

# MAC0422 - Sistemas Operacionais - 2018

## Relatório do EP 4

Danilo Escudero - 10853911  
Renan Tiago - 9793606

### **PROCESSO DE FEITURA**

Foi acrescentado um novo tipo de i-node chamado I\_TEMPORARY, contendo o valor octal 0140000, no arquivo `/usr/src/include/minix/const.h`.

Foi criada uma macro associada ao número 31 no `callnr.h`, `OPEN_TMP`. Ela será associada à função que trata da chamada de sistema no file system.

Criamos um header para a função `open_tmp(char *name, char *mode)`, que manda mensagem para o server em `/usr/src/include/open_tmp.h`. Essa função, além de mandar o nome do arquivo como mensagem, tem o papel de receber o mode e transformar em flags que a função do file system pode interpretar e enviar na mensagem.

No File System Server, criamos o arquivo `open_tmp.c`, que é uma modificação de algumas funções do `open.c`. Ele possui a função associada à chamada de número 31 (`do_open_tmp`), e as funções privadas `newnode` e `common_open`. O papel da função `do_open_tmp` é de chamar a função `common_open` e então cumprir o papel da primeira parte do EP. Porém, não conseguimos obter sucesso após diversas tentativas, e diversas abordagens diferentes ao problema.

### **PONTO A RESSALTAR**

Em outro momento, tentamos também, criar uma `Oflag` em `/usr/include/fcntl.h`, e a chamamos de `OPEN_TMP`. A tentativa era de realizar pequenas modificações na função `do_open` do arquivo `open.c` em `/usr/src/servers/fs`, de modo que fosse capaz de criar arquivos temporários sem que seja necessário criar uma função extra no server. Essa abordagem também se mostrou infrutífera.

### **TESTES**

Há um arquivo c chamado `"test.c"` no diretório `/root`, que serve para demonstrar como seria o funcionamento da chamada de sistema, se implementada da maneira correta.

## **ARQUIVOS MODIFICADOS**

/usr/src/include/minix/const.h

/usr/src/include/minix/callnr.h

/usr/src/include/open\_tmp.h

/usr/src/servers/fs/Makefile

/usr/src/servers/fs/table.c

/usr/src/servers/fs/proto.h

/usr/src/servers/fs/open\_tmp.c