

NAME

ldap_perror, ld_errno, ldap_result2error, ldap_errlist, ldap_err2string - LDAP protocol error handling routines

LIBRARY

OpenLDAP LDAP (libldap, -lldap)

SYNOPSIS

```
#include <ldap.h>
```

```
char *ldap_err2string( int err );
```

DESCRIPTION

The **ldap_err2string()** routine provides short description of the various codes returned by routines in this library. The returned string is a pointer to a static area that should not be modified.

These codes are either negative, indicating an API error code; positive, indicating an LDAP resultCode other than 'success' (0), or - zero, indicating both successful use of the API and the LDAP resultCode 'success' (0).

The code associated with an LDAP session is accessible using **ldap_get_option(3)** and **ldap_set_option(3)** with the **LDAP_OPT_RESULT_CODE** option (previously called **LDAP_OPT_ERROR_NUMBER**).

PROTOCOL RESULT CODES

This section provides a partial list of protocol codes recognized by the library. As LDAP is extensible, additional values may be returned. A complete listing of *registered* LDAP result codes can be obtained from the *Internet Assigned Numbers Authority* <<http://www.iana.org>>.

LDAP_SUCCESS The request was successful.

LDAP_OPERATIONS_ERROR
 An operations error occurred.

LDAP_PROTOCOL_ERROR
 A protocol violation was detected.

LDAP_TIMELIMIT_EXCEEDED

An LDAP time limit was exceeded.

LDAP_SIZELIMIT_EXCEEDED

An LDAP size limit was exceeded.

LDAP_COMPARE_FALSE

A compare operation returned false.

LDAP_COMPARE_TRUE

A compare operation returned true.

LDAP_STRONG_AUTH_NOT_SUPPORTED

The LDAP server does not support strong authentication.

LDAP_STRONG_AUTH_REQUIRED

Strong authentication is required for the operation.

LDAP_PARTIAL_RESULTS

Partial results only returned.

LDAP_NO_SUCH_ATTRIBUTE

The attribute type specified does not exist in the entry.

LDAP_UNDEFINED_TYPE

The attribute type specified is invalid.

LDAP_INAPPROPRIATE_MATCHING

Filter type not supported for the specified attribute.

LDAP_CONSTRAINT_VIOLATION

An attribute value specified violates some constraint (e.g., a postalAddress has too many lines, or a line that is too long).

LDAP_TYPE_OR_VALUE_EXISTS

An attribute type or attribute value specified already exists in the entry.

LDAP_INVALID_SYNTAX

An invalid attribute value was specified.

LDAP_NO_SUCH_OBJECT

The specified object does not exist in The Directory.

LDAP_ALIAS_PROBLEM

An alias in The Directory points to a nonexistent entry.

LDAP_INVALID_DN_SYNTAX

A syntactically invalid DN was specified.

LDAP_IS_LEAF

The object specified is a leaf.

LDAP_ALIAS_DEREF_PROBLEM

A problem was encountered when dereferencing an alias.

LDAP_INAPPROPRIATE_AUTH

Inappropriate authentication was specified (e.g., LDAP_AUTH_SIMPLE was specified and the entry does not have a userPassword attribute).

LDAP_INVALID_CREDENTIALS

Invalid credentials were presented (e.g., the wrong password).

LDAP_INSUFFICIENT_ACCESS

The user has insufficient access to perform the operation.

LDAP_BUSY

The DSA is busy.

LDAP_UNAVAILABLE

The DSA is unavailable.

LDAP_UNWILLING_TO_PERFORM

The DSA is unwilling to perform the operation.

LDAP_LOOP_DETECT

A loop was detected.

LDAP_NAMING_VIOLATION

A naming violation occurred.

LDAP_OBJECT_CLASS_VIOLATION

An object class violation occurred (e.g., a "must" attribute was missing from the entry).

LDAP_NOT_ALLOWED_ON_NONLEAF

The operation is not allowed on a nonleaf object.

LDAP_NOT_ALLOWED_ON_RDN

The operation is not allowed on an RDN.

LDAP_ALREADY_EXISTS

The entry already exists.

LDAP_NO_OBJECT_CLASS_MODS

Object class modifications are not allowed.

LDAP_OTHER

An unknown error occurred.

API ERROR CODES

This section provides a complete list of API error codes recognized by the library. Note that `LDAP_SUCCESS` indicates success of an API call in addition to representing the return of the LDAP 'success' resultCode.

LDAP_SERVER_DOWN

The LDAP library can't contact the LDAP server.

LDAP_LOCAL_ERROR

Some local error occurred. This is usually a failed dynamic memory allocation.

LDAP_ENCODING_ERROR

An error was encountered encoding parameters to send to the LDAP server.

LDAP_DECODING_ERROR

An error was encountered decoding a result from the LDAP server.

LDAP_TIMEOUT

A timelimit was exceeded while waiting for a result.

LDAP_AUTH_UNKNOWN

The authentication method specified to `ldap_bind()` is not known.

LDAP_FILTER_ERROR

An invalid filter was supplied to `ldap_search()` (e.g., unbalanced parentheses).

LDAP_PARAM_ERROR

An `ldap` routine was called with a bad parameter.

LDAP_NO_MEMORY An memory allocation (e.g., `malloc(3)` or other dynamic memory allocator) call failed in an `ldap` library routine.

LDAP_USER_CANCELED

Indicates the user cancelled the operation.

LDAP_CONNECT_ERROR

Indicates a connection problem.

LDAP_NOT_SUPPORTED

Indicates the routine was called in a manner not supported by the library.

LDAP_CONTROL_NOT_FOUND

Indicates the control provided is unknown to the client library.

LDAP_NO_RESULTS_RETURNED

Indicates no results returned.

LDAP_MORE_RESULTS_TO_RETURN

Indicates more results could be returned.

LDAP_CLIENT_LOOP

Indicates the library has detected a loop in its processing.

LDAP_REFERRAL_LIMIT_EXCEEDED

Indicates the referral limit has been exceeded.

DEPRECATED

Deprecated interfaces generally remain in the library. The macro `LDAP_DEPRECATED` can be defined to a non-zero value (e.g., `-DLLDAP_DEPRECATED=1`) when compiling program designed to use deprecated interfaces. It is recommended that developers writing new programs, or updating old programs, avoid use of deprecated interfaces. Over time, it is expected that documentation (and, eventually, support) for deprecated interfaces to be eliminated.

SEE ALSO

ldap(3),

ACKNOWLEDGEMENTS

OpenLDAP Software is developed and maintained by The OpenLDAP Project

<<http://www.openldap.org/>>. **OpenLDAP Software** is derived from the University of Michigan LDAP 3.3 Release.