LDAP

Author: Tam Nguyen
Class: CS491B

What is LDAP?

- Lightweight Directory Access Protocol
- Directories are aimed at the problem of finding things.
- Considered a phone book and/or mall maps.
- http://People.yahoo.com
- http://www.anywho.com
- http://Whowhere.lycos.com

LDAP Features

- Replicate some or all data via the push or pull method allowing you to push data to remote office.
- LDIF
- Directory Synchronization
- Distributed Directory
- TLS (Transport Layer Security) provide secure access and encryption capabilities between client and server.
- SASL (Simple Authentication and Security Layer) allows client and server negotiate authentication method.
- AAA
- API for programming language.
- JNDI and ADSI.

Advantage of LDAP

- Make network administration easier
  - Central management of people information and user accounts.
  - Reduced support costs.
- Unify access to network resources
  - single login to network resources including web services.
- Provide single destination for users to search for information.
  - Contact information.

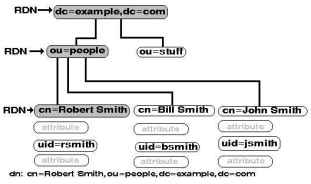
LDAP vs. DATABASES

- Directories are optimized for read-focused rather than write-focused.
- Directory transactions involve only a single operation and a single directory entry.
- Databases are designed to handle large and diverse transactions, spanning multiple data items and many operations.
- Databases provide data that can be easily manipulated and sustain intense processing, with both reading and writing.

Keys vs. RDN

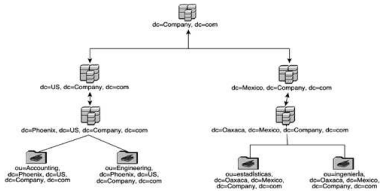
- In database system, keys uniquely identify attribute.
- In ldap, RDN (Relative Distinguished name) attribute provides a unique name identifier for each entry within a container.
- There cannot be two entries with the same RDN value within the same container.

RDN



LDAP Tree

- LDAP Tree Structure



LDAP Information Model

- *Entry*, a collection of information about an object. Often associated with real-world objects, though it is not a requirement.
- Entry is composed of a set of *attributes*, each of which describes one particular trait of the object.
- Each attribute has a *type* and one or more *values*.
- *Type* describes the kind of information contained in the attribute.
- *Value* contains actual data

Directory Entry

CN:	John Doe Bob Doe
SN:	Doe
telephoneNumber:	+1 626 555 1212
Mail:	John_Bob@Doe.com

Directory Schemas

- Any entry in the directory has a set of required attribute types and a set of allowed attributes types.
- CN=(Common Name) is an attribute type associated with the person entry.
  - cn= John Doe
- SN= (Surname).
- DN = (Distinguish Name) entry in the directory which is unique.
  - **DN: cn=Robert Smith,ou=people,dc=example,dc=com**
  . DN is comprised of a series of RDN.
- There is a special attribute that is mandatory to all entries, called
  the *objectclass.*
- Other attributes are allowed, but not required.

Objectclass and Schema

- Determines what rules the entry follows. It govern the content of the entry by specifying the set of attributes that are mandatory and optional.
- Schema determines which object classes are available. Defines the set of rules the directory data must follow.
- Container helps organize other entries by establishing a parent/child relationship.
- For example ou (Organizational unit)

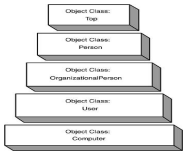
Objectclass

- Object classes associated with an entry serves the following needs:

  - It determines which attributes types must be included in the entry.
  - It dtermines which attribute types may be included in the entry.

Object Class Inheritance

- One object class can be derived from another, in which case it inherits required attribute types of the other class.



LDAP Information Model

- ObjectClass, allows you to control which attributes are required.
- The values of the objectclass attribute determine the schema rules the entry must obey.
- Example of Objectclass:

person	Requires:	Allows:
	sn: Jensen	description: direc
	cn: Babs Jensen	
	objectclass: top	
	person	

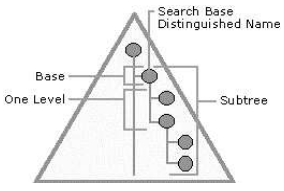
LDAP Functional Model

- Describes the operations that you can perform on the directory using the LDAP protocol.
- *Interrogation operations*, allow you to search the directory and retrieve directory data.
- *Update Operations*, allow you to add, delete, rename, and change directory entries.
- *Authentication/Control Operations*, allow clients to identify themselves to the directory and control aspects of a session.

LDAP Search Operation

- 3 Search Scope
- *Sub (subtree)*, indicates that you want to search the entire subtree from the base object all the way down to the leaves of the tree.
- *Onelevel*, indicates that you want to search only the immediate children of the entry at the top of the search base.
- *Base*, indicates that you want to limit your search to just the base object.

LDAP Search Scope



LDAP Search and Filters

Base DN: dc=mycompany, dc=com
Scope: Subtree
Search Filter=(cn=Brian Arkills)

Filter Search:
(|(cn=Brian Arkills)(cn=John))
- Returns the entries of John and Brian Arkills.

(!(|(cn=Brian Arkills)(cn=John)))
- returns all entries in the entrie directory except John or Brian.

<= less than or equal to : (sn<=Arkills).
>= Greater than or equal to: (sn>=Arkills).
~= Approximate: (sn~Cat) returns entries like sn=Scat, sn=Cast, sn=Hat.

LDIF

- LDAP Data Interchange Format a standard text-based format for describing directory entries.
- Allows you to export your directory data and import it into another directory server.

```
# Barbara's Entry
  dn: cn=Barbara J Jensen, dc=example, dc=com
   cn: Barbara J Jensen
   cn: Babs Jensen
   objectClass: person
   sn: Jensen

# Jennifer's Entry
    dn: cn=Jennifer J Jensen, dc=example, dc=com
    cn: Jennifer J Jensen
    cn: Jennifer Jensen
   objectClass: person sn: Jensen
```

LDIF

- Ldapmodify –h ldap.example.com –D "cn=directory manager" –w password –a <updates.ldif.
- The *add changetype* statement indicates that an entry is to be added to the directory

```
dn: uid=bjensen, ou=people, dc=example, dc=com
 changetype: add
 objectclass: top
  objectclass: organizationalPErson
  objectclass: inetOrgPerson
   cn : Barbara Jensen
    sn: Jensen
     uid: bjensen
     mail :bjensen@example.com
```