Network Working Group
Request for Comments: 4373
Category: Informational

R. Harrison
J. Sermersheim
Novell, Inc.
Y. Dong

January 2006

Lightweight Directory Access Protocol (LDAP) Bulk Update/Replication Protocol (LBURP)

Status of This Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Abstract

The Lightweight Directory Access Protocol (LDAP) Bulk Update/Replication Protocol (LBURP) allows an LDAP client to perform a bulk update to an LDAP server. The protocol frames a sequenced set of update operations within a pair of LDAP extended operations to notify the server that the update operations in the framed set are related in such a way that the ordering of all operations can be preserved during processing even when they are sent asynchronously by the client. Update operations can be grouped within a single protocol message to maximize the efficiency of client-server communication.

The protocol is suitable for efficiently making a substantial set of updates to the entries in an LDAP server.

Table of Contents

1. Introduction

The Lightweight Directory Access Protocol (LDAP) Bulk Update/Replication Protocol (LBURP) arose from the need to allow an LDAP client to efficiently present large quantities of updates to an LDAP server and have the LDAP server efficiently process them. LBURP introduces a minimum of new operational functionality to the LDAP protocol because the update requests sent by the client encapsulate standard LDAP [RFC2251] update operations. However, this protocol greatly facilitates bulk updates by allowing the client to send the update operations asynchronously and still allow the server to maintain proper ordering of the operations. It also allows the server to recognize the client’s intent to perform a potentially large set of update operations and then to change its processing strategy to more efficiently process the operations.

2. Conventions Used in This Document

Imperative keywords defined in RFC 2119 [RFC2119] are used in this document, and carry the meanings described there.

All Basic Encoding Rules (BER) [X.690] encodings follow the conventions found in section 5.1 of [RFC2251].

The term "supplier" applies to an LDAP client or an LDAP server (acting as a client) that supplies a set of update operations to a consumer.

The term "consumer" applies to an LDAP server that consumes (i.e., processes) the sequenced set of update operations sent to it by a supplier.

3. Overview of Protocol

LBURP frames a set of update operations within a pair of LDAP extended operations that mark the beginning and end of the update set. These updates are sent via LDAP extended operations, each containing a sequence number and a list of one or more update operations to be performed by the consumer. Except for the fact that they are grouped together as part of a larger LDAP message, the update operations in each subset are encoded as LDAP update operations and use the LDAP Abstract Syntax Notation One (ASN.1) [X.680] message types specified in [RFC2251].

3.1. Update Initiation

The protocol is initiated when a supplier sends a StartLBURPRequest extended operation to a consumer as a notification that a stream of associated LBURPUpdateRequests will follow. The supplier associates semantics with this stream of requests by including the Object Identifier (OID) of the bulk update/replication style in the StartLBURPRequest. The consumer responds to the StartLBURPRequest with a StartLBURPResponse message.

3.2. Update Stream

After the consumer responds with a StartLBURPResponse, the supplier sends a stream of LBURPUpdateRequest messages to the consumer. Messages within this stream may be sent asynchronously to maximize the efficiency of the transfer. The consumer responds to each LBURPUpdateRequest with an LBURPUpdateResponse message.

3.2.1. LBURPUpdateRequest

Each LBURPUpdateRequest contains a sequence number identifying its relative position within the update stream and an UpdateOperationList containing an ordered list of LDAP update operations to be applied to the Directory Information Tree (DIT). The sequence number enables the consumer to process LBURPUpdateRequest messages in the order they were sent by the supplier even when they are sent asynchronously. The consumer processes each LBURPUpdateRequest according to the sequence number by applying the LDAP update operations in its UpdateOperationList to the DIT in the order they are listed.

3.2.2. LBURPUpdateResponse

When the consumer has processed the update operations from an UpdateOperationList, it sends an LBURPUpdateResponse to the supplier indicating the success or failure of the update operations contained within the corresponding LBURPUpdateRequest.

3.3. Update Termination

After the supplier has sent all of its LBURPUpdateRequest messages, it sends an EndLBURPRequest message to the consumer to terminate the update stream. Upon servicing all LBURPOperation requests and receiving the EndLBURPRequest, the consumer responds with an EndLBURPResponse, and the update is complete.

3.4. Applicability of Protocol

LBURP is designed to facilitate the bulk update of LDAP servers. It can also be used to synchronize directory information between a single master and multiple slaves.

No attempt is made to deal with the issues associated with multiple-master replication environments (such as keeping modification times of attribute values) so that updates to the same entry on different replicas can be correctly ordered. For this reason, when LBURP alone is used for replication, proper convergence of the data between all replicas can only be assured in a single-master replication environment.

4. Description of Protocol Flow

This section describes the LBURP protocol flow and the information contained in each protocol message. Throughout this section, the client or server acting as a supplier is indicated by the letter "S", and the server acting as a consumer is indicated by the letter "C". The construct "S -> C" indicates that the supplier is sending an LDAP message to the consumer, and "C -> S" indicates that the consumer is sending an LDAP message to the supplier. Note that the protocol flow below assumes that a properly authenticated LDAP session has already been established between the supplier and consumer.

S -> C: StartLBURPRequest message. The parameter is:

1) OID for the LBURP update style (see section 5.1.1).

C -> S: StartLBURPResponse message. The parameter is:

1) An optional maxOperations instruction (see section 5.2.1).

S -> C: An update stream consisting of zero or more LBURPUpdateRequest messages. The requests MAY be sent asynchronously. The parameters are:

1) A sequence number specifying the order of this LBURPUpdateRequest with respect to the other LBURPUpdateRequest messages in the update stream (see section 5.3.1).

2) LBURPUpdateRequest.updateOperationList, a list of one or more LDAP update operations (see section 5.3.2).

The consumer processes the LBURPUpdateRequest messages in the order of their sequence numbers and applies the LDAP update operations contained within each LBURPUpdateRequest to the DIT in the order they are listed.

C -> S: LBURPUpdateResponse message. This is sent when the consumer completes processing the update operations from each LBURPUpdateRequest.updateOperationList.

S -> C: EndLBURPRequest message. This is sent after the supplier sends all of its LBURPUpdateRequest messages to the consumer. The parameter is:

1) A sequence number that is one greater than the sequence number of the last LBURPUpdateRequest message in the update stream. This allows the EndLBURPRequest to also be sent asynchronously.

C -> S: EndLBURPResponse message. This is sent in response to the EndLBURPRequest after the consumer has serviced all LBURPOperation requests.

5. Elements of Protocol

LBURP uses two LDAP ExtendedRequest messages--StartLBURPRequest and EndLBURPRequest--to initiate and terminate the protocol. A third LDAP ExtendedRequest message--LBURPUpdateRequest--is used to send update operations from the supplier to the consumer. These three requests along with their corresponding responses comprise the entire protocol.

LBURP request messages are defined in terms of the LDAP ExtendedRequest [RFC2251] as follows:

```
     ExtendedRequest ::= [APPLICATION 23] SEQUENCE {
         requestName    [0] LDAPOID,
         requestValue   [1] OCTET STRING OPTIONAL
     }
```

LBURP response messages are defined in terms of the LDAP ExtendedResponse [RFC2251] as follows:

```
    ExtendedResponse ::= [APPLICATION 24] SEQUENCE {
        COMPONENTS of LDAPResult,
        responseName  [10] LDAPOID OPTIONAL,
        response      [11] OCTET STRING OPTIONAL
     }
```

5.1. StartLBURPRequest

The requestName value of the StartLBURPRequest is OID 1.3.6.1.1.17.1.

The requestValue of the StartLBURPRequest contains the BER-encoding of the following ASN.1:

```
    StartLBURPRequestValue ::= SEQUENCE {
        updateStyleOID LDAPOID
    }
```

LDAPOID is defined in [RFC2251], section 4.1.2.

5.1.1. updateStyleOID

The updateStyleOID is an OID that uniquely identifies the LBURP update style being used. This document defines one LBURP update semantic style that can be transmitted between the StartLBURPRequest and EndLBURPRequest. The updateStyleOID is included in the protocol for future expansion of additional update styles. For example, a future specification might define an update style with semantics to replace all existing entries with a new set of entries and thus only allows the Add operation.

The updateStyleOID for the LBURP Incremental Update style is 1.3.6.1.1.17.7. The semantics of this update style are described in section 6.

5.2. StartLBURPResponse

The responseName of the StartLBURPResponse is the OID 1.3.6.1.1.17.2.

The optional response element contains the BER-encoding of the following ASN.1:

```
    StartLBURPResponseValue ::= maxOperations

    maxOperations ::= INTEGER (0 .. maxInt)

    maxInt INTEGER ::= 2147483647 -- (2^^31 - 1) --
```

5.2.1. maxOperations

When present, the value of maxOperations instructs the supplier to send no more than that number of update operations per LBURPUpdateRequest.updateOperationList (see section 5.3.2). If the consumer does not send a maxOperations value, it MUST be prepared to accept any number of update operations per LBURPUpdateRequest.updateOperationList. The supplier MAY send fewer but MUST NOT send more than maxOperations update operations in a single LBURPUpdateRequest.updateOperationList.

5.3. LBURPUpdateRequest

The LBURPUpdateRequest message is used to send a set of zero or more LDAP update operations from the supplier to the consumer along with sequencing information that enables the consumer to maintain the proper sequencing of multiple asynchronous LBURPUpdateRequest messages.

The requestName of the LBURPUpdateRequest is the OID 1.3.6.1.1.17.5.

The requestValue of an LBURPOperation contains the BER-encoding of the following ASN.1:

```
    LBURPUpdateRequestValue ::= SEQUENCE {
        sequenceNumber INTEGER (1 .. maxInt),
        updateOperationList UpdateOperationList
    }
```

5.3.1. sequenceNumber

The sequenceNumber orders associated LBURPOperation requests. This enables the consumer to process LBURPOperation requests in the order specified by the supplier. The supplier MUST set the value of sequenceNumber of the first LBURPUpdateRequest to 1, and MUST increment the value of sequenceNumber by 1 for each succeeding LBURPUpdateRequest. In the unlikely event that the number of LBURPUpdateRequest messages exceeds maxInt, a sequenceNumber value of 1 is deemed to be the succeeding sequence number following a sequence number of maxInt.

5.3.2. UpdateOperationList

The UpdateOperationList is a list of one or more standard LDAP update requests and is defined as follows:

```
    UpdateOperationList ::= SEQUENCE OF SEQUENCE{
        updateOperation CHOICE {
           addRequest       AddRequest,
           modifyRequest    ModifyRequest,
           delRequest       DelRequest,
           modDNRequest     ModifyDNRequest
        },
        controls       [0] Controls OPTIONAL
    }
```

AddRequest, ModifyRequest, DelRequest, and ModifyDNRequest are defined in [RFC2251], sections 4.6, 4.7, 4.8, and 4.9.

The LDAP update requests in the UpdateOperationList MUST be applied to the DIT in the order in which they are listed.

5.4. LBURPUpdateResponse

An LBURPUpdateResponse message is sent from the consumer to the supplier to signal that all of the update operations from the UpdateOperationList of an LBURPUpdateRequest have been completed and to give the results for the update operations from that list.

The responseName of the LBURPUpdateResponse is the OID 1.3.6.1.1.17.6.

If the consumer server cannot successfully decode an LBURPUpdateRequest in its entirety, the resultCode for the corresponding LBURPUpdateResponse is set to protocolError and the response element is omitted. Updates from the LBURPUpdateRequest SHALL NOT be committed to the DIT in this circumstance.

If the status of all of the update operations being reported by an LBURPUpdateResponse message is success, the resultCode of the LBURPUpdateResponse message is set to success and the response element is omitted.

If the status of any of the update operations being reported by an LBURPUpdateResponse message is something other than success, the resultCode for the entire LBURPUpdateResponse is set to other to signal that the response element is present.

5.4.1. OperationResults

When a response element is included in an LBURPUpdateResponse message, it contains the BER-encoding of the following ASN.1:

```
    OperationResults ::= SEQUENCE OF OperationResult

    OperationResult ::= SEQUENCE {
       operationNumber    INTEGER,
       ldapResult         LDAPResult
    }
```

An OperationResult is included for each operation from the UpdateOperationList that failed during processing.

5.4.1.1. operationNumber

The operationNumber identifies the LDAP update operation from the UpdateOperationList of the LBURPUpdateRequest that failed. Operations are numbered beginning at 1.

5.4.1.2. ldapResult

The ldapResult included in the OperationResult is the same ldapResult that would be sent for the update operation that failed if it had failed while being processed as a normal LDAP update operation. LDAPResult is defined in [RFC2251], section 4.1.10.

5.5. EndLBURPRequest

The requestName of the EndLBURPRequest is the OID 1.3.6.1.1.17.3.

The requestValue contains the BER-encoding of the following ASN.1:

```
     EndLBURPRequestValue::= SEQUENCE {
         sequenceNumber INTEGER (1 .. maxInt)
     }
```

5.5.1. sequenceNumber

The value in sequenceNumber is one greater than the last LBURPUpdateRequest.sequenceNumber in the update stream. It allows the server to know when it has received all outstanding asynchronous LBURPUpdateRequests.

5.6. EndLBURPResponse

The responseName of the EndLBURPResponse is the OID 1.3.6.1.1.17.4.

There is no response element in the EndLBURPResponse message.

6. Semantics of the Incremental Update Style

The initial state of entries in the consumer’s DIT plus the LBURPUpdateRequest messages in the update stream collectively represent the desired final state of the consumer’s DIT. All LDAP update operations defined in [RFC2251]--Add, Modify, Delete, and Modify DN--are allowed in the incremental update stream. All of the semantics of those operations are in effect, so for instance, an attempt to add an entry that already exists will fail just as it would during a normal LDAP Add operation.

7. General LBURP Semantics

The consumer server may take any action required to efficiently process the updates sent via LBURP, as long as the final state is equivalent to that which would have been achieved if the updates in the update stream had been applied to the DIT using normal LDAP update operations.

The LBURPUpdateRequest messages that form the update stream MAY be sent asynchronously by the supplier to the consumer. This means that the supplier need not wait for an LBURPUpdateResponse message for one LBURPUpdateRequest message before sending the next LBURPUpdateRequest message.

When the LBURP update stream contains a request that affects multiple Directory System Agents (DSAs), the consumer MAY choose to perform the request or return a resultCode value of affectsMultipleDSAs. As with any LDAP operation, a consumer MAY send a resultCode value of referral as part of the OperationResult element for any operation on an entry that it does not contain. If the consumer is configured to do so, it MAY chain on behalf of the supplier to complete the update operation instead.

While a consumer server is processing an LBURP update stream, it may choose not to service LDAP requests on other connections. This provision is designed to allow implementers the freedom to implement highly-efficient methods of handling the update stream without being constrained by the need to maintain a live, working DIT database while doing so.

If a consumer chooses to refuse LDAP operation requests from other suppliers during LBURP update, it is RECOMMENDED that the consumer refer those requests to another server that has the appropriate data to complete the operation.

Unless attribute values specifying timestamps are included as part of the update stream, updates made using LBURP are treated the same as other LDAP operations wherein they are deemed to occur at the present. Consumers MAY store timestamp values sent by suppliers but are not required to do so.

Implementations may choose to perform the operations in the update stream with special permissions to improve performance.

Consumer implementations should include functionality to detect and terminate connections on which an LBURP session has been initiated but information (such as the EndLBURPRequest) needed to complete the LBURP session is never received. A timeout is one mechanism that can be used to accomplish this.

8. Security Considerations

Implementations should ensure that a supplier making an LBURP request is properly authenticated and authorized to make the updates requested. There is a potential for loss of data if updates are made to the DIT without proper authorization. If LBURP is used for replication, implementers should note that unlike other replication protocols, no existing replication agreement between supplier and consumer is required. These risks increase if the consumer server also processes the update stream with special permissions to improve performance. For these reasons, implementers should carefully consider which permissions should be required to perform LBURP operations and take steps to ensure that only connections with appropriate authorization are allowed to perform them.

The data contained in the update stream may contain passwords and other sensitive data. Care should be taken to properly safeguard this information while in transit between supplier and consumer. The StartTLS [RFC2830] operation is one mechanism that can be used to provide data confidentiality and integrity services for this purpose.

As with any asynchronous LDAP operation, it may be possible for an LBURP supplier to send asynchronous LBURPUpdateRequest messages to the consumer faster than the consumer can process them. Consumer implementers should take steps to prevent LBURP suppliers from interfering with the normal operation of a consumer server by issuing a rapid stream of asynchronous LBURPUpdateRequest messages.

9. IANA Considerations

Registration of the following values has been made by the IANA [RFC3383].

9.1. LDAP Object Identifier Registrations

The IANA has registered LDAP Object Identifiers identifying the protocol elements defined in this technical specification. The following registration template was provided:

```
Subject: Request for LDAP OID Registration
Person & email address to contact for further information:
    Roger Harrison
    rharrison@novell.com
Specification: RFC 4373
Author/Change Controller: IESG
Comments:
Seven delegations will be made under the assigned OID.  The
following 6 OIDs are Protocol Mechanism OIDs of type "E"
(supportedExtension):

1.3.6.1.1.17.1 StartLBURPRequest LDAP ExtendedRequest message
1.3.6.1.1.17.2 StartLBURPResponse LDAP ExtendedResponse message
1.3.6.1.1.17.3 EndLBURPRequest LDAP ExtendedRequest message
1.3.6.1.1.17.4 EndLBURPResponse LDAP ExtendedResponse message
1.3.6.1.1.17.5 LBURPUpdateRequest LDAP ExtendedRequest message
1.3.6.1.1.17.6 LBURPUpdateResponse LDAP ExtendedResponse message

The following 1 OID is a Protocol Mechanism OID of type "F"
(supportedFeature):

1.3.6.1.1.17.7 LBURP Incremental Update style OID
```

10. Normative References

[RFC2119] Bradner, S., "Key Words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

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

[RFC3383] Zeilenga, K., "Internet Assigned Numbers Authority (IANA) Considerations for the Lightweight Directory Access Protocol (LDAP)", BCP 64, RFC 3383, September 2002.

[X.680]   ITU-T Recommendation X.680 (07/2002) | ISO/IEC 8824-1:2002 "Information Technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation"

[X.690]   ITU-T Rec. X.690 (07/2002) | ISO/IEC 8825-1:2002, "Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", 2002.

11. Informative References

[RFC2830] Hodges, J., Morgan, R., and M. Wahl, "Lightweight Directory Access Protocol (v3): Extension for Transport Layer Security", RFC 2830, May 2000.

Authors' Addresses

Roger Harrison
Novell, Inc.
1800 S. Novell Place
Provo, UT 84606

Phone: +1 801 861 2642
EMail: rharrison@novell.com

Jim Sermersheim
Novell, Inc.
1800 S. Novell Place
Provo, UT 84606

Phone: +1 801 861 3088
EMail: jimse@novell.com

Yulin Dong

EMail: yulindong@gmail.com

Full Copyright Statement

Intellectual Property

Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).