Locating Native Internet LDAP Servers

Preliminary Draft

STATUS OF THIS MEMO

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress".

WARNING: The specification in this document is subject to change, and will certainly change. It is inappropriate AND STUPID to implement to the proposed specification in this document. In particular, anyone who implements to this specification and then complains when it changes will be properly viewed as an idiot, and any such complaints shall be ignored. YOU HAVE BEEN WARNED.

To learn the current status of any Internet-Draft, please check the "1id-abstracts.txt" listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Distribution of this document is unlimited. Please send comments to the ASID working group at <ietf-asid@umich.edu>. Discussions of the working group are archived at <URL: ftp://terminator.rs.1td.umich.edu/ietf-asid/archive>. 
ABSTRACT

The LDAPv3 protocol (as specified in [1]) is designed to be a lightweight access protocol for directory services supporting X.500 models. This may be the X.500 directory itself, but the LDAP specification explicitly allows it to be a different directory. Let us define a "native LDAP server" to be one that is not a front end to the X.500 directory service. Let us further define an "Internet based organization" as one that has a domain name, and an "Internet LDAP server" to be one containing a directory entries for such an organization.

A native LDAP server can not rely upon the X.500 directory's knowledge base to locate the appropriate server to service a request on an object in a part of the directory tree (DIT) other than the naming context(s) it stores.

This draft defines a way that native Internet LDAP servers can make use of the DNS's knowledge base (which it stores as "glue" records) to perform the same function, while still supporting integration with the X.500 directory.

This draft builds on recent work by Kille and Wahl [2] to define a mechanism by which collections of native Internet LDAP servers can be integrated to create a directory service. That work supports this cause by defining a mapping from DNS names into LDAP DNs.

In an Internet context, many of the names about which users seek information are DNS names, or include DNS names. A native LDAP based directory service for the Internet should make it convenient to process such names -- there is a huge social investment spanning two decades to get to the point where names like "john.doe@somewhere.com" and "http://www.sony.com" can appear in newspaper articles, TV commercials, and on billboards and millions of people understand what do with them.

As a result, we assume that Internet based organizations wish to preserve this investment, yet also want to deploy directory services.

Table of Contents

1. Introduction...........................................................3
2. Motivation..........................................................Error! Bookmark not defined.
3. Overview............................................................4
4. Client behavior........................................................4
1. Introduction

The LDAP protocol is designed to be a lightweight access protocol for directory services supporting X.500 models \[1\]. This may be the X.500 directory itself, or it could be a different directory. One particularly desirable use is as the access protocol for a directory service for the Internet.

In the Internet context, the named entities which exist today about which users seek information almost all either have, or contain, DNS names \[3,4\], since it is by far the most widely used naming service in the Internet. There has a huge social investment spanning two decades to get to the point where names like "john.doe@somecorp.com" and "http://www.sony.com" can appear in newspaper articles, TV commercials, and on billboards, and millions of people understand what do with them.

Recent work by Kille and Wahl \[2\] makes a good start at being able to use LDAP as an access protocol for information about objects with DNS names by defining a mapping from DNS names to Distinguished Names (DNs). In this draft, we define how to organize a collection of LDAP servers to be able to resolve and do searches on such DNs. We also describe how to select a "good" instances from among replicated instances of such servers, using DNS SRV and LOC RRs. We preserve the ability to integrate with the X.500 directory.

2. Overview

The problem we wish to solve is: Starting with a LDAP distinguished name (DN), find an LDAP server that holds the naming context (NC) containing the entry denoted by that DN.

We say that a DN is "DNS-based" if it has a suffix which has been constructed according to the rules of \[2\] - i.e., it ends with one or more DC components. Otherwise, we say it is "X.500 based".

Using these definitions, we can now be more precise about the definition of native Internet LDAP server: a native Internet LDAP server is one that whose NCs have DNs that are all DNS-based.
We assume that all native LDAP servers have DNS names.

DO NOT IMPLEMENT TO THIS DOCUMENT!
We assume that there exists a procedure, which is outside the scope of this draft, that servers can use for generating referrals for X.500 based DNs if a DN is not in the NC of the server.

That leaves us with the task of specifying how a client selects the initial LDAP server to contact to resolve a DN, and the procedure for a server to use to generate referrals for DNS-based DNs if a DN is not in the NC of the server.

## 3. Client behavior

An LDAP client resolves a "target" DN using the following procedure:

1. Obtain the DNS name of an LDAP server, based on the type of the DN:
   - DNS-based
     It should extract the DNS name from the target DN using the rules in [2]. For example, if the DN is "CN=Joe%20Blow,DC=somecorp,DC=com", then the DNS name is "somecorp.com".
   - X.500-based
     - It MAY select the DNS name of any LDAP server that it knows, since any LDAP server is likely to at least able to give it a referral to a more appropriate LDAP server. If it happens that a particular selection fails, then it can just select another LDAP server.
     - All clients MUST be configured to know the DNS name of one "default" LDAP server.

2. Resolve the DNS name according to the rules in [7] or [8] to open a connection to the LDAP server, then send the request to it.

3. If the result is a referral, extract the DNS name of the server, then go to step 2.

Optimizations in step 1 that apply regardless of the type of the DN:

- Clients MAY be configured to know one or more NCs and the DNS name of the LDAP server(s) for those NCs. If so, it SHOULD select one of the servers holding the NC with the longest DN that matches a suffix of the target DN.
- Clients SHOULD cache the DNs and server DNS names from recently received referrals. The server name associated with
the cached DN with the matching longest suffix in common with
the target SHOULD be used.

DO NOT IMPLEMENT TO THIS DOCUMENT!  [Page 4]
4. Server behavior

An LDAP server, upon receipt of a request, has to decide to either process the request or return a referral to another LDAP server.

It would process the request itself if the DN specified in the request is in one of the naming contexts handled by the server, and not in any subordinate naming context. (If it is a gateway or proxy, there may have be no such naming contexts.)

Otherwise, it MAY chain the request to another server (if allowed by the request and the server's configuration); if it chooses to do this, it should behave as specified by the client procedure in the previous section.

Otherwise, if is a DNS-based name (the DN specified in the request has DC components), it should create a DNS name from it as specified in [2], and return it in a referral.

Otherwise, it is an X.500-based DN, and it create a referral using the presumed-to-exist procedure that is outside the scope of this document.

5. Security Considerations

TBS

6. References


01/15/1996.

DO NOT IMPLEMENT TO THIS DOCUMENT! [Page 5]


7. Author's address

Paul J. Leach
Microsoft
1 Microsoft Way
Redmond, Washington, 98052, U.S.A.
Email: paulle@microsoft.com