

EP1 - MAC0422

Bruno Carneiro da Cunha - NUSP: 10376388
Guilherme Yambanis Thomaz - NUSP: 8041265

September 10, 2018

Abstract

EP 1 de MAC422 - Sistemas Operacionais. O Objetivo do EP é a instalação do MINIX em uma máquina virtual e a criação de uma shell rudimentar.
O Usuario tem nome root e senha root

1 Instalação

A primeira etapa consistiu na instalação da máquina virtual e do minix.

1.1 Máquina Virtual

Como máquina virtual, utilizamos o VirtualBox da Oracle. Ao criar a imagem do Minix, utilizamos as configurações sugeridas de 1024 Mb de memória e 512 Mb de disco VDI dinamicamente alocado.

1.2 MINIX

Utilizamos a versão 3.1.2a do sistema, conforme sugerido.

2 MAC422Shell

2.1 Comandos

2.1.1 `protegepracaramba <arquivo>`

O comando `protegepracaramba` faz com que o arquivo dado como parâmetro tenha proteção 000. Utilizamos a chamada de função `chmod`, contida em `/sys/stat.h` com argumento 000, que faz o system call correspondente.

```
chmod(arg, 000)
```

Utilizamos ainda a funcao `stat` para fazer a checagem do novo modo do arquivo.

```
stat(arg, &buf);  
int statchmod = buf.st_mode & (S_IRWXU | S_IRWXG | S_IRWXO);  
printf("novo modo do arquivo: %o\n",statchmod);
```

2.1.2 `liberageral <arquivo>`

O comando `liberageral` faz com que o arquivo dado como parâmetro tenha proteção 777. Utilizamos a chamada de função `chmod`, contida em `/sys/stat.h` com argumento 777, que faz o system call correspondente.

```
chmod(arg, 777)
```

Utilizamos ainda a funcao `stat` para fazer a checagem do novo modo do arquivo.

```
stat(arg, &buf);  
int statchmod = buf.st_mode & (S_IRWXU | S_IRWXG | S_IRWXO);  
printf("novo modo do arquivo: %o\n",statchmod);
```

2.1.3 rodeveja <programa>

O comando rodeveja executa o programa indicado utilizando as chamadas `fork()`, `execve()` e `wait()`. A shell deve emitir uma mensagem indicando o código de retorno.

```
fork():  
  
    pid_t pid = fork();  
  
execve():  
  
    execve(bin, NULL, NULL);  
  
wait():  
  
    pid_t tid = waitpid(pid, &childStatus, 0);
```

Utilizamos o seguinte comando para imprimir o código do retorno

```
printf("=> programa '%s' retornou %d\n", bin, WEXITSTATUS(childStatus));
```

2.1.4 rode <programa>

Executa o programa indicado em background utilizando as chamadas `fork()` e `execve()`. A shell monopoliza o teclado, a tela deve continuar mostrando a saída da shell e do programa. Ao contrário do comando anterior, a shell não deve retornar o código de saída do programa.

```
fork():  
  
    pid_t pid = fork();  
  
execve():  
  
    execve(bin, NULL, NULL);
```

2.2 compilação no minix

Utilizamos o compilador nativo do minix, o `cc`, para compilar o programa `mac422shell.c` e gerar o executável. Colocamos o executável `mac422shell` na pasta `/usr/local/bin` e o código fonte `/usr/local/src`.

```
cc -o mac422shell mac422shell.c
```