

Introduction to LDAP

LDAP (*Lightweight Directory Access Protocol*) is a standard [protocol](#) allowing directories to be managed, i.e. access information bases on the users of a network using [TCP/IP](#) protocols.

The information bases are generally related to the users, but are sometimes used for other purposes such as managing a company's hardware.

The aim of the LDAP protocol, developed in 1993 by the University of Michigan, was to replace the DAP protocol (used to access X.500 directory services by OSI) by integrating according to TCP/IP. From 1995, DAP became a *standalone LDAP* so that it was no longer used only to access X500 type directories. LDAP is thus a lighter version of the DAP protocol, hence its name of **Lightweight Directory Access Protocol**.

Presentation of LDAP

The LDAP protocol defines the method of accessing data on the server at client level, and not the manner in which the information is stored.

LDAP protocol is currently at version 3 and has been standardised by the IETF (Internet Engineering Task Force). So, there is a [RFC](#) for each version of LDAP, making up a reference document:

- [RFC 1777](#) for LDAP v.2
- [RFC 2251](#) for LDAP v.3

So LDAP supplies the user with methods enabling him to:

- connect
- disconnect
- search for information
- compare information
- insert entries
- change entries
- delete entries

Furthermore, LDAP protocol (in version 3) offers encryption ([SSL](#), ...) and authentication mechanisms allowing secure access to information stored in the base.

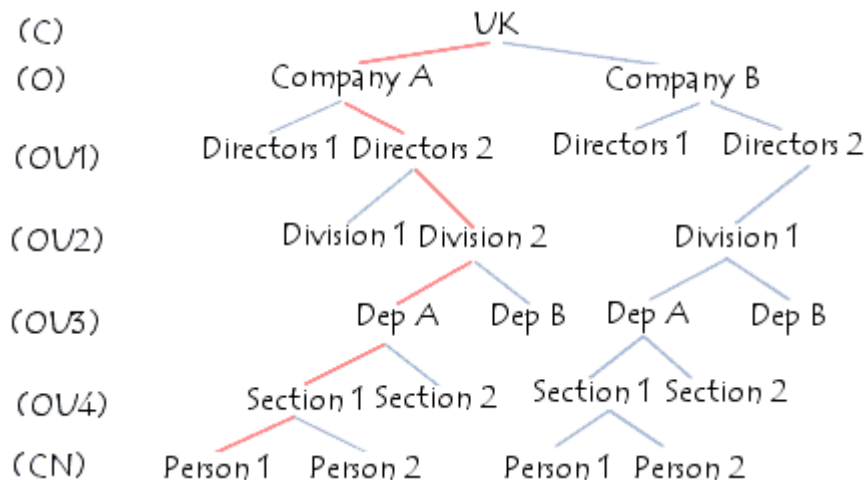
Information tree structure (DIT)

LDAP presents information in the form of a hierarchical tree structure called a **DIT** (*Directory Information Tree*), in which the information, called **entries** (or even *DSE*, *Directory Service Entry*), is represented in branches.

A branch located at the root of a branch is called the *root entry*.

Each entry in the LDAP directory relates to an abstract or real object (for example a person, a piece of hardware, parameters, etc.).

Each entry is made up of a collection of key/value pairs called **attributes**.



Entry attributes

Each entry is made up of a collection of attributes (key/value pairs) enabling the object that the entry defines to be distinguished. There are two types of attributes:

- **Normal attributes:** these are the usual attributes (surname, name, ...) distinguishing the object.
- **Operational attributes:** these are the attributes which only the server can access in order to manipulate the directory data (modification dates, etc.)

An entry is indexed by a **distinguished name (DN)** enabling an item in the tree structure to be uniquely identified.

A DN consists of taking the name of the element, called the *Relative Distinguished Name (RDN)*, i.e. the path of the entry in relation to its parents), and adding the entire name of the parent entry to it.

It is a question of using a series of key/value pairs making it possible to uniquely locate an entry. Here is a series of keys which are generally used:

- **uid** (*userid*), this is a compulsory unique ID
- **cn** (*common name*), this is the person's name
- **givenname**, this is the person's first name
- **sn** (*surname*), this is the person's surname
- (*organization*), this is the person's company
- **u** (*organizational unit*), this is the department of the company in which the person works
- **mail**, this is the email address of the person (of course)

So a *Distinguished Name* will take the form:

```
uid=jeapil,cn=pillou,givenname=jean-francois
```

Le *Relative Distinguished Name* étant ici "uid=jeapil".

Thus, the collection of object and attribute definitions that a LDAP server can manage is called a **schema**. This makes it possible, for example to define if an attribute can possess one or several values. Furthermore, an attribute called *objectclass* makes it possible to define whether attributes are compulsory or optional...

Consulting data

LDAP provides a collection of functions (procedures) to carry out queries on the data in order to search for, change and delete entries in the directories.

Here is the list of the main operations that LDAP can perform:

Operation	Description
Abandon	Abandon the previous operation sent to the server
Add	Add an entry to the directory
Bind	Start a new session on the LDAP server
Compare	Compare the entries in a directory according to the criteria
Delete	Delete an entry from a directory
Extended	Carry out extended operations
Rename	Change the name of an entry
Search	Search for entries in a directory
Unbind	End a new session on the LDAP server

LDAP data interchange format

LDAP provides a data interchange format (**LDIF**, *Lightweight Data Interchange Format*) allowing data to be imported and exported from a directory using a simple text file. The majority of LDAP servers support this format, which allows great interoperability between them.

The syntax for this format is as follows:

```
[<id>]
dn: <distinguished name>
<attribute>: <value>
<attribute>: <value>
...
```

In this file, *id* is optional, it is a positive whole number allowing the entry in the database to be identified.

Each new entry must be separated from the previous entry definition using an empty line.

It is possible to define an attribute over several lines by beginning the following lines by a space or tab space.



It is possible to define several values for an attribute by repeating the string *name:value* on the separated lines.

When the value contains a special character (non printable, a space or *.*), the attribute must be followed by *::* then the value encoded in base64.