

Escreva programas em dtrace para:

1. Fazer o traçado das chamadas ao sistema `open()` que deverá imprimir a seguinte informação por linha:

- nome do ficheiro executável e respetivos: PID do processo, UID do utilizador e GID do grupo.
- Caminho absoluto para o ficheiro que for aberto.
- A cadeia de caracteres com as "flags" da chamada ao sistema `open()`, `O_RDONLY`, `O_WRONLY`, `O_RDWR`, `O_APPEND`, `O_CREAT`
- O Valor de retorno de chamada de sistema

Testar o programa com as hipóteses que seguem:

- `cat /etc/inittab > /tmp/test`
- `cat /etc/inittab >> /tmp/test`
- `cat /etc/inittab | tee /tmp/test`
- `cat /etc/inittab | tee -a /tmp/test`

Opcional: Modificar o programa para que apenas os ficheiros com `"/etc"` no caminho sejam detetados:

Aviso: Em Solaris 11 a chamada de sistema `open ()` foi substituído por outra mais genérico `openat ()`.

Flag	Solaris	Linux (x86)
<code>O_RDONLY</code>	bits 0-1 are not set	
<code>O_WRONLY</code>	1	1
<code>O_RDWR</code>	2	2
<code>O_APPEND</code>	8	1024
<code>O_CREAT</code>	256	64

2. Mostrar para os processos que estão a correr no sistema as seguintes estatísticas, com valores obtidos durante cada iteração:

a)

- número de tentativas de abrir ficheiros existentes;
- número de tentativas para criar ficheiros;
- número de tentativas bem-sucedidas.

b) Repetidamente, com um período (especificado em segundos) passado como argumentos da linha de comandos deve imprimir:

- hora e dia atual em formato legível.
- as estatísticas recolhidas por PID e respetivo o nome.

3. Replicar o comportamento programa **`strace -c <programa>`**. A opção `-c` permite contabilizar o número de ocorrência de cada chamada ao sistema e o tempo despendido, durante a execução do programa.