NAME

truss - trace system calls

SYNOPSIS

truss [**-facedDHS**] [**-o** *file*] [**-s** *strsize*] **-p** *pid*
truss [**-facedDHS**] [**-o** *file*] [**-s** *strsize*] *command* [*args*]

DESCRIPTION

The **truss** utility traces the system calls called by the specified process or program. Output is to the specified output file, or standard error by default. It does this by stopping and restarting the process being monitored via ptrace(2).

The options are as follows:

-f Trace descendants of the original traced process created by fork(2), vfork(2), etc. To distinguish events between processes, the process ID (PID) of the process is included in the output of each event.

-a Show the argument strings that are passed in each execve(2) system call.

-c Do not display individual system calls or signals. Instead, before exiting, print a summary containing for each system call: the total system time used, the number of times the call was invoked, and the number of times the call returned with an error.

-e Show the environment strings that are passed in each execve(2) system call.

-d Include timestamps in the output showing the time elapsed since the trace was started.

-D Include timestamps in the output showing the time elapsed since the last recorded event.

-H Include the thread ID of in the output of each event.

-S Do not display information about signals received by the process. (Normally, **truss** displays signal as well as system call events.)

-o *file* Print the output to the specified *file* instead of standard error.

-s *strsize*
Display strings using at most *strsize* characters. If the buffer is larger, "..." will be displayed at the end of the string. The default *strsize* is 32.

-p *pid* Follow the process specified by *pid* instead of a new command.

command [*args*]
Execute *command* and trace the system calls of it. (The **-p** and *command* options are mutually exclusive.)

EXAMPLES

Follow the system calls used in echoing "hello":

```
$ truss /bin/echo hello
```

Do the same, but put the output into a file:

```
$ truss -o /tmp/truss.out /bin/echo hello
```

Follow an already-running process:

```
$ truss -p 34
```

SEE ALSO

dtrace(1), kdump(1), ktrace(1), ptrace(2), utrace(2)

HISTORY

The **truss** command was written by Sean Eric Fagan for FreeBSD. It was modeled after similar commands available for System V Release 4 and SunOS.