

Zarządzanie systemem Linux

Cały dział znajduje się w książce od strony 174 do 182. Zagadnień jest dużo:

- Zarządzanie plikami i folderami,
- Wyszukiwanie plików,
- Dowiązania do plików,
- Błędy związane z dowiązaniem,
- Zbiory ukryte,
- Ścieżki dostępu,
- Uprawnienia do plików i katalogów,
- Właściciele zbiorów.

Link do programu, który ułatwi Wam zarządzanie plikami i katalogami:

<https://midnight-commander.org/>

Możecie oczywiście korzystać z innych.

A tutaj link do forum (co zrobić, gdy nie możemy zainstalować mc):

<https://askubuntu.com/questions/1071392/how-can-i-install-midnight-commander-on-ubuntu-18-04-1>

Dowiązania do plików- <https://pasja-informatyki.pl/sieci-komputerowe/linux-ubuntu-server-linki-dowiazania/>

<https://websetnet.net/pl/how-to-find-broken-symlinks-on-linux-ubuntu-and-delete-them/>

Zbiory ukryte:

<https://serwerweb.pl/2019/08/21/jak-pokazac-ukryte-pliki-i-foldery-w-ubuntu/>

Uprawnienia:

<https://pasja-informatyki.pl/sieci-komputerowe/ubuntu-server-uprawnienia-plikow-folderow/>

Oczywiście to co Wam pokazuję jest uzupełnieniem materiałów w książce.

Na koniec, w ramach relaksu- <https://www.spidersweb.pl/2020/04/windows-10-pliki-linuxa.html>

ZAGADNIENIA

- Zarządzanie plikami i katalogami w systemie Linux
- Dowiązania
- Przypisywanie uprawnień do plików i katalogów

System Linux jest **systemem wielodostępnym**, w którym dostęp do plików jest regulowany przez prawo. Określone polecenia umożliwiają zarządzanie uprawnieniami plików, katalogów i użytkowników.

15.1. Zarządzanie plikami i folderami

Do wygodnego zarządzania plikami i katalogami można wykorzystać programy narzędziowe, np. **Midnight Commander**, lub środowisko graficzne. Oprócz tego Linux dysponuje poleceniami umożliwiającymi wykonanie wszystkich zadań z poziomu konsoli. Najczęściej używane polecenia do zarządzania plikami:

- **mkdir** – tworzy katalog, np.

```
sudo mkdir uczen
```

- **rmdir** – usuwa pusty katalog, np.

```
sudo rmdir uczen
```

(sprawdź w pomocy, jak usunąć niepusty katalog);

- **cp** – kopiuje pliki, np.

```
sudo cp /home/uczen/plik.txt /home/uczen2/
```

- **mv** – przenosi pliki, np.

```
sudo mv /home/uczen1/plik.txt /home/uczen2
```

- **rm** – usuwa pliki, np.

```
sudo rm stare.txt
```

- **cd** – zmienia aktualny katalog roboczy, np.

```
cd /
```

- **pwd** – wyświetla ścieżkę dostępu do aktualnego katalogu roboczego;

- **ls** – listuje pliki z katalogu, np. **ls**, najczęściej używa się polecenia z opcją **-la** (sprawdź w pomocy, co to zmieni w działaniu polecenia);

- **find** – wyszukuje, np.
`sudo find /muzyka -name uczen`
- **touch** – tworzy plik tekstowy, np.
`sudo touch plik.txt`
- **ln** – tworzy nowe dowiązanie do pliku, np.
`sudo ln -s link`
- **rename** – umożliwia zmianę nazwy pliku;
- **cat** – wyświetla zawartość pliku tekstowego.

Większość z powyższych poleceń działa analogicznie do systemu Windows. Omówienia wymagają jednak polecenia **find** i **ln**.

PRZYKŁAD 15.1

Zarządzanie plikami i katalogami

Skorzystaj z poleceń systemu Linux i utwórz w swoim katalogu domowym następującą strukturę zbiorów.

```
styczen
|-tydzien1
|   |-poniedzialek
|   |   |-plik1.txt
|   |   |-plik2.txt
|   |   |-plik3.txt
|   |-wtorek
|   |-sroda
|   |-czwartek
|   |-piatek
|-tydzien2
|   |-poniedzialek
|   |-wtorek
|   |-sroda
|   |-czwartek
|   |-piatek
|-tydzien3
|   |-poniedzialek
|   |-wtorek
|   |-sroda
|   |-czwartek
|   |-piatek
|-tydzien4
|   |-poniedzialek
|   |-wtorek
|   |-sroda
|   |-czwartek
|   |-piatek
```

Pozostaw strukturę zbiorów na dysku – będzie wykorzystywana w kolejnych ćwiczeniach.

15.2. Wyszukiwanie plików

Polecenie `find` służy do wyszukiwania plików. Jeżeli użytkownik chce znaleźć plik `tekst.txt` w katalogu `/home/student`, to wystarczy, że napisze:

```
sudo find /home/student -name tekst.txt
```

Pierwszy argument oznacza przeszukiwany katalog wraz z podkatalogami, drugi parametr określa warunek, np. `-name` oznacza, że użytkownik chce szukać pliku o określonej nazwie (sprawdź w pomocy, jak znaleźć wszystkie pliki, których jesteś właścicielem). Polecenie `find` akceptuje również nazwy plików określone za pomocą znaków globalnych, np. `sudo find /home/uczen -name *.bat`

PRZYKŁAD 15.2

Wyszukiwanie plików

Skorzystaj z konta użytkownika `root` i wyszukaj w systemie plików elementy przedstawione poniżej.

1. Wszystkie pliki zakończone `.txt` znajdujące się w strukturze zbiorów utworzonej w przykładzie 15.1.
2. Pliki `plik1.txt` i `plik2.txt` w strukturze zbiorów utworzonej w przykładzie 15.1.
3. Zbiory, których właścicielem jest użytkownik `uczen`.
4. Pliki `passwd`, `group`, `hosts`, `fstab`, `inittab`.

15.3. Dowiązania do plików

Linux przechowuje informacje o zbiorach, np. plikach lub katalogach, w strukturach nazywanych **i-węzłami** (ang. *i-node*). Każdy plik ma jeden węzeł. Identyfikacja pliku odbywa się na podstawie jego numeru, unikatowego w obrębie danego systemu plików. Człowiek woli posługiwać się nazwami plików, które również są skojarzone z odpowiednimi i-węzłami. **Dowiązania** (ang. *links*) umożliwiają odwoływanie się do jednego pliku za pomocą różnych nazw, pozwalają również na umieszczenie jednego pliku w wielu miejscach w strukturze plików.

Dowiązania dzielą się na **twarde** (ang. *hard links*) i **symboliczne** (ang. *symbolic links*). **Dowiązanie twarde** to referencja wskazująca konkretny, istniejący wcześniej i-węzeł w obrębie tej samej partycji (systemu plików). Dla systemu operacyjnego dowiązanie takie jest po prostu dodatkową nazwą wskazywanego obiektu. Plik mający `n` dowiązań ma też `n` nazw. Aby skasować obiekt w systemie plików, trzeba usunąć wszystkie odwołujące się do niego dowiązania.

Do tworzenia dowiązań twardych służy polecenie:

```
sudo ln cel_dowiązania dowiązanie,
```

gdzie:

`cel_dowiązania` – plik, do którego chcemy zrobić dowiązanie,
`dowiązanie` – dodatkowa nazwa pliku, równoprawna z wcześniej utworzonymi nazwami, np.

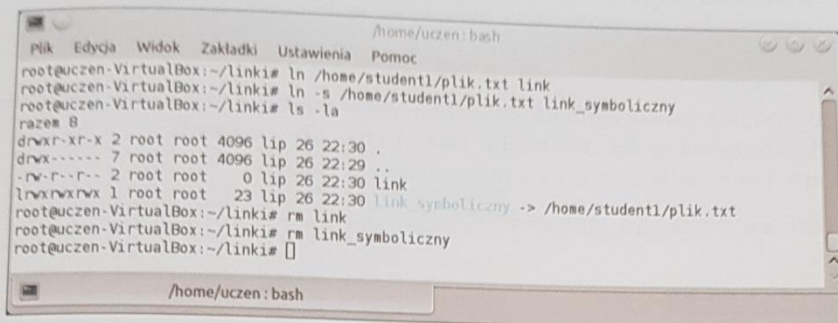
```
sudo ln /home/uczen/plik.txt /root/plik_ucznia.txt
```

Dowiązanie symboliczne to skojarzenie nowej nazwy z istniejącą wcześniej nazwą zbioru (nie bezpośrednio z i-węzłem). Dowiązanie symboliczne wskazuje na nazwę pliku lub

katalogu, która dopiero wskazuje na i-węzeł. Odpowiednikiem dowiązania symbolicznego w systemie Windows jest skrót. Dowiązania symboliczne tworzy się analogicznie do dowiązania twardego, dodając jednak do polecenia `ln` parametr `-s`, np.

```
sudo ln -s /home/uczen/plik.txt /root/plik_ucznia.txt
```

Dowiązania kasuje się tak jak pliki. Przykłady użycia poleceń związanych z tworzeniem i usuwaniem dowiązań pokazano na rys. 15.1.



```

/home/uczen: bash
Plik Edycja Widok Zakładki Ustawienia Pomoc
root@uczen-VirtualBox:~/linki# ln /home/student1/plik.txt link
root@uczen-VirtualBox:~/linki# ln -s /home/student1/plik.txt link_symboliczny
root@uczen-VirtualBox:~/linki# ls -la
razem 8
drwxr-xr-x 2 root root 4096 lip 26 22:30 .
drwx----- 7 root root 4096 lip 26 22:29 ..
-rw-r--r-- 2 root root 0 lip 26 22:30 link
lrwxrwxrwx 1 root root 23 lip 26 22:30 link_symboliczny -> /home/student1/plik.txt
root@uczen-VirtualBox:~/linki# rm link
root@uczen-VirtualBox:~/linki# rm link_symboliczny
root@uczen-VirtualBox:~/linki#

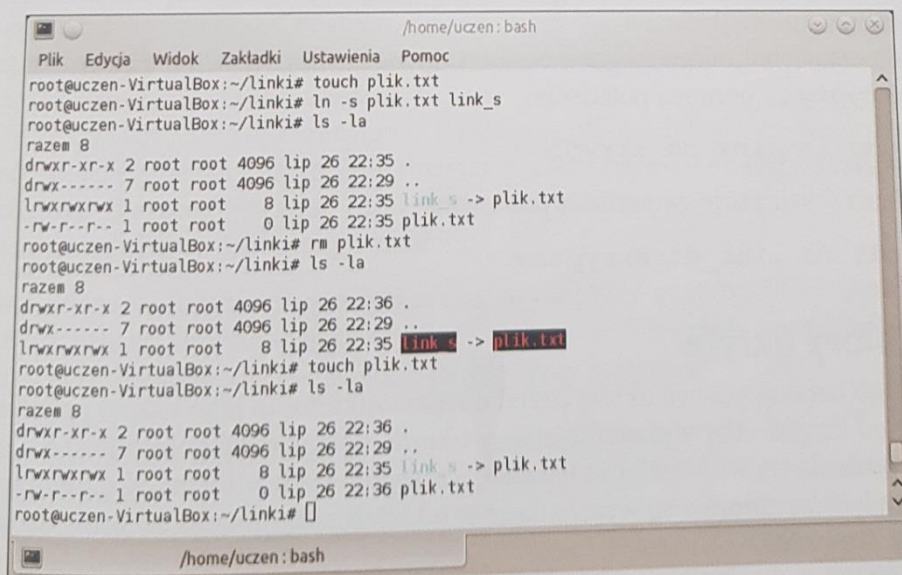
```

Rys. 15.1. Polecenie do tworzenia i usuwania dowiązań

15.4. Błędy związane z dowiązaniem

Dowiązanie twarde wskazuje na miejsce na dysku. Jeżeli plik zostanie skasowany i w jego miejscu zostanie umieszczony inny plik, to dowiązanie będzie wskazywało na to samo miejsce, ale zupełnie inny plik.

Dowiązanie symboliczne jest związane z nazwą pliku. Na rys. 15.2 pokazano dowiązanie symboliczne `link_s` utworzone do pliku `plik.txt`. Jeżeli skasujemy `plik.txt`, do którego prowadzi dowiązanie, to system wyświetli błąd. Dowiązanie istnieje, ale prowadzi do nieistniejącego pliku. Jeżeli zostanie utworzony plik o takiej samej nazwie, to dowiązanie będzie wskazywać nowy plik, mimo że jego zawartość może być zupełnie inna. Przykładową sytuację dotyczącą błędów z dowiązaniem przedstawiono na rys. 15.2.



```

/home/uczen: bash
Plik Edycja Widok Zakładki Ustawienia Pomoc
root@uczen-VirtualBox:~/linki# touch plik.txt
root@uczen-VirtualBox:~/linki# ln -s plik.txt link_s
root@uczen-VirtualBox:~/linki# ls -la
razem 8
drwxr-xr-x 2 root root 4096 lip 26 22:35 .
drwx----- 7 root root 4096 lip 26 22:29 ..
lrwxrwxrwx 1 root root 8 lip 26 22:35 link_s -> plik.txt
-rw-r--r-- 1 root root 0 lip 26 22:35 plik.txt
root@uczen-VirtualBox:~/linki# rm plik.txt
root@uczen-VirtualBox:~/linki# ls -la
razem 8
drwxr-xr-x 2 root root 4096 lip 26 22:36 .
drwx----- 7 root root 4096 lip 26 22:29 ..
lrwxrwxrwx 1 root root 8 lip 26 22:35 link_s -> plik.txt
root@uczen-VirtualBox:~/linki# touch plik.txt
root@uczen-VirtualBox:~/linki# ls -la
razem 8
drwxr-xr-x 2 root root 4096 lip 26 22:36 .
drwx----- 7 root root 4096 lip 26 22:29 ..
lrwxrwxrwx 1 root root 8 lip 26 22:35 link_s -> plik.txt
-rw-r--r-- 1 root root 0 lip 26 22:36 plik.txt
root@uczen-VirtualBox:~/linki#

```

Rys. 15.2. Błędy związane z dowiązaniem

PRZYKŁAD 15.3

Tworzenie i korzystanie z dowiązań twardych
 Dowiązania może tworzyć każdy użytkownik, ale pod warunkiem posiadania odpowiednich uprawnień. Aby utworzyć dowiązanie twarde z katalogu domowego użytkownika **root** do pliku **/etc/passwd**, wykonaj czynności przedstawione poniżej.

1. Zaloguj się na konto użytkownika **root**.
2. Utwórz w katalogu domowym użytkownika dowiązanie twarde do pliku **/etc/passwd** (polecenie `sudo ln /etc/passwd link_do_passwd`).
3. Wyświetl zawartość aktualnego katalogu, aby sprawdzić, czy dowiązanie zostało utworzone.
4. Skorzystaj z utworzonego wcześniej dowiązania i wyświetl zawartość pliku **/etc/passwd** (polecenie `sudo cat link_do_passwd`).
5. Usuń dowiązanie za pomocą polecenia:

```
sudo rm link_do_passwd
```

PRZYKŁAD 15.4

Tworzenie dowiązań symbolicznych i korzystanie z nich

Aby utworzyć dowiązanie symboliczne z katalogu domowego użytkownika **root** do katalogu **/home/uczen/skrypty**, wykonaj czynności przedstawione poniżej.

1. Skorzystaj z pierwszej konsoli i zaloguj się na konto użytkownika **root**.
2. Skorzystaj z drugiej konsoli i zaloguj się na konto użytkownika **uczen**.
3. Skorzystaj z konta **uczen** i utwórz w katalogu domowym podkatalog **skrypty** poleceniem `sudo mkdir skrypty` oraz plik **skrypt** poleceniem `sudo touch skrypty/skrypt`.
4. Utwórz dowiązanie symboliczne do katalogu **skrypty** w katalogu domowym użytkownika **uczen**. Skorzystaj z konta **root** i w katalogu domowym użytkownika **root** wpisz polecenie:

```
sudo ln /home/uczen/skrypty/link_do_skryptow
```

5. Wyświetl zawartość aktualnego katalogu, aby sprawdzić, czy dowiązanie zostało utworzone.
6. Skorzystaj z utworzonego wcześniej dowiązania i wyświetl zawartość katalogu **skrypty** za pomocą polecenia:

```
sudo ls link_do_skryptow
```

7. Usuń dowiązanie za pomocą polecenia:

```
sudo rm link_do_skryptow
```

15.5. Zbiory ukryte

Jeżeli plik lub katalog ma być ukryty przed użytkownikiem, to jego nazwa powinna rozpoczynać się od kropki. Aby wyświetlić zbiory ukryte, należy użyć polecenia **ls** z opcją **-a**. Opcja **-l** pozwala na wyświetlenie bardziej szczegółowych informacji o zbiorach. Informacje o każdym ze zbiorów są wyświetlane w oddzielnej linii. Opis zbioru składa się z:

- 10 bitów protekcji,
- liczby dowiązań do zbioru,
- identyfikatora właściciela zbioru.

- identyfikatora grupy właścicieli zbioru,
- wielkości zbioru,
- czasu ostatniej modyfikacji,
- nazwy zbioru.

Przykład działania polecenia listującego zawartość katalogu ze zbiorami ukrytymi pokazano na rys. 15.3.

```

uczen@uczen-ZSP:~$ ls
Dokumenty Muzyka Obrazy Pobrane Publiczny Pulpit Szablony Wideo
uczen@uczen-ZSP:~$ ls -la
razem 752
drwxr-xr-x 17 uczen uczen 4096 mar  9 22:12 .
drwxr-xr-x  5 root root  4096 mar  9 20:37 ..
-rw-r----- 1 uczen uczen  989 mar  9 23:08 .bash_history
-rw-r--r--  1 uczen uczen  220 paź  2 09:37 .bash_logout
-rw-r--r--  1 uczen uczen 3771 paź  2 09:37 .bashrc
drwxrwxr-x 12 uczen uczen 4096 mar  9 22:17 .cache
drwxr-xr-x  9 uczen uczen 4096 mar  9 22:18 .config
drwx----- 3 uczen uczen 4096 paź  2 09:49 .dbus
drwxr-xr-x  2 uczen uczen 4096 paź  2 09:49 Dokumenty
-rw-r--r--  1 uczen uczen 14965 paź  2 09:37 .face
lrwxrwxrwx  1 uczen uczen    5 paź  2 09:37 .face.icon -> .face
drwx----- 2 uczen uczen 4096 mar  9 22:12 .gconf
drwx----- 3 uczen uczen 4096 paź  2 09:49 .gnupg
-rw-rw-r--  1 uczen uczen  336 paź  2 09:49 .gtkrc-2.0
drwxr-xr-x  3 uczen uczen 4096 paź  2 09:49 .kde
drwxr-xr-x  3 uczen uczen 4096 paź  2 09:49 .local
drwxr-xr-x  2 uczen uczen 4096 paź  2 09:49 Muzyka
drwxr-xr-x  2 uczen uczen 4096 paź  2 09:49 Obrazy
drwxr-xr-x  2 uczen uczen 4096 paź  2 09:49 Pobrane
-rw-r--r--  1 uczen uczen  807 paź  2 09:37 .profile
drwxr-xr-x  2 uczen uczen 4096 paź  2 09:49 Publiczny
drwxr-xr-x  2 uczen uczen 4096 paź  2 09:49 Pulpit
-rw-r--r--  1 uczen uczen    0 mar  9 18:10 .sudo_as_admin_successful

```

Rys. 15.3. Polecenie listujące zawartość katalogu ze zbiorami ukrytymi

15.6. Ścieżki dostępu

Podobnie jak w systemie Windows, tak i w Linuksie do odnajdywania zbiorów na dysku można posłużyć się **ścieżkami dostępu**. Ścieżki mogą być bezwzględne lub względne.

Bezwzględna ścieżka dostępu rozpoczyna się od katalogu głównego, pierwszym znakiem ścieżki bezwzględnej jest ukośnik /.

Względna ścieżka dostępu przedstawia lokalizację pliku lub katalogu względem katalogu bieżącego. Jeśli ścieżka dostępu nie rozpoczyna się ukośnikiem, jest ścieżką względną. Aby z katalogu bieżącego przejść niżej w strukturze drzewa, wystarczy wpisać ścieżkę, rozpoczynając od nazwy następnego katalogu.

Katalogi, które są bardzo często wykorzystywane przez użytkowników Linuksa, mają specjalne oznaczenia umożliwiające szybkie przemieszczanie się między nimi.

Lokalizacja tych katalogów jest zawsze określona w systemie:

- / – katalog główny
- ~ – katalog domowy użytkownika,
- . – katalog bieżący
- .. – katalog bezpośrednio nadrzędny.

Symboli tych można używać w ścieżkach dostępu, np.

```
sudo cp ~/skrypty/konta/skrypty.
```

PRZYKŁAD 15.5

Korzystanie ze ścieżek dostępu
Zaloguj się na konto użytkownika **root**. Podczas wprowadzania ścieżek dostępu skorzystaj z mechanizmu uzupełniania nazw (klawisz [Tab]). Za pomocą poleceń zmień katalog bieżący na elementy przedstawione poniżej.

1. Przejdź do katalogu głównego za pomocą polecenia:

```
sudo cd /
```

2. Przejdź do katalogu domowego za pomocą polecenia:

```
sudo cd ~
```

3. Przejdź do katalogu **styczen** (utworzonego w przykładzie 15.1) w katalogu domowym za pomocą polecenia:

```
sudo cd ~/styczen
```

4. Skorzystaj ze ścieżki względnej i zmień katalog na **tydzien1/poniedzialek** za pomocą polecenia:

```
sudo cd tydzien1/poniedzialek
```

5. Utwórz w katalogu **~/styczen/tydzien1/wtorek** plik **plik.txt** za pomocą polecenia:

```
sudo touch ~/styczen/tydzien1/wtorek/plik.txt
```

6. Skopiuj plik utworzony w punkcie 5. do katalogu **~/styczen/tydzien1/sroda** za pomocą polecenia:

```
sudo cp ~/styczen/tydzien1/wtorek/plik.txt ~/styczen/tydzien1/sroda)
```

15.7. Uprawnienia do plików i katalogów

Każdy użytkownik niebędący administratorem Linuksa ma dostęp tylko do niektórych plików. Dzięki temu nikt nie może np. zmodyfikować plików należących do innego użytkownika. Do każdego pliku jest przypisany **identyfikator właściciela u** (ang. *user*) – użytkownika, który stworzył ten plik, oraz **grupy g** (ang. *group*), czyli zbioru użytkowników, którzy mają do tego pliku uprawnienia, w przeciwieństwie do **pozostałych o** (ang. *others*). Uprawnienia można też określać dla wszystkich użytkowników **a** (ang. *all*).

Istnieją trzy podstawowe prawa dostępu do pliku:

- **r** (ang. *read*) – prawo do **odczytu**;
- **w** (ang. *write*) – prawo do **zapisu**;
- **x** (ang. *execute*) – prawo do **uruchomienia**, jeśli plik jest programem.

Prawa te są nadawane niezależnie właścicielowi pliku, grupie, do której plik należy i pozostałym użytkownikom. Aby dowiedzieć się, jakie uprawnienia są ustawione dla poszczególnych plików, używa się komendy **ls -l**. Każdy plik i katalog w systemie Linux ma 10 bitów protekcji w formacie **drwxrwxrwx**, gdzie:

- bit 1 – identyfikacja rodzaju zbioru:
 - **d** (ang. *directory*) – to katalog;
 - **-** – to plik;
 - **l** (ang. *link*) – to link do pliku;

- bity 2–4 – uprawnienia właściciela pliku;
- bity 5–7 – uprawnienia grupy, do której należy właściciel;
- bity 8–10 – uprawnienia pozostałych użytkowników.

Jeżeli na bitach 2–10 występuje litera, to oznacza ustawione uprawnienie, natomiast kreska to brak uprawnienia.

Administrowanie uprawnieniami jest możliwe za pomocą polecenia **chmod**, np.

```
sudo chmod ugo+w plik.txt
```

powoduje dodanie (+) prawa do zapisu **w** (ang. *write*) dla pliku o nazwie **plik.txt** właścicielowi **u** (ang. *user*), użytkownikom należącym do tej samej grupy co właściciel pliku **g** (ang. *group*) oraz dodanie prawa do zapisu pliku wszystkim pozostałym użytkownikom **o** (ang. *others*).

Każde z praw dostępu ma przypisany odpowiedni parametr cyfrowy:

- **r** – prawo do odczytu 4;
- **w** – prawo do zapisu 2;
- **x** – prawo do uruchomienia 1;
- **--** – brak praw dostępu 0.

Po dodaniu odpowiednich parametrów, zestaw trzech praw można przedstawić za pomocą jednej cyfry.

Oto możliwe kombinacje:

- **---** 0 brak praw;
- **--x** 1 prawo do uruchomienia;
- **-w-** 2 prawo do zapisu;
- **-wx** 3 prawo do zapisu i wykonania;
- **r--** 4 prawo do odczytu;
- **r-x** 5 prawo do odczytu i uruchomienia;
- **rw-** 6 prawo do odczytu i zapisu;
- **rwX** 7 prawo do odczytu, zapisu i uruchomienia.

W poleceniu **chmod** można podać uprawnienia w postaci liczby trzycyfrowej, w której kolejne cyfry od lewej strony oznaczają uprawnienia dla właściciela, grupy i pozostałych użytkowników. Aby ustawić uprawnienia, należy wpisać polecenie:

```
chmod 750 plik.txt
```

Zostaną przypisane uprawnienia 7 dla właściciela (**rwX**), 5 dla grupy (**r-x**), 0 dla pozostałych użytkowników (**---**).

15.8. Właściciele zbiorów

Właścicielami każdego pliku i katalogu są użytkownik i grupa użytkowników. Przeniesić własność zbioru na innego użytkownika może tylko administrator. Do zmiany właściciela używa się polecenia, którego składnia jest następująca (rys. 15.4):

```
chown użytkownik:grupa plik
```

```

/home/uczen: bash
Plik Edycja Widok Zakładki Ustawienia Pomoc
root@uczen-VirtualBox:~/pliki# ls -la
raze# B
drwxr-xr-x 2 root root 4096 lip 26 22:42 .
drwx----- 7 root root 4096 lip 26 22:42 ..
-rw-r--r-- 1 root root  0 lip 26 22:42 plik.txt
root@uczen-VirtualBox:~/pliki# chown student1:student1 plik.txt
root@uczen-VirtualBox:~/pliki# ls -la
raze# B
drwxr-xr-x 2 root    root    4096 lip 26 22:42 .
drwx----- 7 root    root    4096 lip 26 22:42 ..
-rw-r--r-- 1 student1 student1  0 lip 26 22:42 plik.txt
root@uczen-VirtualBox:~/pliki#

```

Rys. 15.4. Przenoszenie własności pliku na innego użytkownika

! UWAGA

Zmiana właściciela zbioru może spowodować zmianę uprawnień przypisanych użytkownikom do danego zbioru.

PRZYKŁAD 15.6

Zarządzanie uprawnieniami i prawami własności

Aby zarządzać uprawnieniami i prawami własności, należy dysponować odpowiednimi uprawnieniami. Aby przydzielić uprawnienia do pliku **prawa.txt** użytkownikowi **uczen**, wykonaj czynności przedstawione poniżej.

1. Zaloguj się na konto użytkownika **root**.
2. Utwórz w katalogu domowym użytkownika **uczen** plik **prawa.txt** za pomocą polecenia:

```
sudo touch /home/uczen/prawa.txt
```

3. Zmień właściciela pliku na użytkownika **uczen** i grupę **uczen** za pomocą polecenia:

```
sudo chown uczen:uczen /home/uczen/prawa.txt
```

4. Przydziel uprawnienia **rwX-wX--** do pliku **prawa.txt** za pomocą polecenia:

```
sudo chmod 734 /home/uczen/prawa.txt
```

5. Wyświetl zawartość katalogu **/home/uczen/** w celu sprawdzenia właściciela i uprawnień.

🚩 SPRAWDŹ SWOJĄ WIEDZĘ

1. Scharakteryzuj zarządzanie plikami i folderami w systemie Linux.
2. Na czym polegają dowiązania?
3. Scharakteryzuj zarządzanie zbiorami ukrytymi.
4. Jakie wyróżniamy rodzaje ścieżek dostępu?
5. Jakiego polecenia używamy do zmiany właściciela zbiorów?