INTERNET-DRAFT
draft-ietf-ldup-subentry-08.txt

Ed Reed
Reed-Matthews, Inc.
April 6, 2001

LDAP Subentry Schema

1 Status of this Memo

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.

This Internet-Draft expires on October 6, 2001.

2 Abstract / Description

This document describes an administrative model for LDAP, and an object class called ldapSubEntry and a control ldapSubentriesControl (to control the visibility of entries of type ldapSubEntry) that are to be used by directory servers claiming support for the administrative model defined here.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119]. The sections below reiterate these definitions and include some additional ones.

3 Table of Contents

4 Administrative Areas in the Directory

4.1 X.501 Administrative Model Overview

[X501] contains the definitive description of Administrative Areas and their role in the management and administration of directories. The LDAP administrative model defined here is intended to be a compatible, proper subset of the [X501] model. The description here draws heavily on the descriptions and concepts laid out in [X501].

An administrative area is a sub-tree of the directory information tree, rooted at an administrative point (the root-most entry in the sub-tree), where administrative entries (perhaps including subentries, operational attributes, or both) are located. Autonomous administrative areas are distinct partitions of the directory information tree whose entries are all administered by a single administrative authority. Each entry in the directory information tree is administered by

exactly one autonomous administrative authority.

There may be many aspects of administration defined by the directory and other applications for specific purposes, such as subschema administration areas, access control administration areas, collective-attribute administration areas, context default administrative areas, and service administrative areas. Within an autonomous administrative area, specific administrative areas for these (and other) different aspects may overlap one another.

Specific administrative areas may be sub-partitioned by the applications or services which define them to facilitate delegation of authority or for other purposes. That means that a single entry in the directory may be part of many different specific administrative areas, but only be part of one specific administrative area (or sub-area) of each aspect of administration.

The [X501] subentry specification optionally uses a SubtreeSpecification to indicate a subset of entries in a sub-tree with which the subentry is concerned. When the SubtreeSpecification is empty the scope of the [X501] subentry is implicitly defined by the context in which it occurs.

4.2 An LDAP Administrative Model

The administrative model for LDAP defined here is a simplified version of the one described in [X501], in that the scope defined for the ldapSubentry object class is limited.

The LDAP Subentry definition below specifically does not include a SubtreeSpecification, so its scope is explicitly the complete set of entries in the specific administrative area (or sub-area) in which it occurs. All administrative areas are considered to be specific administrative areas within an autonomous administrative area.

If a specific administration area is not partitioned, then its extent (or scope) is said to be that of the autonomous administrative area in which it is defined.

Applications and services which define specific administrative areas must specify whether the areas may be partitioned or not. By default, the scope of LDAP

Subentries is limited to the sub-area of the partitioned specific administrative area in which they are present.

5 Object Class Definition

5.1 ldapSubEntry Class

```
( 2.16.840.1.113719.2.142.6.1.1 NAME 'ldapSubEntry'
   DESC 'LDAP Subentry class, version 1'
     SUP top STRUCTURAL
     MAY ( cn ) )
```

The class ldapSubEntry is intended to be used as a super-class when defining other structural classes to be used as LDAP Subentries, and as the structural class to which Auxiliary classes may be added for application specific subentry information. Where possible, the use of Auxiliary classes to extend LDAP Subentries is strongly preferred.

The presence of ldapSubEntry in the list of super-classes of an entry in the directory makes that entry an LDAP Subentry. Object classes derived from ldapSubEntry are themselves considered ldapSubEntry classes, for the purpose of this discussion.

5.1.1Naming and Structure constraints

LDAP Subentries MAY be named by their commonName attribute [RFC2251]. Other naming attributes are also permitted. For compatibility with [X501], the commonName attribute is optional ([X501] requires EITHER cn OR a SubTreeSpecification), but note that in the absence of any other naming attribute, a cn is required to name the LDAP Subentry.

LDAP Subentries MAY be containers, unlike their [X501] counterparts. This is a departure from [X501], but is considered an important extension to increase the ability to more easily construct richer (i.e., more complex) policy representations in the directory using LDAP Subentries. Using LDAP Subentry containers to hold entries that are not themselves LDAP Subentries is prohibited, as that would significantly affect compatibility with [X501] services.

LDAP Subentries MAY be contained by, and will usually be located in the directory information tree immediately

subordinate to their administrative points. LDAP
Subentries MAY also be contained by other LDAP Subentries
(the way organizational units may be contained by other

organizational units). Deep nesting of LDAP Subentries are discouraged, but not prohibited. Developers are warned that deep nesting of LDAP Subentries may not be supported by all (or indeed, by any) LDAP server implementations. For compatibility with [X501], a sub-tree made up of a collection of LDAP Subentries may be mapped onto a single (possibly very complex) [X501] subentry, and vice versa.

5.1.2Scope Rules

The default scope of an LDAP Subentry is limited to the specific administrative area (or sub-area) in which it is defined. Specifically, the subtree of the directory namespace based at the administrative point most immediately superior to the LDAP Subentry, down to but not including any subordinate administrative points or areas of the same aspect or type. Policy defined in an LDAP Subentry is not inheritable, unless such inheritance is explicitly defined by an application-specific policy.

If an LDAP Subentry is subordinate to another LDAP Subentry, it takes the same default scope as the parent LDAP Subentry.

Applications MAY define alternative scope semantics for classes they define which are derived from the ldapSubEntry class. This means that an application can derive a new class from the ldapSubEntry class and add an attribute, like SubTreeSpecification [X501] to define a new scope rule for that application to use.

Applications MUST NOT define alternative scope rules for auxiliary classes used to decorate entries of the ldapSubEntry class. This restriction is required to avoid having conflicting or contradictory scope definitions applied by different applications to the same LDAP Subentry.

6 Visibility Control

6.1 ldapSubentriesControl

This control is included in the searchRequest message as part of the controls field of the LDAPMessage, as defined in Section 4.1.12 of [RFC2251].

The controlType is set to "1.3.6.1.4.1.7628.5.101.1". The criticality MAY be set to either TRUE or FALSE. The controlValue is absent.

There is no corresponding response control defined.

LDAP servers that support this control MUST treat LDAP Subentries as "operational objects" in much the same way that "operational attributes" are not returned in search results and [X511] read operations when only user attributes are requested.

Entries which are not LDAP Subentries may be referenced in the base object of search operations where the ldapSubentriesControl is present in the request.

In the absence of the LDAP Subentries visibility control, subentries are not visible to search operations UNLESS the target/base of the operation is a subentry.

In presence of the subentry visibility control, ONLY subentries are visible.

6.1.1Other LDAP operations

The ldapSubentriesControl is not defined for any LDAP operation other than Search. However, an LDAPv3 Extension MAY define a use of this control with that extension as long as such use is consistent with this specification.

6.1.2Correspondence to [X511]

In presence of the visibility control, the semantics of the LDAPSubentriesControl are in accordance with the indication of the [X511] common argument serviceControls options

subentries being set.

In [X511] a ServiceControl option is used to govern the visibility of [X501] subentries. The subentry ServiceControl option is a specific bit of a bitstring that, when set in the common arguments of an [X511] Search or List operation, indicates that the operation is to access ONLY the subentries found in the context of the list or search. In fact, normal entries are explicitly NOT returned in the result of a list or search operation when the [X511] subentries ServiceControl is set.

Entries which are not subentries may be used in the base object of list and search operations where the subentries control is set.

The [X511] subentries ServiceControl has no meaning for operations other than Search and List (i.e., it is not defined for Read, Modify, Delete, etc.).

7 Security Considerations

LDAP Subentries will frequently be used to hold data which reflects either the actual or intended behavior of the directory service. As such, permission to read such entries MAY need to be restricted to authorized users. More importantly, if a directory service treats the information in an LDAP Subentry as the authoritative source of policy to be used to control the behavior of the directory, then permission to create, modify, or delete such entries MUST be carefully restricted to authorized administrators.

8 References

[RFC2119] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March 1997

[RFC2251] S. Kille, M. Wahl, and T. Howes, "Lightweight Directory Access Protocol (v3)", RFC 2251, December 1997

[X501] ITU-T Rec. X.501, "The Directory: Models", 1993 and subsequent versions

[X501] ITU-T Rec. X.511, "The Directory: Abstract Service Definition", 1993 and subsequent versions

9 Copyright Notice

10 Acknowledgements

The utility of subEntry object class was originally suggested as a means to store Replica and Replication Agreement information with a the lucid explanation by Mark Wahl, (then of Innosoft), of how they could be used and extended.

which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the

IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

11 Author's Address

Edwards E. Reed
Reed-Matthews, Inc.
1064 E 140 North
Lindon, UT 84042
USA
E-mail: eer@oncalldba.com

LDUP Mailing List: ietf-ldup@imc.org
LDAPEXT Mailing List: ietf-ldapext@netscape.com

Expires October 6, 2001