NAME

t_getinfo – get protocol-specific service information

SYNOPSIS

cc [*flag* ...] *file* ... **–lnsl** [*library* ...]

#include <tiuser.h>

int t_getinfo(int *fildes***, struct t_info** ∗*info***);**

MT-LEVEL

MT-Safe

DESCRIPTION

This function returns the current characteristics of the underlying transport protocol associated with file descriptor *fildes*. The *info* structure is used to return the same information returned by **t_open**(3N). This function enables a transport user to access this information during any phase of communication.

The *info* argument points to a **t_info** structure, which contains the following members:

```
long addr;       /* max size of transport protocol address */
long options;    /* max number of bytes of protocol-specific options */
long tsdu;       /* max size of transport service data unit (TSDU) */
long etsdu;      /* max size of expedited transport service data unit (ETSDU) */
long connect;    /* max amount of data allowed on connection establishment
                    functions */
long discon;     /* max amount of data allowed on t_snddis(3N) and
                    t_rcvdis(3N) functions */
long servtype;   /* service type supported by the transport provider */
```

The values of the fields have the following meanings:

addr A value greater than or equal to zero indicates the maximum size of a transport protocol address; a value of –1 specifies that there is no limit on the address size; and a value of –2 specifies that the transport provider does not provide user access to transport protocol addresses.

options A value greater than or equal to zero indicates the maximum number of bytes of protocol-specific options supported by the provider; a value of –1 specifies that there is no limit on the option size; and a value of –2 specifies that the transport provider does not support user-settable options.

tsdu A value greater than zero specifies the maximum size of a transport service data unit (**TSDU**); a value of zero specifies that the transport provider does not support the concept of **TSDU**, although it does support the sending across a connection of a data stream with no logical boundaries preserved; a value of –1 specifies that there is no limit on the size of a **TSDU**; and a value of –2 specifies that the transfer of normal data is not supported by the transport provider.

etsdu A value greater than zero specifies the maximum size of an expedited transport service data unit (**ETSDU**); a value of zero specifies that the transport provider does not support the concept of **ETSDU**, although it does support the sending across a connection of an expedited data stream with no logical boundaries preserved. A value of −1 specifies that there is no limit on the size of an **ETSDU**; and a value of −2 specifies that the transfer of expedited data is not supported by the transport provider.

connect A value greater than or equal to zero specifies the maximum amount of data that may be associated with connection establishment functions; a value of −1 specifies that there is no limit on the amount of data sent during connection establishment; and a value of −2 specifies that the transport provider does not allow data to be sent with connection establishment functions.

discon A value greater than or equal to zero specifies the maximum amount of data that may be associated with the **t_snddis**(3N) and **t_rcvdis**(3N) functions; a value of −1 specifies that there is no limit on the amount of data sent with these abortive release functions; and a value of −2 specifies that the transport provider does not allow data to be sent with the abortive release functions.

servtype This field specifies the service type supported by the transport provider, as described below.

If a transport user is concerned with protocol independence, the above sizes may be accessed to determine how large the buffers must be to hold each piece of information. Alternatively, the **t_alloc( )** function may be used to allocate these buffers. An error will result if a transport user exceeds the allowed data size on any function. The value of each field may change as a result of option negotiation, and **t_getinfo( )** enables a user to retrieve the current characteristics.

The **servtype** field of *info* may specify one of the following values on return:

T_COTS The transport provider supports a connection-mode service but does not support the optional orderly release facility.

T_COTS_ORD The transport provider supports a connection-mode service with the optional orderly release facility.

T_CLTS The transport provider supports a connectionless-mode service. For this service type, **t_open**(3N) will return −2 for **etsdu**, **connect**, and **discon**.

RETURN VALUES

t_getinfo( ) returns 0 on success. On failure **t_getinfo( )** returns −1, **t_errno** is set to indicate the error, and possibly **errno** is set.

ERRORS

On failure, **t_errno** will be set to one of the following:

TBADF The specified file descriptor does not refer to a transport endpoint.

TSYSERR A system error has occurred during execution of this function, **errno** will be set to the specific error.

SEE ALSO

t_open(3N), **t_rcvdis**(3N), **t_snddis**(3N)

Transport Interfaces Programming Guide

NOTES

This interface is safe in multithreaded applications.