

NAME

sasl_server_step - Cyrus SASL documentation

SYNOPSIS

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

```
int sasl_server_step(sasl_conn_t *conn,  
    const char *clientin,  
    unsigned clientinlen,  
    const char ** serverout,  
    unsigned * serveroutlen);
```

DESCRIPTION

```
int sasl_server_step(sasl_conn_t *conn,  
  
const char *clientin,  
  
unsigned clientinlen,  
  
const char ** serverout,  
  
unsigned * serveroutlen);
```

sasl_server_step() performs a step in the authentication negotiation. It returns **SASL_OK** if the whole negotiation is successful and **SASL_CONTINUE** if this step is ok but at least one more step is needed.

Parameters

- ⊕ **conn** - is the SASL connection context
- ⊕ **clientin** - is the data given by the client (decoded if the protocol encodes requests sent over the wire)
- ⊕ **clientinlen** - is the length of *clientin*
- ⊕ **serverout** - set by the library and should be sent to the client.
- ⊕ **serveroutlen** - length of *serverout*.

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.

SASL_OK indicates that the authentication is complete.

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_server_init(3)*, *sasl_server_new(3)*, *sasl_server_start(3)*, *sasl_errors(3)*

AUTHOR

The Cyrus Team

COPYRIGHT

1993-2016, The Cyrus Team