IPv6 Neighbor Discovery Prefix Registration

draft-ietf-6lo-prefix-registration

Pascal Thubert

IETF 117

San Francisco

6LoWPAN ND (IPv6 Stateful Address Autoconfiguration)

RFC 6775 (original 6LoWPAN ND)

Defines ARO for registration and DAD operations for stateful AAC

RFC 8505 (Issued 11/2018)

The protocol agnostic registration for ULA/GUA for proxy ND and routing services

Analogous to a Wi-Fi association but at Layer 3: a deterministic and query-able state for all addresses

RFC 8929 (Issued 11/2020)

Federates 6lo meshes over a high-speed backbone

ND proxy analogous to Wi-Fi bridging but at Layer 3

RFC 8928 (Issued 11/2020)

Protects addresses against theft (Crypto ID in registration)

draft-ietf-6lo-multicast-registration

Extends RFC 8505 for multicast and anycast

draft-thubert-6lo-unicast-lookup

Provides a 6LBR on the backbone to speed up DAD and lookup

Coexistence with classical ND

draft-ietf-6lo-prefix-registration

Extends RFC 8505 for prefixes

Let it be for prefixes!

- Hosts may own prefixes -> and routers may connect to prefixes
  - Network in Node / recursive networking
  - Kubernetes / Private IPv4 realms
  - Directly connected (no routing)

Registering a Prefix

SGP – agnostic UNI interface between prefix owner and router

Overload Status field with PLEN in NS message

R flag to redistribute in SGP

F flag to signal source vs destination matching. Useful ?

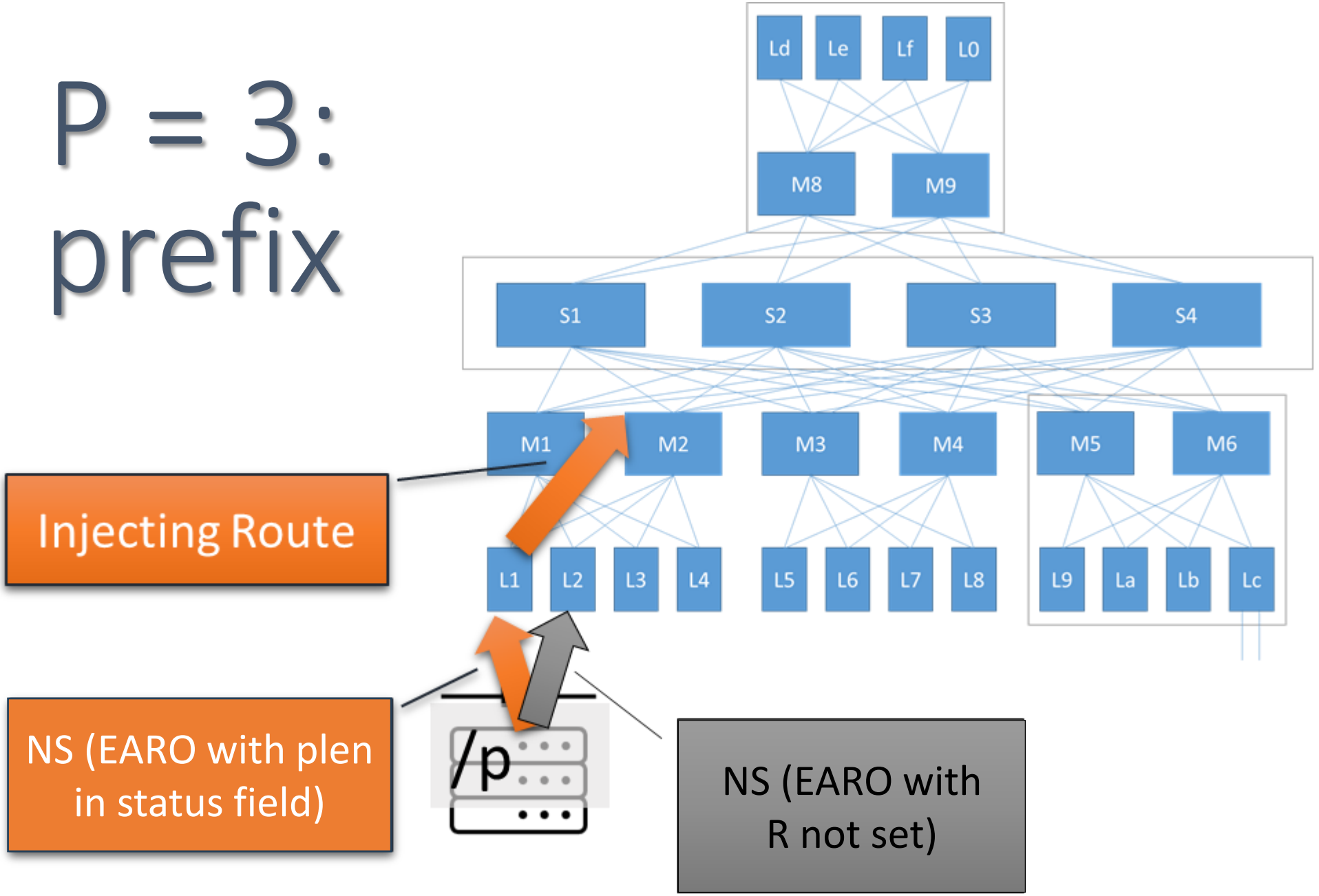
But field getting saturated

Extending the P field

- P is a 2-bits field in EARO, DAR, and RTO
- Defined the Multicast Address Registration draft

Value	Meaning	Reference
00	Registration for a Unicast Address	mcast RFC
01	Registration for a Multicast Address	mcast RFC
10	Registration for an Anycast Address	mcast RFC
~~11~~	~~Unassigned~~	~~mcast RFC~~
11	Registration for a prefix	This RFC

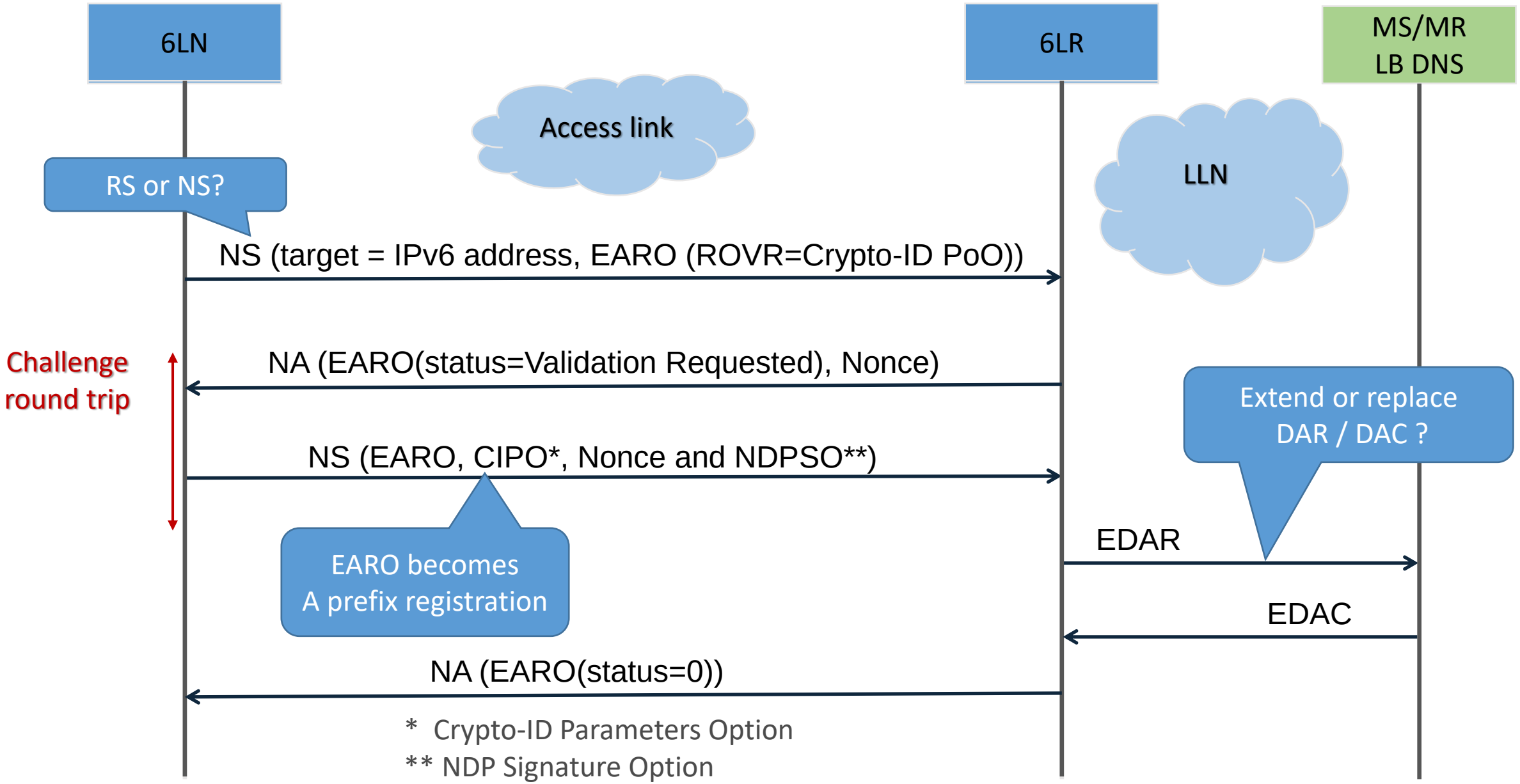
P = 3: prefix


Ld
Le
Lf
L0
M8
M9
S1
S2
S3
S4
M1
M2
M3
M4
M5
M6
L1
L2
L3
L4
L5
L6
L7
L8
L9
La
Lb
Lc
Injecting Route
NS (EARO with plen in status field)
/p
NS (EARO with R not set)

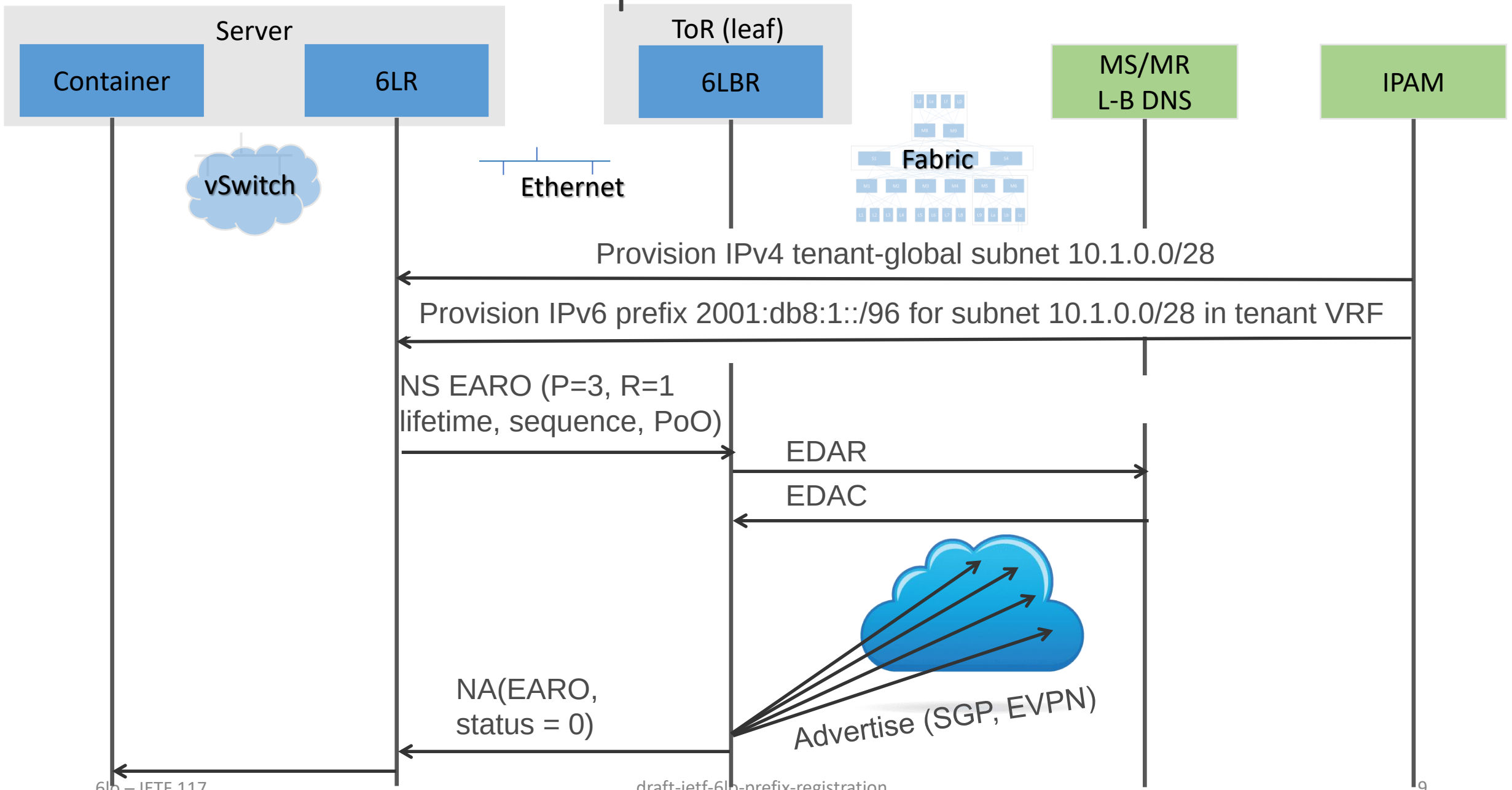
What becomes of DAD?

Need to consider prefix aggregation and nesting

- Provisioned Mobile Networks should be unique
- Auto-allocation?


6LN
6LR
MS/MR
LB DNS
Access link
LLN
RS or NS?
NS (target = IPv6 address, EARO (ROVR=Crypto-ID PoO))
Challenge
round trip
NA (EARO(status=Validation Requested), Nonce)
Extend or replace
DAR / DAC ?
NS (EARO, CIPO*, Nonce and NDPSO**)
EARO becomes
A prefix registration
EDAR
EDAC
NA (EARO(status=0))
* Crypto-ID Parameters Option
** NDP Signature Option

IPv6 address encapsulation



Could do's

- Adding stub prefix advertisement vs. host today
  - Indicate prefix type e.g., a /96 to embed an IPv4 address
  - Proof of ownership (PoO) per RFC 8928
- Adding policy / ACLs
  - Signal partial micro-segmentation (offload), who can talk to me
- Adding preference to influence load balancing
  - worker capacity (clusters / containers)
  - Access bandwidth /
  - multihoming / preferred interface / anycast
- Tenant ID / VRF ID / RPL instanceID
  - Route tags, RH

Thanks!

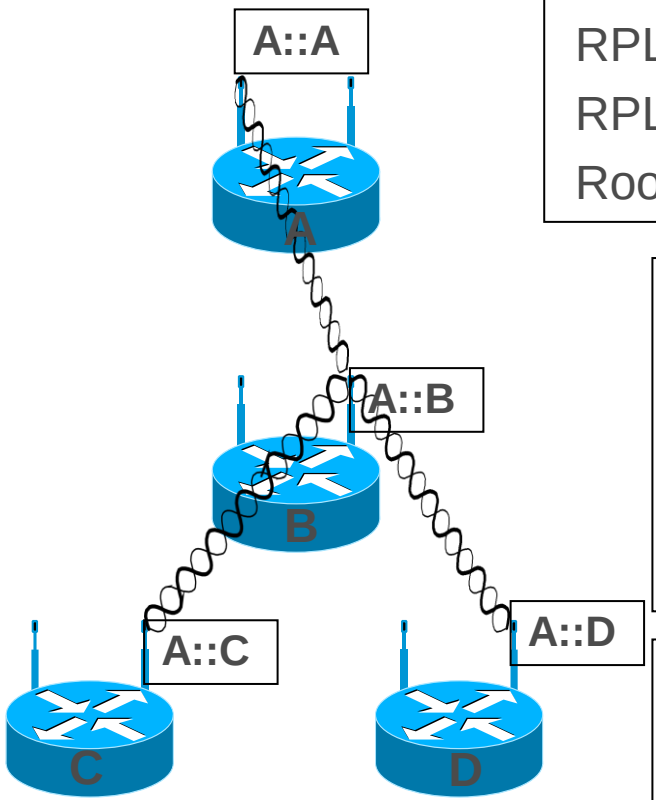
Questions?

Redistributing RFC 8505 in routing ?

- Already done for host routes with the “R” flag
  - e.g., RFC 9010 into RPL, or even RFC 8929 into IPv6 ND
  - Also draft-thubert-bess-secure-evpn-mac-signaling using BGP, or RIFT
  - Provides a host / router interface that is agnostic to the IGP beyond the router

Multi-link Subnet Routing (non-storing mode)

Parent is default GW, propagates root PIO (L-bit off)
Parent Address in the PIO (with R bit)
RPL Router autoconfigures Address from parent PIO
RPL Router advertises Address via Parent to Root
Root recursively builds a Routing Header back



C:
::/0 via **A::B**
A::B connected
A::C self
A:: ~onlink

D:
::/0 via **A::B**
A::B connected
A::D self
A:: ~onlink

Target A::C via Transit A::B

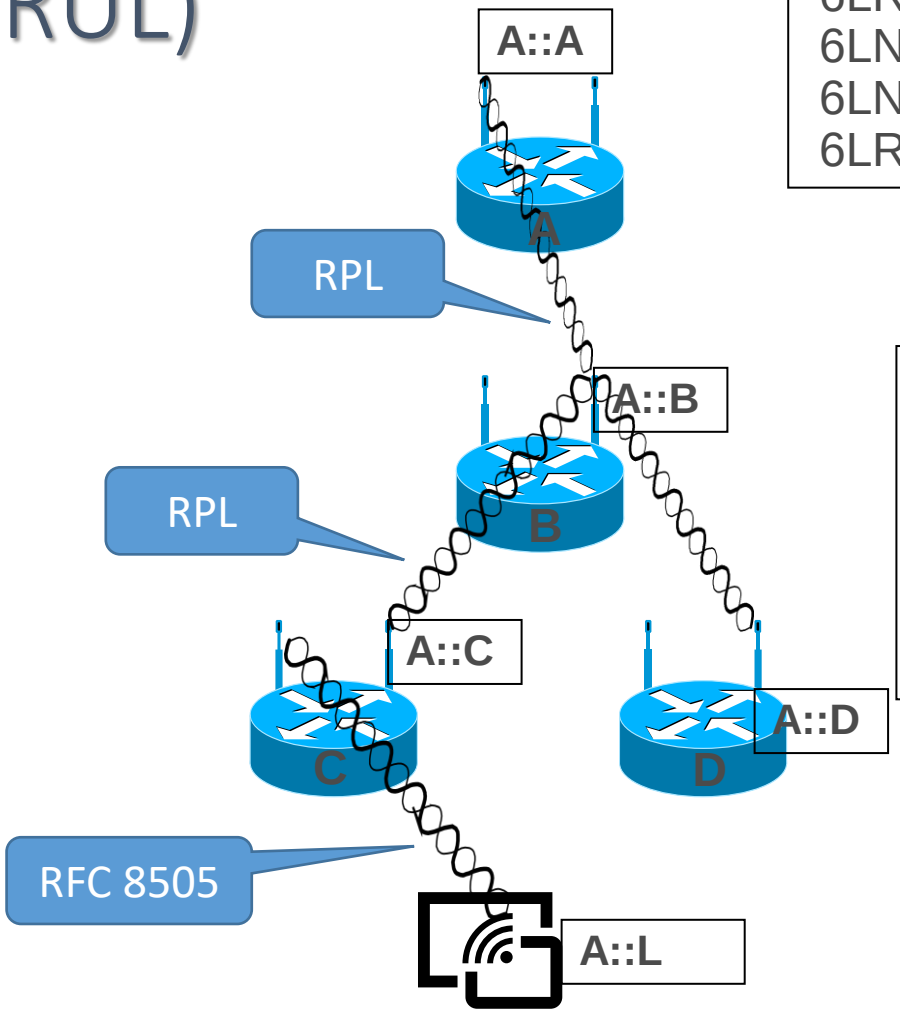
B:
::/0 via **A::A**
A::A connected
A::B self
A:: ~onlink

A: (root)
A::A self
A::B connected
A::C via **A::B**
A::D via **A::B**
A:: ~onlink

A::C via **A::B** connected

RFC 9010 (RUL)

6LR advertises A:: in RAs
6LN autoconfigures A::L
6LN registers A::L with « R » flag set
6LR injects the address as external host route in RPL



C:
::/0 via A::B
A::B connected
A::C self
A:: ~onlink

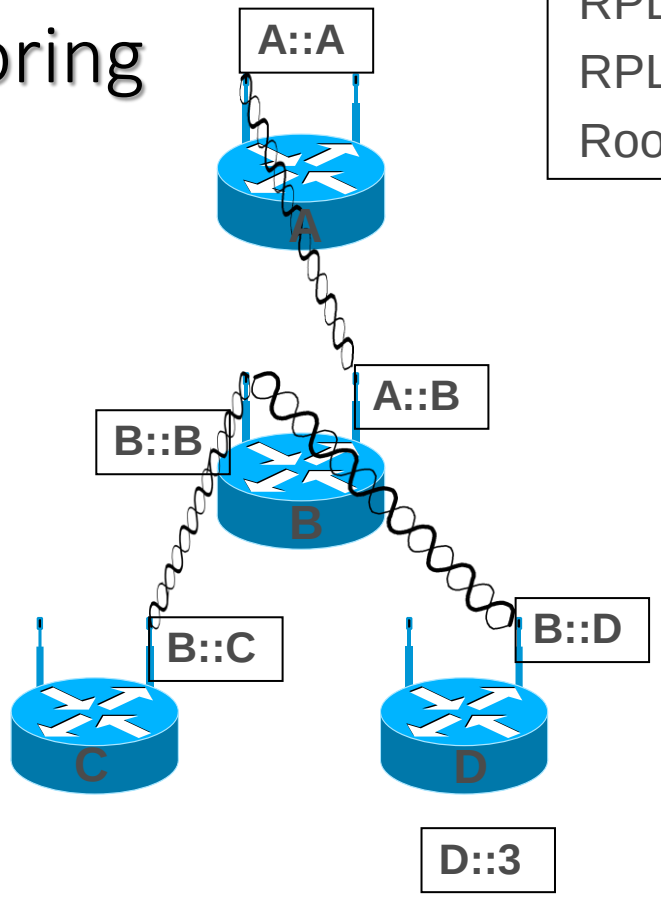
Target A::L via
Transit A::C (Ext)

A: (root)
A::A self
A::B connected
A::C via A::B
A::L via A::C
A::D via A::B
A:: ~onlink

A::L via A::C via A::B connected

Owned prefix routing (non-storing mode)

Parent is default GW, advertizes owned PIO (L bit on)
RPL Router autoconfigures Address from parent PIO
RPL Router advertises Prefix via Address to Root
Root recursively builds a Routing Header back



C:
::/0 via B::B
B:: connected
C:: connected

Target C::/ via
Transit B::C

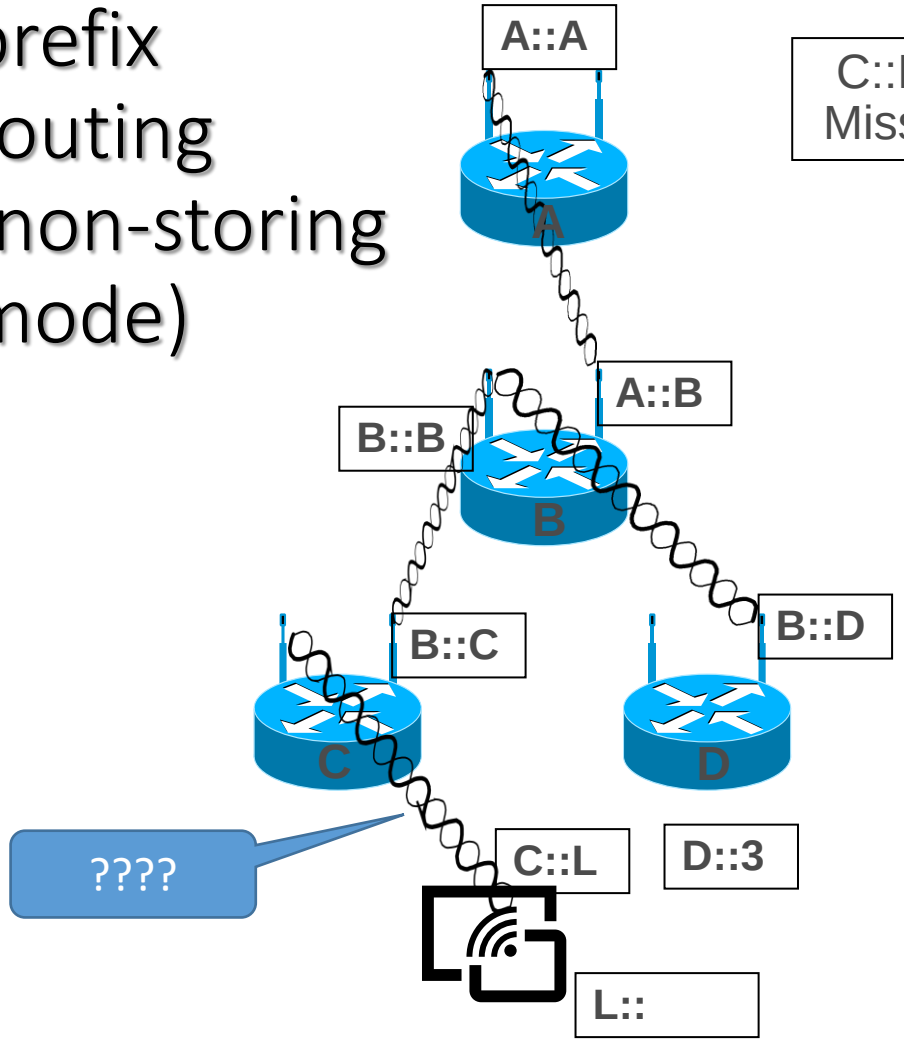
B:
::/0 via A::A
A:: connected
B:: connected

A: (root)
A:: connected
B:: via A::B
C:: via B::C
D:: via B::D

D:
::/0 via B::B
B:: connected
D:: connected

D::3 via B::D via A::B connected

Owned prefix routing (non-storing mode)



C::L is reachable but L:: is not
Missing equivalent of RFC 8505/9010 for prefixes

C:
::/0 via B::B
B:: connected
C:: connected

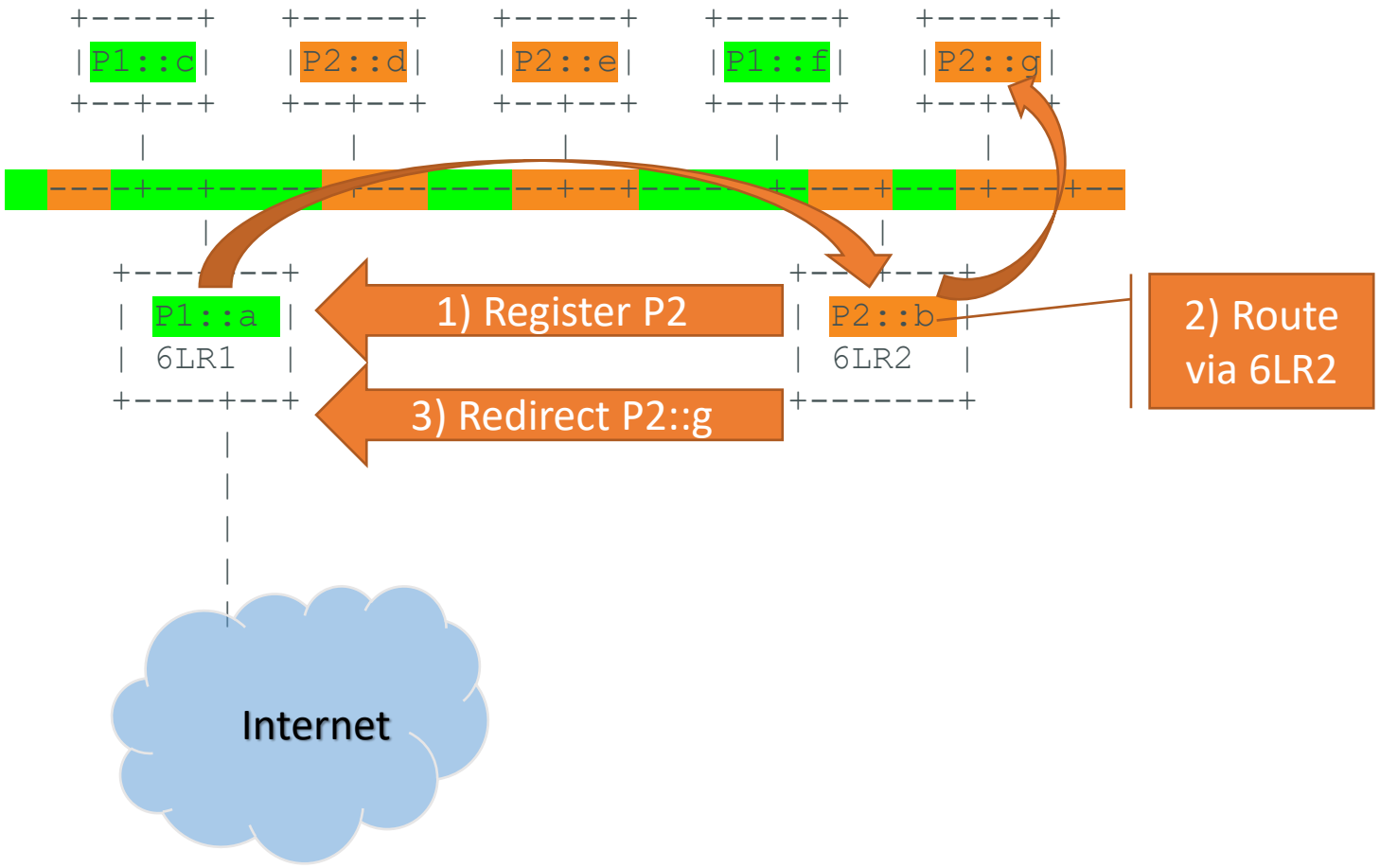
Target C::/ via
Transit B::C

A: (root)
A:: connected
B:: via A::B
C:: via B::C
D:: via B::D

L:: unreachable

C::L via B::C via A::B connected

Non LLN (SNAC) Use case 1: Shared Link



Non LLN (SNAC) Use case 2: Hub and Stubs