LDAP: Integrating Authentication Across Operating Systems and Applications

Kevin Falcone
kevin@jibsheet.com
NIS - Current Infrastructure

- NIS - Network Information Services (formerly yp)
- Centralized NIS server with slaves
- NIS distributes password, group, and NFS information
Advantages and Disadvantages of NIS

- **Disadvantages**
  - No security model
  - Cleartext communication of passwords
    - yp tools (ypcat passwd)
  - yp tools (ypcat passwd)

- **Advantages**
  - Excellent integration with Solaris
  - Integration with Linux, BSD, OS X, and even Windows
  - Easy replication and failover
LDAP - What is it?

- Lightweight Directory Access Protocol (RFC 2251)
- Fast retrieval of an entry based on an attribute
  - attributes may be building, room, last name, username, machine name, group name, etc
- Tree Structure (RFC 2253)
  - uid=user,ou=majors,dc=ccs,dc=neu,dc=edu
- goal: store username/password/group information in the LDAP tree
Advantages of LDAP

- Security
  - SASL
  - TLS using SSL Certificates
  - Restriction of data access
- Integration with multiple Operating Systems and Applications
Disadvantages of LDAP

- Complexity
  - Replication/Mirroring much more complex than NIS
  - More complex configuration and data storage than NIS
  - Lack of tools for managing data
- Lack of a standard for transferring data between implementations (i.e. OpenLDAP and iPlanet Directory Server)
Increasing Security: the network layer

- SASL - Simple Authentication and Security Layer
  - Kerberos or Digest-MD5 instead of plaintext
- External security using TLS (Transport Layer Security)
  - OpenSSL support in server allows SSL-secured connection to clients
- Distribute known-good certificates to your clients
Increasing Security: the user level

- Access Control
  - Grant privileges to anonymous, authenticated, or groups of users
  - Allow users to change their password, but not read other user’s passwords
  - Only allow access to all passwords to the proxy user the authentication subsystems binds as
    - Some ldap authentication mechanisms can do an anonymous bind properly for authentication
Using LDAP to Replace NIS

• Current Information
  • User accounts and passwords
  • Groups and netgroups
  • NFS mountpoints

• Goal
  • User and Group information
  • NFS continues from NIS for now
Choosing an LDAP Server

• Vendors
  - Netscape/Sun
  - Lotus (used internally by Notes)
  - Microsoft Active Directory
  - OpenLDAP
OpenLDAP - slapd

- slapd - standalone LDAP daemon
  - Runs on your main authentication host
  - Knows about security configurations
  - Has access to the user database
    - stored in various backends
  - Can be replicated to slave servers
slapd configuration

database       bdb
suffix         "dc=ccs,dc=neu,dc=edu"
rootdn         "cn=manager,dc=ccs,dc=neu,dc=edu"
rootpw         {MD5}cryptedpassword
index          objectClass,uid,uidNumber,gidNumber eq
index          cn,mail,surname,givenname,eq,subinitial
password-hash  {crypt}
password-crypt-salt-format  "$1$%.8s"
Access Configuration

access to dn=".*,dc=ccs,dc=neu,dc=edu" attr=userPassword
   by dn="cn=manager,dc=ccs,dc=neu,dc=edu" write
   by self write
   by * auth

access to dn=".*,dc=ccs,dc=neu,dc=edu"
   by * read
Securing slapd with ssl

slapd.conf:
LSCipherSuite HIGH:MEDIUM:+SSLv2
TLSCertificateFile /wherever/our/key/is/slapd.pem
TLSCertificateKeyFile /wherever/our/key/is/slapd.pem

startup:
slapd -h ""ldap:/// ldaps:///""
Configuring SASL

- Digest-MD5 or Kerberos
- Digest-MD5 requires password-hash \{CLEARTEXT\}
- assumes a cn=digest-md5 cn=auth
- remap using sasl-regexp

sasl-regexp
uid=\(\.*\),cn=ccs.neu.edu,cn=digest-md5,cn=auth
uid=$1,cn=majors,dc=ccs,dc=neu,dc=edu
Choosing Your Schemas

- Schemas - Predefined LDAP Objects
- nis.schema
  - account data
  - password data
  - host-based login control
- core.schema
  - basic personal data
  - address and phone data
Sample Data: groups

dn: cn=systems,ou=Group,dc=ccs,dc=neu,dc=edu
objectClass: posixGroup
objectClass: top
objectClass: groupOfMember # needed for mod_perl
cn: users
gidNumber: 1000
memberUid: me
memberUid: someone_else
Sample Data: users

dn: uid=me,ou=majors,dc=ccs,dc=neu,dc=edu
uid: me

objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount

userPassword: {crypt}password

loginShell: /bin/bash

uidNumber: 1001
gidNumber: 1000

homeDirectory: /home/me
Adding LDAP Data

- slapadd
  - uses LDIF files
- PADL scripts for migration
  - create LDIF files from /etc files and NIS maps
- Net::LDAP
- Visual Tools
  - GQ
  - vlad
Operating Systems

• Two Types of Operating Systems
  • Supported
    • Solaris
    • Windows (XP labs, 2000 servers)
  • Unsupported
    • Linux
    • OpenBSD
    • OS X
Operating Systems Testbed

- One Solaris 9 Machine
- Multiple Linux machines (Debian testing)
- One OS X machine (10.2.3)
- Two OpenBSD machines (3.3 and CURRENT)
- One Windows machine (2003 Server)
LDAP on Linux

- Use PAM - Pluggable Authentication Modules
  - pam_ldap
  - configure from nsswitch.conf
  - "supersedes" auth, account, and password directives
- No PAM?
  - upgrade
  - libpam is straightforward to build from source
Linux Configuration

ldap.conf: on the server
host 127.0.0.1
base dc=ccs,dc=neu,dc=edu
rootbinddn cn=manager,dc=ccs,dc=neu,dc=edu
scope one
pam_filter objectclass=posixaccount
pam_login_attribute uid
pam_member_attribute gid
pam_password md5
nss_base_passwd        ou=majors,dc=ccs,dc=neu,dc=edu?one
nss_base_shadow        ou=majors,dc=ccs,dc=neu,dc=edu?one
nss_base_group         ou=majors,dc=ccs,dc=neu,dc=edu?one
Linux Configuration (cont)

- /etc/nsswitch.conf

  passwd:  files ldap
  shadow:  files ldap
  group:   files ldap

  getent group passwd or shadow should return the expected information
Linux Configuration (cont)

- /etc/pam.d/auth
  auth sufficient /path/to/pam_ldap.so use_first_pass
  account sufficient /path/to/pam_ldap.so
  password sufficient /path/to/pam_ldap.so use_authhtok
  session requires /path/to/pam_mkhomedir.so skel=/etc/skel \ umask=0022
  session optional /path/to/pam_ldap.so
LDAP on Solaris

- Supports SASL and SSL in the Solaris 9 client
- Prefers the Solaris 'iPlanet Directory Server' LDAP daemon
- Idapclient
  - allows for automated configuration of clients from a profile
  - also allows manual override of options
LDAP on Solaris (cont)

- Solaris supports PAM and NSS
  - configuration is similar to Linux
- `/etc/nsswitch.ldap` - merge with `/etc/nsswitch`
- `/etc/pam.conf` - add `pam_ldap.so.1` lines
  - for login, other auth, passwd auth, other password
- `/var/ldap` needs database of certs and client keys for SSL
LDAP on OpenBSD

- login_ldap
  - add login_ldap.conf to /etc/login.conf
  - lack of NSS requires adding users to /etc/passwd
    - change class to the ldap class to trigger lookup
  - uses SSL but not SASL
- Authentication against FreeRadius
  - uses a Radius server to abstract away the LDAP server
  - complicated to implement unless you already have a radiusd running
LDAP on OS X

- Directory Services
- LDAPv3 Support
LDAP on Windows

- Need to replicate to Active Directory
- Or configure Samba as a Domain Controller
- Data can be pulled easily from OpenLDAP or NIS once
  - further merges require programming
- LISA Papers
Applications

- Apache
  - mod_perl
- Subversion
- Jabber
- IMAP
- Calendaring/Lotus
Apache and LDAP

- mod_auth_ldap
  - non-standard in 1.3.x, standard in 2.0
  - Allows user and group Basic authentication

```<Location /ldap>
AuthLDAPURL "ldaps://ldap.server/dc=ccs,dc=neu,dc=edu?uid"
LDAPTrustedCA /path/to/key/for/server
require group cn=systems,ou=Group,dc=ccs,dc=neu,dc=edu
</Location>
```

LDAPBindDN for Proxy User
mod_perl and LDAP

- Apache::AuthPerLDAP
- Apache::AuthenLDAP
- Apache::AuthNetLDAP
- Apache::AuthzNetLDAP
Old mod_perl Modules

- AuthPerLDAP
  - uses Netscape's LDAP libraries
  - not updated
- AuthenLDAP
  - not updated since 2001
  - no plans to port to mod_perl2
Apache::AuthNetLDAP

- Wrapper around Net::LDAP
  - pure perl query library
- supports basic auth
- AuthenHandler, so only understands uid authentication
- Inherits capabilities from Net::LDAP
  - Net::LDAPS
  - Authen::SASL
Configuring Apache::AuthNetLDAP

<Directory /auth/required>
AuthName "LDAPville"
AuthType Basic

PerlSetVar BaseDN dc=ccs,dc=neu,dc=edu
PerlSetVar LDAPSserver ldaps://ldap.server
PerlSetVar UIDAttr uid # uid is default, email

PerlAuthenHandler Apache::AuthNetLDAP
require valid-user
</Directory>
Attributes?

- One of our requirements was group authentication
- Also, maybe you want to validate against multiple attributes
- We'll need an AuthzHandler
Apache::AuthzNetLDAP

- Integrates cleanly with AuthNetLDAP
- Allows you to limit user access
  - require user specialperson
  - groupdn
- Allows you to limit to any dn which is a groupOfMember or groupOfUniqueMember
- Arbitrary LDAP URL
  - ldaps://ldap.server/ou=majors,dc=ccs,dc=neu,dc=edu?loginShell=/bin/csh
Configuring Apache::AuthzNetLDAP

<Directory /require/auth>
AuthName "LDAPville"
AuthType Basic

PerlSetVar AuthenBaseDN dc=ccs,dc=neu,dc=edu
PerlSetVar AuthenLDAPServer ldaps://ldap.server
PerlSetVar AuthenUIDAttrtype uid # uid is default, email

PerlAuthenHandler Apache::AuthenLDAP
PerlAuthzHandler Apache::AuthzNetLDAP
require group "cn=systems,ou=Group,dc=ccs,dc=neu,dc=edu"
</Directory>
Non-Basic Authentication

- If you need non-Basic authentication
  - roll your own with Net::LDAP
  - Store data into one of the many Session modules
Subversion

- Uses (for now) Apache Basic authentication

```
<Location /svn/repo>
  DAV svn
  SVNPath /svn/repo
  AuthType Basic
  AuthName "Subversion repository"
  AuthLDAPURL "ldaps://ldap.server/dc=ccs,dc=neu,dc=edu?uid"
  LDAPTrustedCA /path/to/key/for/server
  require group cn=svnusers,ou=Group,dc=ccs,dc=neu,dc=edu
</Location>
```
Jabber with LDAP

- xdbldap
  - old, uses OpenLDAP1 libraries
- fljub
  - perl proxy server, sits between LDAP and Jabber
  - Implements JUD
- xdb_ldap
  - jabberstudio project
  - C plugin for Jabber
xdbldap

- Supports SSL, no SASL
- Load jabber.schema into OpenLDAP
- Edit jabber.xml, add jabber-xdb.xml
- copy xdb ldap into jabber plugins
- configure a jabber-admin ldap user
  - follow sample slapd.conf file
IMAP

- Washington University IMAPd
- Uses PAM on our Solaris host
  - Transparently uses LDAP for authentication
- Any PAM-capable IMAPd will work the same
Further Integration

- Faculty want Netscape Calendar
  - Will not work with OpenLDAP
  - Work may be replaced with iPlanet Directory Service
- Lotus Notes Installation
  - Projects for future
    - Can addressbook/calendar be replicated out?
Conclusions

• LDAP is difficult to configure and implement upfront
• The gains in network security for our primary platforms (Linux, Solaris, Windows) are significant
• More applications support LDAP authentication than NIS
• OpenLDAP libraries and Net::LDAP provide opportunity for further expansion
• Hope to see LDAP in use before my first reunion