

NAME

rpc_svc_err, svcerr_auth, svcerr_decode, svcerr_noproc, svcerr_noprog, svcerr_progvers, svcerr_systemerr, svcerr_weakauth — library routines for server side remote procedure call errors

SYNOPSIS

```
#include <rpc/rpc.h>

void
svcerr_auth(SVCXPRT *xpirt, enum auth_stat why);

void
svcerr_decode(SVCXPRT *xpirt);

void
svcerr_noproc(SVCXPRT *xpirt);

void
svcerr_noprog(SVCXPRT *xpirt);

void
svcerr_progvers(SVCXPRT *xpirt, rpcvers_t low_vers, rpcvers_t high_vers);

void
svcerr_systemerr(SVCXPRT *xpirt);

void
svcerr_weakauth(SVCXPRT *xpirt);
```

DESCRIPTION

These routines are part of the RPC library which allows C language programs to make procedure calls on other machines across the network.

These routines can be called by the server side dispatch function if there is any error in the transaction with the client.

Routines

See `rpc(3)` for the definition of the *SVCXPRT* data structure.

svcerr_auth()

Called by a service dispatch routine that refuses to perform a remote procedure call due to an authentication error.

svcerr_decode()

Called by a service dispatch routine that cannot successfully decode the remote arguments (see **svc_getargs()** in `rpc_svc_reg(3)`).

svcerr_noproc()

Called by a service dispatch routine that does not implement the procedure number that the caller requests.

svcerr_noprog()

Called when the desired program is not registered with the RPC package. Service implementors usually do not need this routine.

svcerr_progvers()

Called when the desired version of a program is not registered with the RPC package. The *low_vers* argument is the lowest version number, and *high_vers* is the highest version number.

Service implementors usually do not need this routine.

svcerr_systemerr()

Called by a service dispatch routine when it detects a system error not covered by any particular protocol. For example, if a service can no longer allocate storage, it may call this routine.

svcerr_weakauth()

Called by a service dispatch routine that refuses to perform a remote procedure call due to insufficient (but correct) authentication arguments. The routine calls **svcerr_auth**(*xprt*, *AUTH_TOOWEAK*).

AVAILABILITY

These functions are part of libtirpc.

SEE ALSO

rpc(3), **rpc_svc_calls(3)**, **rpc_svc_create(3)**, **rpc_svc_reg(3)**