Домашнее задание к занятию "3.3 Операционные системы лекция 1"

1. Вызовем команду **strace** на **/bin/bash -c 'cd /tmp'** вывод с терминала:

hachiko@hachik-O:~\$ strace /bin/bash -c 'cd /tmp'

найдем системных вызовов который относится именно к **cd** это **chdir("/tmp")** скриншот с терминала

```
= 0
ioctl(2, TIOCGPGRP, [4088])
rt_sigaction(SIGCHLD, {sa_handler=0x559bff55cd40, sa_
2fa40, 8) = 0
prlimit64(0, RLIMIT_NPROC, NULL, {rlim_cur=31483, rlim
rt_sigprocmask(SIG_BLOCK, NULL, [], 8)
                                         = 0
getpeername(0, 0x7ffe47581c80, [16])
                                         = -1 ENOTSOCK
rt_sigprocmask(SIG_BLOCK, NULL, [], 8)
                                        = 0
newfstatat(AT_FDCWD, "/tmp", {st_mode=S_IFDIR|S_ISVTX
chdir("/tmp")
                                         = 0
rt\_sigprocmask(SIG\_BLOCK, [CHLD], [], 8) = 0
rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0
exit_group(0)
+++ exited with 0 +++
```

2. Вывод с терминала:

hachiko@hachik-O:~\$ file /dev/tty

/dev/tty: character special (5/0)

hachiko@hachik-O:~\$ file /dev/sda

/dev/sda: block special (8/0)

hachiko@hachik-O:~\$ file /bin/bash

/bin/bash: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamically

linked, interpreter /lib64/ld-linux-x86-64.so.2,

BuildID[sha1]=2a9f157890930ced4c3ad0e74fc1b1b84aad71e6, for GNU/Linux 3.2.0, stripped

Запустим strace на file /dev/tty

Вывод с терминала:

```
hachiko@hachik-O:~$ strace file /dev/tty 2>&1 | grep open openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3 openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libmagic.so.1", O_RDONLY|O_CLOEXEC) = 3 openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3 openat(AT_FDCWD, "/lib/x86_64-linux-gnu/liblzma.so.5", O_RDONLY|O_CLOEXEC) = 3
```

```
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libbz2.so.1.0",
O_RDONLY|O_CLOEXEC) = 3
openat(AT FDCWD, "/lib/x86 64-linux-gnu/libz.so.1", O RDONLY|O CLOEXEC) = 3
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libpthread.so.0",
O RDONLY|O CLOEXEC) = 3
openat(AT_FDCWD, "/usr/lib/locale/locale-archive", O_RDONLY|O_CLOEXEC) = 3
openat(AT_FDCWD, "/etc/magic.mgc", O_RDONLY) = -1 ENOENT (Нет такого
файла или каталога)
openat(AT FDCWD, "/etc/magic", O RDONLY) = 3
openat(AT_FDCWD, "/usr/share/misc/magic.mgc", O_RDONLY) = 3
openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/gconv/gconv-modules.cache",
O RDONLY) = 3
hachiko@hachik-O:~$ file /usr/share/misc/magic.mgc
/usr/share/misc/magic.mgc: symbolic link to ../../lib/file/magic.mgc
hachiko@hachik-O:~$ file ../../lib/file/magic.mgc
../../lib/file/magic.mgc: magic binary file for file(1) cmd (version 14) (little endian)
```

файл расположен: /usr/share/misc/magic.mgc и /usr/lib/file/magic.mgc

3. Запустим пинг в запись в файл

Вывод с терминала:

root@vagrant:/home/vagrant# ping 8.8.8.8 > log_ping.txt

۸Z

[1]+ Stopped ping 8.8.8.8 > log ping.txt

root@vagrant:/home/vagrant# bg

[1]+ ping 8.8.8.8 > log_ping.txt &

Запустим root@vagrant:/home/vagrant# ps aux

Скриншот

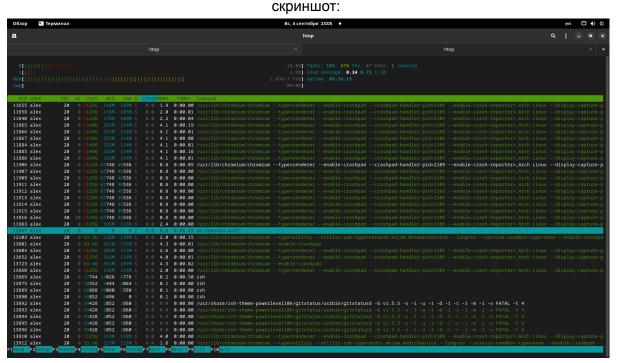
```
| State | Stat
```

Удалим файл log_ping.txt и запустим команду lsof - р 1339 (PID процесса ping) Командой **echo 'ping' > /proc/1339/fd/4** перенаправляем вывод и обнуляем удаленный файл.

скриншот

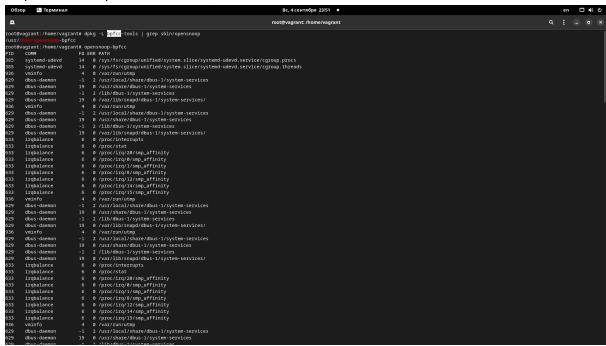
```
root@vagrant:/home/vagrant# rm log_ping.txt
root@vagrant:/home/vagrant# lsof -p 1339
COMMAND PID USER FD
                         TYPE DEVICE SIZE/OFF
                                                   NODE NAME
ping
        1339 root
                   cwd
                           DIR 253,0
                                          4096 1314700 /home/vagrant
ping
        1339 root
                   rtd
                           DIR 253,0
                                          4096
                                                      2 /
ping
        1339 root
                           REG 253,0
                                         72776 1050875 /usr/bin/ping
                   txt
                                253,0 3035952 1048854 /usr/lib/locale/locale-archive
ping
        1339 root
                   mem
                           REG
ping
        1339 root
                           REG 253,0
                                       137584 1054865 /usr/lib/x86_64-linux-gnu/libgpg-error.so.0.28.0
                   mem
ping
        1339 root
                   mem
                           REG 253,0 2029592 1049479 /usr/lib/x86_64-linux-gnu/libc-2.31.so
                                253,0 101352 1050126 /usr/lib/x86_64-linux-gnu/libresolv-2.31.so 253,0 1168056 1054855 /usr/lib/x86_64-linux-gnu/libgcrypt.so.20.2.5
                           REG 253,0
ping
        1339 root
                   mem
ping
        1339 root
                           REG
                   mem
ping
                                253,0
                                        31120 1054811 /usr/lib/x86_64-linux-gnu/libcap.so.2.32
        1339 root
                           REG
                   mem
        1339 root
                           REG
                                253,0
                                         191504 1049280 /usr/lib/x86_64-linux-gnu/ld-2.31.so
ping
                           CHR
                                136,0
                                           0t0
                                                      3 /dev/pts/0
ping
        1339 root
                     0u
ping
        1339 root
                      1w
                           REG
                                253,0
                                           6881 1311812 /home/vagrant/log_ping.txt (deleted)
ping
        1339 root
                           CHR
                                            0t0
                                                    3 /dev/pts/0
                      2u
                                136,0
ping
                                                  28771 00000000:0001->00000000:0000
        1339 root
                          icmp
                                  0,9
ping
        1339 root
                      4u sock
                                            0t0
                                                  28772 protocol: PINGv6
root@vagrant:/home/vagrant# echo 'ping' >/proc/1339/fd/4
```

4. Создадим зомби процесс, делал по инструкции https://stackoverflow.com/questions/25172425/create-zombie-process



Зомби-процессы освобождает все свои ресурсы, но остается запись в таблице процессов

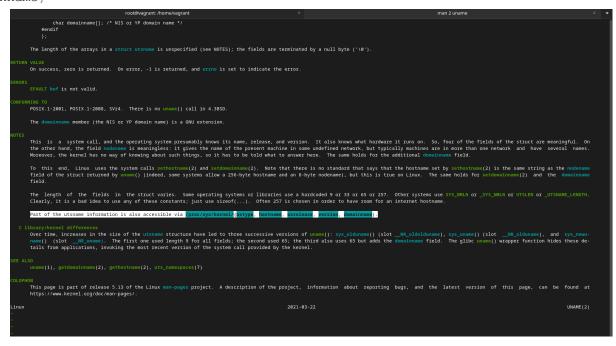
5. Скриншот с терминала с выводом команды



6.

системный вызов uname()

СКРИНШОТ mana CTPOKa /proc/sys/kernel/{ostype, hostname, osrelease, version, domainname}



- 7. ; позволяет запускать несколько команд за один раз, и выполнение команды происходит последовательно.
 - **&&** оператор будет выполнять вторую команду только в том случае, если первая команда выполнена успешно.
 - Использовать **set -e** не имеет смысла, т.к. при ошибке выполнение команд прекратиться.

- 8. Разберем опции set -euxo pipefail:
 - **-е** прекращает выполнение скрипта если команда завершилась ошибкой, выводит в stderr строку с ошибкой
 - $-\mathbf{u}$ прекращает выполнение скрипта, если встретилась несуществующая переменная
 - **-х** выводит выполняемые команды в stdout перед выполнением
 - **-о pipefail** прекращает выполнение скрипта, даже если одна из частей пайпа завершилась ошибкой

При запуске скрипт получается безопасным, происходит автоматическая обработка ошибок.

9. Вывод с терминала:

hachiko@hachik-O:~\$ ps -o stat

STAT

Ss

R+

Ss - неактивные процессы

R+ - выполняющиеся в группе приоритетных

дополнительные к заглавной букве - это дополнительные значения состояния процесса:

- < высокий приоритет
- **N** низкий приоритет
- L имеет страницы, заблокированные в памяти
- **s** является лидером сеанса
- I является многопоточным
- + находится в группе приоритетных процессов