Full credit is given to the above companies including the OS that this PDF file was generated!

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

$ man posix_trace_eventid_equal.3p

POSIX_TRACE_EVENTID_EQUAL(3POSIX Programmer's ManPOSIX_TRACE_EVENTID_EQUAL(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_eventid_equal, posix_trace_eventid_get_name, posix_trace_trid_eventid_open ? manipulate the trace event type identi? fier (TRACING)

SYNOPSIS

```
#include <trace.h>
int posix_trace_eventid_equal(trace_id_t trid, trace_event_id_t event1,
    trace_event_id_t event2);
int posix_trace_eventid_get_name(trace_id_t trid,
    trace_event_id_t event, char *event_name);
int posix_trace_trid_eventid_open(trace_id_t trid,
    const char *restrict event_name,
    trace_event_id_t *restrict event);
```

DESCRIPTION

The posix_trace_eventid_equal() function shall compare the trace event type identifiers event1 and event2 from the same trace stream or the same trace log identified by the trid argument. If the trace event type

identifiers event1 and event2 are from different trace streams, the re?
turn value shall be unspecified.

The posix_trace_eventid_get_name() function shall return, in the argu?
ment pointed to by event_name, the trace event name associated with the
trace event type identifier identified by the argument event, for the
trace stream or for the trace log identified by the trid argument. The
name of the trace event shall have a maximum of {TRACE_EVENT_NAME_MAX}
characters (which has the minimum value {_POSIX_TRACE_EVENT_NAME_MAX}).
Successive calls to this function with the same trace event type iden?
tifier and the same trace stream identifier shall return the same event
name.

The posix_trace_trid_eventid_open() function shall associate a user
trace event name with a trace event type identifier for a given trace
stream. The trace stream is identified by the trid argument, and it
shall be an active trace stream. 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_trid_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 process being traced in the
trace stream identified by the trid argument, and is returned in the
variable pointed to by the event argument. If the user trace event name
has already been mapped for the traced process, then the previously as?
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_trid_even?
tid_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 the trace stream iden?
tified by the trid argument, and is returned in the variable pointed to
by the event 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 (see Table 2-7,
Trace Option: User Trace Event) user trace event shall be returned.

RETURN VALUE

Upon successful completion, the posix_trace_eventid_get_name() and
posix_trace_trid_eventid_open() functions shall return a value of zero.
Otherwise, they shall return the corresponding error number.
The posix_trace_eventid_equal() function shall return a non-zero value
if event1 and event2 are equal; otherwise, a value of zero shall be re?
turned. No errors are defined. If either event1 or event2 are not valid
trace event type identifiers for the trace stream specified by trid or
if the trid is invalid, the behavior shall be unspecified.
The posix_trace_eventid_get_name() function stores the trace event name
value in the object pointed to by event_name, if successful.
The posix_trace_trid_eventid_open() function stores the trace event
type identifier value in the object pointed to by event, if successful.

ERRORS

The posix_trace_eventid_get_name() and posix_trace_trid_eventid_open()
functions shall fail if:
EINVAL The trid argument was not a valid trace stream identifier.
The posix_trace_trid_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 posix_trace_eventid_get_name() function shall fail if:

EINVAL The trace event type identifier event was not associated with any name.

The following sections are informative.

EXAMPLES

None.

APPLICATION USAGE

None.

RATIONALE

None.

FUTURE DIRECTIONS

The posix_trace_eventid_equal(), posix_trace_eventid_get_name(), and posix_trace_trid_eventid_open() functions may be removed in a future version.

SEE ALSO

Table 2-7, Trace Option: User Trace Event, exec, posix_trace_event(), posix_trace_getnext_event()

The Base Definitions volume of POSIX.1?2017, <trace.h>

COPYRIGHT