



LDAP Configuration Choices

Jason L. W. Lynn

jlwlynn@uab.edu

University of Alabama at Birmingham



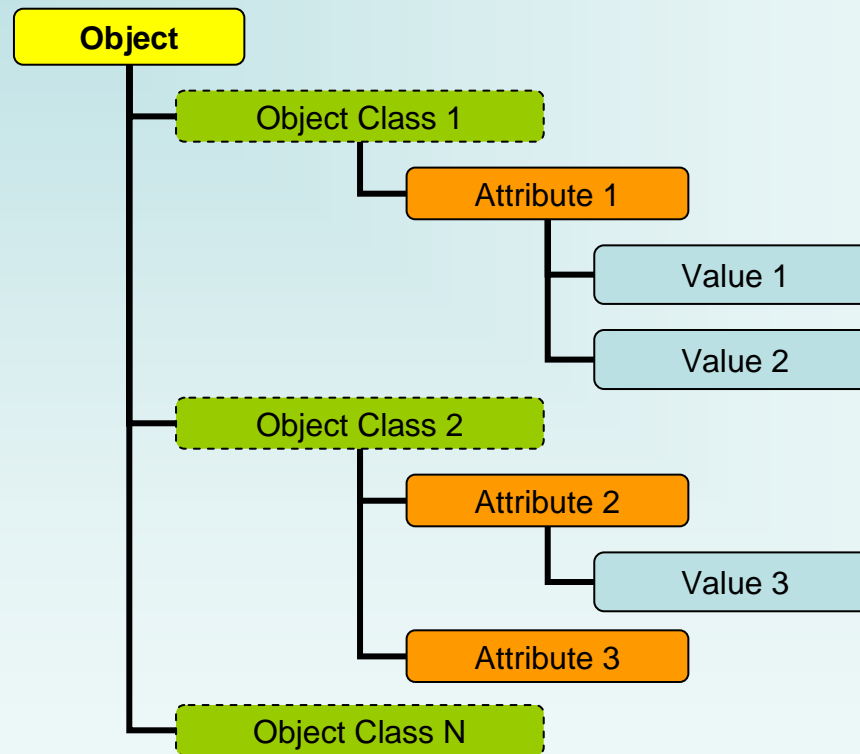
Outline

- Directory Server Configurations
- Gotchas





Understanding an LDAP Object





Understanding Security



- LDAP implementation requires thinking about :
 - Data security
 - Access Control Information
 - Network security
 - Secure Socket Layer





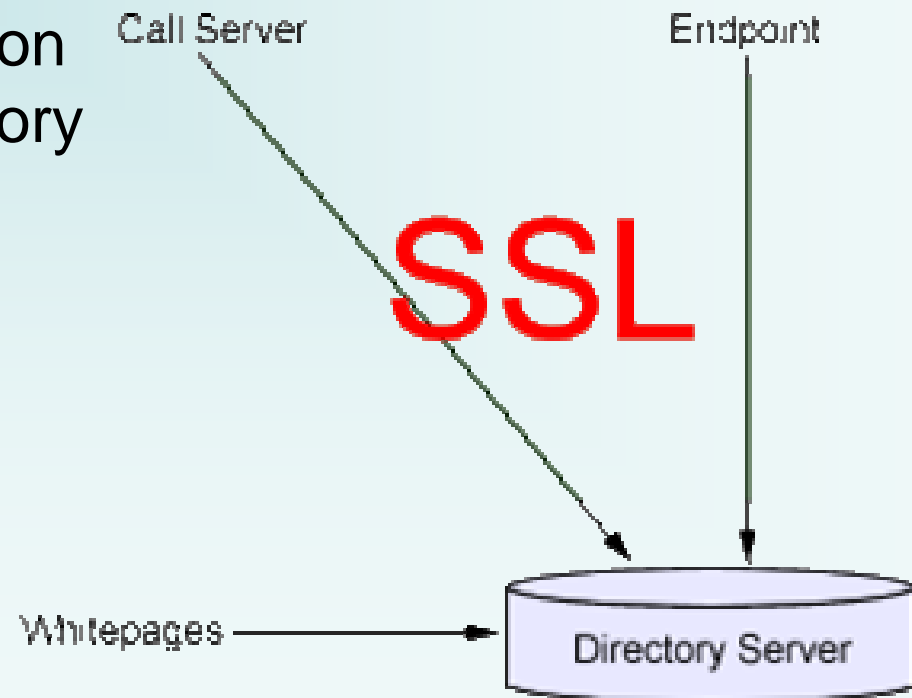
Understanding LDAP ACI Rules

- Allow or deny access to LDAP information
- Rule set not standardized
- Syntax not standardized
 - OpenLDAP Example:
access to *
by self write
by dn.base="uid=admin,dc=uab,dc=edu" write
by * read
 - Sun One Example:
(targetattr != "userPassword")
(version 3.0;
acl "Anonymous access";
allow (read,compare,search)
(userdn = "ldap:///anyone")
;)



Understanding LDAP and SSL

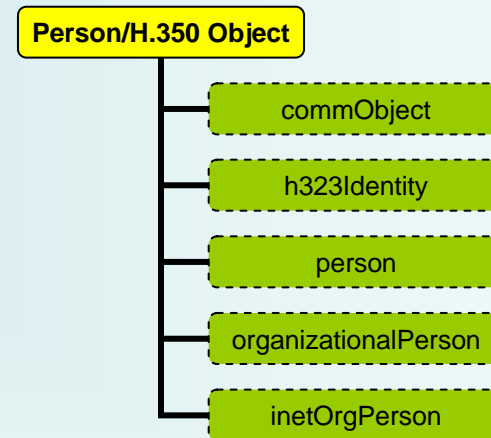
- Passwords sent in clear without SSL
- Secures transmission from client to directory server
- SSL encrypts data transmission





Single Directory Server, Single Object Approach

- Only one Directory Server is used
- H.350 Information is stored in Enterprise Directory 'person' Object





Single Directory Server, Single Object Approach



- Authentication is simple
 - One Directory
 - One BIND



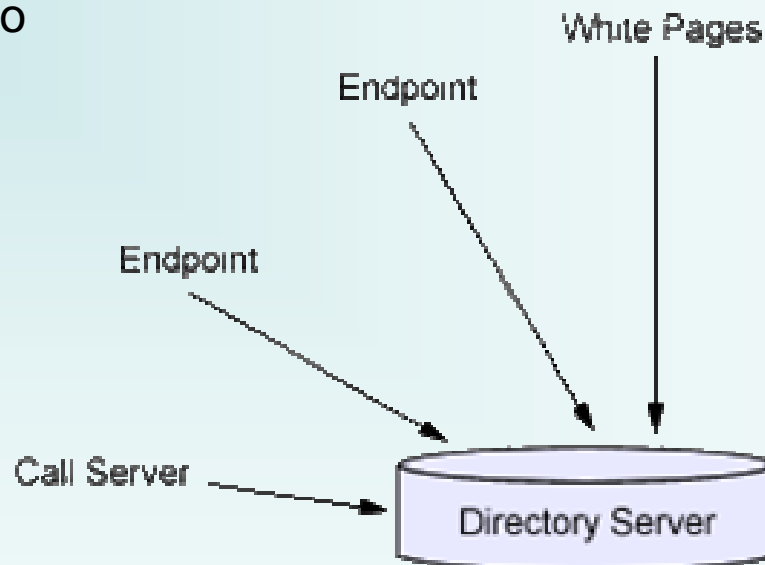
- Authorization is simple
 - All information contained in one object
 - Current ACI rules are easy to write for this scenario





Single Directory Server, Single Object Approach

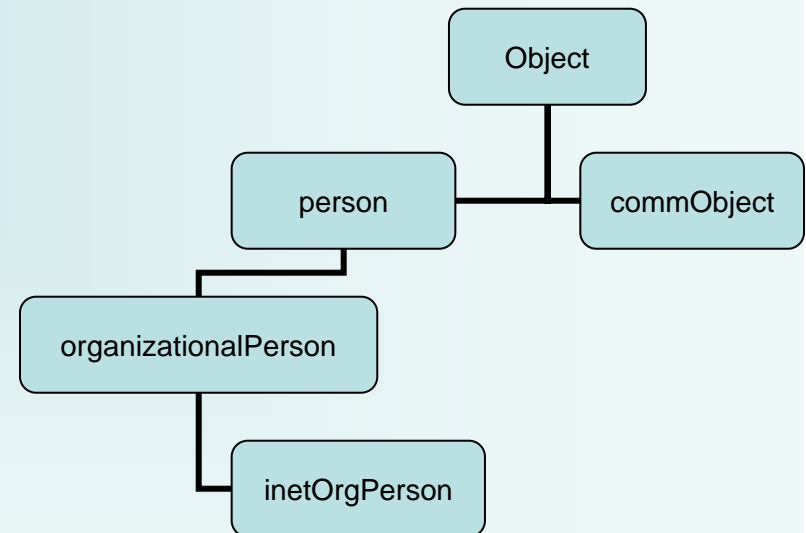
- Increased Load on the Enterprise Directory
 - All requests are coming to this one directory
 - Enterprise Directory may serve other purposes





Single Directory Server, Single Object Approach

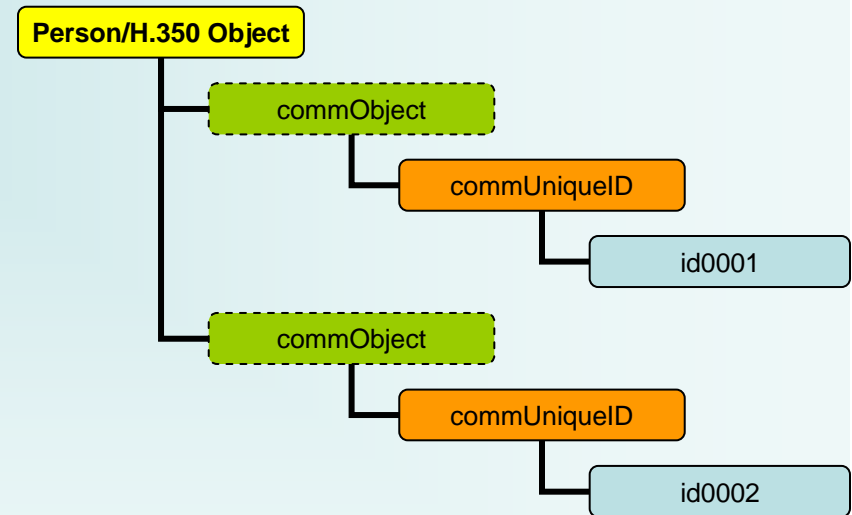
- Will not work with all Directory Servers
 - Only one structural chain per object (OpenLDAP)
 - X.501: “An object or alias entry is characterized by precisely one structural object class superclass chain which has a single structural object class as the most subordinate object class.”
 - **commObject** is one structural chain
 - **inetOrgPerson** is one structural chain





Single Directory Server, Single Object Approach

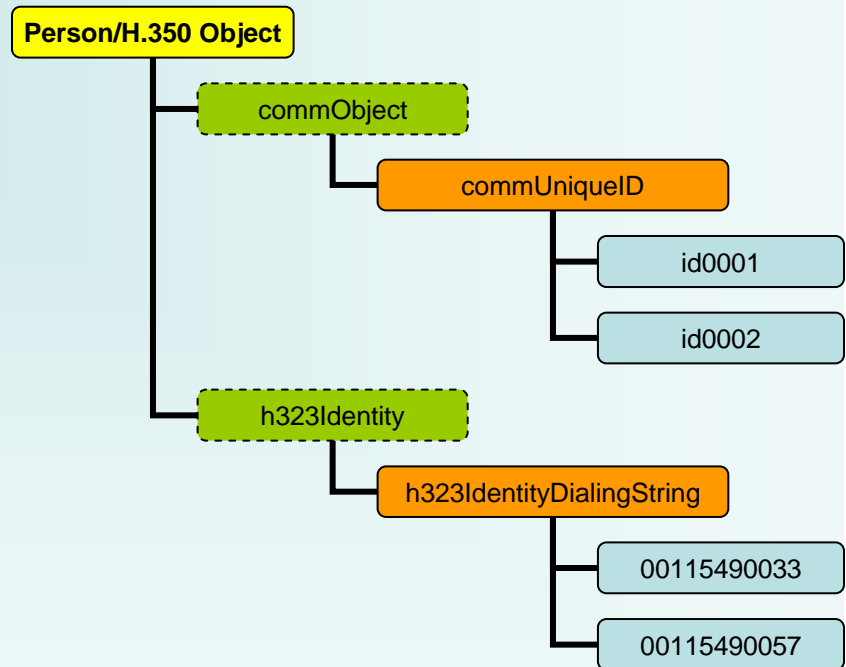
- One Endpoint limitation
 - Object classes (or attributes) can only be added to an object ONCE





Single Directory Server, Single Object Approach

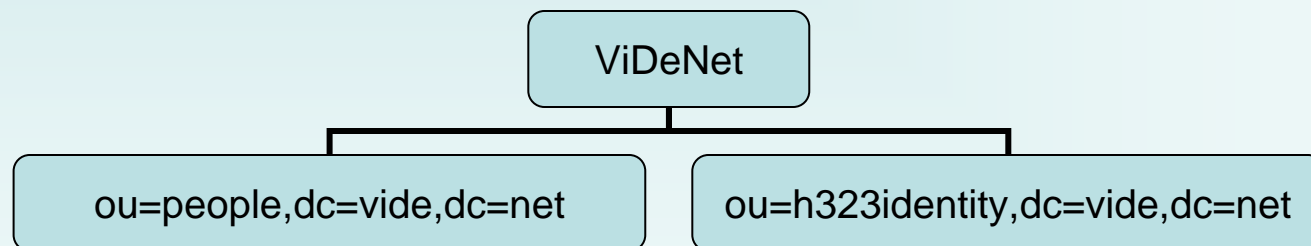
- One Endpoint limitation
 - Object classes can only be added to an object ONCE
 - With multiple attribute values, which one goes with which?





Single Directory Server, Multiple Object Approach

- Only one Directory Server is used
- One “tree” of the directory is used for Enterprise information
- One “tree” of directory is used for H.350 information





Single Directory Server, Multiple Object Approach

- Authentication is simple
 - One Directory, One BIND DN





Single Directory Server, Multiple Object Approach



- Authorization is more difficult
 - Information stored in different ‘trees’ of the directory
- Increased Load on the Enterprise Directory
 - All requests are coming to this one directory
 - Enterprise Directory may server other purposes





Multiple Directory Server Approach

- Two Directory Servers are used
- One Directory Server is dedicated to Enterprise information and use
- One Directory Server is dedicated to H.350 information and use

UAB Enterprise Directory

ou=people,dc=uab,dc=edu

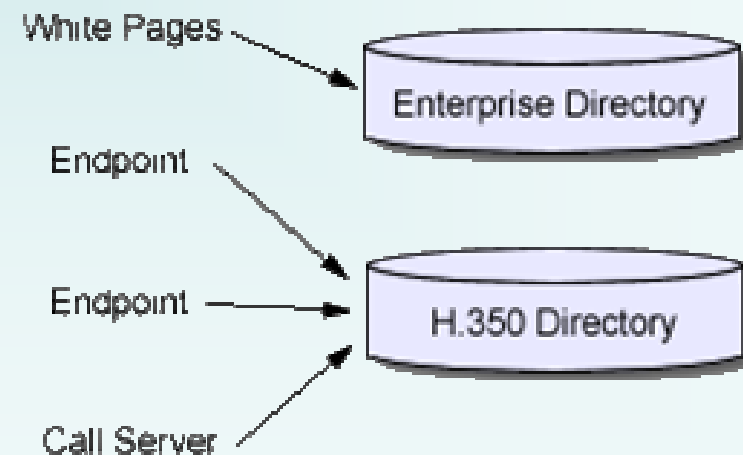
UAB H.350 Directory

ou=commobjects,dc=ac,dc=uab,dc=edu



Multiple Directory Server Approach

- Load placed on the H.350 Directory
 - Two separate directories
 - H.350 Directory ‘shoulders’ its own load





Multiple Directory Server Approach



- Authentication is more difficult
 - Two separate directories; two BINDs required
- Authorization is more difficult
 - H.350 information is stored on separate directory
 - Trying to allow/deny access based on Enterprise identity





So which way is best?



- One Object Approach not recommended
- Otherwise, it depends



- Is there an existing Enterprise Directory?
 - Do you have full control of the directory?
- What purpose will your Enterprise Directory serve?
 - Is H.350 the single purpose?
 - Are there any other applications that will need the directory?
 - Will that purpose change in the future?
- How many people will the Enterprise Directory support?
 - Is it a small number of people?
 - Will the number of people change in the future?





Multiple Services



- Assume that your network has multiple call servers
 - voip.unc.edu
 - video.unc.edu
 - im.unc.edu
- Choices
 - All H.350 entries in a single tree
 - Separate directory 'tree' for each service
 - Separate server for each service

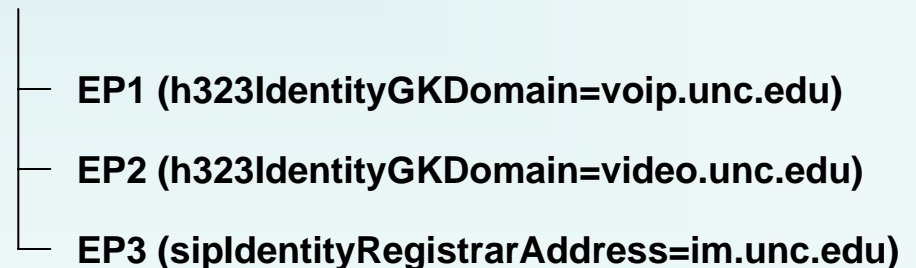




All entries in a single tree

- Distinguish realms by attribute values
 - h323IdentityGKDomain
 - sipIdentityRegistrarAddress
 - etc.
- Difficult to manage

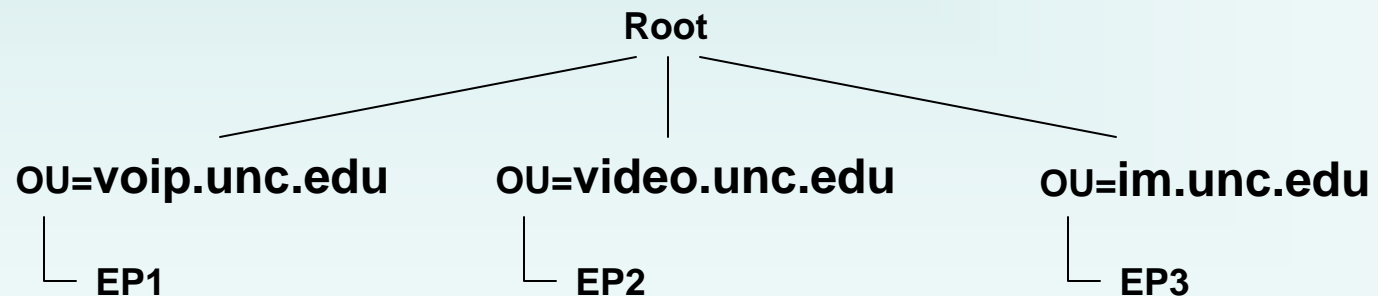
OU=ip_conf





Separate subtree for each service

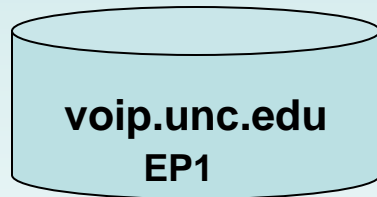
- Single server to maintain
- Can still tune it for call server access
- Easy to access an entire subtree





Separate directory server for each H.350 directory

- Highly tunable for each service
- Scalable
- Expensive because maintaining multiple servers





Gotchas



There are many little “quirks” that one encounters when implementing H.350 across an institution. Following are some of those problems, along with solutions, that many of us have encountered.





How do you BIND to an H.350 Directory with an Enterprise Identity?

- With separate directory servers, it is necessary to leverage your enterprise identity on the H.350 directory
- Solution
 - Chaining (pass through authentication)
 - H.350 Directory contacts Enterprise Directory on behalf of the client
 - A successful BIND to the Enterprise Directory yields a successful BIND to the H.350 Directory



How do you display an endpoint in a whitepages type application?

- Solution
 - Use label portion of commURI for display purposes
 - Hyperlink to a server side processing script
 - Pass commURI as a variable
 - Server side processing script should be able to follow link and display relevant information
 - [Example](#)



How do you determine the protocol from a commURI?

- commURI provides no 'hint' as to the protocol
- Solutions
 - Perform lookup on commObject's objectClass attribute
 - Check h235Identity, h323Identity, sipIdentity, etc.
 - 'Tag' the commURI label
 - ldap://vc.ac.uab.edu/dc=ac,dc=uab,dc=edu??sub?(communiq ueid=3) [**H323**] Desktop system
 - ldap://vc.ac.uab.edu/dc=ac,dc=uab,dc=edu??sub?(communiq ueid=3) [**SIP**] Desktop system



How do you generate commUniqueIDs?

- Need to provide unique identification
- Solutions
 - Within LDAP Directory
 - Create an object
 - Add attribute to act as counter
 - Read then increment
 - Other external means
 - Textfile
 - Database
 - Etc.



How do you generate dialing strings?

- Need for unique H.323 dialing strings
 - Global Dialing String
- Solutions
 - Within LDAP Directory
 - Create an object
 - Add attribute to act as a counter
 - Read then increment
 - Other external means
 - Textfile
 - Database
 - Etc.



How do you hide passwords?



- Many Videoconferencing / VoIP passwords to hide
 - H235
 - SIP



- Solutions

- ACL Rules
- Programmatically
 - Will only hide for that particular application





How do you hide entries for users?



- commPrivate

- Solutions

- ACI Rules

- Programmatically

- Will only hide for that particular application





How do you determine the owner of a commObject?



- Using multiple directories or single directory with separate trees presents problems



- Authorization

- Solutions

- One Object Approach

- commObjectOwnerDN

- Contains attribute 'owner' (OID: 2.5.4.32)





How do you synchronize data with a 'hostile' Enterprise Directory?

- Decide on information transfer format
 - Example :
 - ADD <uniqueid> <commuri>
 - DEL <uniqueid> <commuri>
- Decide on information transfer protocol
 - Email
 - FTP



Where do we go from here?



- Populating and Managing Endpoints
 - Scripts available in Video Middleware Cookbook
 - More Scripts on their way!



- Join the testbed
 - Mailing list
 - Email jlwlynn@uab.edu

