Service Location Working Group
INTERNET DRAFT

Jonathan Wood
Roberto Tam
Sun Microsystems, Inc.
25 June 1999

The LDAP Service Type
draft-ieft-svrloc-ldap-scheme-02.txt

Status of This Memo

This document is a submission by the Service Location Working Group of the Internet Engineering Task Force (IETF). Comments should be submitted to the srvloc@srvloc.org mailing list.

Distribution of this memo is unlimited.

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026. 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.''

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Abstract

This document describes the LDAP service type. This service type defines the service: URL and attributes necessary for discovering LDAP servers.

1. Introduction

This document describes a template providing a service: URL and attributes useful for dynamically discovering LDAP servers; this type can be used with SLP [1]. Service templates and service: schemes are defined in [2].

LDAP (Lightweight Directory Access Protocol) [3] directories are

now being used as repositories for UNIX-style system information. As such, LDAP service is suitable to be included in the naming-directory class. This type is intended to be used as a concrete portion of the abstract naming-directory type defined in [4]. The LDAP type includes all attributes from the naming-directory abstract type, and defines new attributes pertaining to security and access protocols.

For usage examples for non-LDAP specific scenarios, refer to [4].

2. Differentiating Servers by Authentication Mechanisms

Possible means of authenticating LDAP transactions are defined in [5]. In order to agree on a particular authentication mechanism, a client and server might need to go through a number of iterations and levels of negotiation. Currently there are three levels of mechanisms: The first level consists of the basic mechanisms, anonymous, simple, and TLS. The second layer consists of the SASL negotiation layer [6]. The third layer consists of GSSAPI [7] mechanisms, and possible the GSS-SPNEGO negotiation sequence [8]. Thus it is possible that a client wishing to use the Kerberos V5 GSSAPI mechanism may need to negotiate its way through SASL and GSS-SPNEGO before coming to an agreement with the server.

Since LDAP clients are already aware of what mechanisms they have been configured to use when connecting to an LDAP server, the attributes in this template have been designed to allow clients to optimize both their search for servers and the following negotiation sequence for authentication mechanisms. Clients may specify as little or as much of their desired negotiation path. For example, all of the following SLP [1] search filters are valid:

(security=sasl)

(&(security=sasl)(|(sasl-mechs=cram-md5)(sasl-mechs=external)))

(&(security=sasl)(sasl-mechs=GSSAPI)(gss-mechs=1.3.5.1.5.2))

The more fully a client specifies the negotiation path, the greater the likelihood that the client will discover a server which supports the same mechanisms as the client. If a server does not support the required mechanisms, clients will need to move on to other discovered servers, and repeat the negotiation process. This can be a costly process. Additionally, clients should be able to optimize the resulting negotiation, bypassing mechanisms which are not acceptable to one of the parties.

The allowable values for the "security" attribute follows those

defined in [5].

The allowable values for the "sasl-mechs" and "gss-mechs" attributes are meant to be fluid, following the decisions of their respective working groups.

3. The LDAP Service Type

Names of submitters: Jonathan Wood <jonathan.wood@eng.sun.com>
Roberto Tam <roberto.tam@eng.sun.com>
Language of service template: en
Security Considerations:
This LDAP service type inherits the security considerations from the naming-directory service type [4], the SLP specification [1]. Implementors should also be aware of the security considerations discussed in [5].

Template text:

```
------------------------template begins here----------------------
template-type=naming-directory:ldap

template-version=0.0

template-description=
  This is a concrete type; the abstract type for this service
  is naming-directory (described in [4]). This type is used
  by LDAP servers to advertise their services and LDAP clients
  which wish to discover LDAP servers.

template-url=syntax=
  url-path = ldap URL as defined in [9]

security=string M
  # Security mechanisms supported by this server. Permitted values
  # are drawn from draft-ietf-ldapext-authmeth-03 (Authentication
  # Methods for LDAP). If SASL is supported, clients may further
  # differentiate servers with the sasl-mechs attribute.
anonymous,simple,tls,sasl

sasl-mechs=string M
  # SASL mechanisms supported by this server. If the GSSAPI or GSS-SPNEGO
  # mechanisms are supported, clients may further differentiate servers
  # with the gss-mechs attribute. SASL mechanisms are registered with
  # IANA; legal values of this attribute are the mechanism keywords
  # registered with IANA. SASL is defined in RFC 2222.

gss-mechs=string M
  # GSSAPI mechanisms supported by this server. The mechanisms are
```

```
  # named by their object identifiers (OIDs). GSSAPI is defined
  # in RFC 2078, and GSS-SPNEGO is defined in RFC 2478.

qop= string
  # quality of protection. The refers to how strongly messages are
  # protected. There are three possibilities: none, integrity
  # (meaning that the integrity and endpoints of the message can
  # be guaranteed), and privacy (meaning that the message is
  # encrypted).
none,integrity,privacy

transport= string
  # the transport used to communicate with this server. Possible
  # values are connection-oriented (cots) and connectionless
  # (clts).
cots,clts

version= string M
  # Which version(s) of LDAP this server supports. "v3" corresponds to
  # the protocol as defined by RFC 2251, and "v2" corresponds to the
  # protocol as defined by RFC 1777.
v2,v3

extensions= string M
  # This is an open-ended attribute intended to contain any standard or
  # non-standard (i.e. vendor-specific) extensions this server supports.

--------------------------template ends here-----------------------
```

References:

[1] E. Guttman, C. Perkins, J. Veizades, M. Day. Service Location Protocol. RFC 2608, April 1999

[2] E. Guttman, C. Perkins, J. Kempf, Service Templates and service: Schemes. RFC 2609, February 1999

[3] W. Yeong, T. Howes, S. Kille, Lightweight Directory Access Protocol, RFC 1777 March 1995

[4] J. Wood, R. Tam, The Naming and Directory Service Abstract Type. draft-ietf-svrloc-naming-directory-01.txt, June 1999 (work in progress)

[5] M. Wahl, H. Alvestrand, J. Hodges, RL Morgan. Authentication Methods for LDAP, draft-ietf-ldapext-authmeth-04.txt, November 1998 (work in progress)

[6] J. Meyers, Simple Authentication and Security Layer (SASL) RFC 2222 October 1997

[7] J. Linn, Generic Security Service Application Program Interface, Version 2, RFC 2078 January 1997

[8] E. Baize, D. Pinkas, The Simple and Protected GSS-API Negotiation Mechanism, RFC 2478 December 1998

[9] T. Howes, M. Smith, The LDAP URL Format, RFC 2255 December 1997