



Red Hat Enterprise Linux Release 9.2 Manual Pages on 'posix_trace_event.3p' command

\$ man posix_trace_event.3p

POSIX_TRACE_EVENT(3P) POSIX Programmer's Manual POSIX_TRACE_EVENT(3P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

posix_trace_event, posix_trace_eventid_open ? trace functions for instrumenting application code (TRACING)

SYNOPSIS

```
#include <sys/types.h>
#include <trace.h>
void posix_trace_event(trace_event_id_t event_id,
    const void *restrict data_ptr, size_t data_len);
int posix_trace_eventid_open(const char *restrict event_name,
    trace_event_id_t *restrict event_id);
```

DESCRIPTION

The `posix_trace_event()` function shall record the `event_id` and the user data pointed to by `data_ptr` in the trace stream into which the calling process is being traced and in which `event_id` is not filtered out. If the total size of the user trace event data represented by `data_len` is not greater than the declared maximum size for user trace event data, then the truncation-status attribute of the trace event recorded is

POSIX_TRACE_NOT_TRUNCATED. Otherwise, the user trace event data is truncated to this declared maximum size and the truncation-status attribute of the trace event recorded is POSIX_TRACE_TRUNCATED_RECORD.

If there is no trace stream created for the process or if the created trace stream is not running, or if the trace event specified by event_id is filtered out in the trace stream, the posix_trace_event() function shall have no effect.

The posix_trace_eventid_open() function shall associate a user trace event name with a trace event type identifier for the calling process.

The trace event name is the string pointed to by the argument event_name. It shall have a maximum of {TRACE_EVENT_NAME_MAX} characters (which has the minimum value {_POSIX_TRACE_EVENT_NAME_MAX}). The number of user trace event type identifiers that can be defined for any given process is limited by the maximum value {TRACE_USER_EVENT_MAX}, which has the minimum value {POSIX_TRACE_USER_EVENT_MAX}.

If the Trace Inherit option is not supported, the posix_trace_eventid_open() function shall associate the user trace event name pointed to by the event_name argument with a trace event type identifier that is unique for the traced process, and is returned in the variable pointed to by the event_id argument. If the user trace event name has already been mapped for the traced process, then the previously assigned signed trace event type identifier shall be returned. If the per-process user trace event name limit represented by {TRACE_USER_EVENT_MAX} has been reached, the pre-defined POSIX_TRACE_UNNAMED_USEREVENT (see Table 2-7, Trace Option: User Trace Event) user trace event shall be returned.

If the Trace Inherit option is supported, the posix_trace_eventid_open() function shall associate the user trace event name pointed to by the event_name argument with a trace event type identifier that is unique for all the processes being traced in this same trace stream, and is returned in the variable pointed to by the event_id argument. If the user trace event name has already been mapped for the traced processes, then the previously assigned trace event type identifier shall

be returned. If the per-process user trace event name limit represented by `{TRACE_USER_EVENT_MAX}` has been reached, the pre-defined `POSIX_TRACE_UNNAMED_USEREVENT` (Table 2-7, Trace Option: User Trace Event) user trace event shall be returned.

Note: The above procedure, together with the fact that multiple processes can only be traced into the same trace stream by inheritance, ensure that all the processes that are traced into a trace stream have the same mapping of trace event names to trace event type identifiers.

If there is no trace stream created, the `posix_trace_eventid_open()` function shall store this information for future trace streams created for this process.

RETURN VALUE

No return value is defined for the `posix_trace_event()` function.

Upon successful completion, the `posix_trace_eventid_open()` function shall return a value of zero. Otherwise, it shall return the corresponding error number. The `posix_trace_eventid_open()` function stores the trace event type identifier value in the object pointed to by `event_id`, if successful.

ERRORS

The `posix_trace_eventid_open()` function shall fail if:

ENAMETOOLONG

The size of the name pointed to by the `event_name` argument was longer than the implementation-defined value `{TRACE_EVENT_NAME_MAX}`.

The following sections are informative.

EXAMPLES

None.

APPLICATION USAGE

None.

RATIONALE

None.

FUTURE DIRECTIONS

The `posix_trace_event()` and `posix_trace_eventid_open()` functions may be removed in a future version.

SEE ALSO

Table 2-7, Trace Option: User Trace Event, `exec`, `posix_trace_eventid_equal()`, `posix_trace_start()`

The Base Definitions volume of POSIX.1-2017, `<sys_types.h>`, `<trace.h>`

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html>.

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.kernel.org/doc/man-pages/reporting_bugs.html.

IEEE/The Open Group 2017 POSIX_TRACE_EVENT(3P)