

ltrace

用来跟踪进程调用库函数的情况

补充说明

ltrace命令 是用来跟踪进程调用库函数的情况。

语法

```
ltrace [option ...] [command [arg ...]]
```

选项

- a 对齐具体某个列的返回值。
- c 计算时间和调用，并在程序退出时打印摘要。
- C 解码低级别名称（内核级）为用户级名称。
- d 打印调试信息。
- e 改变跟踪的事件。
- f 跟踪子进程。
- h 打印帮助信息。
- i 打印指令指针，当库调用时。
- l 只打印某个库中的调用。
- L 不打印库调用。
- n, --indent=NR 对每个调用级别嵌套以NR个空格进行缩进输出。
- o, --output=file 把输出定向到文件。
- p PID 附着在值为PID的进程号上进行ltrace
- r 打印相对时间戳。
- s STRLEN 设置打印的字符串最大长度。
- S 显示系统调用。
- t, -tt, -ttt 打印绝对时间戳。
- T 输出每个调用过程的时间开销。
- u USERNAME 使用某个用户id或组ID来运行命令。
- V, --version 打印版本信息，然后退出。
- x NAME treat the global NAME like a library subroutine.（求翻译）

实例

最基本应用，不带任何参数：

```
[guest@localhost tmp]$ ltrace ./a.out
__libc_start_main(0x80484aa, 1, 0xbfc07744, 0x8048550, 0x8048540 <unfinished
...>
printf("no1:%d \t no2:%d \t diff:%d\n", 10, 6, 4no1:10 no2:6 diff:4 ) = 24
printf("no1:%d \t no2:%d \t diff:%d\n", 9, 7, 2no1:9 no2:7 diff:2 ) = 23
printf("no1:%d \t no2:%d \t diff:%d\n", 8, 8, 0no1:8 no2:8 diff:0 ) = 23
--- SIGFPE (Floating point exception) ---
+++ killed by SIGFPE +++
```

输出调用时间开销：

```
[guest@localhost tmp]$ ltrace -T ./a.out
__libc_start_main(0x80484aa, 1, 0xbf81d394, 0x8048550, 0x8048540 <unfinished
...>
printf("no1:%d \t no2:%d \t diff:%d\n", 10, 6, 4no1:10 no2:6 diff:4 ) = 24
<0.000972>
printf("no1:%d \t no2:%d \t diff:%d\n", 9, 7, 2no1:9 no2:7 diff:2 ) = 23
<0.000155>
printf("no1:%d \t no2:%d \t diff:%d\n", 8, 8, 0no1:8 no2:8 diff:0 ) = 23
<0.000153>
--- SIGFPE (Floating point exception) ---
+++ killed by SIGFPE +++
```

显示系统调用：

```
[guest@localhost tmp]$ ltrace -S ./a.out
SYS_brk(NULL) = 0x9e20000
SYS_access(0xa4710f, 4, 0xa4afc0, 0, 0xa4b644) = 0
SYS_open("/etc/ld.so.preload", 0, 02) = 3
SYS_fstat64(3, 0xbfb7a94, 0xa4afc0, -1, 3) = 0
SYS_mmap2(0, 17, 3, 2, 3) = 0xb7f2a000
SYS_close(3) = 0
SYS_open("/lib/libcwait.so", 0, 00) = 3
SYS_read(3, "\177ELF\001\001\001", 512) = 512
SYS_fstat64(3, 0xbfb76fc, 0xa4afc0, 4, 0xa4b658) = 0
SYS_mmap2(0, 4096, 3, 34, -1) = 0xb7f29000
SYS_mmap2(0, 5544, 5, 2050, 3) = 0x423000
SYS_mmap2(0x424000, 4096, 3, 2066, 3) = 0x424000
.....省去若干行
```

From:

<https://rd.irust.top/> - 学习笔记

Permanent link:

<https://rd.irust.top/doku.php?id=command:ltrace>

Last update: **2021/10/15 14:58**

