NAME

rdma_get_cm_event - Retrieves the next pending communication event.

SYNOPSIS

#include <rdma/rdma_cma.h>

int rdma_get_cm_event (struct rdma_event_channel ****channel*, struct rdma_cm_event *****event***);**

ARGUMENTS

channel — Event channel to check for events.

event — Allocated information about the next communication event.

DESCRIPTION

Retrieves a communication event. If no events are pending, by default, the call will block until an event is received.

RETURN VALUE

Returns 0 on success, or -1 on error. If an error occurs, errno will be set to indicate the failure reason.

NOTES

The default synchronous behavior of this routine can be changed by modifying the file descriptor associated with the given channel. All events that are reported must be acknowledged by calling rdma_ack_cm_event. Destruction of an rdma_cm_id will block until related events have been acknowledged.

EVENT DATA

Communication event details are returned in the rdma_cm_event structure. This structure is allocated by the rdma_cm and released by the rdma_ack_cm_event routine. Details of the rdma_cm_event structure are given below.

id — The rdma_cm identifier associated with the event. If the event type is RDMA_CM_EVENT_CONNECT_REQUEST, then this references a new id for that communication.

listen_id — For RDMA_CM_EVENT_CONNECT_REQUEST event types, this references the corresponding listening request identifier.

event — Specifies the type of communication event which occurred. See EVENT TYPES below.

status Returns any asynchronous error information associated with an event. The status is zero if the operation was successful, otherwise the status value is non-zero and is either set to an errno or a transport specific value. For details on transport specific status values, see the event type information below.

param Provides additional details based on the type of event. Users should select the conn or ud subfields based on the rdma_port_space of the rdma_cm_id associated with the event. See UD EVENT DATA and CONN EVENT DATA below.

UD EVENT DATA

Event parameters related to unreliable datagram (UD) services: RDMA_PS_UDP and RDMA_PS_IPOIB. The UD event data is valid for RDMA_CM_EVENT_ESTABLISHED and RDMA_CM_EVENT_MULTICAST_JOIN events, unless stated otherwise.

private_data References any user-specified data associated with RDMA_CM_EVENT_CONNECT_REQUEST or RDMA_CM_EVENT_ESTABLISHED events. The data referenced by this field matches that specified by the remote side when calling rdma_connect or rdma_accept. This field is NULL if the event does not include private data. The buffer referenced by this pointer is deallocated when calling rdma_ack_cm_event.

private_data_len
The size of the private data buffer. Users should note that the size of the private data buffer may be larger than the amount of private data sent by the remote side. Any additional space in the buffer will be zeroed out.

ah_attr Address information needed to send data to the remote endpoint(s). Users should use this structure when allocating their address handle.

qp_num QP number of the remote endpoint or multicast group.

qkey QKey needed to send data to the remote endpoint(s).

CONN EVENT DATA

Event parameters related to connected QP services: RDMA_PS_TCP. The connection related event data is valid for RDMA_CM_EVENT_CONNECT_REQUEST and RDMA_CM_EVENT_ESTABLISHED events, unless stated otherwise.

private_data References any user-specified data associated with the event. The data referenced by this field matches that specified by the remote side when calling rdma_connect or

rdma_accept. This field is NULL if the event does not include private data. The buffer referenced by this pointer is deallocated when calling rdma_ack_cm_event.

private_data_len
: The size of the private data buffer. Users should note that the size of the private data buffer may be larger than the amount of private data sent by the remote side. Any additional space in the buffer will be zeroed out.

responder_resources
: The number of responder resources requested of the recipient. This field matches the initiator depth specified by the remote node when calling rdma_connect and rdma_accept.

initiator_depth
: The maximum number of outstanding RDMA read/atomic operations that the recipient may have outstanding. This field matches the responder resources specified by the remote node when calling rdma_connect and rdma_accept.

flow_control Indicates if hardware level flow control is provided by the sender.

retry_count For RDMA_CM_EVENT_CONNECT_REQUEST events only, indicates the number of times that the recipient should retry send operations.

rnr_retry_count
: The number of times that the recipient should retry receiver not ready (RNR) NACK errors.

srq Specifies if the sender is using a shared-receive queue.

qp_num Indicates the remote QP number for the connection.

EVENT TYPES

The following types of communication events may be reported.

RDMA_CM_EVENT_ADDR_RESOLVED
: Address resolution (rdma_resolve_addr) completed successfully.

RDMA_CM_EVENT_ADDR_ERROR
: Address resolution (rdma_resolve_addr) failed.

RDMA_CM_EVENT_ROUTE_RESOLVED
: Route resolution (rdma_resolve_route) completed successfully.

RDMA_CM_EVENT_ROUTE_ERROR
: Route resolution (rdma_resolve_route) failed.

RDMA_CM_EVENT_CONNECT_REQUEST
: Generated on the passive side to notify the user of a new connection request.

RDMA_CM_EVENT_CONNECT_RESPONSE
: Generated on the active side to notify the user of a successful response to a connection request. It is only generated on rdma_cm_id's that do not have a QP associated with them.

RDMA_CM_EVENT_CONNECT_ERROR
: Indicates that an error has occurred trying to establish or a connection. May be generated on the active or passive side of a connection.

RDMA_CM_EVENT_UNREACHABLE
: Generated on the active side to notify the user that the remote server is not reachable or unable to respond to a connection request. If this event is generated in response to a UD QP resolution request over InfiniBand, the event status field will contain an errno, if negative, or the status result carried in the IB CM SIDR REP message.

RDMA_CM_EVENT_REJECTED
: Indicates that a connection request or response was rejected by the remote end point. The event status field will contain the transport specific reject reason if available. Under InfiniBand, this is the reject reason carried in the IB CM REJ message.

RDMA_CM_EVENT_ESTABLISHED
: Indicates that a connection has been established with the remote end point.

RDMA_CM_EVENT_DISCONNECTED
: The connection has been disconnected.

RDMA_CM_EVENT_DEVICE_REMOVAL
: The local RDMA device associated with the rdma_cm_id has been removed. Upon receiving this event, the user must destroy the related rdma_cm_id.

RDMA_CM_EVENT_MULTICAST_JOIN
: The multicast join operation (rdma_join_multicast) completed successfully.

RDMA_CM_EVENT_MULTICAST_ERROR

An error either occurred joining a multicast group, or, if the group had already been joined, on an existing group. The specified multicast group is no longer accessible and should be rejoined, if desired.

RDMA_CM_EVENT_ADDR_CHANGE

The network device associated with this ID through address resolution changed its HW address, eg following of bonding failover. This event can serve as a hint for applications who want the links used for their RDMA sessions to align with the network stack.

RDMA_CM_EVENT_TIMEWAIT_EXIT

The QP associated with a connection has exited its timewait state and is now ready to be re-used. After a QP has been disconnected, it is maintained in a timewait state to allow any in flight packets to exit the network. After the timewait state has completed, the rdma_cm will report this event.

SEE ALSO

rdma_ack_cm_event(3), rdma_create_event_channel(3), rdma_resolve_addr(3), rdma_resolve_route(3), rdma_connect(3), rdma_listen(3), rdma_join_multicast(3), rdma_destroy_id(3), rdma_event_str(3)