LISP MIB

draft-schudel-lisp-mib-00

Beijing IETF - LISP WG
Gregg Schudel, Amit Jain, Victor Moreno
November 2010
Problem Statement

• Define the LISP MIB
  - One MIB covering all LISP Devices
    • xTR, MS, MR, PxTR
    • ALT routers use BGP and routing table MIB for EID-prefix routes

• Requirements
  - Track LISP device “configuration”
    • LISP features and attributes configured on the device
  - Track current LISP Table “values” such as:
    • Map-Cache and Mapping-Database entries
    • LISP site registration entries
  - Track current operational “statistics” such as:
    • Packet encapsulation and decapsulation
LISP MIB Structure

- LISP MIB is composed of ten tables

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lisp</td>
<td>Information representing the various lisp features that can be enabled on LISP devices</td>
</tr>
<tr>
<td>LispMappingDatabase</td>
<td>The EID-to-RLOC database that contains the EID-prefix to RLOC mappings configured on an ETR</td>
</tr>
<tr>
<td>LispMappingDatabaseLocator</td>
<td>The set of routing locators contained in the EID-to-RLOC database configured on an ETR</td>
</tr>
<tr>
<td>LispMapCache</td>
<td>The short-lived, on-demand table on an ITR that stores, tracks, and times-out and otherwise validates EID-to-RLOC mappings</td>
</tr>
<tr>
<td>LispMapCacheLocator</td>
<td>The set of locators per EID prefix contained in the map-cache table of an ITR</td>
</tr>
<tr>
<td>LispSite</td>
<td>Properties of each lisp site served by this device when configured as a Map-Server</td>
</tr>
<tr>
<td>LispSiteLocator</td>
<td>Properties of all locators per lisp site served by this device when configured as a Map-Server</td>
</tr>
<tr>
<td>LispMapServers</td>
<td>Properties of all Map-Servers this device is configured to use</td>
</tr>
<tr>
<td>LispMapResolvers</td>
<td>Properties of all Map-Resolvers this device is configured to use</td>
</tr>
<tr>
<td>lispUseProxyEtr</td>
<td>Properties of all Proxy ETRs this device is configured to use</td>
</tr>
</tbody>
</table>
lispTable OBJECT-TYPE
SYNTAX SEQUENCE OF lispEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"This table represents the various lisp features that can be enabled on lisp devices."

lispEntry ::= SEQUENCE {
lispAddressFamily AddressFamilyNumbers,
lispItrEnabled TruthValue,
lispEtrEnabled TruthValue,
lispProxyItrEnabled TruthValue,
lispProxyEtrEnabled TruthValue,
lispMapServerEnabled TruthValue,
lispMapResolverEnabled TruthValue,
lispMapCacheSize Unsigned32,
lispMapCacheLimit Unsigned32,
lispEtrMapCacheTtl Unsigned32,
lispRlocProbeEnabled TruthValue,
lispEtrAcceptMapDataEnabled TruthValue,
lispEtrAcceptMapDataVerifyEnabled TruthValue,
lispMapRequestsIn Counter64,
lispMapRequestsOut Counter64,
lispMapRepliesIn Counter64,
lispMapRepliesOut Counter64,
lispMapRegistersIn Counter64,
lispMapRegistersOut Counter64
}
LispAddressType ::= TEXTUAL-CONVENTION
STATUS current
DESCRIPTION
"LISP architecture can be applied to a wide variety of
address-families. This textual-convention is a
generalization for representing addresses that belong
to those address-families. For convenience, this
document refers to any such address as a lisp address.
LispAddressType textual-convention consists of
the following four tuples:
1. IANA Address Family Numbers: This tuple follows
the AddressFamilyNumbers textual-convention
described in [IANA]. The enumerations are listed
in [IANA]. Note that the list of address family
numbers is maintained by IANA.
2. Length of LISP address: This tuple is an INTEGER
to give the octet length of the next tuple.
3. Lisp address: A lisp address can be an address
belonging to any of the IANA Address Families.
Particularly, when the address family is Lisp
Canonical Address Format (LCAF) [LCAF] with IANA
assigned Address Family Number 16387, then
the first octet of this tuple indicates the LCAF
type, and the rest of this tuple is same as the
encoding format of the LISP Canonical Address
after the length field, as defined in [LCAF].
4. Mask-length of lisp address."
REFERENCE "[LISP]"
SYNTAX OCTET STRING (SIZE (0..1024))
MIB’s useful even for experimental protocols
Proposing lisp-mib draft as a working group document
Comments?