NAME

sasl_client_start - Cyrus SASL documentation

SYNOPSIS

#include <sasl/sasl.h>

int sasl_client_start(sasl_conn_t * conn,
     const char * mechlist,
     sasl_interact_t ** prompt_need,
     const char ** clientout,
     unsigned * clientoutlen,
     const char ** mech);

DESCRIPTION

int sasl_client_start(sasl_conn_t * conn,

const char * mechlist,

sasl_interact_t ** prompt_need,

const char ** clientout,

unsigned * clientoutlen,

const char ** mech);

sasl_client_start() selects a mechanism for authentication and starts the authentication session. The mechlist is the list of mechanisms the client might like to use. The mech- anisms in the list are not necessarily supported by the client or even valid. SASL determines which of these to use based upon the security preferences specified earlier. The list of mechanisms is typically a list of mechanisms the server supports acquired from a capability request.

If **SASL_INTERACT** is returned the library needs some values to be filled in before it can proceed. The *prompt_need* structure will be filled in with requests. The application should fulfill these requests and call sasl_client_start again with identical parameters (the *prompt_need* parameter will be the same pointer as before but filled in by the application).

Parameters

  ⊕ **conn** - is the SASL connection context

⊕ **mechlist** - is a list of mechanisms the server has available. Punctuation is ignored.

⊕ **prompt_need** - is filled in with a list of prompts needed to continue (if necessary).

⊕ **clientout** -

is created. It is the initial client response to send to the server. It is the job of the client to send it over the network to the server. Any protocol specific encoding (such as base64 encoding) necessary needs to be done by the client.

If the protocol lacks client-send-first capability, then set clientout to NULL.

If there is no initial client-send, then *clientout will be set to NULL on return.

⊕ **clientoutlen** - length of *clientout*.

⊕ **mech** - contains the name of the chosen SASL mechanism (on success)

RETURN VALUE

SASL callback functions should return SASL return codes. See sasl.h for a complete list. **SASL_CONTINUE** indicates success and that there are more steps needed in the authentication.

Other return codes indicate errors and should either be handled or the authentication session should be quit.

SEE ALSO

RFC 4422,:saslman:*sasl(3)*, sasl_callbacks(3), sasl_client_init(3), sasl_client_new(3), sasl_client_step(3), sasl_errors(3)

AUTHOR

The Cyrus Team

COPYRIGHT

1993-2016, The Cyrus Team