

**NAME**

ldap\_control\_create, ldap\_control\_find, ldap\_control\_dup, ldap\_controls\_dup, ldap\_control\_free, ldap\_controls\_free - LDAP control manipulation routines

**LIBRARY**

OpenLDAP LDAP (libldap, -lldap)

**SYNOPSIS**

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

```
int ldap_control_create(const char *oid, int iscritical, struct berval *value, int dupval, LDAPControl **ctrlp);
```

```
LDAPControl *ldap_control_find( const char *oid, LDAPControl **ctrls, LDAPControl ***nextctrlp);
```

```
LDAPControl *ldap_control_dup(LDAPControl *ctrl);
```

```
LDAPControl **ldap_controls_dup(LDAPControl **ctrls);
```

```
void ldap_control_free(LDAPControl *ctrl);
```

```
void ldap_controls_free(LDAPControl **ctrls);
```

**DESCRIPTION**

These routines are used to manipulate structures used for LDAP controls.

**ldap\_control\_create()** creates a control with the specified *OID* using the contents of the *value* parameter for the control value, if any. The content of *value* is duplicated if *dupval* is non-zero. The *iscritical* parameter must be non-zero for a critical control. The created control is returned in the *ctrlp* parameter. The routine returns **LDAP\_SUCCESS** on success or some other error code on failure. The content of *value*, for supported control types, can be prepared using helpers provided by this implementation of libldap, usually in the form **ldap\_create\_<control name>\_control\_value()**. Otherwise, it can be BER-encoded using the functionalities of liblber.

**ldap\_control\_find()** searches the NULL-terminated *ctrls* array for a control whose *OID* matches the *oid* parameter. The routine returns a pointer to the control if found, NULL otherwise. If the parameter *nextctrlp* is not NULL, on return it will point to the next control in the array, and can be passed to the **ldap\_control\_find()** routine for subsequent calls, to find further occurrences of the same control type. The use of this function is discouraged; the recommended way of handling controls in responses consists in going through the array of controls, dealing with each of them in the returned order, since it

could matter.

**ldap\_control\_dup()** duplicates an individual control structure, and **ldap\_controls\_dup()** duplicates a NULL-terminated array of controls.

**ldap\_control\_free()** frees an individual control structure, and **ldap\_controls\_free()** frees a NULL-terminated array of controls.

## SEE ALSO

**ldap(3)**, **ldap\_error(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.