

LDAP Authentication

- Introduction
- What is LDAP?
- Why use LDAP?
- Credits
- Setting Up An OpenLDAP Server
- Install Master Server
- Secure Master Server with SSL/TLS
- Populate Master Server
- Summary of User Object Classes and Attributes Table
- Install Slave Servers
- Configure UNIX Hosts for LDAP Logins
- LDAP enable Samba PDC For NT Workstation Logins
- Configure NT Workstations for LDAP Logins
- LDAP enable Samba PDC for NT Workstation Logins
- Configure NT Workstations for LDAP Logins Introduction What is LDAP:

LDAP stands for Lightweight Directory Access Protocol. OpenLDAP is an Open Source project that uses LDAP to deliver a fast, free, distributed directory service to organisations without locking them into dependence upon a single software vendor.

For example, prior to the use of centralized directory services separate directories were required for the domain itself, mailboxes, remote access, databases, and other applications. OpenLDAP enables a systems administrator to make a single entry in a directory which then gives a user account access to the network, access their email, access to corporate CRM systems or other mission-critical applications. In short, by using OpenLDAP as a multi-purpose directory an organisation enables single sign-on for its users. Once a user is authenticated by the network using OpenLDAP they will automatically unlock all of the applications or services that they have been enabled for. Why use LDAP?:

- Single source of authentication.
- Simplifies administration.
- Obviates need for baroque/homebrew user account replication.
- Replacement for NIS, /etc/passwd, /etc/groups, /etc/shadow, NT users/groups, etc.
- Platform agnostic.
- Can be back-end for authenticating most services (email, ftp, proxy services, etc).
- Can be used for much more than just authentication (HR, phone lists, address books, etc).

Basically, the process is:

- Set up OpenLDAP server
- Configure UNIX Hosts for LDAP Logins
- LDAP enable Samba PDC For NT/2000 Workstations Credits:

The following people were involved in the production of this HOWTO:

Regan Burke Setting Up An OpenLDAP Server Install Master Server:

The latest stable version (currently 2.1.16) of OpenLDAP can be downloaded from [here](#).

The Administrators Guide (essential reading) may be downloaded from [here](#).

Source Install Under Solaris 8: Solaris 8 is quite broken out of the box. We strongly suggest you immediately implement the sage advice found in `fixsolaris8.txt` before continuing. Download the latest stable release

```
./configure
make depend
make
make install
```

Source Install Under Debian GNU/Linux 3.0:

You will need the following packages installed prior to building the OpenLDAP server (apt-get is your friend!):

- gcc
- automake
- autoconf
- autotools-dev
- m4
- autoconf2.13
- libtool

- libtdl3
- libtdl3-dev

You will also need to build the Berkeley DB software from source

- Download latest stable release (currently 4.1.25) from here and un-gzip it into wherever you build your local software (say /usr/local/src)
- cd to the 'build_unix' directory
- ../dist/configure
- make
- make install
- Download the latest stable release and un-gzip it into wherever you build your local software (say /usr/local/src)
- ./configure
- make depend
- make
- make install Non-Source Install Under Linux:

Debian:

The required packages are:

- libldap2
- slapd
- libiodbc2
- ldap-utils

Install using apt-get or dselect.

Configure according to the debian-specific documentation.

The slapd.conf file may require some further editing.

Firstly you will need to decide which schemas to include.

The core.schema must always be included.

If you wish to support Solaris 8, you must include the cosine.schema, nis.schema, inetorgperson.schema and the solaris.schema.

The nis.schema supplied with the distribution is borked. Please please contact us for a replacement.

If you intend to support Solaris 8, please contact us.

If you intend to use Samba, please contact us for the correct schema.

They should be placed in /etc/ldap/schema along with the others.

The latest Debian unstable also has migrationtools. Secure Master Server with SSL/TLS:

This section has not been completed. please contact us for further details. Populate Master Server:

How the server is populated depends on which operating systems you wish to support authentication on.

Currently, this howto covers Linux, FreeBSD, Solaris 8 and the various incarnations of Microsoft Windows.

It is possible (and desirable) to keep non-authentication related information in your server, such as addresses, email details, and even staff photos(!) if you want to.

You will also have to make decisions on the structure of your LDAP server in terms of groupings of entries.

- Set up base dn with appropriate object classes and attributes.
- Set up organisational units (e.g. People, Groups, Machines, etc.) with appropriate object classes and attributes.
- Add individual entries for People, Groups, Machines, etc.

Please contact us for an example setup file to create the top level plus our suggested organisational units.

Obviously, you will need to customise this file to match your own organisation.

When you are happy with it, it can be installed to the ldap server by issuing the command

"ldapadd -f setup.ldif -h yourserver -x -D cn=admin,dc=siriusit,dc=co,dc=uk -w p455w0rd" as root.

Where 'yourserver', the domain components (dc's), and p455w0rd are suitably adjusted for your site. yourserver may be a hostname or an ip address. User Accounts:

The minimum object classes needed for UN*X clients to authenticate UN*X users against the LDAP server are top, posixAccount and shadowAccount.

The definitive guide to these (and other) object classes is RFC 2307. Suggested object classes to include for a user:

```
top
person
organizationalPerson
inetOrgPerson
```

account
 posixAccount
 shadowAccount
 sambaAccount

Change the value of the following parameters to 2147483647

logoffTime
 kickoffTime

| Attributes | | Required? | | Description | | Objectclass | |
|-------------------------------|----------------------------|----------------|-----------|--------------------------------------|-------------------------|---------------------------|--------------------|
| | Object | Class | Required? | Description | Objectclass | top | ObjectClass |
| required | Basic | person | required | Name | person | | cn |
| required | Common | person | required | Surname | | | sn |
| optional | | | optional | UN*X Password | | | userPassword |
| telephoneNumber | seeAlso | | optional | Phone Number | | | |
| | | | optional | description | Reference | | related LDAP entry |
| Description/Special | Interests | | | | | organizationalPerson | |
| | Extended | Person | | | | | |
| title | optional | Organisational | | Title | | | |
| x121Address | optional | Naming | | Standard defined in CCITT Rec. X.121 | | | |
| | registeredAddress | | optional | Address where recipient | destinationIndicator | optional | |
| guarantees to accept delivery | Used for telegrams | | | value favourite delivery method | preferredDeliveryMethod | | |
| optional | Single | | optional | | | | |
| telexNumber | optional | | optional | telephoneNumber | optional | teletexTerminalIdentifier | optional |
| | | | | internationalISDNNumber | optional | | |
| | street | optional | | facsimileTelephoneNumber | optional | | |
| | optional | | | | | postalCode | optional |
| | | | | postalAddress | optional | | postOfficeBox |
| | physicalDeliveryOfficeName | optional | | | | | |
| | | | | Extended Internet Person | | | optional |
| inetOrgPerson | optional | | | | | | |
| businessCategory | optional | | optional | audio | | departmentNumber | optional |
| | | | | display_name | optional | | carLicence |
| employeeType | optional | employeeNumber | optional | | | | |
| | initial | optional | | homePostalAddress | optional | homePhone | optional |
| | | | | | | labeledURI | optional |
| manager | optional | mail | optional | | | mobile | optional |
| | pager | optional | | | | | |
| | | | | roomNumber | optional | photo | optional |
| | secretary | optional | | | | | |
| uid | optional | | | x500uniqueIdentifier | optional | userCertificate | optional |
| | | | | preferredLanguage | optional | | |
| userPKCS12 | optional | Basic Account | | | | account | optional |
| required | see uid (below) | | | | | userid | optional |
| | localityName | optional | | seeAlso | optional | | |
| organizationName | optional | | | | | | host |
| organizationalUnitName | optional | | | | | | |

optional Â posixAccount Â Â
Unix Account Â cn Â required uid Â
Common Name (if a person, their full name) Â uidNumber
required User's login Â uidNumber
Â required Number that uniquely identifies user to system
Â gidNumber Â required Number that uniquely identifies group to system
directory Â homeDirectory Â required Path to user's home
User's UN*X Password Â loginShell
optional UN*X login shell Â Â gecos
optional Â Â description Â optional
Shadow Account shadowAccount Â Â Unix
userPassword Â uid Â required Â
shadowLastChanges Â optional Â
shadowMin Â optional Â
shadowMax Â optional Â
shadowWarning Â optional Â
shadowInactive Â optional Â
shadowExpire Â optional Â
optional Â Â description Â
Windows User Account Â Â Â Â
identifier Â Â Â Â
ntPassword Â optional Â
logonTime Â optional Â
logoffTime Â optional Â
optional Â Â Â Â
Â Â Â Â kickoffTime
optional Â Â Â Â
Â Â Â Â pwdCanChange
Â Â Â Â pwdMustChange Â optional
Â Â Â Â acctFlags Â optional
Â Â Â Â Â Â
scriptPath Â Â Â Â
profilePath Â optional Â Â Â
optional Â Â Â Â description
optional Â Â Â Â userWorkStations
Â Â Â Â primaryGroupID Â optional
Â Â Â Â domain Â optional

Workstations:
posixAccount
sambaAccount

cn
uid
uidNumber
gidNumber
homeDirectory
rid

clean profiles directory
home directory Group Accounts:

The minimum object classes needed for clients to retrieve groups from the LDAP server are top and posixGroup.

Summary of Group Object Classes and Attributes:
Table with columns: Required?, Description, Objectclass, Attributes. Rows include Group, posixGroup, and Unix.

| Objectclass | Attributes | Required? | Description |
|-------------|---------------------|-----------|---------------|
| top | Object | default | Unix Account |
| required | uidNumber | required | uid |
| optional | gidNumber | required | UN*X uid |
| optional | UN*X login shell | required | UN*X gid |
| optional | UN*X home directory | required | UN*X password |
| optional | UN*X password | optional | description |
| optional | description | optional | gecos |
| optional | description | optional | loginShell |
| optional | description | optional | homeDirectory |
| optional | description | optional | userPassword |
| optional | description | optional | optional |
| optional | description | optional | Install Slave |

Servers:

This ain't done either ;) Configure UNIX Hosts for LDAP Logins

This is achieved by using PADL Software's open source pam_ldap and nss_ldap modules.

The pam_ldap module enables the client operating system to authenticate against your LDAP server, whilst the nss_ldap module enables the client operating system to retrieve session information (GECOS field type information) from your LDAP server.

Configuration is different under each client operating system, and depends upon whether you compile from source or use precompiled modules if available.

A basic intro to PAM may be downloaded (pdf format) from [here](#)

Further detail on the Linux implementation of PAM may be found [here](#)

Sun (who started it all) have a web page [here](#)

The Linux-PAM System Administrators' Guide may be found [here](#) Source Install Under Solaris 8:

Make sure that the following programs from Sunfreeware are installed:

- autoconf
- automake
- libtool
- libiconv
- m4

Download PADL Software's pam_ldap and nss_ldap modules source code. Put them in /usr/local/src, or wherever you keep stuff you compile locally.

Back up your existing /usr/lib/nss_ldap.so.1 and /usr/lib/security/pam_ldap.so.1 files.

cd into the pam_ldap-152 directory.

Run "./configure"

Run "aclocal"

Edit the Makefile to remove "-llber" on line 115

Run "make"

Run "make install"

cd into the nss_ldap-201 directory.

Run "./configure"

Edit the Makefile to remove "-llber" on line 157

Run "make"

Run "chmod +x install-sh"

Run "make install"

Edit /etc/ldap.conf to suit your site.

add ldap.secret

Edit /etc/nsswitch.conf.

Finally, the PAM configuration file must be modified.

Modify the configuration file /etc/pam.conf - for each service you wish to LDAP-enable.

The modifications are quite simple, generally involving adding lines of the form:

type sufficient pam_ldap.so

where type is one of account, auth, password, or session.

`/etc/init.d/nscd restart`

The name service caching daemon (nscd) caches LDAP lookups locally to speed up authentication. There is a problem with synchroniation though. Non-Source Install Under Linux: Debian:

The required packages are:

- libldap2
- libpam0g
- libpam-ldap
- libnss-ldap
- nscd

Install using `apt-get` or `dselect`.

Debconf will ask you a few questions:

- Configuring Libnss-ldap - LDAP Server host - enter the ip address of your ldap server.

Edit `/etc/pam_ldap.conf` to suit your site.

The admin passwords needs to be in the file `/etc/ldap.secret`. `chmod` this file to 0600.

Edit `/etc/libnss-ldap.conf` to suit your site.

Edit `/etc/nsswitch.conf` to suit your site.

Finally, the PAM configuration files must be modified.

`cd` to `/etc/pam.d/` and modify the configuration file for each service you wish to LDAP-enable.

The modifications are quite simple, generally involving adding lines of the form: