

Disciplina: Sistemas Operacionais

Discente: Paulo Guilherme Silva de Araújo - Engenharia de Computação

Atividade Prática - Sistema de Arquivos

1) No terminal, vá até o diretório /proc e liste seu conteúdo (ls -l). Observe que os subdiretórios correspondem aos PIDs dos processos correntes (execute ps -lax e verifique isso).

```
paulo@paulo-VirtualBox: /proc$ ls -l
total 0
dr-xr-xr-x 9 root      root      0 ago 20 20:58 1
dr-xr-xr-x 9 root      root      0 ago 20 20:59 10
dr-xr-xr-x 9 paulo     paulo     0 ago 20 21:01 1001
dr-xr-xr-x 9 root      root      0 ago 20 20:59 101
dr-xr-xr-x 9 rtkit     rtkit     0 ago 20 21:01 1020
dr-xr-xr-x 9 root      root      0 ago 20 20:59 109
dr-xr-xr-x 9 root      root      0 ago 20 20:59 11
dr-xr-xr-x 9 root      root      0 ago 20 20:59 112
dr-xr-xr-x 9 root      root      0 ago 20 20:59 113
dr-xr-xr-x 9 root      root      0 ago 20 20:59 118
dr-xr-xr-x 9 root      root      0 ago 20 20:59 12
dr-xr-xr-x 9 root      root      0 ago 20 20:59 13
dr-xr-xr-x 9 root      root      0 ago 20 20:59 14
dr-xr-xr-x 9 root      root      0 ago 20 21:01 1491
dr-xr-xr-x 9 root      root      0 ago 20 20:59 15
dr-xr-xr-x 9 root      root      0 ago 20 20:59 157
dr-xr-xr-x 9 root      root      0 ago 20 21:01 1573
dr-xr-xr-x 9 root      root      0 ago 20 20:59 158
dr-xr-xr-x 9 root      root      0 ago 20 20:59 16
dr-xr-xr-x 9 root      root      0 ago 20 20:59 17
dr-xr-xr-x 9 paulo     paulo     0 ago 20 21:03 1773
dr-xr-xr-x 9 root      paulo     0 ago 20 21:01 1774
dr-xr-xr-x 9 root      root      0 ago 20 20:59 18
dr-xr-xr-x 9 root      root      0 ago 20 20:59 183
dr-xr-xr-x 9 root      root      0 ago 20 20:59 184
dr-xr-xr-x 9 colord    colord    0 ago 20 21:02 1845
dr-xr-xr-x 9 root      root      0 ago 20 20:59 19
dr-xr-xr-x 9 root      root      0 ago 20 20:59 2
dr-xr-xr-x 9 root      root      0 ago 20 20:59 20
dr-xr-xr-x 9 root      root      0 ago 20 20:59 21
dr-xr-xr-x 9 root      root      0 ago 20 20:59 22
```

```
paulo@paulo-VirtualBox: /proc$ ps -lax
F  UID      PID     PPID  PRI  NI   VSZ   RSS WCHAN    STAT TTY      TIME COMMAND
1  0          1         0  20   0 166456 11648 -          Ss   ?        0:01 /sbin/init splash
1  0          2         0  20   0      0      0 -          S    ?        0:00 [kthreadd]
1  0          3         2  0 -20   0      0      0 -          I<   ?        0:00 [rcu_gp]
1  0          4         2  0 -20   0      0      0 -          I<   ?        0:00 [rcu_par_gp]
1  0          5         2  0 -20   0      0      0 -          I<   ?        0:00 [netns]
1  0          7         2  0 -20   0      0      0 -          I<   ?        0:00 [kworker/0:0H-events_highpri]
1  0         10         2  0 -20   0      0      0 -          I<   ?        0:00 [mm_percpu_wq]
1  0         11         2  20   0      0      0 -          S    ?        0:00 [rcu_tasks_rude_]
1  0         12         2  20   0      0      0 -          S    ?        0:00 [rcu_tasks_trace]
1  0         13         2  20   0      0      0 -          S    ?        0:00 [ksoftirqd/0]
1  0         14         2  20   0      0      0 -          I    ?        0:01 [rcu_sched]
1  0         15         2 -100  -      0      0 -          S    ?        0:00 [migration/0]
1  0         16         2 -51  -      0      0 -          S    ?        0:00 [idle_inject/0]
1  0         17         2  20   0      0      0 -          S    ?        0:00 [cpuhp/0]
5  0         18         2  20   0      0      0 -          S    ?        0:00 [kdevtmpfs]
1  0         19         2  0 -20   0      0      0 -          I<   ?        0:00 [inet_frag_wq]
1  0         20         2  20   0      0      0 -          S    ?        0:00 [kauditd]
1  0         21         2  20   0      0      0 -          S    ?        0:00 [khungtaskd]
1  0         22         2  20   0      0      0 -          S    ?        0:00 [oom_reaper]
1  0         23         2  0 -20   0      0      0 -          I<   ?        0:00 [writeback]
1  0         24         2  20   0      0      0 -          S    ?        0:00 [kcompactd0]
1  0         25         2  25   5      0      0 -          SN   ?        0:00 [ksmd]
1  0         26         2  39  19      0      0 -          SN   ?        0:00 [khugepaged]
1  0         72         2  0 -20   0      0      0 -          I<   ?        0:00 [kintegrityd]
1  0         73         2  0 -20   0      0      0 -          I<   ?        0:00 [kblockd]
1  0         74         2  0 -20   0      0      0 -          I<   ?        0:00 [blkcg_punt_bio]
1  0         75         2  0 -20   0      0      0 -          I<   ?        0:00 [tpm_dev_wq]
1  0         76         2  0 -20   0      0      0 -          I<   ?        0:00 [ata_sff]
1  0         77         2  0 -20   0      0      0 -          I<   ?        0:00 [md]
1  0         78         2  0 -20   0      0      0 -          I<   ?        0:00 [edac-poller]
1  0         79         2  0 -20   0      0      0 -          I<   ?        0:00 [devfreq_wq]
```

2) Lembrando que o bash é o seu interpretador de comandos, você pode verificar o PID do bash executando o comando ps. De posse do PID do seu bash, entre no subdiretório cujo nome seja o PID do seu bash. Ali você encontrará várias informações sobre este processo. Consulte algumas dessas informações para o seu bash:

```
paulo@paulo-VirtualBox:/proc$ more /proc/2964/cmdline
bash
```

```
paulo@paulo-VirtualBox:/proc$ more /proc/2964/maps
55fd29fcd000-55fd29ffc000 r--p 00000000 08:03 524397 /usr/bin/bash
55fd29ffc000-55fd2a0db000 r-xp 0002f000 08:03 524397 /usr/bin/bash
55fd2a0db000-55fd2a115000 r--p 0010e000 08:03 524397 /usr/bin/bash
55fd2a116000-55fd2a11a000 r--p 00148000 08:03 524397 /usr/bin/bash
55fd2a11a000-55fd2a123000 rw-p 0014c000 08:03 524397 /usr/bin/bash
55fd2a123000-55fd2a12e000 rw-p 00000000 00:00 0
55fd2b32b000-55fd2b44f000 rw-p 00000000 00:00 0
7f3279e9c000-7f327ac83000 r--p 00000000 08:03 531173 [heap]
7f327ac83000-7f327ac86000 rw-p 00000000 00:00 0 /usr/lib/locale/locale-archive
7f327ac86000-7f327acae000 r--p 00000000 08:03 531968 /usr/lib/x86_64-linux-gnu/libc.so.6
7f327acae000-7f327ae43000 r-xp 00028000 08:03 531968 /usr/lib/x86_64-linux-gnu/libc.so.6
7f327ae43000-7f327ae9b000 r--p 001bd000 08:03 531968 /usr/lib/x86_64-linux-gnu/libc.so.6
7f327ae9b000-7f327ae9f000 r--p 00214000 08:03 531968 /usr/lib/x86_64-linux-gnu/libc.so.6
7f327ae9f000-7f327aea1000 rw-p 00218000 08:03 531968 /usr/lib/x86_64-linux-gnu/libc.so.6
7f327aea1000-7f327aeae000 rw-p 00000000 00:00 0
7f327aeae000-7f327aebc000 r--p 00000000 08:03 533097 /usr/lib/x86_64-linux-gnu/libtinfo.so.6.3
7f327aebc000-7f327aecd000 r-xp 0000e000 08:03 533097 /usr/lib/x86_64-linux-gnu/libtinfo.so.6.3
7f327aecd000-7f327aedd000 r--p 0001f000 08:03 533097 /usr/lib/x86_64-linux-gnu/libtinfo.so.6.3
7f327aedd000-7f327aede000 r--p 0002c000 08:03 533097 /usr/lib/x86_64-linux-gnu/libtinfo.so.6.3
7f327aede000-7f327aee0000 rw-p 00030000 08:03 533097 /usr/lib/x86_64-linux-gnu/libtinfo.so.6.3
7f327aee0000-7f327aef0000 r--s 00000000 08:03 922393 /usr/lib/x86_64-linux-gnu/gconv/gconv-modules.cache
7f327aef0000-7f327aef2000 rw-p 00000000 00:00 0
7f327aef2000-7f327aef4000 r--p 00000000 08:03 531631 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7f327aef4000-7f327af1e000 r-xp 00002000 08:03 531631 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7f327af1e000-7f327af29000 r--p 0002c000 08:03 531631 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7f327af2a000-7f327af2c000 r--p 00037000 08:03 531631 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7f327af2c000-7f327af2e000 rw-p 00039000 08:03 531631 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7ffe60c28000-7ffe60c49000 rw-p 00000000 00:00 0 [stack]
7ffe60c5d000-7ffe60c61000 r--p 00000000 00:00 0 [vvar]
7ffe60c61000-7ffe60c63000 r-xp 00000000 00:00 0 [vdso]
fffffffff600000-fffffffff601000 --xp 00000000 00:00 0 [vsyscall]
```

```
paulo@paulo-VirtualBox:/proc$ more /proc/2964/stat
2964 (bash) S 2930 2964 2964 34816 3238 4194304 1288 2409 3 8 3 0 3 3 20 0 1 0 38686 20131840 1267 18446744073709551615 94545819713536 9454582
0626701 140730521910320 0 0 0 65536 3686404 1266761467 1 0 0 17 0 0 0 0 0 94545820871344 94545820919376 94545839828992 140730521914037 14073
0521914042 140730521914042 140730521915374 0
paulo@paulo-VirtualBox:/proc$
```

3) Você consegue encontrar o executável do seu SO? Execute ls -l no diretório raiz. Observe que aparece algo assim:

```
paulo@paulo-VirtualBox:/$ ls -l
total 1751120
lrwxrwxrwx 1 root root 7 ago 20 19:19 bin -> usr/bin
drwxr-xr-x 4 root root 4096 ago 20 19:42 boot
```

4) No terminal, vá até o diretório /dev e liste seu conteúdo (ls -l). Observe que o início de cada linha impressa indica o tipo de arquivo (c, b ou d... eventualmente algum l). Um fato curioso sobre os dispositivos está relacionado a existência de quatro arquivos na pasta /dev: full, zero, random e o null. Estes arquivos não correspondem a dispositivos de fato. Você saberia dizer a função de cada um deles?

O /dev/null, ou dispositivo nulo, é um arquivo especial que descarta toda informação escrita nele e que não retorna qualquer informação para um processo que o leia

/dev/zero é um arquivo especial que fornece quantos caracteres nulos (o NULL da tabela ASCII, 0x00; e não o caractere "dígito zero", "0", 0x30) forem lidos dele

/dev/random, é um arquivo especial que serve como gerador de números pseudo-aleatórios

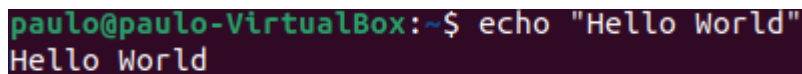
o /dev/full é um arquivo especial que sempre retorna o mesmo código de erro ao ser acessado para escrita: ENOSPC – o que quer dizer “Não há espaço disponível no dispositivo”.

5) No terminal, digite:

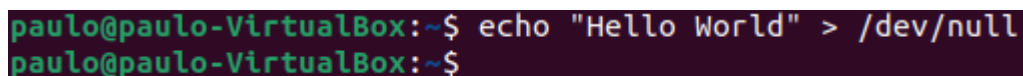
```
$ echo "Hello World"
```

e depois

```
$ echo "Hello World" > /dev/null
```



```
paulo@paulo-VirtualBox:~$ echo "Hello World"
Hello World
```



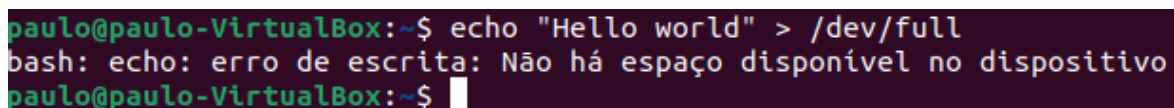
```
paulo@paulo-VirtualBox:~$ echo "Hello World" > /dev/null
paulo@paulo-VirtualBox:~$
```

O arquivo null descartou toda a informação escrita nele e não retornou nada.

6) No terminal, digite o comando abaixo e observe o resultado.

```
$ echo "Hello world" > /dev/full
```

De forma análoga, você consegue dizer o que está acontecendo? Apresente um printscreen com o resultado do comando.



```
paulo@paulo-VirtualBox:~$ echo "Hello world" > /dev/full
bash: echo: erro de escrita: Não há espaço disponível no dispositivo
paulo@paulo-VirtualBox:~$
```

O arquivo full retorna sempre a mesma mensagem: “Não há espaço disponível no dispositivo

7) No terminal, vá até o diretório HOME (cd ~) e digite \$ ls -lai .

Na coluna mais à esquerda, você encontra os números do inode de cada arquivo.

Agora faça a mesma coisa de dentro do diretório raiz. Alguém com o inode 1?

```

paulo@paulo-VirtualBox:~$ cd /
paulo@paulo-VirtualBox:/$ ls -lai
total 1751128
  2 drwxr-xr-x  20 root root    4096 ago 20 19:31 .
  2 drwxr-xr-x  20 root root    4096 ago 20 19:31 ..
 13 lrwxrwxrwx   1 root root      7 ago 20 19:19 bin -> usr/bin
524289 drwxr-xr-x   4 root root    4096 ago 20 19:42 boot
790115 drwxrwxr-x   2 root root    4096 ago 20 19:31 cdrom
   1 drwxr-xr-x  19 root root    4060 ago 20 21:01 dev
262145 drwxr-xr-x 129 root root   12288 ago 20 20:57 etc
131073 drwxr-xr-x   3 root root    4096 ago 20 19:32 home
  14 lrwxrwxrwx   1 root root      7 ago 20 19:19 lib -> usr/lib
  15 lrwxrwxrwx   1 root root      9 ago 20 19:19 lib32 -> usr/lib32
  16 lrwxrwxrwx   1 root root      9 ago 20 19:19 lib64 -> usr/lib64
  17 lrwxrwxrwx   1 root root     10 ago 20 19:19 libx32 -> usr/libx32
  11 drwx-----   2 root root   16384 ago 20 19:18 lost+found
655361 drwxr-xr-x   3 root root    4096 ago 20 20:53 media
393217 drwxr-xr-x   2 root root    4096 ago  9 08:48 mnt
655362 drwxr-xr-x   3 root root    4096 ago 20 20:55 opt
   1 dr-xr-xr-x 246 root root      0 ago 20 20:58 proc
131074 drwx-----   4 root root    4096 ago 20 20:46 root
   1 drwxr-xr-x  32 root root     900 ago 20 21:04 run
  18 lrwxrwxrwx   1 root root      8 ago 20 19:19 sbin -> usr/sbin
393218 drwxr-xr-x  11 root root    4096 ago  9 08:55 snap
262148 drwxr-xr-x   2 root root    4096 ago  9 08:48 srv
  12 -rw-----   1 root root 1793064 ago 20 19:19 swapfile
   1 dr-xr-xr-x  13 root root      0 ago 20 20:58 sys
393219 drwxrwxrwt  18 root root    4096 ago 20 21:14 tmp
524291 drwxr-xr-x  14 root root    4096 ago  9 08:48 usr
655363 drwxr-xr-x  14 root root    4096 ago  9 08:54 var

```

8) No terminal, digite \$ stat NOME_DO_ARQUIVO. Faça isso para diferentes tipos de arquivos (um arquivo de texto, um dispositivo no /dev, um diretório). Observe os campos “Blocos” e “bloco de E/S” (Obs.: podem aparecer em inglês).

```

paulo@paulo-VirtualBox:~/Documentos$ stat texto
Arquivo: texto
  Tamanho: 7          Blocos: 8          bloco de E/S: 4096   arquivo comum
Dispositivo: 803h/2051d Inode: 298724      Links: 1
Acesso: (0664/-rw-rw-r--) Uid: ( 1000/ paulo)  Gid: ( 1000/ paulo)
Acesso: 2022-08-20 21:43:43.635437181 -0300
Modificação: 2022-08-20 21:43:43.795439283 -0300
  Alteração: 2022-08-20 21:43:43.795439283 -0300
  Criação: 2022-08-20 21:43:43.635437181 -0300
paulo@paulo-VirtualBox:~/Documentos$

```

```

paulo@paulo-VirtualBox:/dev$ stat log
Arquivo: log -> /run/systemd/journal/dev-log
  Tamanho: 28          Blocos: 0          bloco de E/S: 4096   link simbólico
Dispositivo: 5h/5d     Inode: 383        Links: 1
Acesso: (0777/lrwxrwxrwx) Uid: (   0/ root)  Gid: (   0/ root)
Acesso: 2022-08-20 20:59:18.547999258 -0300
Modificação: 2022-08-20 20:59:07.819999636 -0300
  Alteração: 2022-08-20 20:59:07.819999636 -0300
  Criação: -
paulo@paulo-VirtualBox:/dev$

```

```
paulo@paulo-VirtualBox:~$ stat Documentos/
Arquivo: Documentos/
  Tamanho: 4096      Blocos: 8      bloco de E/S: 4096  diretório
Dispositivo: 803h/2051d Inode: 303205    Links: 2
Acesso: (0755/drwxr-xr-x)  Uid: ( 1000/   paulo)   Gid: ( 1000/   paulo)
Acesso: 2022-08-20 21:44:23.419952308 -0300
Modificação: 2022-08-20 21:43:43.635437181 -0300
  Alteração: 2022-08-20 21:43:43.635437181 -0300
  Criação: 2022-08-20 20:46:05.889299350 -0300
paulo@paulo-VirtualBox:~$
```

9) No terminal, crie um link simbólico usando `ln -s` e depois verifique o resultado usando `ls -l`.

```
paulo@paulo-VirtualBox:~/Imagens$ ls -l
total 4
drwxrwxr-x 2 paulo paulo 4096 ago 20 21:44 'Capturas de tela'
lrwxrwxrwx 1 paulo paulo  22 ago 20 21:49 Documentos -> /home/paulo/Documents
paulo@paulo-VirtualBox:~/Imagens$
```

10) No terminal, crie um hard link usando `ln` e depois verifique o resultado usando `ls -l`.

Tente acessar o arquivo (more se for um arquivo ASCII) via o link criado.

É mostrado o conteúdo do arquivo original

```
paulo@paulo-VirtualBox:~/Imagens$ cat texto
texto1
paulo@paulo-VirtualBox:~/Imagens$
```

11) Você consegue criar um hardlink para um diretório?

Não é possível

```
paulo@paulo-VirtualBox:~/Imagens$ ln /home/paulo/Downloads /home/paulo/Imagens
ln: /home/paulo/Downloads: não são permitidos links absolutos para diretórios
```