Directory Entries From Email Address

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. Internet-Drafts may be updated, replaced, or made obsolete by other documents at any time. It is not appropriate to use Internet-Drafts as reference material or to cite them other than as a "working draft" or "work in progress".

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

Distribution of this document is unlimited.

Abstract

This draft describes various means for finding a user’s directory entry in a LDAP directory presuming that the user’s electronic mail address is known. This draft does not presume any specific DIT structure or schema modifications.

1. Mechanism

It is crucial to the success of finding services in the Internet that SRV records as defined in [1] be deployed. This draft shows how these records can be used in a straightforward manner to assist in the location of user records. First, assume that a users email address is of the form: name@domain, and domain is of the form: dcn. ... .dc0.tld, where: tld is a top level domain, dc0 is an IETF registered domain name, and dcn through dcl are locally administer subdomains of dcl, and n is greater than or equal to 0. Examples of
valid name forms are:
- bgg@novell.com
- user@scvwd.ca.us
- foo@bar2.bar1.bar0.za

In order to find the directory entry that corresponds to these email addresses, the dc0.tld component of the addresses is first extracted and used in a DNS lookup for an LDAP service according to the mechanisms defined in [1]. If such a service is found, then an LDAP subtree search for a person object with a "mail" attribute EQUAL to the known email address is then used. It is presumed that most directory services will be optimized for fast lookups based on email addresses. If the email address is valid, and the LDAP server for the registered domain either has an entry for the person, or can generate a referral to another directory server (possibly non-LDAP, e.g. X.500, Whois++, etc.), then we’re done, and we have (or will shortly have) the directory entry in question.

On the other hand, if the search fails, there are several avenues available to help find this user.

- the timeLimit parameter of the session control can be raised to a higher limit.
- do a SUBSTRING search against the "mail" attribute with just the name component
- an LDAP service for the dc1.dc0.tld can be looked up, again according to the definitions in [1]
- a well known indexing [2] Internet directory service can be queried for the email address

Note that it is possible that there is no directory entry for the user, in which case all possible lookups will fail. If the user’s email address and directory entry are controlled by different domains with no links between the two domains, it will not be possible to find the user’s directory entry from the email address initially, but if an Indexed Internet directory that has retrieved the user’s directory entry from the second domain, then it is likely that the Indexed Internet directory will be able to generate a referral to the appropriate domain, even though we initially started out with no direct information about that domain. For example, a directory service for a small Internet Service Provider (smallisp.com) might be
maintained by a wider area Directory Service (bigldap.org) on a contract basis. Thus, the search for an LDAP service for smallisp.com might fail, but the ldap lookup to the Indexing Internet Directory would result in a referral to bigldap.com. What is more likely to be the case is that smallisp.com will create an SRV record for its LDAP service that points to bigldap.com.

2. References
