## NAME

idmap_rfc2307 − Samba's idmap_rfc2307 Backend for Winbind

## DESCRIPTION

The idmap_rfc2307 plugin provides a way for winbind to read id mappings from records in an LDAP server as defined in RFC 2307. The LDAP server can be stand−alone or the LDAP server provided by the AD server. An AD server is always required to provide the mapping between name and SID, and the LDAP server is queried for the mapping between name and uid/gid. This module implements only the "idmap" API, and is READONLY.

Mappings must be provided in advance by the administrator by creating the user accounts in the Active Directory server and the posixAccount and posixGroup objects in the LDAP server. The names in the Active Directory server and in the LDAP server have to be the same.

This id mapping approach allows the reuse of existing LDAP authentication servers that store records in the RFC 2307 format.

When connecting to the LDAP server provided by an AD server, the parameter **ldap ssl ads** determines whether SSL should be used. When using a stand−alone LDAP server, **ldap ssl** applies.

## IDMAP OPTIONS

range = low − high
: Defines the available matching UID and GID range for which the backend is authoritative. Note that the range acts as a filter. If specified any UID or GID stored in AD that fall outside the range is ignored and the corresponding map is discarded. It is intended as a way to avoid accidental UID/GID overlaps between local and remotely defined IDs.

ldap_server = <ad | stand−alone >
: Defines the type of LDAP server to use. This can either be the LDAP server provided by the Active Directory server (ad) or a stand−alone LDAP server.

bind_path_user
: Specifies the search base where user objects can be found in the LDAP server.

bind_path_group
: Specifies the search base where group objects can be found in the LDAP server.

user_cn = <yes | no>
: Query cn attribute instead of uid attribute for the user name in LDAP. This option is not required, the default is no.

realm
: Append @realm to cn for groups (and users if user_cn is set) in LDAP queries. This option is not required, the default is not to append the realm.

ldap_domain
: When using the LDAP server in the Active Directory server, this allows one to specify the domain where to access the Active Directory server. This allows using trust relationships while keeping all RFC 2307 records in one place. This parameter is optional, the default is to access the AD server in the current domain to query LDAP records.

ldap_url
: When using a stand−alone LDAP server, this parameter specifies the ldap URL for accessing the LDAP server.

ldap_user_dn
: Defines the user DN to be used for authentication. The secret for authenticating this user should be stored with net idmap secret (see **net**(8)). If absent, an anonymous bind will be performed.

## EXAMPLES

The following example shows how to retrieve id mappings from a stand−alone LDAP server. This example also shows how to leave a small non conflicting range for local id allocation that may be used in internal backends like BUILTIN.

```
[global]
idmap config * : backend = tdb
idmap config * : range = 1000000−1999999

idmap config DOMAIN : backend = rfc2307
idmap config DOMAIN : range = 2000000−2999999
idmap config DOMAIN : ldap_server = stand−alone
idmap config DOMAIN : ldap_url = ldap://ldap1.example.com
idmap config DOMAIN : ldap_user_dn = cn=ldapmanager,dc=example,dc=com
idmap config DOMAIN : bind_path_user = ou=People,dc=example,dc=com
idmap config DOMAIN : bind_path_group = ou=Group,dc=example,dc=com
```

## AUTHOR

The original Samba software and related utilities were created by Andrew Tridgell. Samba is now developed by the Samba Team as an Open Source project similar to the way the Linux kernel is developed.