



ISP RLOC GW selection by EEM EID tagging

by Rees 30-4-2014

Problem statement

AT&T need to have simultaneous use of dynamic EID registration via IGP population of RTR database mapping, and EID tagging per RLOC to influence the entry LMGW to the ISP domain.

Dynamic EID requires the use of a single site with the default prefix, allow more specific limiting all registrations to a single route tag.

LMGW selection per RLOC, requires individual tags configured per EID (associated with an RLOC), which negates the option for dynamic IGP based registration.

Proposed solution

Cisco propose the use of an EEM script running on the MSMR to monitor dynamic registrations and add static mappings for each new EID prefix identified.

These EIDs will be configured using tags derived from community values received from the XTR or Partner.

Use of the EEM script for a given VPN will be controlled by a description of the site location

Each VPN will have a default site defined, with the “accept all” EID definition and a specific route tag. This will be used to register any prefix needed by the customer.



For Traffic From AGN CALA to TAM for Stage 1 of Routing Enhancement)

Set CV values for gateway selection (same as in Routing Enhancement Stage 3)

- CV1 => Gateway A, B then C
- CV2 => Gateway B, A then A
- CVx => ... Etc.

CE (xTR) advertises RLOC with CVx in LISP Transport VRF

CE (xTR) registers EID with RLOC in MSMR

Configure MSMR with default route 0.0.0.0/0 and allow more specific (with tag 999, for example)

Create EEM script to run on MSMR, periodically (frequency TBD)

- Checking for specific EID registration under the default route (or check for specific tag value such as 999)
- Once specific prefix is found, EEM script needs to use the EID prefix to find the associated RLOC in LISP mapping table on MSMR
- EEM script then uses the RLOC to find the associated CVx that is advertised with this RLOC in LISP Transport VRF
- EEM updates the MSMR configuration tag setting per CVx

MSMR passes the tag to LMGW for Local Pref setting



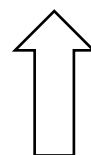


From LISP Domain to BGP in AGN

- **EEM** to retrieve CV value and set tag in MSMR
- Use Tag to define CV to send to LMGW
- LMGW set LP on AGN customer VRF

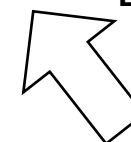


MapServer
MapResolver_{1..n}

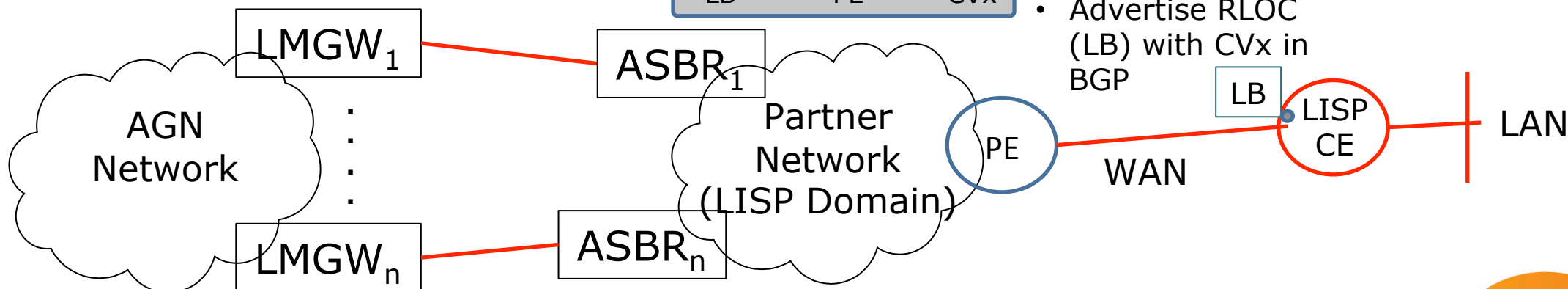


LISP Transport VRF		
<u>Dest</u>	<u>Next-Hop</u>	<u>CV</u>
LB	PE	CVx

- Register EID (LAN with RLOC (LB) in LISP



- Advertise RLOC (LB) with CVx in BGP



Workflow 1 – ssearch (lisp_get_process)

Find configured LISP process numbers

```
#show run | inc router lisp  
er lisp 1  
er lisp 2  
#
```

Scrape config for lisp processes

LISP process numbers

Record process numbers configured – we need to iterate through.

Workflow 2 – ssearch (lisp_eem_managed)

Determine if LISP process is EEM script managed

```
#show lisp 1 site name 0000
```

Show process number site name 0000

```
  name: 0000
```

```
  description: EEM-managed
```

```
  allowed configured locators: any
```

```
  allowed EID-prefixes:
```

Look for description

Look for description – if OK continue, else move to next LISP process ID

Workflow 3a – ssearch (lisp_get_transport_vrf)

For current lisp process MSMR configured for EEM script use.

```
#show lisp 1 ← LISP process ID
  peer-lisp ID:      1
  transport table:   vrf cust1-rloc ← LISP Transport (RLOC) VRF
  instance count:    2
#
```

Record the LISP Transport VRF

Workflow 3b (If we need to define a new site. TBD) - search (lisp_get_AuthKey_EIDvrf_InstanceID)

For current lisp process MSMR configured for EEM script use.

```
lisp 1
  transport-table vrf cust1-rloc
  vpn-table vrf cust1-eid instance-id 101
  ipv4 route-export site-registration
  ipv6 route-export site-registration
  exit
  default 0000
  description EEM-managed
  authentication-key vpn-1-key
  ip address prefix instance-id 101 0.0.0.0/0 route-tag 999 accept-more-specifics
  ip address prefix instance-id 101 ::/0 route-tag 999 accept-more-specifics
  exit
  ipv4 map-server
  ipv4 map-resolver
  ipv6 map-server
  ipv6 map-resolver
  exit
```

- LISP process ID
- LISP Transport (RLOC) VRF
- LISP Instance ID
- LISP VPN (EID) VRF
- LISP Default site
- EEM control description (must match to managed by EEM)
- Shared key for VPN
- LISP match all default statements for IPv4 and IPv6

Record the LISP Transport VRF and authentication key.

Workflow 4 – Look for registered prefixes to iterate through search (lisp_verify_registered_prefixes)

E XTR registers EID on MSMR. MSMR marks with “default” tag.

```
#sh lisp 1 site
```

← Show command for LISP process 1 prefixes

Site Registration Information

Name	Last Register	Up	Who Last Registered	Inst ID	EID Prefix
	never	no	--	101	0.0.0.0/0
	00:00:56	yes	172.16.1.2	101	172.16.100.3/32

← Default prefix (unregistered)

← Dynamic registered prefix

Default site

Registered prefix RLOC

LISP Instance ID

Record the EID Prefix, Instance ID and who registered. Ignore the defaults

Workflow 5 – Examine a prefix - ssearch (sp_examine_prefix)

```
#show lisp 1 site 172.16.100.3/32 instance-id 101
Site Registration Information
```

```
  name: 0000
  description: EEM-managed
  allowed configured locators: any
  tested EID-prefix:
  D-prefix: 172.16.100.3/32 instance-id 101
  First registered: 02:15:34
  Routing table tag: 999
  Origin: Dynamic, more specific of 0.0.0.0/0
  Merge active: No
  Proxy reply: No
  TTL: 1d00h
  State: complete
  Registration errors:
    Authentication failures: 0
    Allowed locators mismatch: 0
```

```
ETR 172.16.1.2, last registered 00:00:15, no proxy-reply, map-notify
  TTL 1d00h, no merge, hash-function sha1, nonce 0xCA9A607C-0x0E974520
  state complete, no security-capability
  xTR-ID 0xFE78594D-0x51D6BCB2-0xE0A7FBE9-0x28CFC0AB
  site-ID unspecified
```

Locator	Local	State	Pri/Wgt	Scope
172.12.12.1	yes	up	1/1	IPv4 none

```
#
```

For each prefix in this process

Verify this is registered as
dynamic with route Tag

ETR Address

Locator Address

Identify that this prefix is being learned by the
dynamic match using either the default tag or the
Dynamic origin line.

Record the ETR address (may be different from the
RLOC seen previously)

Workflow 6 – find a community value - ssearch (sp_get_community)

```
#sh ip bgp vpnv4 vrf cust1-rloc 172.12.12.1
routing table entry for 991:1:172.12.12.1/32, version 214
s: (1 available, best #1, table cust1-rloc)
advertised to update-groups:
  15
fresh Epoch 1
001 2 500 65001
10.10.1.2 (via vrf cust1-rloc) from 10.10.1.2 (10.200.200.1)
  Origin IGP, localpref 100, valid, external, best
  Community: 90
mpls labels in/out 29/nolabel
rx pathid: 0, tx pathid: 0x0
#
```

For each prefix in this process

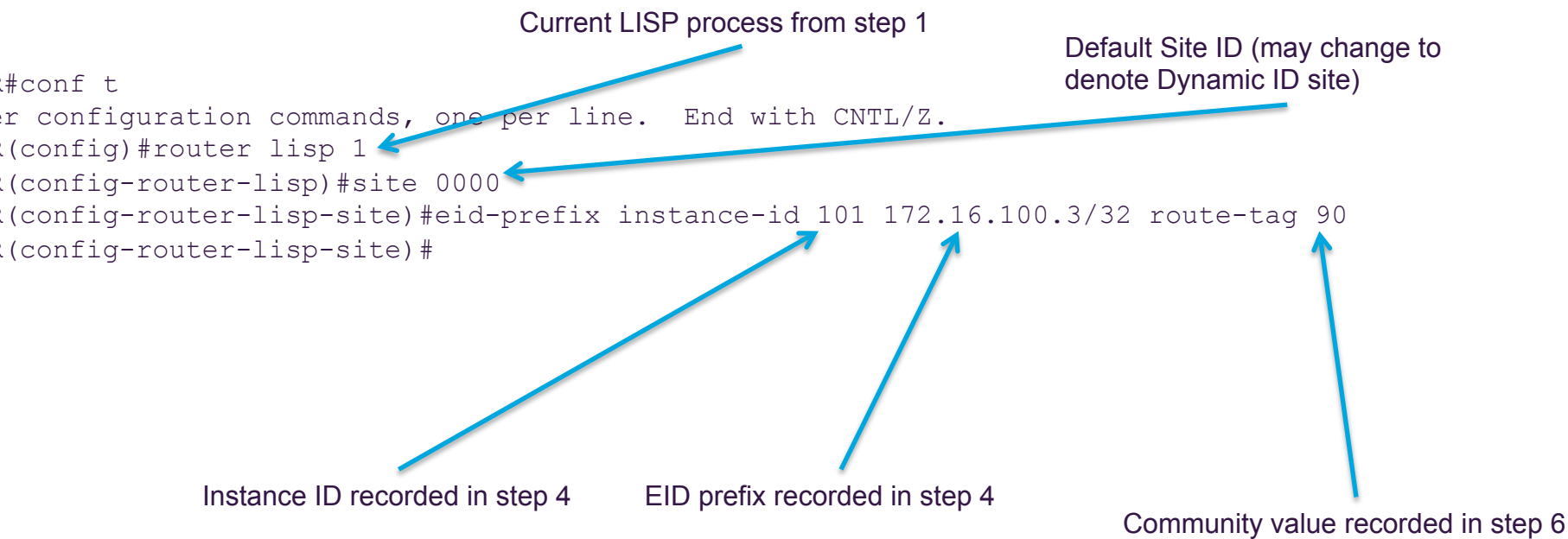
Verify this is registered as dynamic
with route Tag

If not already identified (from previous prefix) Look for
Community value associated with Locator address of
EID in Transport (RLOC) VRF defined in config.

Record the community value.(Will be one of a few
possible)

Workflow 7 – Define new EID - ssearch (ISP_DefineEID.tcl)

Using previously recorded information add new EID definition.



Possible Enhancements

Add CLI to show all LISP processes in one command. Same as show lisp 1 site.

```
show lisp * site
```

Add CLI to show tag associated with EID registration. Includes eid, IID, RLOC and tag.

```
Show lisp * site tag
```

Add CLI to show RLOC VRF associated with LISP process

```
Show lisp * site tag rloc_vrf
```

If possible it would be awesome to have all the options available at once so we could parse the whole lot in one go. So on one line the output would be

Process, IID, EID, RLOC , tag, VRF.

```
Show lisp * site tag rloc_vrf
```