Przedmiot: Systemy operacyjne

Laboratorium Ubuntu Linux.

Opracował: Krzysztof Ropiak

Spis treści

1.	Wybrane komendy powłoki bash 3					
	1.1. Wyświetlanie dokumentacji dla wybranego narzędzia lub zagadnienia z poziomu shel					
1.2. Is (ang. list) – wyświetla informacje o zawartości folderu						
1.3. Komenda cd (ang. change directory) – przejście do innego katalogu w drzewie sys plików.6						
	1.4. robocz	Polecenie pwd (ang. print working directory) wyświetla ścieżkę do aktualnego folderu rego (tego, w którym akurat się znajdujemy)6				
	1.5.	1.5. Polecenie mkdir tworzy katalog o podanej nazwie we wskazanej lokalizacji				
	1.6.	Polecenie touch tworzy nowy plik7				
	1.7.	Kopiowanie plików odbywa się przy wykorzystaniu polecenia cp (ang. copy)8				
	1.8. polece	Przenoszenie i zmiana nazwy plików/folderów może zostać dokonana z pomocą nia πν (ang. move)9				
	1.9.	Narzędzie chmod i zmiana uprawnień do zasobów11				
	1.10.	Zmiana właściciela zasobu przy pomocy narzędzia chown				
2.	Zadani	a wymagające podniesionych uprawnień14				
3.	Tworzenie i zarządzanie użytkownikami oraz grupami14					
4.	Potoki	Potoki i przekierowanie danych. Podstawowa kontrola zadań				
5.	Lista d	odatkowych komend, które warto znać. Gdzie szukać pomocy i informacji ?				
	5.1.	Lista komend, którym warto się bliżej przyjrzeć: 18				
	5.2.	Gdzie szukać informacji o Ubuntu ?18				

1. Wybrane komendy powłoki bash.

1.1. Wyświetlanie dokumentacji dla wybranego narzędzia lub zagadnienia z poziomu shell-a.

Jeżeli znamy komendę, którą chcemy użyć to sprawa jest dość prosta. Polecenie man <komenda> lub nowsza jego wersja info <komenda> wyświetla "manual" dla danej komendy z możliwymi opcjami i czasem przykładami użycia.

Przykład:

> man ls - wyświetli niżej widoczne informacje (ukazane tylko częściowo).

```
😣 🖻 💿 🛛 kropiak@kropiak-VirtualBox: ~/Pobrane/obrazy
LS(1)
                                        User Commands
                                                                                        LS(1)
NAME
        ls - list directory contents
SYNOPSIS
        ls [<u>OPTION</u>]... [<u>FILE</u>]...
DESCRIPTION
        List information about the FILEs (the current directory by default).
Sort entries alphabetically if none of -cftuvSUX nor --sort is speci-
        fied.
        Mandatory arguments to long options are mandatory for short options
        too.
        -a, --all
                do not ignore entries starting with .
        -A, --almost-all
                do not list implied . and ..
        --author
                 with -1, print the author of each file
        -b, --escape
                print C-style escapes for nongraphic characters
        --block-size=SIZE
                  scale sizes by SIZE before printing them. E.g.,
--block-size=M' prints sizes in units of 1,048,576 bytes. See
                scale
                SIZE format below.
        -B, --ignore-backups
                do not list implied entries ending with ~
                with -lt: sort by, and show, ctime (time of last modification of
        -c
                file status information) with -1: show ctime and sort by name otherwise: sort by ctime, newest first
                list entries by columns
        -C
        --color[=WHEN]
                 colorize the output. WHEN defaults to `always' or can be
`never' or `auto'. More info below
        -d, --directory
                list directory entries instead of contents, and do not dereference symbolic links
Manual page ls(1) line 1/243 19% (press h for help or q to quit)
```

Polecenie info ls zwróci podobny wynik, ale jest duże prawdopodobieństwo, że informacje tam zawarte będą bardziej aktualne i obszerniejsze.

Jeżeli nie znamy konkretnego polecenia, którego należy użyć w celu wykonania powierzonego nam zadania, możemy użyć wyszukiwania odpowiedniego "manuala" po słowach kluczowych jak w przykładzie poniżej.

😮 🗐 🗊 kropiak@kr	opiak-VirtualBox: ~/Pobrane/obrazy
kropiak@kropiak-Vi	.rtualBox:~/Pobrane/obrazy\$ man -k permission
access (2)	 check real user's permissions for a file
chmod (2)	 change permissions of a file
eaccess (3)	 check effective user's permissions for a file
euidaccess (3)	 check effective user's permissions for a file
faccessat (2)	 check user's permissions of a file relative to a direc
fchmod (2)	 change permissions of a file
fchmodat (2)	 change permissions of a file relative to a directory f
ioperm (2)	 set port input/output <u>p</u>ermissions
kropiak@kropiak-Vi	rtualBox:~/Pobrane/obrazy\$

Ukaże się lista poleceń, które mają cokolwiek wspólnego z uprawnieniami (permission). Można stosować kilka słów kluczowych jednocześnie, należy ująć je między znakami " oraz".

Kolejną możliwością uzyskania pomocy dla konkretnego polecenia jest skorzystanie z opcji --help po nazwie polecenia, np.: ls --help. Spowoduje to wyświetlenie pomocy podobnej do tej z polecenia man, z tą różnicą, iż mamy większą szansę na uzyskanie pomocy w języku, który został ustawiony jako domyślny język użytkownika (np. polski).

1.2. ls (ang. list) - wyświetla informacje o zawartości folderu

Bez dodatkowych parametrów domyślnie wyświetlana jest zawartość folderu bez plików ukrytych oraz w postaci kolumn nazw zasobów uporządkowanych alfabetycznie. Poprzez dobranie odpowiednich parametrów narzędzie pozwala na sortowanie wyników, zmianę sposobu ich uporządkowania i prezentacji, np. poprzez wyświetlenie informacji o uprawnieniach, dacie modyfikacji, właścicielu.

Przykłady:

ls wyświetla zawartość aktualnego folderu roboczego,

```
    kropiak@kropiak-VirtualBox: ~
    kropiak@kropiak-VirtualBox:~$ ls
    Dokumenty Muzyka Pobrane Pulpit Wideo
    examples.desktop Obrazy Publiczny Szablony
    kropiak@kropiak-VirtualBox:~$
```

ls <ścieżka> wyświetla zawartość folderu wskazanego przez ścieżkę,



ls – a wyświetla zawartość folderu wraz	z zasobami ukrytymi,
---	----------------------

🛞 🗖 🗊 kropiak@kropiak-VirtualBox: ~	
<pre>kropiak@kropiak-VirtualBox:~\$ ls -a </pre>	.pulse-cookie Szablony .thumbnails .vboxclient-clipboard.pid .vboxclient-display.pid .vboxclient-draganddrop.pid .vboxclient-seamless.pid Wideo .Xauthority .xsession-errors .xsession-errors.old

ls –al wyświetla dodatkowe informacje o zasobach w danym folderze.

😣 🖲 🗊 kr	opia	k@kropiak	-VirtualBo	ox: ~				
kropiak@kropiak-VirtualBox:~\$ ls -al razem 172								
drwxr-xr-x	20	kropiak	kropiak	4096	маг	10	13:53	
drwxr-xr-x	4	root	root	4096	маг	10	12:15	
-rw	1	kropiak	kropiak	670	маг	10	13:50	.bash_history
-rw-rr	1	kropiak	kropiak	220	маг	9	20:41	.bash_logout
-rw-rr	1	kropiak	kropiak	3486	маг	9	20:41	.bashrc
drwx	15	kropiak	kropiak	4096	маг	9	20:58	.cache
drwx	10	kropiak	kropiak	4096	маг	10	13:53	.config
drwx	3	kropiak	kropiak	4096	маг	9	20:50	.dbus
-rw-rr	1	kropiak	kropiak	25	маг	10	13:52	.dmrc
drwxr-xr-x	2	kropiak	kropiak	4096	маг	9	20:49	Dokumenty
-rw-rr	1	kropiak	kropiak	8445	маг	9	20:41	examples.desktop
drwx	3	kropiak	kropiak	4096	маг	10	13:52	.gconf
drwx	4	kropiak	kropiak	4096	маг	9	20:50	.gnome2
-rw	1	kropiak	kropiak	0	маг	10	13:50	.goutputstream-CNN3BX
- rw- rw- r	1	kropiak	kropiak	143	маг	10	13:52	.gtk-bookmarks
dr-x	2	kropiak	kropiak	0	маг	10	13:52	.gvfs
-rw	1	kropiak	kropiak	1098	маг	10	13:52	.ICEauthority
drwxr-xr-x	3	kropiak	kropiak	4096	маг	9	20:50	.local
drwx	3	kropiak	kropiak	4096	маг	9	20:50	.mission-control
drwxr-xr-x	2	kropiak	kropiak	4096	маг	9	20:49	Muzyka
drwxr-xr-x	2	kropiak	kropiak	4096	маг	10	13:50	Obrazy
drwxr-xr-x	2	kropiak	kropiak	4096	маг	9	20:49	Pobrane
-								

Rozpoczynając od lewej strony:

Pierwsza litera oznacza typ zasobu - "d" – folder (directory), "-" – oznacza plik.

Kolejne 9 znaków to uprawnienia:

- pierwsze 3 od lewej to uprawnienia dla użytkownika (ang. user, w skrócie "u"),

- kolejne 3 to uprawnienia dla grupy (ang. group, w skrócie "g"),

- ostatnie 3 to uprawnienia dla pozostałych (ang. others, w skrócie "o").

Litery "rwx" oraz znak "-" w uprawnieniach oznaczają:

r – prawo do odczytu (read),

w - prawo do zapisu (write),

x – prawo do wykonania (execute) w przypadku pliku; prawo do przeszukiwania w przypadku katalogu,

"-" brak danego uprawnienia.

Kolejne kolumny to:

- ilość dowiązań twardych do danego zasobu,
- właściciel,
- grupa,
- rozmiar na dysku,
- data i czas ostatniej modyfikacji,
- nazwa zasobu.

Zobacz również komendy dir, vdir.

1.3. Komenda **cd** (ang. change directory) – przejście do innego katalogu w drzewie systemu plików.

Przykłady:



Polecenie **cd** jako parametr może przyjąć ścieżkę względną jak i bezwzględną. Aby przejść o jeden poziom wyżej w strukturze systemu plików należy wykonać polecenie **cd**...

Polecenie **cd** ~ przenosi nas do folderu domowego użytkownika.

1.4. Polecenie **pwd** (ang. print working directory) wyświetla ścieżkę do aktualnego folderu roboczego (tego, w którym akurat się znajdujemy).

1.5. Polecenie **mkdir** tworzy katalog o podanej nazwie we wskazanej lokalizacji.

Przykłady:



1.6. Polecenie touch tworzy nowy plik.

Przykłady:

```
    kropiak@kropiak-VirtualBox: ~/Pobrane/obrazy
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls
obrazy test test2 test3
kropiak@kropiak-VirtualBox: ~/Pobrane$ touch obrazy/przyroda/lista_zdjęć.txt
kropiak@kropiak-VirtualBox: ~/Pobrane$ cd obrazy
kropiak@kropiak-VirtualBox: ~/Pobrane/obrazy$ ls -R
.:
przyroda
./przyroda:
lista_zdjęć.txt
kropiak@kropiak-VirtualBox: ~/Pobrane/obrazy$ [
```

W przypadku dwóch powyższych komend można zadać pytanie – w jaki sposób system "wie" jakie domyślne uprawnienia ustawić dla nowo tworzonych folderów i plików. Mechanizm ten to umask, domyślna maska (szablon) według którego uprawnienia są nadawane. Poleceniem umask można wyświetlić wartość tej maski dla użytkownika. Wartość umask może być ustawiona w pliku ~/.bashrc lub ~/.profile. Jeżeli chcemy ustawić wartość umask globalnie dla wszystkich użytkowników to musimy dokonać odpowiedniego wpisu w pliku /etc/login.defs.

1.7. Kopiowanie plików odbywa się przy wykorzystaniu polecenia cp (ang. copy).

Przykłady:

😰 亘 💿 kropiak@kropiak-VirtualBox: ~/Pobrane kropiak@kropiak-VirtualBox:~/Pobrane/test\$ ls plik2.txt plik3.txt plik.txt kropiak@kropiak-VirtualBox:~/Pobrane/test\$ cd .. kropiak@kropiak-VirtualBox:~/Pobrane\$ ls test kropiak@kropiak-VirtualBox:~/Pobrane\$ mkdir test2 kropiak@kropiak-VirtualBox:~/Pobrane\$ ls test test2 kropiak@kropiak-VirtualBox:~/Pobrane\$ cp test test2 cp: katalog "test" został pominięty kropiak@kropiak-VirtualBox:~/Pobrane\$ ls test test2 kropiak@kropiak-VirtualBox:~/Pobrane\$ cd test2 kropiak@kropiak-VirtualBox:~/Pobrane/test2\$ ls kropiak@kropiak-VirtualBox:~/Pobrane/test2\$ cd .. kropiak@kropiak-VirtualBox:~/Pobrane\$ cp -R test test2 kropiak@kropiak-VirtualBox:~/Pobrane\$ ls test test2 kropiak@kropiak-VirtualBox:~/Pobrane\$ cd test2 kropiak@kropiak-VirtualBox:~/Pobrane/test2\$ ls test kropiak@kropiak-VirtualBox:~/Pobrane/test2\$ ls -R . : test ./test: plik2.txt plik3.txt plik.txt kropiak@kropiak-VirtualBox:~/Pobrane/test2\$ cd .. kropiak@kropiak-VirtualBox:~/Pobrane\$ ls test test2 kropiak@kropiak-VirtualBox:~/Pobrane\$ mkdir test3 kropiak@kropiak-VirtualBox:~/Pobrane\$ ls test test2 test3 kropiak@kropiak-VirtualBox:~/Pobrane\$ cp test/* test3 kropiak@kropiak-VirtualBox:~/Pobrane\$ ls -R . : test test2 test3 ./test: k plik2.txt plik3.txt plik.txt ./test2: test ./test2/test: plik2.txt plik3.txt plik.txt ./test3: plik2.txt plik3.txt plik.txt kropiak@kropiak-VirtualBox:~/Pobrane\$

Warto zwrócić uwagę, iż polecenie **cp** bez dodatkowych parametrów pomija foldery podczas operacji kopiowania. Dopiero użycie opcji **–R** umożliwia przeprowadzenie takiej operacji.

1.8. Przenoszenie i zmiana nazwy plików/folderów może zostać dokonana z pomocą polecenia mv (ang. move).

Przykłady:

a.) zmiana nazwy pliku

```
kropiak@kropiak-VirtualBox:~/Pobrane
kropiak@kropiak-VirtualBox:~/Pobrane$ ls
obrazy test test2 test3
kropiak@kropiak-VirtualBox:~/Pobrane$ ls test
plik2.txt plik3.txt plik.txt
kropiak@kropiak-VirtualBox:~/Pobrane$ mv test/plik.txt test/plik4.txt
kropiak@kropiak-VirtualBox:~/Pobrane$ ls test
plik2.txt plik3.txt plik4.txt
kropiak@kropiak-VirtualBox:~/Pobrane$
```

b.) przeniesienie pliku z zachowaniem nazwy



c.) przeniesienie wszystkich plików z wybranego folderu do innego katalogu

```
@@@ kropiak@kropiak-VirtualBox:~/Pobrane/test
kropiak@kropiak-VirtualBox:~/Pobrane/test$ ls -R
.:
test2 test3
./test2:
plik2.txt plik3.txt plik.txt
./test3:
kropiak@kropiak-VirtualBox:~/Pobrane/test$ mv test2/* test3/
kropiak@kropiak-VirtualBox:~/Pobrane/test$ ls -R
.:
test2 test3
./test2:
./test2:
./test3:
plik2.txt plik3.txt plik.txt
kropiak@kropiak-VirtualBox:~/Pobrane/test$
```

d.) zachowanie kopii pliku docelowego w przypadku, gdy dany plik już istnieje i operacja przeniesienia nadpisałaby jego zawartość

× • •	kropiak@kropiak-VirtualBox: ~/Pobrane/test/test3	
kropiak(plik1.t; kropiak(kropiak(plik2.t; kropiak(@kropiak-VirtualBox:~/Pobrane/test/test3\$ ls xt plik2.txt @kropiak-VirtualBox:~/Pobrane/test/test3\$ mw @kropiak-VirtualBox:~/Pobrane/test/test3\$ ls xt @kropiak-VirtualBox:~/Pobrane/test/test3\$ []	plik1.txt plik2.txt
ו•	kropiak@kropiak-VirtualBox: ~/Pobrane/test/test3	
kropiak plik1.t kropiak 2.txt	@kropiak-VirtualBox:~/Pobrane/test/test3\$ ls xt plik2.txt @kropiak-VirtualBox:~/Pobrane/test/test3\$ mv	; /suffix=.bak plik1.txt plik
kropiak plik2.t: kropiak	@kropiak-VirtualBox:~/Pobrane/test/test3\$ ls xt plik2.txt.bak @kropiak-VirtualBox:~/Pobrane/test/test3\$	

e.) przeniesienie zawartości tylko wtedy, gdy w docelowej lokalizacji brak pliku źródłowego lub jest on starszy niż przenoszony,

```
🔊 🗇 🗊 🛛 kropiak@kropiak-VirtualBox: ~/Pobrane
kropiak@kropiak-VirtualBox:~/Pobrane$ ls -AlR
razem 8
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:40 test
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:40 test2
./test:
razem 0
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 13:39 plik1.txt
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 13:40 plik2.txt
./test2:
razem 0
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 13:40 plik1.txt
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 13:28 plik4.txt
kropiak@kropiak-VirtualBox:~/Pobrane$ mv -u test/* test2/
kropiak@kropiak-VirtualBox:~/Pobrane$ ls -AlR
                                                       k
.:
razem 8
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test2
./test:
razem 0
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 13:39 plik1.txt
./test2:
razem 0
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 13:40 plik1.txt
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 13:40 plik2.txt
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 <u>1</u>3:28 plik4.txt
kropiak@kropiak-VirtualBox:~/Pobrane$
```

Użyj narzędzia *rm* do usuwania plików i folderów.

1.9. Narzędzie chmod i zmiana uprawnień do zasobów.

Przy okazji omawiania komendy **ls** zostały omówione uprawnienia, które towarzyszą wylistowanym zasobom. Komenda **chmod** pozwala na zmianę tych uprawnień. Omawiana komenda posiada wiele możliwości więc najlepiej jak zostaną one omówione poprzez przykłady:

a.) dodanie pojedynczego uprawnienia do zapisu dla pozostałych użytkowników a

```
następnie ponowne zdjęcie tego uprawnienia
```

```
kropiak@kropiak-VirtualBox: ~/Pobrane
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
kropiak@kropiak-VirtualBox: ~/Pobrane$ chmod o+w test
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
kropiak@kropiak-VirtualBox: ~/Pobrane$ chmod o-w test
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
```

b.) zdjęcie uprawnienia do zapisu i wykonania dla wszystkich poza właścicielem

```
kropiak@kropiak-VirtualBox: ~/Pobrane
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test2
kropiak@kropiak-VirtualBox: ~/Pobrane$ chmod go-wx test
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxr-r-- 2 kropiak kropiak 4096 mar 11 13:41 test
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
```

c.) powyższe rozwiązania mają jedną wadę – aby zmienić te uprawnienia trzeba najpierw sprawdzić jakie są aktualne ich wartości. Można posłużyć się operatorem "=", który spowoduje dodanie lub zdjęcie uprawnień tak, aby przyjęły pożądaną wartość.

```
kropiak@kropiak-VirtualBox: ~/Pobrane
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxr--r-- 2 kropiak kropiak 4096 mar 11 13:41 test
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test2
kropiak@kropiak-VirtualBox: ~/Pobrane$ chmod go=rw test
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxrw-rw- 2 kropiak kropiak 4096 mar 11 13:41 test2
kropiak@kropiak-VirtualBox: ~/Pobrane$
```

d.) przydatną opcją jest "kopiowanie" uprawnień dla zasobu z innego istniejącego zasobu

```
kropiak@kropiak-VirtualBox: ~/Pobrane
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxrw-rw- 2 kropiak kropiak 4096 mar 11 13:41 test2
kropiak@kropiak-VirtualBox: ~/Pobrane$ chmod --reference=test2 test
kropiak@kropiak-VirtualBox: ~/Pobrane$ ls -Al
razem 8
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
kropiak@kropiak-VirtualBox: ~/Pobrane$
```

- e.) ustawienie uprawnień dla wszystkich zasobów podrzędnych na podstawie uprawnień
 - obiektu nadrzędnego

```
😣 🗐 🗊 kropiak@kropiak-VirtualBox: ~/Pobrane
kropiak@kropiak-VirtualBox:~/Pobrane$ ls -AlR
. :
razem 8
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test2
./test:
razem 0
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 13:39 plik1.txt
./test2:
razem 0
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 13:40 plik1.txt
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 13:40 plik2.txt
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 13:28 plik4.txt
kropiak@kropiak-VirtualBox:~/Pobrane$ chmod -R --reference=test2 test2/
kropiak@kropiak-VirtualBox:~/Pobrane$ ls -AlR
razem 8
                                                                      k
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test2
./test:
razem 0
-rw-rw-r-- 1 kropiak kropiak 0 mar 11 13:39 plik1.txt
./test2:
razem 0
-rwxrwxr-x 1 kropiak kropiak 0 mar 11 13:40 plik1.txt
-rwxrwxr-x 1 kropiak kropiak 0 mar 11 13:40 plik2.txt
-rwxrwxr-x 1 kropiak kropiak 0 mar 11 13:28 plik4.txt
kropiak@kropiak-VirtualBox:~/Pobrane$ 🗌
```

Do zmiany grupy dla zasobu używamy narzędzia chgrp.

Istnieje jeszcze wiele innych kombinacji oraz udogodnień usprawniających modyfikacje uprawnień do zasobów i w celu zapoznania się z nimi odsyłam do:

http://en.wikipedia.org/wiki/Chmod

1.10. Zmiana właściciela zasobu przy pomocy narzędzia chown.

Jak wynika z zrzutu ekranu widocznego poniżej, aby zmienić właściciela dla zasobu należy posiadać uprawnienia super użytkownika.

😣 🖻 🗉 kropiak@kropiak-VirtualBox: ~/Pobrane
kropiak@kropiak-VirtualBox:~/Pobrane\$ ls -l razem 8
drwxrwxrwx 2 kropiak kropiak 4096 mar 11 13:41 <mark>test</mark>
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 <mark>test2</mark>
kropiak@kropiak-VirtualBox:~/Pobrane\$ chown test test
chown: zmiana właściciela "test": Operacja niedozwolona
kropiak@kropiak-VirtualBox:~/Pobrane\$ sudo chown test test
kropiak@kropiak-VirtualBox:~/Pobrane\$ ls -l
razem 8
drwxrwxrwx 2 test kropiak 4096 mar 11 13:41 🚾
drwxrwxr-x 2 kropiak kropiak 4096 mar 11 13:41 test2
kropiak@kropiak-VirtualBox:~/Pobrane\$ 📋

Zmiana właściciela powoduje, że teraz nowy właściciel (oprócz super użytkowników) ma prawo do zmiany uprawnień do zasobu a poprzedni może stracić część uprawnień w zależności od ustawień.

2. Zadania wymagające podniesionych uprawnień.

W przypadku dystrybucji Linux Ubuntu wykonywanie czynności administracyjnych odbywa się nieco inaczej niż ma to miejsce w przypadku większości dystrybucji linuksowych. Najważniejszym użytkownikiem systemu wciąż jest **root**, ale zamiast narzędzia **su** (substitute user ID), które domyślnie nie działa w Ubuntu, do wykonywania czynności jako super użytkownik wykorzystujemy narzędzie **sudo** (substitude user ID do lub czasem superuser do). Różnica między tymi dwoma jest zasadnicza. W przypadku polecenia **su** najpierw podajemy hasło, uzyskujemy dostęp do specjalnych uprawnień a następnie możemy wykonywać polecenia jako super użytkownik. Z kolei w przypadku **sudo** najpierw podajemy polecenie i jego opcje a następnie jesteśmy proszeni o podanie hasła (ale nie hasła root-a tylko naszego). Jeżeli wszystko się zgadza polecenie zostanie wykonane z podniesionymi uprawnieniami a następnie ponownie uprawnienia zostaną zmniejszone.

Informacje o konfiguracji narzędzia *sudo* można znaleźć w pliku /etc/sudoers .

3. Tworzenie i zarządzanie użytkownikami oraz grupami.

Większość istotnych informacji o użytkownikach jest przechowywana w plikach:

- /etc/passwd podstawowe informacje o kontach użytkowników (ew. zaszyfrowane hasło),
- /etc/group podstawowe informacje o grupach użytkowników,
- /etc/shadow rozszerzone informacje o kontach użytkowników (np. daty ważności) i zaszyfrowane hasło (w systemie shadow),
- /etc/gshadow rozszerzone informacje o grupach użytkowników (w systemie shadow).

Przykładowa linia z pliku /etc/passwd:

test:x:1001:1001:Jan Testowy,,,;/home/test:/bin/bash

Znaczenie wartości poczynając od lewej strony:

- nazwa użytkownika,

- pole hasła, w tym przypadku wartość 'x' oznacza, że informacje i haśle znajdują się w pliku /etc/shadow

- ID użytkownika, wartość numeryczna powiązana z użytkownikiem,
- ID grupy, której domyślnym członkiem jest dany użytkownik,
- pełna nazwa użytkownika oraz inne informacje tekstowe, np. adres, telefon o ile zostały zdefiniowane
- katalog domowy użytkownika,
- aplikacja uruchamiana po zalogowaniu się użytkownika, w tym przypadku powłoka (shell),

Linia z informacjami o grupie z pliku /etc/group wygląda następująco:

test:x:1001:

Analogicznie do wpisów z informacjami o użytkowniku mamy tutaj kolejno dane o nazwie grupy, haśle (również w pliku **/etc/shadow**) oraz ID grupy. Na końcu może się również znajdować lista użytkowników (rozdzielona przecinkiem), którzy należą do danej grupy.

Listę grup, do których należy zalogowany użytkownik można sprawdzić poleceniem groups.

Najpopularniejsze sposoby tworzenia kont użytkowników to:

- wykorzystanie narzędzia useradd,
- wykorzystanie narzędzia adduser,
- ręczna edycja plików z definicjami użytkowników.

Istniejące konto można modyfikować na kilka sposobów:

- polecenie chfn zmienia informacje GECOS (imię, nazwisko, itp.) o użytkowniku,
- polecenie chsh zmienia powłokę,
- narzędzie usermod modyfikuje dowolne parametry konta,
- narzędzie **groupmod** j.w. dla grupy,
- instrukcja passwd zakłada nowe hasło, a w systemie shadow passwords zmienia daty ważności konta.

Usuwanie kont i grup:

- konto użytkownika można usunąć przy pomocy polecenia userdel,
- powyższe polecenie z opcją -r usuwa katalog domowy użytkownika,
- polecenie groupdel usuwa grupy użytkowników,
- przy pomocy polecenia instrukcji **find** można odnaleźć i usunąć pliki których właścicielem jest podany użytkownik lub grupa.

Blokowanie dostępu do konta:

- poleceniem passwd z opcją -l,
- przez ręczną modyfikację hasła w pliku passwd/shadow,
- poprzez zmianę powłoki użytkownika na program nie dopuszczający logowania.

W zależności od narzędzia system przyjmie pewne wartości domyślne w przypadku tworzenia użytkowników np. wybrana powłoka, domyślny folder domowy. Część informacji domyślnych dla całego systemu znajduje się w pliku **/etc/profile**, ale znacznie więcej znajdziemy w **/etc/login.defs.**

Jeżeli korzystamy z narzędzia *adduser* warto zobaczyć jakie są jego domyślne ustawienia, które znajdziemy w **/etc/adduser.conf**. Dodatkowo podczas tworzenia konta przy pomocy *adduser* tworzony jest folder użytkownika oraz pewna struktura folderów wewnątrz. Istnieje możliwość wpływania na to jak ta struktura ma wyglądać – zobacz **/etc/skel**.

Po utworzeniu konta użytkownika informacje o jego ustawieniach można znaleźć w jego folderze domowym. Jeżeli nie znamy nazw plików, które te informacje przechowują musimy wyświetlić zawartość folderu ~ wraz z ukrytymi plikami.

4. Potoki i przekierowanie danych. Podstawowa kontrola zadań.

Przekierowanie danych to inaczej wysłanie wyniku pracy komendy/polecenia do pliku. Rozpatrzmy poniższy przykład:

\$ Is > /home/kropiak/lista.txt

Wynik działania komendy *Is* zostanie przekierowany (symbolizuje to znak ">") do w/w pliku, tzn., że lista która wyświetliłaby się na ekranie monitora zostanie zapisana do pliku. Jeżeli plik nie istnieje to zostanie utworzony pod warunkiem posiadania odpowiednich uprawnień. Jeżeli plik istnieje to zostanie nadpisany nową zawartością.

Aby dopisać dane do pliku musimy zamienić znak ">" na ">>", więc ponowne wywołanie

\$ Is >> /home/kropiak/lista.txt

spowoduje dopisanie danych do pliku lista.txt.

Czasami jednak wynik działania komendy chcemy przekazać jako dane wejściowe innej komendy i pośrednie tworzenie plików może być niewygodne i zbyt czasochłonne. Z pomocą przychodzi nam mechanizm **potoku** (ang. pipe), który w wierszu poleceń jest oznaczony symbolem "]".

Przykład: *\$ Is –al | wc –I*

Wynik polecenia *Is-al* zostanie przekierowany jako dane wejściowe dla *wc –l* a to polecenie zliczy ilość linii.

Jedną z wad pracy na jednym oknie konsoli jest konieczność uruchamiania poleceń po zakończeniu wykonania innej. Przynajmniej początkującym użytkownikom tak może się wydawać. Otóż istnieje mechanizm, który pozwala w prosty sposób uruchamiać zadania w tle i przełączać się między nimi. Przykład uruchomienia zadania w tle:

\$ nano plik.txt & [1] 7009

Pierwsza linia to wprowadzona przez nas komenda, która wygląda standardowo poza znakiem *\$* na końcu. Ten znak mówi powłoce, że zadanie ma być uruchomione w tle. Linia druga to wynik zwrócony przez powłokę informujący nas o numerze naszego zadania *[1]* oraz numerze identyfikacyjnym (PID) procesu w systemie *7009*. Obie te liczby mogą się przydać. Po wykonaniu powyższej metody efekt jego działania, który normalnie powinien pojawić się w ekranie konsoli nie jest widoczny. Aby przywołać zadanie uruchomione w tle należy wykonać polecenie:

\$ %1

gdzie % informuje powłokę, iż mamy do czynienia z mechanizmem przywołania zadania a **1** to numer zadania do przywołania.

Innym poleceniem, które przywołuje zadanie w tle jest komenda *fg* (ang. foreground). Analogicznie aby zadanie ponownie umieścić w tle używamy *bg* (ang. backgroud). Innym sposobem jest wybranie CTRL + Z podczas pracy jakiegoś zadania (np. w oknie edytora nano) co spowoduje wyświetlenie linii podobnej do:

[2]+ Stopped nano plik.txt

[2] to numer zadania, które ponownie możemy przywrócić.

Żeby samodzielnie zakończyć zadanie w tle możemy skorzystać z komendy:

kill %1 lub *kill 7009*

W pierwszym przypadku podejmujemy próbę zakończenia zadania w tle o numerze 1 a w kolejnym procesu o identyfikatorze 7009.

5. Lista dodatkowych komend, które warto znać. Gdzie szukać pomocy i informacji ?

5.1. Lista komend, którym warto się bliżej przyjrzeć:

- tar
- export
- cat
- gzip
- shutdown
- reboot
- poweroff
- free
- df
- ps
- ifconfig
- rpm
- date
- find
- wget
- cmp
- wc
- du
- df
- whereis

5.2. Gdzie szukać informacji o Ubuntu ?

Oczywiście jeżeli napisze, że pytając "wujka" Google to nie powiem nic odkrywczego. Jednak po pierwsze trzeba szukać umiejętnie a po drugie istnieją strony, od których warto zacząć i w razie wątpliwości lub braku informacji na owych, można szukać dalej:

- <u>http://www.thegeekstuff.com/</u> dużo informacji o wbudowanych narzędziach Linuxa wraz z wieloma przykładami, polecam,
- <u>http://www.ubuntu.com/</u> strona domowa Ubuntu, pierwsze źródło informacji o nowościach i wszystkim co z Ubuntu związane,
- https://help.ubuntu.com/12.04/ubuntu-help/index.html podręcznik dla Ubuntu 12.04 LTS,
- <u>https://wiki.debian.org</u> WIKI dla projektu Debian, z którego Ubuntu się wywodzi,
- <u>http://dug.net.pl</u> polskojęzyczna strona z poradami dla Debiana,
- <u>http://askubuntu.com/</u> popularna strona, która kojarzy pytających i odpowiadających tym razem w temacie Ubuntu,

- <u>http://manpages.ubuntu.com/manpages/precise/pl/man1/</u> strona z zawartością wszystkich stron z narzędzia *man* dla wersji 12.04,
- <u>http://ubuntu.pl/</u> polska strona o Ubuntu,