SRv6

Clarence Filsfils

Cisco Fellow - cf@cisco.com

Segment Routing

- Source Routing
  - the topological and service (NFV) path is encoded in the packet header
- Scalability
  - the network fabric does not hold any per-flow state for TE or NFV
- Simplicity
  - automation: TILFA
  - protocol elimination: LDP, RSVP-TE, NSH...
- End-to-End
  - DC, Metro, WAN

Industry at large backs up SR

Strong customer adoption
WEB, SP, Enterprise

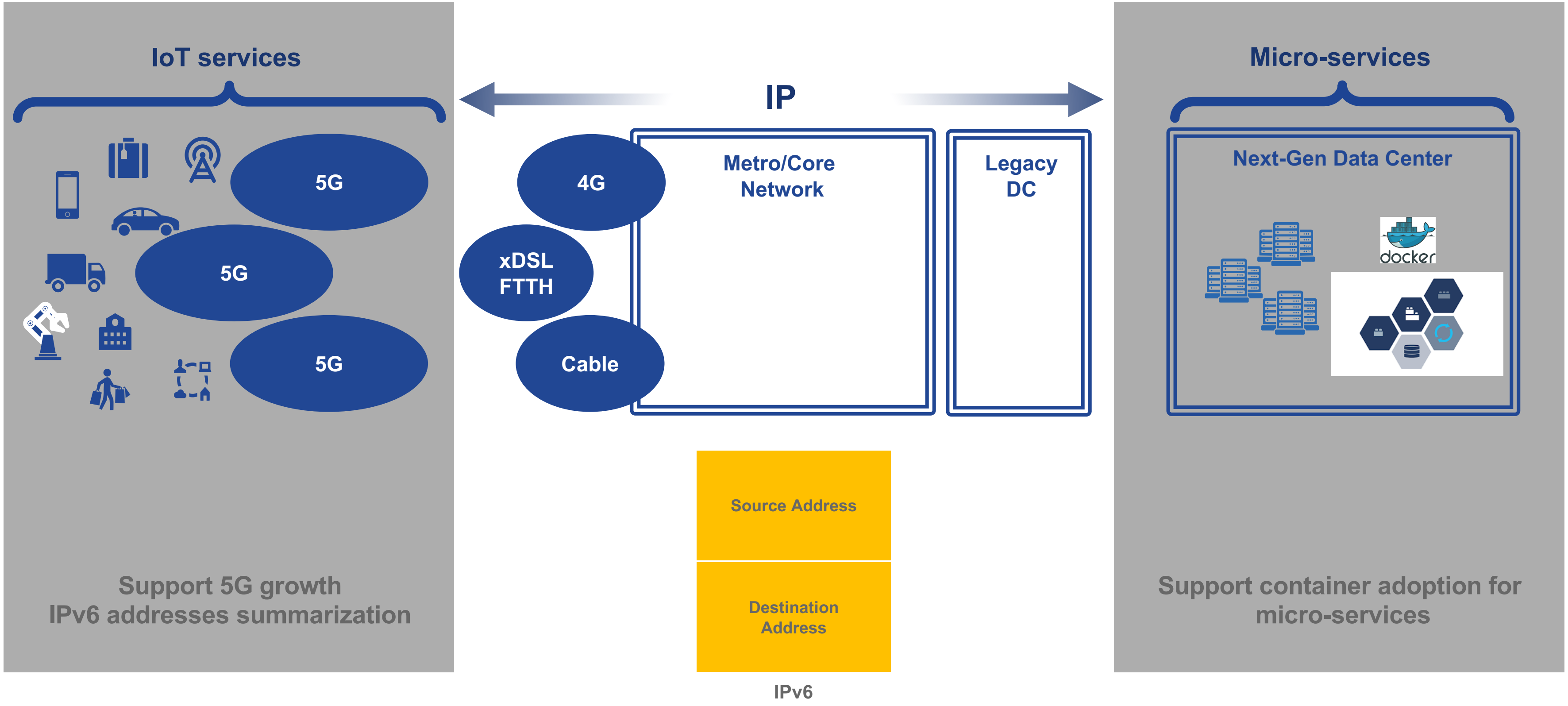
Standardization
IETF

De-Facto SDN Architecture

Multi-vendor Consensus
Interop testings

Objective of SRv6

IPv6 provides reachability



SRv6 for underlay

RSVP for FRR/TE → Horrendous states scaling in k*N^2

IPv6 for reach

SRv6 for underlay

SRv6 for Underlay → Simplification, FRR, TE, SDN

IPv6 for reach

Opportunity for further simplification

NSH for NFV	→	Additional Protocol and State
UDP+VxLAN Overlay	→	Additional Protocol just for tenant ID
SRv6 for Underlay	→	Simplification, FRR, TE, SDN
IPv6 for reach		

- Multiplicity of protocols and states hinder network economics

Network Programming

Network instruction

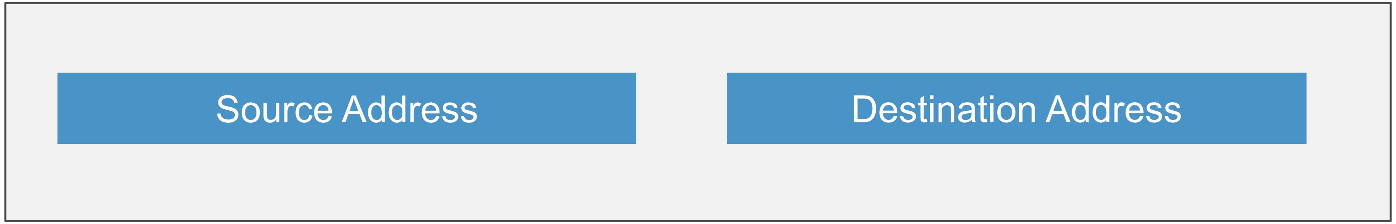
Locator	Function(arg)

- 128-bit SRv6 SID
  - Locator: routed to the node performing the function
  - Function: any possible function (optional argument)

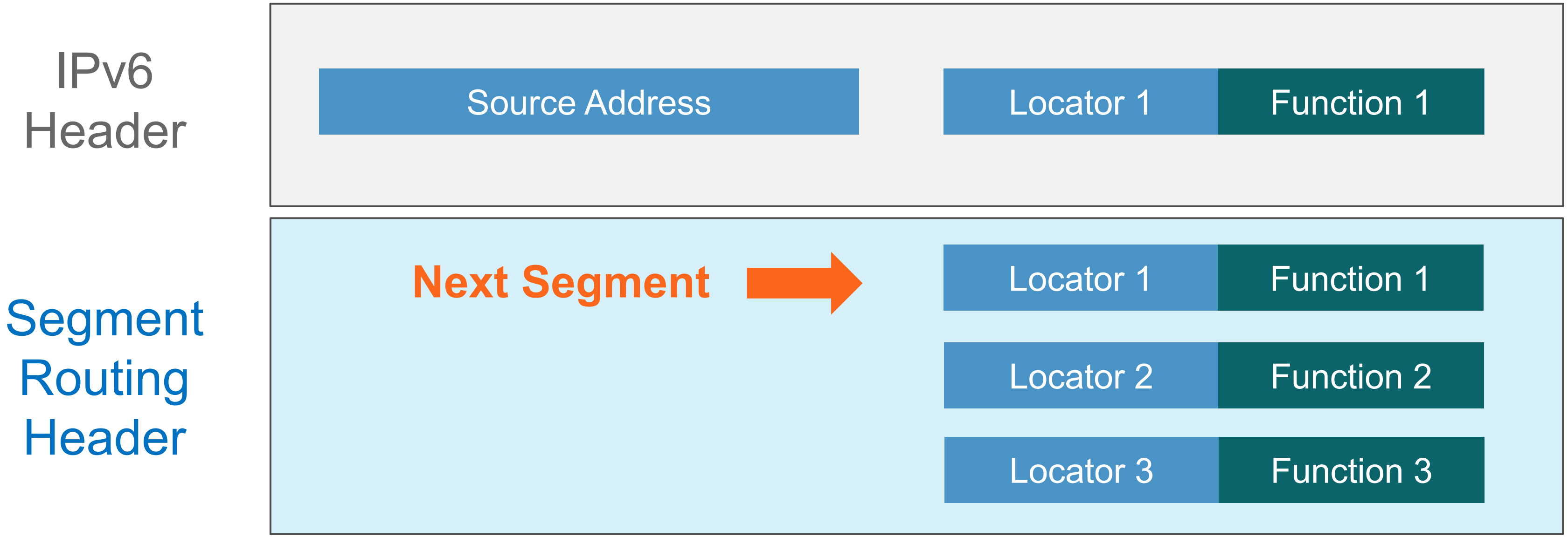
    either local to NPU or app in VM/Container
  - Flexible bit-length selection

Network Program in the packet header

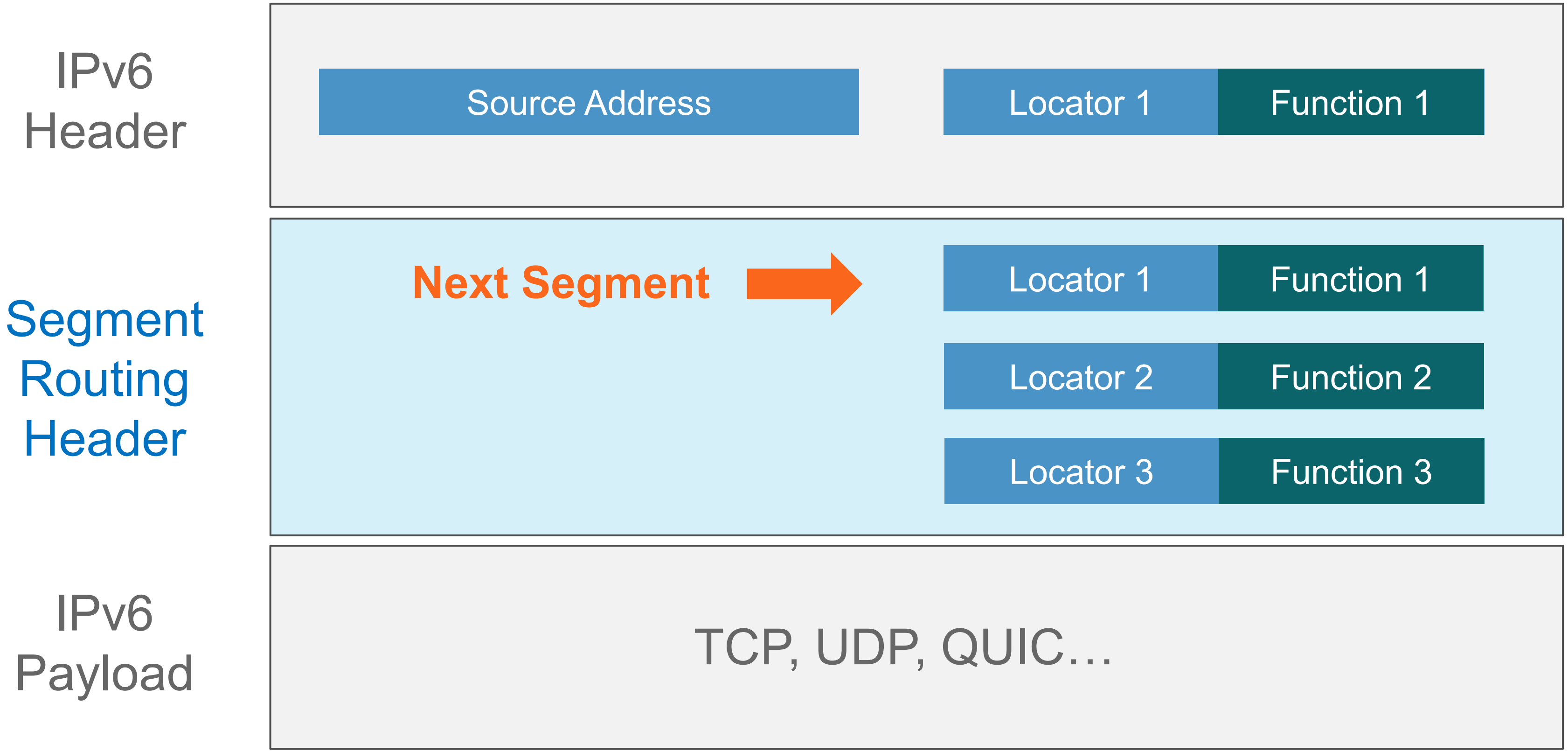
IPv6
Header



Network Program in the packet header



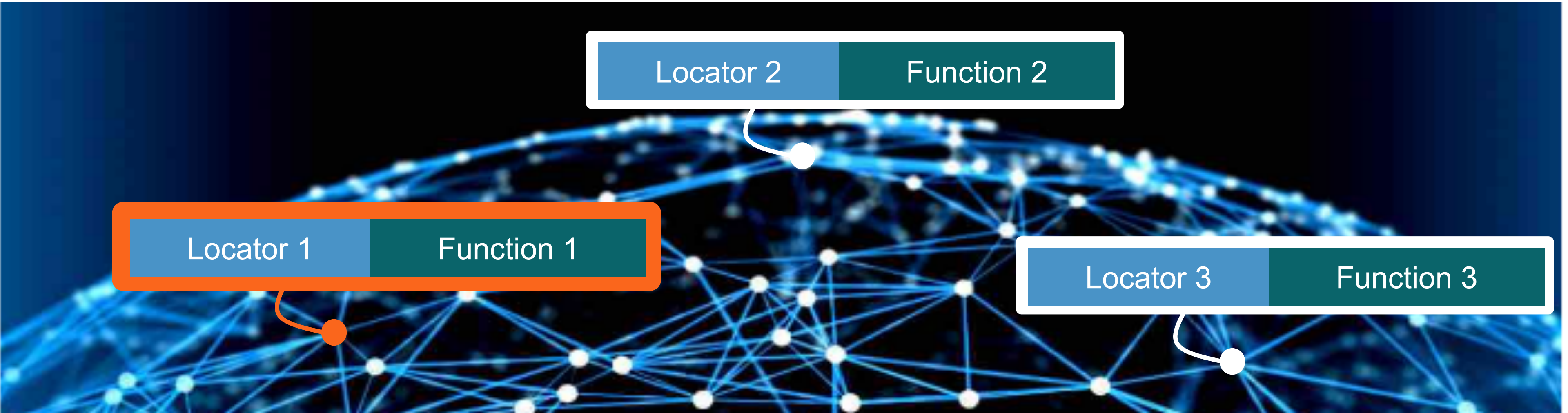
Network Program in the packet header



Network Program

Next Segment

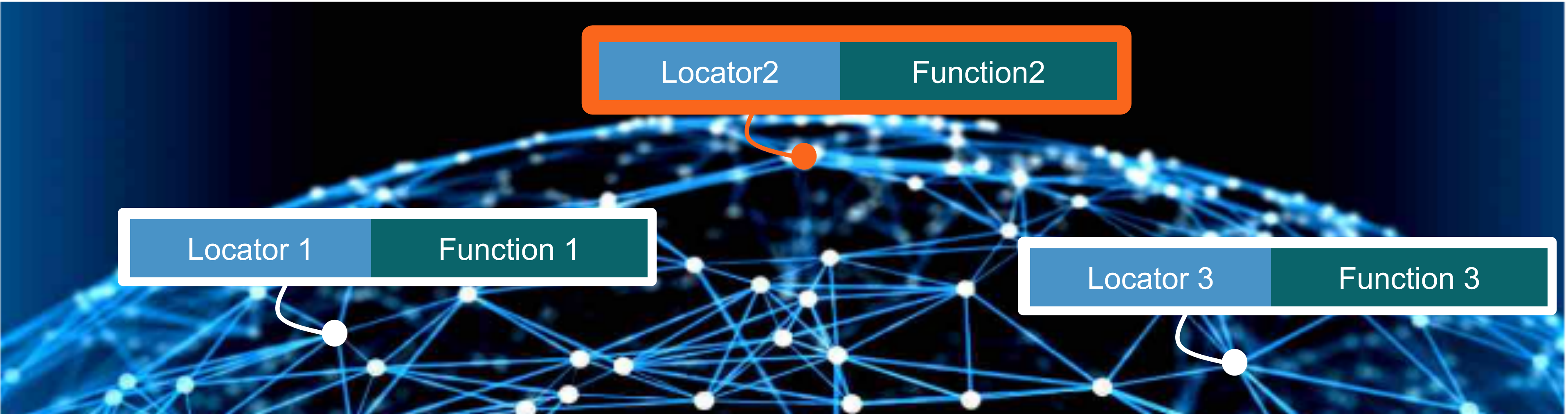
Locator 1	Function 1
Locator 2	Function 2
Locator 3	Function 3



Network Program

Next Segment →

Locator 1	Function 1
Locator 2	Function 2
Locator 3	Function 3

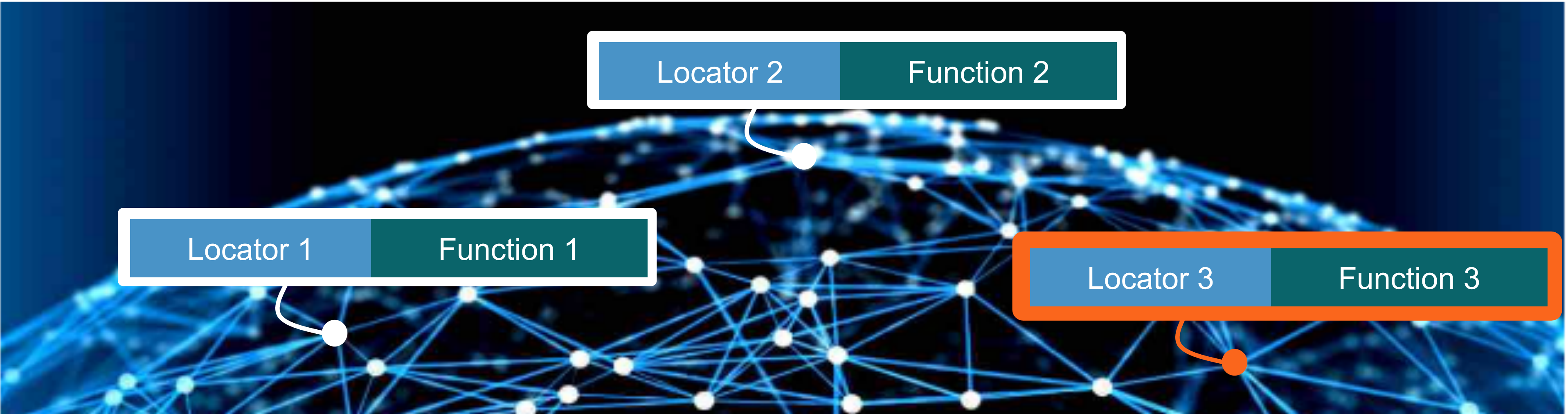


Network Program

Locator 1 | Function 1

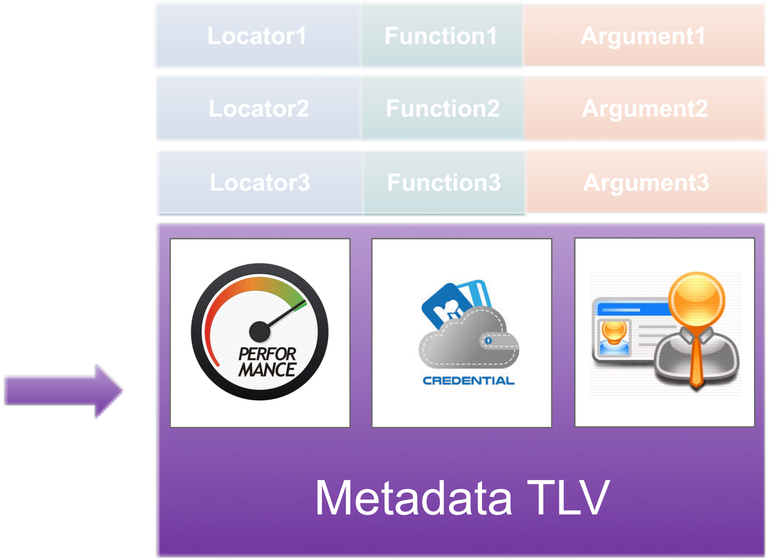
Locator 2 | Function 2

Next Segment → Locator 3 | Function 3

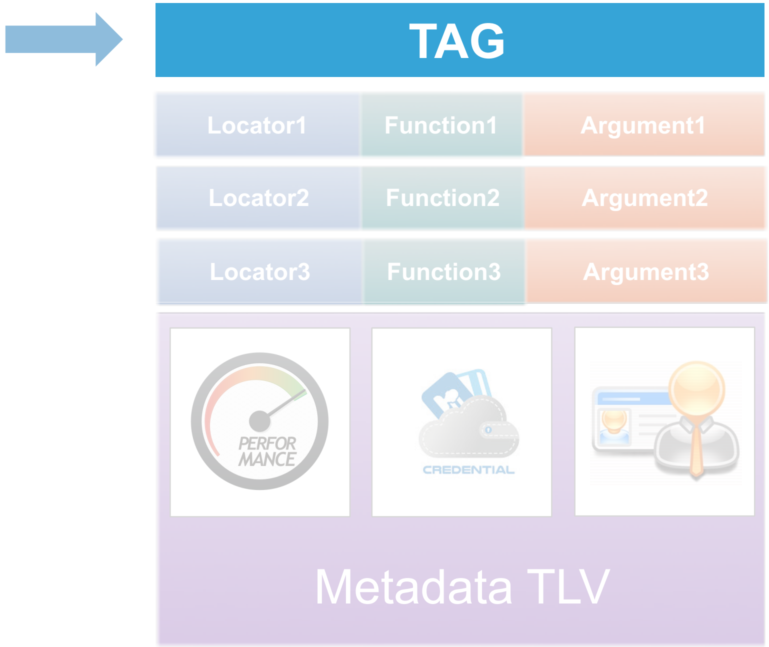


Argument shared between functions

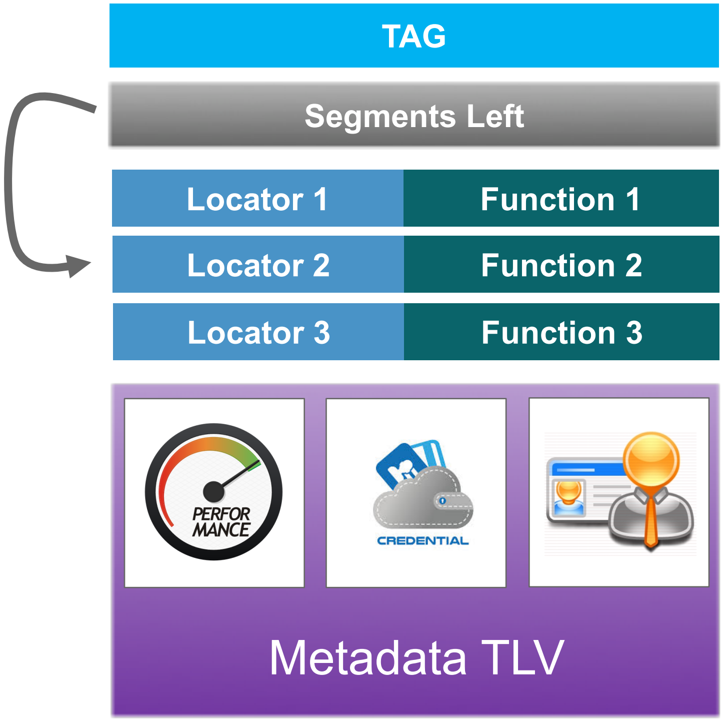
“Global” Argument



Group-Based Policy

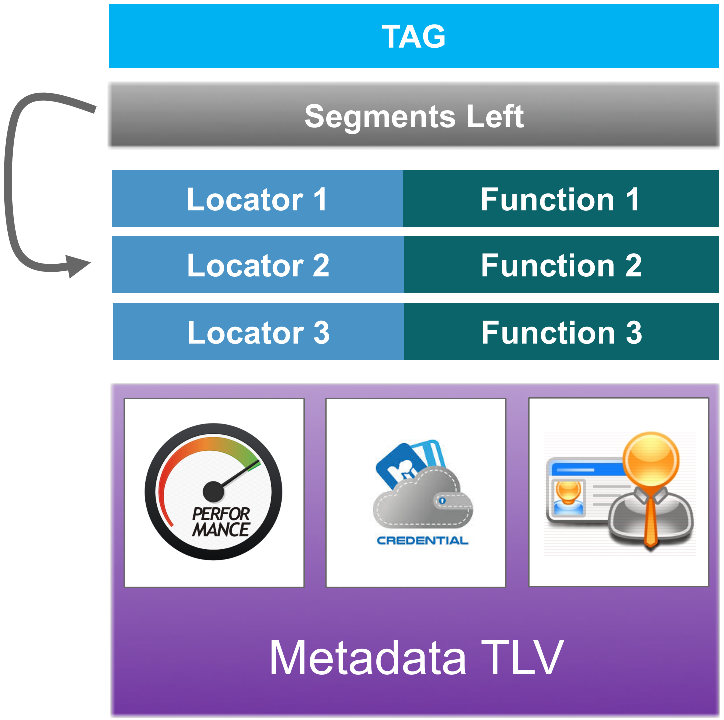


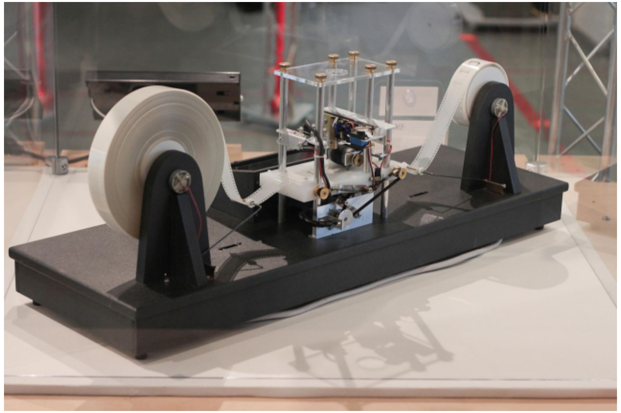
SR Header




```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Next Header   |  Hdr Ext Len  | Routing Type  | Segments Left |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Last Entry   |     Flags     |              Tag              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|            Segment List[0] (128 bits IPv6 address)            |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                                                               |
                              ...
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|            Segment List[n] (128 bits IPv6 address)            |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                                                             //
//         Optional Type Length Value objects (variable)       //
//                                                             //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

SRv6 for anything





Turing

SRv6 for anything

TAG

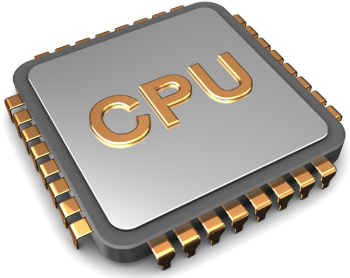
Segments Left

Locator 1	Function 1
Locator 2	Function 2
Locator 3	Function 3

Metadata TLV

Optimized for HW processing
e.g. Underlay & Tenant use-cases

Optimized for SW processing
e.g. NFV, Container, Micro-Service



Lead Operators

- Standardization
- Multi-Vendor Consensus

```
SPRING                                                              C. Filsfils
Internet-Draft                                                Cisco Systems, Inc.
Intended status: Standards Track                                      J. Leddy
Expires: September 10, 2017                                            Comcast
                                                                      D. Voyer
                                                                    D. Bernier
                                                                   Bell Canada
                                                                  D. Steinberg
                                                           Steinberg Consulting
                                                                     R. Raszuk
                                                                  Bloomberg LP
                                                                 S. Matsushima
                                                              SoftBank Telecom
                                                                     D. Lebrun
                                               Universite catholique de Louvain
                                                                    B. Decraene
                                                                        Orange
                                                                     B. Peirens
                                                                       Proximus
                                                                     S. Salsano
                                              Universita di Roma "Tor Vergata"
                                                                       G. Naik
                                                              Drexel University
                                                                     H. Elmalky
                                                                       Ericsson
                                                                P. Jonnalagadda
                                                                      M. Sharif
                                                              Barefoot Networks
                                                                    A. Ayyangar
                                                                         Arista
                                                                       S. Mynam
                                                            Dell Force10 Networks
                                                                    A. Bashandy
                                                                        K. Raza
                                                                       D. Dukes
                                                                        F. Clad
                                                              P. Camarillo, Ed.
                                                             Cisco Systems, Inc.
                                                                  March 9, 2017

                         SRv6 Network Programming
            draft-filsfils-spring-srv6-network-programming-00
```

Inter-Domain Routing
Internet-Draft
Intended status: Standards Track
Expires: September 13, 2017

G. Dawra, Ed.
C. Filsfils
D. Dukes
P. Brissette
P. Camarilo
Cisco Systems
J. Leddy
Comcast
D. Voyer
D. Bernier
Bell Canada
D. Steinberg
Steinberg Consulting
R. Raszuk
Bloomberg LP
B. Decraene
Orange
S. Matsushima
SoftBank Telecom Japan
March 12, 2017

BGP Signaling of IPv6-Segment-Routing-based VPN Networks
draft-dawra-bgp-srv6-vpn-00.txt

Network Working Group
Internet Draft
Intended status: Standard Track
Expires: September 2017

A. Bashandy, Ed.
C. Filsfils
L. Ginsberg
Cisco Systems
Bruno Decraene
Orange
March 10, 2017

IS-IS Extensions to Support Segment Routing over IPv6 Dataplane
draft-bashandy-isis-srv6-extensions-00

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 14, 2017

S. Previdi, Ed.
C. Filsfils
K. Raza
D. Dukes
Cisco Systems, Inc.
J. Leddy
B. Field
Comcast
D. Voyer
D. Bernier
Bell Canada
S. Matsushima
Softbank
I. Leung
Rogers Communications
J. Linkova
Google
E. Aries
Facebook
T. Kosugi
NTT
E. Vyncke
Cisco Systems, Inc.
D. Lebrun
Universite Catholique de Louvain
D. Steinberg
Steinberg Consulting
R. Raszuk
Bloomberg
March 13, 2017

IPv6 Segment Routing Header (SRH)
draft-ietf-6man-segment-routing-header-06

SRv6 for Next-generation Mobile

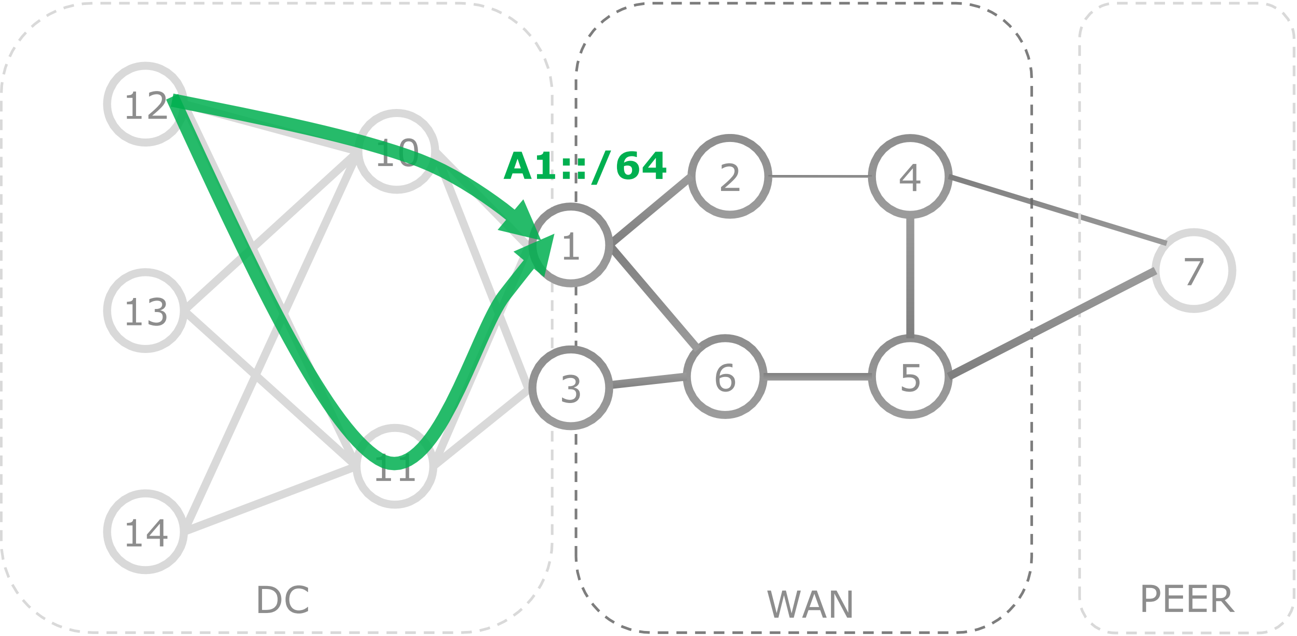
```
SPRING and DMM                                              S. Matsushima
Internet-Draft                                                   SoftBank
Intended status: Standards Track                              C. Filsfils
Expires: January 18, 2018                             Cisco Systems, Inc.
                                                            July 17, 2017


                         SRv6 for Mobile User-Plane
               draft-matsushima-spring-dmm-srv6-mobile-uplane-01
```

Use-Cases

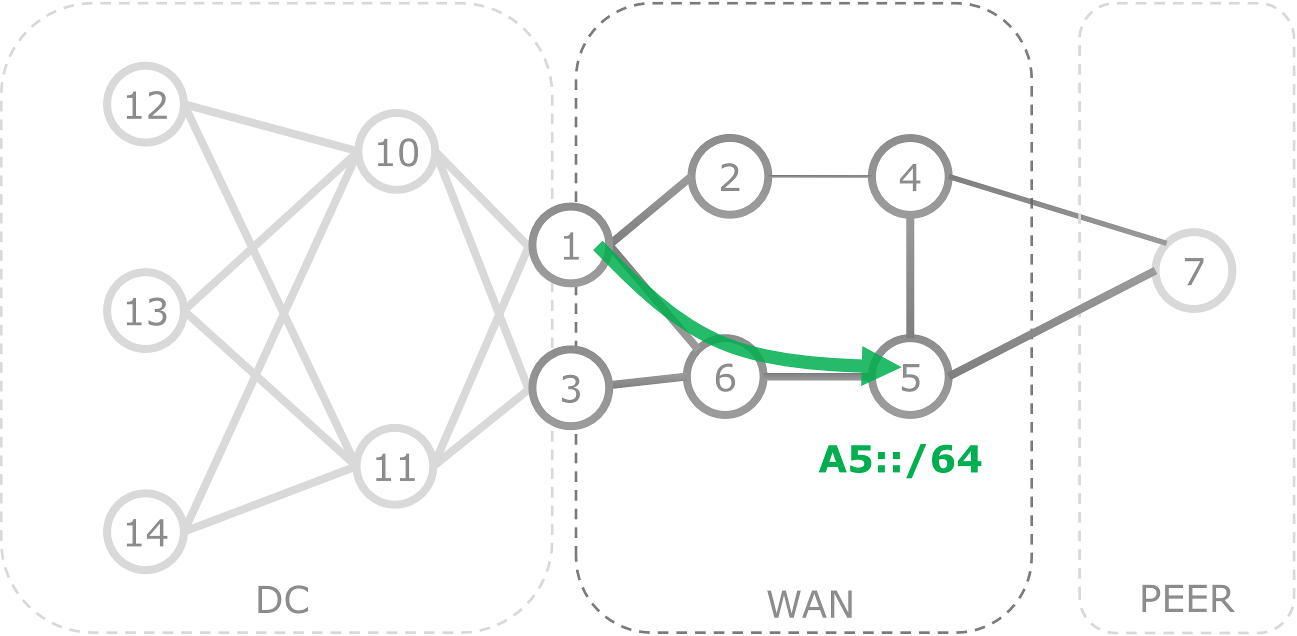
SID allocation for illustration purpose

- For simplicity
- Node K advertises prefix AK::/64
- The function is encoded in the last 64 bits



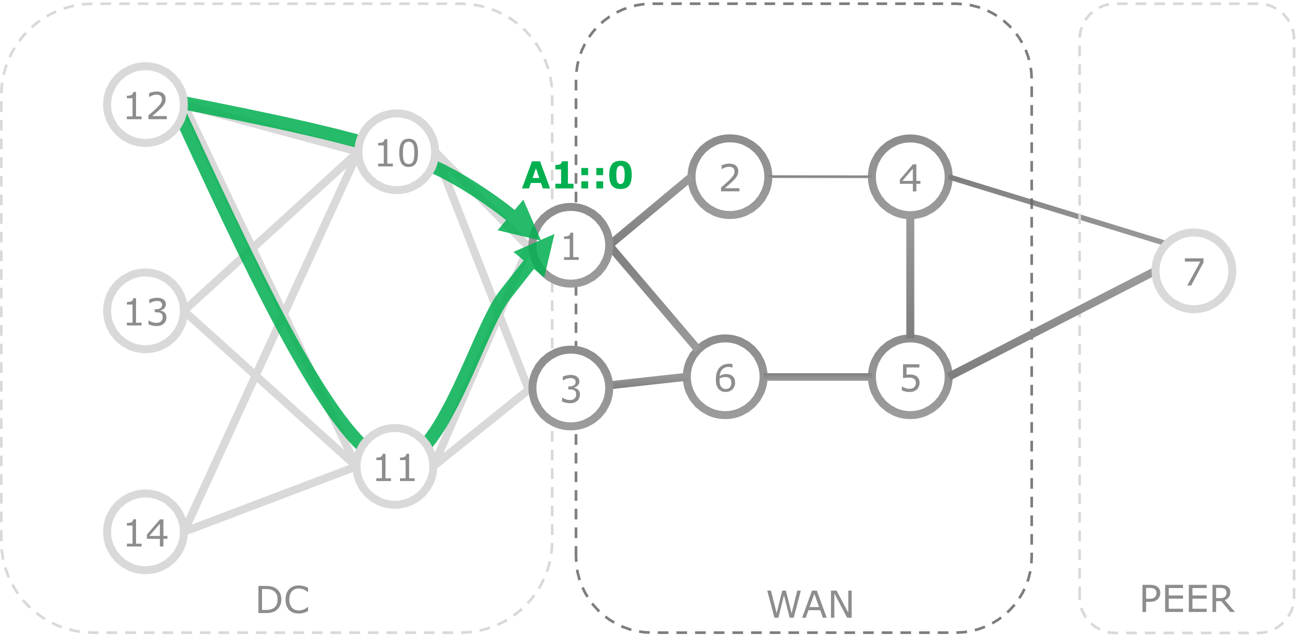
SID allocation for illustration purpose

- For simplicity
- Node K advertises prefix AK::/64
- The function is encoded in the last 64 bits



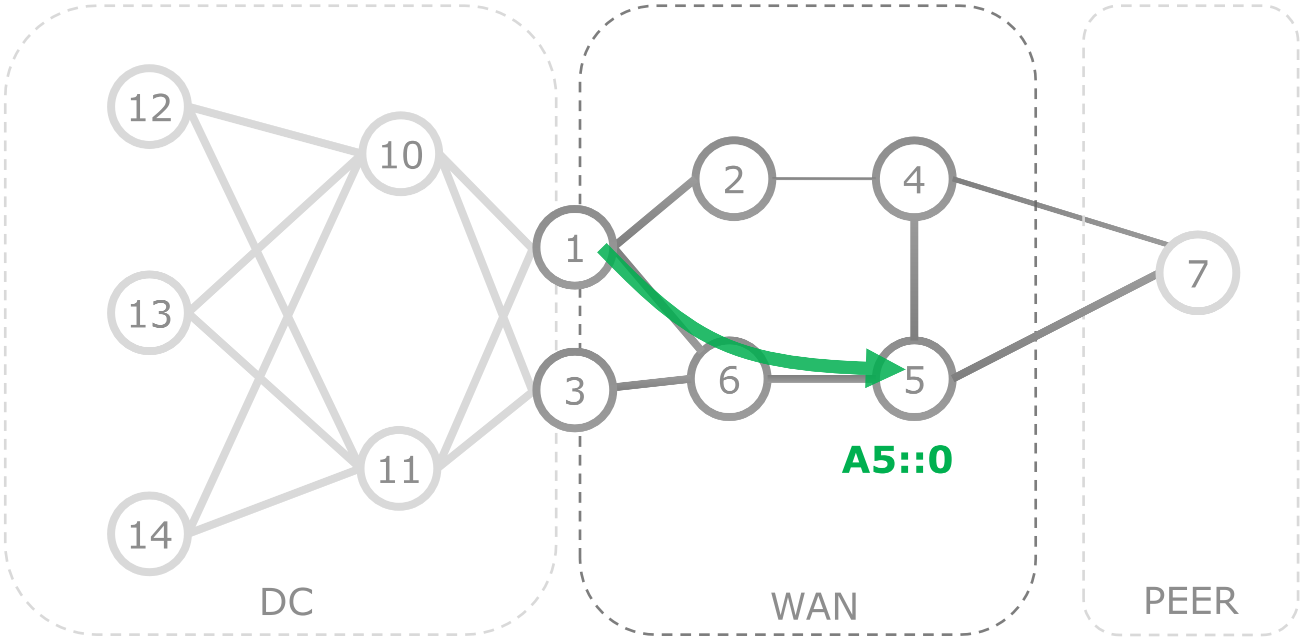
Endpoint

- For simplicity
- Function 0 denotes the most basic function
- Shortest-path to the Node

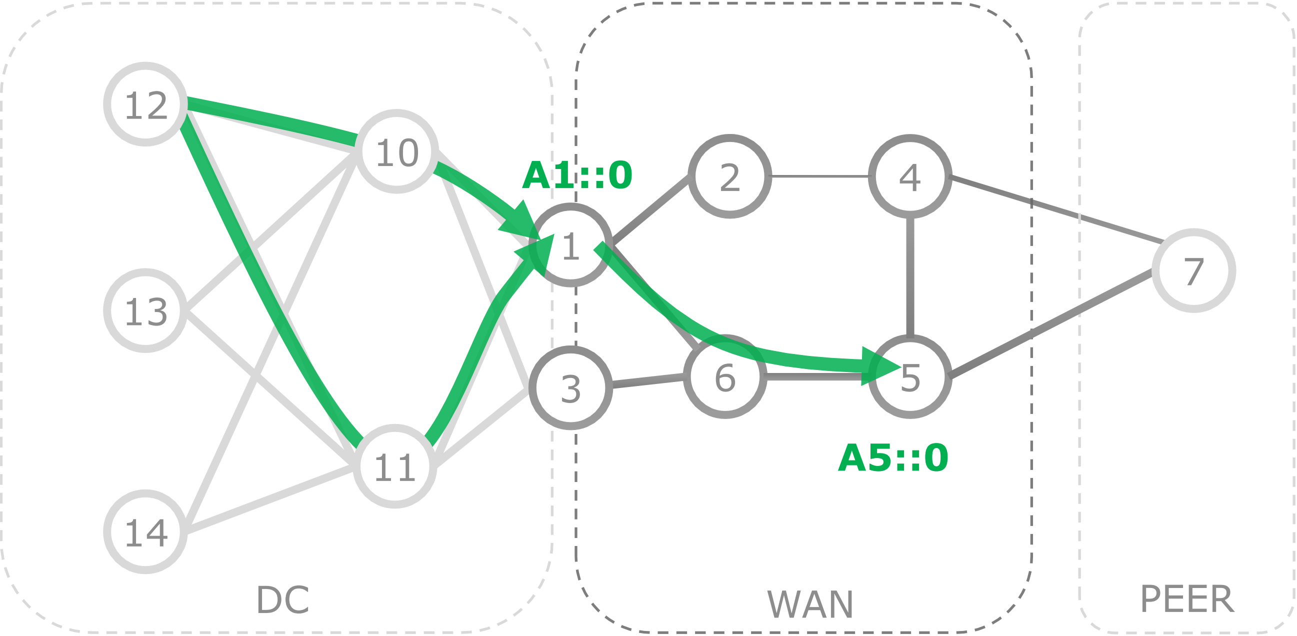


Endpoint

- For simplicity
- Function 0 denotes the most basic function
- Shortest-path to the Node



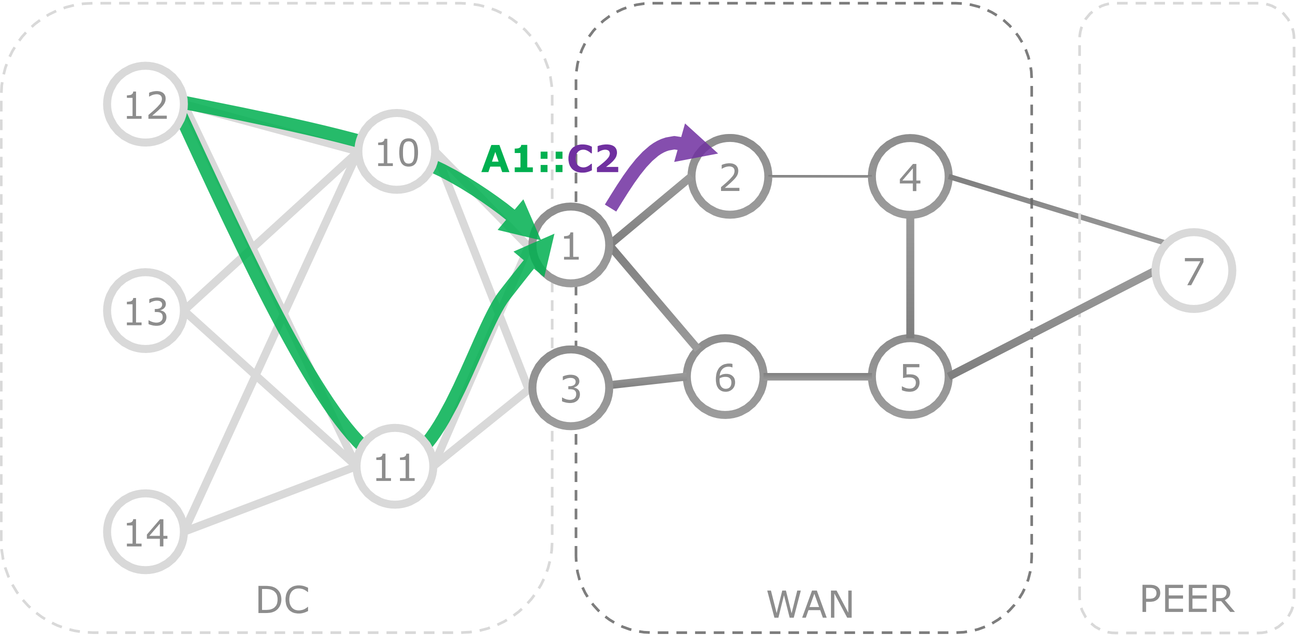
A1::0 and then A5::0



Endpoint then xconnect to neighbor

- For simplicity
- AK::CJ denotes

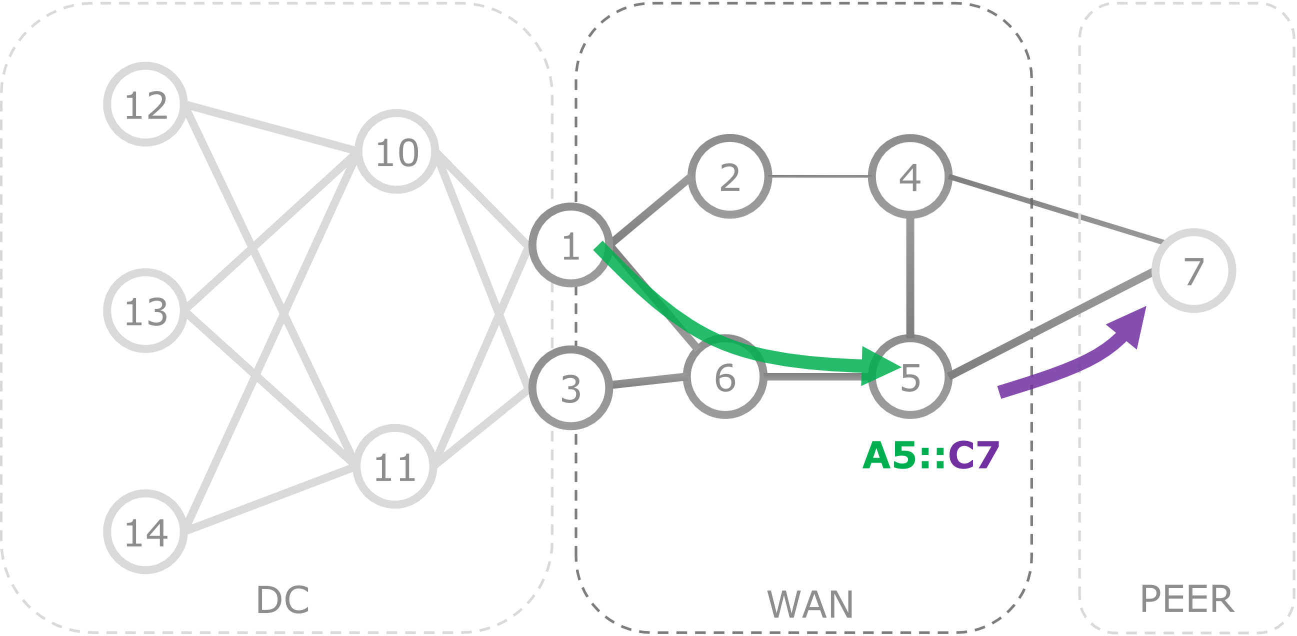
Shortest-path to the Node K and then x-connect (function C) to the neighbor J



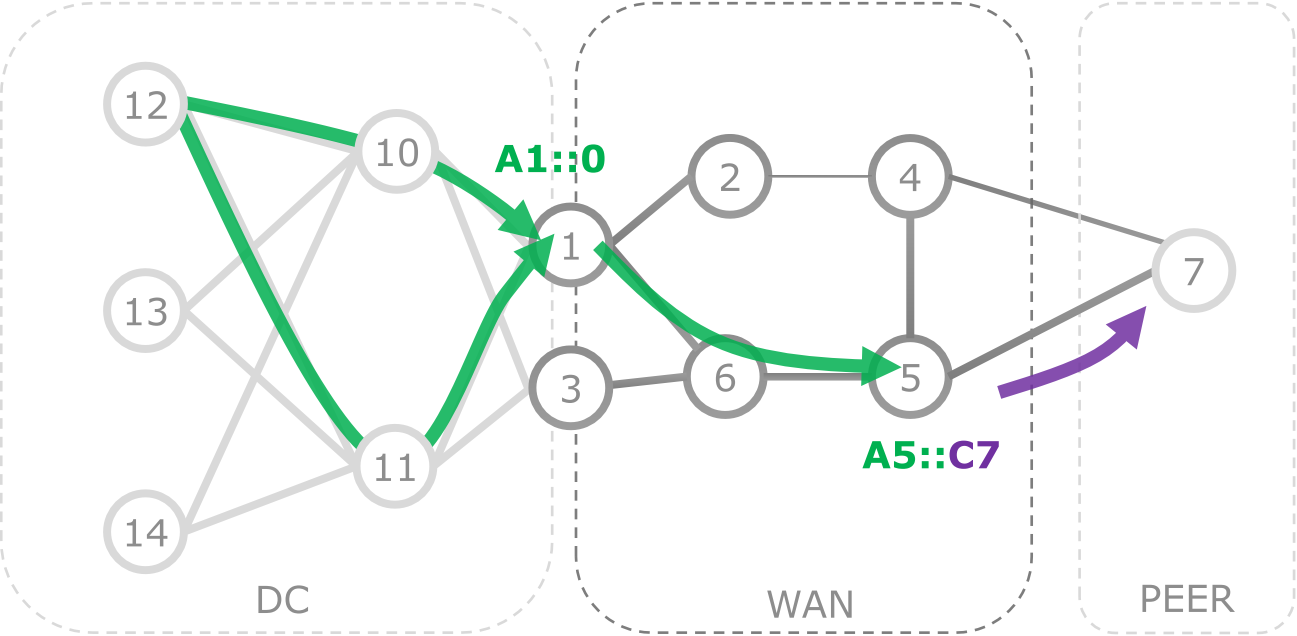
Endpoint then xconnect to neighbor

- For simplicity
- AK::CJ denotes

Shortest-path to the Node K and then x-connect (function C) to the neighbor J

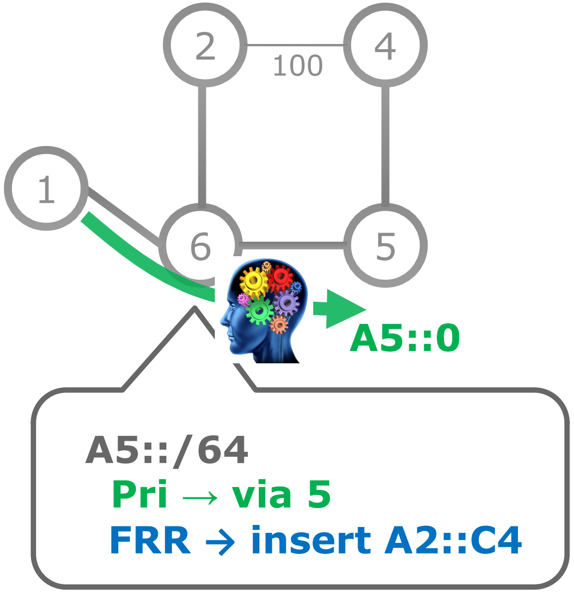


A1::0 and then A5::C7

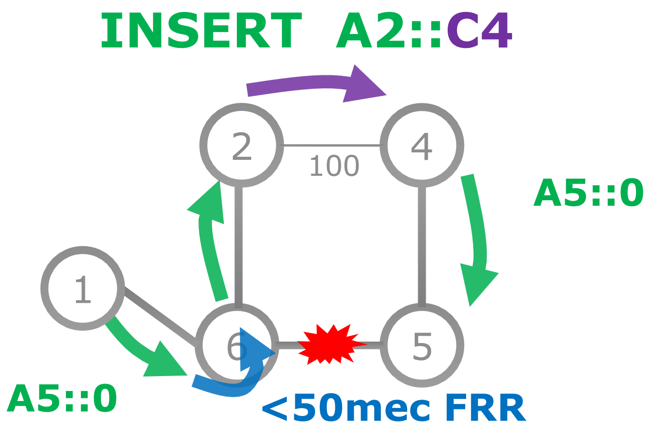


TILFA

- 50msec Protection upon local link, node or SRLG failure
- Simple to operate and understand
  - automatically computed by the router's IGP process
  - 100% coverage across any topology
  - predictable (backup = postconvergence)
- Optimum backup path
  - leverages the post-convergence path, planned to carry the traffic
  - avoid any intermediate flap via alternate path
- Incremental deployment
- Distributed and Automated Intelligence

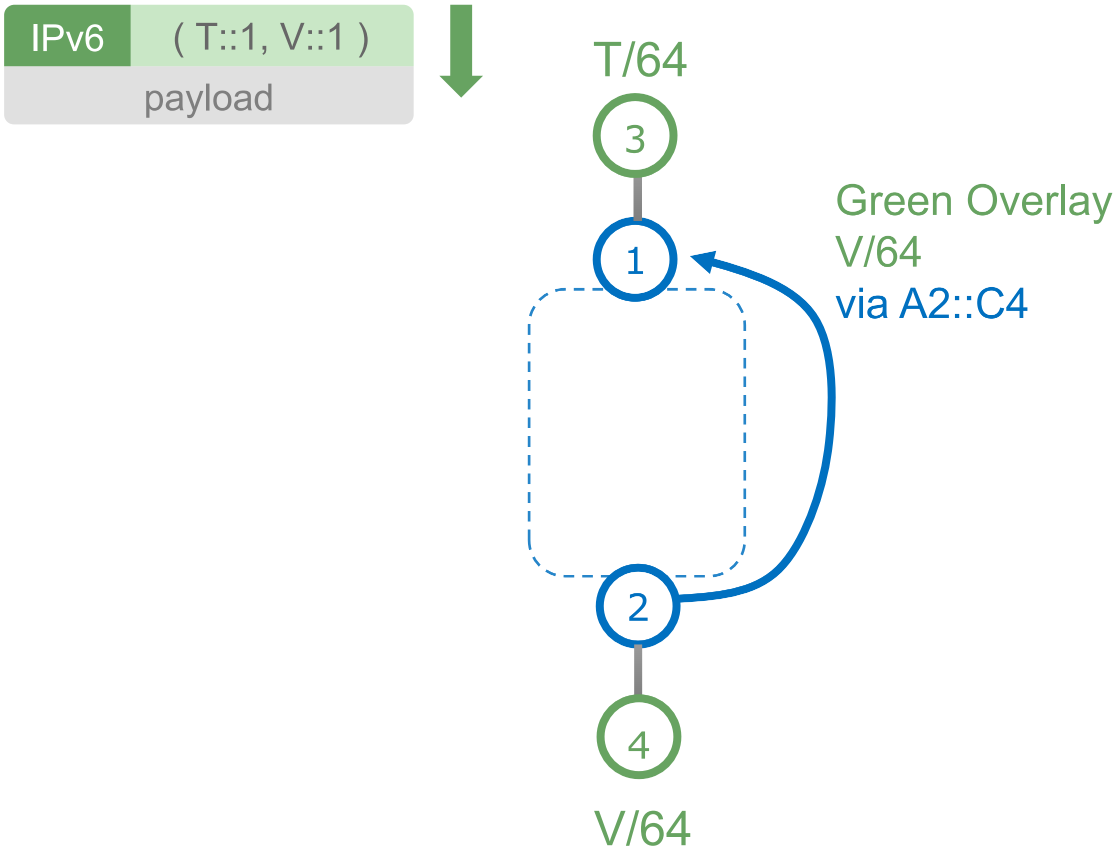


TILFA

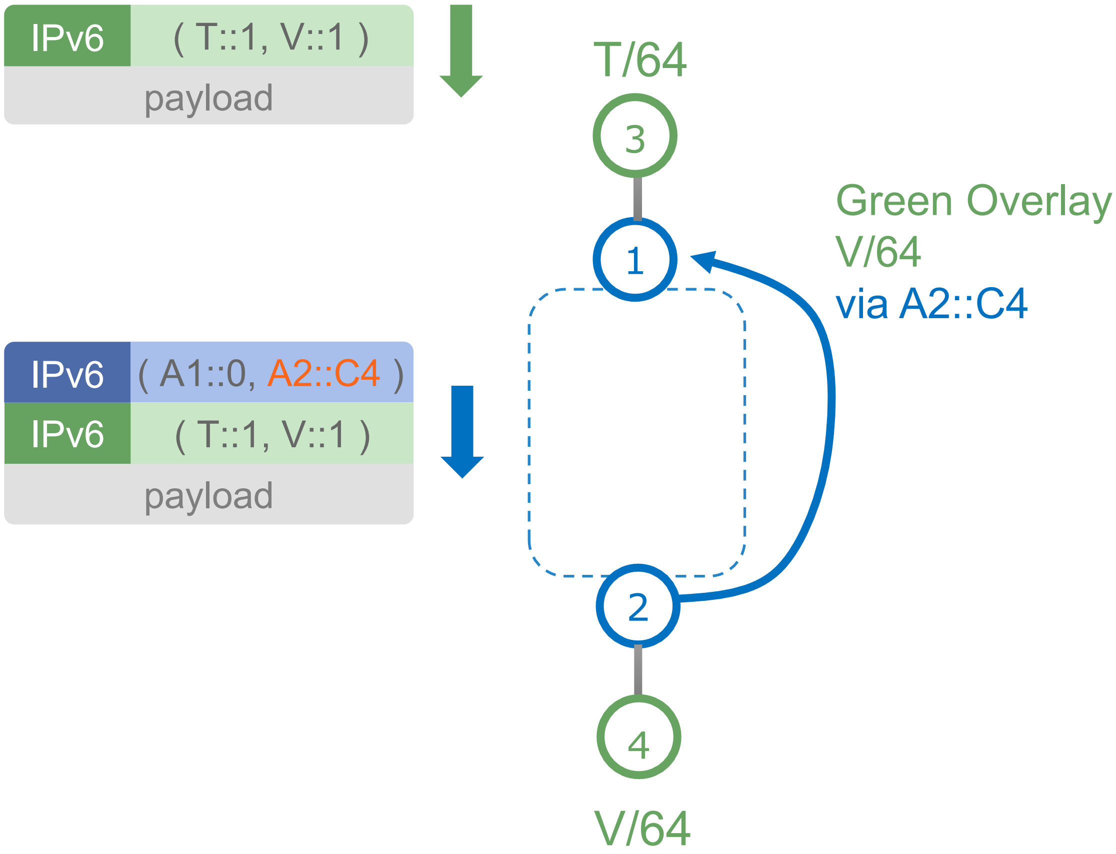


- 50msec Protection upon local link, node or SRLG failure
- Simple to operate and understand
  - automatically computed by the router's IGP process
  - 100% coverage across any topology
  - predictable (backup = postconvergence)
- Optimum backup path
  - leverages the post-convergence path, planned to carry the traffic
  - avoid any intermediate flap via alternate path
- Incremental deployment
- Distributed and Automated Intelligence

Overlay



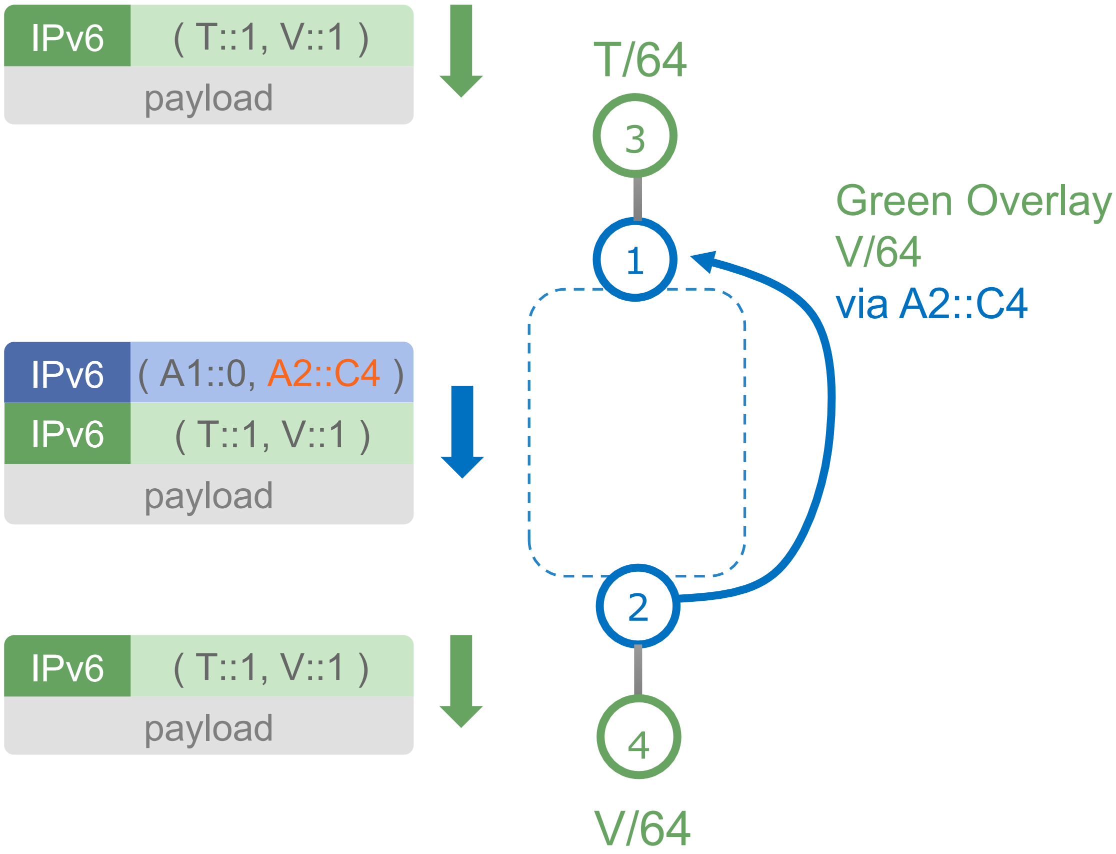
Overlay



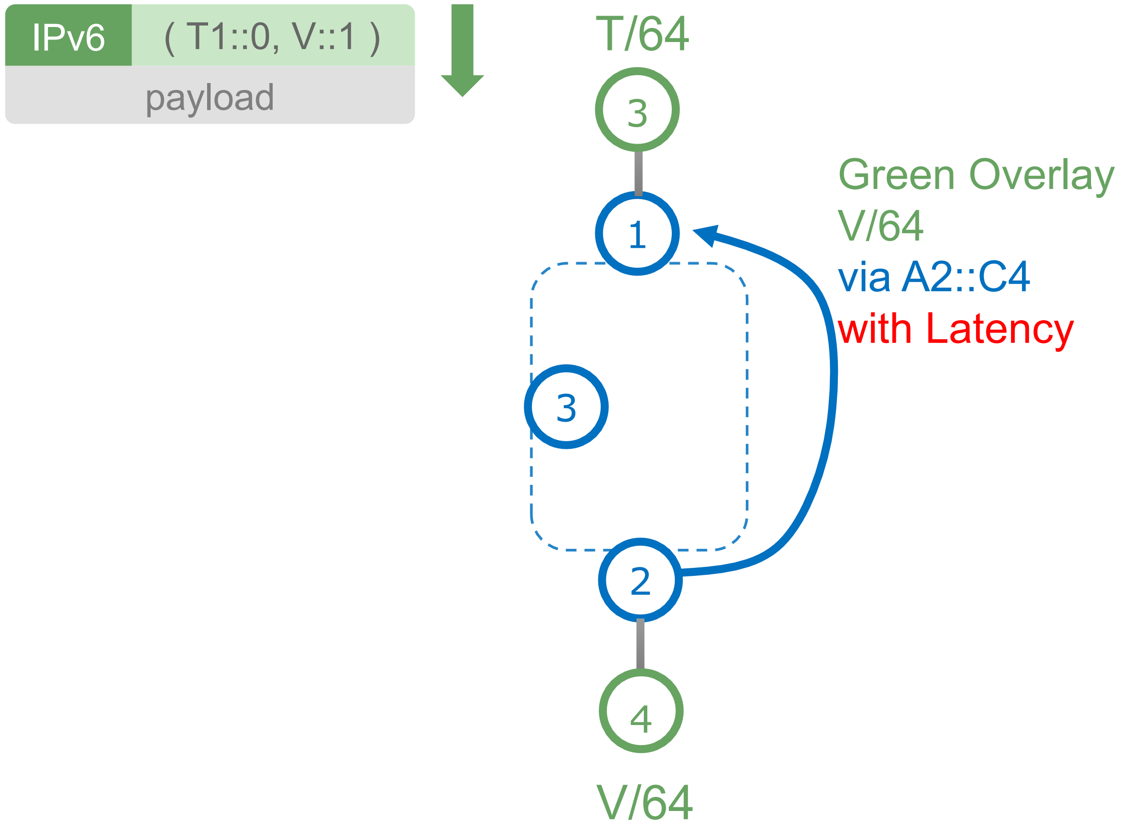
IPv6
(T::1, V::1)
payload
IPv6
(A1::0, A2::C4)
IPv6
(T::1, V::1)
payload
T/64
3
1
Green Overlay
V/64
via A2::C4
2
4
V/64

Overlay

- Simple
  - Protocol elimination
- Automated
  - No tunnel to configure
- Efficient
  - SRv6 for everything
  - Reuse BGP/VPN signaling



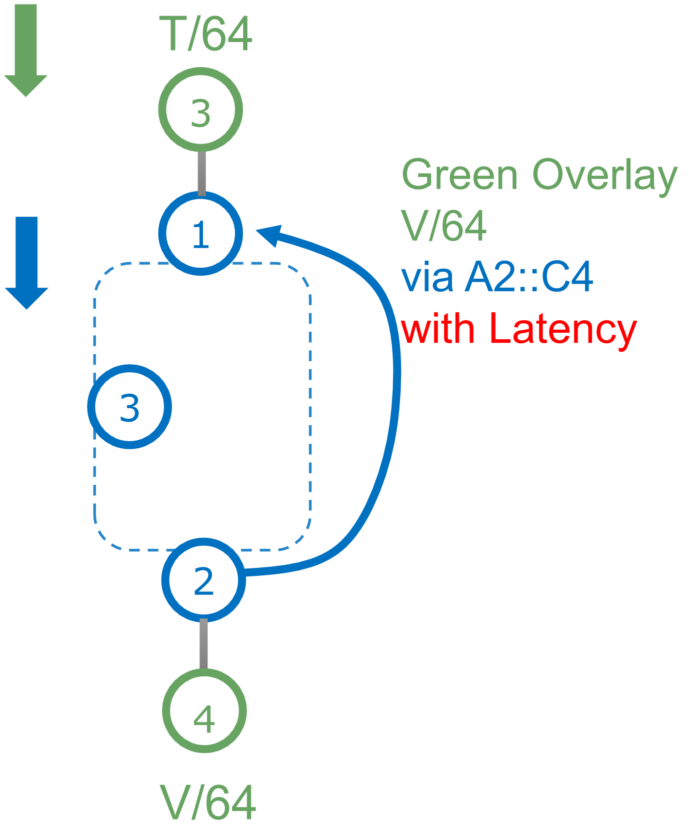
Overlay with Underlay SLA



Overlay with Underlay SLA

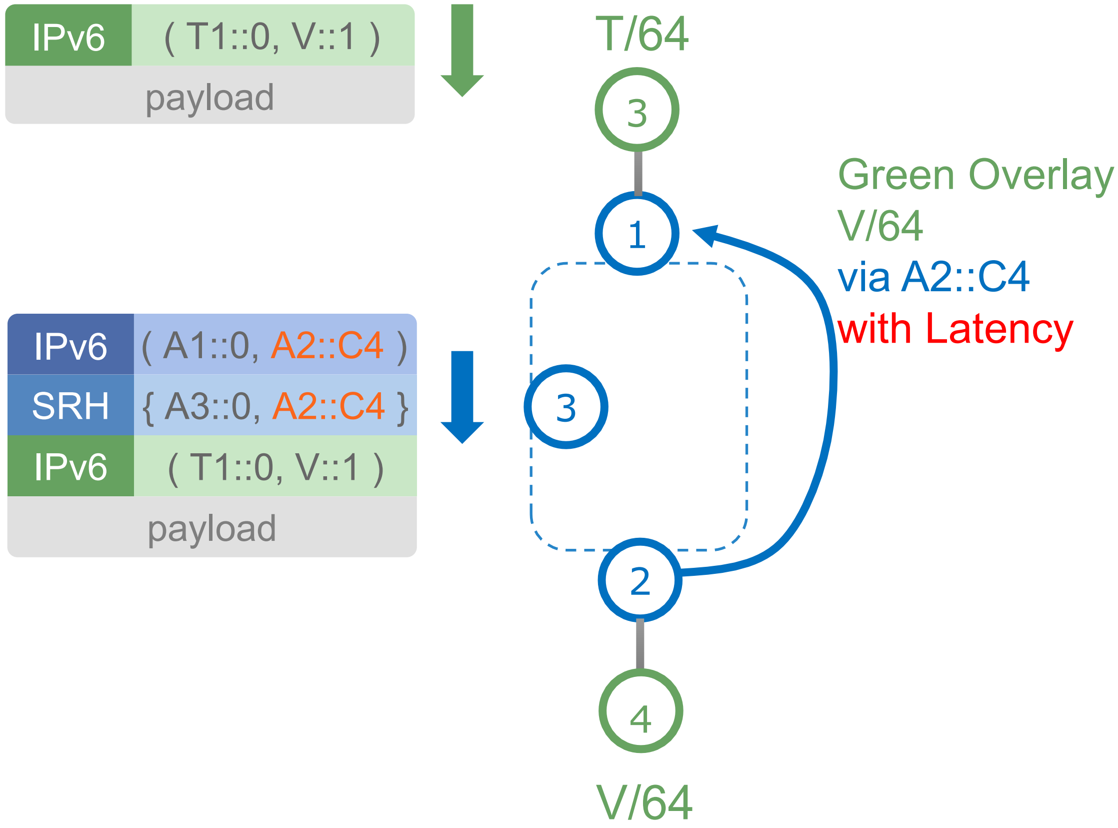
IPv6	(T1::0, V::1)
payload	

IPv6	(A1::0, A3::0)
SRH	{ A3::0, A2::C4 }
IPv6	(T1::0, V::1)
payload	



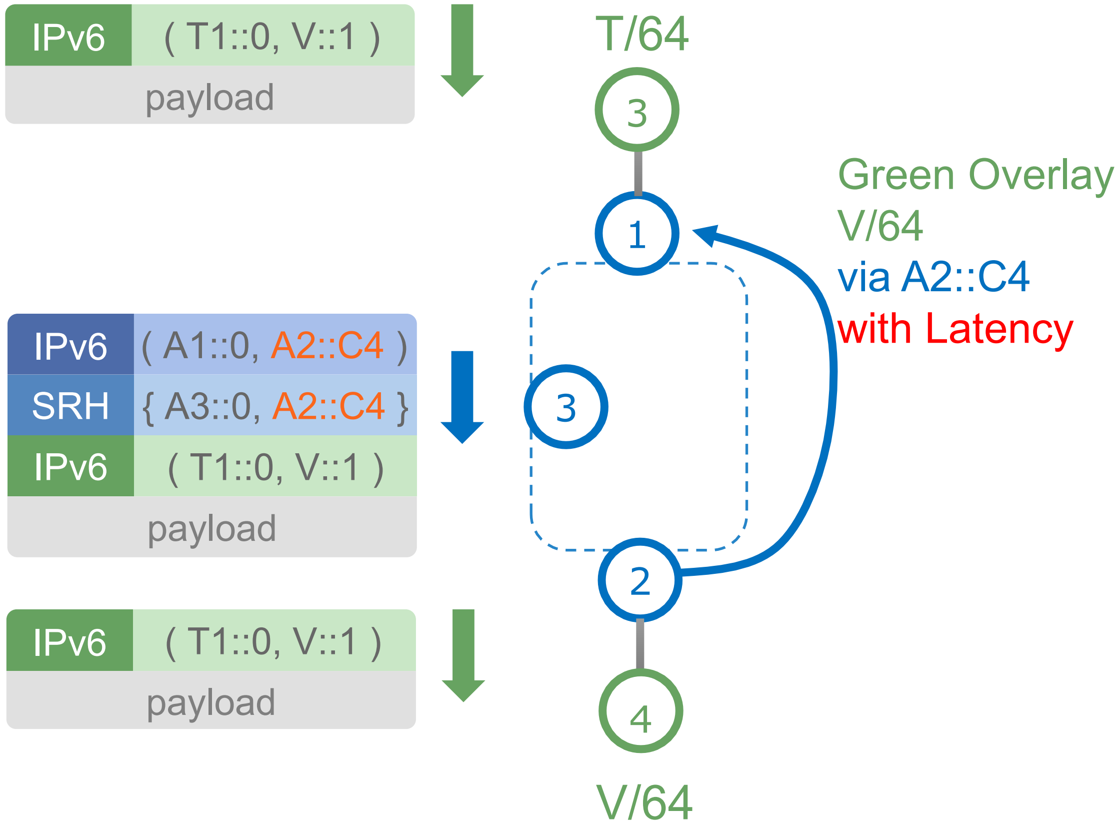
![Cisco logo]

Overlay with Underlay SLA

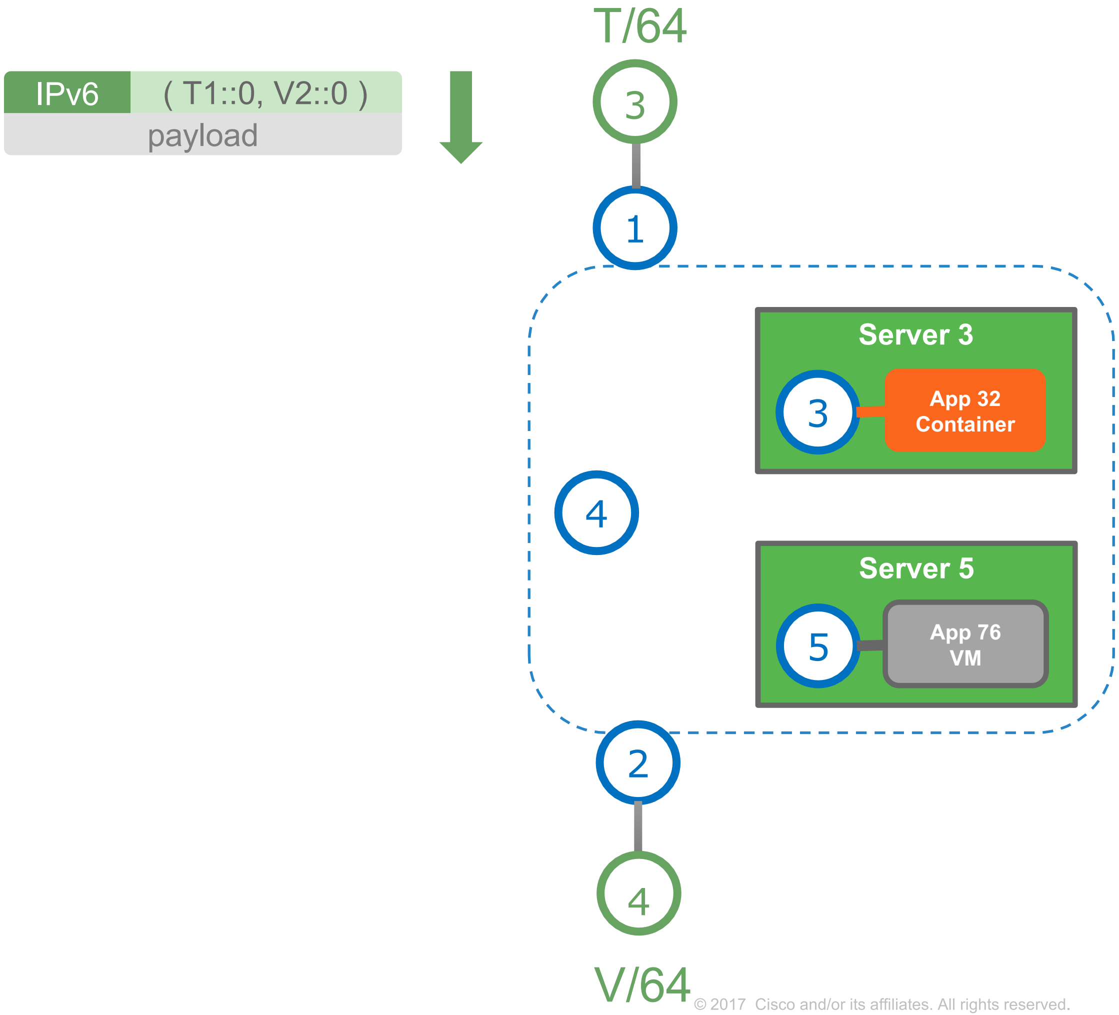


Overlay with Underlay SLA

- SRv6 does not only eliminate unneeded overlay protocols
- SRv6 solves problems that these protocols cannot solve
- Also support IPv4 and Ethernet VPN’s



Integrated NFV

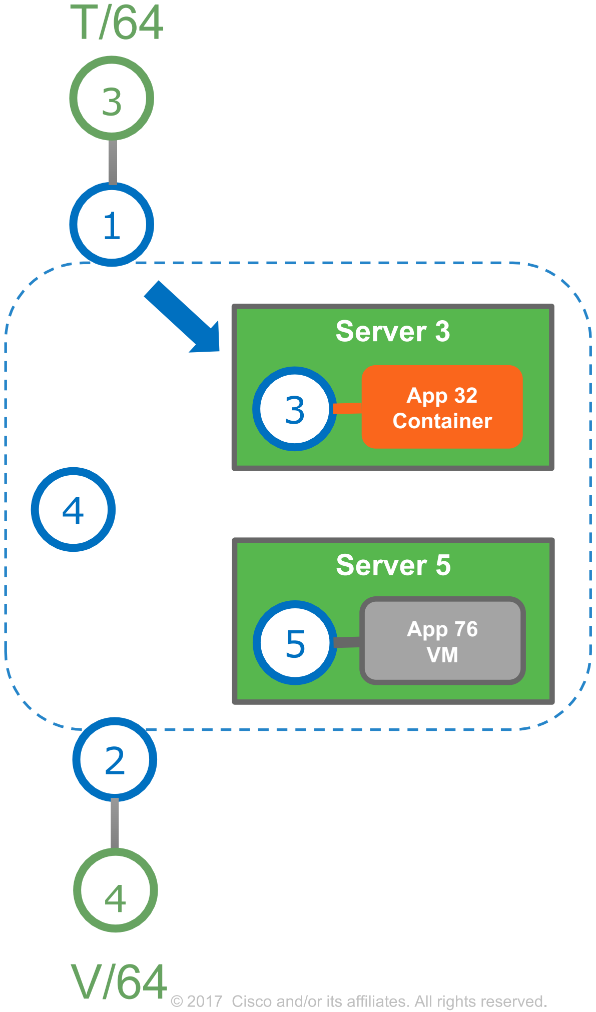


Integrated NFV

IPv6	(T1::0, V2::0)
payload	

- Stateless Service Chaining
  - NSH creates per-chain state in the fabric
  - SR does not
- App is SR aware or not
- App can work on IPv6 or IPv4 inner packets

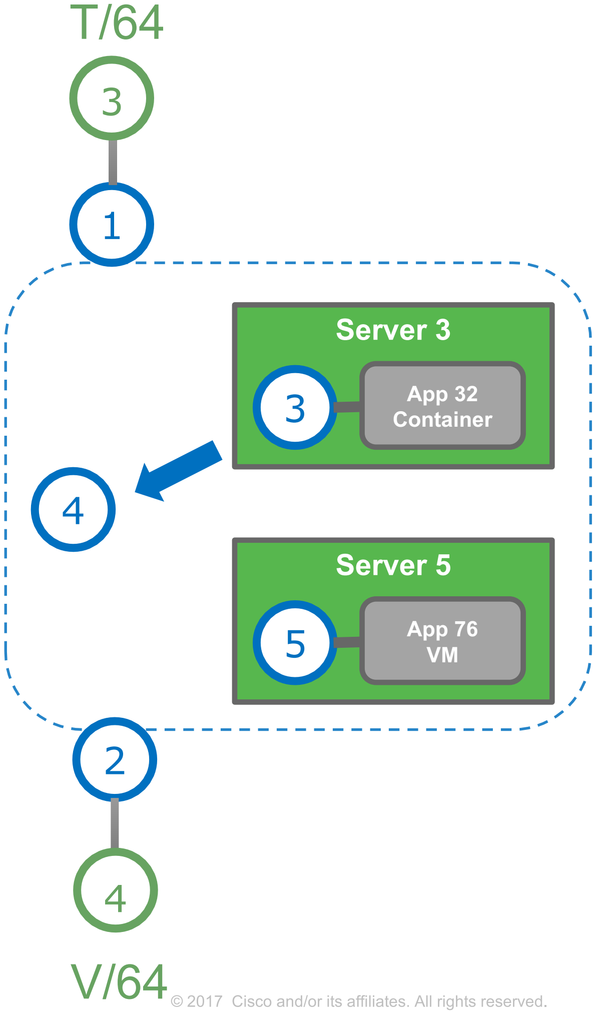
IPv6	(A1::0, A3::A32)
SRH	{ A3::A32, A4::0, A5::A76, A2::C4 }
IPv6	(T1::0, V2::0)
payload	



Integrated NFV

- Integrated SLA

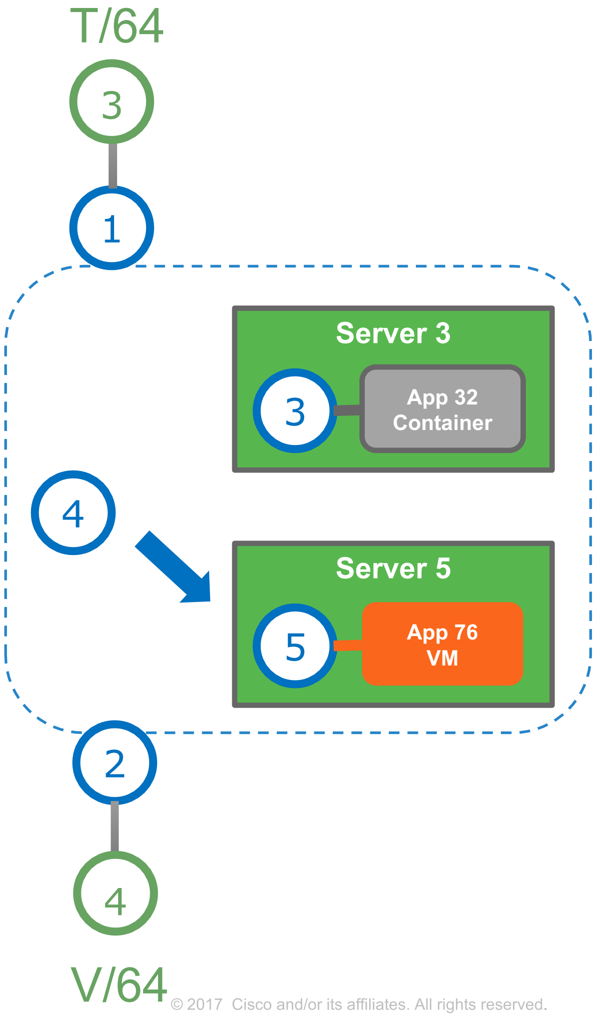
IPv6	(A1::0, A4::0)
SRH	{ A3::A32, A4::0, A5::A76, A2::C4 }
IPv6	(T1::0, V2::0)
payload	



Integrated NFV

- Stateless Service Chaining
  - NSH creates per-chain state in the fabric
  - SR does not
- App is SR aware or not
- App can work on IPv6 or IPv4 inner packets

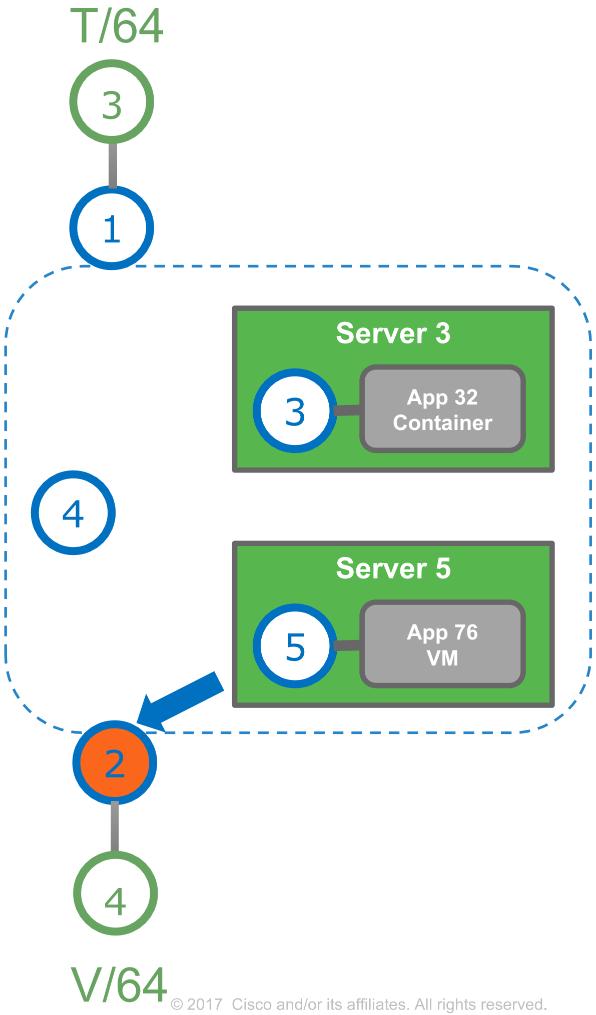
IPv6	(A1::0, A5::A76)
SRH	{ A3::A32, A4::0, A5::A76, A2::C4 }
IPv6	(T1::0, V2::0)
payload	



Integrated NFV

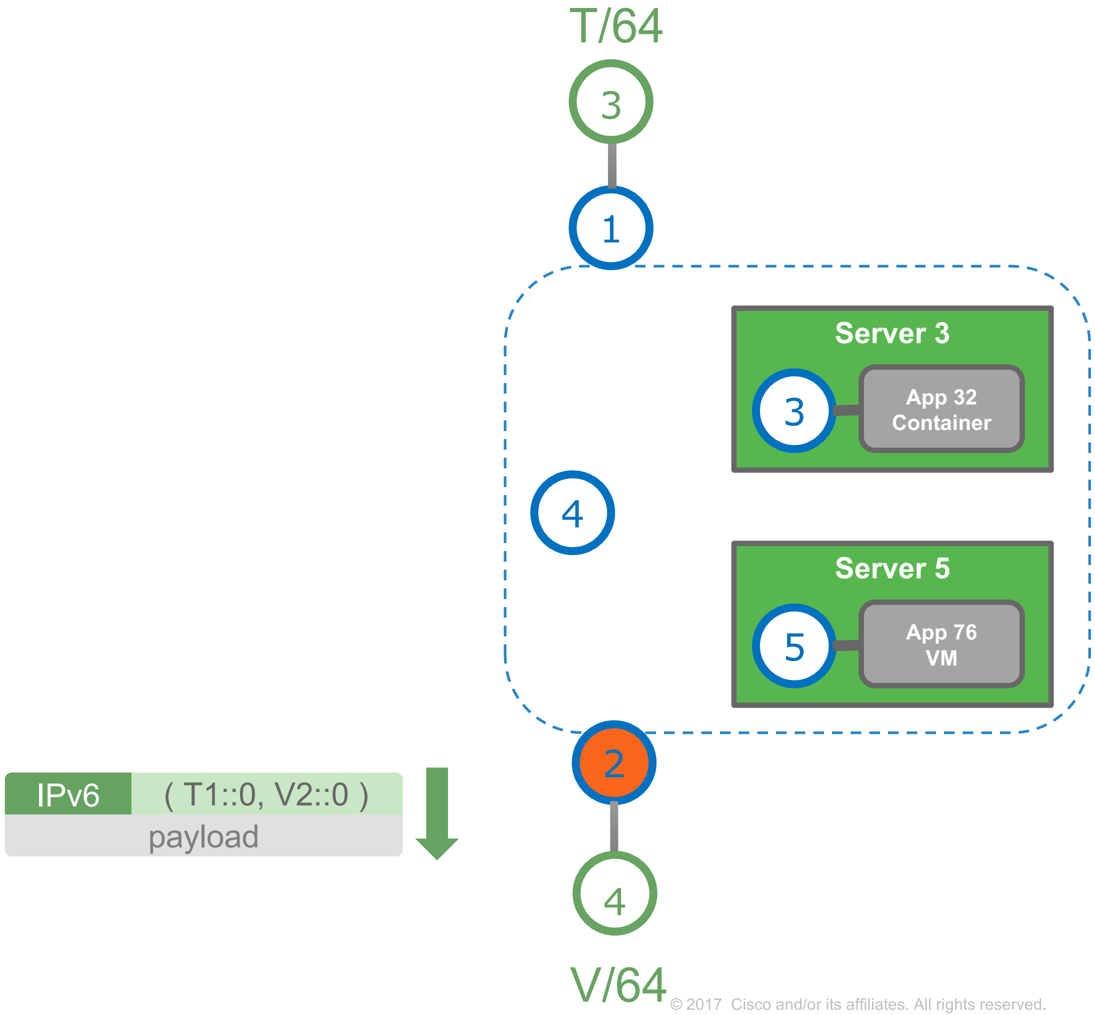
- Integrated with Overlay

IPv6	(A1::0, A2::C4)
SRH	{ A3::A32, A4::0, A5::A76, A2::C4 }
IPv6	(T1::0, V2::0)
payload	



Integrated NFV

- Integrated with Overlay



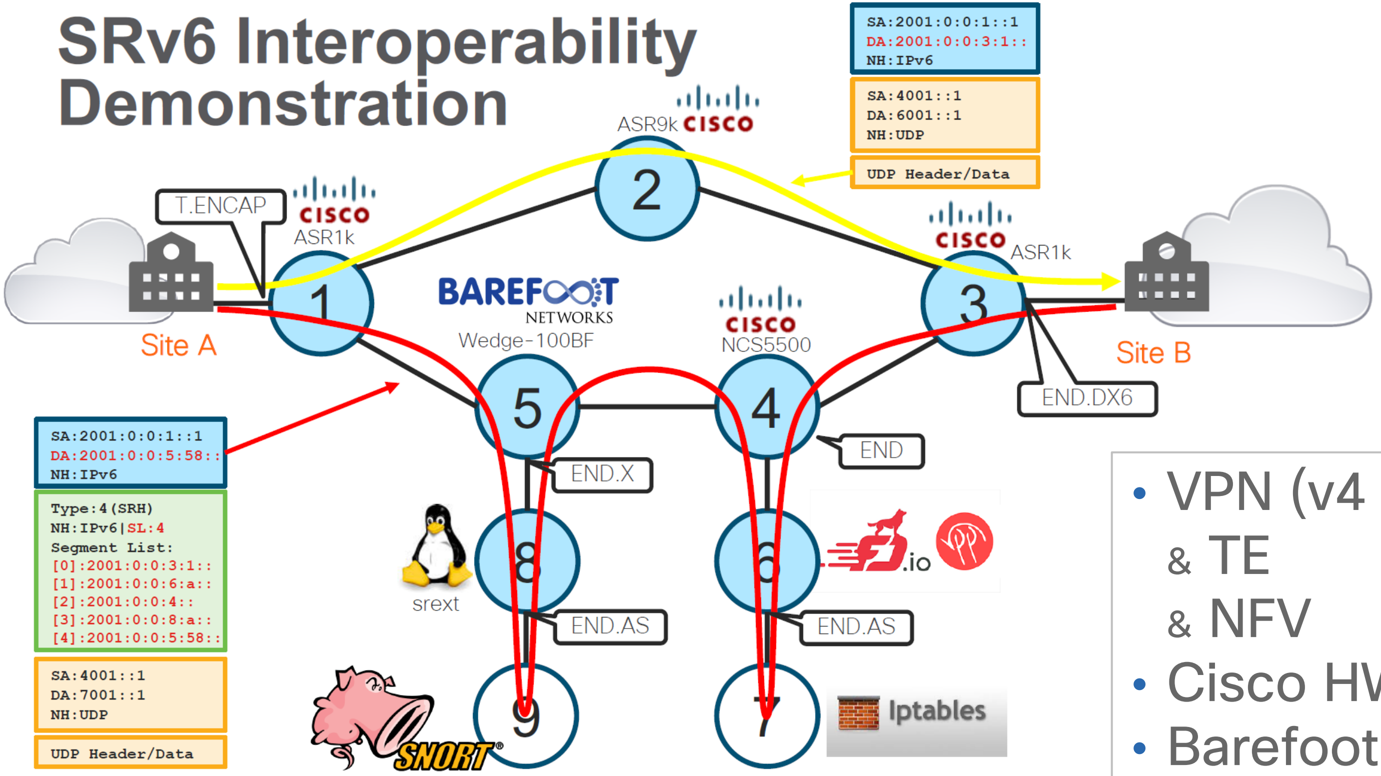
More use-cases

- 6CN: enhancing IP to search for Content
- 6LB: enhancing load-balancers
  - Better flow stickiness and load distribution
- Video Pipeline
- 5G Slicing
- 5G Ultra-Low Latency

Implementations

- Cisco HW
  - NCS5k - XR
  - ASR9k - XR
  - ASR1k - XE
- Open-Source
  - Linux 4.10
  - FD.IO
- Barefoot HW

SRv6 Interoperability Demonstration



- VPN (v4 and v6)
  & TE
  & NFV
- Cisco HW with XR and XE
- Barefoot HW with P4 code
- FD.IO
- Linux

blogs.cisco.com/sp/segment-routing-ipv6-interoperability-demo-is-already-there

Deployments

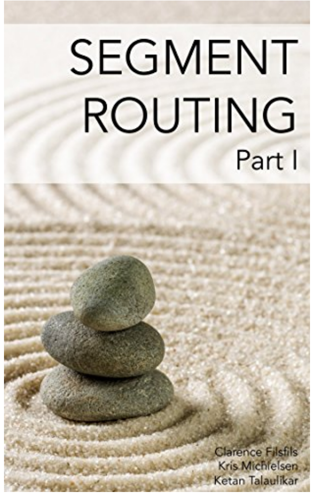
- Comcast
  - Efficient transport of video traffic
- Others

Conclusion

Segment Routing

- Strong industry support
- Fantastic deployment rate
- Bold architecture: network programming
- Numerous use-cases
- Feel free to join the lead-operator team!

Stay Up-To-Date



amzn.com/B01I58LSUO

segment-routing.net

linkedin.com/groups/8266623

twitter.com/SegmentRouting

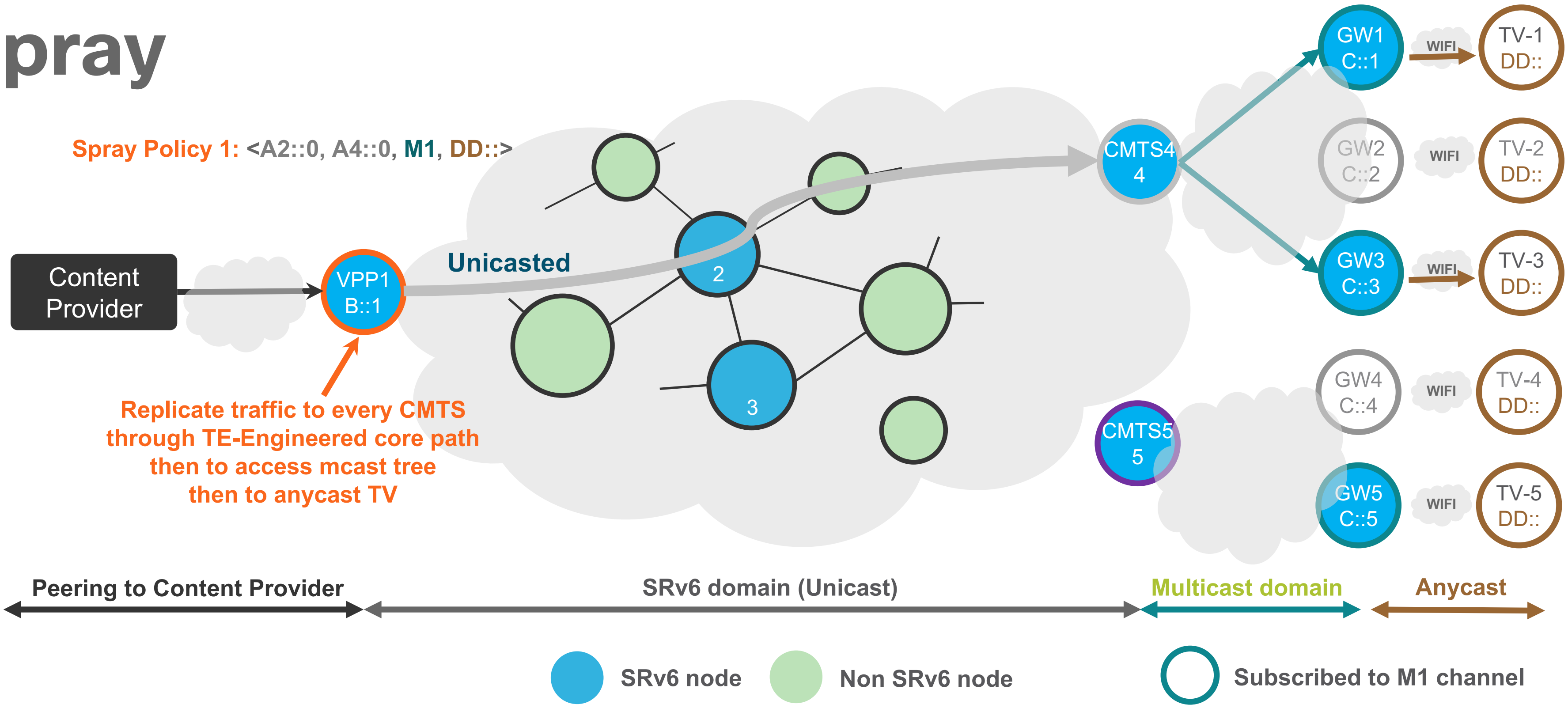
facebook.com/SegmentRouting/

Thank you

Appendix

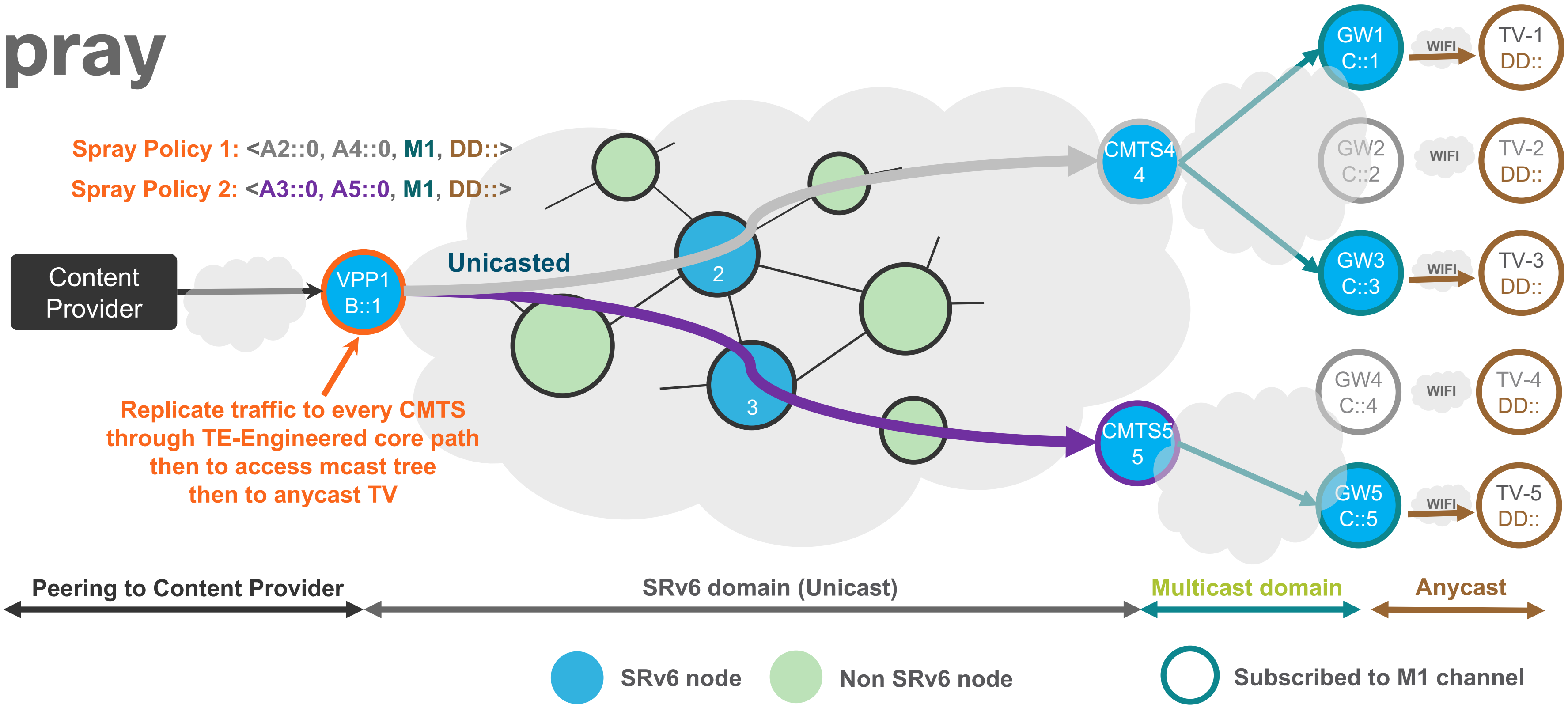
Other use-cases

Spray



Flexible, SLA-enabled and Efficient content injection without multicast core

Spray

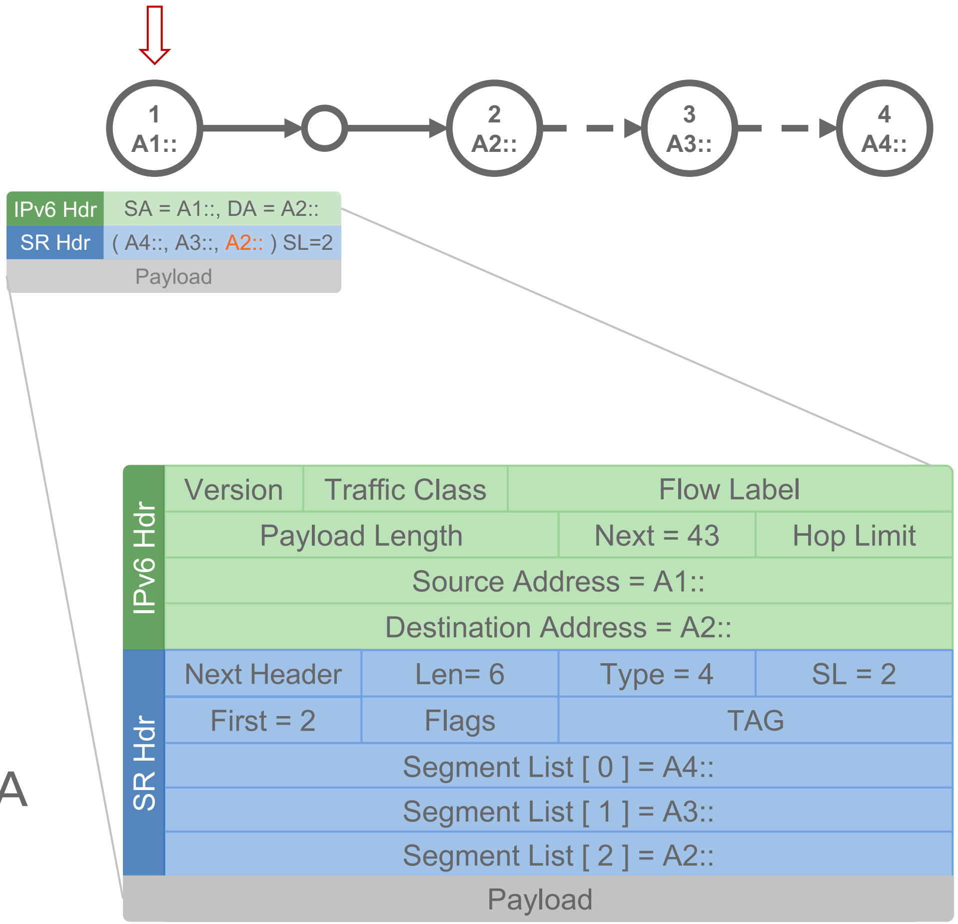


Flexible, SLA-enabled and Efficient content injection without multicast core

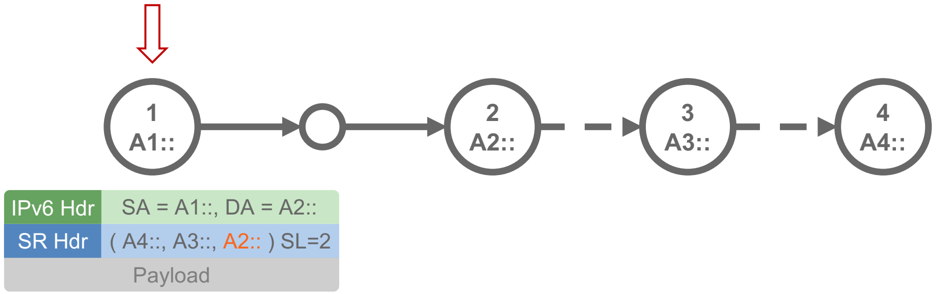
SRH Processing

Source Node

- Source node is SR-capable
- SR Header (SRH) is created with
  - Segment list in reversed order of the path
    - > Segment List [0] is the LAST segment
    - > Segment List [$n-1$] is the FIRST segment
  - Segments Left is set to $n-1$
  - First Segment is set to $n-1$
- IP DA is set to the first segment
- Packet is send according to the IP DA
  - Normal IPv6 forwarding



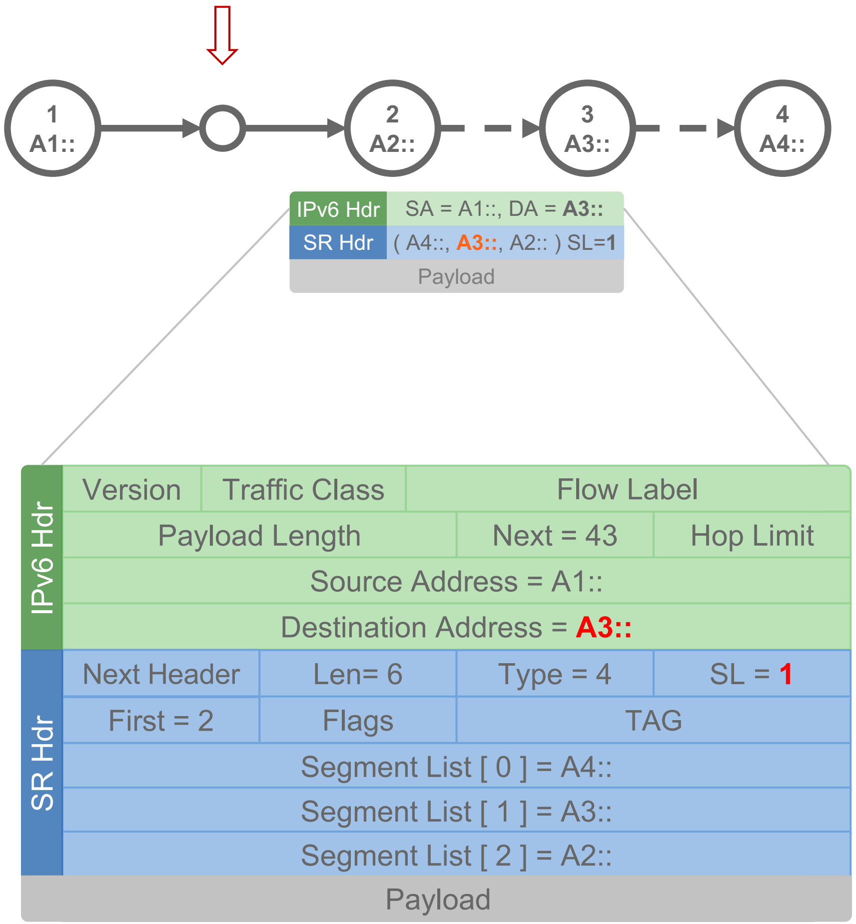
Non-SR Transit Node



- Plain IPv6 forwarding
- Solely based on IPv6 DA
- No SRH inspection or update

SR Segment Endpoints

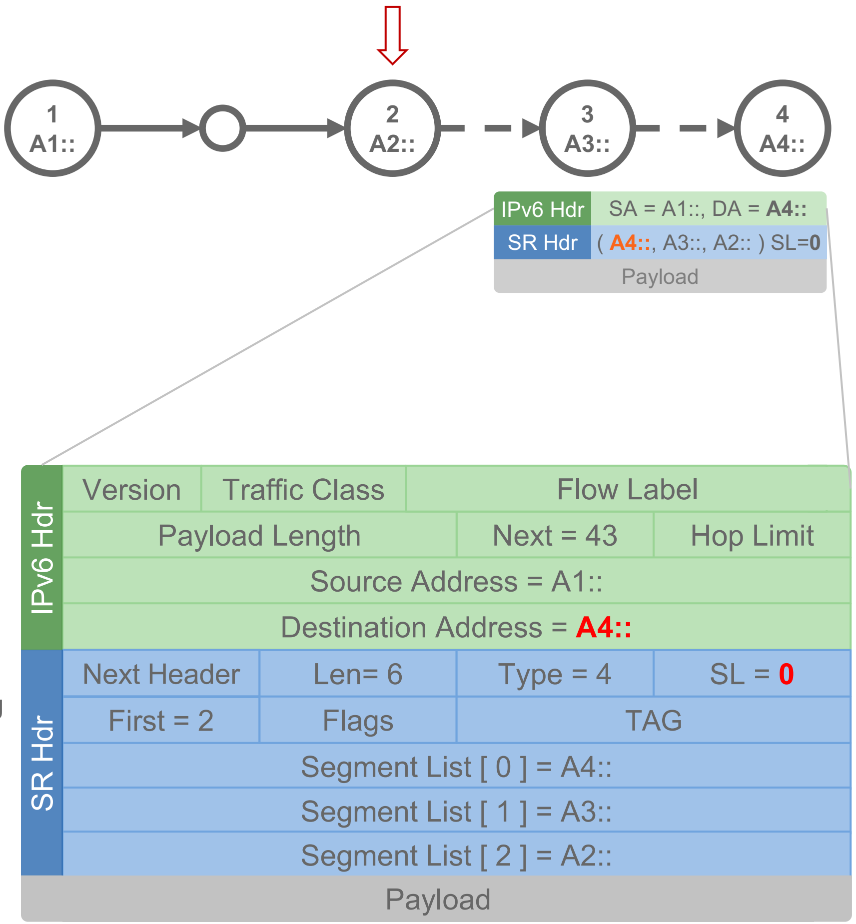
- SR Endpoints: SR-capable nodes whose address is in the IP DA
- SR Endpoints inspect the SRH and do:
  - IF Segments Left > 0, THEN
    - > Decrement Segments Left (-1)
    - > Update DA with Segment List [Segments Left]
    - > Forward according to the new IP DA



SR Segment Endpoints

- SR Endpoints: SR-capable nodes whose address is in the IP DA
- SR Endpoints inspect the SRH and do:
  - IF Segments Left > 0, THEN
    > Decrement Segments Left (-1)
    > Update DA with Segment List [Segments Left]
    > Forward according to the new IP DA
  - ELSE (Segments Left = 0)
    > Remove the IP and SR header
    > Process the payload:
    - Inner IP: Lookup DA and forward
    - TCP / UDP: Send to socket
    - …

Standard IPv6 processing
The final destination does not have to be SR-capable.



Some obvious SID functions

Segment format

Locator | *Function*
1111 : 2222 : 3333 : 4444 : 5555 : 6666 : 7777 : 8888

- SRv6 SIDs are 128-bit addresses
  - **Locator**: most significant bits are used to **route** the segment to its **parent node**
  - **Function**: least significant bits identify the **action** to be performed on the **parent node**
    - **Argument** [optional]: Last bits can be used as a local function argument
- Flexible bit-length allocation
  - Segment format is **local knowledge** on the parent node
- SIDs have to be **specifically enabled** as such on their parent node
  - A local address **is not** by default a local SID
  - A local SID does not have to be associated with an interface

Segment format

Locator				*Function*			
1111	2222	3333	4444	5555	6666	7777	8888

- SRv6 SIDs are 128-bit addresses
  - **Locator**: most significant bits are used to **route** the segment to its **parent node**
  - **Function**: least significant bits identify the **action** to be performed on the **parent node**
    - **Argument** [optional]: Last bits can be used as a local function argument
- Flexible bit-length allocation
  - Segment format is **local knowledge** on the parent node
- SIDs have to be **specifically enabled** as such on their parent node
  - A local address **is not** by default a local SID
  - A local SID does not have to be associated with an interface

Segment format

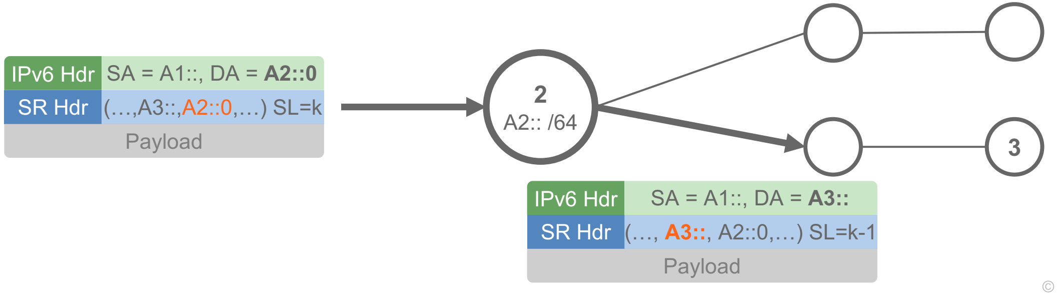
Locator *Function*

1111 : 2222 : 3333 : 4444 : 5555 : 6666 : 7777 : 8888

- SRv6 SIDs are 128-bit addresses
  - **Locator**: most significant bits are used to **route** the segment to its **parent node**
  - **Function**: least significant bits identify the **action** to be performed on the **parent node**
    - **Argument** [optional]: Last bits can be used as a local function argument
- Flexible bit-length allocation
  - Segment format is **local knowledge** on the parent node
- SIDs have to be **specifically enabled** as such on their parent node
  - A local address **is not** by default a local SID
  - A local SID does not have to be associated with an interface

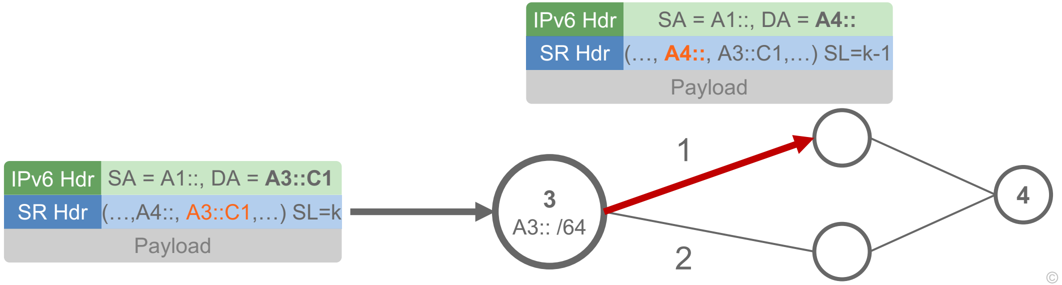
END – Default endpoint

- *Default endpoint* behavior (node segment)
  - Decrement Segments Left, update DA
  - Forward according to new DA
- Node 2 advertises prefix A2::/64 (A2::/64 is the SID **locator**)
  - Packets are forwarded to node 2 along the default routes (shortest path)
- On 2, the *default endpoint* behavior is associated with ID 0 (0 is the **function**)
- The SID corresponding to the *default endpoint* behavior on node 2 is A2::0



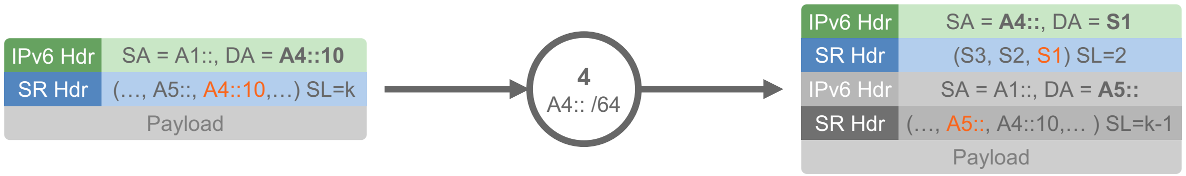
END.X – Endpoint then Xconnect

- *Endpoint xconnect* behavior (adjacency segment)
  - Decrement Segments Left, update DA
  - **Forward on the interface associated with the Xconnect segment**
- Node 3 advertises prefix A3::/64
  - Packets are forwarded to node 3 along the default routes (shortest path)
- On 3, the *endpoint xconnect* behavior for link 1 is associated with ID C1
- The SID corresponding to *endpoint xconnect-1* behavior on node 3 is **A3::C1**



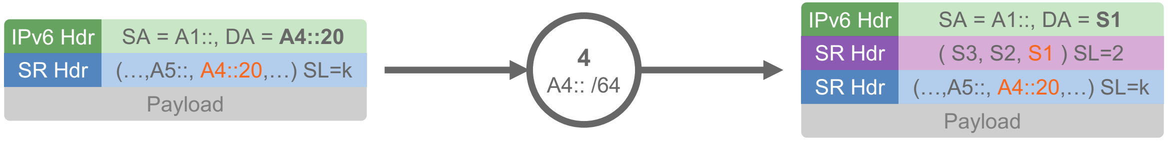
END.B6.ENCAPS – IPv6 Binding Segment (encap)

- *IPv6 binding segment*
  - Decrement Segments Left, update DA
  - **Push outer IP and SR headers associated with the binding segment**
  - Forward according to outer header DA (first segment of the new SRH)
- Node 4 advertises prefix A4::/64
- The SR *encaps* policy (SA = A4::, SL = 〈 S1, S2, S3 〉) is associated with ID 10
- The corresponding *binding SID* is **A4::10**



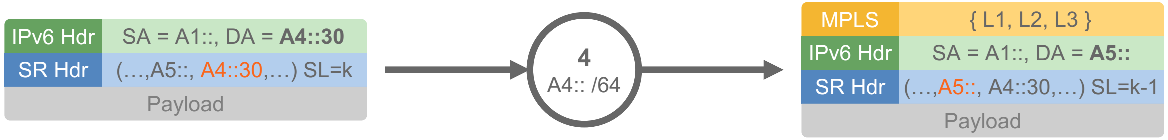
END.B6 – IPv6 Binding Segment (insert)

- *IPv6 binding segment*
  - **Do not** decrement Segments Left
  - **Push outer SR header associated with the binding segment**
  - **Update DA with the first segment of the outer SR header**
  - Forward according to outer header DA (first segment of the new SRH)
- Node 4 advertises prefix A4::/64
- On 4, the SR *insert* policy 〈 S1, S2, S3 〉 is associated with ID 20
- The corresponding *binding SID* is **A4::20**



END.BM – MPLS Binding Segment

- *MPLS binding segment*
  - Decrement Segments Left
  - **Push outer MPLS label stack associated with the binding segment**
  - Forward according to the top MPLS label
- Node 4 advertises prefix A4::/64
- On 4, the MPLS SR policy <L1, L2, L3> is associated with ID 30
- The corresponding *binding SID* is **A4::30**

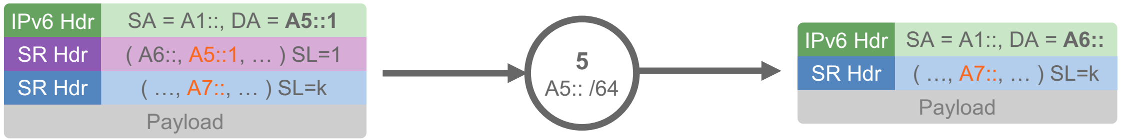


END.PSP – Penultimate Segment Popping

- *Penultimate Segment Popping (PSP)* behavior
  - Decrement Segments Left, update DA
  - **If Segments Left = 0, remove SRH**
  - Forward according to new DA
- Node 5 advertises prefix A5::/64
- On 5, the *Penultimate Segment Popping* behavior is associated with ID 1
- The corresponding SID is A5::1

END.PSP – Penultimate Segment Popping

- *Penultimate Segment Popping (PSP)* behavior
  - Decrement Segments Left, update DA
  - **If Segments Left = 0, remove SRH**
  - Forward according to new DA
- Node 5 advertises prefix A5::/64
- On 5, the *Penultimate Segment Popping* behavior is associated with ID 1
- The corresponding SID is A5::1



END.USP – Ultimate Segment Popping

- *Ultimate Segment Popping (USP)* behavior
  - **If Segments Left = 0**
    - **Pop the top SRH**
    - **Restart the END function processing on the modified packet**
      - Decrement Segments Left, update DA
      - Forward according to new DA
- Node 6 advertises prefix A6::/64
- A6:: is the last segment in the top SRH

SID Function – Anything

- SID functions are locally defined on their parent node
  - They can do anything...
- An SR header contains a network program