (lub w bazie):

```
uczelnia:x:1003:
```

Przypisanie użytkownika do grupy (uwaga: usuwa poprzednie przypisania do grup – zmienia podstawową grupę użytkownika!!!):

```
usermod -g uczelnia piotr_nowak
```

Dodanie użytkownika do kolejnej - istniejącej grupy:

```
usermod -aG szkoła piotr_nowak
```

Wypisanie listy grup do których należy użytkownik:

```
groups piotr_nowak
```

```
piotr_nowak : uczelnia szkoła
```

3. Lista kontroli dostępu ACL.

Polecenie **chmod** systemu linux pozwala definiować prawa dostępu do plików na trzech poziomach: dla właściciela pliku, grupy oraz pozostałych użytkowników systemu. W wielu przypadkach takie możliwości nie są wystarczające. Nie można przy jego pomocy np. nadawać praw dostępu do plików dla wybranych użytkowników lub grup. Rozwiązaniem tego problemu jest stosowanie listy kontroli dostępu (ACL – Access Control List), która jest dostępna w większości dystrybucji Linuksa.

Podstawowym uprawnieniom odpowiadają odpowiednie wpisy na liście ACL. Można je wyświetlić za pomocą polecenia **getfacl**. Polecenie **getfacl** wyświetla informacje o pliku: nazwę, właściciela, grupę oraz ACL, czyli listę wszystkich ustawionych praw dostępu do pliku. Każda linia opisuje prawa dostępu do jednej z trzech podstawowych klas użytkowników:

```
$getfacl katalog plik
# file: katalog
#owner: anka
#group: anka
user::rwx
group:---
other:---
#file: plik
#owner: anka
#group: anka
user::rw-
group:---
other:---
```

Do modyfikacji listy ACL służy polecenie **setfacl**. Umożliwia ona nadawanie praw dostępu do pliku dla dowolnego użytkownika oraz dowolnej grupy. Na liście ACL występuje pięć typów wpisów. Określają one:

- ACL_USER_OBJ - prawa użytkownika (odpowiada podstawowym prawom na poziomie właściciela pliku - user)
- ACL_USER - prawa dowolnego użytkownika
- ACL_GROUP_OBJ - prawa grupy (odpowiada podstawowym prawom dla grupy -group)
- ACL_GROUP - prawa dowolnej grupy
- ACL_MASK – maskę (określa maksymalne prawa dostępu do pliku dla wszystkich użytkowników z wyjątkiem właściciela (user) oraz innych (other)
- ACL_OTHER - prawa innych (odpowiada podstawowym prawom dla reszty użytkowników systemu - other).

Aby nadać użytkownikowi jas prawo pisania do pliku plik należy użyć opcji m (modify) polecenia setfacl:

```
#setfacl -m u:jas:w plik
```

Polecenie ls -l sygnalizuje istnienie listy ACL symbolem +, który pojawia się za prawami dostępu do pliku:

```
-rw--w----+ 1 anka anka 0 2009-03-24 17:44 plik
```

Pojawiło się prawo w dla grupy, które bez stosowania ACL oznaczałoby możliwość zapisu do pliku plik dla członków grupy pliku. Jednak w przypadku ACL efektywne prawa dostępu dla członków grupy tworzone jest na podstawie wpisu dotyczącego praw grupy (w tym przypadku: ---).

Polecenie **getfacl --omit-header** wyświetla pełne informacje o prawach dostępu do pliku z pominięciem trzech linii nagłówkowych:

```
user::rw-
user:jas:-w-
group:----
mask::-w-
other:----
```

Poza wpisem określającym prawa dla użytkownika jas pojawił się wpis określający maskę. Dotyczy ona wszystkich użytkowników (poza właścicielem) i grup z listy ACL i oznacza, że prawo, które nie występuje w masce nie będzie nadane. W naszym przypadku żaden użytkownik, ani grupa nie uzyskają praw r i x. Maska -w- i blokuje prawo odczytu pliku oraz wykonania dla wszystkich użytkowników z pominięciem właściciela pliku.

Do usuwania wpisów z listy ACL służy opcja x polecenia setfacl.

Polecenie:

```
#setfacl -x u:jas plik
```

usuwa prawa dostępu do pliku plik dla użytkownika jas. Z listy ACL pliku plik usuwane są wpisy dotyczące użytkownika jas.

Jeśli po nadaniu uprawnień do pliku dla określonego użytkownika zmienimy maskę pliku, prawa dostępu do pliku mogą zostać ograniczone. Zbyt restrykcyjna maska blokuje możliwość dostępu do pliku dla wszystkich użytkowników. W poniższym przykładzie użytkownik jas uzyskuje prawa rwx do katalogu kat, a następnie w wyniku polecenia chmod maska pliku zostaje zmodyfikowana i w efekcie użytkownik jas traci prawa rw do pliku. Polecenie getfacl sygnalizuje tę sytuację wyświetlając efektywne prawa użytkownika.

Przykład:

```
#umask 077
#mkdir kat
#ls -ld kat
drwx----- 1 anka anka 0 2009-03-26 17:44 kat
#setfacl -m u:jas:rwx kat
#ls -ld kat
drwxrwx---+ 1 anka anka 0 2009-03-26 17:44 kat
#getfacl --omit-header kat
user::rwx
user:jas:rwx
group:----
mask::rwx
other:----
#chmod g=x kat
#ls -ld kat
drwx--x---+ 1 anka anka 0 2009-03-26 17:44 kat
#getfacl --omit-header kat
user::rwx
user:jas:rwx #effective:--x
group:----
mask:--x
other:----
```

Po nadaniu pełnych praw dla grupy użytkownik jas odzyskuje prawa rwx do katalogu kat:

```
#chmod g=rwx kat
#getfacl --omit-header kat
user::rwx
user:jas:rwx
group:---
mask::rwx
other:---
```

Przy każdej próbie dostępu do pliku wykonywany jest algorytm, który sprawdza poprawność odwołania przeglądając wpisy w ACL. W poniższym algorytmie zmienna ACCES przyjmuje wartość TRUE, gdy dostęp jest dozwolony, a FALSE w przeciwnym przypadku.

Przyjęte oznaczenia:

EUID – efektywny identyfikator użytkownika

EGID – efektywny identyfikator grupy

USER – właściciel pliku

GROUP – grupa pliku

```
if ( EUID==USER )
    if ( wpis ACL_USER_OBJ zawiera odpowiednie prawa )
        ACCESS=TRUE
    else
        ACCESS=FALSE
else if ( EUID odpowiada wpisowi ACL_USER )
    if( wpis ACL_USER zawiera odpowiednie prawa && wpis ACL_MASK zawiera    odpowiednie
prawa)
        ACCESS=TRUE
    else
        ACCESS=FALSE
else if ( EGID lub ID jednej z grup procesu odpowiada GROUP lub wpisowi ACL_GROUP)
    if ( istnieje wpis ACL_MASK )
        if ( ACL_MASK zawiera odpowiednie prawa && któryś z powyższych wpisów
dla grupy zawiera odpowiednie prawa)
            ACCESS=TRUE
        else
            ACCESS=FALSE
    else
        if( któryś z powyższych wpisów dla grupy zawiera odpowiednie prawa)
            ACCESS=TRUE
        else
            ACCESS=FALSE
else if ( wpis ACL_OTHER zawiera odpowiednie prawa )
    ACCESS=TRUE
else
    ACCESS=FALSE
```

*Treści oznaczone kursywą pochodzą z różnych źródeł internetowych.