Directory Solutions Using OpenLDAP

Michael MacIsaac - IBM - mikemac@us.ibm.com
February 23rd, 4:30 PM
SHARE Session 9207

Abstract

Directory services are becoming the central location in the enterprise to store and retrieve information relating to users, groups, passwords, machines, printers and more. Most directories are based on the Lightweight Directory Access Protocol (LDAP). Some of the leading directory server implementations are Novell's eDirectory (formerly NDS), Microsoft's Active Directory (AD), Sun's iPlanet, IBM's Directory Server, and the open source package OpenLDAP. This presentation describes directory services at a high level describing basics such as DITs, schema, referrals, backends, objectClasses, binding, security, etc. It then drills down to the details of implementing OpenLDAP. Code and command examples are supplied that you can easily take back home and use with your Linux on zSeries images.

Outline for this hour

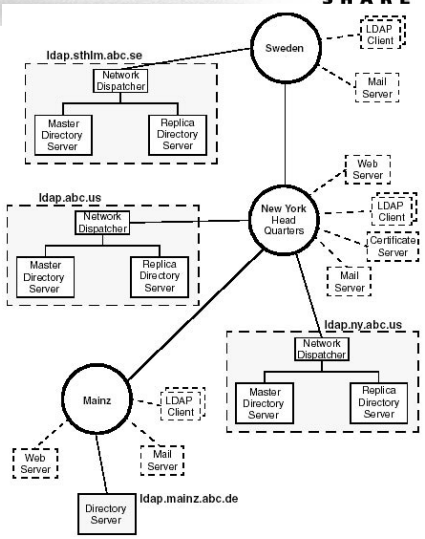
- Introductions
- Overview of LDAP
- Background on SSL and TLS security
- Linux directory solutions
  - Set up OpenLDAP
  - Use GUI clients to view data
  - Add TLS security to OpenLDAP
  - Set Samba up to use OpenLDAP
- Documentation and resources

Introductions - Who am I, Who are you?

- Who am I?
  - Michael MacIsaac, 17 years with IBM
    - 10 years programmer (Fortran, C, C++)
    - 7 years with S/390 (Linux tech support, ITSO project lead)
  - Linux (open source/freeware) advocate
  - e-mail - mikemac@us.ibm.com
- Who are you?
  - LDAP in your organization?
  - Linux servers in production?
  - Linux on your desktop?

Overview - complex LDAP solution

LDAP can become a complex solution - Example of a more sophisticated LDAP architecture:



Overview - What is LDAP?

- Lightweight Directory Access Protocol
- A network *protocol* for accessing information in a directory
- Hierarchical data reflecting political, geographic or organizational boundaries
- Based on the "heavyweight" X.500 standard - used OSI and is probably over-engineered
- A system designed for reading more than writing
- The basis of IBM "BluePages"

Overview - How can LDAP be used?

- Personnel information lookup
- Centralized login - User authentication, Password maintenance
- Centralized home directories - automount and NFS
- e-mail system
- File, Print, Centralized Windows login - Samba

Overview - What are some LDAP clients?

- Linux ldap* commands:
  - ldapadd, ldapdelete, ldapsearch, ldapcompare, ldapmodify, ldappasswd, etc.
- Linux library via nsswitch
  - /lib/libnss_ldap.so.2
- e-mail clients:
  - Outlook, OS X Mail, Eudora, Netscape/Mozilla, QuickMail Pro, and Mulberry.
- Samba
- GUIs
  - gq - installed in default SLES-8
    `http://biot.com/gq/`
  - web2ldap
    `http://www.web2ldap.de/`
  - directory administrator
    `http://diradmin.open-it.org/index.php`
- Custom Perl, C or dynamicWeb apps with LDAP back-ends

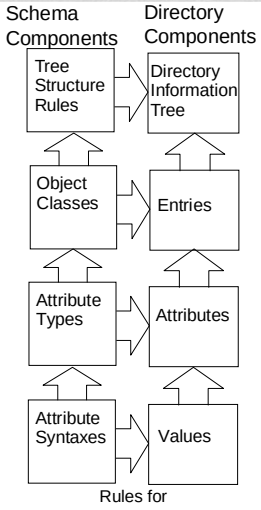
Overview - LDAP terms

- Commonly used LDAP terms
  - Suffix, base or root - the base of the local tree
    - Country/Organization-based - e.g "c=us, o=acme"
    - DNS-based - e.g. dc=ibm, dc=com
  - DN - distiguished name - refers to an entry unambiguously
    - uid=ldapuser,ou=People,dc=poklcc,dc=ibm,dc=com
  - RN - relative name
  - OU - organizational unit
  - CN - common name
  - DIT - Directory Information Tree - the hierarchical data
  - Schema - definition of objects, metadata
    - Object Class
    - Super Class and inheritance - "top" is the super-est class
    - Auxiliary Class - cannot stand on its own like a "structural" class
    - Attribute Type
    - Attribute Definitions

Overview - LDAP implementations

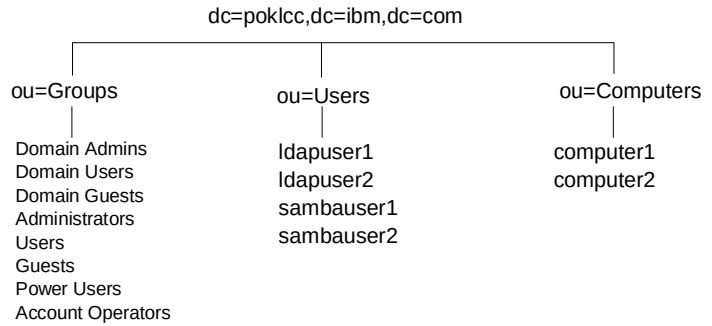
- IBM Directory Server
  - free!
- Microsoft Active Directory - considered proprietary:
  - Active Directory requires API developers to perform external application integration that a pure LDAP server would handle.
  - Active Directory has limited schema support within directory structures.
  - Microsoft will be introducing a new version of Active Directory called Active Directory Application Mode (ADAM) in Windows Server 2003.
- Novell eDirectory
  - Formerly NDS - now available on Linux (still not on zSeries Linux)
  - Excellent track record
- OpenLDAP
  - Based on original University of Michigan LDAP implementation
  - Packaged with SuSE SLES and RHEL
- Sun ONE (Open Networking Environment)
  - Formerly Netscape iPlanet

Overview - Schema vs. data block diagram



Overview - DIT exampe

Example of a Directory Information Tree (DIT)

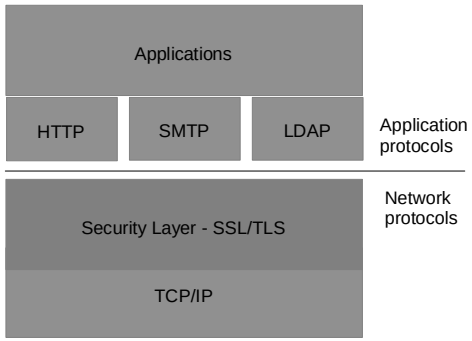


Note: RFC 2247 recommends this DNS-style root or base
X.500 used o=org,l=location,c=country style

Background: security

LDAP v3 authentication options:

- Anonymous
- Simple authentication
- Simple authentication over SSL/TLS
- Simple authentication and Security Layer (SASL)



Solution - Set up OpenLDAP

- Steps to set up sample OpenLDAP server:
  - Get OpenLDAP
  - Configure the LDAP server
  - Configure the LDAP client
  - Create an LDIF file
  - Start LDAP server
  - Insert base objects
  - Configure Linux to authenticate with LDAP
- Get OpenLDAP
  - SuSE SLES-8 in this example has OpenLDAP installed
  - If your distribution does not, get the source tar file at:

    `http://www.openldap.org/`
  - Verify that you have OpenLDAP installed:

```
# rpm -qa | grep openldap
openldap2-client-2.1.4-26
openldap2-2.1.4-26
# rcldap status
Checking for service ldap:                    unused
```

Solution - Set up OpenLDAP (cont'd)

- Configure the LDAP server
  - Configuration file is /etc/openldap/slapd.conf

```
# vi slapd.conf  -> add two schemas, set suffix, rootdn, rootpw
# global directives
include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include         /etc/openldap/schema/nis.schema
pidfile         /var/run/slapd/slapd.pid
argsfile        /var/run/slapd/slapd.args

# database definition and configuration directives
database        bdb
suffix          "dc=poklcc,dc=ibm,dc=com"
rootdn          "cn=Manager,dc=poklcc,dc=ibm,dc=com"
rootpw          12345
directory       /var/lib/ldap
index   objectClass     eq
```

Solution - Set up OpenLDAP (cont'd)

- Configure the LDAP client (/etc/openldap/ldap.conf)

```
# vi ldap.conf
host    localhost
BASE    dc=poklcc,dc=ibm,dc=com
```

- Create an LDIF (LDap Interchange Format) file
  - PADL has migration tools - see:

    `http://www.padl.com/OSS/MigrationTools.html`
  - Four important files
    - **migrate_common.ph** Contains variables to be used by next three scripts
    - **migrate_base.pl** Creates naming context entries. e.g. "dc=ibm,dc=com"
    - **migrate_group.pl** Migrates groups in /etc/group.
    - **migrate_passwd.pl** Migrates users in /etc/passwd.
  - Get the tar file from the Web and untar:

```
# cd /usr/src
# tar xzf MigrationTools-44.tar.gz
# cd MigrationTools-44
```

Solution - Set up OpenLDAP (cont'd)

- Create an LDIF (LDap Interchange Format) file (cont'd)
  - The file migrate_common.ph contains the default base, or suffix.

```
# vi migrate_common.ph  --> Modify the default base on line 74:
$DEFAULT_BASE = "dc=poklcc,dc=ibm,dc=com";
```

  - Run the scripts to create a base DIT, then migrate the group and user information in the files /etc/group and /etc/passwd:

```
# ./migrate_base.pl > initial.ldif
# ./migrate_group.pl /etc/group >> initial.ldif
# ./migrate_passwd.pl /etc/passwd >> initial.ldif
```

  - To the end of the LDIF file add a single LDAP user:

```
dn: uid=ldapuser1,ou=People,dc=poklcc,dc=ibm,dc=com
uid: ldapuser1
cn: ldapuser1
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
loginShell: /bin/bash
uidNumber: 501
gidNumber: 100
homeDirectory: /home/ldapuser1
```

  - You now have an LDIF file to create an initial DIT

Solution - Set up OpenLDAP (cont'd)

- Start the LDAP server
  - Start LDAP for this session (rcldap) and upon reboot (chkconfig)

```
# rcldap status
Checking for service ldap:                                  unused
# rcldap start
Starting ldap-server                                        done
# chkconfig ldap
ldap  off
# chkconfig ldap on
```

- Insert base objects
  - View emtpy data directory then add contents of LDIF file

```
# ls -l /var/lib/ldap
# ldapadd -x -h localhost -D "cn=manager,dc=poklcc,dc=ibm,dc=com"
  -f initial.ldif -W
Enter LDAP Password:
adding new entry "dc=poklcc,dc=ibm,dc=com"
adding new entry "ou=Groups,dc=poklcc,dc=ibm,dc=com"
...
```

Solution - Set up OpenLDAP (cont'd)

- Insert base objects (cont'd)
  - View data via ldapsearch

```
# ldapsearch -x
... (should show all entries added)
```

  - View populated data directory

```
# ls -l /var/lib/ldap
total 176
-rw-------  ldap     8192 Jan 31 11:40 __db.001        <- The Schema
-rw-------  ldap   270336 Jan 31 11:40 __db.002
-rw-------  ldap    98304 Jan 31 11:40 __db.003
-rw-------  ldap   360448 Jan 31 11:40 __db.004
-rw-------  ldap    16384 Jan 31 11:40 __db.005
-rw-------  ldap     8192 Jan 31 11:40 dn2id.bdb
-rw-------  ldap    32768 Jan 31 11:40 id2entry.bdb
-rw-------  ldap    79663 Jan 31 11:42 log.0000000001
-rw-------  ldap    20480 Jan 31 11:42 objectClass.bdb  <- The DATA
```

Solution - Set up OpenLDAP (cont'd)

- Configure Linux to authenticate with LDAP
  - Configure the name service switch - be careful

```
# cd /etc
# cp nsswitch.conf nsswitch.conf.orig
# vi nsswitch.conf            # replace the compat references:
passwd:  files ldap
group:   files ldap
...
```

  - Modify SSH PAM config file - be careful (keep a root session open)

```
# cd /etc/pam.d
# vi sshd  -> modify the "auth" lines as follows
#%PAM-1.0
auth    required        pam_nologin.so
auth    required        pam_env.so
auth    sufficient      pam_ldap.so
auth    required        pam_unix2.so    nullok use_first_pass
account required        pam_unix2.so
account required        pam_nologin.so
password required       pam_pwcheck.so
password required       pam_unix2.so    use_first_pass use_authtok
session required        pam_unix2.so    none     # trace or debug
session required        pam_limits.so
```

Solution - Set up OpenLDAP (cont'd)

- Configure Linux to authenticate with LDAP (cont'd)
  - Restart ssh daemon to reread PAM config file:

```
# rcsshd restart
Shutting down SSH daemon                                    done
Starting SSH daemon                                         done
```

  - Restart nscd (name service cache daemon) to flush user name cache.

```
# rcnscd restart
Shutting down Name Service Cache Daemon                     done
Starting Name Service Cache Daemon                          done
```

  - The id command should now work on LDAP users

```
# id ldapuser
uid=501(ldapuser) gid=100(users) groups=100(users)
```

    - Make a home directory and set ownership for ldapuser then set password

```
# mkdir ~ldapuser
# chown ldauser.users ~ldapuser
# ldappasswd -x -D "cn=Manager,dc=poklcc,dc=ibm,dc=com" -W -S
  "uid=ldapuser,ou=People,dc=poklcc,dc=ibm,dc=com"
```

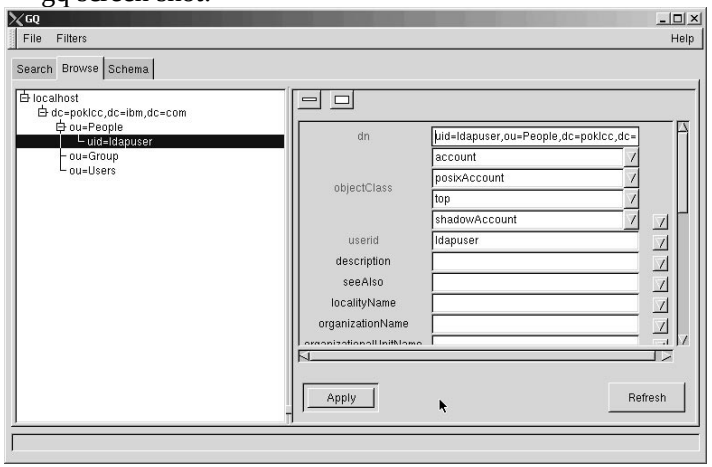
  - Test with ssh client (putty) with none/bad/good password against local and LDAP users
    - Total of 6 permutations

Solution - Use GUI clients to view data

- Choices
  - gq
  - directory administrator
  - web2ldap
- gq attributes
  - LDAP V3 Schema browser
  - Template builder
  - Export subtree or whole server to LDIF
  - Use any number of servers
  - Search based on single argument or LDAP filter
  - Edit and delete entries
  - Add entries using an existing entry, or based on your own template
  - Supports LDAP syntaxes and special attributes by presenting them in a custom way. Currently there are custom "displaytypes" for jpegPhoto, userPassword and X509 certificates and CRLs.

Solution - Use GUI clients to view data

- gq screen shot:



Solution - Use GUI clients to view data

- web2ldap: Must be built
  - Pieces needed:
    - python-xml - RPM on SLES-8 supplementary CD2
    - python-ldap - Get from Internet and build
    - pyweblib - Get from Internet and build
    - web2ldap - Get from Internet and run
    - md4.py - needed?
  - Python-xml
    - First mount SLES-8 supplementary CD1

```
# cd /mnt
# mkdir cd1 cd2 cd3 sup1 sup2 sp2
# mount 9.117.73.30:/sles8/sup1 sup1
```

    - Then install the RPM

```
# rpm -ivh sup1/suse/s390/python-xml-2.2.1-35.s390.rpm
```

Solution - Use GUI clients to view data (cont'd)

- web2ldap: Must be built - pieces needed:
  - python-xml - RPM on SLES-8 supplementary CD2
  - python-ldap - Get from Internet and build
  - pyweblib - Get from Internet and build
  - web2ldap - Get from Internet and run
  - md4.py - needed?
- Python-xml
  - First mount SLES-8 supplementary CD1

```
# cd /mnt
# mkdir cd1 cd2 cd3 sup1 sup2 sp2
# mount 9.117.73.30:/sles8/sup1 sup1
```

  - Then install the RPM

```
# rpm -ivh sup1/suse/s390/python-xml-2.2.1-35.s390.rpm
```

Solution - Use GUI clients to view data (cont'd)

- Python-ldap
  - Home page

```
http://python-ldap.sourceforge.net/
```

  - Download

```
http://sourceforge.net/project/showfiles.php?group_id=2072
```

  - To build:

```
# cd /usr/src
# ftp <my file server>
ftp> get web2ldap-0.11.24.tar.gz
ftp> get python-ldap-2.0.0pre13.tar.gz
ftp> get pyweblib-1.2.2.tar.gz
ftp> quit
# tar xzf python-ldap-2.0.0pre13.tar.gz
# cd python-ldap-2.0.0pre13/
# cp setup.cfg setup.cfg.orig
# vi setup.cfg  -> change sasl2 to sasl
# diff setup.cfg setup.cfg.orig
26c26
< libs = ldap_r lber sasl ssl crypto
---
> libs = ldap_r lber sasl2 ssl crypto
# python setup.py install -O2
...
```

Solution - Use GUI clients to view data (cont'd)

- Pyweblib
  - Untar and build

```
# tar xzf pyweblib-1.2.2.tar.gz
# cd pyweblib-1.2.2/
# python setup.py install -O2
```

- Web2ldap
  - Untar

```
# cd /usr/src
# tar xzf web2ldap-0.11.24.tar.gz
```

  - Modify one parameter to allow access from all hosts

```
# cd web2ldap/etc/web2ldap/web2ldapcnf/
# cp standalone.py standalone.py.orig
# vi standalone.py  -> switch the comments around line 20
access_allowed = ['0.0.0.0/0.0.0.0']
#access_allowed = ['127.0.0.0/255.0.0.0']
```

  - Start

```
# cd ../../../sbin
# ./web2ldap.py -l 0.0.0.0:1760
...
```

Solution - Add TLS security to LDAP

- TLS is the follow-on to SSL
  - It encrypts LDAP communications
- Overall steps
  - Create a certificate
  - Configure the LDAP server
  - Configure the LDAP client
- Create a certificate
  - 3 key files are needed:
    - ldap.cert The server certificate
    - ldap.key The private key that matches the TLSCertificateFile file
    - ca.cert Certificates for all CAs that slapd will recognize
- Good reference Web site

```
http://www.openldap.org/faq/data/cache/185.html
```

Solution - Set Samba to use OpenLDAP

- Copy the samba schema to the /etc/openldap/schema/ directory:

```
# cd /etc/openldap
# cp /usr/share/doc/packages/samba/examples/LDAP/samba.schema schema
```

- Add inetorgperson and Samba schemas

```
# vi slapd.conf            # add 2 lines after the other schemas
include         /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/samba.schema
```

- Verify nsswitch is still using ldap

```
# grep ldap /etc/nsswitch.conf
passwd:         files ldap
group:          files ldap
```

- Change Samba system configuration file to use LDAP-built Samba

```
# cd /etc/sysconfig
# vi samba                # set SAMBA_SAM to ldap:
#SAMBA_SAM="classic"
SAMBA_SAM="ldap"
# SuSEconfig --module samba
...
```

Solution - Set Samba to use OpenLDAP (cont'd)

- Modify the Samba configuation file to use LDAP:

```
# cd /etc/samba
# vi smb.conf         # ... add the following lines to [global]
        ldap port = 389
        ldap server = localhost
        ldap suffix = dc=poklcc,dc=ibm,dc=com
        ldap admin dn = cn=Manager,dc=poklcc,dc=ibm,dc=com
        ldap ssl = no
```

- Start (or restart) Samba and OpenLDAP

```
# rcnmb restart
Shutting down Samba classic NMB daemon                    done
Starting Samba ldap NMB daemon                            done
# rcsmb restart
Shutting down Samba classic SMB daemon                    done
Starting Samba ldap SMB daemon                            done
# rcldap restart
Shutting down ldap-server                                 done
Starting ldap-server                                      done
```

Solution - Set Samba to use OpenLDAP (cont'd)

- Set the LDAP manager password in /etc/samba/secrets.tdb file:

```
# smbpasswd -w 12345
Setting password for "cn=Manager,dc=poklcc,dc=ibm,dc=com" in secrets.tdb
```

- Add a test user to /etc/passwd, and create a home directory

```
# useradd sambauser
# mkdir ~sambauser
# chown sambauser.users ~sambauser
```

- Set the LDAP password via the smbpasswd command

```
# smbpasswd -a sambauser
New SMB password: sambauser
Retype new SMB password: sambauser
LDAP search "(&(uid=sambauser)(objectclass=sambaAccount))" returned 0 entries.
Added user sambauser.
```

- Test the new user

```
# id sambauser
uid=502(sambauser) gid=100(users) groups=100(users)
```

- Now you should be able to get an SMB share from a Windows desktop
  - For example, from a DOS prompt:

```
C:\>net use * \\9.117.119.67\sambauser
```

Resources

- Books, papers
  - IBM Redbook *Understanding LDAP*, SG24-4986, Heinz Johner, et al
    `http://www.redbooks.ibm.com/abstracts/sg244986.html`
  - *OpenLDAP 2.1 Administrator's Guide*, OpenLDAP team
    `http://www.openldap.org/doc/admin21/`
  - *Directory Services for Linux in comparison with Novell NDS and Microsoft Active Directory*, Norbert Klasen Master's Thesis
    `http://www.daasi.de/staff/norbert/thesis/`
  - *LDAP System Administration*, Gerald Carter, O'Reilly
    `http://www.oreilly.com/catalog/ldapsa/index.html`
- Web sites
  - OpenLDAP
    `http://www.openldap.org/`
  - PADL Software PDY Ltd.
    `http://www.padl.com/`
  - web2ldap
    `http://www.web2ldap.de/`

Questions ??