Mechanizmy lokalnej kontroli dostępu (ACL)

1. Wprowadzenie

Mechanizm POSIX ACL został opracowany, aby rozszerzyć standardowy mechanizm uprawnień, który kontroluje dostęp do pliku (lub katalogu) dla właściciela, grupy oraz innych użytkowników w systemach Linux/Unix. Rozszerzenie dotyczy możliwości definiowania uprawnień dla wskazanych użytkowników i/lub grup. W przypadku *minimal* ACL, dostępnych domyślnie w systemie Unix/Linux, uprawnienia ograniczają się do prawa odczytu (read), zapisu (write) i wykonywania/przeszukiwania (execute).

Używanie ACL jest możliwe w wielu rodzajach systemów plików w systemie operacyjnym Linux, np.: ext2, ext3, reiserfs, nfs.

Mechanizm ACL jest również wspierany w systemach firmy Microsoft. Jednak nie zachowują one pełnej zgodności ze standardem POSIX i opierają się jedynie o system plików NTFS.

Celem ćwiczenia jest przyjrzenie się możliwościom lokalnej kontroli dostępu w systemach Linux oraz Windows.

2. Lokalna Kontrola Dostępu w systemie Linux

Algorytm sprawdzania uprawnień dostępu

Standard POSIX ACL oferuje możliwość maskowania uprawnień poprzez pole maski. Efektywnie uprawnienia do pliku są sumą bitową uprawnień użytkownika/grupy i maski.

Kolejne kroki algorytmu sprawdzenia uprawnień dostępu:

- jeśli użytkownik jest właścicielem pliku – dopuść,
- jeżeli użytkownik jest na liście nazwanych użytkowników i ma odpowiednie efektywne uprawnienia – dopuść,
- jeżeli jedna z grup użytkownika jest grupą właściciela i posiada odpowiednia efektywne prawa – dopuść,
- jeżeli jedna z grup użytkownika występuje jako grupa nazwana i posiada odpowiednie efektywne prawa – dopuść,
- jeżeli jedna z grup użytkownika jest grupą właściciela lub należy do grup nazwanych, ale nie posiada dostatecznych efektywnych uprawnień – dostęp jest zabroniony,
- następnie uprawnienia innych (others) określają możliwość dostępu.

Polecenia

Dostępne są dwa polecenia, jedno służy do odczytu rozszerzonych praw, drugie do ich ustawiania:

- getfacl
- setfacl

1. Polecenie getfacl

Program wypisuje rozszerzone uprawnienia do plików i katalogów.

```
% ls -l
 -rw-r--r-- 1 user group 1000 2004-10-01 09:00 plik.txt

% getfacl plik.txt
# file: plik.txt
# owner: user
# group: group
user::rw-
group::r--
other::r--

% getfacl plik.txt --omit-header
user::rw-
group::r--
other::r--
```

2. Polecenie setfacl

Polecenie pozwala zmienić, dodać lub usunąć uprawnienia z rozszerzonych uprawnień.

Dodanie uprawnień:

```
% setfacl -m user:kowalski:rwx plik.txt
% getfacl plik.txt --omit-header
user::rw-
user:kowalski:rwx
group::r--
mask::rwx
other::r--
```

Zmiana uprawnień:

```
% setfacl -m u:kowalski:r plik.txt
% getfacl plik.txt --omit-header
user::rw-
user:kowalski:r
group::r--
mask::rwx
other::r--
```

Usunięcie uprawnień:

```
% setfacl -x u:kowalski plik.txt
% getfacl plik.txt --omit-header
user::rw-
group::r--
mask::r--
other::r--
```

Uprawnienia domyślne dotyczą tylko katalogów i umożliwiają automatyczne nadawanie rozszerzonych uprawnień do nowotworzonych plików/katalogów w katalogu, któremu nadaliśmy uprawnienia domyślne.

Dodanie uprawnień domyślnych:

```
% setfacl -d -m group:students:wx katalog
% getfacl katalog --omit-header
user::rwx
group::r-x
other::r-x
default:user::rwx
default:group::r-x
default:group:students:-wx
default:mask::rwx
default:other::r-x
```

Modyfikacja i usuwanie domyślnych uprawnień jest analogiczne.

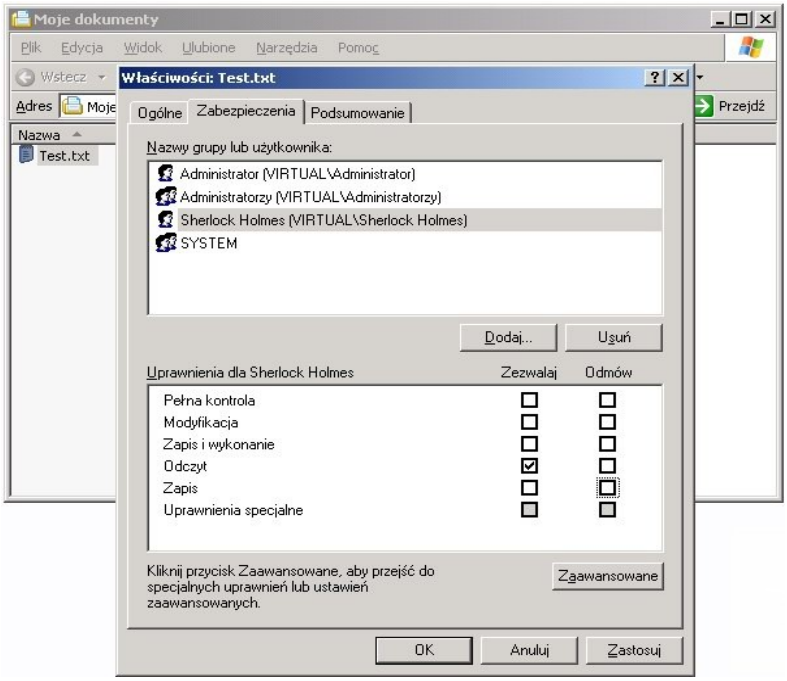
Możliwość ustawiania pola maski jest możliwa poprzez następujące polecenie:

```
% setfacl -m mask::rwx plik.txt
```

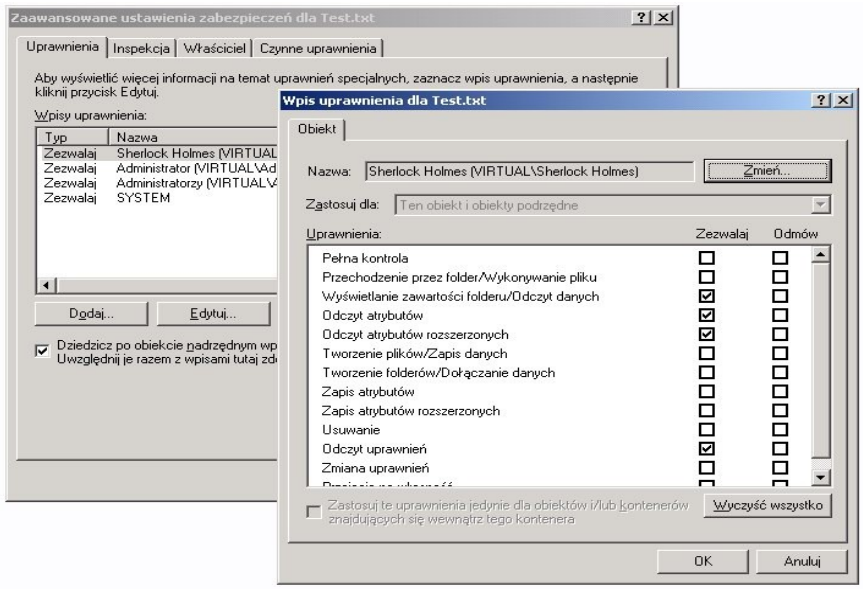
Istnieją jeszcze opcje pozwalające całkowicie skasować rozszerzone uprawnienia (`-b`) oraz domyślne uprawnienia (`-k`).

3. Lokalna Kontrola Dostępu w systemie Windows XP

System plików NTFS umożliwia związanie z każdym zasobem plikowym (w tym: katalogiem) list kontroli dostępu ACL (*Access Control List*). Dostęp do prostych ustawień ACL pliku (katalogu) jest możliwy z poziomu np. Eksploratora Windows w opcji Właściwości (menu Plik lub kontekstowe).



Rozszerzone listy ACL są dostępne po wyborze uprawnień Zaawansowanych.



4. Zadania

- Stworzyć katalog z prawami 0750 i zobaczyć listę rozszerzonych praw do tego katalogu.
- Dodać uprawnienia do zapisu i przeszukiwania do powyższego katalogu koledze siedzącemu przy sąsiednim komputerze.
- Sprawdź czy możesz zapisać jakiś plik do katalogu kolegi. Jeśli nie – sprawdź przyczynę.
- Wykonać chmod g-w <katalog>, następnie sprawdzić czy kolega może stworzyć plik w tym katalogu. Sprawdzić rozszerzone uprawnienia dla tego katalogu.
- Wykonać chmod g+w <katalog> i ponownie wykonać punkt poprzedni.
- Nadać katalogowi uprawnienia domyślne. Stworzyć w katalogu kolegi plik, katalog. Sprawdzić rozszerzone uprawniania nowo utworzonego pliku i katalogu.
- Utworzyć nowy katalog ~/public z uprawnieniami "`wx`" dla grupy "`students`" oraz "`rwx`" dla grupy "`staff`". Dodać domyślne uprawnienia dla samego siebie.
- Utworzyć w katalogu *Moje dokumenty* plik *Test.txt*. Dla tego pliku wykonać następujące operacje:
    - korzystając z prostych ACL nadać uprawnienia do odczytu dla użytkownika **scott**
    - sprawdzić rozszerzone ACL dla tego użytkownika
    - sprawdzić jakie czynne uprawnienia posiada ten użytkownik
- Następnie dla pliku Test.txt:
    - sprawdzić jakie czynne uprawnienia posiada użytkownik **gość**
    - korzystając z prostych ACL odebrać uprawnienia do zapisu użytkownikowi **gość**
    - sprawdzić rozszerzone ACL dla tego użytkownika
    - sprawdzić jakie czynne uprawnienia posiada ten użytkownik
- Sprawdzić czy dla obu tych użytkowników, rzeczywiście mogą oni skorzystać z odpowiednich praw dostępu do analizowanego pliku. Co należy dodatkowo zrobić, aby dostęp ten był możliwy?

5. Problemy do dyskusji

- Porównać mechanizmy ACL systemu Linux i Windows.
- Jakie są wady, zalety poszczególnych mechanizmów ACL?
- Czy warto wykorzystywać bardziej zaawansowane funkcje mechanizmów ACL?
- Czy istnieje lepszy mechanizm ACL niż przedstawione powyżej, jakie?

6. Bibliografia

[ACL] http://acl.bestbits.at/