

**NAME**

**boottrace** - Boot-time, run-time, and shutdown-time tracing facility

**SYNOPSIS**

```
#include <sys/boottrace.h>
```

**DESCRIPTION**

**boottrace** is a kernel-userspace interface for capturing trace events during system boot and shutdown (in particular, one-shot events).

Event annotations are present in:

- the boot and shutdown paths in the kernel
- some key system utilities (init(8), shutdown(8), reboot(8))
- rc(8) scripts

**boottrace** is unconditionally compiled into the kernel and disabled by default.

**EVENT TABLES**

Events are stored in three event tables: boot-time events, run-time events, and shutdown-time events.

<b>Table Name</b>	<b>Event Description</b>
boot-time events	Boot, kernel initialization, and rc(8) execution; until init(8) transitions into multi-user mode
run-time events	From when the system has completed booting (including rc(8) execution) and init(8) transitions to multi-user mode until the beginning of shutdown procedures
shutdown-time events	After initialization of a shutdown, a reboot, or a kernel panic

**LOADER TUNABLES**

Tunables can be set at the loader(8) prompt before booting the kernel or stored in loader.conf(5).

**boottrace** features the following loader tunables:

*kern.boottrace.dotrace\_kernel*

Set to '1' to enable tracing of kernel events. Default: '1' (enabled).

*kern.boottrace.dotrace\_user*

Set to '1' to enable tracing of userspace events. Default: '1' (enabled).

**SYSCTL VARIABLES**

The following variables are available as both sysctl(8) variables and loader(8) tunables:

*kern.boottrace.boottrace*

Create a new trace event and write it to the boot-time table.

A new trace event consists of a process name and an event description, separated by a colon (':'). If the colon is missing or if the provided string for the process name is empty, the process name is inferred from the invoking process (which is its executable name).

*kern.boottrace.enabled*

Set to '1' to enable tracing. This is a read-only sysctl(8) variable. Default: '0' (disabled).

*kern.boottrace.log*

Show the events stored in boot-time and run-time tables. This is an opaque sysctl(8) variable.

*kern.boottrace.runtrace*

Same as *kern.boottrace.boottrace*, but write to the run-time table.

*kern.boottrace.shuttrace*

Same as *kern.boottrace.boottrace*, but write to the shutdown-time table.

*kern.boottrace.shutdown\_trace*

Log shutdown-time events to the console before the system halts.

*kern.boottrace.shutdown\_trace\_threshold*

Set a time threshold for logging shutdown-time events in milliseconds. An event is ignored if the time difference to the previous event is less than the threshold value. Default: '0' (logs all events).

**EXAMPLES**

Create a new trace event with a process name "foo" and an event description "bar" using sysctl(8):

```
sysctl kern.boottrace.boottrace="foo:bar"
```

Here is a sample output of *kern.boottrace.log* (shortened with "[...]" for readability):

CPU	msecs	delta	process	event	PID	CPUtime	IBlks	OBlks
0	44872811	0	kernel	sysinit 0x2100001	0	0.00	0	0
0	44872812	1	kernel	sysinit 0x2110000	0	0.00	0	0
0	44872812	0	kernel	sysinit 0x2140000	0	0.00	0	0
[...]								
0	44872817	0	kernel	sysinit 0x2800000	0	0.00	0	0

```

0 44873820 1003 kernel      sysinit 0x2880000      0 0.00 0 0
0 44873820 0 kernel        sysinit 0x2888000      0 0.00 0 0
[...]
1 44875735 0 kernel        sysinit 0xffffffff      0 0.00 0 0
1 44875735 0 swapper       mi_startup done        0 0.00 0 0
0 44875750 15 init         init(8) starting...    1 0.00 0 0
0 44875751 1 init         /etc/rc starting...    1 0.00 0 0
0 44875831 80 boottrace    /etc/rc.d/rctl start   26 0.00 0 0
1 44875839 8 boottrace    /etc/rc.d/rctl done    26 0.00 2 0
[...]
0 44876446 0 boottrace    /etc/rc.d/netif start  390 0.00 0 0
1 44881116 4670 boottrace /etc/rc.d/netif done   390 0.12 34 0
[...]
0 44882866 1 boottrace    /etc/rc.d/securelevel start 1679 0.00 0 0
0 44882872 6 boottrace    /etc/rc.d/securelevel done  1679 0.00 0 0
1 44882879 7 init         /etc/rc finished      1 2.22 743 15
Total measured time: 10068 msecs

```

```

CPU  msec  delta process      event                      PID CPUtime IBkls OBkls
1 44882880 0 init             multi-user start          1 2.22 743 15
0 44918215 35335 kldload         hwpmc.ko: sysinit 0xd800000 1698 0.00 0 0
Total measured time: 35335 msecs

```

**SEE ALSO**

tslog(4), boottrace(8), sysctl(8)

**HISTORY**

NetApp created **boottrace** to diagnose slow devices and subsystems. Once upstreamed, **boottrace** was first publicly released with FreeBSD 14.0.

**AUTHORS**

This manual page was written by Mateusz Piotrowski <Omp@FreeBSD.org>.