Temat: Prawa dostępu i prawa własności do plików w Linux.

Celem ogólnym lekcji jest zapoznanie uczniów z zagadnieniami dotyczącymi praw dostępu i praw własności do plików w systemie operacyjnym Linux.

Cele szczegółowe lekcji:

1. Zrozumienie pojęć praw dostępu i praw własności plików w systemie Linux.
2. Nauczenie się korzystania z polecenia ls -l do wyświetlania informacji o prawach dostępu plików.
3. Poznanie sposobów ustawiania praw dostępu dla użytkowników, grup i innych użytkowników przy użyciu systemu numerycznego i literowego.
4. Przybliżenie idei specjalnych uprawnień (SetUID, SetGID, Sticky Bit).
5. Nauczenie się definiowania domyślnych uprawnień dla nowo tworzonych plików i katalogów przy użyciu polecenia umask.
6. Nauczenie się kopiowania plików między użytkownikami na tej samej stacji oraz zmiany uprawnień do plików poprzez użycie polecenia chmod.
7. Nauczenie się tworzenia katalogów, kopiowania i usuwania plików, zmiany uprawnień do plików oraz korzystania z polecenia rm.
8. Nauczenie się tworzenia użytkowników, logowania na różnych terminalach, tworzenia katalogów i zmiany uprawnień do nich, a także usuwania plików i sprawdzania uprawnień.
9. Nauczenie się tworzenia katalogów, sprawdzania uprawnień do katalogów, kopiowania plików, zmiany umask oraz tworzenia i usuwania plików.
10. Nauczenie się zmiany grupy przypisanej do pliku lub katalogu, korzystania z polecenia chown oraz sprawdzania uprawnień do plików i katalogów.
11. Zwiększenie świadomości w zakresie bezpieczeństwa danych i ochrony prywatności w systemie Linux poprzez odpowiednie zarządzanie prawami dostępu i własnością plików.
12. Umożliwienie uczniom wykonania notatek na temat praw dostępu do plików i katalogów w systemie Linux.

Wprowadzenie

Polecenie ls -l podaje m.in. prawa dostępu do pliku o podanej nazwie (np. test.0), dla właściciela pliku, dla członków grupy i dla pozostałych użytkowników np. **ls -l test.0**

typ obiektu (d, - ,l)	uprawnienia właściciela			uprawnienia grupy			uprawnienia pozostałych użytkowników		
-	r	w	x	r	w	x	r	w	x
	4	2	1	4	2	1	4	2	1

-r (read) - prawo do odczytu

- w (write)- prawo do zapisu

- x (execute) – prawo do uruchomienia

	u	g	o
r	4	4	4
w	2	–	–
x	–	–	–
	6	4	4

$\Rightarrow 644$

Identyfikator właściciela – **u** (user) – użytkownika, który stworzył plik

Identyfikator grupy - **g** (group) – zbiór użytkowników, którzy mają do tego pliku uprawnienia

Pozostali - -**o** (others)

Wszyscy użytkownicy - **a** (all)

- brak praw – 0

chmod [u g o a] [+ - =] [r w x] nazwa_pliku_lub folderu

znak definiujący zmianę praw

- + - nadanie praw
- – - usunięcie praw
- = - ustawienie jedynie praw podanych, pozostałe są usuwane

Poleceniem **chmod** można też ustawić specjalne uprawnienia.

Znak	Numer	Nazwa	Plik	Katalog
t	1	Sticky bit – bit lepkości	Nie dotyczy	Użytkownicy mogą kasować pliki tylko wtedy, gdy są ich właścicielami, właścicielem katalogu. Zwykle stosuje się do katalogu /tmp
s	2	SGID (set GroupID) ustaw ID grupy	Kiedy program startuje, GrupID procesu ustawiony jest na GID grupy pliku.	Pliki tworzone w tym katalogu należą do grupy katalogu a nie użytkownika. Nowe dziedziczą bit
s	4	SUID (set UuserID) ustaw ID użytkownika	Kiedy program startuje, UserID procesu ustawiony jest na UserID grupy pliku.	Nie dotyczy

Można ustawić „sticky bit” poleceniem **chmod**, używając liter lub cyfr. Koniec notatki. (zgłoszenie) 0

Poniżej do zadania są przykłady a nie zadanie do wykonania.

Przykłady zastosowania systemu numerycznego:

chmod 755 nazwa_pliku - nadanie pełnych praw dla użytkownika, odczytu i wykonywania dla grupy i innych do pliku nazwa_pliku

chmod 600 nazwa_pliku - odebranie prawa czytania pliku nazwa_pliku wszystkim, oprócz właściciela pliku (rw)

Przykład zastosowania systemu literowego:

chmod u-x nazwa_pliku - odebranie użytkownikowi prawa do wykonywania

chmod g+w nazwa_pliku - nadanie prawa do zapisu dla grupy

chmod a-x nazwa_pliku - odebranie wszystkim (a=ugo) prawa do wykonywania

chmod a=rwx nazwa_pliku - nadanie wszystkim praw przez przypisanie, w tym przypadku nadanie rwx

chmod go=rx nazwa_pliku - przypisanie grupie i innym prawa do odczytu i wykonywania

Przykład zastosowania praw specjalnych:

chmod u+s system.sh - aktywuje SetUID dla skryptu system.sh

chmod g+s system.sh - oznacza nadanie SetGID

chmod o+t /tmp lub cyfr chmod 1777 /tmp

Podczas sprawdzania praw „sticky bit" zostanie wyświetlony w grupie praw.

Definiowanie domyślnych uprawnień dla plików

Każdy nowo utworzony plik czy katalog posiada z góry zdefiniowane wartości właściciela oraz grupy, a także uprawnienia.

Właścicielem zostaje użytkownik, który stworzył plik lub katalog, natomiast grupą - domyślna grupa właściciela.

Prawa dostępu przypisane automatycznie tworzonym plikom, definiuje maska uprawnień. Aktualną wartość maski uzyskamy poleceniem

umask 0022

Jak to odczytać?

Pierwsza cyfra odpowiada za nieczęsto wykorzystywane prawa specjalne, a w przypadku niektórych dystrybucji Linuksa w ogóle nie jest wyświetlana.

Kolejne to uprawnienia postaci absolutnej, które należy odjąć od pełnych uprawnień katalogu (777) oraz pliku (666 – wykonanie w przypadku plików jest zdecydowanie rzadziej wykorzystywane).

Utworzone katalogi będą miały pełne uprawnienia dla jego właściciela oraz uprawnienia odczytu (wylistowania zawartości) oraz wejścia do folderu dla pozostałych użytkowników.

Pierwszy znaku w czterocyfrowym wyniku umask. - wartość „0" oznacza brak zdefiniowanych praw specjalnych.

Możemy wyświetlić umask w zdecydowanie przyjemniejszej dla oka postaci:

umask -S

u=rwx,g=rx,o=rx

umask, modyfikowanie,

Wystarczy, że jako argument polecenia podamy nową wartość maski - i gotowe.

Aby nadać pełne uprawnienia nowo tworzonych plików dla właściciela, zaś dla reszty uprawnienia zerowe użyjemy polecenia:

umask 0077

Możemy użyć formy „standardowej”:

umask -S u=rwx,g=,o=

Wydanie powyższych poleceń jest jednorazowe - po wylogowaniu i ponownym zalogowaniu umask wróci do domyślnych ustawień. Dodając wpis (np. **umask 0077**) do pliku konfiguracyjnego powłoki.

Dla użytkowników Ubuntu Basha ten plik to ~/.bashrc.

W zeszycie podaj i wyjaśnij polecenia, które użyjesz, aby:

- wyświetlić informacje o pliku/katalogu
- zmienić prawa dostępu do pliku/katalogu
- zmienić właściciela pliku/katalogu

Jeśli nie posiadasz wykonaj migawkę przed ćwiczeniem.

Przed przystąpieniem do ćwiczenia sprawdź czy ustawienie maszyny wirtualnej pozwala na dostęp do Internetu, jeżeli ustawienia są niezgodne wykonaj konfigurację pierwszej karty, a następnie uruchom Ubuntu desktop.

Po uruchomieniu Ubuntu podaj **login**: ubuntu **Password**: ubuntu

W uruchomionym terminalu wpisz sudo -s **Password**: ubuntu

```
ubuntu@ubuntu-VirtualBox:~$ sudo -s
[sudo] hasło użytkownika ubuntu:
root@ubuntu-VirtualBox:/home/ubuntu#
```

Lub jeżeli chcesz pracować na konsoli tekstowej wybierz Ctrl+Alt+F4 aby zalogować się do użytkownika root podaj hasło 1234

```
Ubuntu 22.04.2 LTS ubuntu-VirtualBox tty4

ubuntu-VirtualBox login: root
Password:
```

```
root@ubuntu-VirtualBox:~#
```

Przygotowanie:

Przed ćwiczeniem przywróć pierwszą migawkę.

Wykonaj zadanie na maszynie wirtualnej z Ubuntu desktop 22.04.

Zadanie 1.

a) Sprawdź nazwę swojego katalogu domowego i zapisz ją w zeszycie.

- przejście do konta **ubuntu**

su ubuntu

- przejście do domowego

cd ~

- zapytanie w jakim katalogu jestem

pwd

```
root@ubuntu-VirtualBox:/home/ubuntu# su ubuntu
ubuntu@ubuntu-VirtualBox:~$ cd ~
ubuntu@ubuntu-VirtualBox:~$ pwd
/home/ubuntu
```

b) Zapisz w zeszycie interpretację praw dostępu dla przykładowego ukrytego pliku i ukrytego katalogu z katalogu domowego.

ls -la ~

```
ubuntu@ubuntu-VirtualBox:~$ ls -la ~
razem 664072
drwxr-x--- 16 ubuntu ubuntu      4096 sie 12 14:07 .
drwxr-xr-x  3 root   root        4096 lip 15 12:33 ..
-rw-------  1 ubuntu ubuntu        44 lip 17 19:57 .bash_history
-rw-r--r--  1 ubuntu ubuntu       220 lip 15 12:33 .bash_logout
-rw-r--r--  1 ubuntu ubuntu      3771 lip 15 12:33 .bashrc
drwx------ 10 ubuntu ubuntu      4096 lip 17 15:29 .cache
drwx------ 11 ubuntu ubuntu      4096 lip 15 12:51 .config
```

c) Korzystając z polecenia **echo** utwórz w swoim katalogu plik o nazwie **plik1** i zawartości **aaaaa** a następnie dopisz do tego pliku zawartość **bbbbbbb**. Korzystając z polecenia **echo** dopisz do pliku cccccccc jak poniżej.

echo "aaaaa" > plik1

echo "bbbbbbb" >> plik1

echo "cccccccc" >> plik1

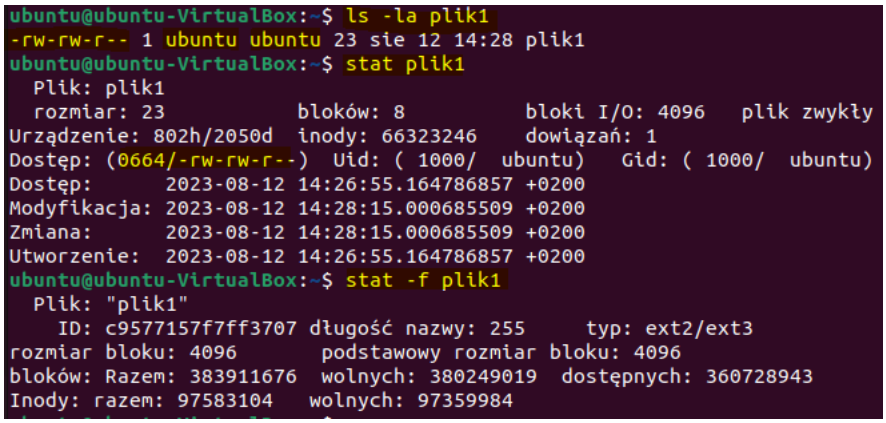
```
ubuntu@ubuntu-VirtualBox:~$ echo "aaaaa" > plik1
ubuntu@ubuntu-VirtualBox:~$ echo "bbbbbbb" >> plik1
ubuntu@ubuntu-VirtualBox:~$ echo "cccccccc" >> plik1
```

d) Wykonaj poniższe polecenia, a następnie opisz ich efekty dotyczące praw dostępu do pliku "plik1". Podaj symboliczną i liczbową reprezentację praw dostępu.

ls -la plik1

stat plik1

stat -f plik1



Na podstawie podanych poleceń i wyników możemy odczytać różne informacje o pliku "plik1". Oto ich opis:

1. *Prawa dostępu:*

Symboliczna reprezentacja: -rw-rw-r--

Liczbowe prawa dostępu: 0664

Opis praw dostępu: Plik jest dostępny do odczytu i zapisu dla właściciela i grupy, a dla innych użytkowników jest dostępny tylko do odczytu.

2. *Informacje o pliku:*

Rozmiar pliku: 23 bajty

Bloków: 8

Typ pliku: plik zwykły

3. *Informacje o urządzeniu i inodach:*

Urządzenie: 802h/2050d

Numer i-node: 66323246

Liczba dowiązań: 1

4. *Daty związane z plikiem:*

Data dostępu: 2023-08-12 14:26:55.164786857 +0200

Data modyfikacji: 2023-08-12 14:28:15.000685509 +0200

Data zmiany: 2023-08-12 14:28:15.000685509 +0200

Data utworzenia: 2023-08-12 14:26:55.164786857 +0200

5. *Informacje o systemie plików:*

Nazwa pliku: "plik1"

Typ systemu plików: ext2/ext3

Rozmiar bloku: 4096 bajtów

Liczba bloków w systemie plików: 383911676

Wolne bloki: 380249019

Dostępne bloki: 360728943

Razem inodów: 97583104

Wolne inody: 97359984

Wyniki tych poleceń dostarczają informacji na temat uprawnień, właściwości i atrybutów pliku "plik1", a także szczegółów związanych z systemem plików na którym dany plik się znajduje, takich jak ilość wolnych bloków i inodów.

e) Utwórz nowe dowiązanie twarde do pliku "plik1" i nazwij je "dtplik1". Wyświetl i wyjaśnij opis tego nowego dowiązania oraz podaj i wyjaśnij prawa dostępu do "dtplik1".

ln plik1 dtplik1

stat dtplik1

```
ubuntu@ubuntu-VirtualBox:~$ ln plik1 dtplik1
ubuntu@ubuntu-VirtualBox:~$ stat dtplik1
  Plik: dtplik1
  rozmiar: 23           bloków: 8          bloki I/O: 4096   plik zwykły
Urządzenie: 802h/2050d  inody: 66323246    dowiązań: 2
Dostęp: (0664/-rw-rw-r--)  Uid: ( 1000/  ubuntu)   Gid: ( 1000/  ubuntu)
Dostęp:      2023-08-12 14:26:55.164786857 +0200
Modyfikacja: 2023-08-12 14:28:15.000685509 +0200
Zmiana:      2023-08-12 14:41:10.956475851 +0200
Utworzenie:  2023-08-12 14:26:55.164786857 +0200
```

Opis nowego dowiązania "dtplik1":

Dowiązanie "dtplik1" jest odnośnikiem do pliku co "plik1". Oba dowiązania wskazują na tę samą inodę, co oznacza, że zmiany wprowadzane w jednym dowiązaniu będą widoczne w drugim.

Prawa dostępu do "dtplik1":

Symboliczna reprezentacja: -rw-rw-r--

Liczbowe prawa dostępu: 0664

Opis praw dostępu:

"Dtplik1" jest dostępny do odczytu i zapisu dla właściciela i grupy, a dla innych użytkowników jest dostępny tylko do odczytu.

Warto zauważyć, że oba dowiązania ("plik1" i "dtplik1") mają takie same prawa dostępu, ponieważ dowiązania odnoszą się do tego samego pliku i korzystają z tych samych praw dostępu.

f) Utwórz dowiązanie symboliczne (miękkie) o nazwie "dmplik1" do pliku "plik1", a następnie wyświetl i wyjaśnij jego opis oraz wyświetl i wyjaśnij prawa dostępu.

ln -s plik1 dmplik1

stat dmplik1

```
ubuntu@ubuntu-VirtualBox:~$ ln -s plik1 dmplik1
ubuntu@ubuntu-VirtualBox:~$ stat dmplik1
  Plik: dmplik1 -> plik1
  rozmiar: 5            bloków: 0          bloki I/O: 4096   dowiązanie symboliczne
Urządzenie: 802h/2050d  inody: 66322799    dowiązań: 1
Dostęp: (0777/lrwxrwxrwx)  Uid: ( 1000/  ubuntu)   Gid: ( 1000/  ubuntu)
Dostęp:      2023-08-12 14:50:57.533622621 +0200
Modyfikacja: 2023-08-12 14:50:57.533622621 +0200
Zmiana:      2023-08-12 14:50:57.533622621 +0200
Utworzenie:  2023-08-12 14:50:57.533622621 +0200
```

Wyjaśnienie opisu oraz praw dostępu do dowiązania symbolicznego "dmplik1":

Opis dowiązania symbolicznego "dmplik1":

Rozmiar: 5 bajtów

Bloków: 0

Bloki I/O: 4096 bajtów

Typ: Dowiązanie symboliczne

Urządzenie i informacje o inodach:

Urządzenie: 802h/2050d

Numer i-node: 66322799

Liczba dowiązań: 1

Prawa dostępu do dowiązania symbolicznego "dmplik1":

Symboliczna reprezentacja: lrwxrwxrwx

Liczbowe prawa dostępu: 0777

Opis praw dostępu: Dowiązanie symboliczne posiada pełne prawa dostępu dla właściciela, grupy i innych użytkowników. Pozwala to na czytanie, zapisywanie oraz wykonywanie operacji na dowiązaniu.

Daty związane z dowiązaniem symbolicznym:

Data dostępu: 2023-08-12 14:50:57.533622621 +0200

Data modyfikacji: 2023-08-12 14:50:57.533622621 +0200

Data zmiany: 2023-08-12 14:50:57.533622621 +0200

Data utworzenia: 2023-08-12 14:50:57.533622621 +0200

Podsumowując, dowiązanie symboliczne "dmplik1" zostało utworzone jako odnośnik do pliku "plik1". Posiada ono pełne prawa dostępu, co oznacza, że użytkownicy będą mogli dowolnie czytać, zapisywać i wykonywać operacje na tym dowiązaniu symbolicznym.

g) Sprawdź szczegółowe informacje o plikach "dtplik1" i "dmplik1" przez polecenie:

ls -la dtplik1 dmplik1

```
ubuntu@ubuntu-VirtualBox:~$ ls -la dtplik1 dmplik1
lrwxrwxrwx 1 ubuntu ubuntu  5 sie 12 14:50 dmplik1 -> plik1
-rw-rw-r-- 2 ubuntu ubuntu 23 sie 12 14:28 dtplik1
```

Wyjaśni szczegółowo każdą kolumnę:

Prawa dostępu:

"lrwxrwxrwx" dla "dmplik1" oznacza, że to dowiązanie symboliczne ma pełne prawa dostępu (czytanie, zapisywanie i wykonywanie) dla właściciela, grupy i innych użytkowników.

"-rw-rw-r--" dla "dtplik1" oznacza, że to zwykły plik ma prawa dostępu do odczytu i zapisu dla właściciela i grupy, a dla innych użytkowników ma tylko prawa do odczytu.

Liczba dowiązań:

"1" dla "dmplik1", ponieważ to jest jedno dowiązanie symboliczne.

"2" dla "dtplik1", ponieważ istnieją dwa dowiązania do tego samego pliku (dwie inody).

Właściciel i grupa:

"ubuntu ubuntu" dla obu plików, co oznacza, że właścicielem i grupą jest użytkownik "ubuntu".

Rozmiar w bajtach:

"5" dla "dmplik1", co jest rozmiarem dowiązania symbolicznego.

"23" dla "dtplik1", co jest rozmiarem zwykłego pliku.

Data i godzina ostatniej modyfikacji:

"sie 12 14:50" dla "dmplik1", co oznacza, że dowiązanie symboliczne zostało zmodyfikowane 12 sierpnia o godzinie 14:50.

"sie 12 14:28" dla "dtplik1", co oznacza, że zwykły plik został zmodyfikowany 12 sierpnia o godzinie 14:28.

Nazwa pliku lub dowiązania:

"dmplik1" to nazwa dowiązania symbolicznego, a "-> plik1" wskazuje na nazwę pliku, do którego dowiązanie prowadzi.

"dtplik1" to nazwa zwykłego pliku.

Podsumowując, wynik polecenia ls -la wyświetla szczegółowe informacje o dwóch plikach: dowiązaniu symbolicznym "dmplik1" i zwykłym pliku "dtplik1", w tym ich prawa dostępu, właściciela, rozmiar, datę modyfikacji i więcej.

h) Zapisz w zeszycie jaka jest różnica między dowiązaniem twardym a symbolicznym w kontekście praw dostępu do pliku.

Różnica między dowiązaniem twardym a dowiązaniem symbolicznym w kontekście praw dostępu do pliku polega na tym, jak są one interpretowane przez system operacyjny i jak wpływają na dostęp do pliku docelowego. Charakteryzacja obu typów dowiązań w kontekście praw dostępu:

1. Dowiązanie Twarde (Hard Link):

Dowiązanie twarde to dodatkowy wskaźnik na istniejący plik w systemie plików.

Wszystkie dowiązania twarde do tego samego pliku mają identyczny numer i-node, co oznacza, że nie można odróżnić oryginalnego pliku od dowiązań twardych.

Dowiązanie twarde działa w taki sposób, jakby to była ta sama zawartość, co oryginalny plik, a zmiany w jednym dowiązaniu odzwierciedlają się w pozostałych.

Prawa dostępu do dowiązań twardych nie mają znaczenia, ponieważ wszystkie dowiązania odnoszą się do tego samego numeru i-node. Prawa dostępu do pliku są w pełni kontrolowane przez właściciela pliku i prawa dostępu do innych dowiązań.

2. Dowiązanie Symboliczne (Symbolic Link):

Dowiązanie symboliczne to osobny plik, który przechowuje ścieżkę do pliku docelowego.

Dowiązanie symboliczne może wskazywać na plik w innym systemie plików lub nawet na nieistniejący plik. Prawa dostępu do dowiązania symbolicznego są niezależne od pliku docelowego. Dowiązanie symboliczne posiada swoje własne prawa dostępu, które określają, kto może czytać, zapisywać lub wykonywać to dowiązanie.

Zmiany w prawach dostępu do dowiązania symbolicznego nie mają wpływu na prawa dostępu do pliku docelowego.

Podsumowując, dowiązanie twarde odnosi się bezpośrednio do oryginalnego pliku, a prawa dostępu są identyczne dla wszystkich dowiązań. W przypadku dowiązań symbolicznych, prawa dostępu do dowiązania i pliku docelowego są niezależne, co pozwala na większą elastyczność w zarządzaniu dostępem do plików.

i) Wykonaj duplikat pliku o nazwie "plik1" w tej samej lokalizacji, nadając mu nazwę "plik2", a następnie zapisz opis (tylko obejmujące informacje do linii jak na zrzucie) właśnie utworzonego pliku.

cp plik1 plik2

stat plik2

```
ubuntu@ubuntu-VirtualBox:~$ cp plik1 plik2
ubuntu@ubuntu-VirtualBox:~$ stat plik2
  Plik: plik2
  rozmiar: 23           bloków: 8          bloki I/O: 4096   plik zwykły
Urządzenie: 802h/2050d  inody: 66322788    dowiązań: 1
Dostęp: (0664/-rw-rw-r--)  Uid: ( 1000/  ubuntu)   Gid: ( 1000/  ubuntu)
```

Opisu pliku "plik2":

1. *Informacje o pliku "plik2":*

Rozmiar pliku: 23 bajty

Bloków: 8

Typ pliku: plik zwykły

2. *Informacje o urządzeniu i inodach pliku "plik2":*

Urządzenie: 802h/2050d

Numer i-node: 66322788

Liczba dowiązań: 1

3. *Prawa dostępu do pliku "plik2":*

Symboliczna reprezentacja: -rw-rw-r--

Liczbowe prawa dostępu: 0664

Opis praw dostępu: Plik jest dostępny do odczytu i zapisu dla właściciela i grupy, a dla innych użytkowników jest dostępny tylko do odczytu.

j) Zapisz w zeszycie jaka jest różnica między plikiem plik1 a plik2 w kontekście praw dostępu do pliku.

W kontekście praw dostępu do pliku "plik1" i "plik2" istnieje żadna różnica, ponieważ zarówno "plik1", jak i "plik2" mają takie same prawa dostępu. Oba pliki posiadają następujące prawa dostępu: -rw-rw-r--

lub numerycznie 0664. Oznacza to, że właściciel i grupa pliku mają uprawnienia do odczytu i zapisu, podczas gdy inni użytkownicy mają tylko uprawnienia do odczytu.

k) Wykonaj kolejne czynności: przejdź do katalogu domowego użytkownika, utwórz nowy katalog o nazwie "katalog", wyświetl szczegółową listę zawartości katalogu "katalog", w tym ukryte pliki (opcja -a oraz dodatkowe szczegóły (opcja -l), wyświetl statystyki katalogu "katalog" za pomocą polecenia "stat".

cd ~

mkdir katalog

ls -la katalog

stat katalog

```
ubuntu@ubuntu-VirtualBox:~$ cd ~
ubuntu@ubuntu-VirtualBox:~$ mkdir katalog
ubuntu@ubuntu-VirtualBox:~$ ls -la katalog
razem 8
drwxrwxr-x  2 ubuntu ubuntu 4096 sie 12 15:29 .
drwxr-x--- 17 ubuntu ubuntu 4096 sie 12 15:29 ..
ubuntu@ubuntu-VirtualBox:~$ stat katalog
  Plik: katalog
  rozmiar: 4096          bloków: 8          bloki I/O: 4096   katalog
Urządzenie: 802h/2050d  inody: 66323021    dowiązań: 2
Dostęp: (0775/drwxrwxr-x)  Uid: ( 1000/  ubuntu)   Gid: ( 1000/  ubuntu)
```

l) Wykonaj kolejne czynności: przenieś plik "dmplik1" do katalogu "katalog" i nadaj mu nazwę "plik1", wyświetl zawartość katalogu "katalog" w celu potwierdzenia czy plik "plik1" faktycznie się tam znajduje, wyświetl za pomocą polecenia "stat" statystyki dowiązania symbolicznego do "plik1" w katalogu "katalog".

mv dmplik1 ~/katalog/plik1

ls ~/katalog/plik1

stat ~/katalog/plik1

```
ubuntu@ubuntu-VirtualBox:~$ mv dmplik1 ~/katalog/plik1
ubuntu@ubuntu-VirtualBox:~$ ls ~/katalog/plik1
ls: nie ma dostępu do '/home/ubuntu/katalog/plik1': Za duże zagnieżdżenie dowiązań symbolicznych
ubuntu@ubuntu-VirtualBox:~$ stat ~/katalog/plik1
  Plik: /home/ubuntu/katalog/plik1 -> plik1
  rozmiar: 5             bloków: 0          bloki I/O: 4096   dowiązanie symboliczne
Urządzenie: 802h/2050d  inody: 66322799    dowiązań: 1
Dostęp: (0777/lrwxrwxrwx)  Uid: ( 1000/  ubuntu)   Gid: ( 1000/  ubuntu)
```

m) Wykonaj kolejne czynności: przenieś plik o nazwie "dtplik1" do katalogu "plik2" znajdującego się w ścieżce "/home/ubuntu/katalog", wyświetl szczegółową listę zawartości katalogu "katalog" w katalogu domowym użytkownika (opcja -a), wyświetl za pomocą polecenia "stat" statystyki pliku "plik2" znajdującego się w katalogu "katalog".

mv dtplik1 ~/katalog/plik2

ls -la ~/katalog

stat ~/katalog/plik2

```
ubuntu@ubuntu-VirtualBox:~$ mv dtplik1 ~/katalog/plik2
ubuntu@ubuntu-VirtualBox:~$ ls -la ~/katalog
razem 12
drwxrwxr-x  2 ubuntu ubuntu 4096 sie 12 15:46 .
drwxr-x--- 17 ubuntu ubuntu 4096 sie 12 15:46 ..
lrwxrwxrwx  1 ubuntu ubuntu    5 sie 12 14:50 plik1 -> plik1
-rw-rw-r--  2 ubuntu ubuntu   23 sie 12 14:28 plik2
ubuntu@ubuntu-VirtualBox:~$ stat ~/katalog/plik2
  Plik: /home/ubuntu/katalog/plik2
  rozmiar: 23           bloków: 8          bloki I/O: 4096   plik zwykły
Urządzenie: 802h/2050d  inody: 66323246    dowiązań: 2
Dostęp: (0664/-rw-rw-r--)  Uid: ( 1000/  ubuntu)   Gid: ( 1000/  ubuntu)
```

n) Usuń katalogu "katalog" w katalogu domowym użytkownika wraz z jego zawartością.

rm -r ~/katalog

To polecenie wykonuje następujące czynności:

rm: To polecenie służy do usuwania plików i katalogów.

-r: Ta opcja oznacza rekursywne usuwanie, co pozwala usunąć katalog wraz z jego zawartością.

~/katalog: Jest to ścieżka do katalogu "katalog" w katalogu domowym użytkownika.

```
ubuntu@ubuntu-VirtualBox:~$ rm -r ~/katalog
```

o) Potwierdź, że nie ma już tego katalogu w podanej lokalizacji:

ls -la ~/katalog

```
ubuntu@ubuntu-VirtualBox:~$ ls -la ~/katalog
ls: nie ma dostępu do '/home/ubuntu/katalog': Nie ma takiego pliku ani katalogu
```

(zgłoszenie) 1

Zadanie 2.

Przeczytaj polecenia i wykonaj poniższe czynności (rozpocznij będąc zalogowanym do konta ubuntu):

a) Rozpocznij sesję superużytkownika (root) za pomocą polecenia

sudo -s

b) Użyj polecenia useradd u1 w celu dodania nowego użytkownika o nazwie "u1".

useradd u1

```
root@ubuntu-VirtualBox:/home/ubuntu# useradd u1
```

c) Utwórz pusty pliku o nazwie "Mojplik" w bieżącym katalogu.

tuch Mojplik

```
root@ubuntu-VirtualBox:/home/ubuntu# touch Mojplik
```

d) Jako użytkownik **root** użyj polecenia w celu zmiany właściciela i grupy pliku "Mojplik" na "u1.users", następnie użyj polecenia ls -la Mojplik, aby wyświetlić szczegóły pliku "Mojplik".

Składnia polecenia: **chown nowy_user.nowa_grupa plik**

chown u1.users Mojplik

```
root@ubuntu-VirtualBox:/home/ubuntu# chown u1.users Mojplik
root@ubuntu-VirtualBox:/home/ubuntu# ls -la Mojplik
-rw-r--r-- 1 u1 users 0 sie 12 16:15 Mojplik
```

Użytkownik "u1" jest właścicielem pliku, a grupa "users" jest jego grupą.

e) Zmień właściciela pliku "Mojplik" na "ubuntu" nie zmieniając grupy.

Składnia polecenia: **chown nowy_user plik**

chown ubuntu Mojplik

```
root@ubuntu-VirtualBox:/home/ubuntu# chown ubuntu Mojplik
root@ubuntu-VirtualBox:/home/ubuntu# ls -la Mojplik
-rw-r--r-- 1 ubuntu users 0 sie 12 16:15 Mojplik
```

Właścicielem pliku jest użytkownik "ubuntu", a grupa "users" pozostała niezmieniona.

f) Zmień grupę nie zmieniając użytkownika. Składnia polecenia: **chown .nowa_grupa plik**

chown .ubuntu Mojplik

```
root@ubuntu-VirtualBox:/home/ubuntu# chown .ubuntu Mojplik
root@ubuntu-VirtualBox:/home/ubuntu# ls -la Mojplik
-rw-r--r-- 1 ubuntu ubuntu 0 sie 12 16:15 Mojplik
```

g) Jako użytkownik root zmień przypisanie grupy poleceniem chgrp.

Składnia polecenia: **chgrp nowa_grupa plik**

chgrp users Mojplik

```
root@ubuntu-VirtualBox:/home/ubuntu# chgrp users Mojplik
root@ubuntu-VirtualBox:/home/ubuntu# ls -la Mojplik
-rw-r--r-- 1 ubuntu users 0 sie 12 16:15 Mojplik
```

h) Jako zwykły użytkownik użyj polecenia chown do przydzielenia swojego pliku nowej grupie.

Składnia polecenia: **chown .nowa_grupa plik**

su ubuntu

chown .ubuntu Mojplik

```
root@ubuntu-VirtualBox:/home/ubuntu# su ubuntu
ubuntu@ubuntu-VirtualBox:~$ chown .ubuntu Mojplik
ubuntu@ubuntu-VirtualBox:~$ ls -la Mojplik
-rw-r--r-- 1 ubuntu ubuntu 0 sie 12 16:15 Mojplik
```

Możesz zmienić przydzielenie pliku do grupy tylko w przypadku, gdy jesteś członkiem tej grupy.

i) Najważniejszą opcją dla poleceń chown oraz chgrp jest opcja -R, pozwalająca na zmianę właściciela pliku oraz przypisanej grupy we wszystkich plikach danego katalogu. Wykonaj jak poniżej i wyjaśnij efekt w zeszycie.

chown ubuntu.ubuntu -R ~

```
ubuntu@ubuntu-VirtualBox:~$ chown ubuntu.ubuntu -R ~
chown: zmiana właściciela '/home/ubuntu/google-chrome-stable_current_amd64.deb': Operacja niedozwolona
chown: zmiana właściciela '/home/ubuntu/firefox_2987.assert': Operacja niedozwolona
chown: zmiana właściciela '/home/ubuntu/firefox_2987.snap': Operacja niedozwolona
chown: zmiana właściciela '/home/ubuntu/vlc_3078.snap': Operacja niedozwolona
chown: zmiana właściciela '/home/ubuntu/vlc_3078.assert': Operacja niedozwolona
ubuntu@ubuntu-VirtualBox:~$ ls -la ~
razem 664080
drwxr-x--- 16 ubuntu ubuntu      4096 sie 12 16:15 .
drwxr-xr-x  3 root   root        4096 lip 15 12:33 ..
-rw-------  1 ubuntu ubuntu       512 sie 12 16:04 .bash_history
```

Powyższe polecenie próbowało zmienić właściciela na "ubuntu" i grupę na "ubuntu" dla wszystkich plików i katalogów w katalogu domowym. Pliki (w formacie .deb, .snap i .assert) są zabezpieczone przed zmianami właściciela przez zwykłego użytkownika. Pamiętaj jednak, że zmienianie właściciela i grupy systemowych plików może prowadzić do problemów z działaniem systemu lub oprogramowania, dlatego zaleca się ostrożność w dokonywaniu takich zmian.

Jeśli potrzebowałbyś zmienić właściciela i grupę dla tych plików, możesz użyć polecenia sudo przed poleceniem chown, aby uzyskać uprawnienia superużytkownika. W przypadku sytemu produkcyjnego u klienta nie zalecam takiego rozwiązania, chyba że wynika to z jakiejś wyższej konieczności i „zabezpieczyłeś sobie tyły”.

j) Zmień właściciela i grupę dla plików we wszystkich plikach danego katalogu które są zabezpieczone przed zmianami właściciela przez zwykłego użytkownika.

sudo chown ubuntu.ubuntu -R ~

podaj hasło **ubuntu**

```
ubuntu@ubuntu-VirtualBox:~$ sudo chown ubuntu.ubuntu -R ~
[sudo] hasło użytkownika ubuntu:
ubuntu@ubuntu-VirtualBox:~$ ls -la ~
razem 664080
drwxr-x--- 16 ubuntu ubuntu      4096 sie 12 16:15 .
drwxr-xr-x  3 root   root        4096 lip 15 12:33 ..
-rw-------  1 ubuntu ubuntu       512 sie 12 16:04 .bash_history
```

(zgłoszenie) 2

Zadanie 3

Przeczytaj polecenia i wykonaj poniższe czynności (rozpocznij będąc zalogowanym do konta ubuntu):
Przygotowanie. Wklej w terminal poniższe polecenia:

sudo touch test

sudo chown ubuntu.ubuntu test

Zapisz w zeszycie jakich poleceń użyłeś do wykonaniu poniższych czynności.

Przeczytaj polecenia i wykonaj poniższe czynności (rozpocznij będąc zalogowanym do konta ubuntu):

a) Odbierz sobie (tzn. właścicielowi pliku) za pomocą kodu numerycznego prawo pisania do pliku **test**
b) Sprawdź nowe prawa do pliku **test** (ls -l test).
c) Spróbuj usunąć plik **test** (rm test).

(zgłoszenie) 3

Materiał pomocniczy.

a) Aby odebrać sobie prawo do pisania do pliku "test" za pomocą kodu numerycznego, musisz zmienić odpowiednie cyfry w kodzie numerycznym tak, aby oznaczały brak prawa do zapisu, ale pozostawiły prawa do odczytu i wykonania.

b) Aby sprawdzić nowe prawa dostępu do pliku "test", użyj polecenia ls -l razem z nazwą pliku. Wyświetlą się szczegóły, które pomogą Ci zrozumieć, jakie prawa dostępu zostały zmienione.

c) Aby spróbować usunąć plik "test", upewnij się, że masz odpowiednie prawa do katalogu, w którym plik się znajduje. Jeśli tak, pamiętaj, że polecenie rm usuwa plik bez pytania o potwierdzenie. Jeśli masz problemy z usunięciem, zwróć uwagę na prawa dostępu do pliku i katalogu nadrzędnego.

Odp:
a) chmod 444 test
b) ls -la |grep test
c) rm test y

Zadanie 4

Przeczytaj polecenia i wykonaj poniższe czynności (rozpocznij będąc zalogowanym do konta ubuntu):
Przygotowanie. Wklej w terminal poniższe polecenia:

sudo mkdir -p ~/Info/Sk ~Info/OP

sudo touch ~Info/OP/testop

sudo touch ~/Info/Sk/testsk ~/Info/Sk/testsk1

sudo mkdir -p ~/Info/Sk/sk2

sudo chown ubuntu.ubuntu -R ~Info/OP

sudo chown ubuntu.ubuntu -R ~/Info/Sk/testsk1

a) Sprawdź prawa dostępu do katalogu **~/Info/Sk**

ls -la ~/Info/Sk

b) Sprawdź prawa dostępu do katalogu **~Info/OP**

ls -la ~Info/OP

c) Odbierz właścicielowi katalogu **~/Info/Sk** prawa do czytania

sudo chmod u-r ~/Info/Sk

d) Spróbuj wykonać polecenia jako użytkownik ubuntu a następnie root:

1. przeczytania zawartości katalogu, czyli **ls -la ~Info/OP**
2. usunięcia plików z tego katalogu, czyli **rm ~/Info/Sk/***
3. usunięcie katalogu ~/Info/Sk/ **rmdir ~/Info/Sk**

e) Po wykonaniu tych kroków, zanotuj wyniki i efekty w swoim zeszycie. Porównaj, jak zmieniają się prawa dostępu i dostępność operacji, gdy zmieniasz właściciela, grupę oraz prawa dostępu do katalogów i plików.

Sposób, w jaki możesz zebrać i porównać wyniki oraz efekty w swoim zeszycie:

1. Opis początkowego stanu: Rozpocznij od opisu początkowego stanu, w którym wszystkie katalogi i pliki zostały utworzone z odpowiednimi właścicielami, grupami oraz prawami dostępu.
2. Polecenia i ich wyniki: Dla każdego kroku, który został wykonany, zanotuj użyte polecenie oraz wynikowe prawa dostępu i strukturę katalogów/plików.
3. Zmiany w prawach dostępu: Porównaj, jak zmieniają się prawa dostępu po każdym zastosowanym poleceniu. Czy widzisz różnice w prawach dostępu pomiędzy katalogami i plikami? Czy możesz zidentyfikować, które prawa zostały dodane, a które odebrane?
4. Dostępność operacji: Zanotuj, jakie operacje były możliwe do wykonania dla różnych katalogów i plików. Czy mogłeś wyświetlić zawartość katalogu? Czy mogłeś usunąć pliki? Czy były jakieś operacje, które stały się niemożliwe ze względu na zmiany w prawach dostępu?
5. Wnioski i refleksje: Na podstawie swoich obserwacji, wniosków i refleksji, zanotuj, jak zmiany właściciela, grupy i praw dostępu wpływają na dostępność i operacje na katalogach i plikach. Co się zmienia, gdy zostaną zastosowane różne kombinacje właściciela, grupy i praw dostępu?

Przygotuj tę analizę w formie tekstowej lub tabelarycznej w swoim zeszycie, aby mieć jasny obraz zmian w prawach dostępu i ich wpływie na operacje w systemie plików.

(zgłoszenie) 4

Materiał pomocniczy. Odebranie sobie prawa pisania do plików jest formą zabezpieczenia tych plików przed przypadkowym usunięciem. System zapyta wtedy czy rzeczywiście usunąć plik i jeśli odpowiemy „yes" - to go usunie. Plik nie zostałby usunięty, jeśli użytkownik nie miałby prawa do zapisu do katalogu,

w którym usuwany plik się znajduje. Można użyć polecenia **chmod** do zmiany uprawnień do pliku zarówno użytkownik root, jak i właściciel pliku.

chmod u+x test Nadanie właścicielowi pliku **test** prawa wykonywania

chmod g+rwx test Nadanie członkom grupy wszystkich praw do pliku **test**

chmod ug-x test Odebranie praw wykonywania dla właściciela i członków grupy

chmod g-w,o+r test Odebranie członkom grupy prawa pisania, oraz przyznanie pozostałym użytkownikom prawa czytania do pliku **test**

chmod g=wx test Nadanie członkom grupy prawa pisania i wykonywania.
Jeśli grupa tych użytkowników miała prawo czytania (r) - to prawo to zostanie odebrane.
Operator = przyznaje tylko te prawa, które zostały podane w poleceniu.

Zadanie 5

W zeszycie odpowiedz na pytania i **sprawdź** odpowiedzi w praktyce i przedstaw np. wycinki jako dowody.

Jeżeli plik lub katalog lub użytkownik nie istnieje to należy go założyć jako użytkownika root.

a) Czy możesz skopiować swój plik do katalogu innego użytkownika, który pracuje na tej samej stacji na innym terminalu?
b) Czy możesz skopiować plik z katalogu innego użytkownika?
c) Co zrobić, aby nikt nie mógł kopiować twoich plików?
d) Jakie prawa uzyska plik **test** o dotychczasowych prawach rw- po wydaniu polecenia **chmod a+r test**

Odp.: a) nie, b) tak, c) odebrać prawa czytania (r) do plików dla członków grupy i pozostałych użytkowników, d) rw-r---r---

(zgłoszenie) 5

Zadanie 6 Wykonaj czynności

Jeżeli plik lub katalog nie istnieje to należy go założyć jako użytkownik.

a) Utwórz katalog **jar**

mkdir jar

b) Do katalogu **jar** skopiuj plik **test** (lub inny)

touch test

cp test jar/

c) Odbierz sobie prawo do pisania do pliku **test**

chmod -w jar/test

d) Usuń plik **test**

rm -i jar/test

UWAGA: Po wydaniu polecenia usuwania pliku (rm), wyświetlony zostanie komunikat „Override protection 444 (yes/no)?". Jeśli odpowiemy „yes", plik zostanie usunięty.

e) Jeszcze raz skopiuj do katalogu **jar** plik **test** i odbierz mu prawa do pisania

cp test jar/

chmod -w jar/test

f) Usuń prawo do pisania do katalogu **jar**

chmod -w jar

g) Spróbuj usunąć plik **test**

rm -i jar/test

h) Usuń plik **test**

sudo rm -i jar/test

i) W zeszycie zanotuj, jakie były efekty każdego kroku. Porównaj, w jakim stopniu zmiany praw dostępu wpłynęły na możliwość kopiowania i usuwania plików w katalogu "jar". Porównaj, jak różnice w prawach dostępu do pliku a także do samego katalogu wpłynęły na możliwości zmiany i usuwania plików.
To ćwiczenie pozwoli zrozumieć, jak prawa dostępu działają w kontekście plików i katalogów oraz jakie ograniczenia one nakładają.

(zgłoszenie) 6

Zadanie 7

Jeżeli plik lub katalog lub użytkownik nie istnieje to należy go założyć jako użytkownika root.

Wyjaśnij w zeszycie otrzymany rezultat.

a) Utwórz użytkownika **lolek** z hasłem 1234

sudo adduser lolek

b) Zaloguj się na tej samej stacji na inny terminal > Wybierz **Ctrl**+**Alt**+**F5** podaj **login**: lolek **Password**: 1234)

c) Wybierz **Ctrl**+**Alt**+**F3** wrócisz do konta ubuntu, utwórz katalog **xkat** w katalogu domowym użytkownika lolek.

Katalog nie zostanie utworzony, ponieważ nie mamy odpowiednich praw do wykonania tej czynności, mimo to wykonaj to zadanie zgodnie z swoją wiedzą, zdecyduj co zrobisz i wykonaj.

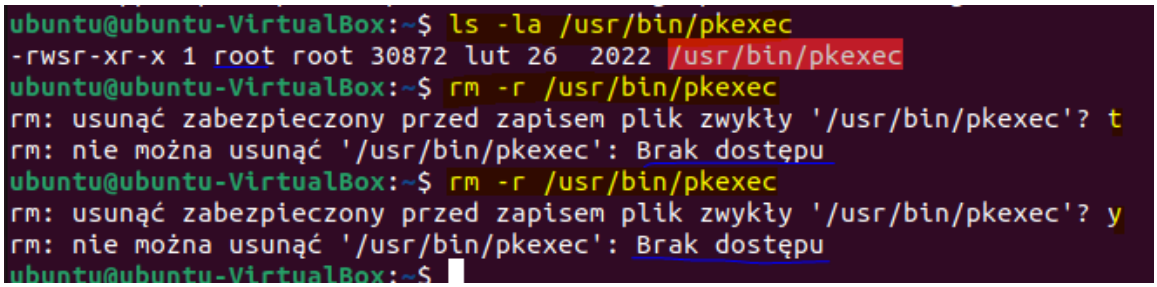
Podpowiedz:

Skoro w poprzednim kroku zalogowaliśmy się jako użytkownik "lolek", teraz powinniśmy mieć dostęp do swojego katalogu domowego. Utwórzmy katalog "xkat": *mkdir ~/xkat*

W tym przypadku, jeśli katalog "xkat" nie zostanie utworzony, możemy próbować stworzyć go, ponieważ jesteśmy aktualnie zalogowani jako właściciel tego konta.

Wyjaśnij w zeszycie otrzymany rezultat. Poproś prowadzącego o sprawdzenie.

d) Usuń plik **/usr/bin/pkexec**



e) Wyjaśnij w zeszycie otrzymany rezultat użyj polecenia: **ls** -l **/ usr/bin/pkexec**

Wynik, który otrzymałeś, wskazuje na to, że plik /usr/bin/pkexec ma ustawiony bit SUID (Set User ID) oraz bit SUID (Set Group ID), co sprawia, że jest to plik "zabezpieczony przed zapisem". Bit SUID pozwala uruchamiać plik z uprawnieniami właściciela pliku, a bit SGID pozwala uruchamiać plik z uprawnieniami grupy pliku.

Ponieważ ten plik ma te bity ustawione i jest on ważnym komponentem systemu, nie możesz go po prostu usunąć. To jest zabezpieczenie, aby zapobiec przypadkowemu lub niepożądanemu usunięciu ważnych plików systemowych.

Aby usunąć ten plik, prawdopodobnie będziesz musiał wykonać to z uprawnieniami superużytkownika, używając sudo przed poleceniem rm. Jednak usuwanie ważnych plików systemowych może mieć poważne konsekwencje dla funkcjonalności systemu, więc zaleca się zachowanie ostrożności i pewności, że wiesz, co robisz.

(zgłoszenie) 7

Zadanie 8

Zapisz w zeszycie co się stało po wykonaniu poniższych czynności.

a) W katalogu macierzystym użytkownika **ubuntu** utwórz pliki **plik1 plik2**

touch plik1 plik2 test

b) W katalogu macierzystym użytkownika **ubuntu** utwórz katalog **bagaz**

mkdir ~/bagaz

c) Sprawdź prawa dostępu do tego katalogu

ls -ld ~/bagaz

d) Usuń prawa do wykonywania (x) dla wszystkich użytkowników (w tym dla właściciel) do katalogu "bagaz"

chmod a-x ~/bagaz

e) Skopiuj do katalogu **bagaz** dwa pliki

sudo cp plik1 plik2 ~/bagaz

f) Wejdź do katalogu **bagaz w trybie superużytkownika**

sudo -i

cd ~/bagaz

g) ustawia swój umask na 0002, usuwa poprzedni plik testowy i tworzy nowy plik testowy i katalog powtórz powyższe czynności z ustawioną maską.

umask 0002

rm test

touch test

mkdir nowy_katalog

Uwaga: aby skopiować plik do katalogu, użytkownik musisz mieć prawa pisania i wykonywania do tego katalogu (wx).

W zeszycie zanotuj efekty i wyniki po wykonaniu powyższych kroków:

1. Przeanalizuj prawa dostępu do katalogu "bagaz" po każdym kroku.
2. Zauważ, jakie zmiany w prawach dostępu i umasce wpływają na możliwość tworzenia, usuwania i modyfikowania plików i katalogów.
3. Porównaj różnice między pierwszym wykonaniem tych kroków (bez ustawionej maski) a drugim wykonaniem (z ustawioną maską).

To ćwiczenie pozwoli Ci na praktyczne zrozumienie, jak prawa dostępu, umask i inne operacje wpływają na zarządzanie plikami i katalogami w systemie Linux.

(zgłoszenie) 8

Zadanie 9

Jeżeli plik lub katalog lub grupa lub użytkownik nie istnieje to należy go założyć jako użytkownika root.

W zeszycie zapisz polecenia, którego użyjesz oraz co się stało po wykonaniu czynności realizujących poniższe zadanie.

a) Zmień grupę przypisaną do utworzonego przez Ciebie katalogu **Nauka** i jego zawartości na utworzoną przez Ciebie grupę **Naukowcy**.

1. Utwórz katalogu Nauka w katalogu domowym

mkdir ~/Nauka

2. Utwórz grupę "Naukowcy" za pomocą polecenia addgroup:

sudo addgroup --force-badname Naukowcy

W konfiguracji systemu istnieje pewne ograniczenie dotyczące akceptowanych nazw grup lub użytkowników. W tym przypadku, błąd wskazuje na to, że nazwa "Naukowcy" nie pasuje do wyrażenia regularnego skonfigurowanego w systemie, dlatego *--force-badname*.

3. Zmień grupę katalogu "Nauka" i jego zawartości na grupę "Naukowcy":

sudo chown :Naukowcy -R ~/Nauka

b) Załóż katalog k2 przypisując mu uprawnienia drwx------ (700), bez korzystania z polecenia chmod.

1. Utworzenie katalogu "k2":

sudo mkdir k2

2. Przypisanie właściciela i grupy do katalogu "k2":

sudo chown root:root k2

3. Przydzielenie uprawnień drwx------ do katalogu "k2" (użytkownik ma prawo do czytania, pisania i wykonania, inni nie mają żadnych praw):

sudo chmod 700 k2

Zapisz te polecenia oraz efekty i wyniki po wykonaniu powyższych kroków w zeszycie:

Zanotuj, jakie polecenia zostały użyte do utworzenia grupy, zmiany grupy dla katalogu "Nauka" oraz tworzenia i ustawienia uprawnień dla katalogu "k2".

Obserwuj, jak zmiany grupy i uprawnień wpływają na dostęp do katalogów i plików.

Porównaj różnice przed i po wykonaniu tych kroków.

(zgłoszenie) 9

Zadanie 10

W zeszycie zapisz polecenie, którego użyjesz oraz co się stało po wykonaniu czynności realizujących poniższe zadanie.

Wyszukanie wszystkich plików o uprawnieniach rwsr-xr-x w systemie Linux można zrealizować za pomocą polecenia find. Upewnij się, że jesteś zalogowany jako użytkownik z odpowiednimi uprawnieniami (np. administrator) lub użyj polecenia z sudo.

Oto jak to zrobić:

sudo find / -type f -perm -4000 -perm /111

W skrócie:

sudo: Uruchamiamy polecenie z uprawnieniami superużytkownika.

find /: Rozpoczynamy wyszukiwanie od głównego katalogu (/).

-type f: Ograniczamy wyniki do plików (nie katalogów).

-perm -4000: Szukamy plików, które mają ustawiony bit SUID (4 w numerze uprawnień).

-perm /111: Szukamy plików, które mają ustawione bit uprawnień wykonania (x) dla użytkownika, grupy i innych (111 w numerze uprawnień).

To polecenie znajdzie wszystkie pliki w systemie, które mają uprawnienia rwsr-xr-x. Zauważ jednak, że wyszukiwanie w całym systemie może zająć trochę czasu i spowodować wyświetlenie dużej liczby wyników.

(zgłoszenie) 10

Po sprawdzeniu przez prowadzącego, jeżeli nie będziesz wykonywał cw_Listy_kontroli_dostepu_acl.pdf przywróć migawkę z przed wykonania zadania.

Podsumowanie:
Po wykonaniu wszystkich czynności z powyższej instrukcji przeczytaj ponownie z zrozumieniem cel ogólny i cele szczegółowe, które znajdują się na pierwszej stronie instrukcji. Jeżeli one zostały niezrealizowane to powtarzaj wykonie tej instrukcji w szkole lub/i w domu do momentu zrealizowania.