

**NAME**

*udaddy* - RDMA CM datagram setup and simple ping-pong test.

**SYNOPSIS**

```
udaddy [-s server_address] [-b bind_address] [-c connections]
        [-C message_count] [-S message_size] [-p port_space]
udaddy -s server_address [-b bind_address] [-c connections]
        [-C message_count] [-S message_size] [-t tos] [-p port_space]
```

**DESCRIPTION**

Establishes a set of unreliable RDMA datagram communication paths between two nodes using the librdmacm, optionally transfers datagrams between the nodes, then tears down the communication.

**OPTIONS**

-s server\_address

The network name or IP address of the server system listening for communication. The used name or address must route over an RDMA device. This option must be specified by the client.

-b bind\_address

The local network address to bind to. To bind to any address with IPv6 use -b ::0 .

-c connections

The number of communication paths to establish between the client and server. The test uses unreliable datagram communication, so no actual connections are formed. (default 1)

-C message\_count

The number of messages to transfer over each connection. (default 10)

-S message\_size

The size of each message transferred, in bytes. This value must be smaller than the MTU of the underlying RDMA transport, or an error will occur. (default 100)

-t tos

Indicates the type of service used for the communication. Type of service is implementation dependent based on subnet configuration.

-p port\_space

The port space of the datagram communication. May be either the RDMA UDP (0x0111) or IPoIB (0x0002) port space. (default RDMA\_PS\_UDP)

**NOTES**

Basic usage is to start udaddy on a server system, then run udaddy -s server\_name on a client system.

Because this test maps RDMA resources to userspace, users must ensure that they have available system resources and permissions. See the libibverbs README file for additional details.

**SEE ALSO**

rdma\_cm(7), ucmatose(1), mckey(1), rping(1)