

# Odpowiedź WDI 2

## Obszar tematyczny 1: Uprawnienia w NTFS (realizacja Windows)

**1. Pliki:** wykazać na przykładzie praw odczytu i zapisu pliku, czy uprawnienia posiadane przez grupę, do której należy użytkownik i przez samego użytkownika sumują się; pokazać konflikt pomiędzy wpisem przyznającym bezpośrednio dla użytkownika a wpisem bezpośrednio odbierającym użytkownikowi to samo uprawnienie odczytu zawartości pliku, wykazać pierwszeństwo jednego z wpisów. W eksperymentach należy wyeliminować wszystkie wpisy z list kontroli dostępu pliku, które zostały dodane w wyniku dziedziczenia.

Robienie nowego katalogu i przechodzenie do niego:

```
mkdir C:/odp
```

```
cd C:/odp
```

Robienie nowego pliku (ja robie 2 pod każdy przypadek)

```
New-Item -Type File plik1
```

```
New-Item -Type File plik2
```

USTAWIĆ TU ZAWARTOŚĆ „echo zawartosc > plik”

Trzeba wyłączyć i usunąć dziedziczone uprawnienia

```
Disable-NTFSAccessInheritance .\plik1 -RemoveInheritedAccessRules
```

```
Disable-NTFSAccessInheritance .\plik2 -RemoveInheritedAccessRules
```

Pierwszy przykład

```
Add-NTFSAccess plik1 -AccessRights Read -AccessType Allow dgradowski
```

```
Add-NTFSAccess plik1 -AccessRights Write -AccessType Allow Użytkownicy
```

Potwierdzenie pierwszego przykładu

```
dgradowski(A)@251524-WDI 21.11.2023 23:56 C:\zad2\odpowiedz> Get-NTFSAccess plik1
```

Account	Access Rights	Applies to	Type	IsInherited	InheritedFrom
BUILTIN\Użytkownicy	Read, Synchronize	ThisFolderOnly	Allow	False	
251524-WDI\dgradowski	Write, Synchronize	ThisFolderOnly	Allow	False	

```
dgradowski(A)@251524-WDI 21.11.2023 23:56 C:\zad2\odpowiedz> cat plik1
chuj
dgradowski(A)@251524-WDI 21.11.2023 23:56 C:\zad2\odpowiedz> echo test > plik1
dgradowski(A)@251524-WDI 21.11.2023 23:56 C:\zad2\odpowiedz> cat plik1
test
dgradowski(A)@251524-WDI 21.11.2023 23:56 C:\zad2\odpowiedz> _
```

Drugi przykład

Add-NTFSAccess plik2 -AccessRights Read -AccessType Allow dgradowski

Add-NTFSAccess plik2 -AccessRights Read -AccessType Deny dgradowski

Potwierdzenie drugiego przykładu:

```
wydziedziczenie: windows PowerShell
dgradowski(A)@251524-WDI 22.11.2023 00:01 C:\zad2\odpowiedz> Get-NTFSAccess plik2

Path: C:\zad2\odpowiedz\plik2 (Inheritance disabled)

Account                Access Rights                Applies to                Type
-----                -
251524-WDI\dgradowski  Read                         ThisFolderOnly           Deny
251524-WDI\dgradowski  Read, Synchronize           ThisFolderOnly           Allow

dgradowski(A)@251524-WDI 22.11.2023 00:01 C:\zad2\odpowiedz> cat plik2
cat : Odmowa dostępu do ścieżki „C:\zad2\odpowiedz\plik2”.
At line:1 char:1
+ cat plik2
+ ~~~~~
+ CategoryInfo          : PermissionDenied: (C:\zad2\odpowiedz\plik2:String) [Get-Content], UnauthorizedAccessException
+ FullyQualifiedErrorId : GetContentReaderUnauthorizedAccessError,Microsoft.PowerShell.Commands.GetContentCommand

dgradowski(A)@251524-WDI 22.11.2023 00:02 C:\zad2\odpowiedz>
```

**2. Katalogi:** wykazać, czy prawo odczytu katalogu jest niezbędne do odczytania zawartości pliku w nim zlokalizowanego, oraz czy jest wystarczające, aby dokonać modyfikacji zawartości pliku oraz zmiany listy plików (np. dodania nowego pliku). Należy zaprezentować wymagane uprawnienie w dostępie do katalogu, które umożliwia dodanie nowego pliku do jego zawartości. W eksperymentach należy wyeliminować wszystkie wpisy z list kontroli dostępu katalogu i pliku, które zostały dodane w wyniku dziedziczenia.

Robienie 2 katalogów:

mkdir folder1

mkdir folder2

ZRÓBCIE W TYCH FOLDERACH JAKIEŚ PLIKI I DAJCIE W NICH ZAWARTOSC ORAZ DAJCIE UPRAWNIENIE DO ODCZYTU DLA UŻYTKOWNIKA

New-Item -Type File folder1/plik

Disable-NTFSAccessInheritance folder1/plik -RemoveInheritedAccessRules

Add-NTFSAccess folder1/plik -AccessRights Read -AccessType Allow dgradowski

New-Item -Type File folder2/plik

idAdd-NTFSAccess folder2/plik -AccessRights Read, Write -AccessType Allow dgradowski

Usuwanie dziedziczenia:

Disable-NTFSAccessInheritance folder1 -RemoveInheritedAccessRules

Disable-NTFSAccessInheritance folder2 -RemoveInheritedAccessRules

Przykład pierwszy:

Get-NTFSAccess folder1 // powinno się nic nie wyświetlić

cat folder1/plik // wypisuje zawartość

Przykład drugi:

Add-NTFSAccess folder2 -AccessRights read -AccessType Allow dgradowski

Get-NTFSAccess folder2

echo zawartosc2 > folder2/plik

New-Item -Type File folder2/plik2 // tu powinien wyskoczyć błąd, przynajmniej mi wyskoczył  
ale zakładam że musi

## **Obszar tematyczny 2: Uprawnienia (realizacja Linux)**

**1. Pliki:** wykazać na przykładzie praw odczytu i zapisu niepustego pliku regularnego, czy uprawnienia posiadane przez grupę, do której należy użytkownik i przez samego użytkownika wykluczają się; zademonstrować przypadek zastosowania praw dla pozostałych (other); obowiązkowe wyjaśnienie znaczenia trzech sekcji uprawnień standardowych.

id // wyświetla identyfikację użytkownika

sg nazwa\_grupy // zmienia grupę bieżącą użytkownika (prawdopodobnie nie trzeba tego robić)

Robienie nowych plików i dodawanie zawartości:

touch plik1

touch plik2

echo zawartosc1 > plik1

echo zawartosc2 > plik2

Przykład pierwszy:

chmod 420 plik1

cat plik1

echo zawartosc\_zmieniona1 > plik1 // tu jest brak dostępu i tak ma być

Przykład drugi:

chmod 006 plik2

cat plik2 // brak dostępu

echo zawartosc\_zmieniona2 > plik2 // brak dostępu

Podsumowanie:

Uprawnienia użytkownika oraz grupy się nie łączą, tak samo jak uprawnienia innych użytkowników.

Komenda 'ls -l' wyświetla zawartość plików wraz z informacją o uprawnieniach, właścicielem pliku oraz grupą do której należy plik.

```
[dgradowski@251524-WDI ttyid:0 wto lis 28 17:17:11 odpowiedz1]$ ls -l
razem 8
-r---w---- 1 dgradowski users 21 11-28 17:09 plik1
-----rw- 1 dgradowski proba  7 11-28 17:16 plik2
```

TYP PLIKU (np. d jeśli katalog lub l jeśli powiązanie)

UPRAWNIENIA WŁAŚCICIELA

UPRAWNIENIA GRUPY

UPRAWNIENIA INNYCH

**r w x**

r - read

w - write

x - execute (wykonywanie)

**2. Katalogi:** wykazać, czy prawo odczytu katalogu jest wystarczające do odczytania zawartości katalogu (wymagane wykonanie wylistowania w formacie długim ls -l dla katalogu zawierającego przynajmniej plik regularny i podkatalog) oraz zawartości pliku w nim położonego; po ustaleniu znaczenia prawa wykonywania dla katalogu w tym scenariuszu sprawdzić, czy prawo to musi być ustawione dla katalogów nadrzędnych wobec badanego, w tym katalogu najbardziej nadrzędnego. Jakie uprawnienie dla katalogu umożliwia zmianę nazwy pliku w nim zawartego i czy nadanie tylko tego uprawnienia dla katalogu będzie skuteczne do wykonania zmiany nazwy pliku.

Trzeba sobie zrobić to o co proszą w zadaniu, czyli jakiś folder z zawartością:

```
mkdir folder1 folder1/folder2
touch folder1/plik1
touch folder1/folder2/plik2
```

W tym momencie polecam użyć sudo su z którego zaraz wyjdziemy:

```
sudo su
chmod 400 folder1 folder1/plik1 folder1/folder2/plik2
chmod 500 folder1/folder2
```

Teraz wychodzimy z sudo i zaczynamy odpowiedź na pierwszą część pytania:

```
ls -l // opcjonalne żeby były widoczne uprawnienia
ls -l folder1
cat folder1/plik
```

```
[dgradowski@251524-WDI ttyid:0 wto lis 28 17:52:55 odpowiedz1]$ ls -l
razem 12
dr----- 3 dgradowski users 4096 11-28 17:42 folder1
-r---w--- 1 dgradowski users   21 11-28 17:09 plik1
-----rw- 1 dgradowski proba   7 11-28 17:16 plik2
[dgradowski@251524-WDI ttyid:0 wto lis 28 17:53:07 odpowiedz1]$ ls -l folder1
ls: nie ma dostępu do 'folder1/plik1': Brak dostępu
ls: nie ma dostępu do 'folder1/folder2': Brak dostępu
razem 0
d???????? ? ? ? ?      ? folder2
-???????? ? ? ? ?      ? plik1
[dgradowski@251524-WDI ttyid:0 wto lis 28 17:53:48 odpowiedz1]$ cat folder1/plik1
cat: folder1/plik1: Brak dostępu
```

ls -l folder1/folder2 // odmówi dostępu bo mimo że folder2 ma wystarczające uprawnienia to folder1 ich nie ma

Część druga:

```
mkdir folder3
touch folder3/plik
echo zawartosc > folder3/plik
chmod 200 folder3
ls -ld folder3
mv folder3/plik folder3/plik2 // nie zadziała, minimalne uprawnienia do zmiany nazwy to
uprawnienie do zapisu (write) i wykonywania (execute)
```

### Obszar tematyczny 3: Właściwości plików, kopiowanie i przenoszenie (realizacja Linux)

**1.** Utworzenie katalogu oraz zawartego w nim nie pustego pliku regularnego, skopiowanie pliku w ramach katalogu przez innego użytkownika, kto jest właścicielem nowego pliku? Które znaczniki czasu pliku i katalogu zmieniły się? Należy również zaprezentować, czy zmieniły się uprawnienia pliku? Skąd domyślnie biorą się uprawnienia nowego pliku? - wymagane zaprezentowanie odpowiedniej wartości.

**2.** Analogiczny eksperyment obejmujący „przeniesienie” (zmianę ścieżki) pliku do utworzonego wcześniej podkatalogu. Które atrybuty pliku zostały zachowane?