

Link Layer Discovery Protocol

Overview

Paul Congdon

3/8/03

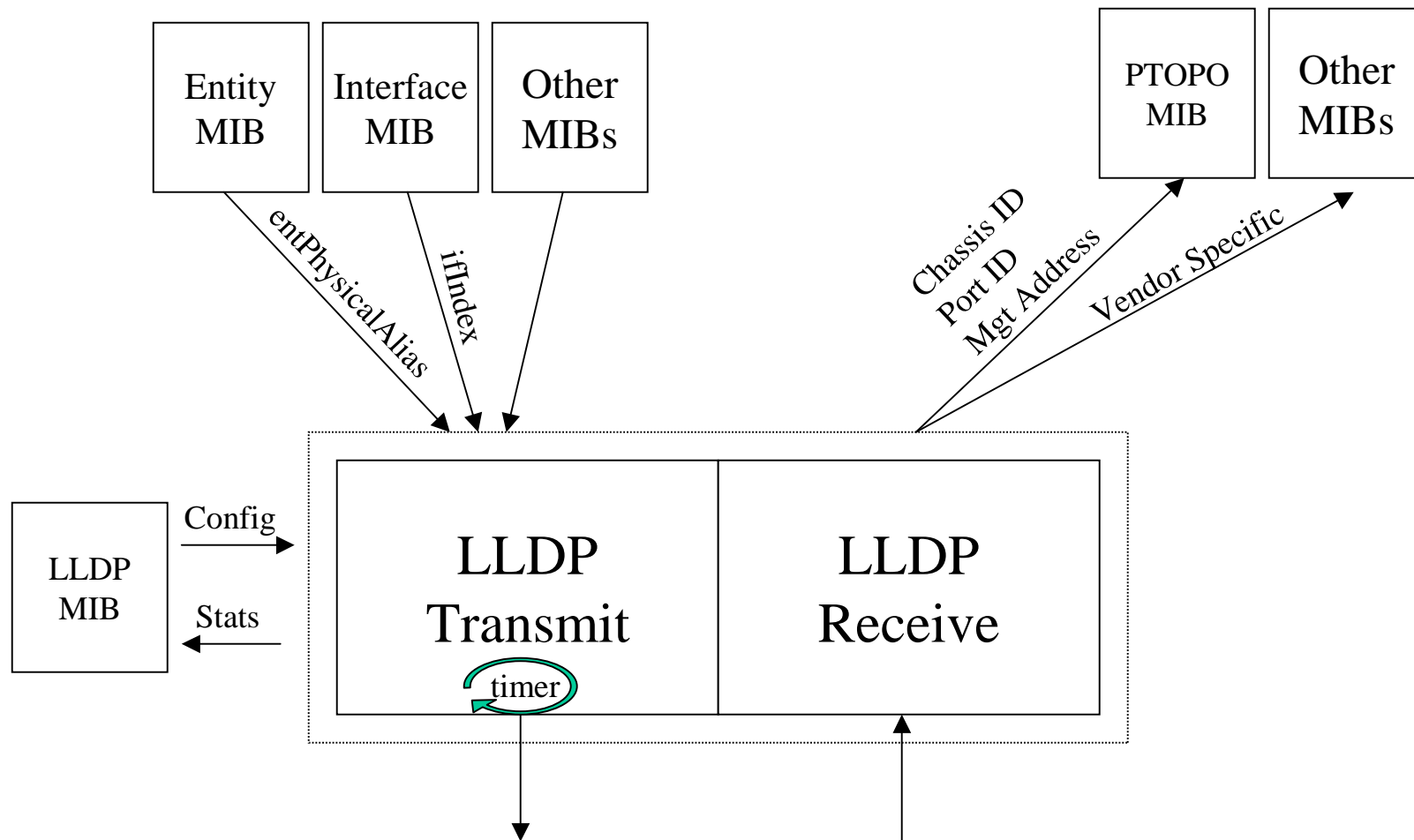
Background

- The bulk of the text is taken from draft-ietf-ptopomib-pdp-03.txt
- Major modifications include:
 - Defined as an IEEE 802 Slow Protocol
 - Replacement of ASN.1 VarBindList with TLVs
 - Vendor specific TLV extension

Protocol Goals/Objectives

- Provide a means to announce necessary information for the purpose of device discovery and physical topology discovery
- Provide a method to populate the PTOPO MIB (RFC 2922)
- Operate under the constraints of Slow Protocols
- Allow for the ageing of announced information.
- Provide a method to accelerated the clean-up of announced information when shutdown is known.
- Constrain the forwarding of announced information
- Allow for vendor extensions of announced information.

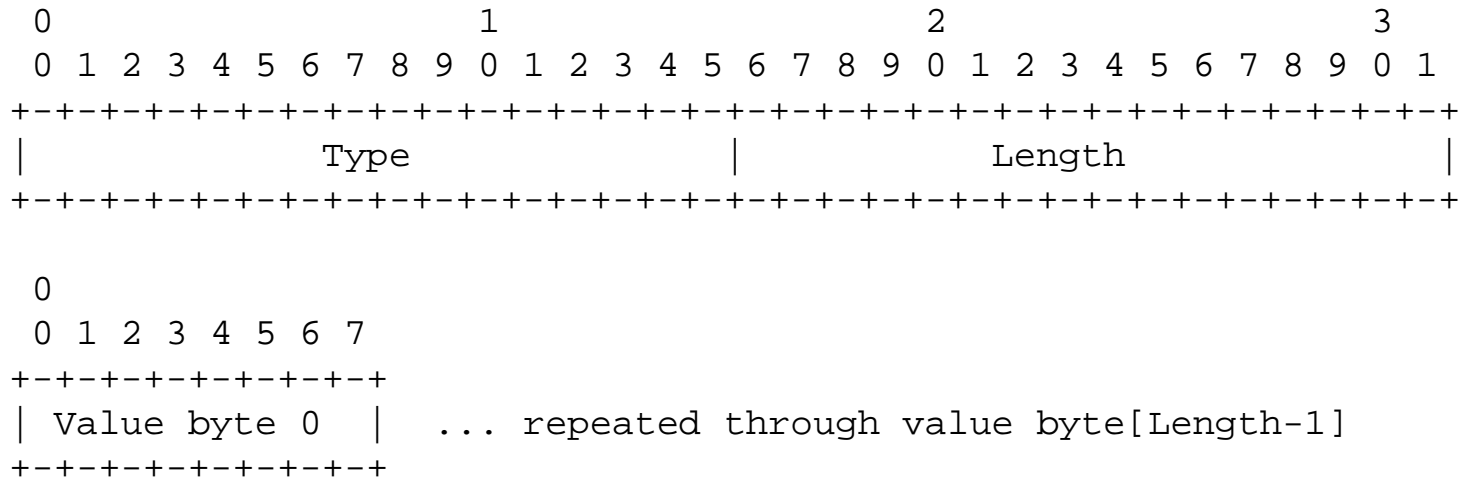
High-Level Operation



What's in the PTOp MIB

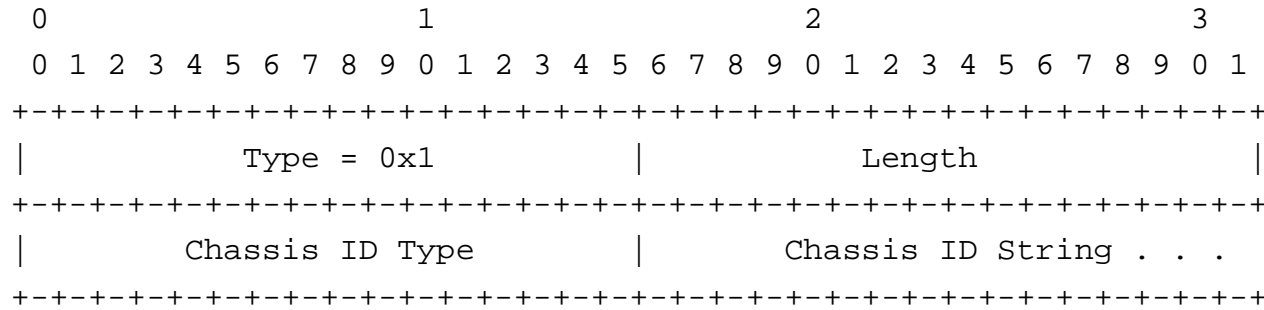
- A connectionTable with information about peer nodes off each port
- Basic Info held for each node known is:
 - Remote systems identification
 - Remote systems port identification
 - How this remote system was discovered (e.g. proto)
 - Management Agent Address of Remote System (e.g. IP addr)
 - Whether there are multiple MACs or IPs out there
 - Whether or not this entry is static (not ageable)
 - Time this entry was last verified
- Critical information to communicate in a protocol includes:
 - Remote Node Address
 - Remote Node Name
 - Remote Node Sending Port
- Additional MIB functionality includes polling reduction timestamp, statistics, configuration and trap configuration.

LLDP TLVs



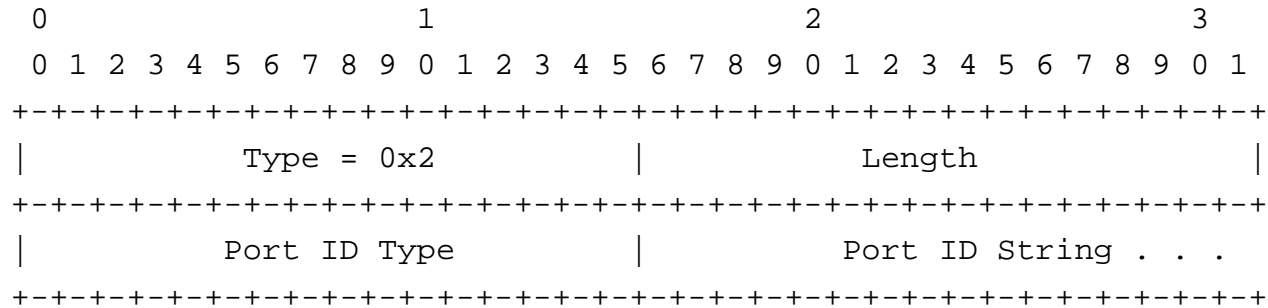
- TLVs must fit completely within a single 802 frame
 - TLVs are 32-bit aligned
 - Type = ChassisId, PortId, MgtAddress, Vendor Specific, ???
 - Length = length of value field in octets (not including pads)
 - Value = binary list of octets + pads for alignment

Chassis ID TLV



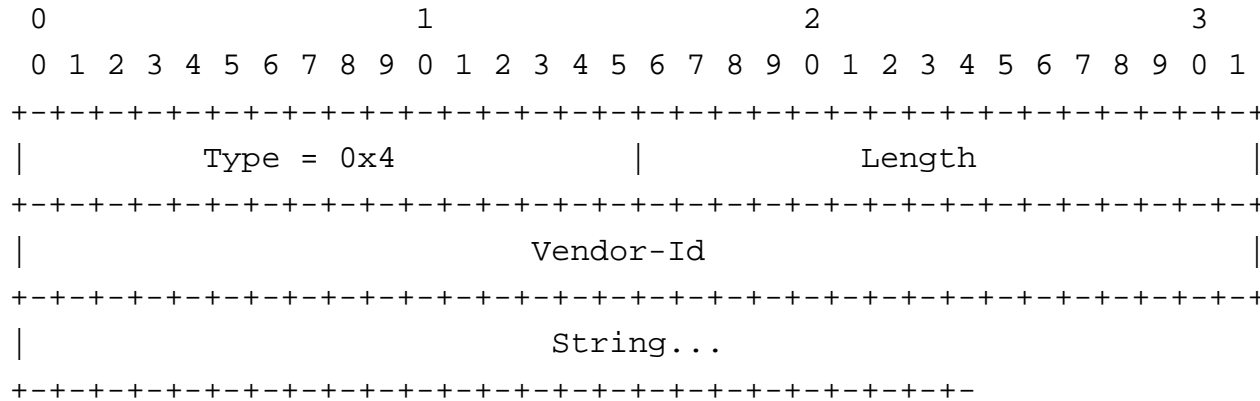
- End System Identifier to populate ptopoConnRemoteChassisType and ptopoConnRemoteChassis
 - TLV Type = 01
 - Length <= 34
 - A number of different Chassis ID types based on other MIBs
 1. entPhysicalAlias for chassis
 2. ifAlias for an interface
 3. entPhysicalAlias for port or backplane
 4. MAC address for the system
 5. A management address for the system

Port ID TLV



- Sending Port Identifier to populate ptopoConnRemotePortType and ptopoConnRemotePort
 - TLV Type = 02
 - Length <= 34
 - A number of different Port ID types based on other MIBs
 1. ifAlias for the source port
 2. entPhysicalAlias for the port
 3. MAC address for the port
 4. A management address for the port

Vendor Specific TLV



- An extension to allow vendor specific information to be transmitted
 - TLV Type = 04
 - Length > 4, but small enough to fit in PDU
 - Vendor ID is defined by the SMI Network Management Private Enterprise Code
 - Vendor specific string may include subtypes, records, etc...
 - Multiple of these TLVs may exist in a single PDU

Some Significant Variables

AdminStatus	A global enable/disable for the protocol
SuppressEntry	A per-port enable/disable
MessageTxInterval	Time interval on-which to transmit LLDP messages
MessageTxHoldMultiplier	Number of time intervals the remote peer should consider information previously transmitted as valid (i.e. used to calculate TTL)

NOTE: there are others associated with PTOPO MIB Agent

LLDP Frame Transmission

- Slow Protocols Transmission Rules
 - No more than 5 frames-per-second
 - No more the 10 slow protocols in operation
 - Attempt to keep the frame small
 - Untagged format only
- Transmit a single LLDP PDU per transmission interval with small amount of intentional jitter
- Set $TTL = \min(65535, (\text{MessageTxInterval} * \text{MessageTxHoldMultiplier}))$
- Include mandatory TLVs: Chassis ID, Management Addr, Port ID (optional for repeaters?)
- Keep transmission statistics

LLDP Frame Reception

- Subject to the Reception rules for Slow Protocols
 - Discard frames with illegal subtypes
 - Pass frames with LLDP subtype to LLDP
 - Pass frames with other supported subtype to those entities
 - Pass frames with unsupported subtypes to the the MAC Client
- Validate Message Headers and increment counters
- Validate TLVs, increment counters, skip unknown TLVs
- Inform clients of protocol that information has been received.

NOTE: Protocol itself is really not responsible for determining if information received is ‘new’ or not. PTOPO MIB holds the connectionTable.

PTOPO MIB Update

- Locate or create ptopoConnEntry
- Update ptopoConnLastVerifyTime for entry
- If new entry, update ptopoConnTabInserts
- Look for multiple MAC and/or IP addresses and update ptopoConnMultiMACSASeen and/or ptopoConnMultiNetSASeen
- If anything other than ptopoConnLastVerifyTime is modified, then update ptopoLastChangeTime

NOTE: Much of this seems to be part of the PTOPO MIB agent, rather than the LLDP Agent. Determine appropriate API.

Interface Shutdown

- If LLDP or the port is administratively disabled, an attempt to inform the peer is made by transmitting a final LLDP message with TTL=0.
- Upon reception of a LLDP message with TTL=0, remove all associated information from PTOPO MIB.

Issues and Questions

- The LLDP MIB is probably not right...
- Is the PTOPO MIB the right thing to populate?
- Where should the interface to the protocol be drawn (above the decoding of new information or below)?
- Does the clean-up shutdown procedure really work? Is it worth it?
- Are there additional TLVs that make sense?
 - sysName, PVID, HW serial numbers, layer-n info, others...
- More efficient PDU field packing?
- Should we use Slow Protocols? (potential issue with repeaters?)
- How to handle LLDP messages larger than a single frame?
- Non-802.3 Slow Protocol Frame encoding
- Any need to be concerned about security?
- Others...