

Laboratorium nr 5 – systemy operacyjne

Przygotowanie

Utworzenie dwóch plików o rozmiarze 100 MB:

- pusty plik – empty_file

```
(szymon@kali)-[~/Desktop/lab5]
$ dd if=/dev/zero of=empty_file bs=1M count=100
100+0 records in
100+0 records out
104857600 bytes (105 MB, 100 MiB) copied, 0.522663 s, 201 MB/s

(szymon@kali)-[~/Desktop/lab5]
$ ls
empty_file
```

- plik z losowymi wartościami – random_file

```
(szymon@kali)-[~/Desktop/lab5]
$ dd if=/dev/urandom of=random_file bs=1M count=100
100+0 records in
100+0 records out
104857600 bytes (105 MB, 100 MiB) copied, 0.969649 s, 108 MB/s

(szymon@kali)-[~/Desktop/lab5]
$ ls
empty_file random_file
```

Zadanie 1.

Spakowanie plików

```
(szymon@kali)-[~/Desktop/lab5]
$ gzip -k empty_file

(szymon@kali)-[~/Desktop/lab5]
$ gzip -k random_file

(szymon@kali)-[~/Desktop/lab5]
$ ls
duplicated_file empty_file empty_file.gz random_file random_file.gz this_file
```

Polecenie du – mówi jak mocno skompresowano dane

```
(szymon@kali)-[~/Desktop/lab5]
$ du empty_file.gz
100 empty_file.gz

(szymon@kali)-[~/Desktop/lab5]
$ du random_file.gz
102420 random_file.gz
```

Zadanie 2.

Stworzenie dowiązania miękkiego dla pliku empty_file.

```
(szymon@kali)-[~/Desktop/lab5]
$ ln empty_file empty_file_link

(szymon@kali)-[~/Desktop/lab5]
$ ln -s empty_file empty_file_softLink
```

Stworzenie dowiązania twardego i miękkiego dla pliku random_file

```
(szymon@kali)-[~/Desktop/lab5]
$ ln random_file random_file_link

(szymon@kali)-[~/Desktop/lab5]
$ ln -s random_file random_file_softLink
```

```
(szymon@kali)-[~/Desktop/lab5]
$ ls
duplicated_file  empty_file_link  random_file.gz  this_file
empty_file       empty_file_softLink random_file_link
empty_file.gz    random_file      random_file_softLink
```

Zajętość pamięci dla dowiązań polecenie du

```
(szymon@kali)-[~/Desktop/lab5]
$ du empty_file_softLink
0 empty_file_softLink

(szymon@kali)-[~/Desktop/lab5]
$ du random_file_softLink
0 random_file_softLink

(szymon@kali)-[~/Desktop/lab5]
$ du random_file_link
102400 random_file_link

(szymon@kali)-[~/Desktop/lab5]
$ du empty_file_link
102400 empty_file_link
```

Zajętość pamięci, polecenie stat dla pliku empty_file_softLink

```
(szymon@kali)-[~/Desktop/lab5]
$ stat empty_file_softLink
File: empty_file_softLink -> empty_file
Size: 10      Blocks: 0      IO Block: 4096   symbolic link
Device: 8,1 Inode: 1480502   Links: 1
Access: (0777/lrwxrwxrwx)  Uid: ( 1000/   szymon)   Gid: ( 1000/   szymon)
Access: 2024-01-01 23:01:27.954122684 +0100
Modify: 2024-01-01 23:01:24.826175304 +0100
Change: 2024-01-01 23:01:24.826175304 +0100
Birth: 2024-01-01 23:01:24.826175304 +0100
```

Zajętość pamięci, polecenie stat dla pliku empty_file_link

```
(szymon@kali)-[~/Desktop/lab5]
$ stat empty_file_link
File: empty_file_link
Size: 104857600 Blocks: 204800 IO Block: 4096 regular file
Device: 8,1 Inode: 1473708 Links: 2
Access: (0644/-rw-r--r--) Uid: ( 1000/ szymon) Gid: ( 1000/ szymon)
Access: 2024-01-01 22:51:13.823257746 +0100
Modify: 2024-01-01 22:48:08.508040970 +0100
Change: 2024-01-01 23:01:46.953779357 +0100
Birth: 2024-01-01 22:32:23.605867347 +0100
```

Zajętość pamięci, polecenie stat dla pliku random_file_softLink

```
(szymon@kali)-[~/Desktop/lab5]
$ stat random_file_softLink
File: random_file_softLink → random_file
Size: 11 Blocks: 0 IO Block: 4096 symbolic link
Device: 8,1 Inode: 1485548 Links: 1
Access: (0777/lrwxrwxrwx) Uid: ( 1000/ szymon) Gid: ( 1000/ szymon)
Access: 2024-01-01 23:02:42.932554410 +0100
Modify: 2024-01-01 23:02:41.580587398 +0100
Change: 2024-01-01 23:02:41.580587398 +0100
Birth: 2024-01-01 23:02:41.580587398 +0100
```

Zajętość pamięci, polecenie stat dla pliku random_file_link

```
(szymon@kali)-[~/Desktop/lab5]
$ stat random_file_link
File: random_file_link
Size: 104857600 Blocks: 204800 IO Block: 4096 regular file
Device: 8,1 Inode: 1470629 Links: 2
Access: (0644/-rw-r--r--) Uid: ( 1000/ szymon) Gid: ( 1000/ szymon)
Access: 2024-01-01 22:51:21.566830981 +0100
Modify: 2024-01-01 22:48:24.683445484 +0100
Change: 2024-01-01 23:02:31.348831941 +0100
Birth: 2024-01-01 22:48:23.723482182 +0100
```

Zadanie 3.

Na pliku empty_file założyłem system plików ext4 i zmontowałem w systemie plików.

```
(szymon@kali)-[~/Desktop/lab5]
$ mkfs.ext4 empty_file
mke2fs 1.46.6 (1-Feb-2023)
Discarding device blocks: done
Creating filesystem with 102400 1k blocks and 25584 inodes
Filesystem UUID: 0dad2e61-17ac-4759-b9ad-95ab5b1c4d7b
Superblock backups stored on blocks:
 8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done
```

UUID: 0dad2e61-17ac-4759-b9ad-95ab5b1c4d7b

```
(szymon@kali)-[~/Desktop/lab5]
$ dumpe2fs empty_file
dumpe2fs 1.46.6 (1-Feb-2023)
Filesystem volume name: <none>
Last mounted on: <not available>
Filesystem UUID: 0dad2e61-17ac-4759-b9ad-95ab5b1c4d7b
Filesystem magic number: 0xEF53
Filesystem revision #: 1 (dynamic)
Filesystem features: has_journal ext_attr resize_inode dir_index filetype extent 64b
lex_bg sparse_super large_file huge_file dir_nlink extra_isize metadata_csum
Filesystem flags: signed_directory_hash
Default mount options: user_xattr acl
Filesystem state: clean
Errors behavior: Continue
Filesystem OS type: Linux
Inode count: 25584
Block count: 102400
Reserved block count: 5120
Overhead clusters: 12067
Free blocks: 90319
Free inodes: 25573
```

Montowanie/odmontowywanie systemu plików możliwe było przez użytkownika bez uprawnień root

```
(szymon@kali)-[~/Desktop/lab5]
$ sudo vim /etc/fstab
```


Modyfikacja pliku fstab

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# systemd generates mount units based on this file, see systemd.mount(5).
# Please run 'systemctl daemon-reload' after making changes here.
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda1 during installation
UUID=0dad2e61-17ac-4759-b9ad-95ab5b1c4d7b /home/szymon/Desktop/lab5 ext4 rw,user,noauto 0 1
UUID=88411b5d-b94c-4556-a15b-c93e3a256067 / ext4 errors=remount-ro 0 1
# swap was on /dev/sda5 during installation
UUID=1ecf5a26-44d7-49bf-ba7d-c400e10a3e03 none swap sw 0 0
/dev/sr0 /media/cdrom0 udf,iso9660 user,noauto 0 0
```

Przetadowuje

```
(szymon@kali)-[~/Desktop/lab5]
$ sudo vim /etc/fstab

(szymon@kali)-[~/Desktop/lab5]
$ mount empty_file
mount: /home/szymon/Desktop/lab5: failed to setup loop device for empty_file.
mount: (hint) your fstab has been modified, but systemd still uses
the old version; use 'systemctl daemon-reload' to reload.

(szymon@kali)-[~/Desktop/lab5]
$ sudo systemctl daemon-reload
```

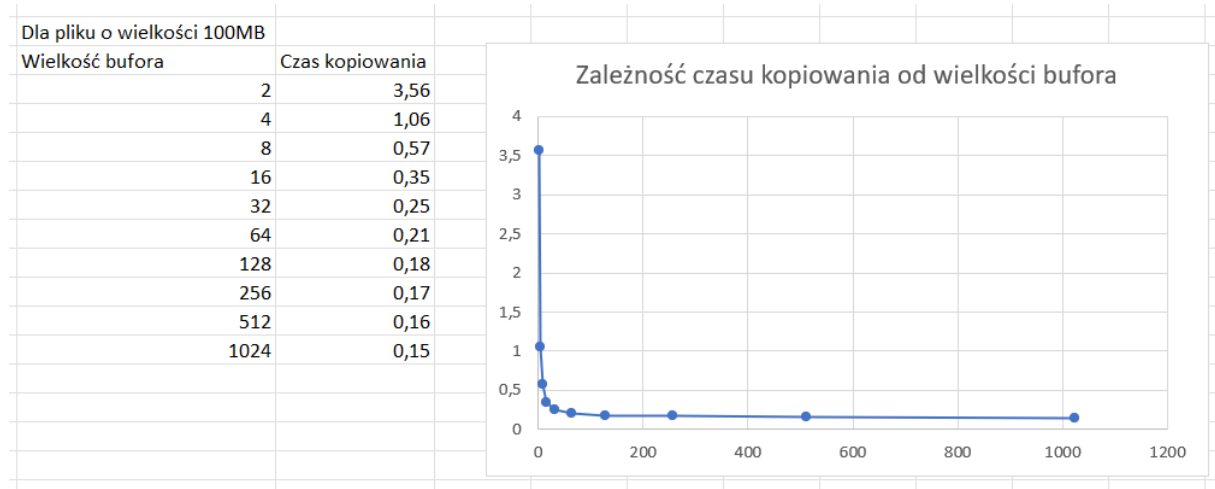
Z opcją -a zadziała

```
(szymon@kali)-[~/Desktop/lab5]
$ mount empty_file
mount: /home/szymon/Desktop/lab5: failed to setup loop device for empty_file.

(szymon@kali)-[~/Desktop/lab5]
$ mount -a empty_file
```

Zadanie 5.

Program w C++ kopiujący zawartość pliku na podstawie parametrów



Zadanie 6.

1.6. Napisz polecenie/skrypt wykrywający wszystkie przypadki zapętłonego linkowania miękkiego.

`./sprawdz-miekkie-linki.sh <katalog>`

Podaj liczbę zapętleń (wypisz je) oraz długość poszczególnych zapętleń.

Brak dowiązań miękkich w podanym przeze mnie katalogu, tj. bashScripts

```
(szymon@kali)-[~]
$ ./sprawdz-miekkie-linki.sh /home/szymon/bashScripts
Brak plików dowiazan miekkich w podanym katalogu.
```

Zadanie 7.

1.7. Napisz polecenie/skrypt wykrywający wszystkie przypadki hardlinkowania.

`./sprawdz-twarde-linki.sh <katalog>`

Podpowiedzi:

- Sprawdzenie, ile dowiązań ma plik (wpis w pliku katalogu): `stat <nazwa pliku>`
- Znalezienie plików o zadanej liczbie dowiązań: `find -links <parametr>`
- Pomiąć rekordy, w których dana konfiguracja występuje jednokrotnie.

```
(szymon@kali)-[~]
$ ./sprawdz-twarde-linki.sh /home/szymon/bashScripts
Brak plikow z twardymi dowiazaniami w podanym katalogu.
```

Dla podanego przeze mnie katalogu, brak plików z twardymi dowiązaniem

Zadanie 1.8

1.8. Napisz polecenie/skrypt wypisujący statystykę plików w zadanym drzewie dla każdej konfiguracji ustawień dostępu.

`./statystyka-uprawnien.sh <katalog>`

W moim katalogu domowym, użyte polecenie, to: `ls -la | grep "^-" | cut -d" " -f1 | sort | uniq -c | sort -rn | awk '{print $1, $2}'`

```
(szymon@kali)-[~]
$ ls -la | grep "^-" | cut -d" " -f1 | sort | uniq -c | sort -rn | awk '{print $1, $2}'
20 -rw-r--r--
8 -rw----- chatCyber.sh
4 -rwxr-xr-x
2 -rwxrw-rw-
```

