



Item A

Aplicações escolhidas:

- cp (copiar)
- mv (mover)

strace_cp

```
[00007fc3f98c6fb7] execve("/usr/bin/cp", ["cp"], 0x7fffe5699278 /* 65 vars */) = 0
[00007fe0bcd24639] brk(NULL) = 0x1807000
[00007fe0bcd2540a] access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)
[00007fe0bcd25357] open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
[00007fe0bcd252e2] fstat(3, {st_mode=S_IFREG|0644, st_size=309556, ...}) = 0
[00007fe0bcd2550a] mmap(NULL, 309556, PROT_READ, MAP_PRIVATE, 3, 0) =
0x7fe0bcee3000
[00007fe0bcd254b7] close(3) = 0
[00007fe0bcd25357] open("/usr/lib/libacl.so.1", O_RDONLY|O_CLOEXEC) = 3
[00007fe0bcd25377] read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\34\0\0\0\0\0"...,
832) = 832
[00007fe0bcd252e2] fstat(3, {st_mode=S_IFREG|0644, st_size=34736, ...}) = 0
[00007fe0bcd2550a] mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|
MAP_ANONYMOUS, -1, 0) = 0x7fe0bcee1000
[00007fe0bcd2550a] mmap(NULL, 2130080, PROT_READ|PROT_EXEC, MAP_PRIVATE|
MAP_DENYWRITE, 3, 0) = 0x7fe0bcb03000
[00007fe0bcd255a7] mprotect(0x7fe0bcb0a000, 2097152, PROT_NONE) = 0
[00007fe0bcd2550a] mmap(0x7fe0bcd0a000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|
MAP_FIXED|MAP_DENYWRITE, 3, 0x7000) = 0x7fe0bcd0a000
[00007fe0bcd254b7] close(3) = 0
[00007fe0bcd25357] open("/usr/lib/libattr.so.1", O_RDONLY|O_CLOEXEC) = 3
[00007fe0bcd25377] read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\3\0>\0\1\0\0\0200\21\0\0\0\0\0"...,
832) = 832
[00007fe0bcd252e2] fstat(3, {st_mode=S_IFREG|0755, st_size=18248, ...}) = 0
[00007fe0bcd2550a] mmap(NULL, 2113560, PROT_READ|PROT_EXEC, MAP_PRIVATE|
MAP_DENYWRITE, 3, 0) = 0x7fe0bc8fe000
[00007fe0bcd255a7] mprotect(0x7fe0bc902000, 2093056, PROT_NONE) = 0
[00007fe0bcd2550a] mmap(0x7fe0bcb01000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x3000) = 0x7fe0bcb01000
[00007fe0bcd254b7] close(3) = 0
[00007fe0bcd25357] open("/usr/lib/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
[00007fe0bcd25377] read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\3\0>\0\1\0\0\0340\5\2\0\0\0\0"...,
832) = 832
[00007fe0bcd252e2] fstat(3, {st_mode=S_IFREG|0755, st_size=1985472, ...}) = 0
[00007fe0bcd2550a] mmap(NULL, 3823824, PROT_READ|PROT_EXEC, MAP_PRIVATE|
```

```

MAP_DENYWRITE, 3, 0) = 0x7fe0bc558000
[00007fe0bcd255a7] mprotect(0x7fe0bc6f5000, 2093056, PROT_NONE) = 0
[00007fe0bcd2550a] mmap(0x7fe0bc8f4000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x19c000) = 0x7fe0bc8f4000
[00007fe0bcd2550a] mmap(0x7fe0bc8fa000, 14544, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7fe0bc8fa000
[00007fe0bcd254b7] close(3) = 0
[00007fe0bcd2550a] mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|
MAP_ANONYMOUS, -1, 0) = 0x7fe0bc000000
[00007fe0bcd0cc55] arch_prctl(ARCH_SET_FS, 0x7fe0bc000000) = 0
[00007fe0bcd255a7] mprotect(0x7fe0bc8f4000, 16384, PROT_READ) = 0
[00007fe0bcd255a7] mprotect(0x7fe0bcb01000, 4096, PROT_READ) = 0
[00007fe0bcd255a7] mprotect(0x7fe0bcd0a000, 4096, PROT_READ) = 0
[00007fe0bcd255a7] mprotect(0x61c000, 4096, PROT_READ) = 0
[00007fe0bcd255a7] mprotect(0x7fe0bcf2f000, 4096, PROT_READ) = 0
[00007fe0bcd25587] munmap(0x7fe0bcee3000, 309556) = 0
[00007fe0bc63d799] brk(NULL) = 0x1807000
[00007fe0bc63d799] brk(0x1828000) = 0x1828000
[00007fe0bc5833c3] open("/usr/lib/locale/locale-archive", O_RDONLY|O_CLOEXEC) = 3
[00007fe0bc6374e2] fstat(3, {st_mode=S_IFREG|0644, st_size=1674704, ...}) = 0
[00007fe0bc64123a] mmap(NULL, 1674704, PROT_READ, MAP_PRIVATE, 3, 0) =
0x7fe0bcd45000
[00007fe0bc583487] close(3) = 0
[00007fe0bc614997] geteuid() = 1000
[00007fe0bc5cd4ea] open("/usr/share/locale/locale.alias", O_RDONLY|O_CLOEXEC) = 3
[00007fe0bc6374e2] fstat(3, {st_mode=S_IFREG|0644, st_size=2997, ...}) = 0
[00007fe0bc5ccf4c] read(3, "# Locale name alias data base.\n#...", 4096) = 2997
[00007fe0bc5ccf4c] read(3, "", 4096) = 0
[00007fe0bc5cbb0b] close(3) = 0
[00007fe0bc586dc1] open("/usr/share/locale/en_US.utf8/LC_MESSAGES/coreutils.mo",
O_RDONLY) = -1 ENOENT (No such file or directory)
[00007fe0bc586dc1] open("/usr/share/locale/en_US/LC_MESSAGES/coreutils.mo", O_RDONLY)
= -1 ENOENT (No such file or directory)
[00007fe0bc586dc1] open("/usr/share/locale/en.utf8/LC_MESSAGES/coreutils.mo", O_RDONLY)
= -1 ENOENT (No such file or directory)
[00007fe0bc586dc1] open("/usr/share/locale/en/LC_MESSAGES/coreutils.mo", O_RDONLY) = -1
ENOENT (No such file or directory)
[00007fe0bc637bf0] write(2, "cp: ", 4) = 4
[00007fe0bc637bf0] write(2, "missing file operand", 20) = 20
[00007fe0bc637bf0] write(2, "\n", 1) = 1
[00007fe0bc637bf0] write(2, "Try 'cp --help' for more informa...", 38) = 38
[00007fe0bc637c4d] lseek(0, 0, SEEK_CUR) = -1 ESPIPE (Illegal seek)
[00007fe0bc5cbb0b] close(0) = 0
[00007fe0bc5cbb0b] close(1) = 0
[00007fe0bc5cbb0b] close(2) = 0
[00007fe0bc613f88] exit_group(1) = ?
[????????????????] +++ exited with 1 +++

```

strace_cp

```
[00007fd52863efb7] execve("/usr/bin/mv", ["mv"], 0x7ffd2cb306e8 /* 65 vars */) = 0
[00007f61d2c39639] brk(NULL) = 0x687000
[00007f61d2c3a40a] access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)
[00007f61d2c3a357] open("/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
[00007f61d2c3a2e2] fstat(3, {st_mode=S_IFREG|0644, st_size=309556, ...}) = 0
[00007f61d2c3a50a] mmap(NULL, 309556, PROT_READ, MAP_PRIVATE, 3, 0) =
0x7f61d2df8000
[00007f61d2c3a4b7] close(3) = 0
[00007f61d2c3a357] open("/usr/lib/libacl.so.1", O_RDONLY|O_CLOEXEC) = 3
[00007f61d2c3a377] read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\34\0\0\0\0\0"...,
832) = 832
[00007f61d2c3a2e2] fstat(3, {st_mode=S_IFREG|0644, st_size=34736, ...}) = 0
[00007f61d2c3a50a] mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|
MAP_ANONYMOUS, -1, 0) = 0x7f61d2df6000
[00007f61d2c3a50a] mmap(NULL, 2130080, PROT_READ|PROT_EXEC, MAP_PRIVATE|
MAP_DENYWRITE, 3, 0) = 0x7f61d2a18000
[00007f61d2c3a5a7] mprotect(0x7f61d2a1f000, 2097152, PROT_NONE) = 0
[00007f61d2c3a50a] mmap(0x7f61d2c1f000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|
MAP_FIXED|MAP_DENYWRITE, 3, 0x7000) = 0x7f61d2c1f000
[00007f61d2c3a4b7] close(3) = 0
[00007f61d2c3a357] open("/usr/lib/libattr.so.1", O_RDONLY|O_CLOEXEC) = 3
[00007f61d2c3a377] read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0200\21\0\0\0\0\0"...,
832) = 832
[00007f61d2c3a2e2] fstat(3, {st_mode=S_IFREG|0755, st_size=18248, ...}) = 0
[00007f61d2c3a50a] mmap(NULL, 2113560, PROT_READ|PROT_EXEC, MAP_PRIVATE|
MAP_DENYWRITE, 3, 0) = 0x7f61d2813000
[00007f61d2c3a5a7] mprotect(0x7f61d2817000, 2093056, PROT_NONE) = 0
[00007f61d2c3a50a] mmap(0x7f61d2a16000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x3000) = 0x7f61d2a16000
[00007f61d2c3a4b7] close(3) = 0
[00007f61d2c3a357] open("/usr/lib/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
[00007f61d2c3a377] read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\3\0>\0\1\0\0\0340\5\2\0\0\0\0\0"...,
832) = 832
[00007f61d2c3a2e2] fstat(3, {st_mode=S_IFREG|0755, st_size=1985472, ...}) = 0
[00007f61d2c3a50a] mmap(NULL, 3823824, PROT_READ|PROT_EXEC, MAP_PRIVATE|
MAP_DENYWRITE, 3, 0) = 0x7f61d246d000
[00007f61d2c3a5a7] mprotect(0x7f61d260a000, 2093056, PROT_NONE) = 0
[00007f61d2c3a50a] mmap(0x7f61d2809000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x19c000) = 0x7f61d2809000
[00007f61d2c3a50a] mmap(0x7f61d280f000, 14544, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f61d280f000
[00007f61d2c3a4b7] close(3) = 0
[00007f61d2c3a50a] mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|
MAP_ANONYMOUS, -1, 0) = 0x7f61d2df3000
[00007f61d2c21c55] arch_prctl(ARCH_SET_FS, 0x7f61d2df3700) = 0
[00007f61d2c3a5a7] mprotect(0x7f61d2809000, 16384, PROT_READ) = 0
```

```

[00007f61d2c3a5a7] mprotect(0x7f61d2a16000, 4096, PROT_READ) = 0
[00007f61d2c3a5a7] mprotect(0x7f61d2c1f000, 4096, PROT_READ) = 0
[00007f61d2c3a5a7] mprotect(0x61d000, 4096, PROT_READ) = 0
[00007f61d2c3a5a7] mprotect(0x7f61d2e44000, 4096, PROT_READ) = 0
[00007f61d2c3a587] munmap(0x7f61d2df8000, 309556) = 0
[00007f61d2552799] brk(NULL) = 0x687000
[00007f61d2552799] brk(0x6a8000) = 0x6a8000
[00007f61d24983c3] open("/usr/lib/locale/locale-archive", O_RDONLY|O_CLOEXEC) = 3
[00007f61d254c4e2] fstat(3, {st_mode=S_IFREG|0644, st_size=1674704, ...}) = 0
[00007f61d255623a] mmap(NULL, 1674704, PROT_READ, MAP_PRIVATE, 3, 0) =
0x7f61d2c5a000
[00007f61d2498487] close(3) = 0
[00007f61d2529997] geteuid() = 1000
[00007f61d2551f4a] ioctl(0, TCGETS, {B38400 opost isig icanon echo ...}) = 0
[00007f61d24e24ea] open("/usr/share/locale/locale.alias", O_RDONLY|O_CLOEXEC) = 3
[00007f61d254c4e2] fstat(3, {st_mode=S_IFREG|0644, st_size=2997, ...}) = 0
[00007f61d24e1f4c] read(3, "# Locale name alias data base.\n#", 4096) = 2997
[00007f61d24e1f4c] read(3, "", 4096) = 0
[00007f61d24e0b0b] close(3) = 0
[00007f61d249bdc1] open("/usr/share/locale/en_US.utf8/LC_MESSAGES/coreutils.mo",
O_RDONLY) = -1 ENOENT (No such file or directory)
[00007f61d249bdc1] open("/usr/share/locale/en_US/LC_MESSAGES/coreutils.mo", O_RDONLY)
= -1 ENOENT (No such file or directory)
[00007f61d249bdc1] open("/usr/share/locale/en.utf8/LC_MESSAGES/coreutils.mo", O_RDONLY)
= -1 ENOENT (No such file or directory)
[00007f61d249bdc1] open("/usr/share/locale/en/LC_MESSAGES/coreutils.mo", O_RDONLY) = -1
ENOENT (No such file or directory)
[00007f61d254cbf0] write(2, "mv: ", 4) = 4
[00007f61d254cbf0] write(2, "missing file operand", 20) = 20
[00007f61d254cbf0] write(2, "\n", 1) = 1
[00007f61d254cbf0] write(2, "Try 'mv --help' for more informa"..., 38) = 38
[00007f61d254cc4d] lseek(0, 0, SEEK_CUR) = -1 ESPIPE (Illegal seek)
[00007f61d24e0b0b] close(0) = 0
[00007f61d24e0b0b] close(1) = 0
[00007f61d24e0b0b] close(2) = 0
[00007f61d2528f88] exit_group(1) = ?
[????????????????] +++ exited with 1 +++

```

Arquivos de strace gerados com ids e chamadas:

log_syscall_cp.txt

```

[00007fc3f98c6fb7] execve
[00007fe0bcd24639] brk
[00007fe0bcd2540a] access
[00007fe0bcd25357] open
[00007fe0bcd252e2] fstat

```

[00007fe0bcd2550a] mmap
[00007fe0bcd254b7] close
[00007fe0bcd25357] open
[00007fe0bcd25377] read
[00007fe0bcd252e2] fstat
[00007fe0bcd2550a] mmap
[00007fe0bcd2550a] mmap
[00007fe0bcd255a7] mprotect
[00007fe0bcd2550a] mmap
[00007fe0bcd254b7] close
[00007fe0bcd25357] open
[00007fe0bcd25377] read
[00007fe0bcd252e2] fstat
[00007fe0bcd2550a] mmap
[00007fe0bcd255a7] mprotect
[00007fe0bcd2550a] mmap
[00007fe0bcd254b7] close
[00007fe0bcd25357] open
[00007fe0bcd25377] read
[00007fe0bcd252e2] fstat
[00007fe0bcd2550a] mmap
[00007fe0bcd255a7] mprotect
[00007fe0bcd2550a] mmap
[00007fe0bcd2550a] mmap
[00007fe0bcd254b7] close
[00007fe0bcd2550a] mmap
[00007fe0bcd0cc55] arch_prctl
[00007fe0bcd255a7] mprotect
[00007fe0bcd255a7] mprotect
[00007fe0bcd255a7] mprotect
[00007fe0bcd255a7] mprotect
[00007fe0bcd255a7] mprotect
[00007fe0bcd25587] munmap
[00007fe0bc63d799] brk
[00007fe0bc63d799] brk
[00007fe0bc5833c3] open
[00007fe0bc6374e2] fstat
[00007fe0bc64123a] mmap
[00007fe0bc583487] close
[00007fe0bc614997] geteuid
[00007fe0bc5cd4ea] open
[00007fe0bc6374e2] fstat
[00007fe0bc5ccf4c] read
[00007fe0bc5ccf4c] read
[00007fe0bc5cbb0b] close
[00007fe0bc586dc1] open
[00007fe0bc586dc1] open
[00007fe0bc586dc1] open

```
[00007fe0bc586dc1] open
[00007fe0bc637bf0] write
[00007fe0bc637bf0] write
[00007fe0bc637bf0] write
[00007fe0bc637bf0] write
[00007fe0bc637c4d] lseek
[00007fe0bc5cbb0b] close
[00007fe0bc5cbb0b] close
[00007fe0bc5cbb0b] close
[00007fe0bc613f88] exit_group
```

log_syscall_mv.txt

```
[00007fd52863efb7] execve
[00007f61d2c39639] brk
[00007f61d2c3a40a] access
[00007f61d2c3a357] open
[00007f61d2c3a2e2] fstat
[00007f61d2c3a50a] mmap
[00007f61d2c3a4b7] close
[00007f61d2c3a357] open
[00007f61d2c3a377] read
[00007f61d2c3a2e2] fstat
[00007f61d2c3a50a] mmap
[00007f61d2c3a50a] mmap
[00007f61d2c3a5a7] mprotect
[00007f61d2c3a50a] mmap
[00007f61d2c3a4b7] close
[00007f61d2c3a357] open
[00007f61d2c3a377] read
[00007f61d2c3a2e2] fstat
[00007f61d2c3a50a] mmap
[00007f61d2c3a5a7] mprotect
[00007f61d2c3a50a] mmap
[00007f61d2c3a4b7] close
[00007f61d2c3a357] open
[00007f61d2c3a377] read
[00007f61d2c3a2e2] fstat
[00007f61d2c3a50a] mmap
[00007f61d2c3a5a7] mprotect
[00007f61d2c3a50a] mmap
[00007f61d2c3a50a] mmap
[00007f61d2c3a4b7] close
[00007f61d2c3a50a] mmap
[00007f61d2c21c55] arch_prctl
[00007f61d2c3a5a7] mprotect
[00007f61d2c3a5a7] mprotect
[00007f61d2c3a5a7] mprotect
```

```
[00007f61d2c3a5a7] mprotect
[00007f61d2c3a5a7] mprotect
[00007f61d2c3a587] munmap
[00007f61d2552799] brk
[00007f61d2552799] brk
[00007f61d24983c3] open
[00007f61d254c4e2] fstat
[00007f61d255623a] mmap
[00007f61d2498487] close
[00007f61d2529997] geteuid
[00007f61d2551f4a] ioctl
[00007f61d24e24ea] open
[00007f61d254c4e2] fstat
[00007f61d24e1f4c] read
[00007f61d24e1f4c] read
[00007f61d24e0b0b] close
[00007f61d249bdc1] open
[00007f61d249bdc1] open
[00007f61d249bdc1] open
[00007f61d249bdc1] open
[00007f61d254cbf0] write
[00007f61d254cbf0] write
[00007f61d254cbf0] write
[00007f61d254cbf0] write
[00007f61d254cc4d] lseek
[00007f61d24e0b0b] close
[00007f61d24e0b0b] close
[00007f61d24e0b0b] close
[00007f61d2528f88] exit_group
```

OBS: como as aplicações cp e mv são muito parecidas, a maioria das chamadas que elas fazem ao sistema coincidem.

Os logs com ids e chamadas são gerados pelo seguinte algoritmo:

_create_log_file

```
def _create_log_file(file="log_cp.txt",arq="teste_cp.txt"):
    with open(file,"r") as file1:
        with open(arq,"w") as arq1:
            for line in file1:
                result = line.split("(")[0]
                if result[1] != "+":
                    arq1.write(result+"\n")

_create_log_file(file="log_cp.txt",arq="teste_cp.txt")
```

Descrição: o código a seguir recebe o id e o nome de cada chamada que a aplicação faz ao sistema a partir dos straces

(I)

Criação dos N-grams

n_gram_cp

execve, brk, access, open, fstat, mmap, close, open, read, fstat
brk, access, open, fstat, mmap, close, open, read, fstat, mmap
access, open, fstat, mmap, close, open, read, fstat, mmap, mmap
open, fstat, mmap, close, open, read, fstat, mmap, mmap, mprotect
fstat, mmap, close, open, read, fstat, mmap, mmap, mprotect, mmap
mmap, close, open, read, fstat, mmap, mmap, mprotect, mmap, close
close, open, read, fstat, mmap, mmap, mprotect, mmap, close, open
open, read, fstat, mmap, mmap, mprotect, mmap, close, open, read
read, fstat, mmap, mmap, mprotect, mmap, close, open, read, fstat
fstat, mmap, mmap, mprotect, mmap, close, open, read, fstat, mmap
mmap, mmap, mprotect, mmap, close, open, read, fstat, mmap, mprotect
mmap, mprotect, mmap, close, open, read, fstat, mmap, mprotect, mmap
mprotect, mmap, close, open, read, fstat, mmap, mprotect, mmap, close
mmap, close, open, read, fstat, mmap, mprotect, mmap, close, open
close, open, read, fstat, mmap, mprotect, mmap, close, open, read
open, read, fstat, mmap, mprotect, mmap, close, open, read, fstat
read, fstat, mmap, mprotect, mmap, close, open, read, fstat, mmap
fstat, mmap, mprotect, mmap, close, open, read, fstat, mmap, mprotect
mmap, mprotect, mmap, close, open, read, fstat, mmap, mprotect, mmap
mprotect, mmap, close, open, read, fstat, mmap, mprotect, mmap, mmap
mmap, close, open, read, fstat, mmap, mprotect, mmap, mmap, close
close, open, read, fstat, mmap, mprotect, mmap, mmap, close, mmap

open, read, fstat, mmap, mprotect, mmap, mmap, close, mmap, arch_prctl
 read, fstat, mmap, mprotect, mmap, mmap, close, mmap, arch_prctl, mprotect
 fstat, mmap, mprotect, mmap, mmap, close, mmap, arch_prctl, mprotect, mprotect
 mmap, mprotect, mmap, mmap, close, mmap, arch_prctl, mprotect, mprotect, mprotect
 mprotect, mmap, mmap, close, mmap, arch_prctl, mprotect, mprotect, mprotect, mprotect
 mmap, mmap, close, mmap, arch_prctl, mprotect, mprotect, mprotect, mprotect, mprotect
 mmap, close, mmap, arch_prctl, mprotect, mprotect, mprotect, mprotect, mprotect, munmap
 close, mmap, arch_prctl, mprotect, mprotect, mprotect, mprotect, mprotect, munmap, brk
 mmap, arch_prctl, mprotect, mprotect, mprotect, mprotect, mprotect, munmap, brk, brk
 arch_prctl, mprotect, mprotect, mprotect, mprotect, mprotect, munmap, brk, brk, open
 mprotect, mprotect, mprotect, mprotect, mprotect, munmap, brk, brk, open, fstat
 mprotect, mprotect, mprotect, mprotect, munmap, brk, brk, open, fstat, mmap
 mprotect, mprotect, mprotect, munmap, brk, brk, open, fstat, mmap, close
 mprotect, mprotect, munmap, brk, brk, open, fstat, mmap, close, geteuid
 mprotect, munmap, brk, brk, open, fstat, mmap, close, geteuid, open
 munmap, brk, brk, open, fstat, mmap, close, geteuid, open, fstat
 brk, brk, open, fstat, mmap, close, geteuid, open, fstat, read
 brk, open, fstat, mmap, close, geteuid, open, fstat, read, read
 open, fstat, mmap, close, geteuid, open, fstat, read, read, close
 fstat, mmap, close, geteuid, open, fstat, read, read, close, open
 mmap, close, geteuid, open, fstat, read, read, close, open, open
 close, geteuid, open, fstat, read, read, close, open, open, open
 geteuid, open, fstat, read, read, close, open, open, open, open
 open, fstat, read, read, close, open, open, open, open, write
 fstat, read, read, close, open, open, open, open, write, write
 read, read, close, open, open, open, open, write, write, write
 read, close, open, open, open, open, write, write, write, write
 close, open, open, open, open, write, write, write, write, lseek
 open, open, open, open, write, write, write, write, lseek, close
 open, open, open, write, write, write, write, lseek, close, close
 open, open, write, write, write, write, lseek, close, close, close
 open, write, write, write, write, lseek, close, close, close, exit_group

n_gram_mv

execve, brk, access, open, fstat, mmap, close, open, read, fstat
 brk, access, open, fstat, mmap, close, open, read, fstat, mmap
 access, open, fstat, mmap, close, open, read, fstat, mmap, mmap
 open, fstat, mmap, close, open, read, fstat, mmap, mmap, mprotect
 fstat, mmap, close, open, read, fstat, mmap, mmap, mprotect, mmap
 mmap, close, open, read, fstat, mmap, mmap, mprotect, mmap, close
 close, open, read, fstat, mmap, mmap, mprotect, mmap, close, open
 open, read, fstat, mmap, mmap, mprotect, mmap, close, open, read
 read, fstat, mmap, mmap, mprotect, mmap, close, open, read, fstat
 fstat, mmap, mmap, mprotect, mmap, close, open, read, fstat, mmap
 mmap, mmap, mprotect, mmap, close, open, read, fstat, mmap, mprotect

[illegible]

Gerados pelo seguinte algoritmo:

`_create_gram`

```
def _create_gram(n, file="log_syscall_mv_without_id.txt", arq="n_gram_mv.txt"):
    with open(file, "r") as file1, open(arq, "w") as arq1:
        data = [x[:-1] for x in file1]
        for i in range(len(data) - n + 1):
            arq1.write(str(data[i:i+n])[1:-1].replace("\'", "'") + "\n")

_create_gram(10, file="log_syscall_mv_without_id.txt", arq="n_gram_mv.txt")
```

Descrição: este código recebe o log com as chamadas de cada aplicação e escreve um arquivo de retorno com a paginação de tamanho 10 até a ultima chamada feita pela aplicação

OBS: para a criação dos n-grams também foram gerados arquivos de log sem o id de cada chamada, usando o seguinte código:

`_create_log_without_id`

```
def _create_log_without_id(file="teste_mv.txt", arq="teste_mv_without_id.txt"):
    with open(file) as file1, open(arq, "w") as arq1:
        for line in file1:
            result = line.split(" ")[1]
            if result[1] != "+":
                arq1.write(result)

_create_log_without_id(file="teste_mv.txt", arq="teste_mv_without_id.txt")
```

Descrição: este código recebe o log gerado anteriormente contendo o ids e nome das chamadas e extrai somente o nome, para facilitar as demais operações.

(II)

Comparação dos arquivos saudável e infectado

Retorno da comparação

Infection: 20.0%

_compare

```
def _compare(healthy="n_gram_cp.txt", infected="n_gram_mv.txt"):
    with open(healthy, "r") as arq_healthy, open(infected, "r") as arq_infected:
        arq1 = [z[:-1] for z in arq_healthy]
        arq2 = [y[:-1] for y in arq_infected]
        inter = set(arq1).intersection(arq2)

        print("Infection: ", (len(arq2)-len(inter))/len(arq2)*100,"%")

_compare(healthy="n_gram_cp.txt", infected="n_gram_mv.txt")
```

Descrição: este código recebe o gram das aplicações cp e mv e faz a comparação para ver qual chamada de sistema não está presente no arquivo saudável, essa chamada é tida como infectada, retornando taxa de infecção presente no arquivo.